NASA CR-166,529

A Reproduced Copy

OF

NASA CR-166,529

NASA Contractor Report 166529

A Grid-Embedding Transonic Flow Analysis Computer Program for Wing/Nacelle Configurations

Essam H. Atta
Joseph Vadyak

Contract NAS2-11285
November 1983

NASA

NASA Contractor Report 166529

A Grid-Embedding Transonic Flow Analysis Computer Program for Wing/Nacelle Configurations

Essam H. Atta
Joseph Vadyak
Lockheed-Georgia Company
Marietta, Georgia

Prepared for Ames Research Center
under Contract NAS2-11285

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

TABLE OF CONTENTS

A GRID EMBEDDING TRANSONIC FLOW ANALYSIS COMPUTER PROGRAM FOR WING/NACELLE CONFIGURATIONS

Essam H. Atta and Joseph Vadyak
Advanced Flight Sciences Department
Lockheed-Georgia Company
Marietta, Georgia 30063

SUMMARY

An efficient grid-interfacing zonal algorithm has been developed for computing the three-dimensional transonic flow field about wing/nacelle multicomponent configurations. The algorithm uses the full-potential formulation and the AF2 fully-implicit approximate factorization scheme. The flow field solution is computed using a component-adaptive grid approach in which separate grids are employed for the individual components in the multicomponent configuration, where each component grid is optimized for a particular geometry such as the wing or nacelle. The wing and nacelle component grids are allowed to overlap, and flow field information is transmitted from one grid to another through the overlap region using trivariate interpolation.

This report presents a discussion of the computational methods used to generate both the wing and nacelle component grids, the technique used to interface the component grids, and the method used to obtain the inviscid multicomponent flow field solution. Computed results and correlations with experiment are presented. Also presented are discussions on the organization of the wing grid generation (GRGEN3) and nacelle grid generation (NGRIDA) computer programs, the grid interface (LK) computer program, and the wing/nacelle flow solution (TWN) computer program. Descriptions of the respective subroutines, definitions of the required input parameters, a brief discussion on interpretation of the output, and sample cases illustrating application of the analysis are provided for each of the four computer programs.

An isolated nacelle transonic flow analysis computer program has been developed as a part of this investigation. The theory and usage of the isolated nacelle program is discussed in a separate report.

SECTION I

INTRODUCTION

Reliable and efficient three-dimensional transonic analysis methods are required to make realistic and cost-effective predictions of aircraft aerodynamics. Early efforts to predict the transonic flow field about aircraft multicomponent configurations are based on the transonic small disturbance formulation. This allows the geometry of the configuration to be greatly simplified and the surface boundary condition to be applied on a mean approximate surface. Accurate prediction of such flow fields, however, requires the use of the full-potential formulation and the generation of a suitable surface-fitted grid. Because each aircraft component (wing, nacelle, fuselage) requires, in general, a grid system that is usually incompatible with the grid systems of the other components, the generation of a single surface-fitted grid for the entire configuration is a difficult task. In such a global grid, control of grid point distribution, skewness, and clustering would be difficult to achieve.

In the present investigation, an alternate approach is employed in computing the transonic flow field about wing/nacelle configurations. The approach taken in developing this multicomponent algorithm uses a component-adaptive grid embedding scheme in which the global computational grid is composed of a series of overlapped component grids, where each component grid is optimized for a particular geometry such as the wing or nacelle. The AF2 approximate factorization algorithm[1] is used to determine the conservative full-potential equation solution on each component grid with trivariate interpolation being used to transfer property information between the component grids. The AF2 algorithm has been applied to the computation of two-dimensional transonic airfoil flows by Holst[2] and to the computation of three-dimensional transonic wing flows by Holst and Thomas[3]. Increases in convergence speed by factors of 4 to 7 have been realized using the AF2 scheme instead of using the standard transonic relaxation scheme, successive-line-over-relaxation.

Using the component-adaptive grid approach requires that the grid generation and flow solution algorithms be available for each component in the configuration. For determining the solution for wing/nacelle configurations, this then requires that an isolated wing algorithm and an isolated nacelle algorithm be available to be linked to form the combined multicomponent algorithm. In the present investigation, the NASA-Ames GRGEN3 wing grid generation Fortran program and the TWING transonic wing flow analysis Fortran program[3] are employed in computing the wing component flow field. To determine the nacelle component flow field, the NGRIDA nacelle grid generation program and the NACELLE nacelle flow analysis program are employed. The NGRIDA and NACELLE Fortran programs were developed as part of the present study and their use for the computation of three-dimensional transonic flows about isolated nacelle/inlet configurations is documented in Reference 4. The TWING and NACELLE programs have been combined to produce a wing/nacelle multicomponent flow analysis algorithm called TWN. An additional Fortran program, called LK, has been developed which interfaces the component grids. Preliminary results employing this analysis method are reported in Reference 5.

SECTION II

COMPUTATIONAL METHOD

1. INTRODUCTION

Both the wing and nacelle body-fitted computational grids are determined using numerical mesh generation techniques. The GRGEN3 and NGRIDA Fortran computer programs determine the wing and nacelle component grids, respectively. After the component grids are established, grid interfacing is performed using the LK Fortran program. The global computational grid is then used by the TWN Fortran flow analysis program in determining the multi-component flow field. The efficient AF2 approximate factorization scheme is used by the TWN algorithm in solving the system of finite-difference equations.

In this section, the computational methods used to generate the curvilinear component meshes and to obtain the inviscid flow solution are presented. Computed results and comparison with experimental data are presented to illustrate application of the analysis.

2. COMPONENT GRID TOPOLOGY AND GENERATION

The global computational grid is composed of the wing and nacelle component grids, each of which is generated separately. Each component grid is body-fitted and is determined using a numerical grid-generation algorithm.

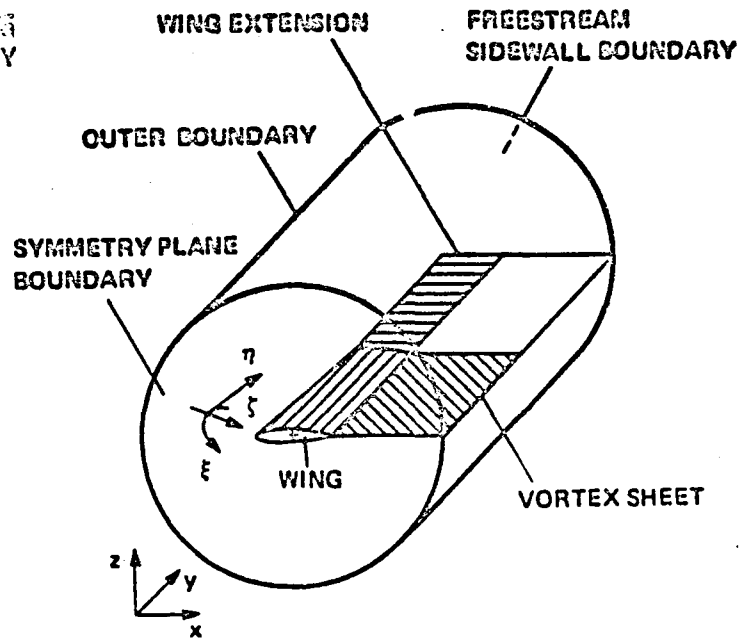
2.1 WING COMPONENT GRID GENERATION

The wing flow field solution is determined on a three-dimensional body-fitted curvilinear mesh. The wing computational mesh is obtained using two-dimensional numerical grid generation techniques for a series of spanwise stations which are distributed along the wing span starting at the root symmetry plane and progressing past the wing tip into the far field.

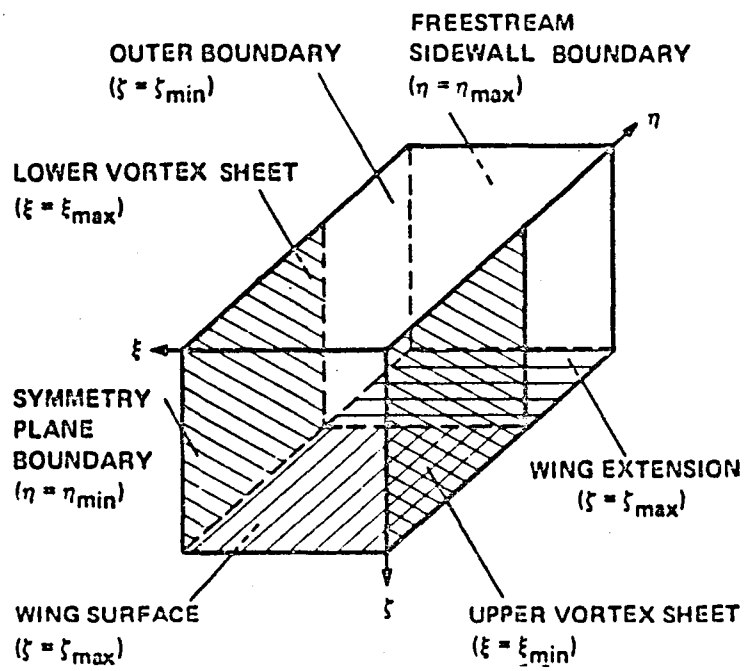
Figure 1a illustrates the O-type body-fitted grid for an isolated wing. The base Cartesian coordinates are denoted by x, y, and z. The computational curvilinear coordinates are denoted by ξ, η, and ζ. The ξ coordinate is in the wraparound direction, initiates at the wing upper surface trailing edge, and terminates at the wing lower surface trailing edge. The η coordinate is in the spanwise direction, initiates at the root symmetry plane, and terminates at the free-stream sidewall boundary. The ζ coordinate is in the radial or normal direction, initiates at the outer computational boundary, and terminates at the wing surface.

The wing surface grid points are clustered near the wing leading and trailing edges using either a conformal mapping distribution or a geometric stretching function expressed in terms of the arc length measured from the wing edge points. The outer boundary points are distributed at equal angular increments around the body.

Once the wing surface and outer boundary mesh point distributions have been determined for a given spanwise station, the interior mesh point



(a) O-TYPE WING GRID



(b) WING COMPUTATIONAL DOMAIN

Figure 1. Isolated wing grid topology.

locations are determined by iteratively solving the two coupled Laplace equations

$$\xi_{xx} + \xi_{zz} = 0 \tag{1}$$

$$\zeta_{xx} + \zeta_{zz} = 0 \tag{2}$$

These equations are transformed to and solved in the computational domain where ξ and ζ are the independent variables and x and z are the dependent variables. An alternating-direction-implicit (ADI) relaxation algorithm is used to solve the governing finite-difference equations[2]. This then establishes the grid point coordinates at each spanwise station used in defining the wing geometry. Clustering of the spanwise stations is performed in the vicinity of the wing root, tip, and span location of the nacelle.

The GRGEN3 Fortran computer program[3] is used to generate the wing component grid. For the case of a wing with no section variation, only the section geometry for the root defining station needs to be entered. In this case, the grid coordinates are found using the numerical relaxation technique for the root and tip stations. The grid coordinates for the remaining wing spanwise stations are found by linearly interpolating between the root and tip stations. Off the wing tip, the grid is generated for wing sections of zero thickness. For arbitrary wing configurations, taper, twist, thickness, and sweep variations can be accounted for by specifying the wing section geometry at a number of spanwise stations.

Figure 1b illustrates the rectangular computational domain which corresponds to the physical domain of Figure 1a. The physical space boundaries transform into boundaries in computational space. This allows for accurate and straightforward boundary condition implementation. The mesh points are equally spaced in computational space thereby permitting the use of standard finite-difference formulae in the flow solution analysis.

To use the isolated wing grid in the present multicomponent flow field analysis, selected interior mesh points are deleted from the grid in order to accommodate the nacelle geometry. Figure 2 illustrates a view of the wing grid at a span station which contains the nacelle. The aperture in the interior of the grid in physical space creates an irregularly shaped void in the wing computational space domain.

2.2 NACELLE COMPONENT GRID GENERATION

The nacelle flow field solution is determined on a three-dimensional body-fitted curvilinear computational mesh. The nacelle computational mesh is obtained by using two-dimensional numerical grid generation techniques for a series of meridional planes which are splayed circumferentially around the body (a meridional plane is a plane containing the longitudinal axis of the inlet).

Figure 3 illustrates the C-type nacelle/inlet body-fitted grid for a nacelle extended in the downstream direction. The base Cartesian coordinates are denoted by x, y, and z. Figure 3a shows the meridional plane grid

