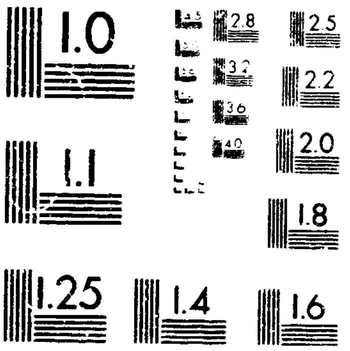
1 OF 1

N92-32422 UNCLAS



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

NASA Technical Memorandum 102851

# Potential Flow Theory and Operation Guide for the Panel Code PMARC

Dale L. Ashby, Michael R. Dudley, Steve K. Iguchi,
Lindsey Browne, and Joseph Katz

January 1991

NASA
National Aeronautics and
Space Administration

NASA Technical Memorandum 102851

# Potential Flow Theory and Operation Guide for the Panel Code PMARC

Dale L. Ashby, Michael R. Dudley, Steve K. Iguchi, Lindsey Browne, and Joseph Katz
Ames Research Center, Moffett Field, California

January 1991

NASA
National Aeronautics and
Space Administration

**Ames Research Center**
Moffett Field, California 94035-1000

# NOMENCLATURE

| | |
|---|---|
| $B_{JK}$ | velocity potential influence coefficient at control point of panel J due to a uniform distribution of unit source on panel K |
| $C_{JK}$ | velocity potential influence coefficient at control point of panel J due to a uniform distribution of unit doublet on panel K |
| $C_P$ | pressure coefficient |
| $dS$ | differential surface element on configuration |
| $\bar{n}$ | unit normal vector to surface |
| $N_s$ | total number of surface panels |
| $N_w$ | total number of wake panels |
| $P$ | an arbitrary point in space |
| $\bar{r}$ | vector between an arbitrary point P and a surface element dS |
| $S$ | surface of the configuration |
| $S_\infty$ | imaginary surface at infinity |
| $t$ | time |
| $\bar{V}$ | velocity vector |
| $\bar{V}_{\mu PK}$ | velocity influence coefficient at point P due to a uniform distribution of unit doublet on panel K |
| $\bar{V}_{\sigma PK}$ | velocity influence coefficient at point P due to a uniform distribution of unit source on panel K |
| $W$ | wake surface |
| $\Phi$ | total velocity potential |
| $\phi$ | perturbation velocity potential |
| $\phi_\infty$ | free-stream velocity potential |

| | |
|---|---|
| $\mu$ | doublet singularity strength per unit area |
| $\sigma$ | source singularity strength per unit area |

**Subscripts**

| | |
|---|---|
| $i$ | interior region |
| $J$ | refers to panel J or its control point |
| $K$ | refers to panel K or its control point |
| $L$ | lower surface |
| $P$ | refers to velocity scan point P |
| $U$ | upper surface |
| $\infty$ | free-stream conditions |

# POTENTIAL FLOW THEORY AND OPERATION GUIDE FOR THE PANEL CODE PMARC

Dale L. Ashby, Michael R. Dudley, Steve K. Iguchi,* Lindsey Browne,* and Joseph Katz*
Ames Research Center

## SUMMARY

The theoretical basis for PMARC, a low-order potential-flow panel code for modeling complex three-dimensional geometries, is outlined. Several of the advanced features currently included in the code, such as internal flow modeling, a simple jet model, and a time-stepping wake model, are discussed in some detail. The code is written using adjustable size arrays so that it can be easily redimensioned for the size problem being solved and the computer hardware being used. An overview of the program input is presented,with a detailed description of the input available in the appendices. Finally, PMARC results for a generic wing/body configuration are compared with experimental data to demonstrate the accuracy of the code. The input file for this test case is given in the appendices.

## INTRODUCTION

A potential flow panel code, called PMARC (Panel Method Ames Research Center), is currently being developed at NASA Ames Research Center to numerically predict flow fields around complex three-dimensional geometries (ref. 1). The creation of PMARC was prompted by the need at Ames for a well-documented code, suitable for powered-lift aerodynamic predictions, with an open architecture which would facilitate making modifications or adding new features. An open code will allow other government agencies and contractors to make additional contributions to the code. A second objective in the development of PMARC was to create an adjustable-size panel code. This allows PMARC to be tailored so an optimum match can be achieved between the computer hardware available to the user and the size of the problem being solved. Currently PMARC can be resized (i.e., the maximum number of panels can be changed) in a matter of minutes. PMARC can be run on computers ranging from a MacIntosh II workstation to a Cray Y-MP. At its present state of development, PMARC contains several features considered to be state-of-the-art for panel methods. These include internal flow modeling for ducts and wind tunnel test sections, simple jet plume modeling essential for the analysis and design of powered-lift aircraft, and a time-stepping wake model which allows the study of both steady and unsteady motions. PMARC is a research tool that is envisioned as being in a continual state of development. Existing routines will be improved or replaced by new routines, and new features and options added as they become available.

One of the decisions that had to be made in the development of PMARC was the type of panel method to be used. Panel methods can be subdivided into two groups: low order and high order. In a

*San Diego State University, San Diego, California.

low-order panel method, the singularities are distributed with constant strength over each panel. In a higher-order method, the singularity strengths are allowed to vary linearly or quadratically over each panel. Higher-order panel methods claim a better accuracy in the modeling of the flow field, but this is at the expense of increased code complexity and computation time. Experience with panel methods such as PANAIR, MACAERO, VSAERO, and QUADPAN, developed under NASA contracts and/or industry internal research and development, has shown that low-order methods can provide nearly the same accuracy as higher order methods over a wide range of cases; however, the computation time for low-order panel methods is much shorter than for higher-order panel methods (ref. 2). Additionally, low-order panel methods do not require exact matching between panels as higher-order methods do. For these reasons, and to reduce program cost and complexity, the low-order panel method was chosen for the basic methodology. The code is structured to accommodate the addition of a higher-order solution method at a later date, if warranted.

To avoid unnecessary duplication of previous work, existing software was utilized whenever possible to reduce development time of PMARC and cost to the government. Of the several low-order panel methods available, the 1000-panel version of VSAERO was felt to be the most robust, mature, and widely accepted by the engineering community. During its ten years of development by Analytical Methods Inc., which was supported largely by government and industry contracts, VSAERO has demonstrated that low-order panel methods are a viable means of predicting aerodynamic flows about complex shapes. Two versions of VSAERO were delivered to Ames Research Center, one in 1982 and the other in 1985 under contracts NAS2-11169 and NAS2-11944, respectively (refs. 3-5). The basic potential flow computational methods and techniques used in PMARC were patterned after the most recent 1000-panel version of VSAERO, which is currently available through Cosmic.

## THEORY

### Potential Flow Model

In PMARC, the flow field around a three-dimensional body is assumed to be inviscid, irrotational, and incompressible. The body is modeled as a closed surface which divides space into two regions as shown in figure 1. One region contains the flow field of interest and the other contains a fictitious flow. Figure 1 shows the external region as the flow field of interest and the internal flow as the fictitious flow. This is the typical arrangement for external flow problems such as a wing in a uniform stream. This arrangement is reversed for internal flow problems. The internal region contains the flow field of interest and the external flow field is fictitious. In either case it is assumed that the velocity potentials in both regions satisfy Laplace's equation:

$$\nabla^2\Phi = 0$$
$$\nabla^2\Phi_i = 0 \qquad (1)$$

The potential at any point P in either region may be evaluated by applying Green's Theorem to both regions. This results in the following integral equation:

$$\Phi_P = \frac{1}{4\pi} \iint\limits_{S+W+S_\infty} (\Phi - \Phi_i)\ \bar{n} \cdot \nabla\left(\frac{1}{\bar{r}}\right) dS \ - \ \frac{1}{4\pi} \iint\limits_{S+W+S_\infty} \left(\frac{1}{\bar{r}}\right)\ \bar{n} \cdot (\nabla\Phi - \nabla\Phi_i)\ dS \qquad (2)$$

where $\bar{r}$ is the distance from the point P to the element dS on the surface and $\bar{n}$ is the unit normal vector to the surface pointing into the flow field of interest. In this equation the first integral represents the disturbance potential from a surface distribution of doublets with strength $(\Phi - \Phi_i)$ per unit area and the second integral represents the contribution from a surface distribution of sources with strength $-\bar{n} \cdot (\nabla\Phi - \nabla\Phi_i)$ per unit area. This equation may be simplified by noting that at the surface at infinity, the perturbation potential due to the configuration is essentially zero, leaving only the potential due to the uniform onset flow. It is assumed that the wake is thin and there is no entrainment, so the source term for the wake disappears and the jump in normal velocity across the wake is zero. Hence the simplified equation becomes:

$$\Phi_P = \frac{1}{4\pi} \iint\limits_{S} (\Phi - \Phi_i)\ \bar{n} \cdot \nabla\left(\frac{1}{\bar{r}}\right) dS \ - \ \frac{1}{4\pi} \iint\limits_{S} \left(\frac{1}{\bar{r}}\right)\ \bar{n} \cdot (\nabla\Phi - \nabla\Phi_i)\ dS$$

$$+ \ \frac{1}{4\pi} \iint\limits_{W} (\Phi_U - \Phi_L)\ \bar{n} \cdot \nabla\left(\frac{1}{\bar{r}}\right) dS \ + \ \phi_{\infty P} \qquad (3)$$

The point P must be excluded from the integration if it lies on the surface, since the integrals become singular in that case at point P. This is done by assuming a hemispherical deformation of the surface centered at P. If the integral is evaluated for this hemispherical deformation as its radius is allowed to go to zero and point P (and hence the hemispherical deformation) are on the outside of the surface, the contribution at point P is $1/2(\Phi\text{-}\Phi_i)_P$. If point P lies on the inside of the surface, the contribution at point P is $-1/2(\Phi\text{-}\Phi_i)_P$. Hence for points P lying on the inside of the surface, equation (3) becomes:

$$\Phi_P = \frac{1}{4\pi} \iint\limits_{S-P} (\Phi - \Phi_i)\ \bar{n} \cdot \nabla\left(\frac{1}{\bar{r}}\right) dS \ - \ \frac{1}{4\pi} \iint\limits_{S} \left(\frac{1}{\bar{r}}\right)\ \bar{n} \cdot (\nabla\Phi - \nabla\Phi_i)\ dS$$

$$+ \ \frac{1}{4\pi} \iint\limits_{W} (\Phi_U - \Phi_L)\ \bar{n} \cdot \nabla\left(\frac{1}{\bar{r}}\right) dS \ + \ \phi_{\infty P} \ - \ \frac{1}{2}(\Phi - \Phi_i)_P \qquad (4)$$

The boundary condition used to solve equation (4) is an internal Dirichlet boundary condition. The total potential $\Phi$ can be viewed as being made up of an onset potential $\phi_\infty$ and a perturbation potential $\phi = \Phi - \phi_\infty$. The potential of the fictitious flow is set equal to the onset potential, $\phi_\infty$. With this boundary condition, the singularities on the surface tend to be smaller than if the potential of the fictitious flow is set to zero because the singularities only have to provide the perturbation potential instead of the total potential. Using this boundary condition and looking at points P inside the surface, equation (4) can be rewritten as:

$$0 = \frac{1}{4\pi}\iint\limits_{S-P} \phi \;\bar{n}\cdot\nabla\left(\frac{1}{\bar{r}}\right) dS \;-\; \frac{1}{4\pi}\iint\limits_{S}\left(\frac{1}{\bar{r}}\right)\;\bar{n}\cdot(\nabla\Phi - \nabla\phi_{\infty})\; dS$$

$$+\;\frac{1}{4\pi}\iint\limits_{W}(\Phi_U - \Phi_L)\;\bar{n}\cdot\nabla\left(\frac{1}{\bar{r}}\right) dS \;-\; \frac{1}{2}\,\phi_P \qquad (5)$$

Refering to the definitions made for equation (2), the following equations may be written for the doublet and source strengths:

$$4\,\pi\,\mu \;=\; \phi \;=\; (\Phi - \phi_{\infty}) \qquad (6)$$

$$4\,\pi\,\sigma \;=\; -\bar{n}\cdot(\nabla\Phi - \nabla\phi_{\infty}) \qquad (7)$$

Looking at equation (7), if it is assumed that the normal velocity at the surface is either zero or some known value, then the source strengths can be solved for immediately. The source strengths on the surface are given by the following equation:

$$\sigma \;=\; \frac{1}{4\pi}\;(V_{norm} - \bar{n}\cdot\bar{V}_{\infty}) \qquad (8)$$

The normal velocity, $V_{norm}$, on the surface is either zero (no flow through the surface) or a user-defined value (to simulate suction or blowing) and the onset velocity vector is known. Substituting equations (6) and (7) into equation (5) leaves the following integral equation with the unknown doublet strength over the surface to solve for:

$$0 = \left[\iint\limits_{S-P}\mu\;\bar{n}\cdot\nabla\left(\frac{1}{\bar{r}}\right) dS - 2\pi\mu_P\right] \;+\; \iint\limits_{S}\left(\frac{\sigma}{\bar{r}}\right) dS \;+\; \iint\limits_{W}\mu_W\;\bar{n}\cdot\nabla\left(\frac{1}{\bar{r}}\right) dS \qquad (9)$$

The general equation for the potential at any point P can be written as:

$$\Phi_P = \left[\iint\limits_{S-P}\mu\;\bar{n}\cdot\nabla\left(\frac{1}{\bar{r}}\right) dS + \;K\,\mu_P\right] + \iint\limits_{S}\left(\frac{\sigma}{\bar{r}}\right) dS \;+ \iint\limits_{W}\mu_W\;\bar{n}\cdot\nabla\left(\frac{1}{\bar{r}}\right) dS \;+\; \phi_{\infty P} \qquad (10)$$

where $K = 0$ if P is not on the surface, $K = 2\pi$ if P is on a smooth part of the outer surface, $K = -2\pi$ if P is on a smooth part of the inner surface, and K = the solid angle contained at the crease if P lies at a crease in the surface.

If the surface is broken up into panels, equation (9) can be written in discretized form, breaking the integrals up into surface integrals over each panel. PMARC assumes constant strength source and doublet distributions over each panel (thus making it a low-order panel method); therefore, the doublet and source strengths can be factored out of the integrals. Taking point P to be at the centroid on the inside surface of one of the panels, the surface integrals over each panel are summed for all panels. For the panel containing point P, the surface integral is zero and only the $-2\pi\mu_P$ tenn remains in the bracketed part of equation (9). For all other panels, the surface integral is used and the $-2\pi\mu_P$ term is zero since the point P is not on the surface of any of the other panels. The process is repeated for point P at the centroid of every panel to yield a set of linear simultaneous equations to be solved for the unknown doublet strength on each panel. The surface integrals represent the velocity potential influence coefficients per unit singularity strength for panel K acting on the control point of panel J. Hence equation (9) becomes:

$$\sum_{K=1}^{N_S}(\mu_K C_{JK}) + \sum_{K=1}^{N_S}(\sigma_K B_{JK}) + \sum_{L=1}^{N_W}(\mu_{W_L} C_{JL}) = 0\Bigg|_{J=1,\ N_S} \tag{11}$$

where

$$B_{JK} = \iint_K \frac{1}{r}\ dS \tag{12}$$

and

$$C_{JK} = \iint_K \bar{n}\cdot\nabla\left(\frac{1}{r}\right)\ dS$$

$$C_{JJ} = -2\pi \tag{13}$$

The coefficients $C_{JK}$ and $B_{JK}$ represent the velocity potential influence coefficients per unit singularity strength for panel K acting on the control point of panel J. Equations (12) and (13) are functions of geometry only and thus can be solved for all panels to form the influence coefficient matrix. Solutions for equations (12) and (13) can be found in reference 6. Since the source values are known, they may be transferred to the right hand side of the matrix equation. The wake doublet values can be determined as functions of the surface doublet values, as will be described in the section on time-stepping wakes, leaving only the surface doublet strengths as unknowns.

## Coordinate Systems for Unsteady Flow Analysis

PMARC was written to allow both steady and unsteady aerodynamic analysis. In PMARC, the geometry of a configuration is described in a body-fixed (global) coordinate system. The origin of the body-fixed coordinate system is assumed to be coincident with the origin of an inertial reference frame at time $t = 0$. The motion of the configuration at time $t > 0$ in the inertial reference frame is then described by the motion of the body-fixed reference frame. Currently in PMARC, the motion of the body fixed reference frame can be described in terms of a constant velocity vector and constant angular rotation rates about the three coordinate axes (see detailed input guide, appendix C). Sine, cosine, or general motion functions could be added to PMARC to allow cyclic or arbitrary motions. The configuration is marched through the prescribed motion in a series of time steps. A PMARC solution is computed at each incremental time step to develop the time history of the flow. The surface source strengths (eq. 8) must be updated to reflect the instantaneous free stream velocity vector in the body-fixed reference frame at each time step. The wake development and the flow field velocity computations, on the other hand, must be done in the inertial reference frame.

## Time-Stepping Wake Model

In PMARC, wakes can be shed from known separation lines on the surface geometry (trailing edges of wings, for example) to fix the rear stagnation point and the circulation of the body. The wakes in PMARC are time-stepping wakes (ref. 7). The first row of wake panels is shed from the separation line at time $t = 0$. With each subsequent time step, a new row of wake panels is added to the wake at the separation line and all the preceding rows of wake panels are convected downstream with the local velocity field (in the inertial reference frame). Alternatively, an initial wake may be prescribed by the user to simulate a steady-state condition. The initial wake can then be time-stepped, if desired, to further develop the wake. This is equivalent to starting the time-stepping at some time $t > 0$.

The Kutta condition is used as a boundary condition to determine the strength of the doublets to be shed into the first row of a wake. The Kutta condition states that the velocity at the rear stagnation point (i.e., wing trailing edge) must be finite. A panel (surface or wake) with a constant strength doublet distribution on it is equivalent to a vortex ring around the panel perimeter with strength equal to the doublet strength (ref. 8). When two or more vortex lines coincide, the strength of the resulting vortex line is equal to the sum of the individual vortex lines. In PMARC, the wake separation line is defined as the common edge between two rows or columns of surface panels from which the wake is shed as shown in figure 2. The resultant strength of the vortex line along the separation line must therefore be zero to satisfy the Kutta condition at that point. The strength of the vortex line due to the two surface panels is the difference in their doublet strengths since the vortex lines from the individual panels go in opposite directions along their common edge (see fig. 2). The strength of the vortex ring on the wake panel that is attached to the separation line must be equal to the difference in doublet strengths of the two surface panels in order to cancel the vortex line along the separation line. Thus the doublet strengths on the first row of wake panels is set equal to the difference in doublet strengths of the two rows or columns of surface panels whose common edge forms the separation line.

On the first time step, if no initial wake is specified, only a single row of wake panels exists on each wake. The doublet strengths on this row of wake panels are known in terms of the strengths of the doublets on the surface panels that the wake separates from, as described above. If an initial wake is prescribed, the doublet strengths on each row of wake panels following the first row is set equal to the doublet strengths on the first row (steady-state flow condition). In either case, the second summation in equation (11) can be combined with the surface panel summation, leaving only the surface doublet strengths as unknowns.

On subsequent time steps, a new row of wake panels is added to each wake at the wake separation line and the remaining rows of wake panels are convected downstream. The doublet strengths on all the wake panels, except for the new first row of panels, are known from the previous time step and can therefore be transfered to the right hand side of equation (11). The doublet strengths on the new first row of wake panels are again defined in terms of the strengths of the doublets on the surface panels that the wake separates from. The terms from the second summation in equation (11) pertaining to the new first row of wake panels are then combined with the surface panel summation, yielding a new matrix equation to be solved at each time step.

## Matrix Solver

The first step in the solution of equation (11) is the determination of the velocity potential influence coefficient matrix elements $C_{JK}$ for the unknown doublet strengths and $B_{JK}$ for the known source strengths. PMARC makes use of an approximation commonly employed in panel methods. For panels that are nearby, the influence coefficients are calculated exactly by treating the singularities as being distributed over the panel and integrating over the panel surface. For panels that are far away, however, the influence coefficients are calculated by treating the panel as though it were a point source or doublet. The distance at which this approximation starts being used is determined by the far-field radius. This distance is nondimensionalized by a characteristic panel size to give a far-field factor. The characteristic panel size is the sum of the distance from the midpoint of one side to the centroid of the panel and the distance from the midpoint of an adjacent side to the centroid. The far-field factor is then defined as the far-field radius divided by the characteristic panel size. In PMARC the default value of 5.0 for the far-field factor produces sufficiently accurate results for most cases. The default value for far-field factor can be changed by the user if so desired. The main purpose in using this approximation is that it provides a considerable savings in time with little loss in accuracy for most configurations.

Once the influence coefficients have been evaluated, the unknown doublet strengths can be solved for. Because the matrix equation that must be solved can become very large (the influence coefficient matrix contains 1,000,000 elements for a 1000-panel case), a fast iterative matrix solver that solves line by line without requiring the whole matrix to be in memory at once is used. PMARC makes use of a matrix solver based on an iterative scheme for computing the eigenvalues of large matrices (ref. 9). The solver can handle matrices of the order of $10^6 \times 10^6$. Only two small matrices of order n × 20, where n is the number of surface panels the arrays in PMARC are dimensioned to handle, and some scratch vectors of size n are needed in memory during the iterative solution process, allowing large matrices to be solved without requiring large amounts of memory. This permits the solution of fairly complex problems (1000 to 4000 panels) on mini- and micro-class computers.

The limit on number of solver iterations is set by the user in the input deck. At each iteration, a correction vector is computed which is applied to the current solution vector guess to get the solution vector guess for the next iteration. As the solution iterations proceed, the solver stores all the solution vector guesses and the correction vectors from each iteration in the two small matrices of order n × 20. The history of solution vector guesses and correction vectors from preceding iterations is used at each iteration to form the correction vector for the current iteration. After every twenty iterations, the current set of small matrices is folded to one (i.e., reset so that it holds only the correction vector and solution guess vector from the previous iteration) to prevent memory requirements from becoming too large (ref. 9). The folding of the small matrices usually adds only one or two extra iterations to the solution in return for allowing unlimited number of solution iterations with minimal memory requirements.

The doublet influence coefficient matrix remains on a disk scratch file and only one line at a time is read into memory to perform the matrix multiplication required at each solution iteration. PMARC has a parameter (MATBUF) that will allow buffered input of the doublet influence coefficient matrix during each iteration of the solver. MATBUF can be set equal to one for no buffering, or it can be set to NL, where NL is the number of lines of the doublet influence coefficient matrix to be transferred from disk scratch file to memory on each read. Setting MATBUF to a value greater than one reduces the number of physical IO requests that are issued, but it also increases the amount of memory required.

The solver requires the diagonal vector of the influence coefficient matrix and the right-hand-side vector to start the iterative process. For the first time step, a starting guess vector is created which is the right-hand-side vector divided by the diagonal vector. On subsequent time steps, the solver uses the doublet solution from the previous time step as its starting guess vector. This generally reduces the number of iterations required for a converged solution on all the time steps after the first one.