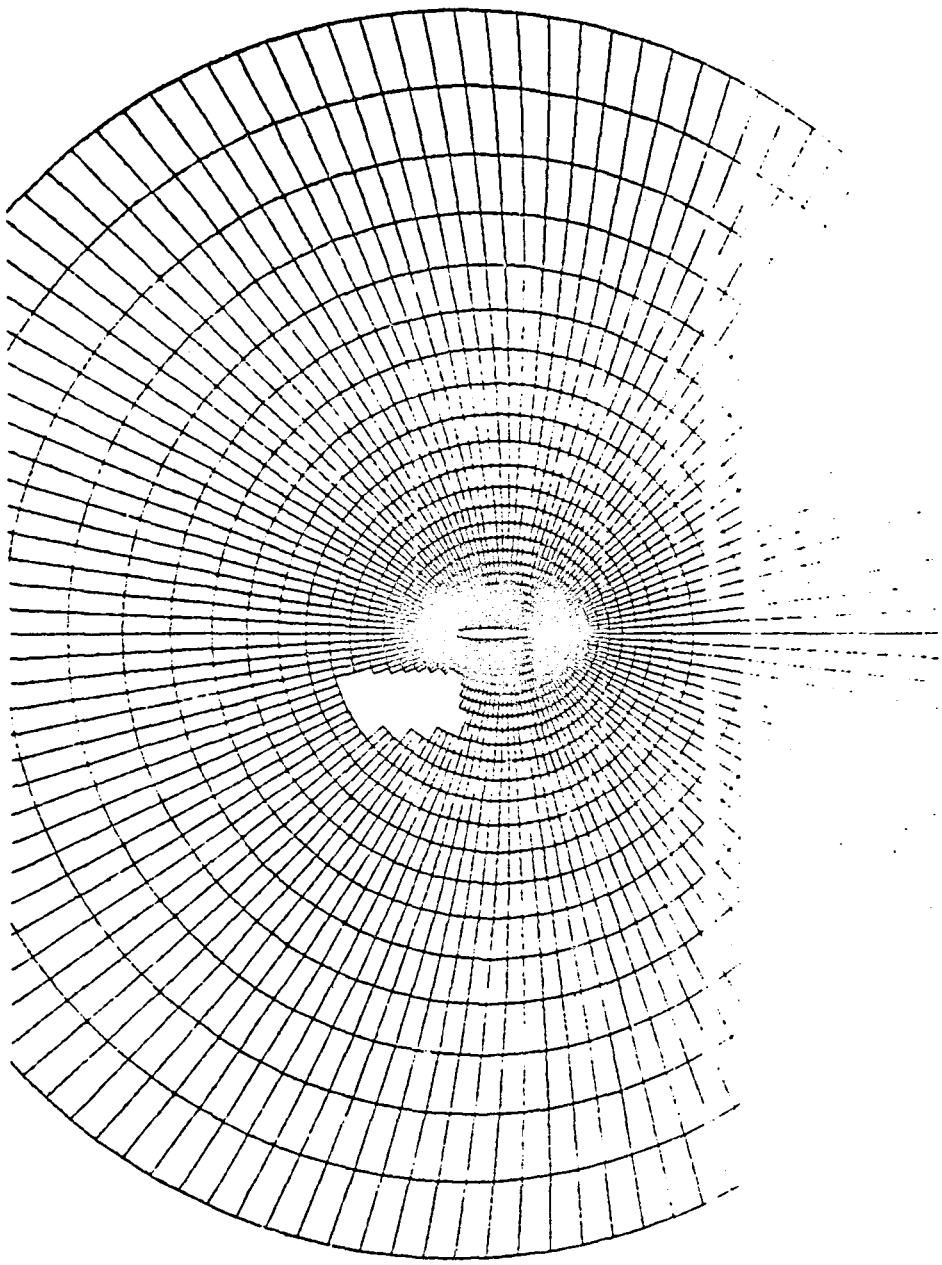
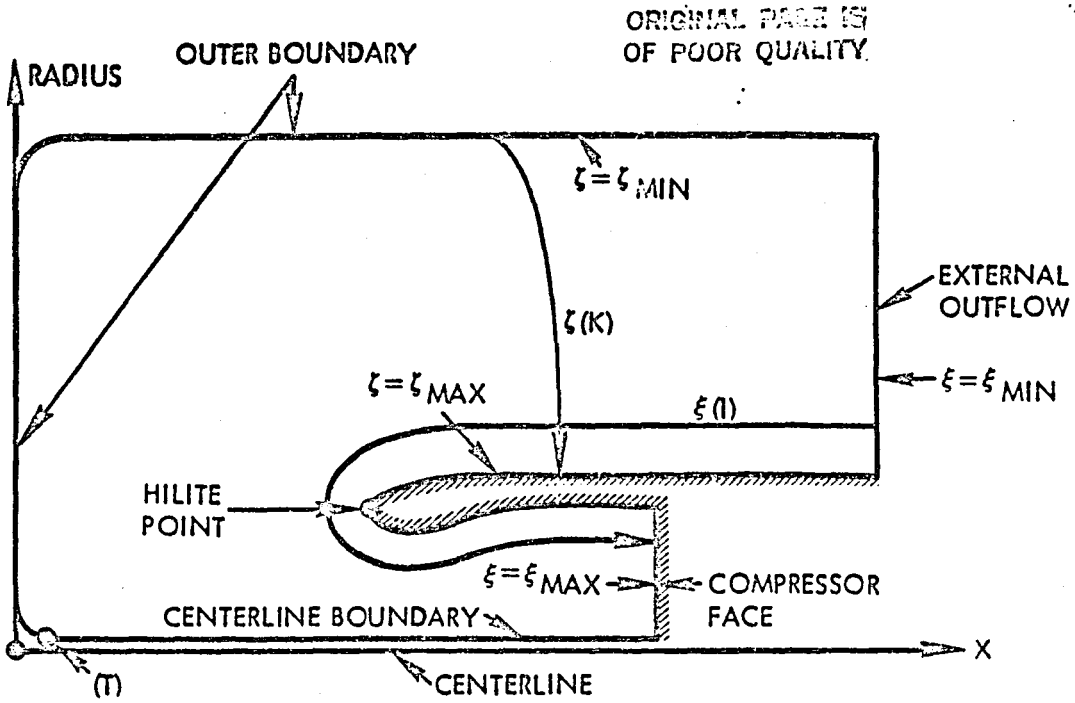
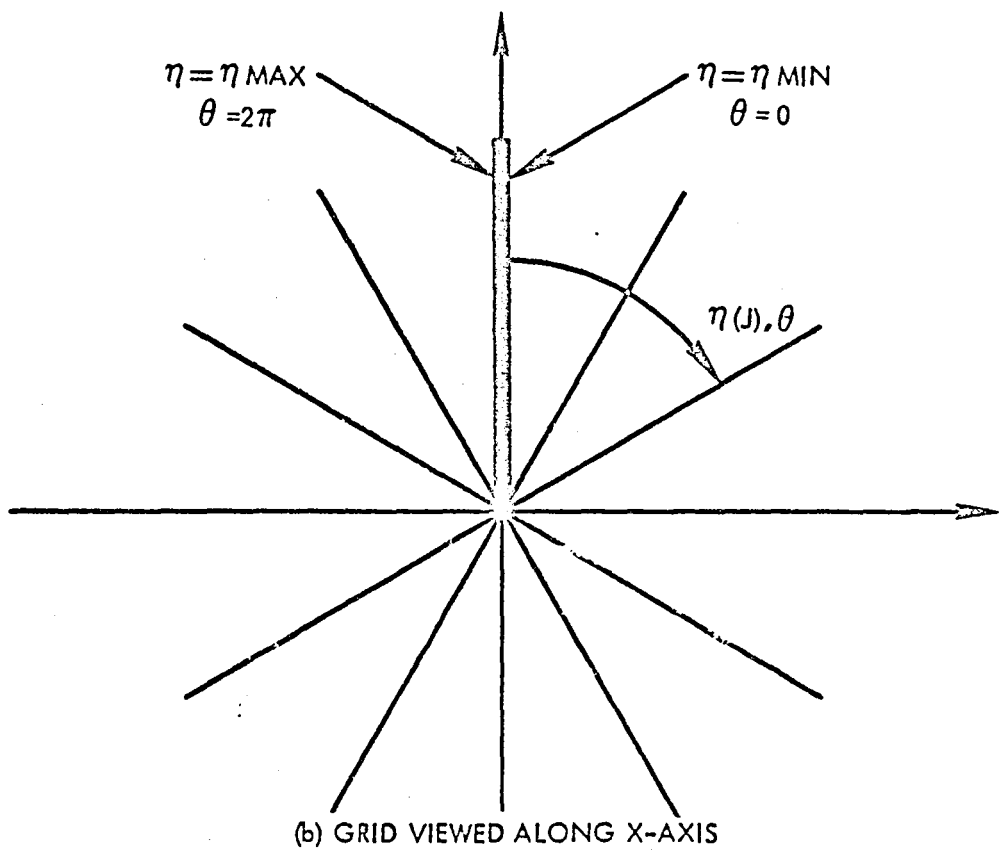


Figure 2. Side view of wing component grid at a span station containing the nacelle.



(a) MERIDIONAL PLANE GRID TOPOLOGY



(b) GRID VIEWED ALONG X-AXIS

Figure 3. Isolated nacelle grid topology.

topology, whereas Figure 3b illustrates the grid as viewed along the longitudinal axis of the inlet which is the local x axis. The computational curvilinear coordinates are denoted by ξ, η, and ζ. The ξ coordinate is in the wraparound direction, initiates at the external outflow surface, and terminates at the compressor face outflow surface. The η coordinate is in the circumferential direction, initiates at the $\theta = 0$ meridional plane, and terminates at the $\theta = 2\pi$ meridional plane. The ζ coordinate is in the radial direction, initiates at the outer computational boundary (or centerline boundary), and terminates at the body surface. The $\theta = 0$ and $\theta = 2\pi$ meridional planes are coincident. If a pylon were present, then the pylon geometry would be contained between these two meridional stations.

The surface grid points are clustered in the region of the nacelle hilite (leading edge point of the nacelle). The clustering is achieved by using a geometric stretching function which is expressed in terms of the arc length measured along the body from the hilite point. The outer computational boundary point distribution is determined by using either an angular distribution function expressed in terms of an angle measured about the nacelle hilite, or by using an arc length distribution along the outer boundary.

Once the surface and outer boundary mesh point distributions have been determined, the interior mesh point locations are computed using the NASA-Ames GRAPE[6] algorithm. The GRAPE algorithm determines the interior field point coordinates by iteratively solving two coupled Poisson equations. For axisymmetric geometries, this two-dimensional grid generation procedure is applied for only one meridional plane, and the grid point locations on the remaining meridional planes are found by using simple reflection techniques. For asymmetric geometries, the two-dimensional grid generation algorithm can be applied for each meridional plane.

The GRAPE two-dimensional grid generation algorithm is contained within the NGRIDA nacelle grid generation program and is used solely to obtain the interior grid point locations. The boundary point locations are computed in NGRIDA and are supplied to the GRAPE subalgorithm to be used as boundary conditions. After the interior field point coordinates have been determined, point reordering, and grid translation, scaling, and reflection are performed in NGRIDA. The detailed theory of the NGRIDA nacelle grid generation algorithm is discussed in Reference 4.

Figure 4 illustrates the rectangular computational domain which corresponds to the physical domain shown in Figure 3. The physical space boundaries transform into boundaries in computational space.

Figure 5 illustrates a typical meridional plane interface grid. The inlet centerline, which is the x axis, represents a singularity in the three-dimensional grid mapping. The computational grid boundary is displaced a small distance away from the x axis. An extrapolation and averaging technique, described later, is used to obtain flow properties on the centerline.

3. GOVERNING EQUATIONS FOR THE INVISCID FLOW

The inviscid flow gas dynamic model is based on the assumption of steady

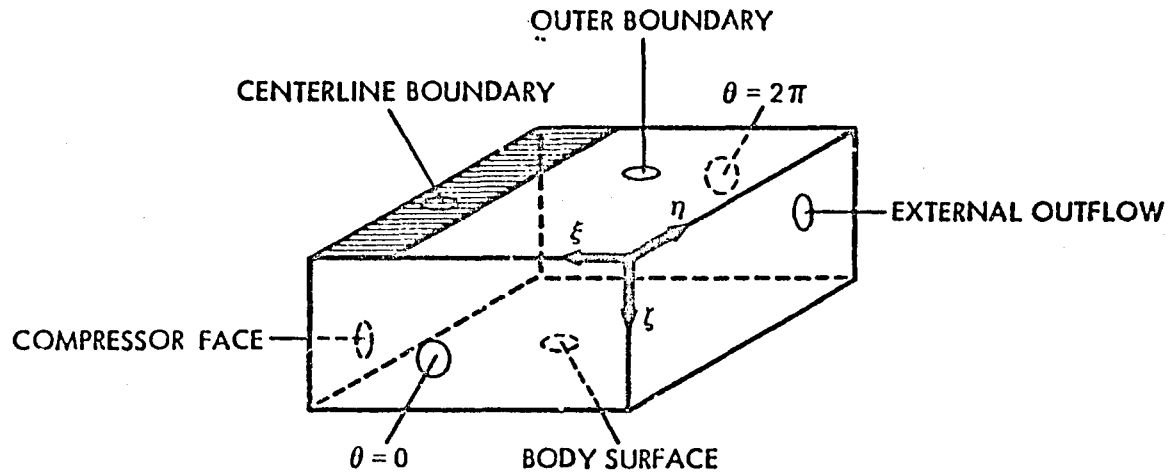


Figure 4. Nacelle computational domain.

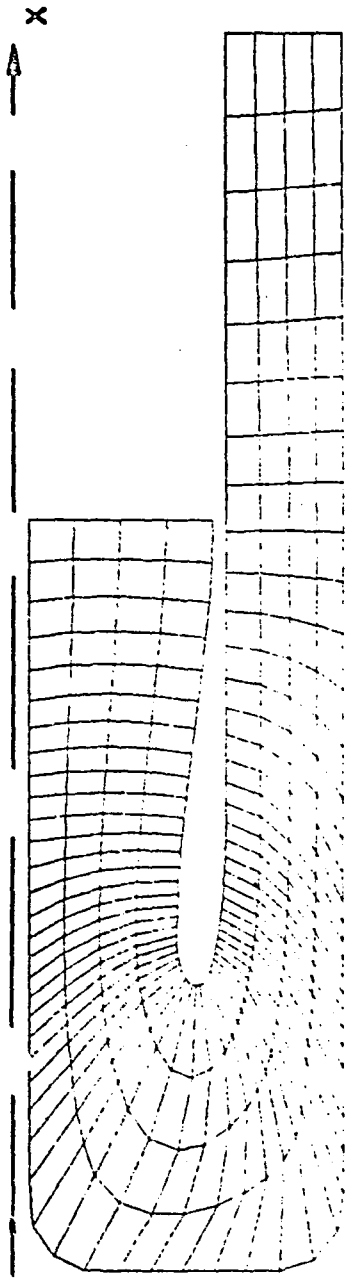


Figure 5. Side view of nacelle component grid.

potential flow which requires that the flow be both irrotational and isentropic. The governing equations for steady three-dimensional potential flow are given by

$$\left(\frac{\rho U}{J}\right)_\xi + \left(\frac{\rho V}{J}\right)_\eta + \left(\frac{\rho W}{J}\right)_\zeta = 0 \tag{3}$$

$$\rho = \left[1 - \frac{\gamma-1}{\gamma+1}\left(U\phi_\xi + V\phi_\eta + W\phi_\zeta\right)\right]^{\frac{1}{\gamma-1}} \tag{4}$$

where ξ, η, and ζ denote the system of curvilinear coordinates, U, V, and W are the contravariant velocity components in the ξ, η, and ζ directions, respectively, ρ is the density, J is the Jacobian of transformation from the Cartesian coordinate system (x, y, z) to the general curvilinear coordinate system (ξ, η, ζ), ϕ is the velocity potential function, and γ is the specific heat ratio. The density and contravariant velocity components are normalized by the stagnation density and critical sonic speed, respectively.

Equation (3) is the full-potential equation in strong conservation law form. It expresses mass continuity for steady three-dimensional flows. Equation (4) expresses entropy conservation and is used to compute the density given the velocity potential field.