The convergence tolerance must also be passed to the solver. The convergence tolerance specified in PMARC is defined as the percent change in the solution vector elements between successive iterations. The element in the solution vector with the largest change is used to determine if the solution is converged or not. The cases run to date indicate that a convergence tolerance of 0.01 to 0.0005 is adequate for most problems, although smaller convergence tolerances may be necessary for certain cases. The solver will generally converge to a solution in 50 to 150 iterations. A single iteration cycle is of the order of 0.0866 CPU seconds for 1000 panels and 0.695 CPU seconds for 4000 panels on the CRAY Y-MP, or 15 CPU seconds for 1000 panels and 119 CPU seconds for 4000 panels on a MicroVAX II.

## On-Body Analysis

Once the unknown doublet strengths have been determined, the singularities on all the panels are known; thus, the velocities at the control points of the panels can be evaluated. The velocities normal to the panels are either zero or the value specified by the user. The tangential velocities on the surface are evaluated in a local panel coordinate system by differentiating the doublet strengths in the appropriate direction for each tangential component of velocity. With the three components of velocity calculated in the panel coordinate system, the velocities can be transformed into the x, y, z coordinate system of the entire configuration and a resultant velocity can be calculated. Using the resultant velocity at each panel control point, the pressure coefficient at each panel control point can be calculated using the following equation:

$$C_{P_K} = 1 - \frac{V_K^2}{V_\infty^2} + \left(\frac{2*4\pi}{V_\infty^2}\right)\left(\frac{\mu(t)-\mu(t-1)}{\Delta t}\right) \qquad (14)$$

The last term in equation (14) is the unsteady term, $d\Phi/dt$.

With the pressure distribution over the body determined, the resultant forces and moments on the body can be evaluated. Forces and moments are summed up, panel by panel, for each patch, component, assembly, and the whole model. Forces and moments are also summed up for each column of panels on user-specified patches to give section coefficients. The summed forces and moments are put in coefficient form and are transformed to wind, stability, and body axes. The section coefficients are written to the output file in wind, stability, and body axes, after each column of aerodynamic data for the patches on which they were computed. The patch, component, assembly, and total force and moment coefficients are written to the output file in wind, stability, and body axes, after the panel aerodynamic data for all the patches has been written. The patch wetted area, nondimensionalized by the reference area, is also written at this time. For the case where a plane of symmetry about $y = 0.0$ was used, the patch, component, and assembly force and moment coefficients are for the paneled geometry only. The total coefficients include the contribution of the reflected image.

### Off-Body Analysis

PMARC has the capability of computing the velocity at arbitrary points in the flow field. The velocities at points off the body are evaluated by taking the gradient of equation (10) with respect to the coordinates of point P. Thus equation (10) becomes:

$$\overline{V}_P = -\iint_S \mu\nabla\left(\bar{n}\cdot\nabla\left(\frac{1}{r}\right)\right)dS - \iint_S \sigma\nabla\left(\frac{1}{r}\right)\,dS - \iint_W \mu_W\,\nabla\left(\bar{n}\cdot\nabla\left(\frac{1}{r}\right)\right)dS + \overline{V}_\infty \qquad (15)$$

Equation (15) can also be written in discretized form similar to the equation for the potential at point P. The resulting discretized equation is:

$$\overline{V}_P = \overline{V}_\infty - \sum_{K=1}^{N_S}\left(\mu_K\ \overline{V}_{\mu PK}\right) - \sum_{L=1}^{N_W}\left(\mu_{W_L}\ \overline{V}_{\mu PL}\right) - \sum_{K=1}^{N_S}\left(\sigma_K\ \overline{V}_{\sigma PK}\right) \qquad (16)$$

where

$$\overline{V}_{\sigma PK} = \iint_K \nabla\left(\frac{1}{r}\right)\ dS \qquad (17)$$

and

$$\overline{V}_{\mu_{PK}} = \iint\limits_{K} \nabla\left(\bar{n}\cdot\nabla\left(\frac{1}{r}\right)\right) \; dS \tag{18}$$

The coefficients $\overline{V}_{\sigma_{PK}}$ and $\overline{V}_{\mu_{PK}}$ represent the velocity influence coefficients per unit singularity strength for panel K (surface or wake panel) acting on the point P. The velocity influence coefficients depend only on the geometry of the configuration and its wakes. The solution to equations (17) and (18) can be found in reference 6. As with the velocity potential influence coefficients, the far-field approximation is employed in computing the velocity influence coefficients. Equations (16), (17), and (18) are used for computing velocities at scan points, computing off-body streamlines, and for convecting the time-stepping wake.

## Internal Flow Model

The code PMARC currently supports modeling of internal flows. This is accomplished by modeling the internal flow geometry as a closed box with the panel normal vectors pointing into the interior of the box (ref. 10). Inflow and outflow to the box (flow through a duct) may be prescribed as normal velocities on groups of panels. The inflow and outflow must be prescribed so that the continuity equation is satisfied. The internal flow geometry can be a simple empty duct, or it may contain other objects such as wings or vanes to simulate a test article in a wind tunnel test section.

For internal flows, the doublet influence coefficient matrix is singular in its conventional form (ref. 11). This is because the potential function for internal flow geometries is known only to within an arbitrary constant, whereas for external flow geometries the arbitrary constant is determined by the potential at infinity ($\phi_\infty$). In order to circumvent this problem, the potential (or a doublet value) must be specified somewhere on the geometry as a boundary condition. Normally, the source values for all the panels are known on a geometry surface (eq. (8)) and the doublet values are solved for as in equation (11). For internal flows, the matrix singularity can be eliminated by arbitrary specification of a known doublet value, usually zero, on one panel. In the matrix equation, this known doublet value is substituted into equation (11) and the corresponding source value on the panel is treated as an unknown. Assuming the known doublet value is specified on the first panel, equation (11) can be rewritten to yield

$$\begin{bmatrix} B_{11} & C_{12} & C_{13} & \cdots \\ B_{21} & C_{22} & C_{23} & \cdots \\ B_{31} & C_{32} & C_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \mu_2 \\ \mu_3 \\ \vdots \end{bmatrix} = \begin{bmatrix} C_{11} & B_{12} & B_{13} & \cdots \\ C_{21} & B_{22} & B_{23} & \cdots \\ C_{31} & B_{32} & B_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} -\mu_1 \\ -\sigma_2 \\ -\sigma_3 \\ \vdots \end{bmatrix} \tag{19}$$

The solution of equation (19) results in a new, and possibly incorrect, source value ($\sigma_1$) for the panel on which the doublet ($\mu_1$) is prescribed. Once the matrix solution is obtained, the source value that was obtained from the matrix solution is replaced by the original source value computed using equation (8). This procedure allows the solution of an otherwise indeterminate matrix.

The panel used for the specified doublet has an impact on the convergence of the solution. Usually the best convergence is obtained if this panel is placed near the end of the matrix (i.e., the last panel). The doublet value used also affects convergence. The first run should be made with the prescribed doublet value set to zero. After the solution for all the doublet strengths is obtained, the prescribed doublet value is set equal to the weighted average of the doublet values on the neighboring panels. This minimizes any local velocity distortions at the panel on which the doublet strength was prescribed. If poor solver convergence is a problem, the prescribed doublet value can be set to a nonzero value and the computation rerun.

In internal flow problems, there is a certain amount of leakage in or out of the internal flow geometry. This leakage is due to the Neumann boundary condition for each panel being imposed only at panel control points (ref. 10). The leakage from an internal flow geometry can be minimized by a combination of several means. First, panel density can be increased on the tunnel or duct walls. Second, the far-field factor can be increased so that more panels are treated as having distributed sources instead of point sources. Finally the free-stream velocity vector can be adjusted so that it is as close as possible to the average velocity in the tunnel or duct. It is possible to increase the panel density and the far-field factor enough to keep the tunnel or duct from leaking even if the free-stream velocity vector is set to zero. However, increasing the panel density and/or the far-field factor both will increase the execution time of PMARC for a given problem. Varying the free-stream velocity vector will not change the execution time of PMARC significantly. Changing the free-stream velocity vector changes the source strengths (eq. 8) and therefore the doublet strengths $\mu$ (eq. 11). The leakage appears to be minimized when the doublet strengths are minimized throughout the internal flow geometry. The free-stream velocity vector is normally set to a value between the minimum and maximum velocities in the internal flow geometry and is adjusted until conservation of mass is achieved. Conservation of mass may be determined by looking at inflow and outflow velocity specifications and the velocities at different cross sections of the geometry.

## Jet Plume Model

PMARC has a simple jet model for simulating a jet plume in a crossflow. The jet plume shape, trajectory, and entrainment velocities are computed using the Adler/Baron jet in crossflow code (ref. 12). This information is then passed back to PMARC where the jet plume is modeled with a surface patch and the entrainment is modeled with normal velocities specified on the jet plume panels. There are several restrictions imposed by the Adler/Baron code on the type of jets that can be modeled. The first restriction is that the jet exit must be symmetric about a plane containing the normal vector to the jet exit and the free-stream velocity vector. The second restriction is that the jet velocity ratio should be between 3.0 and 8.0. The Adler/Baron code gives unpredictable results for jet velocity ratios outside this range. The third restriction is that the angle between the jet and the free-stream velocity vector should not exceed approximately 100°. For angles greater than 100°, the Adler/Baron code tends to break down. Within the restrictions described above, the jet model in PMARC does a reasonable job of modeling the global effects of a jet.

## DATA MANAGEMENT

One of the keys to the success or failure of any numerical method is how well data is managed within the code. In a panel code there are many large arrays and blocks of data that need to be manipulated within the code as well as written to output and plot files. If a good data management scheme is not implemented within the code, the code can quickly become too large, inefficient, and slow to be of any use. A data management scheme has been devised for PMARC which seeks to maximize the number of panels the code can handle while minimizing the amount of memory and disk scratch space required to run the code. Specific aspects of the data management scheme include use of variable dimensioning for all major arrays within the code, provision of a reasonable balance between the amount of memory used and the amount of disk scratch space used, and elimination of redundancy of variables both within the code and in the plot file.

### Code Sizing

PMARC was written using adjustable size arrays throughout the code. A set of parameter statements controls the dimensioning of all the arrays in PMARC (see appendix A). Integer limits for variables and loops within the code are also defined in terms of the parameter values. This eliminates the possibility of forgetting an array or limit in the redimensioning process. The set of parameter statements is included at the beginning of all the major subroutines in PMARC (there are 22 occurrences of the parameter statement set in PMARC). To change the dimensions of arrays in PMARC, the source code must be edited and the appropriate parameter statements modified on a global basis throughout the source code. For example, to change the number of surface panels PMARC can handle from 1000 to 4000, all occurrences of NSPDIM = 1000 must be changed to NSPDIM = 4000. Once the changes have been made, the PMARC source code must be recompiled for the changes to become effective. Thus the size of the code (i.e., the number of panels it can handle) can be changed from several hundred to 10 or 20 thousand or more in a matter of minutes.

The main limitation on the number of panels the code can handle is the amount of memory and disk space available on the machine on which the code is being run. The current version of PMARC can be run with 3000 panels on a MicroVAX II with 3 Mb of memory and 210 Mb of disk space. PMARC has been run on the Cray Y-MP with as many as 6000 panels without any problem. A version of PMARC has even been run with 1000 panels on a MacIntosh II with 2.5 Mb of memory and 20 Mb of disk space. By selectively changing certain parameters, the user can increase the capacity of one particular part of the code, say the number of patches or wakes allowed, without having to increase the capacity of the entire code. This allows the user to customize the size of the code to fit his particular needs and hardware capacity.

Redimensioning PMARC will change the amount of disk space and memory required by the code to run. The only data PMARC writes to disk scratch space is the influence coefficient matrices. The amount of scratch disk space PMARC requires to run can be calculated using the following equation:

$$\text{Disk space (bytes)} = (\text{NSPDIM})^2 * 3 * \text{RBYTES}$$

where NSPDIM is the parameter controling the total number of surface panels the code can handle and RBYTES is the number of bytes required to represent a real number on the computer being used (typically RBYTES = 4). Additional disk space is required for the input, output, and plot files. The amount of disk space required for these three files depends on the options selected in the input deck, but it is usually small compared to the amount of scratch disk space required. Minimizing the disk IO and using common blocks to pass information between subroutines greatly streamlines the coding and produces a faster-running code.

The amount of memory PMARC requires to run is not as easy to compute as the amount of disk space is. The amount of memory required can be divided into two parts. The first is the memory allocated for storing the code itself (the execution instructions). This memory allocation is fixed for a given computer and compiler. The second part is the memory allocated for storing the data (arrays and scalar variables and constants). It is the second part that is significantly affected by redimensioning the code. On a Macintosh IIx using the DCM Mactran Plus Fortran 77 compiler with the optimize option and 4-byte integers and logicals, the storage required for the code is approximately 300 Kbytes. With PMARC dimensioned according to the parameter statements listed in appendix A, the data storage required is approximately 1.83 Mbytes. It should be noted that the DCM Fortran 77 compiler uses static allocation for the data. Compilers that use dynamic allocation will require less memory for data storage. The table in appendix B will give the user a feel for how changing the various parameter values in PMARC affect the amount of memory required for running the code.

## Plot File

The PMARC plot file is designed to contain as much information as possible regarding geometry and aerodynamic data in as compact a space as possible. The idea is to let PMARC do the computing and have a post-processor plotting package process and display the information. Computing should be kept to a minimum within the plotting package to keep its speed high. The general blocking of the data within the PMARC plot file is shown in figure 3. The geometry data is written to the file first. The geometry data is outside the wake time step loop since the geometry data will not change from time step to time step. A block of wake and aerodynamic data is written for each wake time step. The off-body velocity scan and off-body streamline data are appended to the wake and aerodynamic data block following the completion of the wake time step loop.

The arrangement of the geometry data block is as follows. The first record contains the length of the PMARC run and the number of time steps (and the number of boundary layer iterations, currently set to zero as the boundary layer routines have not yet been implemented in PMARC). The next record contains the total number of patches in the geometry. This is followed by a set of records containing patch number, number of columns and rows, and first and last panel number on each patch. After each record identifying the parameters of a patch, is a record with the patch name. The next record after the patch data set contains the total number of panels in the geometry. Next, panel information is written, stepping through each patch, each column on each patch, and each row on each column (see fig. 4 for patch nomenclature). Each record consists of the (x, y, z) coordinates of the first corner point of each panel, the coordinates of the panel centroid, and the panel normal vector at the panel centroid. At the end of each column of panels, an extra record must be included which contains the coordinates of the second corner point of the last panel in the column. After the data for the last column of panels has been written, an extra set of records is included to write the fourth corner point of the panels in the last column and the third corner point of the last panel in the last column. In this fashion all the corner points, centroids, and normal vectors are written to the plot file for each panel on each patch with no duplication within a patch. Following the corner point, centroid, and normal data is the panel neighbor data. Each record contains the neighbor panel number and the side of that neighbor panel adjacent to a given panel for each of the four sides of that panel.

Following the geometry data block is the wake data block. The wake data is written to the plot file in the same way as the geometry data. The only exceptions are that wake number replaces patch number and neighbor data is not written for wake panels. Thus the wake data includes wake panel corner points, centroids, and normal vectors.

The aerodynamic data block is written next. Aerodynamic data is written at the panel corner points and at the centroids using the same logic that is used to write the geometry and wake data. The aerodynamic quantities that are written to the plot file are doublet strength, the (x, y, z) components of velocity, the velocity magnitude, the pressure coefficient, and the local Mach number at each panel corner point and centroid.

The remainder of the data written to the plot file consists of optional data blocks. These include off-body velocity scan data, and off-body streamline data. The off-body streamline data includes the number of lines, the number of points on each line, the (x, y, z) location of each point, the three velocity components and the velocity magnitude, the pressure coefficient, the local Mach number, and the arclength at each point on each streamline. The off-body velocity scan data consists of the number of rectangular volumes and the number of cylindrical volumes, and the number of points in the (i, j, k) directions within each volume. For each scan point the (x, y, z) location, the velocity components and magnitude, the pressure coefficient, and the local Mach number are written.

## OVERVIEW OF PROGRAM INPUT

### Geometry Modeling

PMARC has extensive geometry modeling capabilities for handling complex, three-dimensional surfaces. As with all panel methods, the geometry must be modeled by a set of panels. For convenience, the geometry is usually subdivided into several pieces and modeled with sets of panels called patches. Each patch is constructed from two or more sections. A section is a set of points defining a cross-sectional cut through the geometry that the patch is modeling. Figure 4 shows the PMARC nomenclature and the conventions used for a patch. In general, each patch is four-sided; however, one or two sides of a patch may be of zero length (i.e., a point). A patch may also be folded over on itself so that opposing sides of the patch form a common line. For instance, wings are normally modelled with a folded patch with sides 2 and 4 meeting to form the trailing edge of the wing. The first section used to define a patch becomes side 1 of the patch. The direction of side 1 of the patch is the same as the direction of the first section definition. The direction of sides 2, 3, and 4 proceed counterclockwise around the patch when viewing the outer surface of the patch. The last section used to define a patch becomes side 3 of the patch. Columns of panels are established between sides 1 and 3 of the patch. Rows of panels are established between sides 2 and 4 of the patch. The sequence for panel numbering on a patch is as shown in figure 4.

One important factor to keep in mind when defining sections to create a patch is that the order of input determines whether or not the patch is inside out. To determine whether or not a patch is inside out, a right-hand rule can be used. If the fingers of the right hand are pointing in the direction of section definition, and the thumb of the right hand is pointing from the first section to the last section on the patch, the palm of the right hand will be resting on the outside surface of the patch. If a patch is inside out, it can be easily reversed using the patch reversal option on the PATCH1 namelist in the input deck (see detailed input guide, appendix C).

Because PMARC is a low-order panel method, panels do not have to match exactly across patch boundaries. Small gaps and panel mismatches can be tolerated in PMARC; however, the panel mismatches should not be too severe or PMARC will have trouble computing the surface velocities near the patch boundaries. In general, it is best to provide smooth transitions in panel size and density between patches.

The hierarchy for modeling geometries in PMARC is shown in figure 5. There are several levels of coordinate systems available in PMARC. The bottom level coordinate system is the section coordinate system. The section coordinate system is used to define the sections that make up patches. The next level in the hierarchy is the component coordinate system. Above the component coordinate system is the assembly coordinate system. Finally at the top level of the hierarchy is the global coordinate system. Each coordinate system allows all the elements below it to be translated, rotated, and scaled. PMARC transfers all the panel corner points from the various coordinate systems into the global coordinate system for use within the code and for the output and plot files.

PMARC has the capability to automatically generate a closing tip patch on either side 1 or side 3 of a folded patch. Figure 6 illustrates a tip patch on a folded wing patch. The wing patch is folded such that side 2 and side 4 form a common trailing edge. The tip patch is then generated to close off side 3 of the wing. The input for generating a tip patch is given in the detailed input guide section, appendix C. In order to form the tip patch, PMARC identifies the panel corner points on the side of the patch to be closed off. The total number of panel corner points is then divided in half. The first half of the corner points is used to form the first section of the tip patch and the second half of the corner points is used, in reverse order, to form the last section of the tip patch. In this way, the paneling on the tip patch matches exactly the paneling on the wing. The number of panels to be generated between the two sections of the tip patch is set by the user on the PATCH2 namelist (see detailed input guide, appendix C).

## Wake Modeling

As mentioned earlier, the wake model in PMARC is a time-stepping wake model. The wake is convected downstream from the wake separation line by the local velocity flow field. A new row of wake panels is added to the wake at the wake separation line with each time step. The number of time steps and the step size are set by the user on the BINP5 namelist (see detailed input guide, appendix C). PMARC also allows an initial wake to be specified if desired. The time-stepping functions the same with or without an initial wake specified. The initial wake capability allows the user to look at steady-state problems without going through several time steps to reach the steady-state condition. A third option in PMARC is to have no wakes modeled. In this case a single WAKE1 namelist must be included in the input deck to tell PMARC that there are no wakes.

Figure 7 shows the hierarchy for wake modeling in PMARC. If an initial wake is specified, section coordinate systems are used to define the wake sections. The wake section definitions are then transformed directly into the global coordinate system. If no initial wake is specified and the wake is time-stepped, then PMARC forms the wake directly in the global coordinate system as it goes through the time steps.

The first step in defining a wake (whether there is an initial wake specified or not) is to define the wake separation line. This is done on the WAKE2 namelist. The WAKE2 namelist identifies which patch a wake separates from. It also identifies the side (KWSIDE) of the patch which is parallel to the separation line and the row or column of panels within the patch that the wake separates from. The separation line can extend over the entire row or column of panels or can be limited to a subset of panels within the row or column. If the wake separates from more than one patch, a separate WAKE2 namelist must be included for each patch that the wake separates from (see detailed input guide, appendix C). The wake separation line will be in the same direction as the side KWSIDE. There will be one column of wake panels for each surface panel that the wake separates from. PMARC treats the wake separation line as the first section of the wake. Figure 8 shows the separation line for a wake separating from the trailing edge of a folded wing patch. In this case, the wake was defined to separate from side 2 of the patch. The wake could have been defined to separate from side 4 (since side 2 and side 4 form a common line) and the only difference would be that the direction of the separation line would be reversed.

If an initial wake is to be specified, at least one more section of the wake must be defined in addition to the separation line. Additional wake sections can be specified in the same manner as surface sections are (see detailed input guide, appendix C). There are two restrictions on wake section definitions. The first restriction is that the user-defined wake sections must all go in the same direction as the separation line, as shown in figure 8. The second restriction is that the total number of panels defined on each wake section must equal the total number of surface panels that the wake separates from. Aside from these restrictions, the wake sections can be defined to have any arbitrary shape and can be located anywhere in the global coordinate system the user chooses. PMARC forms the wake by fitting splines through corresponding panel corner points on all sections, starting from the first section (the wake separation line) and going to the last section. Thus an initial wake of arbitrary shape can be defined in much the same way a surface patch is defined.

## Off-Body Velocity Scans

Once the doublet strengths have been determined and the on-body analysis has been completed, velocities can be computed at user-defined points in the flow field. Rectangular or cylindrical volumes of uniformly spaced scan points can be defined by the user. Either type of scan volume can be degenerated into planes, lines, or points. The velocity components, velocity magnitude, pressure coefficient, and Mach number are computed at every scan point.

The rectangular scan volume is specified in terms of three direction vectors (**i**, **j**, and **k**) all originating from a common point, as shown in figure 9. The length of each direction vector defines the corresponding length of that side of the scan volume. The number of points to be evenly distributed along each direction vector of the scan volume must be input. If zero is entered for the number of scan points along one direction vector, that side of the scan volume degenerates to zero length. For example, if zero were entered for the number of points in the k direction, the scan volume would degenerate into the plane of points defined by the **i** and **j** direction vectors. It is important to note that the specification of the **i**, **j**, and **k** direction vectors is completely arbitrary. The **i**, **j**, and **k** direction vectors do not need to form an orthogonal set of vectors.

The cylindrical scan volume is specified in terms of a cylinder axis vector and a second vector used to define the plane from which angle of rotation is measured, as shown in figure 10. The cylinder axis vector and the second vector need not be orthogonal; PMARC uses the two vectors to construct a right-hand orthogonal coordinate system. The cylindrical scan volume can have any orientation desired in the global coordinate system. The beginning and ending radii and rotation angles are specified in the input deck, while the length of the cylindrical scan volume is determined by the cylinder axis vector. The number of scan points to be distributed along the axial, radial, and angular directions must be specified in the input deck. As with the rectangular scan volume, if zero is entered for the number of scan points along one direction, that side of the scan volume degenerates to zero length.

### Off-Body Streamlines

PMARC has the capability of computing streamlines in the flow field. A starting location for each streamline must be specified, as well as the distance upstream and downstream (measured along the streamline) the streamline calculation is to extend and the step size to be used. Care must be exercised in selecting the starting position for streamlines. If a streamline is started too close to a stagnation line, the streamline could penetrate the body. If this occurs, there are three possible remedies: increase the panel density in the region where the streamline penetrates the body, increase the far-field factor so that more panels are treated as having distributed sources and doublets on them, or decreasing the streamline step size. All of these will increase the execution time of PMARC. The best approach to use if streamlines are needed in close proximity to the surface of the body is to specify a set of streamlines whose starting locations begin at the stagnation line and move progressively farther away. The streamlines that penetrate the body can be ignored.

## TEST CASE

A simple symmetric wing/body configuration was one of the test cases used to validate PMARC. The configuration selected was one for which extensive pressure data was available from wind tunnel tests (ref. 13) and which was generic in nature. The body has a circular cross-section and a fineness ratio of 12 (the wind tunnel model was truncated at the trailing edge for mounting on a sting, yielding an effective fineness ratio of 10). The wing is mounted at the body centerline near the point of maximum thickness. The wing has a NACA 65A006 airfoil, an aspect ratio of 4.0, a taper ratio of 0.6, and is swept back 45° at the quarter chord.

The PMARC representation of the wing/body configuration is shown in figure 11. Only half of the configuration was modeled in PMARC. The other half of the configuration was generated by reflecting the model across the plane of symmetry. The wing was represented with 300 panels: 15 divisions in the chordwise direction on the upper and lower surface of the wing with denser spacing near the leading and trailing edges, and 10 divisions in the spanwise direction with denser spacing near the root and tip of the wing. The tip of the wing was closed off with a flat tip patch. The body was represented with 320 panels. The wing/body junction was modeled such that wing and body panels matched up exactly. An initial wake was attached to the trailing edge of the wing and to the aft fuselage and carried downstream 20 chord lengths. Three time steps were specified to allow the wake start to roll up. The input file for this test case can be found in appendix D.

A comparison of pressure coefficients from experimental data (ref. 13) and PMARC data at two spanwise stations on the wing is shown in figure 12. The model is at an angle of attack of 4°. Agreement between PMARC results and the experimental data is excellent. Figure 13 shows a comparison of pressure coefficients from experimental data and PMARC data along the centerline of the body. Again, the model is at an angle of attack of 4°. The PMARC results correlate well with the experimental data. There is some difference near the trailing edge of the body, but this can probably be attributed to the presence of the sting in the experimental results. The sting was not modeled in the PMARC analysis. Figure 13 also illustrates the importance of attaching a wake to the aft part of the body. Without a wake, the body carries no lift. The experimental data shows that there is carryover lift from the wing to the body. By attaching a wake to the aft part of the body, the carryover lift is properly modeled.

## CONCLUDING REMARKS

The theoretical basis for PMARC, a low-order potential-flow panel code for modeling complex three-dimensional geometries, has been outlined. Several of the advanced features currently included in the code, such as internal flow modeling, a simple jet model, and a time-stepping wake model, have been discussed in some detail. The code was written using adjustable size arrays so that it can be easily redimensioned for the size problem being solved and the computer hardware being used. An overview of the program input was presented,with a detailed description of the input available in the appendices. Finally, PMARC results for a generic wing/body configuration were compared with experimental data to demonstrate the accuracy of the code. The input file for this test case is given in the appendices.

# REFERENCES

1. Ashby, D. L.; Dudley, M. R.; and Iguchi, S. K.: Development and Validation of an Advanced Low-Order Panel Method. NASA TM-101024, Oct. 1988.

2. Margason, R. J.; Kjelgaard, S. O.; Sellers, W. L.; Morris, C. E. K.; Walkey, K. B.; and Shields, E. W.: Subsonic Panel Methods—A Comparison of Several Production Codes. AIAA Paper 85-0280, Jan. 1985.

3. Maskew, B.: Program VSAERO, A Computer Program for Calculating the Nonlinear Characteristics of Arbitrary Configurations, Users Manual. NASA CR-166476, Dec. 1982.

4. Maskew, B.: Program VSAERO, A Computer Program for Calculating the Nonlinear Characteristics of Arbitrary Configurations, Theory Document. NASA CR-4023, Sept. 1987.

5. Maskew, B.; Strash, D.; Nathman J.; and Dvorak, F. A.: Investigation to Advance Prediction Techniques of the Low-Speed Aerodynamics of V/STOL Aircraft. NASA CR-166479, Feb.1983.

6. Hess, J.L.; and Smith, A.M.O.: Calculation of Potential Flow About Arbitrary Bodies.Progress in Aeronautical Sciences, Vol. 8, 1967, pp. 1-138.

7. Katz, J.; and Maskew, B.: Unsteady Low-Speed Aerodynamic Model for Complete Aircraft Configurations. AIAA Paper 86-2180, Aug. 1986.

8. Hess, J. L.: Calculation of Potential Flow About Arbitrary Three-Dimensional Lifting Bodies. McDonnell Douglas Corp. Rep. No. MDC J5679-01, Oct. 1972.

9. Davidson, E. R.: The Iterative Calculation of a Few of the Lowest Eigenvalues and Corresponding Eigenvectors of Large Real-Symmetric Matrices. J. Comp. Phys. 17, 1975, pp. 87-94.

10. Ashby, D. L.; and Sandlin, D. R.: Application of a Low Order Panel Method to Complex Three-Dimensional Internal Flow Problems. NASA CR-177424, Sept. 1986.

11. Hunt, B.: The Panel Method for Subsonic Aerodynamic Flows: A Survey of Mathematical Formulations and Numerical Models and an Outline of the New British Aerospace Scheme. Von Karman Institute for Fluid Dynamics, Lecture Series 1978-4, vol. I, March 13-17, 1978.

12. Adler, D.; and Baron, A.: Prediction of a Three-Dimensional Circular Turbulent Jet in Crossflow. AIAA Journal, Vol. 17, No. Sept. 1978.

13. Loving, D. L.; and Estabrooks, B. B.: Analysis of Pressure Distribution of Wing-Fuselage Configuration Having a Wing of 45° Sweepback, Aspect Ratio 4, Taper Ratio 0.6, and NACA 65A006 Airfoil Section. NACA RM L51F07, Sept. 1951.

## APPENDIX A

## PMARC PARAMETER STATEMENT SET

```
C
C  CODE DIMENSIONING PARAMETERS
C
C  NUMBER OF SURFACE PANELS ALLOWED
C
      PARAMETER (NSPDIM = 1000)
C
C  NUMBER OF NEUMANN PANELS ALLOWED
C
      PARAMETER (NNPDIM = 500)
C
C  NUMBER OF PATCHES ALLOWED
C
      PARAMETER (NPDIM = 50)
C
C  NUMBER OF BASIC POINTS ALLOWED FOR SECTION DEFINITION
C  (ALSO NUMBER OF SECTIONS ALLOWED PER PATCH)
C  (ALSO NUMBER OF ROWS OR COLUMNS + 1 ALLOWED ON A PATCH)
C  CAUTION:  DO NOT SET THIS PARAMETER TO LESS THAN 50!
C
      PARAMETER (NBPDIM = 100)
C
C  NUMBER OF WAKE PANELS ALLOWED
C
      PARAMETER (NWPDIM = 1000)
C
C  NUMBER OF WAKE COLUMNS ALLOWED ON EACH WAKE
C
      PARAMETER (NWCDIM = 50)
C
C  NUMBER OF WAKES ALLOWED
C
      PARAMETER (NWDIM = 10)
C
C  NUMBER OF SCAN VOLUMES OF EACH TYPE ALLOWED
C
      PARAMETER (NSVDIM = 10)
C
C  NUMBER OF POINTS PER OFF-BODY STREAMLINE ALLOWED
C
      PARAMETER (NSLPDIM = 1000)
C
C  NUMBER OF GROUPS OF PANELS ON WHICH NONZERO NORMAL
C  VELOCITY IS PRESCRIBED
C
      PARAMETER (NVELDIM = 200)
```

```
C
C  NUMBER OF LINES AT A TIME TO BE READ IN FOR THE INFLUENCE
COEF.
C  MATRIX IN THE SOLVER ROUTINE (BUFFERED INPUT FROM THE
SCRATCH
C  FILE)  (CAUTION:  DO NOT SET LARGER THAN ONE UNLESS YOU ARE
SURE
C  YOU HAVE ENOUGH MEMORY TO HANDLE BUFFERED INPUT!)
C
     PARAMETER (MATBUF = 1)
C
C  NUMBER OF WAKE CORNER POINTS ALLOWED
C
      PARAMETER (NWCPDIM=(NWPDIM + 1)*2)
C
C  NUMBER OF SURFACE CORNER POINTS ALLOWED
C
      PARAMETER (NSCPDIM=(NSPDIM + 1)*2)
C
C  NUMBER OF EDGE PANELS ALLOWED ON A PATCH
C
     PARAMETER (NEPDIM = NBPDIM * 4)
```

# APPENDIX B

## MEMORY REQUIREMENTS FOR DATA STORAGE IN PMARC

The data storage for PMARC can be divided into two types: common block storage and local storage. The memory requirements for both types of storage is presented below in terms of the parameter variables. Storage for local scalar variables and constants is not included in the information presented below. The value of RBYTE below is the number of bytes needed to represent a real number. RBYTE is 4 for most computers.

### Memory allocated for common blocks

| | |
|---|---|
| NSPDIM | * 38 * RBYTE |
| NPDIM | * 16 * RBYTE |
| NSCPDIM = ((NSPDIM + 1) * 2) | * 4 * RBYTE |
| NNPDIM | * 3 * RBYTE |
| NWPDIM | * 15 * RBYTE |
| NWDIM | * 13 * RBYTE |
| NWCPDIM = ((NWPDIM + 1) * 2) | * 3 * RBYTE |
| NWCDIM * NWDIM | * 4 * RBYTE |
| NVELDIM | * 6 * RBYTE |
| 1 | * 125 * RBYTE |

### Memory allocated locally in subroutines by dimension statements

#### Aerodat

| | |
|---|---|
| NPDIM | * 26 * RBYTE |
| 10 | * 36 * RBYTE |
| 1 | * 24 * RBYTE |

#### Cornerpt

| | |
|---|---|
| NSCPDIM = ((NSPDIM + 1) * 2) | * 4 * RBYTE |
| 4 | * 13 * RBYTE |

#### Nabors

| | |
|---|---|
| NPDIM | * 1 * RBYTE |
| NEPDIM * NPDIM = (NBPDIM * NPDIM * 4) | * 2 * RBYTE |
| NSPDIM | * 2 * RBYTE |
| 1 | * 28 * RBYTE |

#### Rhs

| | |
|---|---|
| NSPDIM | * 3 * RBYTE |

#### Solver
#### (Lineq)

| | |
|---|---|
| NSPDIM | * 43 * RBYTE |
| 1 | * 840 RBYTE |

(Frmab)

| | |
|---|---|
| NSPDIM | * 2 * RBYTE |
| NSPDIM * MATBUF | * 1 * RBYTE |

Surmlin

| | |
|---|---|
| NSLPDIM | * 10 * RBYTE |

Surfgen

| | |
|---|---|
| NBPDIM | * 35 * RBYTE |
| NBPDIM * NBPDIM | * 3 * RBYTE |
| 10 | * 22 * RBYTE |
| 1 | * 9 * RBYTE |

Surphi

| | |
|---|---|
| NSPDIM | * 11 * RBYTE |

Vcalc

| | |
|---|---|
| NSPDIM | * 6 * RBYTE |
| NWPDIM | * 6 * RBYTE |
| 1 | * 28 * RBYTE |

Vscan

| | |
|---|---|
| NSVDIM | * 31 * RBYTE |
| 1 | * 9 * RBYTE |

Wakinfl

| | |
|---|---|
| NSPDIM | * 1 * RBYTE |
| NWPDIM | * 1 * RBYTE |
| NSPDIM * MATBUF | * 1 * RBYTE |
| 1 | * 27 * RBYTE |

Wakinit

| | |
|---|---|
| NWCDIM | * 35 * RBYTE |
| NWCDIM * NWCDIM | * 3 * RBYTE |
| NSPDIM | * 1 * RBYTE |
| NPDIM | * 3 * RBYTE |
| 1 | * 21 * RBYTE |

Wakstep

| | |
|---|---|
| NWCPDIM = ((NWPDIM + 1) * 2) | * 3 * RBYTE |
| NWPDIM | * 1 * RBYTE |
| NSPDIM | * 1 * RBYTE |
| NWDIM | * 1 * RBYTE |

# APPENDIX C

# PMARC DETAILED INPUT GUIDE

## Basic Input Section to Run PMARC

The basic data section of the input deck for PMARC consists of a set of namelist definitions. The required format for the basic data section is shown below. The best way to handle the basic data section is to create a template file which can then be included into any PMARC file and the values modified appropriately. All the namelists should always be included as shown below whether or not a particular namelist is needed for the job being run. If a namelist is not needed for a particular job, PMARC merely skips over that namelist. Each namelist must begin with an & in the second column and the namelist name (i.e., BINP2, BINP3, etc.) and end with &END. Blank spaces in a namelist are ignored, so the items in each namelist can be spaced in whatever fashion the user desires. A namelist can extend over as many lines as necessary. A description of each input variable and the valid input values follows. Under the Value column in the input description, the letter I means an integer value and the letter R means a real value.

NOTE: Variables in the namelist definition which are arrays should have their elements listed out following the variable name. For example, if there were three values to be entered into the array NORPCH, the input would be as follows: NORPCH = N1,N2,N3. The rest of the elements in array NORPCH will automatically be left at zero.

```
TYPE YOUR TITLE FOR THIS INPUT FILE HERE

&BINP2      LSTINP=2,     LSTOUT=0,     LSTFRQ=1,     LENRUN=0,            &END
&BINP3      LSTGEO=0,     LSTNAB=0,     LSTWAK=0,     LSTCPV=0,
            LSTJET=0,                                                      &END
&BINP4      MAXIT=75,     SOLRES=0.0005,                                   &END
&BINP5      NTSTPS=1,     DTSTEP=0.1,                                      &END
&BINP6      RSYM=0.0,     RGPR=0.0,     RFF=5.0,      RCORE=0.05,          &END
&BINP7      VINF=1.0,     VSOUND=1116.0,              UNIT=0,
            COMPOP=0.0,                                                    &END
&BINP8      ALDEG=4.0,    YAWDEG=0.0,   THEDOT=0.0,   PSIDOT=0.0,
            PHIDOT=0.0,                                                    &END
&BINP9      CBAR=3.00,    SREF=100.0,   SSPAN=15.0,
            RMPX=0.00,    RMPY=0.00,    RMPZ=0.00,                         &END
&BINP10     NORSET=0,     NBCHGE=0,     NCZONE=0,
            NCZPAN=0,     CZDUB=0.0,    VREF=00.0,                         &END
&BINP11     NORPCH=0,
            NORF=0,                     NORL=0,
            NOCF=0,                     NOCL=0,
            VNORM=0.0,                                                     &END
&BINP12     KPAN=0,       KSIDE=0,      NEWNAB=0,     NEWSID=0,            &END
```

RECORD 1: Job Title

| Variable | Value | Description |
|---|---|---|
| TEXT | | Alphanumeric text identifying the job This record is not entered in namelist format, but merely typed in anywhere on the first line of the input deck. |

BINP2: Job Control

| Variable | Value | Description |
|---|---|---|
| LSTINP | | Input data print options |
| | 0 | Prints all input data except the geometry input. |
| | 1 | Prints all input data except the detailed coordinates of the geometry input. |
| | 2 | Prints all input data. |
| LSTOUT | | Output print options |
| | 0 | Basic print of output. |
| | 1 | Allows any or all of the additional print options to be set manually on BINP3. |
| LSTFRQ | | Controls frequency of printout in the time-stepping loop. |
| | 0 | Prints out detailed panel data only on last step. Force and moment data and solution iteration history printed at every step. |
| | 1 | Prints all data at every step. |
| | I | Prints out detailed panel data at every Ith step, including the first and last step. Force and moment data and solution iteration history printed at every step. |

| Variable | Value | Description |
|---|---|---|
| LENRUN | 0 | Complete run through code. |
| | 2 | Run through geometry only. Geometry is written to plot file. |
| | 3 | Run through geometry and wake initialization routines. Geometry and initial wake data are written to plot file. |

BINP3: Additional Print Options (must be included if LSTOUT=1)

| Variable | Value | Description |
|---|---|---|
| LSTGEO | | Panel geometry printout options. |
| | 0 | Print option off. |
| | 1 | Panel corner points printed for all panels. |
| | 2 | Panel corner points and unit normal vectors printed for all panels. |
| | 3 | Panel corner points, unit normal vectors, and panel sets with prescribed normal velocities are printed out. |
| LSTNAB | | Panel neighbor information printout options. |
| | 0 | Print option off |
| | 1 | Prints neighbor information for all panels. |
| LSTWAK | | Wake data printout options. |
| | 0 | Print option off. |
| | 1 | Prints wake-shedding information for each wake column. |
| | 2 | Prints wake-shedding information for each wake column and wake line geometry. |
| | 3 | Prints wake-shedding information for each wake column, wake line geometry, and wake panel doublet values. |

| Variable | Value | Description |
|---|---|---|
| LSTCPV | | Panel corner point analysis printout options. |
| | 0 | Print option off. |
| | 1 | Prints out panel corner point analysis results. Will be printed according to the LSTFRQ value selected. |
| LSTJET | 0 | Print option off |
| | 1 | Print out jet analysis results |

BINP4: Solver Parameters

| Variable | Value | Description |
|---|---|---|
| MAXIT | I | Limit on number of solver iterations (150 is adequate for most cases) |
| SOLRES | R | Convergence criteria for the matrix solver. Recommended setting is 0.0005. |

BINP5: Time-Step Parameters

| Variable | Value | Description |
|---|---|---|
| NTSTPS | I | Number of wake time-steps. |
| DTSTEP | R | Size of the time-step (seconds). |

BINP6: Symmetry and Computation Parameters

| Variable | Value | Description |
|---|---|---|
| RSYM | 0.0 | Symmetrical case (about Y=0). Code computes the influence of the mirror image of the paneled geometry. The paneled geometry must lie in the +Y side of the global coordinate system and abut the Y=0 plane. |
| | 1.0 | Asymmetrical case (about Y=0). The entire geometry must be paneled. The paneled geometry may lie in +Y or -Y (or both) side of the global coordinate system. |
| RGPR | 0.0 | No ground plane modeled at Z=0. |
| | 1.0 | Ground plane modeled at Z=0. |

| Variable | Value | Description |
|---|---|---|
| RFF | 5.0 | Far-field-factor. (multiplies panel reference length to determine far-field radius for each panel). |
| RCORE | 0.05 | Core radius. Used when computing velocities near a doublet panel edge. This is a dimensional quantity with units consistent with global geometry. Value can be made smaller or larger to make velocity calculations more or less sensitive to panel edges. |

BINP7: Free Stream Conditions

| Variable | Value | Description |
|---|---|---|
| VINF | 1.0 | Nondimensional free stream velocity. (A velocity of 1.0 length unit/sec is used for the time-stepping portion of the code, where length unit is the global units used for the paneled geometry). |
| | R | Dimensional free stream velocity (units should agree with option selected under UNIT below and with global units used for the geometry). |
| VSOUND | R | Speed of sound (units should agree with VINF). |
| UNIT | 0 | All velocities are nondimensionalized by VINF |
| | 1 | Velocities are in (ft/sec). |
| | 2 | Velocities are in (m/sec). |
| COMPOP | | Compressibility option. |
| | 0.0 | Incompressible flow. |
| | 1.0 | Prandtl-Glauert compressibility correction. This is essentially a two-dimensional compressiblity correction that is applied to the pressure coefficients. The geometry is not transformed into compressibility coordinates. |

### BINP8: Angular Position and Rotation Rates

| Variable | Value | Description |
|---|---|---|
| ALDEG | R | Angle of attack in degrees. |
| YAWDEG | R | Yaw angle in degrees. |
| THEDOT | R | Rotation rate about Y axis. (deg/sec) |
| PSIDOT | R | Rotation rate about Z axis. (deg/sec) |
| PHIDOT | R | Rotation rate about X axis. (deg/sec) |

### BINP9: Reference Dimensions

| Variable | Value | Description |
|---|---|---|
| CBAR | R | Reference chord used for normalizing pitching moment. (units must be consistent with units used to define geometry). |
| SREF | R | Reference area for force and moment coefficients. If a plane of symmetry is used, the reference area for the paneled <u>and</u> reflected geometry should be used. (units must be consistent with units used to define geometry). |
| SSPAN | R | Semispan used for normalizing rolling and yawing moments. (units must be consistent with units used to define geometry). |
| RMPX<br>RMPY<br>RMPZ | R | Coordinates of the moment reference point in global coordinate system. |

BINP10: Special Options

| Variable | Value | Description |
|---|---|---|
| NORSET | I | The number of groups of panels on which nonzero normal velocities are to be prescribed. |
| NBCHGE | I | The number of panel neighbor information changes that are to be made. Changing the neighbor information on one side of one panel constitutes one change. |
| NCZONE | 0 | Regular external flow problem. |
|  | 1 | Internal flow problem |
| NCZPAN | I | Panel number of the panel on which the doublet value is specified for internal flow modeling. |
| CZDUB | R | The doublet value that is set on panel NCZPAN for internal flow modeling. A value of 0.0 is recommended unless convergence problems occur in the matrix solution. |
| VREF | R | The reference velocity for computing Cp in internal flow problems. If left at 0.0, then VINF will be used to compute Cp. |

BINP11: Normal Velocity Specification

| Variable | Value | Description |
|---|---|---|
| NORPCH(N) | I | Patch number of patch containing the group of panels to receive a prescribed normal velocity. |
| NORF(N) | I | Number of first and last row of panels NORL(N) in defined panel set. Using 0 defaults to all rows on this patch. |
| NOCF(N) | I | Number of first and last column of NOCL(N) panels in defined panel set. Using 0 defaults to all columns on this patch. |
| VNORM(N) | R | Specified normal velocity for the set of panels identified above. Positive direction is outwards from the surface. |

NOTE: N goes from 1 to NORSET

BINP12: Panel Neighbor Information Change

| Variable | Value | Description |
|---|---|---|
| KPAN(N)<br>KSIDE(N) | I<br>I | Panel number and the side of that panel requiring a modified neighbor. |
| NEWNAB(N)<br>NEWSID(N) | I<br>I | New neighbor and the side of that neighbor adjacent to KSIDE of KPAN. If NEWNAB is set to 0 for a particular panel, then NEWSID should be set to -KSIDE. This effectively cuts the neighbor relationship across side KSIDE. |

NOTE: N goes from 1 to NBCHGE

Input Section for Surface Geometry in PMARC

The surface geometry section of the input deck for PMARC consists of a set of namelist definitions. The required format for the surface geometry input section is shown below. Each namelist must begin with an & in the second column and the namelist name (i.e. PATCH1, SECT1, etc.) and end with &END. Blank spaces in a namelist are ignored, so the items in each namelist can be spaced in whatever fashion the user desires. A namelist can extend over as many lines as necessary. A description of each input variable and the valid input values follows. Under the Value column in the input description, the letter I means an integer value and the letter R means a real value.