The contravariant velocity components can be expressed in terms of the Cartesian velocity components as

$$U = u\xi_x + v\xi_y + w\xi_z \tag{5}$$

$$V = u\eta_x + v\eta_y + w\eta_z \tag{6}$$

$$W = u\zeta_x + v\zeta_y + w\zeta_z \tag{7}$$

where u, v, and w denote the nondimensional velocity components along the x, y, and z Cartesian coordinate axes, respectively. The contravariant velocity components can be also expressed in terms of the velocity potential ϕ as

$$U = A_1\phi_\xi + A_4\phi_\eta + A_5\phi_\zeta \tag{8}$$

$$V = A_4\phi_\xi + A_2\phi_\eta + A_6\phi_\zeta \tag{9}$$

$$W = A_5\phi_\xi + A_6\phi_\eta + A_3\phi_\zeta \tag{10}$$

The metric parameters A_i (i = 1 to 6) and the Jacobian of transformation J are given by

$$\begin{aligned} A_1 &= \xi_x^2 + \xi_y^2 + \xi_z^2 & A_4 &= \xi_x\eta_x + \xi_y\eta_y + \xi_z\eta_z \\ A_2 &= \eta_x^2 + \eta_y^2 + \eta_z^2 & A_5 &= \xi_x\zeta_x + \xi_y\zeta_y + \xi_z\zeta_z \\ A_3 &= \zeta_x^2 + \zeta_y^2 + \zeta_z^2 & A_6 &= \eta_x\zeta_x + \eta_y\zeta_y + \eta_z\zeta_z \end{aligned} \tag{11}$$

$$J = \xi_x\eta_y\zeta_z + \xi_y\eta_z\zeta_x + \xi_z\eta_x\zeta_y - \xi_z\eta_y\zeta_x$$
$$- \xi_y\eta_x\zeta_z - \xi_x\eta_z\zeta_y \qquad (12)$$

The following metric relations are also required in defining the mapping from physical space to computational space:

$$\xi_x = J(y_\eta z_\zeta - y_\zeta z_\eta) \qquad \eta_x = J(y_\zeta z_\xi - y_\xi z_\zeta) \qquad \zeta_x = J(y_\xi z_\eta - y_\eta z_\xi)$$

$$\xi_y = J(x_\zeta z_\eta - x_\eta z_\zeta) \qquad \eta_y = J(x_\xi z_\zeta - x_\zeta z_\xi) \qquad \zeta_y = J(x_\eta z_\xi - x_\xi z_\eta) \qquad (13)$$

$$\xi_z = J(x_\eta y_\zeta - x_\zeta y_\eta) \qquad \eta_z = J(x_\zeta y_\xi - x_\xi y_\zeta) \qquad \zeta_z = J(x_\xi y_\eta - x_\eta y_\xi)$$

The metric parameters are obtained numerically using standard second-order or fourth-order accurate finite-difference formulae to compute derivatives of the form x_ξ, x_η, x_ζ, y_ξ, etc. Then using the metric relations given by equations (13), the inverse quantities ξ_x, η_x, etc., are determined. Substitution of these values into equations (11) and (12) yields the A_1 and Jacobian J.

4. NUMERICAL ALGORITHM

4.1 BASIC FINITE-DIFFERENCE SCHEME

The present numerical algorithm is based on the finite-difference formulation used by Holst and Thomas[3] in computing transonic wing flows. In this algorithm, the full-potential equation residual is given by

$$L\phi_{i,j,k} = \overleftarrow{\delta}_\xi \left(\frac{\rho U}{J}\right)_{i+\frac{1}{2},j,k} + \overleftarrow{\delta}_\eta \left(\frac{\bar{\rho} V}{J}\right)_{i,j+\frac{1}{2},k} +$$
$$\overleftarrow{\delta}_\zeta \left(\frac{\hat{\rho} W}{J}\right)_{i,j,k+\frac{1}{2}} \qquad (14)$$

where $L\phi_{i,j,k}$ denotes the residual operator, and i, j, and k denote the grid point indices in the ξ (wraparound), η (circumferential or spanwise), and ζ (radial) directions, respectively. The magnitude of the residual $L\phi_{i,j,k}$ approaches zero as convergence is attained. The operators $\overleftarrow{\delta}_\xi(\ )$, $\overleftarrow{\delta}_\eta(\ )$, and $\overleftarrow{\delta}_\zeta(\ )$ are first-order accurate backward finite-difference operators (applied at midpoints) for the ξ, η, and ζ directions, respectively. The terms $\tilde{\rho}$, $\bar{\rho}$, and $\hat{\rho}$ are upwind-biased density coefficients given by expressions of the form

$$\tilde{\rho}_{i+\frac{1}{2},j,k} = [(1-\nu)\rho]_{i+\frac{1}{2},j,k}$$
$$+ \nu_{i+\frac{1}{2},j,k}\, \rho_{i+\frac{1}{2}+r,j,k} \qquad (15)$$

where r denotes an upwind point along the ξ direction, and ν is an artificial viscosity coefficient. In equation (15), the physical density ρ is computed using equation (4) with central differences used for determining the derivatives of Φ. The artificial viscosity coefficient ν is given by

$$\nu = 0 \quad \text{IF} \quad M_{i,j,k} < 1$$

$$\nu = C \quad (M^2_{i,j,k} - 1) \quad \text{IF} \quad M_{i,j,k} > 1 \tag{16}$$

where M is the local Mach number, and C is a user-specified constant. The artificial viscosity coefficient C typically ranges from 1.0 to 2.0, with the larger values producing greater upwinding. Expressions similar to equation (15) hold for $\bar{\sigma}$ and $\hat{\rho}$ which effect upwinding in the η and ζ directions, respectively.

The finite-difference equations are solved using the AF2 approximate factorization scheme which has proved to be significantly more efficient than successive-line-overrelaxation schemes[3]. The AF2 algorithm is written in a three-step form as:

First-Step:

$$(\alpha - \frac{1}{A_k} \vec{\delta}_\eta A_j \overleftarrow{\delta}_\eta)\, g^n_{i,j} = \alpha\omega L\phi^n_{i,j,k} + A_{k+1} f^n_{i,j,k+1} \tag{17}$$

Second-Step:

$$(A_k \mp \beta_\xi \vec{\delta}_\xi - \frac{1}{\alpha} \vec{\delta}_\xi A_i \overleftarrow{\delta}_\xi)\, f^n_{i,j,k} = g^n_{i,j} \tag{18}$$

Third-Step:

$$(\alpha + \overleftarrow{\delta}_\zeta) C^n_{i,j,k} = f^n_{i,j,k} \tag{19}$$

In equations (17) to (19), α is a time-like factorization parameter chosen to maintain stability and attain fast convergence, β_ξ is a factor that controls the amount of dissipation required in regions of supersonic flow, ω is a relaxation factor, n is the iteration number, $L\phi_{i,j,k}$ is the mass residual [defined by equation (14)], f and g are intermediate functions which are obtained during the solution process, and $C_{i,j,k}$ is the potential function correction given by

$$C^n_{i,j,k} = \phi^{n+1}_{i,j,k} - \phi^n_{i,j,k} \tag{20}$$

The terms A_i, A_j, and A_k are defined by

$$A_i = (\bar{\rho}A_1/J)^n_{i-\frac{1}{2},j,k} \tag{21}$$

$$A_j = (\bar{\rho}A_2/J)^n_{i,j-\frac{1}{2},k} \tag{22}$$

$$A_k = (\bar{\rho}A_3/J)^n_{i,j,k-\frac{1}{2}} \tag{23}$$

In Steps 1 and 2 the g and f functions are obtained by solving a tridiagonal system of equations while in Step 3 the correction $C_{i,j,k}$ is obtained by solving a bidiagonal system of equations.

The iterative relaxation procedure can be viewed as a time-marching integration algorithm in pseudotime. Using this analogy, the factorization parameter α appearing in equations (17), (18), and (19) can be regarded as the reciprocal of the marching pseudotime step. To eliminate all components of the error frequency spectrum, it is generally preferable to employ a variable time step which sequentially varies with iteration number. The small values of α are particulary effective in reducing the low frequency errors, while the large values of α are effective in reducing the high frequency errors.

The factorization parameter α is computed using a repeating sequence which is based on iteration number[1]. The α sequence that is employed is given by

$$\alpha_k = \alpha_h \, (\alpha_1/\alpha_h)^{\frac{k-1}{M-1}} \qquad (k = 1,2,\cdots,M) \tag{24}$$

where α_1 is the lower limit of α, α_h is the upper limit, and α_k is the value of α for the kth element of the sequence. For all cases presented herein, the number of elements in the sequence M was equal to 8. The optimum values of α_1 and α_h are generally determined by numerical experiment.

4.2 BOUNDARY CONDITIONS

At the surface of the wing or nacelle, the contravariant velocity W in the ζ-coordinate direction is equated to zero as this satisfies the flow tangency condition. At a body surface, the derivative term ϕ_ζ is calculated from

$$\phi_\zeta = -(A_5\phi_\xi + A_6\phi_\eta)/A_3 \tag{25}$$

where the derivatives ϕ_ξ and ϕ_η are found using second-order differencing in the ζ=constant boundary surface.

At the compressor face outflow surface inside the inlet, the contravariant velocity U_o in the ξ-coordinate direction is specified. Given the local Cartesian velocity components u, v, and w, and the metric quantities ξ_x, ξ_y, and ξ_z, U_o can be determined using equation (5). At the compressor face, the flow is assumed to be uniform and in the axial (x) direction. The local flow velocity and density are fixed by specification of the required engine mass flow rate. Once U_o has been determined, the derivative term ϕ_ξ is calculated from

$$\phi_\xi = (U_o - A_4\phi_\eta - A_5\phi_\zeta)/A_1 \qquad (26)$$

where ϕ_η and ϕ_ζ are found using second-order differencing in the ξ=constant boundary surface.

For the nacelle component algorithm, the grid centerline boundary is offset a small distance from the centerline to avoid the mapping singularity at the centerline. To determine property values on the centerline, an extrapolation and averaging procedure is employed, which is fully discussed in Reference 4.

At the wing symmetry plane boundary, the contravariant velocity V in the η-coordinate direction is equated to zero. At the wing free-stream sidewall boundary, the initial free-stream distribution of ϕ is held fixed as is the ϕ distribution for the wing grid outer boundary for non-lifting cases. For lifting calculations, the ϕ distribution on the wing grid outer boundary is updated using a compressible vortex solution as this improves convergence speed[3]. The wing circulation is computed at the end of each wing algorithm iteration by calculating the potential jump along the vortex sheet.

The nacelle component grid outer boundary constitutes the grid overlap region outer boundary. The grid overlap region inner boundary is constituted by the irregularly shaped boundary which is obtained from deleting selected interior field points from the wing component grid in order to accommodate the nacelle geometry (see Figure 2). For the overlap region inner and outer boundaries, the velocity potential ϕ or the velocity potential normal derivative ϕ_n can be specified. Both options were tried in two dimensions[7] and no significant differences in the final results were observed. However, specifying the velocity potential which is a scalar quantity is easier to implement and requires less computational effort as compared to specifying the velocity potential normal derivative.

The portion of the overlap region inner boundary adjacent to the engine exhaust plane at the aft end of the nacelle requires special treatment. A uniform axial velocity is specified at the nacelle aft station and a three-point extrapolated difference is used to provide the required values of the potential function at the wing grid points immediately downstream of this station.

In addition to specifying ϕ on the overlap region boundaries, it is necessary to specify boundary conditions for the f and g intermediate corrections for use in the wing component algorithm. For the f intermediate correction in the ξ-wraparound coordinate direction, the Dirichlet boundary condition $f = 0$ is imposed on the overlap region inner boundary in performing the wing component flow field calculations. For the g intermediate correction in the η-spanwise coordinate direction, the Neumann boundary condition $g_\eta = 0$ is imposed on the overlap region inner boundary. Both the f and g corrections approach zero as the iterative procedure progresses and convergence is attained.

For the nacelle component algorithm, the Dirichlet boundary condition $f = 0$ is imposed at the external outflow surface of the nacelle grid in solving the system of factorized difference equations in the ξ-wraparound coordinate direction. Periodic differencing is employed in the η-circumferential direction.

4.3 ITERATION PROCEDURE

The computation is started by initializing the entire velocity potential field. For the external flow around the wing and nacelle cowling, the initialization of the potential field is performed using free-stream velocity components. For the internal flow initialization inside the inlet, the local velocity is assumed to be axial and is computed using a Mach number which is determined from the implicit relation

$$M\left(1 + \frac{\gamma-1}{2} M^2\right)^{\frac{-(\gamma+1)}{2(\gamma-1)}} = \left(\frac{A_{cf}}{A}\right) M_{cf} \left(1 + \frac{\gamma-1}{2} M_{cf}^2\right)^{\frac{-(\gamma+1)}{2(\gamma-1)}} \qquad (27)$$

where M and M_{cf} denote the local and compressor face Mach numbers, respectively, and A and A_{cf} denote the local and compressor face flow areas, respectively. Equation (27) was obtained using one-dimensional gas dynamic formulae and simply expresses mass continuity for the captured stream tube. To determine the internal potential field, a trapezoidal rule integration is used. The compressor face Mach number M_{cf} is fixed by specification of the required engine mass flow rate.

Once the entire potential field has been initialized, the wing algorithm is then executed for a specified number of iterations (typically 40 - 50) holding ϕ constant on the overlap inner boundary. The nacelle algorithm is then executed for a specified number of iterations (typically 40 - 50) using overlap outer boundary conditions as determined from the wing solution. At this stage, the wing algorithm is again executed using updated overlap inner boundary conditions obtained from the nacelle solution. This process is repeated until overall convergence is achieved. Typically 4 to 5 cycles are required for convergence with 40 to 50 iterations being performed in both the wing and nacelle component algorithms per cycle.

4.4 INTERGRID PROPERTY TRANSFER

Multivariate interpolation is employed to transfer property information

between the component grids. To effect intergrid property transfer for the wing/nacelle configuration requires employing a trivariate interpolation polynomial since the component grids do not have a common coordinate surface. To reduce the coding logic, a trivariate polynomial based on a linear Taylor series expansion was employed. Interpolation is performed in computational (ξ,η,ζ) space using a polynomial of the form

$$f(\xi,\eta,\zeta) = f_o + f_{\xi o}\,(\xi - \xi_o) + f_{\eta o}\,(\eta - \eta_o) + f_{\zeta_o}\,(\zeta - \zeta_o) \qquad (28)$$

where $f(\xi,\eta,\zeta)$ denotes the interpolated flow property at the point (ξ,η,ζ), and the subscript o denotes a selected interpolation base point in computational space. The derivatives appearing in equation (28) are evaluated using standard second-order accurate difference formulae. Point (ξ_o, η_o, ζ_o) is selected as the mesh point closest to the desired interpolation point (ξ, η, ζ).

The use of the trivariate interpolation polynomial given by equation (28) in some cases produced oscillations in the interpolated values of the velocity potential. The nacelle component grid lies in a region where the wing component grid is relatively coarse, thereby creating a disparity in the average cell size between the two grids in the overlap region where the interpolation is performed. It was found that the best results were obtained by employing a relatively fine grid for the wing and a relatively coarse grid for the nacelle. To ensure a monotonic behavior for the interpolated overlap boundary potential values, explicit smoothing is performed in the streamwise direction after each cycle of the wing/nacelle solution.

5. NUMERICAL RESULTS

Selected numerical results are now presented to illustrate application of the analysis.

The first set of computed results are for a hypothetical wing/nacelle configuration. The wing geometry is represented by a NACA 0012 wing which has an aspect ratio of 6.0, zero twist, and which is unswept and untapered. The nacelle geometry is represented by the Lockheed-Georgia GELAC1 axisymmetric nacelle/inlet configuration. The nacelle is located at the 50% span station and has its axis parallel to the wing chord.