The only geometry input data that does not use the namelist format is the basic point coordinate input. The basic point coordinate input is handled using a free format input. One set of three coordinates separated by at least one space must appear on each line. See the sample input below.

```
 &ASEM1     ASEMX=0.00,       ASEMY=0.00,       ASEMZ=0.00,
            ASCAL=1.00,       ATHET=0.00,       NODEA=5,                     &END
 &ASEM2     APXX=0.00,        APYY=0.00,        APZZ=0.00,
            AHXX=0.00,        AHYY=1.00,        AHZZ=0.00,                   &END

 &COMP1     COMPX=    0.0000, COMPY=    0.0000, COMPZ=    0.0000,
            CSCAL=    1.0000, CTHET=    0.0000, NODEC= 5,                    &END
 &COMP2     CPXX=   0.0000,   CPYY=   0.0000,   CPZZ=   0.0000,
            CHXX=   0.0000,   CHYY=   1.0000,   CHZZ=   0.0000,              &END

 &PATCH1    IREV=  0,    IDPAT= 1,    MAKE=   0, KCOMP= 1,  KASS= 1,       &END
 PATCH NAME
 &SECT1     STX=    0.0000, STY=    0.0000, STZ=    0.0000, SCALE=    1.0000,
            ALF=   0.0,     THETA=   0.0,
            INMODE= 1,      TNODS= 3,       TNPS=  5,       TINTS= 3,      &END
            0.0   0.0   0.0
            1.0   0.0   0.0
            1.0   0.1   0.1
            1.1   0.2   0.3
            1.3   0.5   0.7
 &BPNODE    TNODE= 3,    TNPC=  5,    TINTC=  0,                           &END
```

Description of Input Variables

## ASEM1: Assembly Coordinate System Information

| Variable | Value | Description |
|---|---|---|
| ASEMX<br>ASEMY<br>ASEMZ | R<br>R<br>R | Origin of assembly coordinate system in global coordinates. |
| ASCAL | R | Assembly scale. If ASCAL < 0, then namelist ASEM2 must be included. ASCAL < 0 allows rotation of assembly about an arbitrarily defined axis (defined on ASEM2) instead of the default assembly coordinate system Y axis. |
| ATHET | R | Rotation angle of the assembly coordinate system about the rotation axis. The default rotation axis is the assembly coordinate system Y axis. An arbitrary axis may be specified on ASEM2 if ASCAL < 0 above. Positive rotation angle is determined by Right Hand Rule. |
| NODEA | 0 | Another assembly coordinate system to be defined after this one. |
|  | 5 | This is the last assembly coordinate system to be defined. |

NOTE: Up to 10 assembly coordinate systems may be defined. One ASEM1 (and ASEM2 if required) must appear in the input deck for each assembly to be defined. Each ASEM2 that is required must follow immediately after its corresponding ASEM1. The assembly coordinate systems are numbered in the order in which they are defined.

### ASEM2: Assembly Coordinate System Rotation Axis Input

| Variable | Value | Description |
|---|---|---|
| APXX<br>APYY<br>APZZ | R<br>R<br>R | Starting point for vector defining assembly coordinate system arbitrary rotation axis. (entered in assembly coordinates (i.e., prior to scaling by assembly scale factor)). |
| AHXX<br>AHYY<br>AHZZ | R<br>R<br>R | Ending point for vector defining assembly coordinate system arbitrary rotation axis. (entered in assembly coordinates (i.e., prior to scaling by assembly scale factor)). |

### COMP1: Component Coordinate System Information

| Variable | Value | Description |
|---|---|---|
| COMPX<br>COMPY<br>COMPZ | R<br>R<br>R | Origin of component coordinate system in assembly coordinates. |
| CSCAL | R | Component scale. If CSCAL < 0, then namelist COMP2 must be included. CSCAL < 0 allows rotation of component about an arbitrarily defined axis (defined on COMP2) instead of the default component coordinate system Y axis. |
| CTHET | R | Rotation angle of the component coordinate system about the rotation axis. The default rotation axis is the component coordinate system Y axis. An arbitrary axis may be specified on COMP2 if CSCAL < 0 above. Positive rotation angle is determined by Right Hand Rule. |
| NODEC | 0 | Another component coordinate system to be defined after this one. |
| | 5 | This is the last component coordinate system to be defined. |

NOTE: Up to 10 component coordinate systems may be defined. One COMP1 (and COMP2 if required) must appear in the input deck for each component to be defined. Each COMP2 that is required must follow immediately after its corresponding COMP1.The component coordinate systems are numbered in the order in which they are defined.

COMP2: Component Coordinate System Rotation Axis Input

| Variable | Value | Description |
|---|---|---|
| CPXX<br>CPYY<br>CPZZ | R<br>R<br>R | Starting point for vector defining component coordinate system arbitrary rotation axis. (entered in component coordinates (i.e., prior to scaling by component scale factor)). |
| CHXX<br>CHYY<br>CHZZ | R<br>R<br>R | Ending point for vector defining component coordinate system arbitrary rotation axis. (entered in component coordinates (i.e., prior to scaling by component scale factor)). |

PATCH1: Patch Information

| Variable | Value | Description |
|---|---|---|
| IREV | | Patch reversal flag (for inside out patches). |
| | 0 | Patch not reversed. |
| | -1 | Patch reversed. |
| IDPAT | | Patch type. |
| | 1 | Wing type patch. Section force and moment data printed out. |
| | 2 | Body type patch. No section data printed. |
| | 3 | Neumann patch. (Vortex lattice sheet). |
| | 4 | Jet plume patch. (Computed by Adler/Baron code). This option requires JET1 namelist to follow PATCH1. Then a single SECT1 namelist follows (along with necessary basic point coordinates and BPNODE namelists) to define the perimeter of the jet exit. Only half of the jet exit is modeled, with basic point input proceeding in a counterclockwise direction when looking towards the jet exit . The values of INMODE on SECT1 are limited to between 1 and 4, inclusive, for the jet plume patch. |

| Variable | Value | Description |
|---|---|---|
| MAKE | 0 | Normal patch input (namelist SECT1 must follow). |
|  | +I | Automatic tip patch generated for side 3 of patch I. (namelist PATCH2 must follow). |
|  | -I | Automatic tip patch generated for side 1 of patch I. (namelist PATCH2 must follow). |
| KCOMP | I | Number of component coordinate system to which this patch belongs. Component coordinate systems are numbered sequentially as discussed in NOTE above on COMP1. If 0 is entered, KCOMP defaults to 1. |
| KASS | I | Number of assembly coordinate system to which this patch belongs. Assembly coordinate systems are numbered sequentially as discussed in NOTE above on ASEM1. If 0 is entered, KASS defaults to 1. Neighbor relationships are cut between patches on different assemblies. |

RECORD to be inserted after PATCH1 namelist.

| Variable | Value | Description |
|---|---|---|
| PNAME<br>(A24) | Text | Patch name |

### PATCH2: Automatic Tip Patch Generation Information (needed only if MAKE ≠ 0 on PATCH1)

| Variable | Value | Description |
|---|---|---|
| ITYP | | Tip patch type |
| | 1 | Flat tip patch |
| TNODS | 3 | More patches to follow this one. |
| | 5 | Last patch in the surface geometry input. |
| TNPS | I | Number of panels to be generated "across" the open tip. See figure 6. |
| | NOTE: | The tip patch paneling will match the edge paneling of the patch to which the tip patch is being fitted. |
| TINTS | 0 | Full cosine spacing of panels "across" the open tip, with smaller panels near outer perimeter of the tip patch. |
| | 1 | Half cosine spacing of panels with smaller panels near the first section of the tip patch. See figure 6. |
| | 2 | Half cosine spacing of panels with smaller panels near the last section of the tip patch. See figure 6. |
| | 3 | Equal spacing of panels "across" the open tip. |
| NPTTIP | 0 | This variable is not currently in use. |
| | NOTE: | This namelist completes the input required for this patch. |

## SECT1: Section Coordinate System Information

| Variable | Value | Description |
|---|---|---|
| STX | R | Origin of section coordinate system in component coordinates. |
| STY | R | |
| STZ | R | |
| SCALE | R | Section scale. If SCALE = 0.0, the defined section reduces to a single point at the origin of the section coordinate system. |
| ALF | R | Rotation angle of the section coordinate system about its Y axis. A positive rotation angle is defined by the Right Hand Rule. |
| THETA | R | Rotation angle of the section coordinate system about its Z axis. A positive rotation angle is defined by the Right Hand Ruie. |

| Variable | Value | Description |
|---|---|---|
| INMODE | 0 | Copies section definition of previous section. |
| | 1 | Input Y, Z, DX coordinates to define section. The X coordinate is defaulted to 0.0, but local deviations can be entered in DX. (basic point coordinates and BPNODE namelists follow this namelist as needed). |
| | 2 | Input X, Z, DY coordinates to define section. The Y coordinate is defaulted to 0.0, but local deviations can be entered in DY. (basic point coordinates and BPNODE namelists follow this namelist as needed). |
| | 3 | Input X, Y, DZ coordinates to define section. The Z coordinate is defaulted to 0.0, but local deviations can be entered in DZ. (basic point coordinates and BPNODE namelists follow this namelist as needed). |
| | 4 | Input X, Y, Z coordinates to define section. (basic point coordinates and BPNODE namelists follow this namelist as needed). |
| | 5 | Generate a NACA 4 digit airfoil section. (SECT2 namelist must follow this namelist). |
| | 7 | Input R, θ, X coordinates to define section. R is measured perpendicular to the section X axis and θ is measured from the section +Y axis with the positive angular direction defined by the Right Hand Rule. (basic point coordinates and BPNODE namelists follow this namelist as needed). |

| Variable | Value | Description |
|---|---|---|
| TNODS | 0 | First or intermediate section of patch. |
|  | 1 | Break point on patch with continuous slope into the next region of patch. |
|  | 2 | Break point on patch with discontinuous slope into the next region of patch. |
|  | 3 | Last section definition on this patch. |
|  | 5 | Last section definition on last patch of surface geometry. |
| TNPS | I | Number of panels to be generated between this break point and the previous break point (or the first section of this patch if this is the first or only break point). If TNPS = 0 at a break point, the input sections between this break point and the previous one will be used to define the panel edges. |
| TINTS | 0 | Full-cosine spacing of panels between this break point and the previous one, with smaller panels near the two break points. |
|  | 1 | Half-cosine spacing of panels between this break point and the previous one, with smaller panels near the previous break point. |
|  | 2 | Half-cosine spacing of panels between this break point and the previous one, with smaller panels near this break point. |
|  | 3 | Equal spacing of panels between this break point and the previous break point. |

## SECT2: NACA 4 digit airfoil section generation information (needed only if INMODE = 5 on SECT1)

| Variable | Value | Description |
|---|---|---|
| RTC | R | The thickness to chord ratio for the airfoil. |
| RMC | R | The maximum chamber to chord ratio for the airfoil. |
| RPC | R | The chordwise position of the maximum chamber (expressed as a ratio to chord). |
| IPLANE | | The plane in the section coordinate system used to generate the airfoil coordinates. |
| | 1 | The YZ plane. |
| | 2 | The XZ plane. |
| | 3 | The XY plane. |
| TNPC | I | The number of panels to be distributed between the trailing edge and the leading edge of the airfoil. The same number of panels are distributed on the upper and lower surfaces. |
| TINTC | | The type of panel spacing to be used on the upper and lower surfaces of the airfoil. |
| | 0 | Full-cosine spacing with smaller panels near the leading and trailing edges.(This is the recommended spacing). |
| | 1 | Half-cosine spacing with smaller panels near the trailing edge. |
| | 2 | Half-cosine spacing with smaller panels near the leading edge. |
| | 3 | Equal spacing. |

RECORD : Section Basic Point Coordinate Input (This record is repeated for each basic point defining this section)

| Variable | Value | Description |
|---|---|---|
| B1 | R | Basic point coordinates for section definition The values that go in B1, B2, B3 depend on the value of INMODE on SECT1. |
| B2 | R | |
| B3 | R | |

NOTE: The values of B1, B2, B3 are entered as triplets in free format, with at least one space separating each value. One triplet is entered per line.

BPNODE: Break Point Input (inserted between basic point coordinates on a section definition as needed. Must terminate basic point input for a section with a BPNODE namelist)

| Variable | Value | Description |
|---|---|---|
| TNODE | 0 | First or intermediate point (i.e., not a break point. Values entered for TNPC and TINTC are ignored). |
| | 1 | Break point with continuous slope into the next region on this section. |
| | 2 | Break point with discontinuous slope into the next region on this section |
| | 3 | Final break point. End of this section definition. |
| TNPC | I | Number of panels to be generated between this break point and the previous one (or the first point of the section definition if this is the first or only break point). If TNPC = 0 at a break point, the input points will be used as the panel corner points between this break point and the previous one. |
| | NOTE: | The total number of panels to be generated on each section of a given patch must be the same. |

| Variable | Value | Description |
|---|---|---|
| TINTC | 0 | Full cosine spacing of panels between this break point and the previous one, with smaller panels near the two break points. |
| | 1 | Half cosine spacing of panels between this break point and the previous one, with smaller panels near the previous break point. |
| | 2 | Half cosine spacing of panels between this break point and the previous one, with smaller panels near this break point. |
| | 3 | Equal spacing of panels between this break point and the previous break point. |

JET1: Jet Plume Generation Information (needed only if IDPAT=4 on PATCH1).

| Variable | Value | Description |
|---|---|---|
| VJET | R | The jet exit velocity. Units must be consistent with VINF. |
| NJDS | I | The number of jet diameters the jet plume is to be extended downstream. |
| DZ0 | R | The step size (in jet diameters or fraction of a jet diameter) for moving down the jet plume and computing the jet parameters. |
| JETIN | I | The number of the panel set with prescribed normal velocity (i.e. panel set #1, #2, #3, etc. under the NORSET option in the basic data input) which corresponds to the inlet for this jet. If there is no inlet for this jet, just enter 0. |
| | NOTE: | The minimum number of columns of panels that will be computed for the jet plume patch can be estimated as:<br>NJDS/DZ0 + 1<br>There is currently a limit of 50 columns of panels that can be computed for the jet plume patch. Thus NJDS and DZ0 must be set with this limit in mind. |

Input Section for Time-stepping Wakes in PMARC

The wake geometry section of the input deck for PMARC consists of a set of namelist definitions. The required format for the wake geometry input section is shown below. Each namelist must begin with an & in the second column and the namelist name (i.e., WAKE1, SECT1, etc.) and end with &END. Blank spaces in a namelist are ignored, so the items in each namelist can be spaced in whatever fashion the user desires. A namelist can extend over as many lines as necessary. A description of each input variable and the valid input values follows. Under the Value column in the input description, the letter I means an integer value and the letter R means a real value.

The only wake input data that does not use the namelist format is the basic point coordinate input. The basic point coordinate input is handled using a free format input. One set of three coordinates separated by at least one space must appear on each line. See the sample input below.

```
 &WAKE1      IDWAK=1,          IFLXW=0,                                                   &END
  TYPE WAKE  NAME HERE
 &WAKE2      KWPACH=1,         KWSIDE=2,         KWLINE=0,         KWPAN1=0,
             KWPAN2=0,         NODEW=5,          INITIAL=1,                               &END
 &SECT1      STX=     0.0000,  STY=     0.0000,  STZ=     0.0000,  SCALE= 1.0000,
             ALF=    0.0,      THETA=   0.0,
             INMODE= 4,        TNODS= 3,         TNPS=  10,        TINTS= 3,   &END
             0.0   0.0   0.0
             0.0   1.0   0.0
             0.0   2.0   0.0
             0.0   5.0   0.0
 &BPNODE     TNODE= 3,    TNPC=  10,    TINTC=  3,                                        &END
```

Description of input variables

WAKE1: Wake identification

| Variable | Value | Description |
|---|---|---|
| IDWAK | | Wake type |
| | 0 | No wakes |
| | 1 | Regular wake |
| IFLXW | 0 | Flexible wake. Wake will be time-stepped with the local velocity. |
| | 1 | Rigid wake. Wake will be time-stepped with the free-stream velocity only. |

RECORD: Wake Name (record to be inserted immediately following WAKE1 namelist).

| Variable | Value | Description |
|---|---|---|
| WNAME<br>(A24) | | Text identifying the wake |

WAKE2: Wake Separation Line Information

| Variable | Value | Description |
|---|---|---|
| KWPACH | I | Surface geometry patch number that this wake separates from. If this wake separates from more than one patch, then additional WAKE2 namelists must be included for each patch this wake separates from. |
| KWSIDE | I | Side of the patch which is parallel to separation line. Separation line will be in same "direction" as the patch side specified (see fig. 8). |
| KWLINE | I | Row or column number within patch from which the wake separates. The side of the panels on row or column KWLINE from which the wake separates will be the same as KWSIDE. If KWLINE=0, separation is from patch edge (see fig. 4 for patch nomenclature). |
| KWPAN1 | I | Number of first panel on row or column from which wake separates (numbered locally on row or column, i.e., the first panel on the row or column is 1, the second is 2, etc.). KWPAN1=0 defaults to the first panel on the row or column. |
| KWPAN2 | I | Number of last panel on row or column from which wake separates (numbered locally on row or column). KWPAN2=0 defaults to the last panel on the row or column. |

| Variable | Value | Description |
|---|---|---|
| NODEW | 0 | Indicates that another WAKE2 namelist will follow to continue the wake separation line definition for this wake. |
| | 3 | Indicates this wake separation line definition is complete and there are more wakes to be defined after this wake. |
| | 5 | Indicates this wake separation line definition is complete and this is the last wake to be defined. |
| INITIAL | 0 | No initial wake geometry to be specified. |
| | 1 | Initial wake geometry to be specified. (SECT1 namelist must follow this namelist). |

NOTE: When specifying a wake which separates from more than one patch, the order in which the separation patches (KWPACH) are input must be such that a single continuous separation line is defined.

SECT1: Section Coordinate System Information

| Variable | Value | Description |
|---|---|---|
| STX<br>STY<br>STZ | R<br>R<br>R | Origin of section coordinate system in global coordinates. |
| SCALE | R | Section scale. If SCALE=0.0, the section reduces to a point at the section coordinate system origin. |
| ALF | R | Rotation angle of the section coordinate system about its Y axis. A positive rotation angle is defined by the Right-Hand Rule. |
| THETA | R | Rotation angle of the section coordinate system about its Z axis. A positive rotation angle is defined by the Right-Hand Rule. |

| Variable | Value | Description |
|---|---|---|
| INMODE | -1 | Copies the basic point coordinates of previous section and the values entered for STX, STY, and STZ on this section are displacement distances from the origin of the previous section. |
| | 0 | Copies the basic point coordinates of previous section. |
| | 1 | Input Y, Z, DX coordinates to define section. The X coordinate is defaulted to 0.0, but local deviations can be entered in DX. (basic point coordinates and BPNODE namelists follow this namelist as needed). |
| | 2 | Input X, Z, DY coordinates to define section. The Y coordinate is defaulted to 0.0, but local deviations can be entered in DY. (basic point coordinates and BPNODE namelists follow this namelist as needed). |
| | 3 | Input X, Y, DZ coordinates to define section. The Z coordinate is defaulted to 0.0, but local deviations can be entered in DZ. (basic point coordinates and BPNODE namelists follow this namelist as needed). |
| | 4 | Input X, Y, Z coordinates to define section. (basic point coordinates and BPNODE namelists follow this namelist as needed). |

| Variable | Value | Description |
|---|---|---|
| TNODS | 0 | First or intermediate section of wake. |
| | 1 | Break point on wake with continuous slope into the next region of wake. |
| | 2 | Break point on wake with discontinuous slope into the next region of wake. |
| | 3 | Last section definition on this wake. |
| TNPS | I | Number of panels to be generated between this break point and the previous break point (or the first section of this wake if this is the first or only break point). If TNPS = 0 at a break point, the input sections between this break point and the previous one will be used to define the panel edges. |
| TINTS | 0 | Full-cosine spacing of panels between this break point and the previous one, with smaller panels near the two break points. |
| | 1 | Half-cosine spacing of panels between this break point and the previous one, with smaller panels near the previous break point. |
| | 2 | Half-cosine spacing of panels between this break point and the previous one, with smaller panels near this break point. |
| | 3 | Equal spacing of panels between this break point and the previous break point. |

RECORD : Section Basic Point Coordinate Input (This record is repeated for each basic point defining this section)

| Variable | Value | Description |
|---|---|---|
| B1 | R | Basic point coordinates for section definition. The values that go in B1, B2, B3 depend on the value of INMODE on SECT1. |
| B2 | R | |
| B3 | R | |

NOTE: The values of B1, B2, B3 are entered as triplets in free format, with at least one space separating each value. One triplet is entered per line.

BPNODE: Break Point Input (inserted between basic point coordinates on a section definition as needed. Must terminate basic point input for a section with a BPNODE namelist)

| Variable | Value | Description |
|---|---|---|
| TNODE | 0 | First or intermediate point (i.e., not a break point. Values entered for TNPC and TINTC are ignored). |
| | 1 | Break point with continuous slope into the next region on this section. |
| | 2 | Break point with discontinuous slope into the next region on this section. |
| | 3 | Final break point. End of this section definition. |
| TNPC | I | Number of panels to be generated between this break point and the previous one (or the first point of the section definition if this is the first or only break point). If TNPC = 0 at a break point, the input points will be used as the panel corner points between this break point and the previous one. |
| | NOTE: | The total number of panels to be generated on each section of this wake must be the same as the total number of surface geometry panels that this wake separates from. |

| Variable | Value | Description |
|---|---|---|
| TINTC | 0 | Full-cosine spacing of panels between this break point and the previous one. with smaller panels near the two break points. |
| | 1 | Half-cosine spacing of panels between this break point and the previous one. with smaller panels near the previous break point. |
| | 2 | Half-cosine spacing of panels between this break point and the previous one, with smaller panels near this break point. |
| | 3 | Equal spacing of panels between this break point and the previous break point. |

## Input Section for Special Options in PMARC

Onbody streamlines and boundary layer analysis are not currently functional in PMARC because these routines are in the process of being replaced.

### Off-body velocity scan input section

### Description of Input Variables

The off-body velocity scan input data follows immediately after the end of the wake input section. The off-body velocity scan input section of PMARC consists of a set of namelist definitions. The required format for the velocity scan input section is shown below. The best way to handle the velocity scan input section is to create a template file which can then be included into any PMARC file and the values modified appropriately. All the namelists should always be included as shown below whether or not a particular namelist is needed for the job being run. If a namelist is not needed for a particular job, PMARC merely skips over that namelist. Each namelist must begin with an & in the second column and the namelist name (i.e., VS1, VS2, etc.) and end with &END. Blank spaces in a namelist are ignored, so items in each namelist can be spaced in whatever fashion the user desires. A namelist can extend over as many lines as necessary. A description of each input variable and the valid input values follows. Under the Value column in the input description, the letter I means an integer value and the letter R means a real value.

NOTE: Variables in the namelist definition which are arrays should have their elements listed out following the variable name. For example, if there were three values to be entered into the array X0, the input would be as follows: X0 = R1,R2,R3. The rest of the elements in array X0 will automatically be left at zero.

```
 &VS1       NVOLR= 1,        NVOLC= 1,                                       &END
 &VS2       X0=    -2.0000,  Y0=      0.0000,  Z0=     -2.0000,               &END
 &VS3       X1=     2.0000,  Y1=      0.0000,  Z1=     -2.0000,  NPT1= 20,    &END
 &VS4       X2=    -2.0000,  Y2=      0.0000,  Z2=     -2.0000,  NPT2=  0,    &END
 &VS5       X3=    -2.0000,  Y3=      0.0000,  Z3=      2.0000,  NPT3= 40,    &END

 &VS6       XR0=    0.0000,  YR0=     0.0000,  ZR0=     0.0000,               &END
 &VS7       XR1=    0.0000,  YR1=    10.0000,  ZR1=     0.0000,
            XR2=    0.0000,  YR2=     0.0000,  ZR2=     1.0000,               &END
 &VS8       R1=     0.5000,  R2=      5.0000,  PHI1=   0.0,      PHI2=330.0,  &END
 &VS9       NRAD=  10,       NPHI=   12,       NLEN=   5,                     &END
```

VS1:

| Variable | Value | Description |
|---|---|---|
| NVOLR | I | Number of rectangular scan volumes |
| NVOLC | I | Number of cylindrical scan volumes |

VS2:

| Variable | Value | Description |
|---|---|---|
| X0(N)<br>Y0(N)<br>Z0(N) | R<br>R<br>R | Coordinates of origin of rectangular scan volume N. See figure 9. |

VS3:

| Variable | Value | Description |
|---|---|---|
| X1(N)<br>Y1(N)<br>Z1(N) | R<br>R<br>R | Coordinates of corner in i direction for rectangular scan volume N. See figure 9. |
| NPT1(N) | I | Number of scan points to be distributed along side i of scan volume N. |

VS4:

| Variable | Value | Description |
|---|---|---|
| X2(N)<br>Y2(N)<br>Z2(N) | R<br>R<br>R | Coordinates of corner in j direction for rectangular scan volume N. See figure 9. |
| NPT2(N) | I | Number of scan points to be distributed along side j of scan volume N. |

VS5:

| Variable | Value | Description |
|---|---|---|
| X3(N) | R | Coordinates of corner in **k** direction for rectangular scan volume N. See figure 9. |
| Y3(N) | R | |
| Z3(N) | R | |
| NPT3(N) | I | Number of scan points to be distributed along side **k** of scan volume N. |

NOTE: N goes from 1 to NVOLR
NOTE: If NPT1, NPT2, or NPT3 is zero, the corresponding side of the rectangular scan volume collapses to a point. Thus a rectangular scan volume can be reduced to a plane, a line, or a point.

VS6:

| Variable | Value | Description |
|---|---|---|
| XR0(N) | R | Coordinates of origin of cylindrical scan volume N. See figure 10. |
| YR0(N) | R | |
| ZR0(N) | R | |

VS7:

| Variable | Value | Description |
|---|---|---|
| XR1(N) | R | Coordinates of point defining axis (from XR0, YR0, ZR0) of cylindrical scan volume N. (Cannot be XR0, YR0, ZR0). See figure 10. |
| YR1(N) | R | |
| ZR1(N) | R | |
| XR2(N) | R | Coordinates of point defining vector (from XR0, YR0, ZR0) from which PHI is measured for scan volume N. See figure 10. |
| YR2(N) | R | |
| ZR2(N) | R | |

VS8:

| Variable | Value | Description |
|---|---|---|
| R1(N) | R | Inner radius of cylindrical scan volume N. |
| R2(N) | R | Outer radius of cylindrical scan volume N. |
| PHI1(N) | R | Starting angle (measured from the vector (XR2-XR0),(YR2-YR0), (ZR2-ZR0)) for cylindrical scan volume N. Positive angle is determined by the Right Hand Rule. |
| PHI2(N) | R | Ending angle (measured from the vector (XR2-XR0),(YR2-YR0), (ZR2-ZR0)) for cylindrical scan volume N. Positive angle is determined by the Right Hand Rule. |

VS9:

| Variable | Value | Description |
|---|---|---|
| NRAD(N) | I | Number of points to be distributed in the radial direction for cylindrical scan volume N. |
| NPHI(N) | I | Number of points to be distributed in the f direction for cylindrical scan volume N. |
| NLEN(N) | I | Number of points to be distributed in the axial direction for cylindrical scan volume N. |

NOTE: N goes from 1 to NVOLC
NOTE: The cylindrical scan volume can be reduced to a plane, a line, or a point by setting NLEN, NPHI, or NRAD equal to zero.

## Off-body streamline input section

### Description of Input Variables

The off-body streamline input data must follow immediately after the off-body velocity scan data. The off-body streamline input section of PMARC consists of a namelist which defines the number of streamlines there will be for the job and a namelist definition which is repeated for each separate streamline. The required format for the off-body streamline input section is shown below. The best way to handle the off-body streamline input section is to create a template file with a single streamline which can then be included into any PMARC file and the values modified appropriately. Both of the namelists shown below should always be included in the input deck, whether or not there will be any off-body streamlines. If a namelist is not needed for a particular job, PMARC merely skips over that namelist. Each namelist must begin with an & in the second column and the namelist name (i.e., SLIN1, SLIN2, etc.) and end with &END. Blank spaces in a namelist are ignored, so the items in each namelist can be spaced in whatever fashion the user desires. A namelist can extend over as many lines as necessary. A description of each input variable and the valid input values follows. Under the Value column in the input description, the letter I means an integer value and the letter R means a real value.

```
 &SLIN1     NSTLIN=1,                                                   &END
 &SLIN2     SX0=    -3.0000,  SY0=      0.0000,  SZ0=      0.0500,
            SU=      0.0000,  SD=       6.5000,  DS=       0.0250,      &END
```

SLIN1:

| Variable | Value | Description |
|---|---|---|
| NSTLIN | I | Number of streamlines to be defined. |

SLIN2:

| Variable | Value | Description |
|---|---|---|
| SX0 | R | Global coordinates for starting point of streamline. |
| SY0 | R | |
| SZ0 | R | |
| SU | R | Distance streamline to be traced in upstream direction (same units of length as geometry). |
| SD | R | Distance streamline to be traced in downstream direction (same units of length as geometry). |
| DS | R | Step size to be used in tracing streamline (D distance) |

NOTE: Record SLIN2 must be repeated NSTLIN times (one for each streamline).