The wing geometry is the default geometry contained within the GRGEN3 wing grid generation code. The nacelle geometry is the default geometry contained within the NGRIDA nacelle grid generation code. The results of isolated nacelle flow field computations for the GELAC1 nacelle are presented in Reference 4.

The grid topologies used for this wing/nacelle multicomponent flow field computation are illustrated in Figures 6, 7, and 8. Figure 6 shows a frontal view of the composite overlapped grid system. The outer wing grid is generated for a series of spanwise stations. The span stations are clustered near the location of the nacelle. The inner nacelle grid is generated for a series of meridional planes spaced in equal angular increments around the body. A side view of the composite grid system at the span station coincident

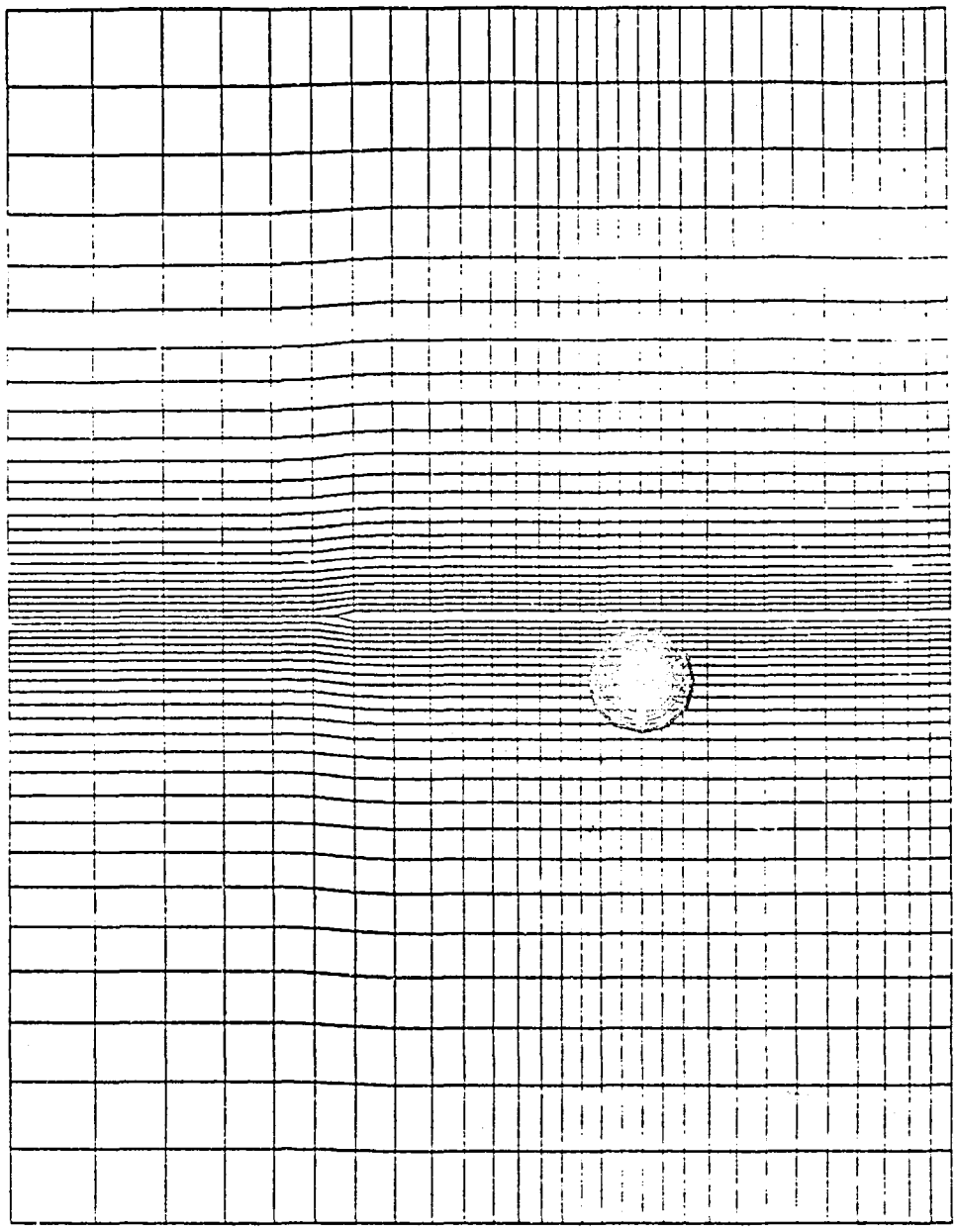


Figure 6. Front view of NACA0012-wing/GELAC1-nacelle configuration global grid.

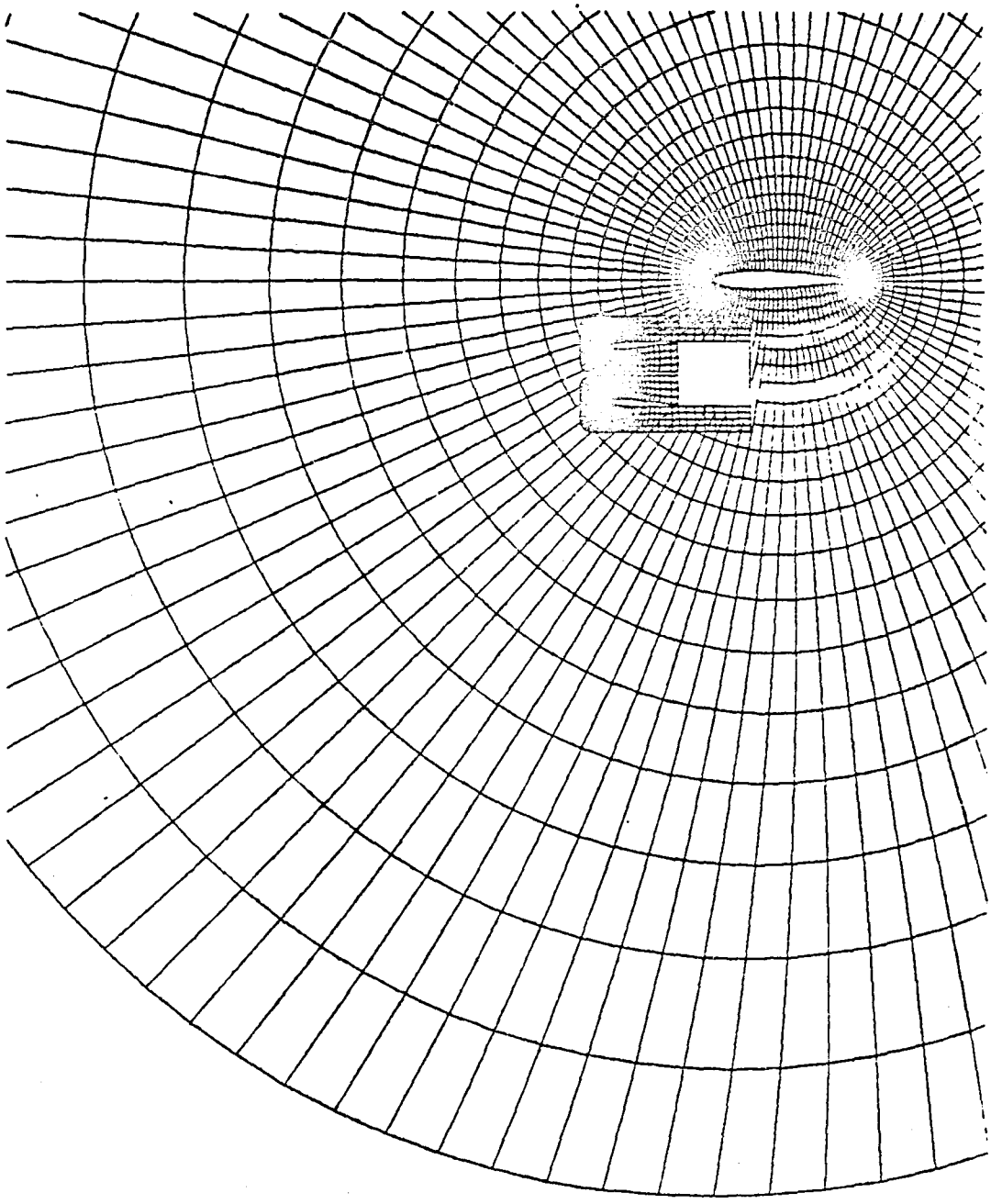


Figure 7. Side view of NACA0012-wing/GELAC1-nacelle configuration global grid.

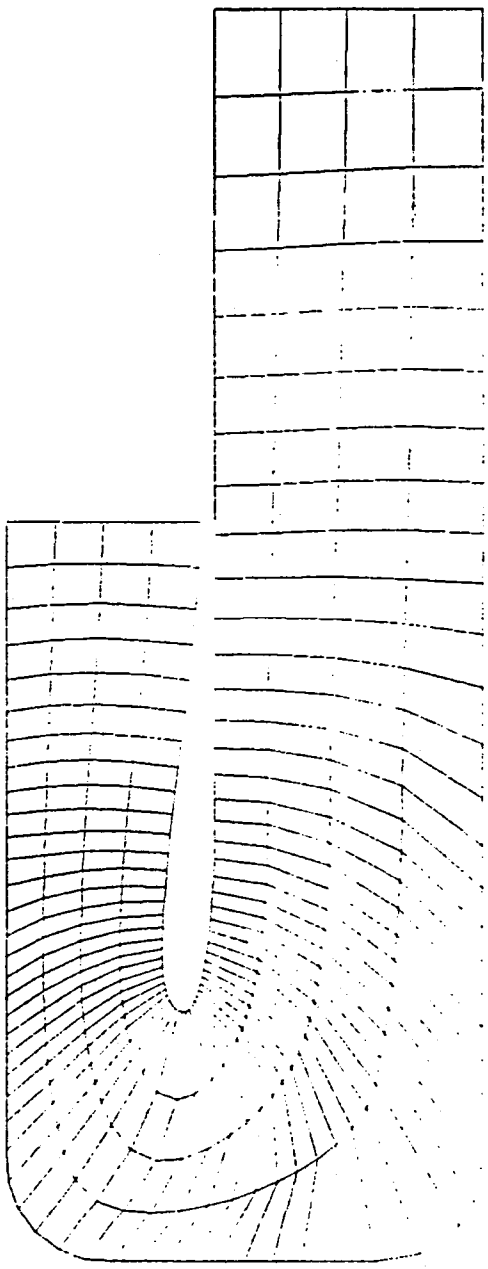


Figure 8. GELAC1 nacelle component grid.

with the nacelle symmetry plane is given in Figure 7. The overlap region between the wing O-grid and the nacelle C-grid is clearly illustrated in this figure. The nacelle interface grid is illustrated in isolation in Figure 8. The grid generation procedure used to obtain a nacelle interface grid is almost identical to that used to obtain an isolated nacelle grid except that for an interface grid the outer computational boundary is placed close to the body and typically the number of radial stations is reduced. The wing component grid for this case consisted of 101 wraparound stations, 22 spanwise stations, and 30 normal stations. The nacelle component grid consisted of 67 wraparound stations, 13 circumferential stations, and 5 radial stations.

Figure 9 illustrates the computed surface solution for this wing/nacelle configuration at a free-stream Mach number M_∞ of 0.63 and at an angle of attack α of 2.0 degrees. Shown in this figure is the surface pressure coefficient c_p versus the fractional distance along the chord x/c (distance/chord length) for the 50% span station ($\eta = 0.5$). Results are presented for the wing/nacelle multicomponent flow calculation and for an isolated wing calculation. The wing/nacelle calculation was performed assuming that the inlet capture ratio was 0.7 and that the ratio of the engine exhaust plane velocity to the free-stream velocity was 2.25. The effect of including the nacelle is to reduce the suction level on the upper wing surface and to increase it on the lower surface near the aft portion of the nacelle (wing leading edge). The high engine exhaust jet velocity increases the flow velocity near the lower wing surface, thereby reducing the circulation and hence the lift. The midspan wing section lift coefficient for the isolated wing calculation was 0.287. For the wing/nacelle calculation, the midspan section lift coefficient was reduced to 0.24. These results agree qualitatively with what is observed experimentally[8]. The wing/nacelle mutual interference calculation required four cycles through both the wing and nacelle component algorithms to achieve convergence which corresponds to a three order of magnitude reduction in the maximum residual. The computation required approximately 5 hours of CPU execution time on the DEC VAX-11/780 computer. Approximately one million words of core storage were required for the computation.

Figure 10 illustrates the computed midspan wing pressure distribution for the NACA0012-wing/GELAC1-inlet configuration for the conditions of a free-stream Mach number of 0.75 and 1 degree angle of attack. For this computation, the inlet capture ratio was maintained at 0.7 but the ratio of the engine exit plane velocity to the free stream velocity was now equated to 1.15. Again, results are presented for an isolated wing solution and a wing/nacelle mutual interference solution. In this case, the wing upper surface pressure distribution is slightly altered by the presence of the nacelle, however, the lower surface pressure distribution is markedly different. In this case, the lower surface flow undergoes a rapid expansion followed by a strong shock at the 40% chord location. The net effect is a reduction in the sectional lift. These results indicate that the sense and magnitude of the nacelle interference effects are strongly dependent on the exhaust jet velocity, free-stream conditions, and nacelle location.

The next set of computed results are for a recently-designed Lockheed-Georgia wing/nacelle configuration. The wing geometry employs supercritical airfoil sections and is both swept and tapered. The nacelle geometry is again the GELAC1 axisymmetric nacelle/inlet configuration. The

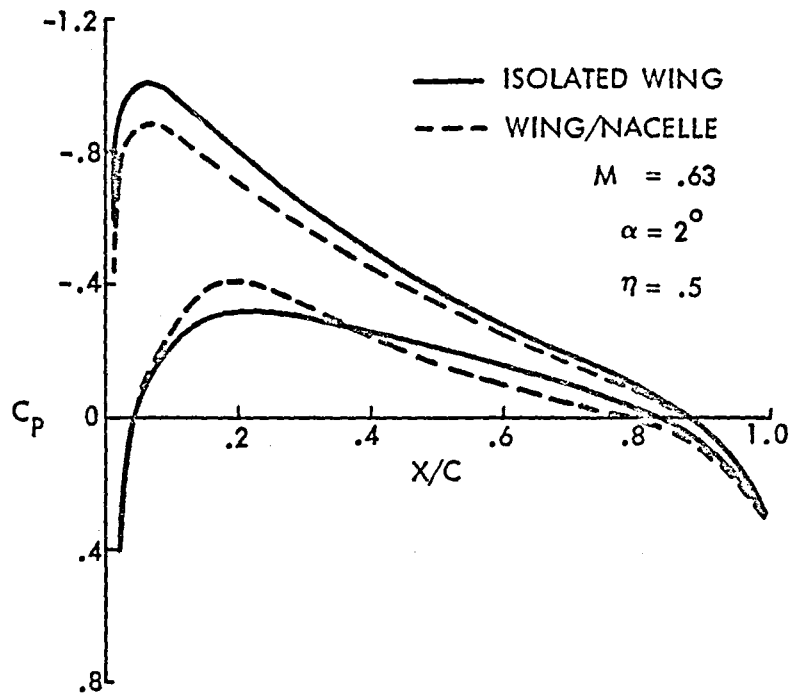


Figure 9. Wing pressure distribution for NACA0012-wing/GELAC1-nacelle configuration at $M_\infty = .63$ and $\alpha = 2°$.