# APPENDIX D

## SYMMETRIC WING/BODY TEST CASE INPUT FILE

```
WING BODY COMBINATION TEST CASE

&BINP2     LSTINP=2,     LSTOUT=0,      LSTFRQ=1,      LENRUN=0,      &END
&BINP3     LSTGEO=0,     LSTNAB=0,      LSTWAK=3,      LSTCPV=0,
           LSTJET=0,                                                  &END
&BINP4     MAXIT=200,    SOLRES=0.0005,                               &END
&BINP5     NTSTPS=3,     DTSTEP=3.0,                                  &END
&BINP6     RSYM=0.0,     RGPR=0.0,      RFF=5.0,       RCORE=0.10,    &END
&BINP7     VINF=1.0,     VSOUND=1116.0,       UNIT=0,
           COMPOP=0.0,                                                &END
&BINP8     ALDEG=4.0,    YAWDEG=0.0,  THEDOT=0.0,    PSIDOT=0.0,
           PHIDOT=0.0,                                                &END
&BINP9     CBAR=6.125,   SREF=147.0,    SSPAN=12.0,
           RMPX=9.00,    RMPY=0.00,     RMPZ=0.00,                    &END
&BINP10    NORSET=0,     NBCHGE=0,      NCZONE=0,
           NCZPAN=0,     CZDUB=0.0,     VREF=00.0,                    &END
&BINP11    NORPCH=0,
           NORF=0,       NORL=0,
           NOCF=0,       NOCL=0,
           VNORM=0.0,                                                 &END
&BINP12    KPAN=0,       KSIDE=0,       NEWNAB=0,      NEWSID=0,      &END



&ASEM1     ASEMX=        0.0000,  ASEMY=        0.0000,  ASEMZ=        0.0000,
           ASCAL=        1.0000,  ATHET=        0.0,     NODEA=        0,
                                                                      &END
&ASEM2     APXX=         0.0000,  APYY=         0.0000,  APZZ=0.0000,
           AHXX=         0.0000,  AHYY=         1.0000,  AHZZ=0.0000,  &END

&ASEM1     ASEMX=        0.0000,  ASEMY=        0.0000,  ASEMZ=        0.0000,
           ASCAL=        1.0000,  ATHET=        0.0,     NODEA=        5,
                                                                      &END
&ASEM2     APXX=         0.0000,  APYY=         0.0000,  APZZ=0.0000,
           AHXX=         0.0000,  AHYY=         1.0000,  AHZZ=0.0000,  &END

&COMP1     COMPX=        0.0000,  COMPY=        0.0000,  COMPZ=        0.0000,
           CSCAL=        1.0000,  CTHET=        0.0,     NODEC=        5,
                                                                      &END
&COMP2     CPXX=         0.0000,  CPYY=         0.0000,  CPZZ= 0.0000,
           CHXX=         0.0000,  CHYY=         1.0000,  CHZZ=0.0000,  &END
```

```
&PATCH1      IREV= 0,     IDPAT=       1,       MAKE=        0,
             KCOMP=       1,      KASS= 1,
    WING                                                         &END
&SECT1       STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=    1.0000,
             ALF=  0.0,    THETA=        0.0,
             INMODE=      4,      TNODS=       0,      TNPS= 0,
             TINTS=       0,
21.4641      1.6481       0.0000                                 &END
21.3866      1.6482       -0.0059
21.1575      1.6484       -0.0228
20.7868      1.6488       -0.0493
20.2908      1.6495       -0.0847
19.6855      1.6440       -0.1260
18.9955      1.6313       -0.1662
18.2571      1.6195       -0.1979
17.5037      1.6103       -0.2125
16.7691      1.6043       -0.2083
16.0776      1.5933       -0.1902
15.4628      1.5811       -0.1616
14.9568      1.5731       -0.1252
14.5818      1.5692       -0.0855
14.3560      1.5698       -0.0434
14.2985      1.5751       0.0000
14.3560      1.5698       0.0434
14.5818      1.5692       0.0855
14.9568      1.5731       0.1252
15.4628      1.5811       0.1616
16.0776      1.5933       0.1902
16.7691      1.6043       0.2083
17.5037      1.6103       0.2125
18.2571      1.6195       0.1979
18.9955      1.6313       0.1662
19.6855      1.6440       0.1260
20.2908      1.6495       0.0847
20.7868      1.6488       0.0493
21.1575      1.6484       0.0228
21.3866      1.6482       0.0059
21.4641      1.6481       0.0000
&BPNODE      TNODE=       3,      TNPC= 0,     TINTC=       0,     &END
```

```
&SECT1      STX=  25.3750,       STY=  12.0000,     STZ=  0.0000,
SCALE=      4.5000,        ALF=  0.0,   THETA=      0.0,
INMODE=     2,    TNODS=         3,     TNPS= 10,   TINTS=        0,
                                                                  &END
1.0000      0.00000       0.0
0.9500      -0.00370      0.0
0.9000      -0.00727      0.0
0.8500      -0.01083      0.0
0.8000      -0.01437      0.0
0.7500      -0.01775      0.0
0.7000      -0.02087      0.0
0.6500      -0.02364      0.0
0.6000      -0.02602      0.0
0.5500      -0.02793      0.0
0.5000      -0.02925      0.0
0.4500      -0.02992      0.0
0.4000      -0.02996      0.0
0.3500      -0.02945      0.0
0.3000      -0.02842      0.0
0.2500      -0.02687      0.0
0.2000      -0.02474      0.0
0.1500      -0.02194      0.0
0.1000      -0.01824      0.0
0.0750      -0.01591      0.0
0.0500      -0.01313      0.0
0.0250      -0.00981      0.0
0.0125      -0.00718      0.0
0.0075      -0.00563      0.0
0.005       -0.00464      0.0
0.0         0.0           0.0
&BPNODE     TNODE=        2,     TNPC= 15,    TINTC=       0,     &END
0.0         0.0           0.0
0.005       0.00464       0.0
0.0075      0.00563       0.0
0.0125      0.00718       0.0
0.0250      0.00981       0.0
0.0500      0.01313       0.0
0.0750      0.01591       0.0
0.1000      0.01824       0.0
0.1500      0.02194       0.0
0.2000      0.02474       0.0
0.2500      0.02687       0.0
0.3000      0.02842       0.0
0.3500      0.02945       0.0
0.4000      0.02996       0.0
0.4500      0.02992       0.0
0.5000      0.02925       0.0
0.5500      0.02793       0.0
0.6000      0.02602       0.0
0.6500      0.02364       0.0
0.7000      0.02087       0.0
0.7500      0.01775       0.0
0.8000      0.01437       0.0
0.8500      0.01083       0.0
0.9000      0.00727       0.0
0.9500      0.00370       0.0
1.0000      0.00000       0.0
&BPNODE     TNODE=        3,     TNPC= 15,    TINTC=       0,     &END
```

```
&PATCH1      IREV= 0,      IDPAT=       1,     MAKE=         1,
             KCOMP=        1,      KASS= 2,                              &END
     WING TIP
&PATCH2      ITYP= 1,      TNODS=       3,     TNPS= 3,        TINTS= 3,
                                                                         &END


&PATCH1      IREV= 0,      IDPAT=       2,     MAKE=         0,
             KCOMP=        1,      KASS= 1,                              &END
     FORWARD FUSELAGE
&SECT1       STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=       0.0000,
             ALF=  0.0,    THETA=      0.0,
             INMODE=       1,      TNODS=      0,      TNPS= 0,
             TINTS= 0,                                               &END
0.0000       -1.0000       0.0000
0.1735       -0.9845       0.0000
0.3420       -0.9395       0.0000
0.5000       -0.8660       0.0000
0.6425       -0.7660       0.0000
0.7660       -0.6425       0.0000
0.8660       -0.5000       0.0000
0.9395       -0.3420       0.0000
0.9845       -0.1735       0.0000
1.0000       0.0000        0.0000
0.9845       0.1735        0.0000
0.9395       0.3420        0.0000
0.8660       0.5000        0.0000
0.7660       0.6425        0.0000
0.6425       0.7660        0.0000
0.5000       0.8660        0.0000
0.3420       0.9395        0.0000
0.1735       0.9845        0.0000
0.0000       1.0000        0.0000
&BPNODE      TNODE=        3,      TNPC=  10,     TINTC=        3,    &END
&SECT1       STX=  0.2000, STY=  0.0000, STZ=  0.0000, SCALE=       0.0924,
             ALF=  0.0,    THETA=      0.0,
             INMODE=       0,      TNODS=      0,      TNPS= 0,
             TINTS=        0,                                        &END
&SECT1       STX=  0.5000, STY=  0.0000, STZ=  0.0000, SCALE=       0.1712,
             ALF=  0.0,    THETA=      0.0,
             INMODE=       0,      TNODS=      0,      TNPS= 0,
             TINTS=        0,                                        &END
&SECT1       STX=  1.0000, STY=  0.0000, STZ=  0.0000, SCALE=       0.2889,
             ALF=  0.0,    THETA=      0.0,
             INMODE=       0,      TNODS=      0,      TNPS= 0,
             TINTS=        0,                                        &END
&SECT1       STX=  2.0000, STY=  0.0000, STZ=  0.0000, SCALE=       0.4820,
             ALF=  0.0,    THETA=      0.0,
             INMODE=       0,      TNODS=      0,      TNPS= 0,
             TINTS=        0,                                        &END
&SECT1       STX=  4.0000, STY=  0.0000, STZ=  0.0000, SCALE=       0.7884,
             ALF=  0.0,    THETA=      0.0,
             INMODE=       0,      TNODS=      0,      TNPS= 0,
             TINTS=        0,                                        &END
&SECT1       STX=  8.0000, STY=  0.0000, STZ=  0.0000, SCALE=       1.2360,
             ALF=  0.0,    THETA=      0.0,
             INMODE=       0,      TNODS=      0,      TNPS= 0,
             TINTS=        0,                                        &END
```

```
&SECT1      STX=   12.0000,      STY=  0.0000,  STZ=   0.0000,  SCALE=1.49[illegible],
            ALF=  0.0,     THETA=        0.0,
            INMODE=        0,     TNODS=        0,      TNPS= 0,
            TINTS=         0,                                         &END
&SECT1      STX=  0.0000,  STY=  0.0000,  STZ=   0.0000,  SCALE=      1.0000,
            ALF=  0.0,     THETA=        0.0,
            INMODE=        4,     TNODS=        3,      TNPS= 7,
            TINTS=         0,                                         &END
14.2983     0.0000         -1.5829
14.2983     0.4893         -1.5050
14.2983     0.9307         -1.2802
14.2983     1.2808         -0.9297
14.2983     1.5053         -0.4881
14.2985     1.5751         0.0000
14.2983     1.5048         0.4900
14.2983     1.2800         0.9309
14.2983     0.9299         1.2807
14.2983     0.4888         1.5051
14.2983     0.0000         1.5829
&BPNODE     TNODE=         3,     TNPC=  0,     TINTC=        0,      &END


&PATCH1     IREV= 0,       IDPAT=        2,     MAKE=         0,
            KCOMP=         0,     KASS= 0,                            &END
      LOWER MID FUSELAGE
&SECT1      STX=  0.0000,  STY=  0.0000,  STZ=   0.0000,  SCALE=      1.0000,
            ALF=  0.0,     THETA=        0.0,
            INMODE=        4,     TNODS=        0,      TNPS= 0,
            TINTS=         0,                                         &END
14.2983     0.0000         -1.5829
14.2983     0.4893         -1.5050
14.2983     0.9307         -1.2802
14.2983     1.2808         -0.9297
14.2983     1.5053         -0.4881
14.2985     1.5751         0.0000
&BPNODE     TNODE=         3,     TNPC=  0,     TINTC=        0,      &END
&SECT1      STX=  0.0000,  STY=  0.0000,  STZ=   0.0000,  SCALE=      1.0000,
            ALF=  0.0,     THETA=        0.0,
            INMODE=        4,     TNODS=        0,      TNPS= 0,
            TINTS=         0,                                         &END
14.3558     0.0000         -1.5846
14.3558     0.4811         -1.5091
14.3558     0.9169         -1.2919
14.3558     1.2664         -0.9527
14.3558     1.4962         -0.5234
14.3560     1.5698         -0.0434
&BPNODE     TNODE=         3,     TNPC=  0,     TINTC=        0,      &END
&SECT1      STX=  0.0000,  STY=  0.0000,  STZ=   0.0000,  SCALE=      1.0000,
            ALF=  0.0,     THETA=        0.0,
            INMODE=        4,     TNODS=        0,      TNPS= 0,
            TINTS=         0,                                         &END
14.5815     0.0000         -1.5912
14.5815     0.4751         -1.5176
14.5815     0.9070         -1.3067
14.5815     1.2565         -0.9767
14.5815     1.4913         -0.5575
14.5818     1.5692         -0.0855
&BPNODE     TNODE=         3,     TNPC=  0,     TINTC=        0,      &END
```

```
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=     1.0000.
            ALF=  0.0,    THETA=        0.0,
            INMODE=       4,     TNODS=     0,      TNPS= 0,
            TINTS=        0,                                     &END
14.9565     0.0000        -1.6014
14.9565     0.4707        -1.5295
14.9565     0.8999        -1.3239
14.9565     1.2500        -1.0016
14.9565     1.4896        -0.5908
14.9568     1.5731        -0.1252
&BPNODE     TNODE=        3,     TNPC=  0,  TINTC=       0,      &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=     1.0000,
            ALF=  0.0,    THETA=        0.0,
            INMODE=       4,     TNODS=     0,      TNPS= 0,
            TINTS=        0,                                     &END
15.4626     0.0000        -1.6139
15.4626     0.4676        -1.5432
15.4626     0.8952        -1.3420
15.4626     1.2463        -1.0260
15.4626     1.4904        -0.6219
15.4628     1.5811        -0.1616
&BPNODE     TNODE=        3,     TNPC=  0,  TINTC=       0,      &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=     1.0000,
            ALF=  0.0,    THETA=        0.0,
            INMODE=       4,     TNODS=     0,      TNPS= 0,
            TINTS=        0,                                     &END
16.0776     0.0000        -1.6271
16.0776     0.4664        -1.5573
16.0776     0.8935        -1.3589
16.0776     1.2461        -1.0470
16.0776     1.4940        -0.6472
16.0776     1.5933        -0.1902
&BPNODE     TNODE=        3,     TNPC=  0,  TINTC=       0,      &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=     1.0000,
            ALF=  0.0,    THETA=        0.0,
            INMODE=       4,     TNODS=     0,      TNPS= 0,
            TINTS=        0,                                     &END
16.7691     0.0000        -1.6396
16.7691     0.4668        -1.5701
16.7691     0.8948        -1.3730
16.7691     1.2492        -1.0627
16.7691     1.5000        -0.6645
16.7691     1.6043        -0.2083
&BPNODE     TNODE=        3,     TNPC=  0,  TINTC=       0,      &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=     1.0000,
            ALF=  0.0,    THETA=        0.0,
            INMODE=       4,     TNODS=     0,      TNPS= 0,
            TINTS=        0,                                     &END
17.5036     0.0000        -1.6507
17.5036     0.4693        -1.5808
17.5036     0.8997        -1.3830
17.5036     1.2563        -1.0715
17.5036     1.5089        -0.6716
17.5037     1.6103        -0.2125
&BPNODE     TNODE=        3,     TNPC=  0,  TINTC=       0,      &END
```

```
&SECT1      STX=   0.0000,  STY=   0.0000,  STZ=    0.0000,  SCALE=        1.0000,
            ALF=   0.0,     THETA=          0.0,
            INMODE=         4,      TNODS=          0,      TNPS= 0,
            TINTS=          0,                                             &END
18.2570     0.0000          -1.6594
18.2570     0.4747          -1.5884
18.2570     0.9097          -1.3869
18.2570     1.2691          -1.0699
18.2570     1.5221          -0.6635
18.2571     1.6195          -0.1979
&BPNODE     TNODE=          3,      TNPC=  0,       TINTC=          0,     &END
&SECT1      STX=   0.0000,  STY=   0.0000,  STZ=    0.0000,  SCALE=        1.0000,
            ALF=   0.0,     THETA=          0.0,
            INMODE=         4,      TNODS=          0,      TNPS= 0,
            TINTS=          0,                                             &END
18.9955     0.0000          -1.6650
18.9955     0.4825          -1.5920
18.9955     0.9236          -1.3844
18.9955     1.2860          -1.0582
18.9955     1.5377          -0.6413
18.9955     1.6313          -0.1662
&BPNODE     TNODE=          3,      TNPC=  0,       TINTC=          0,     &END
&SECT1      STX=   0.0000,  STY=   0.0000,  STZ=    0.0000,  SCALE=        1.0000,
            ALF=   0.0,     THETA=          0.0,
            INMODE=         4,      TNODS=          0,      TNPS= 0,
            TINTS=          0,                                             &END
19.6855     0.0000          -1.6670
19.6855     0.4910          -1.5918
19.6855     0.9385          -1.3769
19.6855     1.3032          -1 0401
19.6855     1.5522          -0.6110
19.6855     1.6440          -0.1260
&BPNODE     TNODE=          3,      TNPC=  0,       TINTC=          0,     &END
&SECT1      STX=   0.0000,  STY=   0.0000,  STZ=    0.0000,  SCALE=        1.0000,
            ALF=   0.0,     THETA=          0.0,
            INMODE=         4,      TNODS=          0,      TNPS= 0,
            TINTS=          0,                                             &END
20.2908     0.0000          -1.6660
20.2908     0.4985          -1.5887
20.2908     0.9515          -1.3669
20.2908     1.3177          -1.0199
20.2908     1.5630          -0.5794
20.2908     1.6495          -0.0847
&BPNODE     TNODE=          3,      TNPC=  0,       TINTC=          0,     &END
&SECT1      STX=   0.0000,  STY=   0.0000,  STZ=    0.0000,  SCALE=        1.0000,
            ALF=   0.0,     THETA=          0.0,
            INMODE=         4,      TNODS=          0,      TNPS= 0,
            TINTS=          0,                                             &END
20.7869     0.0000          -1.6635
20.7869     0.5044          -1.5844
20.7869     0.9614          -1.3570
20.7869     1.3281          -1.0018
20.7869     1.5697          -0.5522
20.7868     1.6488          -0.0493
&BPNODE     TNODE=          3,      TNPC=  0,       TINTC=          0,     &END
```

```
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=      1.0000,
            ALF=  0.0,    THETA=        0.0,
            INMODE=       4,     TNODS=         0.     TNPS= 0.
            TINTS=        0.                                       &END
21.1576     0.0000  -1.6605
21.1576     0.5085  -1.5802
21.1576     0.9682  -1.3487
21.1576     1.3349  -0.9875
21.1576     1.5733  -0.5313
21.1575     1.6484  -0.0228
&BPNODE     TNODE=        3,     TNPC=  0,      TINTC=        0,   &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=      1.0000,
            ALF=  0.0,    THETA=        0 0,
            INMODE=       4,     TNODS=         0,     TNPS= 0,
            TINTS=        0,                                       &END
21.3867     0.0000        -1.6583
21.3867     0.5110        -1.5771
21.3867     0.9722        -1.3431
21.3867     1.3389        -0.9781
21.3867     1.5751        -0.5179
21.3866     1.6482        -0.0059
&BPNODE     TNODE=        3,     TNPC=  0,      TINTC=        0,   &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=      1.0000,
            ALF=  0.0,    THETA=        0.0,
            INMODE=       4,     TNODS=         3,     TNPS= 0,
            TINTS=        0,                                       &END
21.4642     0.0000        -1.6574
21.4642     0.5124        -1.5759
21.4642     0.9745        -1.3404
21.4642     1.3412        -0.9734
21.4642     1.5762        -0.5110
21.4641     1.6481        0.0000
&BPNODE     TNODE=        3,     TNPC=  0,      TINTC=        0,   &END


&PATCH1     IREV= 0,      IDPAT=        2,      MAKE=         0.
            KCOMP=        0,     KASS= 0,                          &END
       UPPER MID FUSELAGE
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=      1.0000,
            ALF=  0.0,    THETA=        0.0,
            INMODE=       4,     TNODS=         0,     TNPS= 0,
            TINTS=        0,                                       &END
14.2985     1.5751        0.0000
14.2983     1.5048        0.4900
14.2983     1.2800        0.9309
14.2983     0.9299        1.2807
14.2983     0.4888        1.5051
14.2983     0.0000        1.5829
&BPNODE     TNODE=        3,     TNPC=  0,      TINTC=        0,   &END
```

```
&SECT1      STX=  0.0000, STY=  0.0000, STZ=   0.0000, SCALE=      1.0000,
            ALF=  0.0,    THETA=      0.0,
            INMODE=       4,     TNODS=       0,     TNPS= 0,
            TINTS=        0,                                        &END
14.3560     1.5698        0.0434
14.3558     1.4962        0.5234
14.3558     1.2664        0.9527
14.3558     0.9169        1.2919
14.3558     0.4811        1.5091
14.3558     0.0000        1.5846
&BPNODE     TNODE=        3,     TNPC=  0,    TINTC=         0,     &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=   0.0000, SCALE=      1.0000,
            ALF=  0.0,    THETA=      0.0,
            INMODE=       4,     TNODS=       0,     TNPS= 0,
            TINTS=        0,                                        &END
14.5818     1.5692        0.0855
14.5815     1.4913        0.5575
14.5815     1.2565        0.9767
14.5815     0.9070        1.3067
14.5815     0.4751        1.5176
14.5815     0.0000        1.5912
&BPNODE     TNODE=        3,     TNPC=  0,    TINTC=         0,     &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=   0.0000, SCALE=      1.0000,
            ALF=  0.0,    THETA=      0.0,
            INMODE=       4,     TNODS=       0,     TNPS= 0,
            TINTS=        0,                                        &END
14.9568     1.5731        0.1252
14.9565     1.4896        0.5908
14.9565     1.2500        1.0016
14.9565     0.8999        1.3239
14.9565     0.4707        1.5295
14.9565     0.0000        1.6014
&BPNODE     TNODE=        3,     TNPC=  0,    TINTC=         0,     &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=   0.0000, SCALE=      1.0000.
            ALF=  0.0,    THETA=      0.0,
            INMODE=       4,     TNODS=       0,     TNPS= 0,
            TINTS=        0,                                        &END
15.4628     1.5811        0.1616
15.4626     1.4904        0.6219
15.4626     1.2463        1.0260
15.4626     0.8952        1.3420
15.4626     0.4676        1.5432
15.4626     0.0000        1.6139
&BPNODE     TNODE=        3,     TNPC=  0,    TINTC=         0,     &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=   0.0000, SCALE=      1.0000,
            ALF=  0.0,    THETA=      0.0,
            INMODE=       4,     TNODS=       0,     TNPS= 0,
            TINTS=        0,                                        &END
16.0776     1.5933        0.1902
16.0776     1.4940        0.6472
16.0776     1.2461        1.0470
16.0776     0.8935        1.3589
16.0776     0.4664        1.5573
16.0776     0.0000        1.6271
&BPNODE     TNODE=        3,     TNPC=  0,    TINTC=         0,     &END
```

```
&SECT1      STX=  0.0000,  STY=  0.0000,  STZ=   0.0000,  SCALE=        1.0000,
            ALF=  0.0,     THETA=        0.0,
            INMODE=        4,     TNODS=         0,       TNPS= 0,
            TINTS=         0,                                           &END
16.7691     1.6043         0.2083
16.7691     1.5000         0.6645
16.7691     1.2492         1.0627
16.7691     0.8948         1.3730
16.7691     0.4668         1.5701
16.7691     0.0000         1.6396
&BPNODE     TNODE=         3,     TNPC=  0,      TINTC=          0,     &END
&SECT1      STX=  0.0000,  STY=  0.0000,  STZ=   0.0000,  SCALE=        1.0000,
            ALF=  0.0,     THETA=        0.0,
            INMODE=        4,     TNODS=         0,       TNPS= 0,
            TINTS=         0,                                           &END
17.5037     1.6103         0.2125
17.5036     1.5089         0.6716
17.5036     1.2563         1.0715
17.5036     0.8997         1.3830
17.5036     0.4693         1.5808
17.5036     0.0000         1.6507
&BPNODE     TNODE=         3,     TNPC=  0,      TINTC=          0,     &END
&SECT1      STX=  0.0000,  STY=  0.0000,  STZ=   0.0000,  SCALE=        1.0000,
            ALF=  0.0,     THETA=        0.0,
            INMODE=        4,     TNODS=         0,       TNPS= 0,
            TINTS=         0,                                           &END
18.2571     1.6195         0.1979
18.2570     1.5221         0.6635
18.2570     1.2691         1.0699
18.2570     0.9097         1.3869
18.2570     0.4747         1.5884
18.2570     0.0000         1.6594
&BPNODE     TNODE=         3,     TNPC=  0,      TINTC=          0,     &END
&SECT1      STX=  0.0000,  STY=  0.0000,  STZ=   0.0000,  SCALE=        1.0000,
            ALF=  0.0,     THETA=        0.0,
            INMODE=        4,     TNODS=         0,       TNPS= 0,
            TINTS=         0,                                           &END
18.9955     1.6313         0.1662
18.9955     1.5377         0.6413
18.9955     1.2860         1.0582
18.9955     0.9236         1.3844
18.9955     0.4825         1.5920
18.9955     0.0000         1.6650
&BPNODE     TNODE=         3,     TNPC=  0,      TINTC=          0,     &END
&SECT1      STX=  0.0000,  STY=  0.0000,  STZ=   0.0000,  SCALE=        1.0000,
            ALF=  0.0,     THETA=        0.0,
            INMODE=        4,     TNODS=         0,       TNPS= 0,
            TINTS=         0,                                           &END
19.6855     1.6440         0.1260
19.6855     1.5522         0.6110
19.6855     1.3032         1.0401
19.6855     0.9385         1.3769
19.6855     0.4910         1.5918
19.6855     0.0000         1.6670
&BPNODE     TNODE=         3,     TNPC=  0,      TINTC=          0,     &END
```

```
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=     1.0000,
            ALF=  0.0,    THETA=      0.0,
            INMODE=       4,     TNODS=       0,      TNPS= 0,
            TINTS=        0,                                     &END
20.2908     1.6495        0.0847
20.2908     1.5630        0.5794
20.2908     1.3177        1.0199
20.2908     0.9515        1.3669
20.2908     0.4985        1.5887
20.2908     0.0000        1.6660
&BPNODE     TNODE=        3,     TNPC=  0,    TINTC=        0,   &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=     1.0000,
            ALF=  0.0,    THETA=      0.0,
            INMODE=       4,     TNODS=       0,      TNPS= 0,
            TINTS=        0,                                     &END
20.7868     1.6488        0.0493
20.7868     1.5697        0.5522
20.7868     1.3281        1.0018
20.7868     0.9614        1.3570
20.7868     0.5044        1.5844
20.7868     0.0000        1.6635
&BPNODE     TNODE=        3,     TNPC=  0,    TINTC=        0,   &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=     1.0000,
            ALF=  0.0,    THETA=      0.0,
            INMODE=       4,     TNODS=       0,      TNPS= 0,
            TINTS=        0,                                     &END
21.1575     1.6484        0.0228
21.1576     1.5733        0.5313
21.1576     1.3349        0.9875
21.1576     0.9682        1.3487
21.1576     0.5085        1.5802
21.1576     0.0000        1.6605
&BPNODE     TNODE=        3,     TNPC=  0,    TINTC=        0,   &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=     1.0000,
            ALF=  0.0,    THETA=      0.0,
            INMODE=       4,     TNODS=       0,      TNPS= 0,
            TINTS=        0,                                     &END
21.3866     1.6482        0.0059
21.3867     1.5751        0.5179
21.3867     1.3389        0.9781
21.3867     0.9722        1.3431
21.3867     0.5110        1.5771
21.3867     0.0000        1.6583
&BPNODE     TNODE=        3,     TNPC=  0,    TINTC=        0,   &END
&SECT1      STX=  0.0000, STY=  0.0000, STZ=  0.0000, SCALE=     1.0000,
            ALF=  0.0,    THETA=      0.0,
            INMODE=       4,     TNODS=       3,      TNPS= 0,
            TINTS=        0,                                     &END
21.4641     1.6481        0.0000
21.4642     1.5756        0.5132
21.4642     1.3402        0.9748
21.4642     0.9736        1.3411
21.4642     0.5118        1.5760
21.4642     0.0000        1.6574
&BPNODE     TNODE=        3,     TNPC=  0,    TINTC=        0,   &END
```

```
&PATCH1     IREV= 0,     IDPAT=       2,     MAKE=        0,
            KCOMP=       0,      KASS= 0,                          &END
     AFT FUSELAGE
&SECT1      STX=  0.0000, STY=  0.0000, STZ=   0.0000,  SCALE=     1.0000,
            ALF=  0.0,   THETA=       0.0,
            INMODE=      4,      TNODS=       0,       TNPS= 0,
            TINTS=       0,                                        &END
21.4642     0.0000       -1.6574
21.4642     0.5124       -1.5759
21.4642     0.9745       -1.3404
21.4642     1.3412       -0.9734
21.4642     1.5762       -0.5110
21.4641     1.6481       0.0000
21.4642     1.5756       0.5132
21.4642     1.3402       0.9748
21.4642     0.9736       1.3411
21.4642     0.5118       1.5760
21.4642     0.0000       1.6574
&BPNODE     TNODE=       3,      TNPC=  0,    TINTC=         0,    &END
&SECT1      STX=   24.0000,      STY=   0.0000, STZ=   0.0000, SCALE=1.6096,
            ALF=   0.0,  THETA=       0.0,
            INMODE=      1,      TNODS=       0,       TNPS= 0,
            TINTS=       0,                                        &END
0.0000      -1.0000      0.0000
0.1735      -0.9845      0.0000
0.3420      -0.9395      0.0000
0.5000      -0.8660      0.0000
0.6425      -0.7660      0.0000
0.7660      -0.6425      0.0000
0.8660      -0.5000      0.0000
0.9395      -0.3420      0.0000
0.9845      -0.1735      0.0000
1.0000      0.0000       0.0000
0.9845      0.1735       0.0000
0.9395      0.3420       0.0000
0.8660      0.5000       0.0000
0.7660      0.6425       0.0000
0.6425      0.7660       0.0000
0.5000      0.8660       0.0000
0.3420      0.9395       0.0000
0.1735      0.9845       0.0000
0.0000      1.0000       0.0000
&BPNODE     TNODE=       3,      TNPC=  10,   TINTC=         3,    &END
&SECT1      STX=   28.0000,      STY=   0.0000, STZ=   0.0000, SCALE=1.4248,
            ALF=   0.0,  THETA=       0.0,
            INMODE=      0,      TNODS=       0,       TNPS= 0,
            TINTS=       0,                                        &END
&SECT1      STX=   32.0000,      STY=   0.0000, STZ=   0.0000, SCALE=1.0104,
            ALF=   0.0,  THETA=       0.0,
            INMODE=      0,      TNODS=       0,       TNPS= 0,
            TINTS=       0,                                        &END
&SECT1      STX=   33.3330,      STY=   0.0000, STZ=   0.0000, SCALE=0.8332,
            ALF=   0.0,  THETA=       0.0,
            INMODE=      0,      TNODS=       0,       TNPS= 0,
            TINTS=       0,                                        &END
```

```
&SECT1      STX=  36.0000,         STY=  0.0000,  STZ=   0.0000,  SCALE=0.4500,
            ALF=  0.0,     THETA=       0.0
            INMODE=        0,      TNODS=         0,       TNPS= 0,
            TINTS=         0,                                           &END
&SECT1      STX=  38.0000,         STY=  0.0000,  STZ=   0.0000,  SCALE=0.1756,
            ALF=  0.0,     THETA=       0.0,
            INMODE=        0,      TNODS=         0,       TNPS= 0,
            TINTS=         0,                                           &END
&SECT1      STX=  40.0000,         STY=  0.0000,  STZ=   0.0000,  SCALE=0.0000,
            ALF=  0.0,     THETA=       0.0,
            INMODE=        0,      TNODS=         5,       TNPS= 10,
            TINTS=         0,                                           &END


&WAKE1      IDWAK=1,       IFLXW=0,                                     &END
      WING/BODY   WAKE
&WAKE2      KWPACH=6,      KWSIDE=4,     KWLINE=6,     KWPAN1=0,
            KWPAN2=0,      NODEW=0,      INITIAL=1,                     &END
&WAKE2      KWPACH=1,      KWSIDE=2,     KWLINE=0,     KWPAN1=0,
            KWPAN2=0,      NODEW=5,      INITIAL=1,                     &END
&SECT1      STX=  150.0000,        STY=  0.0000,  STZ=   0.0000,  SCALE=1.0000,
            ALF=  0.0,     THETA=       0.0,
            INMODE=-1,     TNODS=       3,       TNPS= 20,
            TINTS=         1,                                           &END


&VS1  NVOLR=       0,      NVOLC=       0,                              &END
&VS2  X0=     3.0000,  Y0=     2.0000,  Z0=     -2.0000,                &END
&VS3  X1=     18.0000,Y1=      2.0000,  Z1=     -2.0000,     NPT1= 16,  &END
&VS4  X2=     3.0000,  Y2=     16.0000,Z2=      -2.0000,     NPT2= 3,   &END
&VS5  X3=     3.0000,  Y3=     2.0000,  Z3=     2.0000,      NPT3= 18,  &END

&VS6  XR0=  0.0000,  YR0=  0.0000,  ZR0=   0.0000,                      &END
&VS7  XR1=  0.0000,  YR1=  10.0000,ZR1=    0.0000,
      XR2=  0.0000,  YR2=  0.0000,  ZR2=   1.0000,                      &END
&VS8  R1=     0.5000.  R2=     5.0000,  PHI1=  0.0,    PHI2=330.0,      &END
&VS9  NRAD=10,       NPHI=  12,       NLEN= 5,                          &END


&SLIN1        NSTLIN=0,                                                 &END
&SLIN2        SX0=  2.0000,  SY0=   5.0000,  SZ0=    -0.5000,
              SU=   0.0000,  SD=    15.0000,DS=      0.1000,            &END
&SLIN2        SX0=  2.0000,  SY0=   5.0000,  SZ0=    -0.4000,
              SU=   0.0000,  SD=    15.0000,DS=      0.1000,            &END
&SLIN2        SX0=  2.0000,  SY0=   5.0000,  SZ0=    -0.3000,
              SU=   0.0000,  SD=    15.0000,DS=      0.1000,            &END
&SLIN2        SX0=  2.0000,  SY0=   5.0000,  SZ0=    -0.2000,
              SU=   0.0000,  SD=    15.0000.DS=      0.1000,            &END
&SLIN2        SX0=  2.0000,  SY0=   5.0000,  SZ0=    -0.1000,
              SU=   0.0000,  SD=    15.0000,DS=      0.1000,            &END
&SLIN2        SX0=  2.0000,  SY0=   5.0000,  SZ0=    0.0000,
              SU=   0.0000,  SD=    15.0000,DS=      0.1000,            &END
```

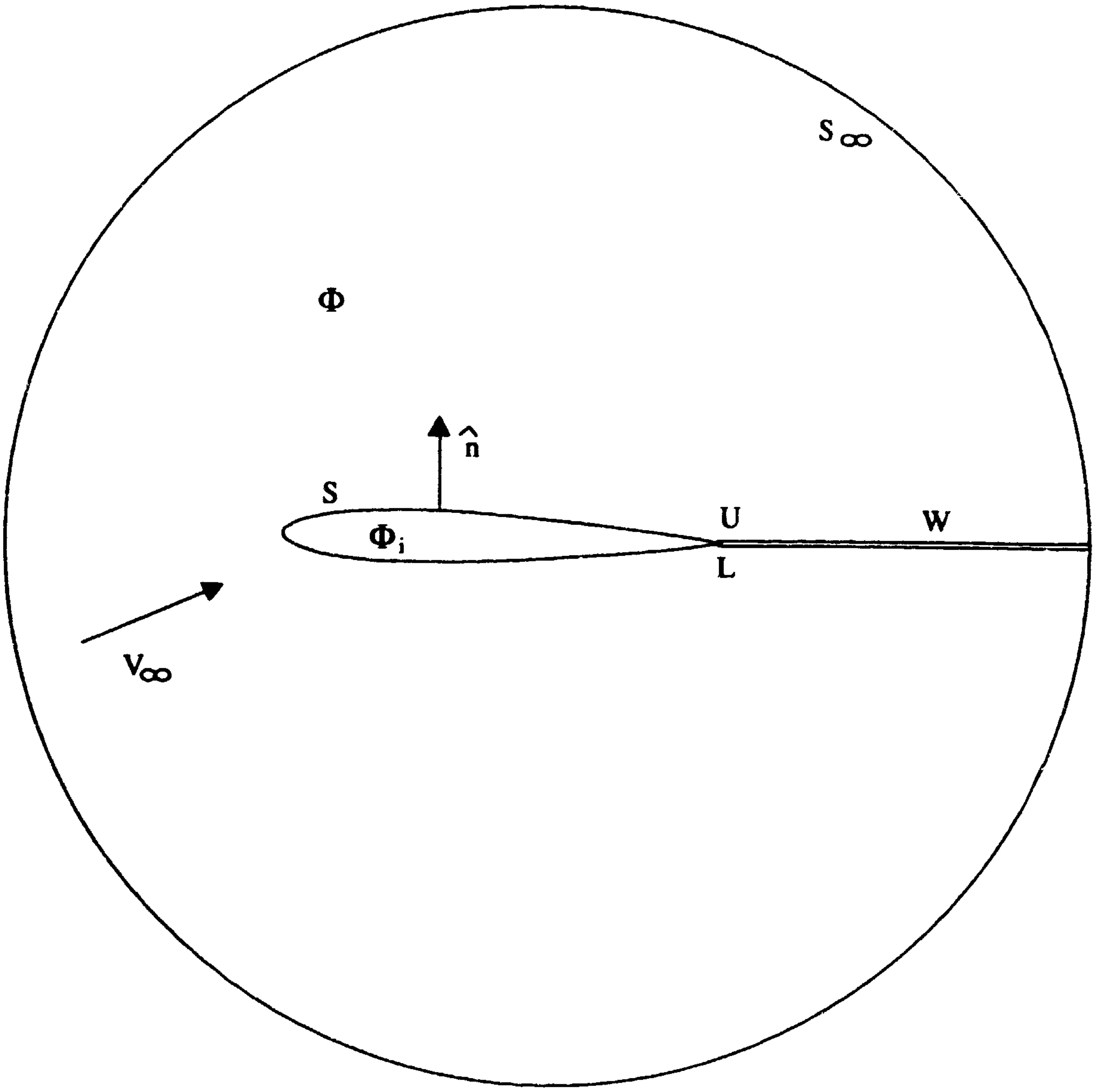
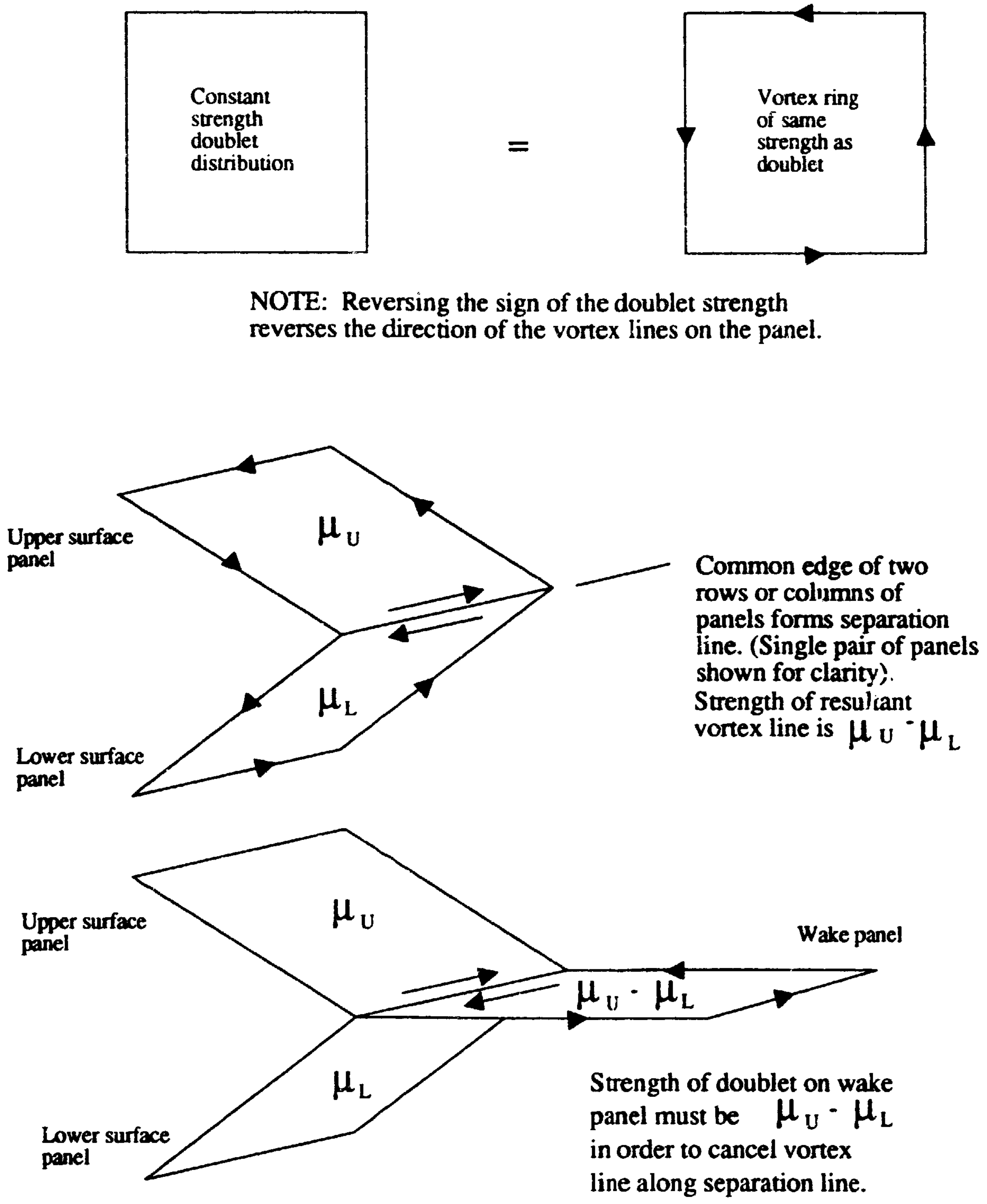


Figure 1. Potential flow model for PMARC.

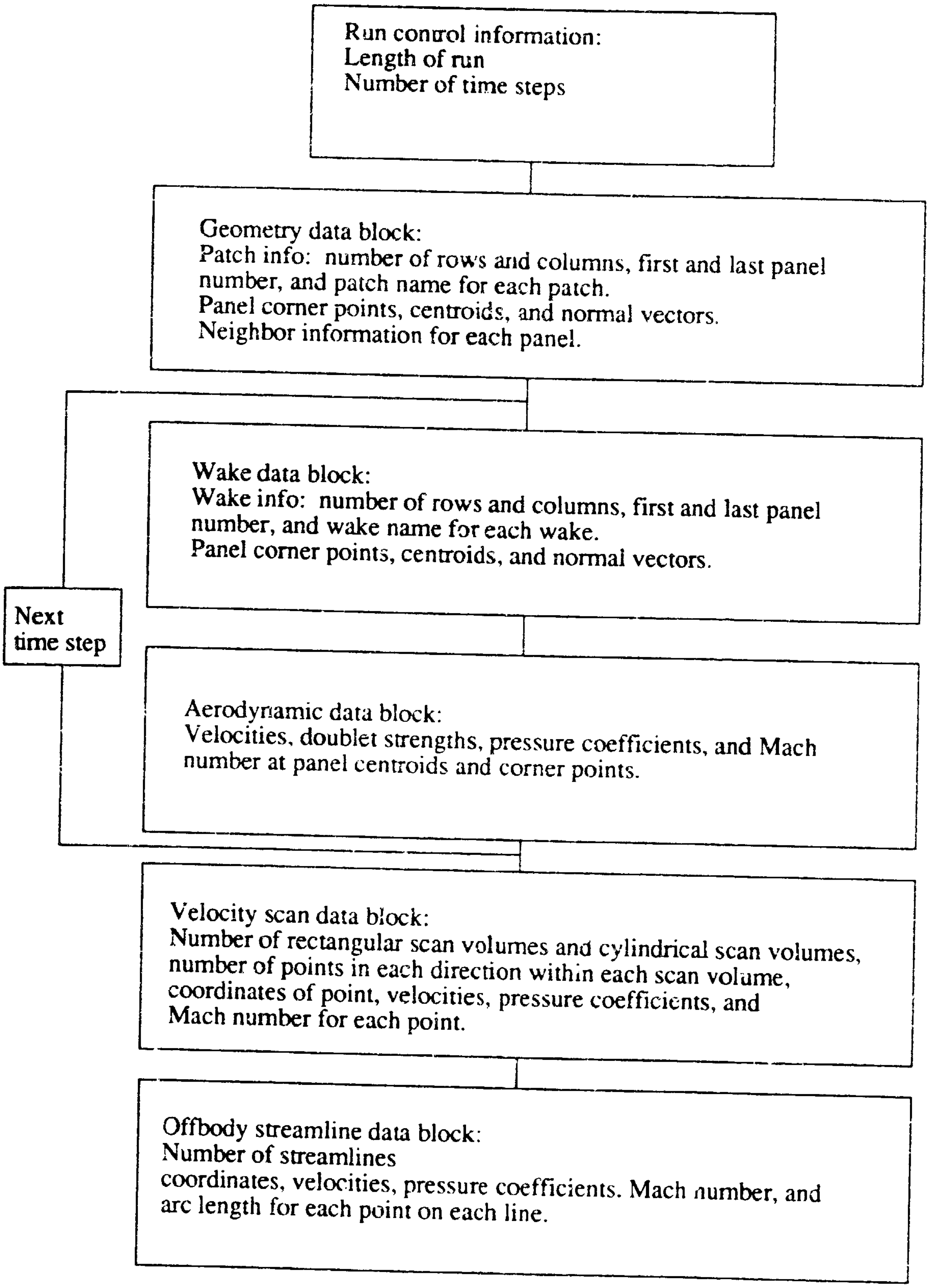


NOTE: Reversing the sign of the doublet strength reverses the direction of the vortex lines on the panel.

Common edge of two rows or columns of panels forms separation line. (Single pair of panels shown for clarity). Strength of resultant vortex line is $\mu_U - \mu_L$

Strength of doublet on wake panel must be $\mu_U - \mu_L$ in order to cancel vortex line along separation line.

Figure 2. Determination of doublet strength shed into wake.



Run control information:
Length of run
Number of time steps

Geometry data block:
Patch info: number of rows and columns, first and last panel number, and patch name for each patch.
Panel corner points, centroids, and normal vectors.
Neighbor information for each panel.

Wake data block:
Wake info: number of rows and columns, first and last panel number, and wake name for each wake.
Panel corner points, centroids, and normal vectors.

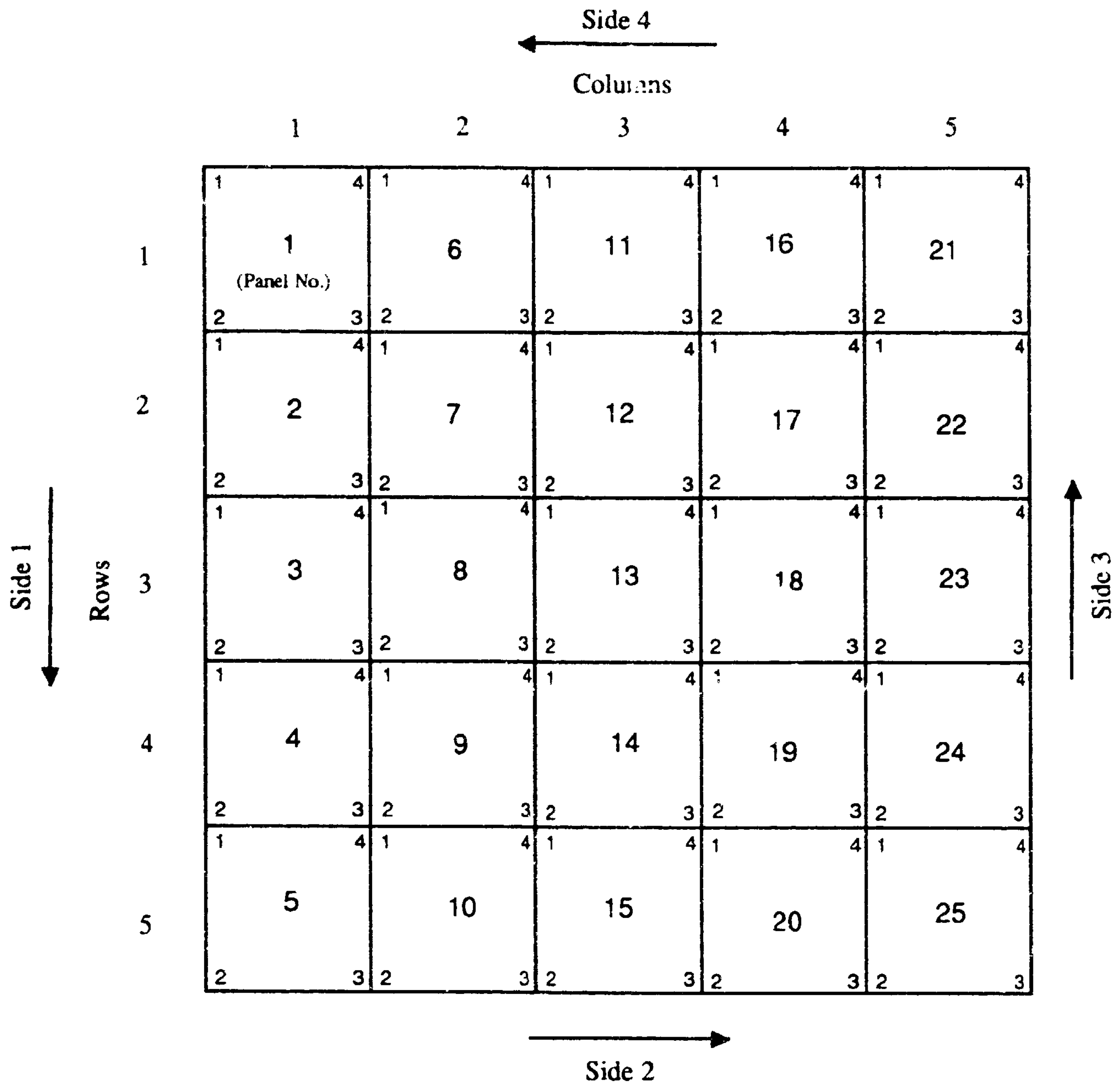
Next time step

Aerodynamic data block:
Velocities, doublet strengths, pressure coefficients, and Mach number at panel centroids and corner points.

Velocity scan data block:
Number of rectangular scan volumes and cylindrical scan volumes, number of points in each direction within each scan volume, coordinates of point, velocities, pressure coefficients, and Mach number for each point.

Offbody streamline data block:
Number of streamlines
coordinates, velocities, pressure coefficients. Mach number, and arc length for each point on each line.

Figure 3. Data arrangement within PMARC plot file.



NOTE: Sides of individual panels follow the same order and direction as the sides of the whole patch.

Figure 4. PMARC surface patch nomenclature.

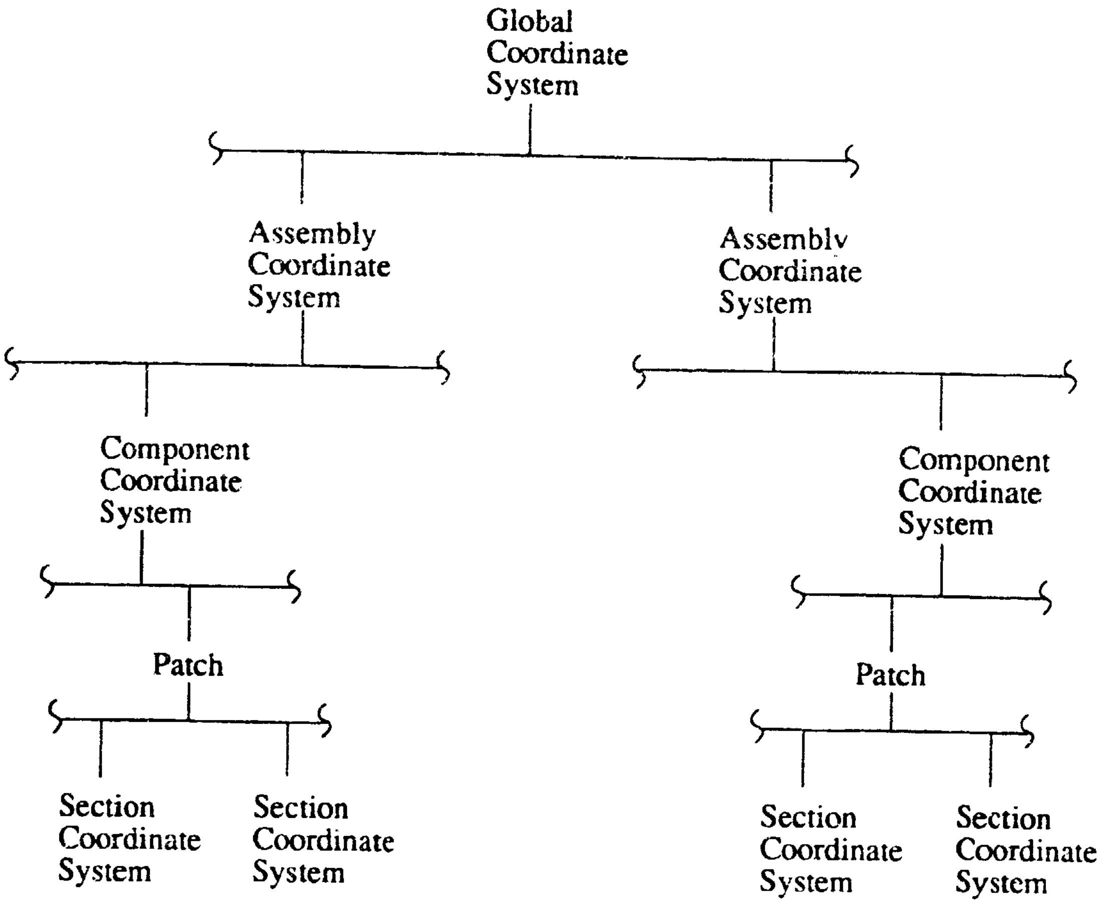
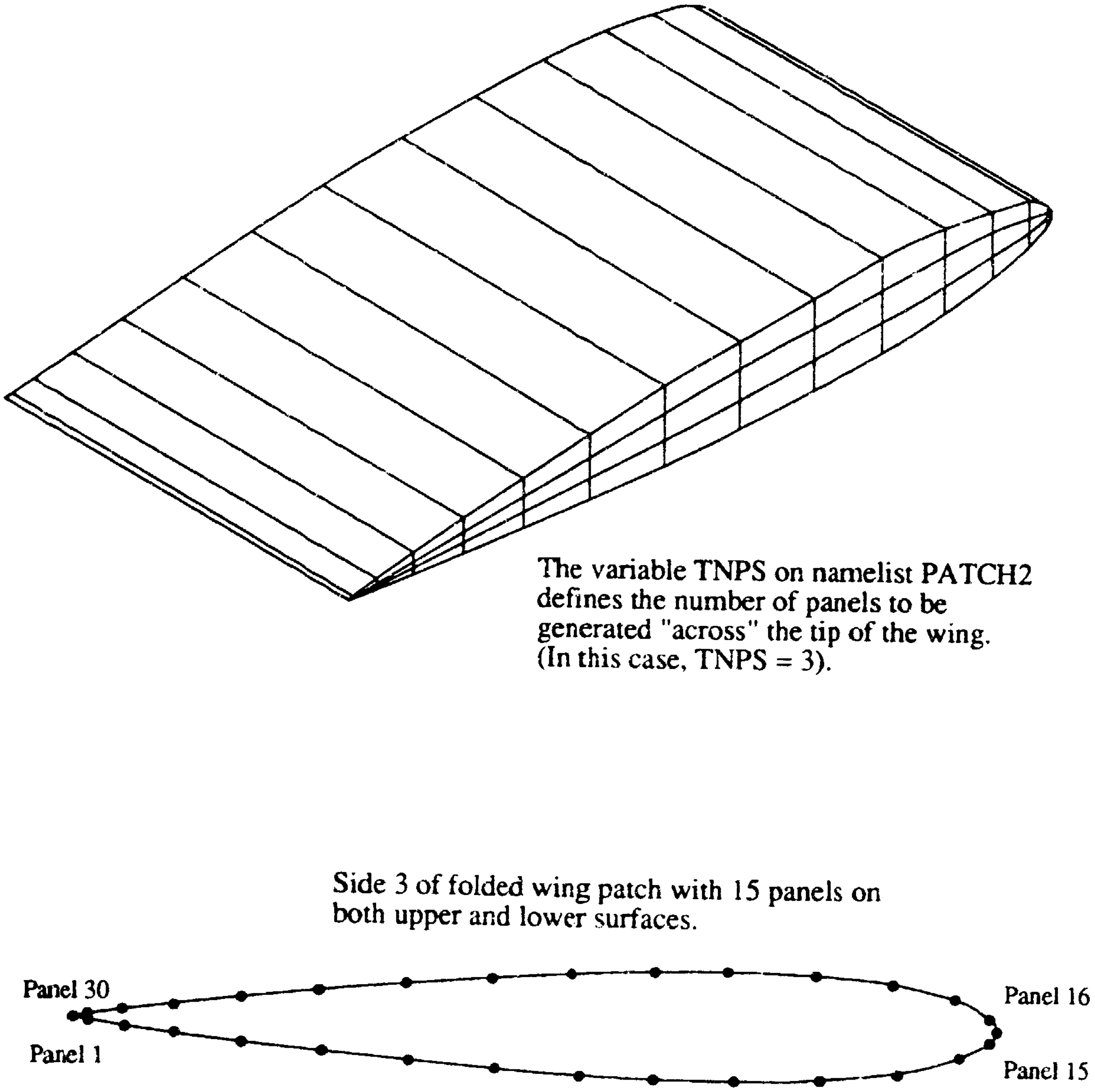

Global Coordinate System
Assembly Coordinate System
Assemblv Coordinate System
Component Coordinate System
Component Coordinate System
Patch
Patch
Section Coordinate System
Section Coordinate System
Section Coordinate System
Section Coordinate System

Figure 5. PMARC surface geometry modeling hierarchy.



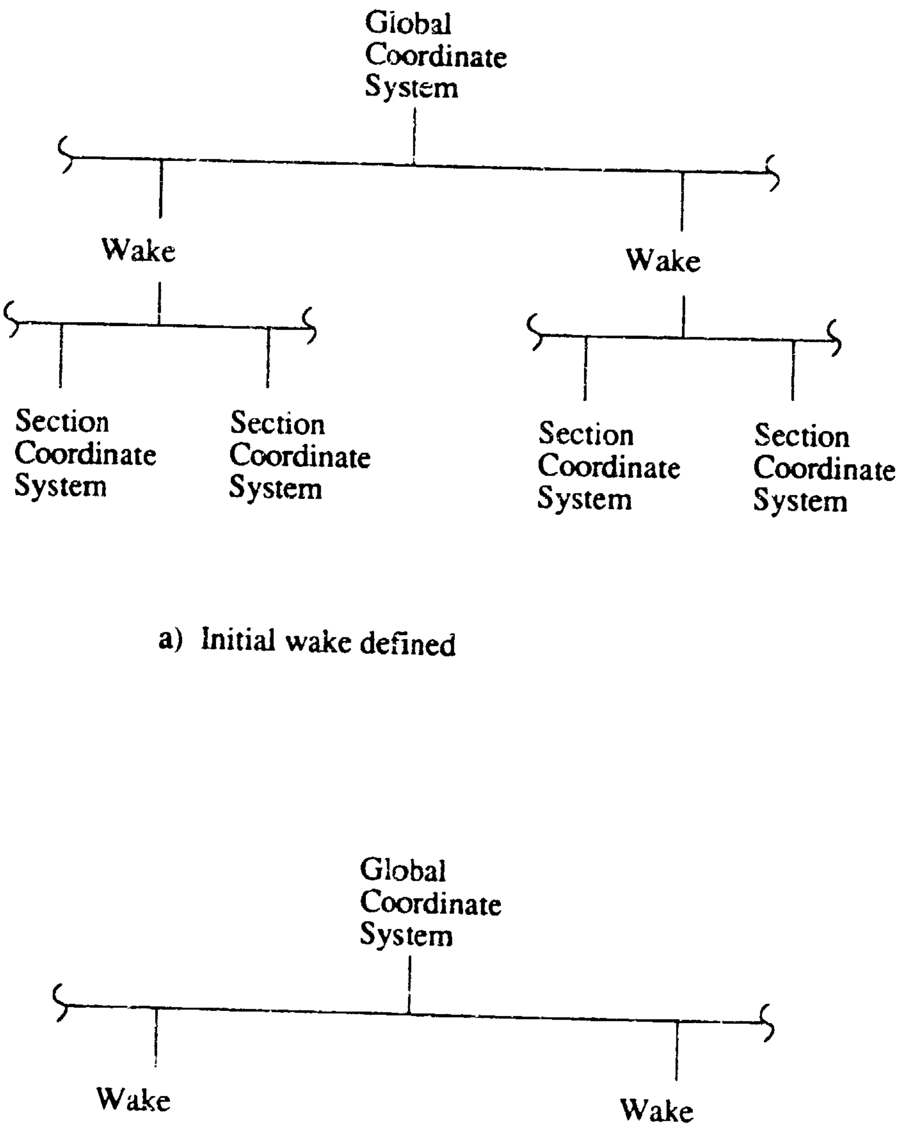
The variable TNPS on namelist PATCH2 defines the number of panels to be generated "across" the tip of the wing. (In this case, TNPS = 3).