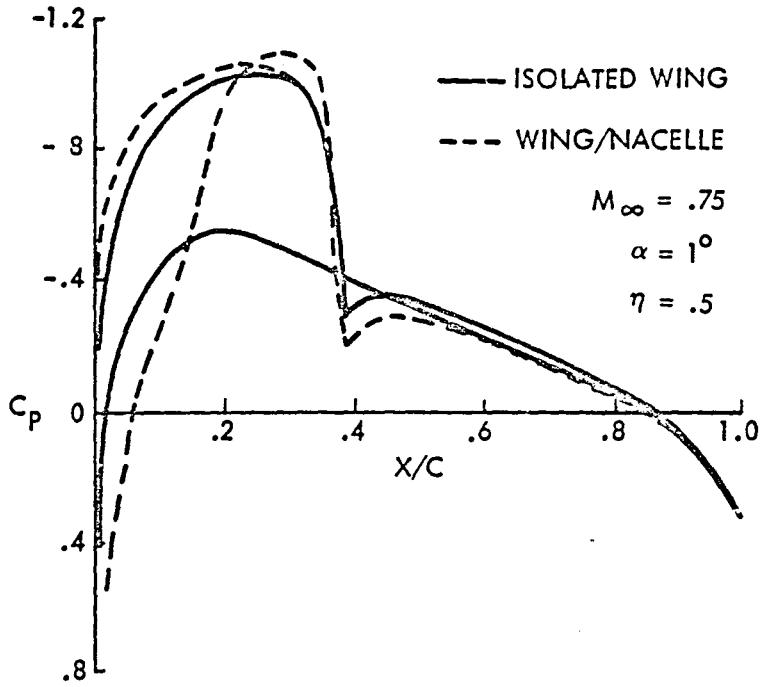


Figure 10. Wing pressure distribution for NACA0012-wing/CELAC1-nacelle configuration at M_∞ = .75 and α = 1°.

nacelle plane of symmetry is again located at the wing midspan ($\eta = 0.5$). Figures 11, 12, and 13 illustrate the global computational grid for this wing/nacelle configuration. Figure 11 shows a frontal view of the overlapped component grid system, whereas Figure 12 shows a side view at the span location of the nacelle symmetry plane. Figure 13 illustrates a planform view of the wing and nacelle. The wing component grid employed 101 wraparound stations, 22 spanwise stations, and 30 normal stations. The nacelle component grid employed 67 wraparound stations, 13 circumferential stations, and 5 radial stations.

Figures 14, 15, and 16 present the computed wing pressure distributions for this wing/nacelle configuration at the 46%, 50%, and 60% span stations, respectively. The free-stream Mach number and incidence used in the computations correspond to 0.77 and 2 degrees, respectively. On each of the three figures, the computed solution is shown for an isolated wing calculation and for wing/nacelle calculations using exhaust jet to free stream velocity ratios (v_j/v_∞) of 1.2 and 1.9. The inlet capture ratio was 0.7. The nacelle interference effect for each of the three spanwise locations was to reduce the suction level on the wing upper surface with this effect being more pronounced as the exhaust jet velocity was increased. As the upper surface suction level was reduced, the shock was moved slightly forward for each of the three span locations. The interference effect of the nacelle on the wing lower surface was to increase the suction levels near midchord. Again this effect was more pronounced at the higher exhaust jet velocity. On the lower surface near the wing leading edge, the surface pressure was higher for the wing/nacelle configuration than the isolated wing except at the inboard station where the pressures were comparable. For all three span stations illustrated, the interference effect of the nacelle decreased the sectional lift coefficients as compared to the isolated wing results. Moreover, the reduction in sectional lift became more pronounced as the exhaust jet velocity was increased. Overall, the computed effects of the nacelle interference with the wing flow field agree qualitatively with experimental observations.

The final set of results are for the Lockheed C-5A military transport aircraft[9]. Figures 17, 18, and 19 present a frontal view, side view, and planform view of the overlapped grid system used for this computation. Figure 20 illustrates the C-5A nacelle interface grid. The contours of the nacelle used in the computation represent a flow-through nacelle which is used to simulate the actual turbofan flight nacelle geometry which is illustrated in Figure 21. The wing component grid employed 101 wraparound stations, 22 spanwise stations, and 30 normal stations. The nacelle component grid employed 67 wraparound stations, 13 circumferential stations, and 5 radial stations.

Figures 22 and 23 illustrate the computed and experimental wing pressure distributions at the 39% and 46% span stations, respectively. These results are for a free-stream Mach number of 0.775 and zero incidence. The nacelle capture ratio was assumed to be 0.55 and the engine exit plane exhaust velocity was taken as 1.125 times that of the free-stream velocity. Figure 22 presents computed and experimental results for both an isolated wing and a wing/nacelle configuration. Figure 23 presents results for the wing/nacelle case only. Referring to the experimental results shown on Figure 22, the effect of adding the pylon and nacelle is to decrease the peak suction level on the wing upper surface. This trend is predicted by the analysis qualitatively. The analysis, however, overpredicts the suction level on the

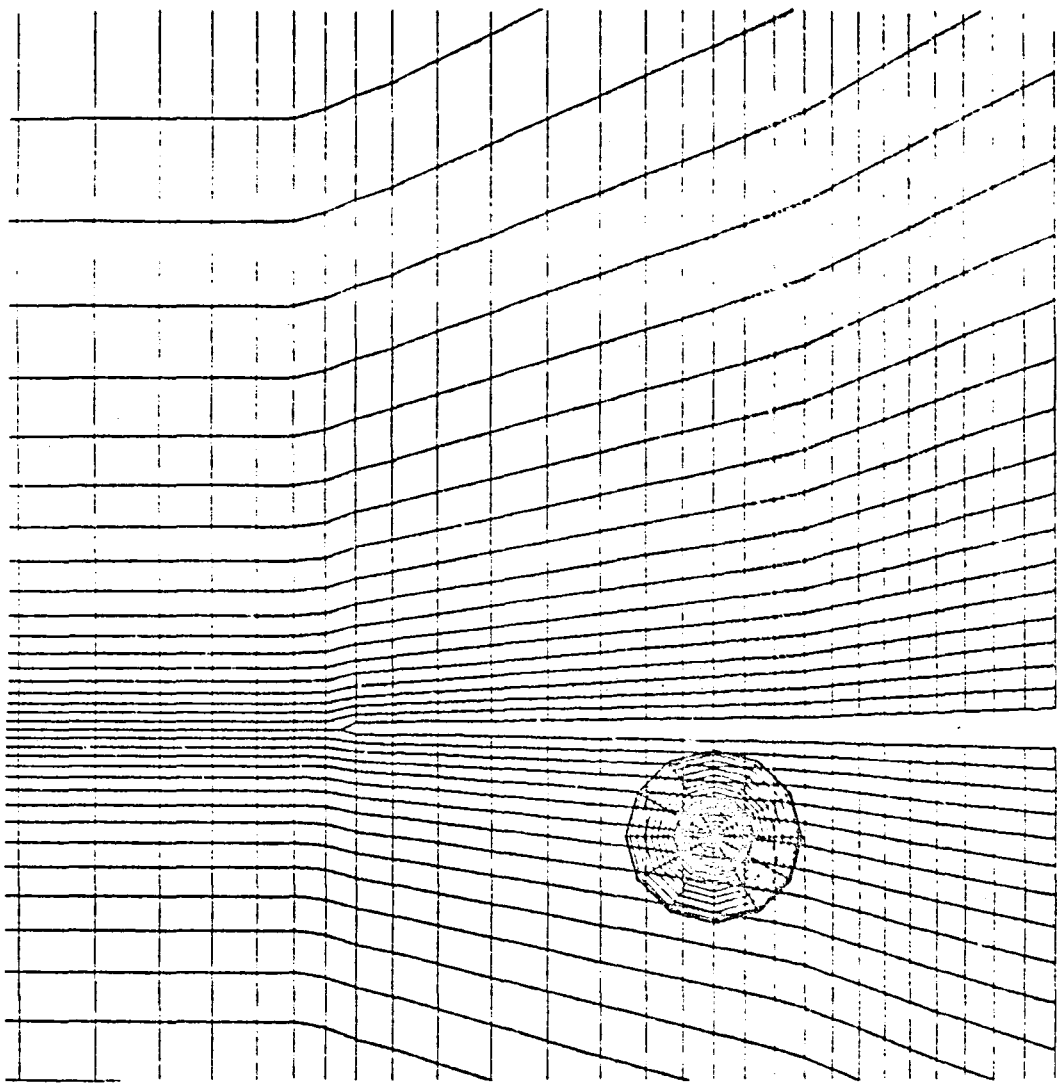


Figure 11. Front view of Lockheed-Georgia wing/nacelle configuration global grid.

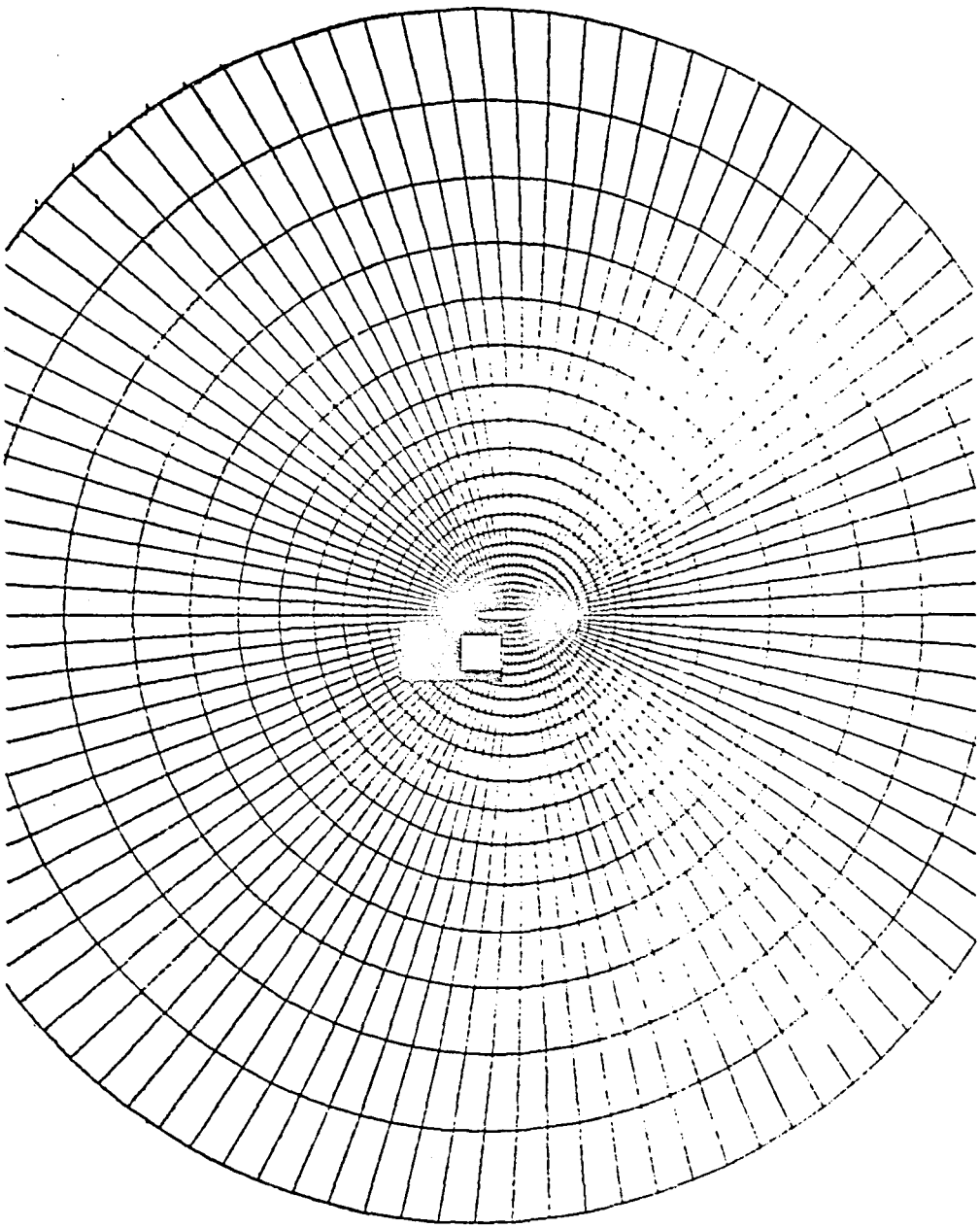


Figure 12. Side view of Lockheed-Georgia wing/nacelle configuration global grid.

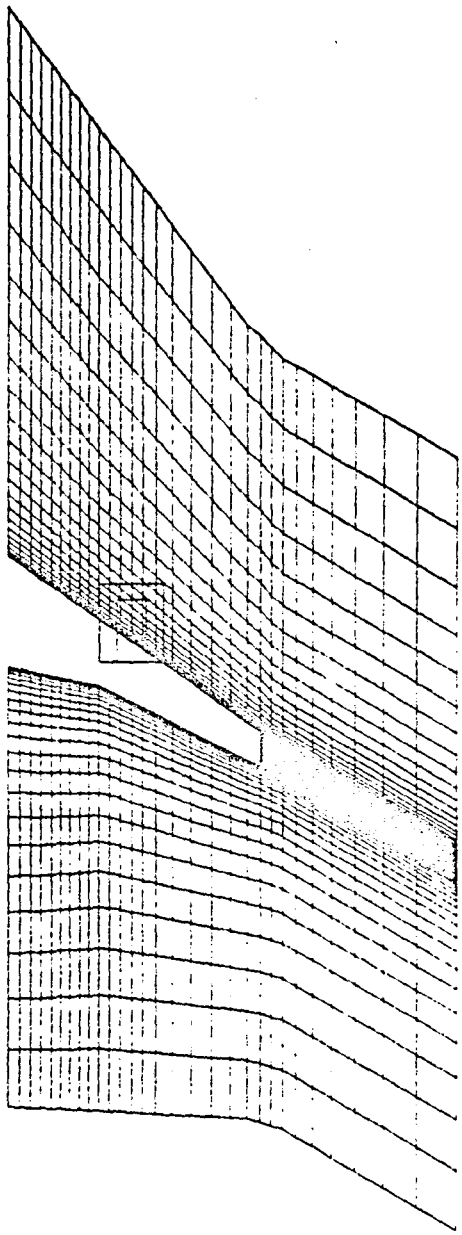


Figure 13. Planform view of Lockheed-Georgia wing/nacelle configuration global grid.