Side 3 of folded wing patch with 15 panels on both upper and lower surfaces.

Corner points of panels 1 - 15 define first section of tip patch.

Corner points of panels 30 - 16 (note reverse order) define last section of tip patch.

Figure 6. PMARC automatic tip patch option.


Global Coordinate System
Wake
Wake
Section Coordinate System
Section Coordinate System
Section Coordinate System
Section Coordinate System
Global Coordinate System
Wake
Wake

a) Initial wake defined

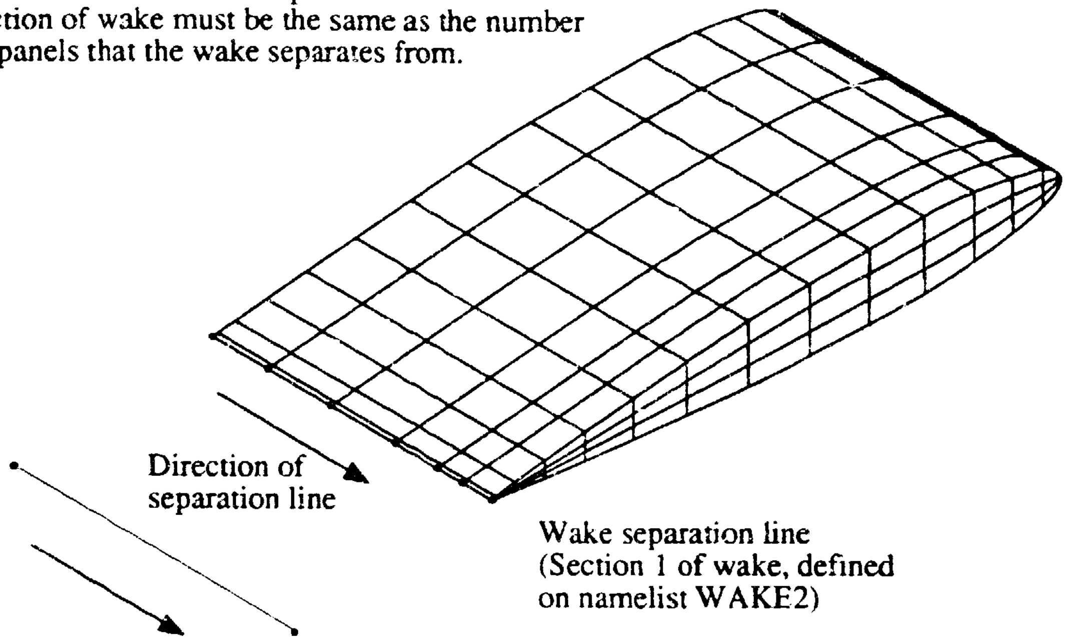
b) No initial wake defined (completely time-stepped wake)

Figure 7. PMARC wake modeling hierarchy.

NOTE: User can define sections of arbitrary shape to define complex initial wake shapes. A single straight line section is shown here for simplicity.

NOTE: Direction of separation line is for a wake separating from side 2 of the folded wing patch. Direction of separation line would be reversed if separation were from side 4.

NOTE: Total number of panels defined on each section of wake must be the same as the number of panels that the wake separates from.



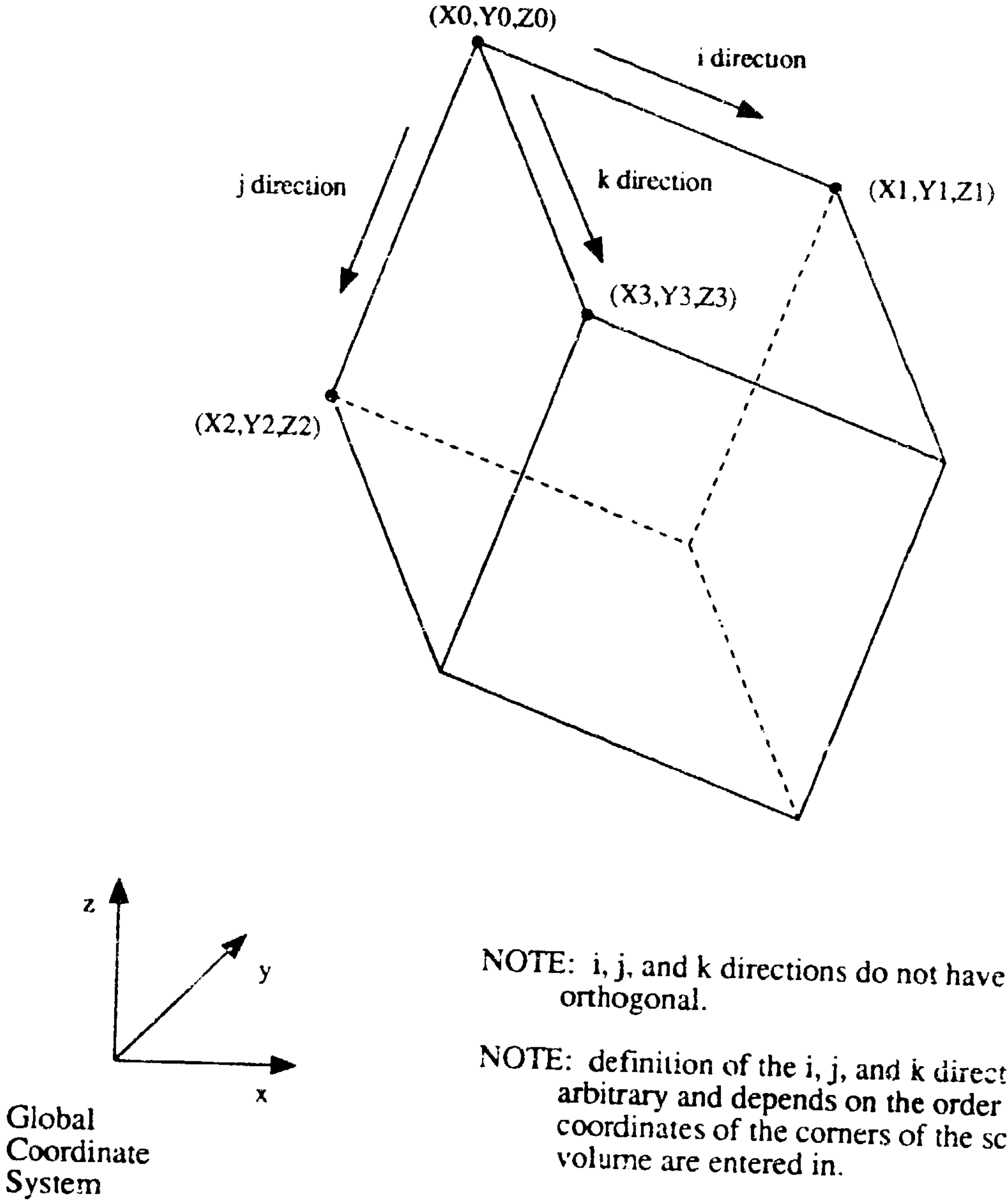
Direction of separation line

Wake separation line (Section 1 of wake, defined on namelist WAKE2)

NOTE: Section definition must be in same direction as separation line

Section 2 of wake, defined with SECT1 namelist, basic point coordinates, and BPNODE namelist

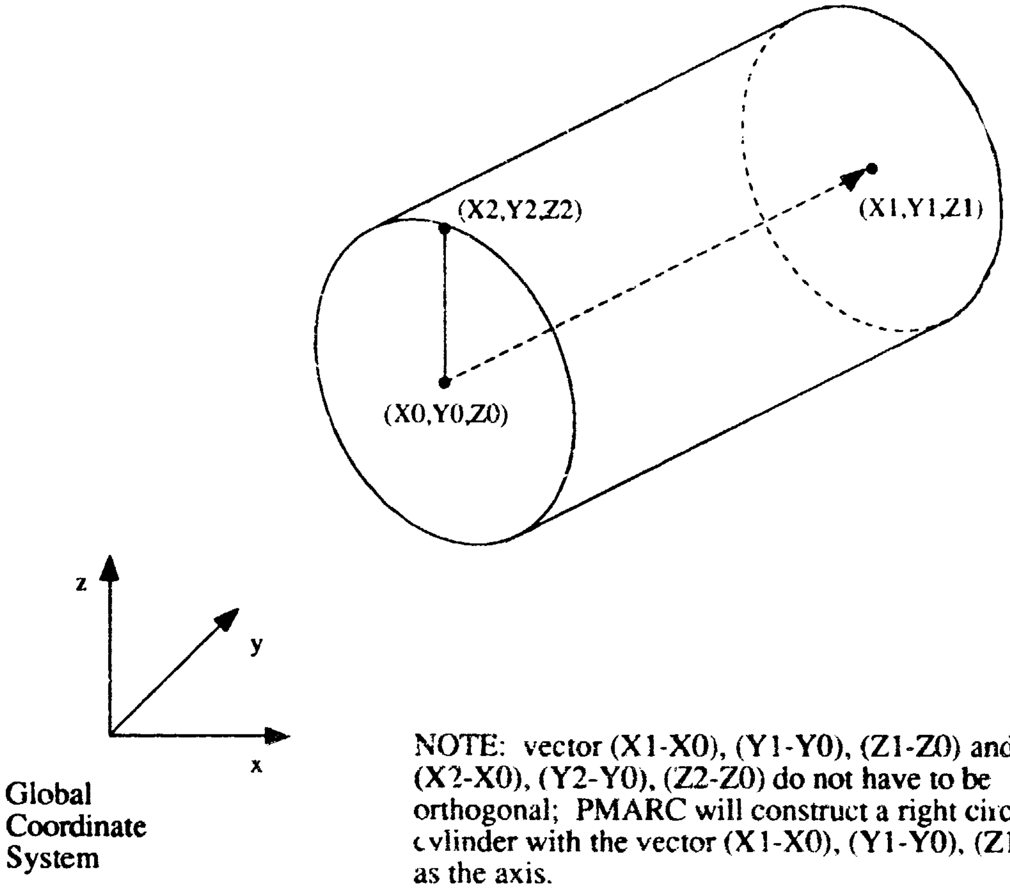
Figure 8. PMARC wake modeling nomenclature.



NOTE: i, j, and k directions do not have to be orthogonal.

NOTE: definition of the i, j, and k directions is arbitrary and depends on the order the coordinates of the corners of the scan volume are entered in.

Figure 9. Rectangular velocity scan volume nomenclature.



NOTE: vector (X1-X0), (Y1-Y0), (Z1-Z0) and vector (X2-X0), (Y2-Y0), (Z2-Z0) do not have to be orthogonal; PMARC will construct a right circular cylinder with the vector (X1-X0), (Y1-Y0), (Z1-Z0) as the axis.

NOTE: The angles PHI1 and PHI2 (from namelist VS8) are measured from the plane containing points 0, 1, and 2, with positive angles being defined by the Right Hand Rule about the cylinder axis.

Figure 10. Cylindrical velocity scan volume nomenclature.

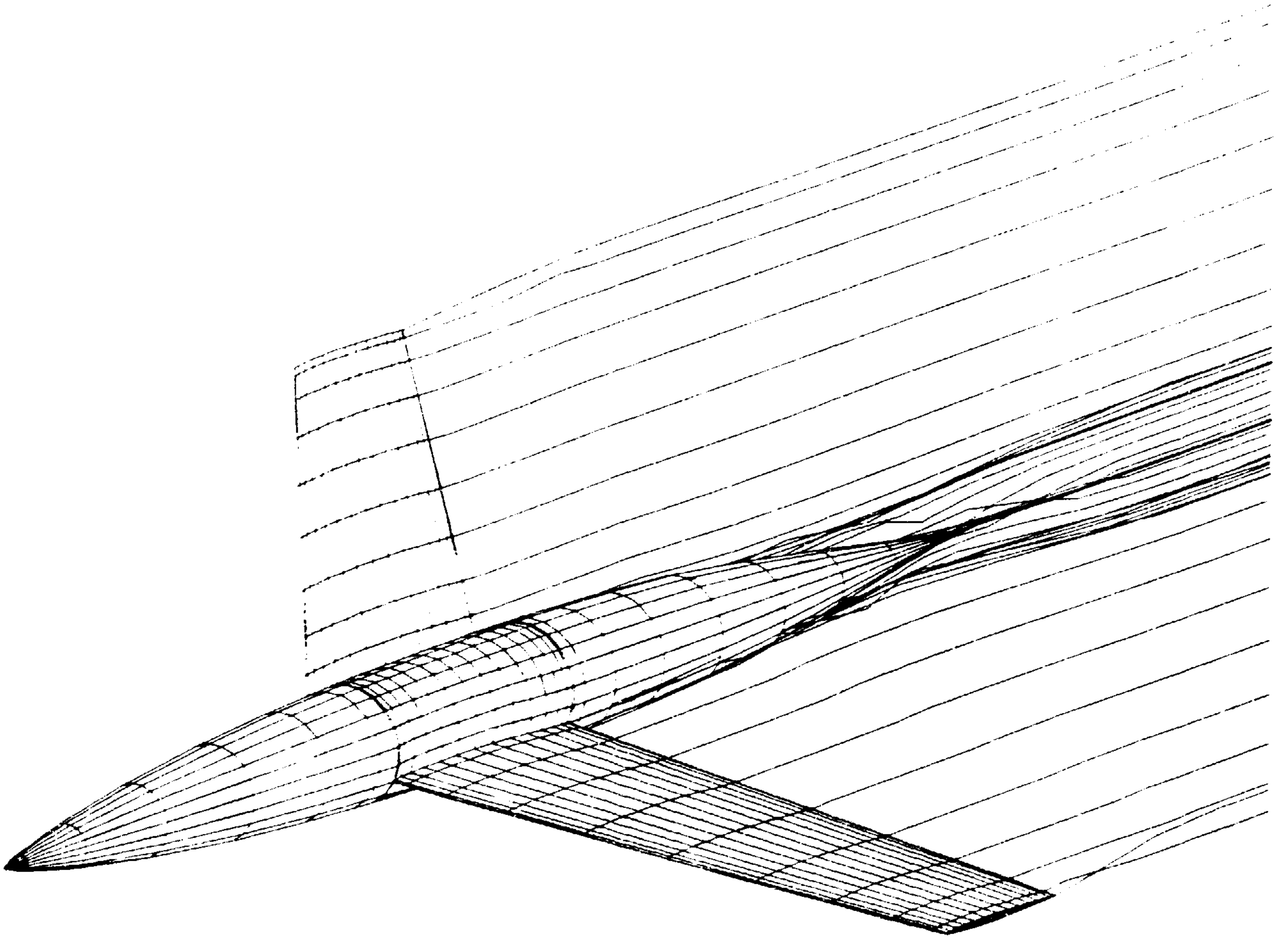


Figure 11. PMARC representation of a symmetric wing/body.

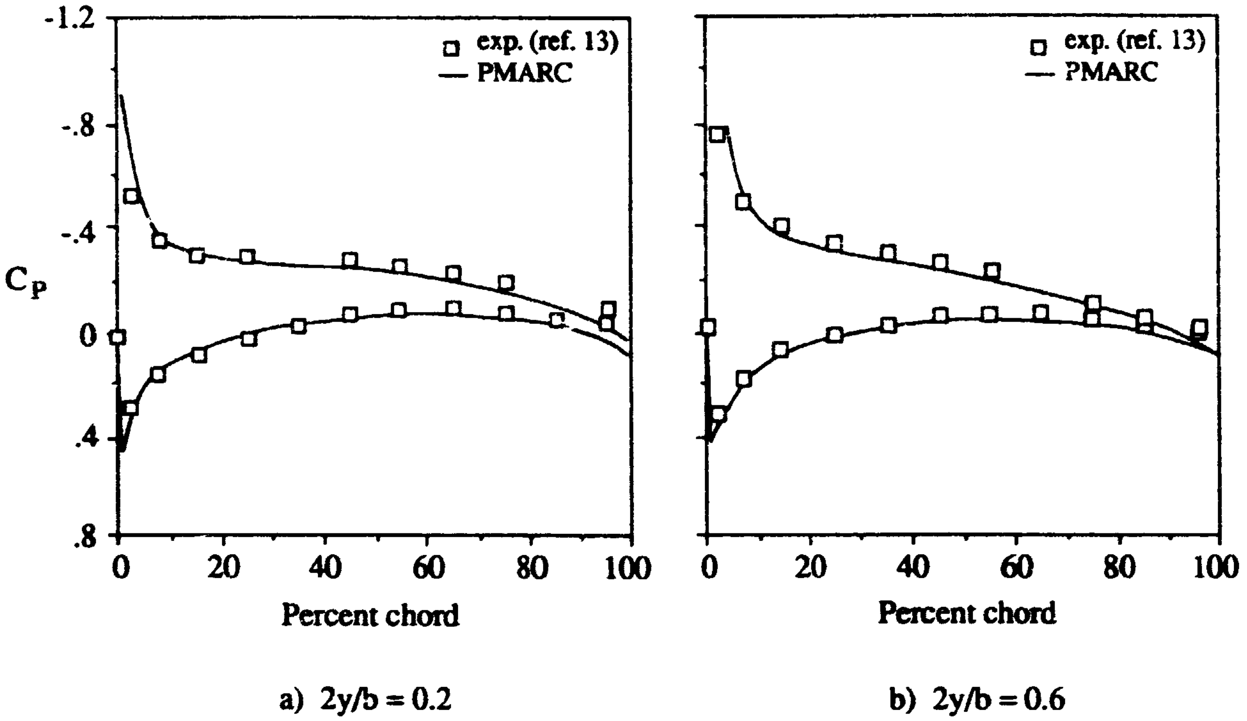


Figure 12. Comparison of experimental data and PMARC results for two spanwise stations on the wing of the wing/body configuration. Angle of attack is 4°.

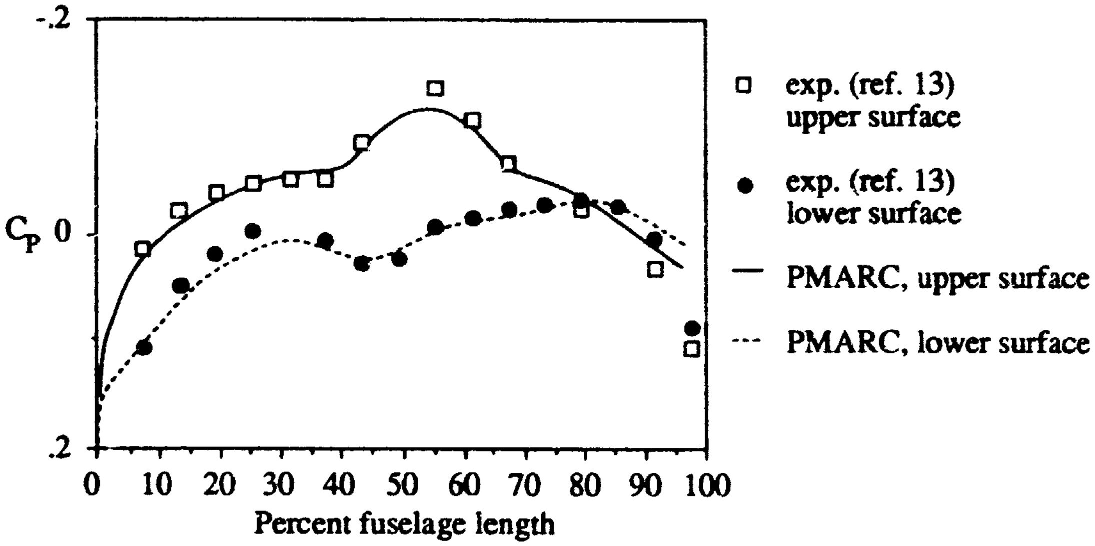


Figure 13. Comparison of experimental data and PMARC results along the fuselage centerline of the wing/body configuration. Angle of attack is 4°.

NASA
National Aeronautics and Space Administration

# Report Documentation Page

1. Report No.
NASA TM-102851

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
Potential Flow Theory and Operation Guide for the Panel Code PMARC

5. Report Date
January 1991

6. Performing Organization Code

7. Author(s)
Dale L. Ashby, Michael R. Dudley, and Steve K. Iguchi, Lindsey Browne, and Joseph Katz (San Diego State University, San Diego, California)

8. Performing Organization Report No
A-90244

10. Work Unit No.
505-61-71

9. Performing Organization Name and Address
Ames Research Center
Moffett Field, CA 94035-1000

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, DC 20546-0001

13. Type of Report and Period Covered
Technical Memorandum

14. Sponsoring Agency Code

15. Supplementary Notes
Point of Contact: Dale L. Ashby, Ames Research Center, MS 247-2, Moffett Field, CA 94035-1000
(415) 604-5047 or FTS 464-5047

16. Abstract

The theoretical basis for PMARC, a low-order potential-flow panel code for modeling complex three-dimensional geometries, is outlined. Several of the advanced features currently included in the code, such as internal flow modeling, a simple jet model, and a time-stepping wake model, are discussed in some detail. The code is written using adjustable size arrays so that it can be easily redimensioned for the size problem being solved and the computer hardware being used. An overview of the program input is presented, with a detailed description of the input available in the appendices. Finally, PMARC results for a generic wing/body configuration are compared with experimental data to demonstrate the accuracy of the code. The input file for this test case is given in the appendices.

17. Key Words (Suggested by Author(s))
Panel method
Potential flow
Unsteady aerodynamics

18. Distribution Statement
Unclassified-Unlimited

Subject Category - 02

| 19. Security Classif. (of this report) | 20. Security Classif. (of this page) | 21. No. of Pages | 22. Price |
|---|---|---|---|
| Unclassified | Unclassified | 90 | A05 |

For sale by the National Technical Information Service, Springfield, Virginia 22161