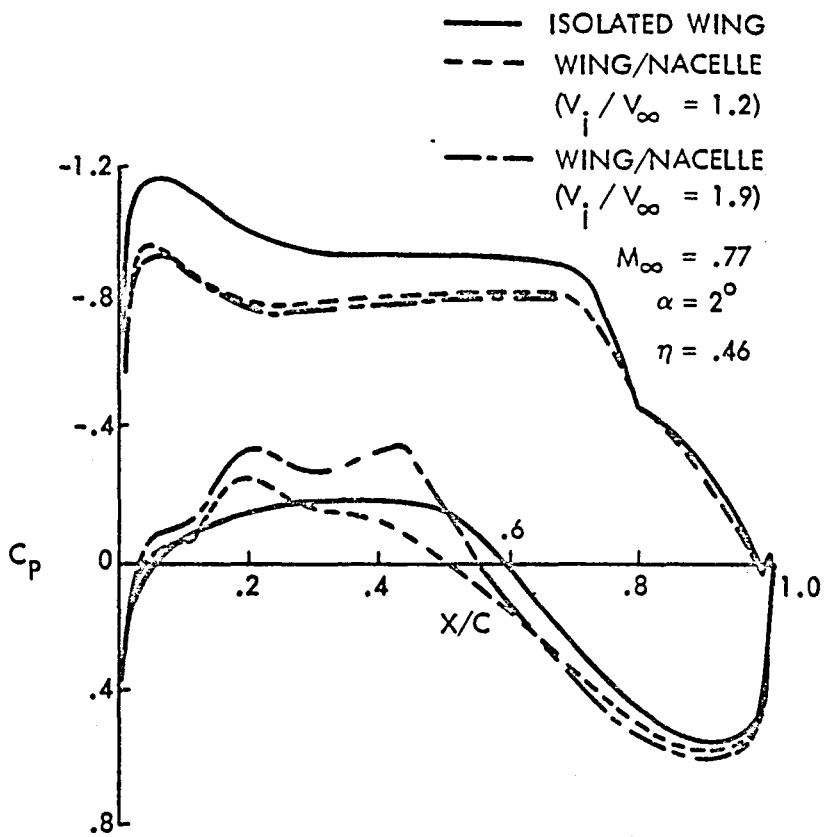


Figure 14. Wing pressure distribution for Lockheed-Georgia wing/nacelle configuration at 46% span.

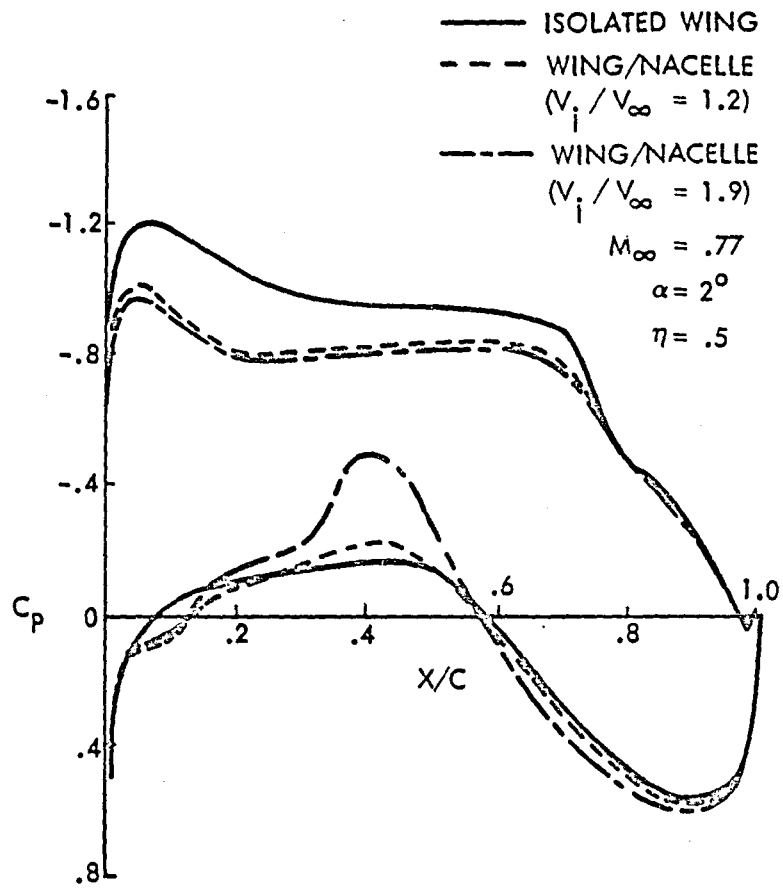


Figure 15. Wing pressure distribution for Lockheed-Georgia wing/nacelle configuration at 50% span.

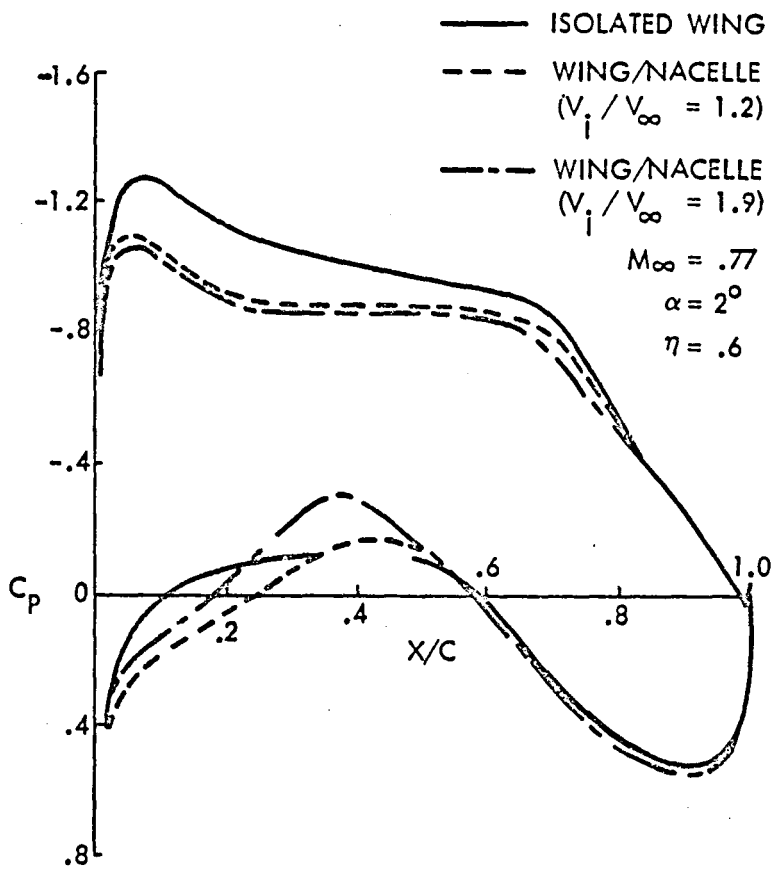


Figure 16. Wing pressure distribution for Lockheed-Georgia wing/nacelle configuration at 60% span.

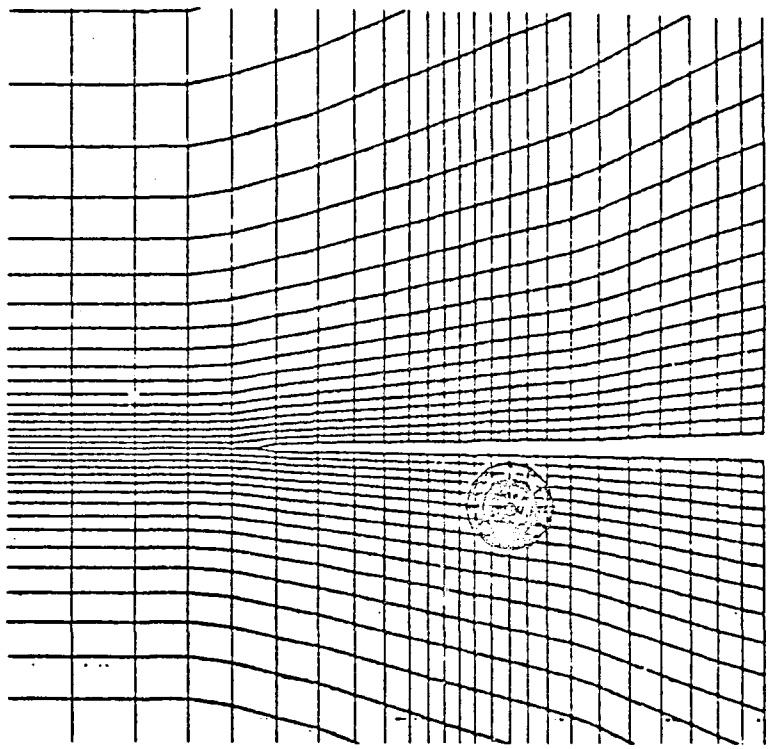


Figure 17. Front view of Lockheed C-5A wing/nacelle configuration global grid.

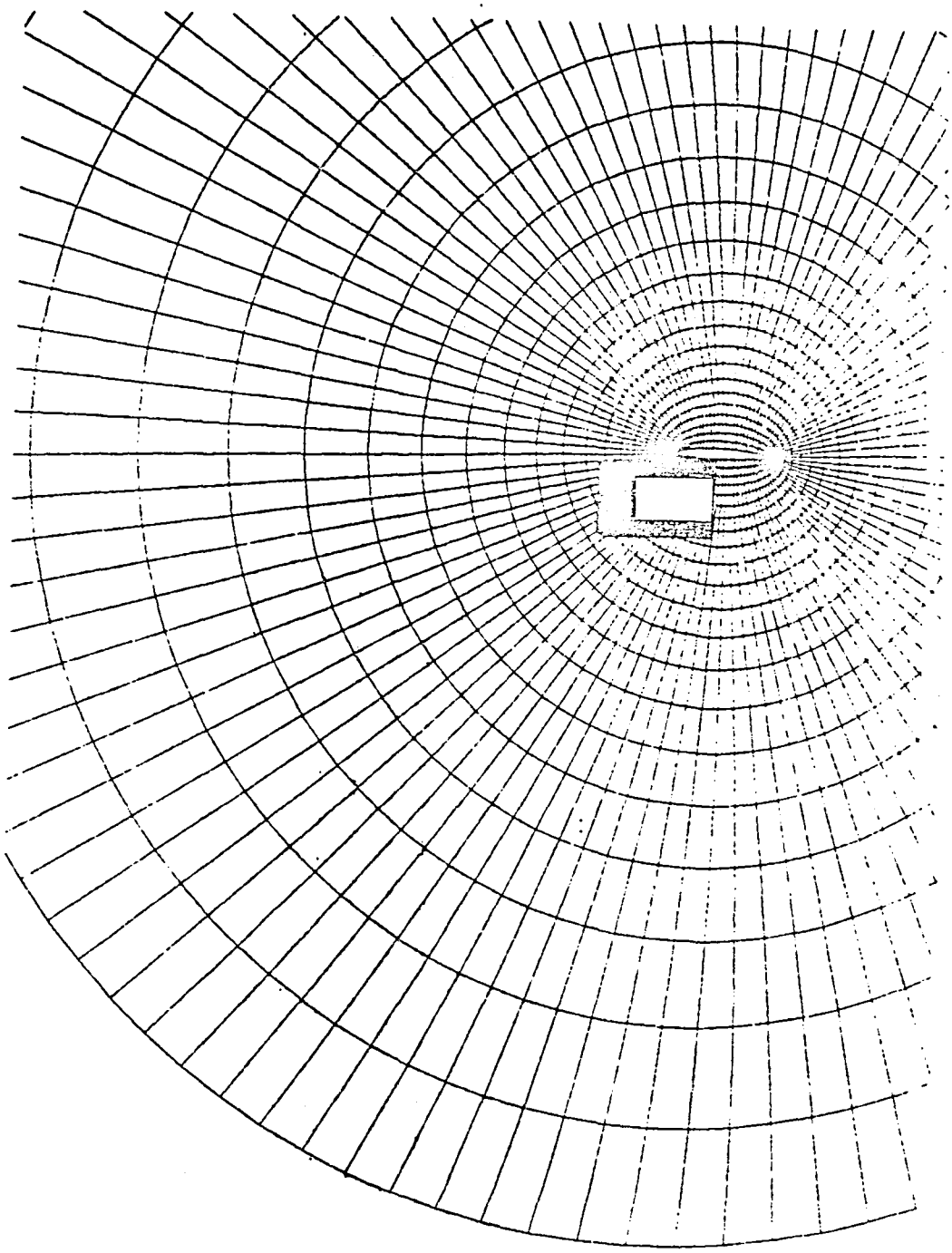


Figure 18. Side view of Lockheed C-5A wing/nacelle configuration global grid.

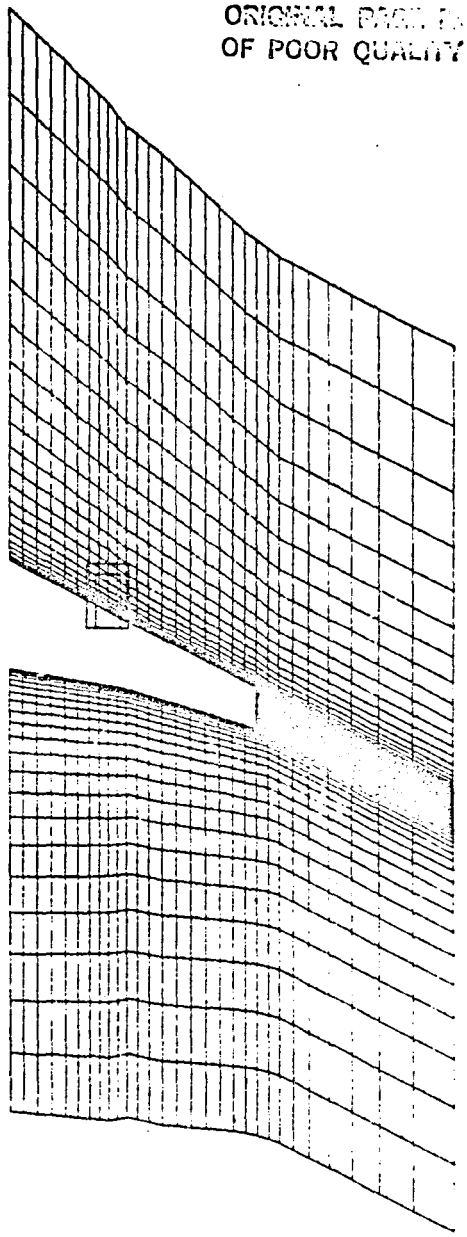


Figure 19. Planform view of Lockheed C-5A wing/nacelle configuration global grid.

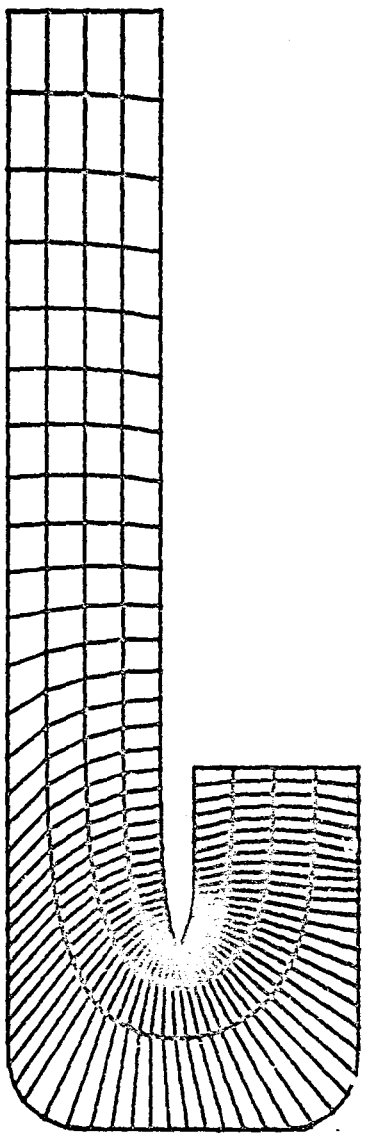


Figure 20. Side view of Lockheed C-5A nacelle and grid used in computation.

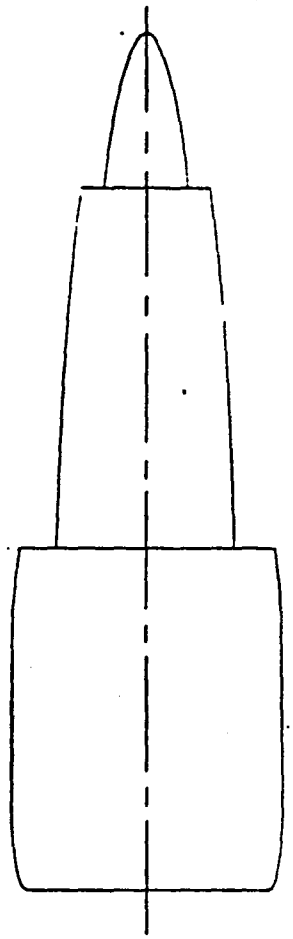


Figure 21. Side view of Lockheed C-5A turbofan nacelle flight hardware.

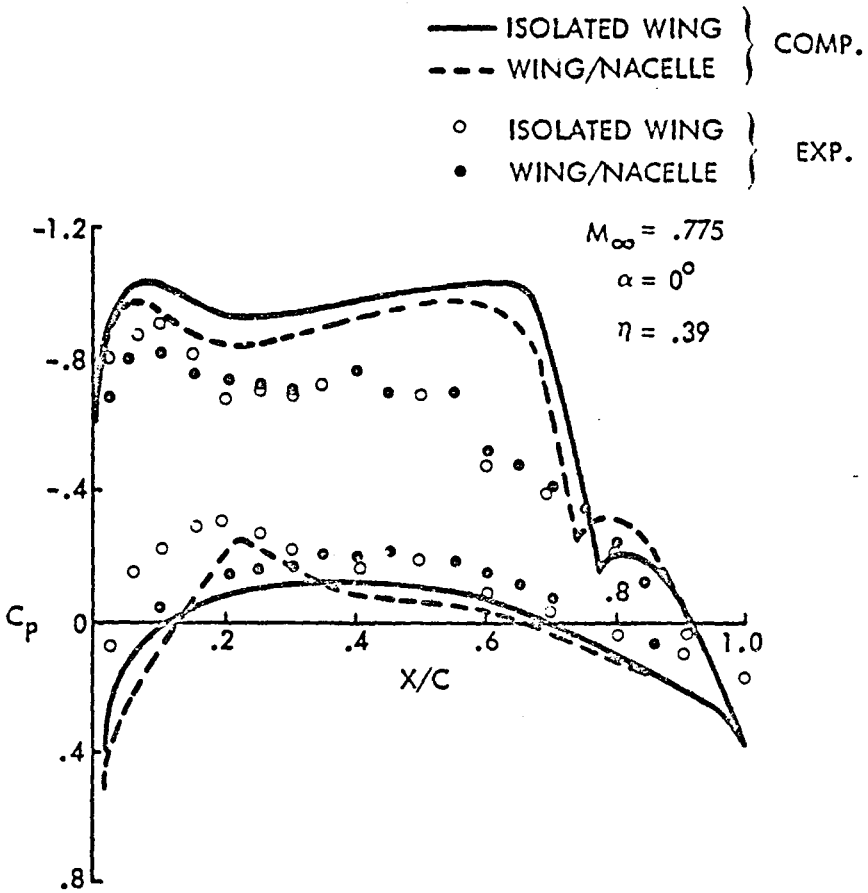


Figure 22. Wing pressure distribution for Lockheed C-5A wing/nacelle configuration at 39% span.

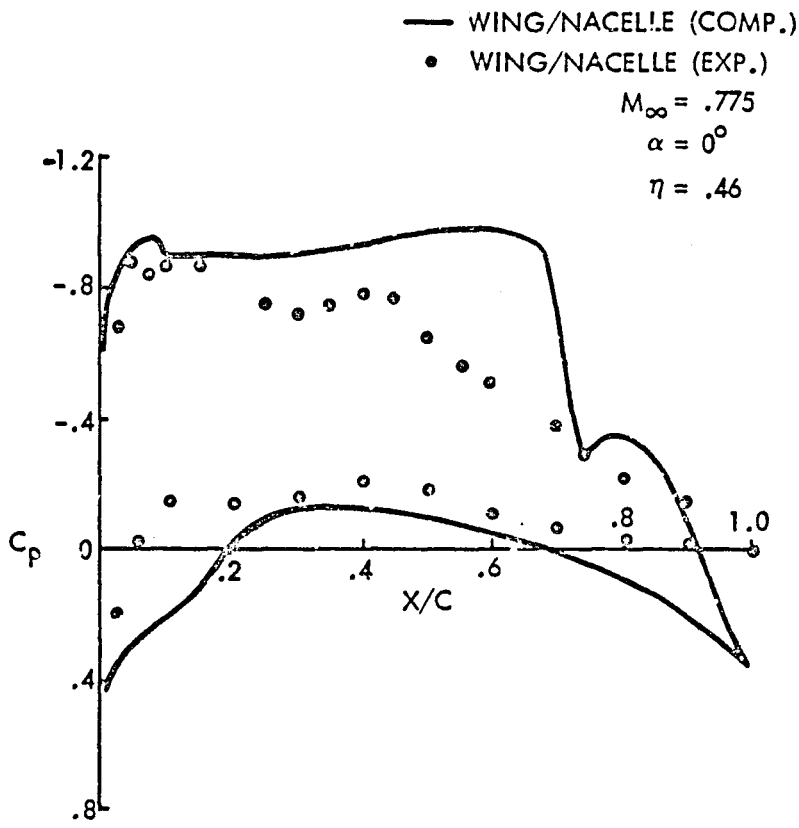


Figure 23. Wing pressure distribution for Lockheed C-5A wing/nacelle configuration at 46% span.

upper surface from approximately the 30% to 70% chord stations. This overprediction is present both in the isolated wing and wing/nacelle computations. The agreement between analysis and experiment for the wing lower surface is substantially better. The analysis overpredicts the lower surface static pressure near the leading edge.

The difference between the results of the analysis and experiment on the wing upper surface are attributed mainly to using conservative differencing without employing a boundary layer correction and to wind tunnel interference effects. The use of conservative differencing generally predicts stronger shocks than nonconservative differencing. However, invoking a boundary layer correction will generally correct the computed shock strength. The angle of attack used in the analysis was the same as that of the experiment. The experimental wing lift coefficient was not available[9]. Consequently, it was not possible to determine the angle of attack to be used in the analysis that would match the measured lift coefficient and then to compare the pressure distributions at the corrected incidence. The difference between the results of the analysis and experiment on the wing lower surface near the leading edge are attributed mainly to the absence of pylon modeling in the analysis.

Figure 24 compares the results of the analysis and experiment for the C-5A configuration at the 60% span station. This case corresponds to a free-stream Mach number of 0.7, a 2 degree angle of attack, an inlet capture ratio of 0.55, and exhaust jet to free-stream velocity ratio of 1.2. Results are presented for an isolated wing and a wing/nacelle configuration. In this case, the agreement between analysis and experiment is encouraging on the wing upper surface for the wing/nacelle configuration. The agreement on the wing lower surface is also good in the region from the wing midchord aft.

An effort was made to include swept pylon effects into the analysis. A pylon of infinitesimal thickness was positioned at the spanwise station containing the nacelle symmetry plane. A flow tangency condition was imposed on the pylon surface enforcing the contravariant velocity V in the η-direction to be zero on both sides of the pylon. This condition was imposed in both the wing and nacelle component algorithms. Difficulties were encountered, however, in achieving converged solutions with the pylon model included. This was felt to be due to the fact that the grid was not aligned with the swept pylon leading edge and to the fact that with wing sweep a spanwise velocity exists which causes the pylon to be loaded thereby requiring the inclusion of a vortex sheet which emanates from the pylon trailing edge. Time constraints in the present investigation did not allow these difficulties to be completely resolved.

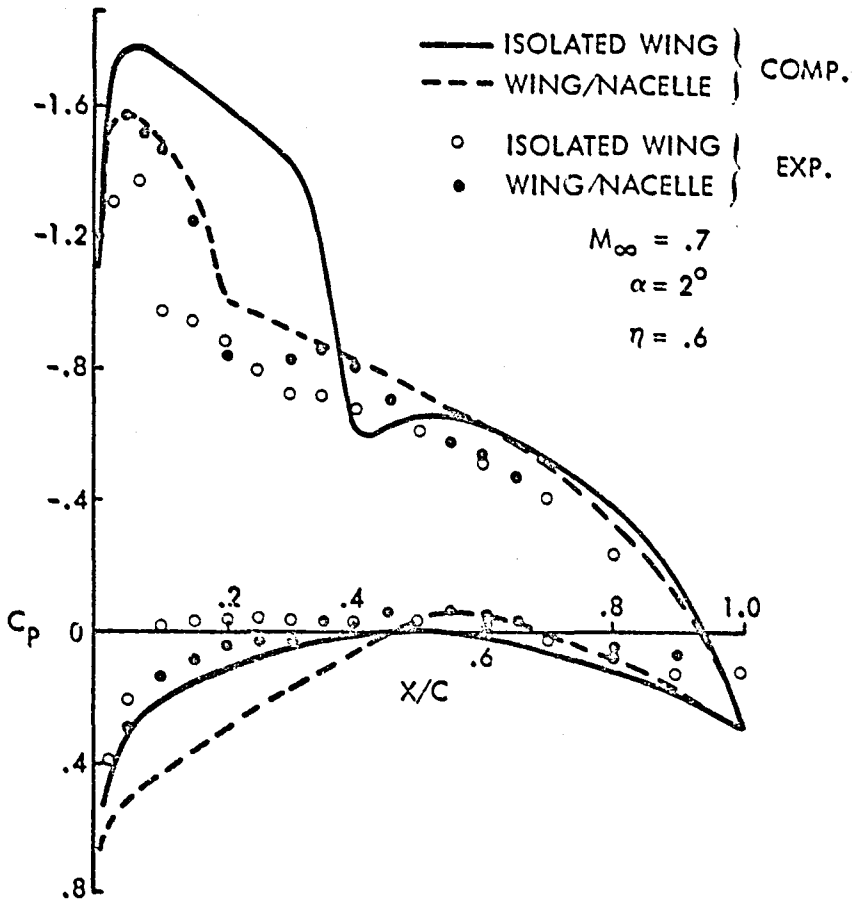


Figure 24. Wing pressure distribution for Lockheed C-5A wing/nacelle configuration at 60% span.

SECTION III

SUBROUTINE DESCRIPTIONS

1. INTRODUCTION

In this section, a brief description is given of the function of each subroutine in GRGEN3 and NGRIDA grid generation Fortran programs, the LK grid interface Fortran program, and the TWN flow solution Fortran program. This information supplements the information available in the form of comment statements within the respective programs.

2. SUBROUTINE DESCRIPTIONS FOR THE GRGEN3 WING GRID GENERATION PROGRAM

GRGEN3. This program routine is the main control routine in the GRGEN3 wing grid generation program. The program execution is initiated by entering NAMELIST GRID3D which defines the parameters needed to determine the three-dimensional body-fitted wing grid. Routine GRGEN3 then calls SUBROUTINE YDIST to establish the grid plane distribution in the spanwise direction. SUBROUTINE GRGEN is then called to determine the grid point distribution for each spanwise station that is used to define the wing geometry. The finished grid coordinate data are then loaded onto disk files TAPE 9 and TAPE 10 for input to the LK grid interface program.

YDIST. This subroutine is used to establish the spanwise grid coordinate distribution. The spanwise grid planes are usually clustered near the wing root and wing tip and near the span location of the nacelle. SUBROUTINE YDIST is called from routine GRGEN3.

GRGEN. This subroutine is used to determine the grid point distribution on a given spanwise station. NAMELIST GRIDIN is first read to enter the parameters required in the ADI relaxation scheme. SUBROUTINE GEMPAC is then called to determine the coordinates of the local wing section. At this stage, SUBROUTINE INNER is called to ascertain the surface grid point distribution by clustering points near the leading and trailing edges of the wing. SUBROUTINE OUTER is then called to determine the outer boundary grid point locations. Finally, SUBROUTINE GRGEN calls SUBROUTINE ADI to determine the interior field point locations by iteratively solving two coupled Laplace equations. SUBROUTINE GRGEN is called from routine GRGEN3.

GEMPAC. This subroutine determines the wing section coordinates for each span station that is used to define the wing geometry. Seven basic geometry definitions are contained within SUBROUTINE GEMPAC and are controlled by the input parameter IOPT. SUBROUTINE GEMPAC is called from SUBROUTINE GRGEN.

INNER. This subroutine is used to determine the grid point distribution on the wing surface for each spanwise station at which the wing geometry

is defined. The surface grid point clustering is controlled by the BINN input parameter. SUBROUTINE INNER is called from SUBROUTINE GRGEN.

OUTER. This subroutine is used to determine the grid point distribution on the circular outer computational boundary for each spanwise station at which the wing geometry is defined. The circular outer boundary is centered at the wing section leading edge. SUBROUTINE OUTER is called from SUBROUTINE GRGEN.

GOUT. This subroutine is used to print the computed grid point distribution. The amount of printout is controlled by the input parameters NOUT5, IINCR, and JINCR.

TRIB. This subroutine is used to solve a tridiagonal system of simultaneous linear equations with fixed boundary conditions. SUBROUTINE TRIB is called from SUBROUTINE ADI.

TRIP. This subroutine is used to solve a tridiagonal system of simultaneous linear equations with periodic boundary conditions. SUBROUTINE TRIP is called from SUBROUTINE ADI.

ADI. This subroutine is used to determine the final interior grid point distribution by solving two coupled Laplace equations using an alternating-direction-implicit iteration algorithm. SUBROUTINE ADI is called from SUBROUTINE GRGEN.

CSPLIN. This subroutine is used to perform cubic spline interpolation to obtain the final distribution of surface grid points at each spanwise station where the wing geometry is defined. SUBROUTINE CSPLIN is called from SUBROUTINE INNER.

INTERP. This subroutine is used to perform linear interpolation for the grid point coordinates between the spanwise stations at which the wing geometry is defined. SUBROUTINE INTERP is called from routine GRGEN3.

3. SUBROUTINE DESCRIPTIONS FOR THE NGRIDA NACELLE GRID GENERATION PROGRAM

NGRIDA. This program routine is the main control routine in the NGRIDA grid generation program. Routine NGRIDA first calls SUBROUTINE NINPUT for data input, parameter initialization, and printing of preliminary information. SUBROUTINE GENER is then called to control the actual numerical generation of the surface-fitted computational grid.

NINPUT. This subroutine is used to enter the input data, perform parameter initialization, and print preliminary information. The input data are entered by specification of the four NAMELISTS LIST1, LIST2, LIST3, and LIST4. All input parameters have specified default values, and a default axisymmetric nacelle/inlet geometry data set has been loaded onto DATA statements within the program. At the end of SUBROUTINE NINPUT, the input namelist data are printed to initiate the computer output. SUBROUTINE NINPUT is called from routine NGRIDA.

CUBICS. This subroutine determines the coefficients for a set of cubic spline polynomials which are used to interpolate for the surface geometry given the axial positions and corresponding surface radii for N tabular data points. The use of cubic splines ensures continuity of surface radius, slope, and curvature for the interpolated geometry. The spline coefficients are found in terms of the surface curvature values at each of the supplied tabular data points. A sparse system of simultaneous linear equations, which determines the curvature values, is solved by calling SUBROUTINE GELG. SUBROUTINE CUBICS is called from SUBROUTINE GENER.

GENER. This subroutine contains the main control logic for numerically generating the nacelle curvilinear computational grid. The initial step in GENER is to generate the cubic spline polynomial coefficients for both the external and internal nacelle surfaces by calling SUBROUTINE CUBICS. The cubic spline polynomials are used to interpolate for the surface geometry. At this stage, the arc length measured from the nacelle hilite is determined at each of the user supplied tabular data points that define the geometry. From a set of user-supplied stretching factors which effect clustering around the hilite, the arc length distributions for the final surface grid point locations are determined. From the surface grid point arc length distributions, the axial and radial coordinates of the final surface grid points are found by interpolation using the cubic spline polynomials. At this stage, the outer computational boundary grid point distribution is determined by one of two options. The first option determines the outer boundary point locations by using an angular distribution where the boundary points are located on rays spaced in equal angular increments around the nacelle hilite. The second option determines the outer boundary point locations by using an arc length distribution with the spacing between points being controlled by user-supplied stretching parameters. At this stage, a file is written for the GRAPE algorithm which determines the interior field point locations by iteratively solving two coupled Poisson equations. After the GRAPE algorithm has been executed, conversion of the data format and plotting of the finished grid is performed by calling SUBROUTINE CONVERT. At this stage, a reflection of the grid point coordinates is performed to determine the point distributions on the remaining meridional planes. Finally, certain transition point indices are determined and the grid point distribution is printed and also stored on a disk file for input to the flow solution program. SUBROUTINE GENER is called from routine NGRIDA.

CONVERT. This subroutine is used to convert the output from the GRAPE algorithm into a format which is acceptable by the TWN flow solution algorithm. The GRAPE-generated grid point coordinates are entered through a binary read of TAPE 7. The corners on the leading edge of the grid are then rounded and point reordering is performed to make the grid generator output consistent with the input format required by the TWN program. Translation and scaling of the grid are then performed. If desired, plotting of the finished grid is performed. The plotting routines use the ISSCO-DISSPLA[10] Fortran library of plot functions. If this subroutine library is not available, then the

plotting section of the code will have to be modified to be compatible with the user's particular installation. SUBROUTINE CONVERT is called from SUBROUTINE GENER.

GELG. This subroutine is used for solving a system of simultaneous linear equations. The system is solved using Gaussian elimination with complete pivoting. This subroutine is IBM Library SUBROUTINE GELG. SUBROUTINE GELG is called from SUBROUTINE CUBICS.

The remaining subroutines in the NGRIDA grid generation program comprise the GRAPE algorithm. Cursory descriptions of the functions of these subroutines are presented herein. Detailed subroutine descriptions are given in Reference 6.

GRAPE. This subroutine is the main control routine in the GRAPE algorithm. SUBROUTINE GRAPE is called from SUBROUTINE GENER after the boundary grid point locations have been determined. SUBROUTINE GRAPE determines the interior field point locations by calling in turn SUBROUTINES INPUT, INCHK, INNER, OUTER, SOLVE, and OUTPUT. The grid point locations are transmitted back to SUBROUTINE GENER through use of a scratch disk file.

CKSMTH. This GRAPE algorithm subroutine is used to check for smoothness in an input array by passing a parabola through the three nearest neighbors of a point and then observing the difference between the actual value and that interpolated from the parabola.

CSPLIN. This GRAPE algorithm subroutine is used to perform cubic spline curve fitting.

IC. This GRAPE algorithm subroutine is used to initialize the grid coordinate and forcing function arrays prior to starting the iterative relaxation procedure to determine the interior grid point locations.

INCHK. This GRAPE algorithm subroutine is used to test for errors on selected input parameters for the GRAPE program.

INNER. This GRAPE algorithm subroutine is used to locate grid points on the inner computational boundary.

INPUT. This GRAPE algorithm subroutine is used to enter the input data that is required for execution of the GRAPE program.

INTERP. This GRAPE algorithm subroutine is used to interpolate from the coarse grid solution to obtain initial conditions for the fine grid solution.

OUTER. This GRAPE algorithm subroutine is used to locate grid points on the outer computational boundary.

OUTPUT. This GRAPE algorithm subroutine is used to print the finished grid point coordinates.

RELAX. This GRAPE algorithm subroutine is used to solve the governing Pois-

son equations using a successive-line-over-relaxation iteration scheme.

SOLVE. This GRAPE algorithm subroutine is used to control the coarse and fine grid solutions of the governing Poisson equations.

TRIB. This GRAPE algorithm subroutine is used to solve a tridiagonal system of simultaneous linear equations with fixed boundary conditions.

TRIP. This GRAPE algorithm subroutine is used to solve a tridiagonal system of simultaneous linear equations with periodic boundary conditions.

4. SUBROUTINE DESCRIPTIONS FOR THE LK GRID INTERFACE PROGRAM

LK. This program routine is the main control routine in the LK grid interface program. Routine LK enters the wing grid coordinate data by reading disk files TAPE 9 and TAPE 10 along with the nacelle grid coordinate data by reading disk file TAPE 14. From the parameters entered on NAMELIST NACEL, routine LK then scales the nacelle coordinate data to be consistent with the wing reference length. The nacelle grid coordinates are then transformed into the wing coordinate system. At this stage, the indices of the overlap region inner and outer boundaries are ascertained. SUBROUTINES FIX and MIX are then called to determine the interpolation grid point stencils for the points on the overlap boundaries. Finally, the interpolation point stencils and other variables required for interfacing the two grids are stored on the disk files TAPE 19 and TAPE 44.

MIX. This subroutine is used to search the wing grid to ascertain those points that are nearest to a given point on the outer boundary of the nacelle grid. SUBROUTINE MIX also performs a search of the nacelle grid to ascertain those points that are nearest to a given point in the wing grid that is on the overlap region inner boundary. SUBROUTINE MIX is called from routine LK.

FIX. This subroutine ascertains the interpolation point stencils used to determine the coefficients in a linear trivariate interpolation polynomial that is employed to transfer property information between the component grids. The interpolation polynomial s based on a first-order accurate trivariate Taylor series expansion. SUBROUTINE FIX is called from routine LK.

5. SUBROUTINE DESCRIPTIONS FOR THE TWN FLOW SOLUTION PROGRAM

TWN. This program routine is the main control routine in the TWN flow analysis program. Routine TWN calls SUBROUTINES INPUT and INIT to enter the input data and to perform the parameter initialization which are required for the computation of the wing component flow field. SUBROUTINES NINPUT and NINIT are then called to enter the input data and to perform the parameter initialization which are required for the computation of the nacelle component flow field. The wing and nacelle flow field solutions are then computed in a cyclic iterative manner until overall convergence is achieved. SUBROUTINES INTW and INTC are called in the course of the computation to

interchange property data from the wing and nacelle component grids.

INPUT. This subroutine is used to enter the input data and perform initialization of selected parameters which are used in the computation of the wing component flow field. The input data are entered by specification of the NAMELIST FLOWIN. The coordinate data for the wing component grid are entered by reading disk files TAPE 9 and TAPE 10. SUBROUTINE INPUT is called from routine TWN.

NINPUT. This subroutine is used to enter the input data and perform initialization of selected parameters which are used in the computation of the nacelle component flow field. The input data are entered by specification of the NAMELIST FLOWC. SUBROUTINE NINPUT is called from routine TWN.

INIT. This subroutine is used to initialize selected parameters employed in the computation of the wing component flow field. SUBROUTINE INIT reads disk file TAPE 19 to enter the grid overlap region coordinate data that are generated by the LK grid interface program. SUBROUTINE INIT then calls SUBROUTINES METRIX and WRITMS to compute and store, respectively, the metric parameters used in the wing grid mapping from physical space to computational space. Finally, the wing potential field is initialized using free-stream velocity components. SUBROUTINE INIT is called from routine TWN.

INTW. This subroutine is used to interpolate the velocity potential values from the nacelle flow field solution which are then used as boundary conditions on the overlap region inner boundary for an ensuing wing flow field calculation. SUBROUTINE INTW is called KCX times after every ITMAX iterations performed in the nacelle grid. SUBROUTINE INTW is called from routine TWN.

INTC. This subroutine is used to interpolate the velocity potential values from the wing flow field solution which are then used as boundary conditions on the overlap region outer boundary for an ensuing nacelle flow field calculation. SUBROUTINE INTC is called KCX times after every MAXIT iterations performed in the wing grid. SUBROUTINE INTC is called from routine TWN.

NINIT. This subroutine is used to initialize selected parameters employed in the computation of the nacelle component flow field. SUBROUTINE NINIT first computes functions of the specific heat ratio and binomial expansion coefficients used in the computation of the physical density. SUBROUTINE METRIC is then called to determine the metric parameters used in the nacelle grid mapping from physical space to computational space. At this stage, the free-stream velocity magnitude and Cartesian velocity components are computed along with the appropriate outflow boundary condition parameters. Finally, the potential field is initialized. The external flow field points are initialized using free-stream velocity components, whereas the internal flow field points are initialized using velocities based on a local Mach number which itself is found from the area ratio and compressor face Mach number using one-dimensional gas dynamic formulae. SUBROUTINE NINIT is called from routine TWN.

SOLVE. This subroutine is the main wing flow field integration control routine and is used to apply the AF2 approximate factorization scheme to determine the inviscid wing flow solution. The AF2 scheme is applied in an iterative manner with the iteration sequence being terminated by either achieving convergence or by reaching a maximum permissible number of iterations. Per iteration, a three-step process is used to obtain the potential function correction. The first two steps calculate intermediate potential function corrections arising from solving systems of factorized difference equations in the ξ-wraparound and η-spanwise directions. The third step calculates the actual potential correction by solving a system of factorized difference equations for the ζ-radial coordinate. Steps 1 and 2 require sweeping the computational mesh from the body surface outwards. The third step sweeps from the outer computational boundary towards the body surface. SUBROUTINE SOLVE calls SUBROUTINES RO, ROCO, and RESID for calculation of the physical density coefficients, modified density coefficients, and mass residuals, respectively. SUBROUTINES TRIB and TRIP are called in the course of applying the AF2 scheme to solve systems of tridiagonal simultaneous linear equations with fixed and periodic boundary conditions, respectively. SUBROUTINE SOLVE is called by routine TWN.

METRIX. This subroutine is used to compute the metric parameters for all points in the wing component grid. The metric coefficients define the three-dimensional grid mapping from physical space to computational space. The metric coefficients are computed using fourth-order accurate finite-difference formulae. SUBROUTINE METRIX is called from SUBROUTINE INIT.

METCO. This subroutine is used to compute and store the metric quantities that are used in computing the mass residuals at the wing surface. SUBROUTINE METCO is called from SUBROUTINE SOLVE.

RO. This subroutine is used to calculate the physical density coefficients at the ξ-half points in the wing computational mesh. From the current potential function field, the derivatives of the velocity potential in the ξ, η, and ζ directions are computed using second-order accurate finite-difference formulae. The potential function derivatives are then used in a form of the isentropic energy equation to determine the density. At the body surface, the flow tangency condition is used in computing the density. At the external computational boundary, the density is assumed to be equal to the free-stream density. SUBROUTINE RO is called from SUBROUTINE SOLVE.

LININT. This subroutine is used to perform linear interpolation. SUBROUTINE LININT is called from SUBROUTINE RO.

ROCO. This subroutine is used to calculate the modified density coefficients at the ξ, η, and ζ half points in the wing computational mesh. The modified density coefficients are computed by forming the appropriate averages of the physical density coefficients calculated by SUBROUTINE RO. If the flow is locally supersonic, then upwinding is effected by forming a weighted average of the physical density at the

point in consideration and the density at a point in the upwind direction along the appropriate curvilinear coordinate. Upwinding is always performed along the ξ-wraparound coordinate. Upwinding in the η-spanwise direction and in the ζ-normal direction are user-defeatable options. SUBROUTINE ROCO is called from SUBROUTINE SOLVE.

RESID. This subroutine is used to apply the full-potential equation residual operator to the current velocity potential field to obtain the mass residuals at all points in the wing computational mesh except for those points on the outer computational boundary. The mass residual at a given point is computed by forming the sum of the finite-difference derivatives of the mass fluxes in the ξ, η, and ζ coordinate directions. Central difference expressions are used at interior field points while special procedures are used at the body surface. SUBROUTINE RESID employs the modified density coefficients calculated by SUBROUTINE ROCO. SUBROUTINE RESID is called from SUBROUTINE SOLVE.

OUTPUT. This subroutine is used to print the wing solution convergence history and to print the surface pressure coefficient distribution for each wing section. If desired, the density and Mach number contours may also be output. SUBROUTINES FORCE and TFORCE are called to compute the sectional and total force and moment coefficients, respectively. SUBROUTINE OUTPUT is called from SUBROUTINE SOLVE.

GETMET. This subroutine is used to retrieve the metric parameters from disk storage by in turn calling SUBROUTINE READMS. SUBROUTINE GETMET is called from SUBROUTINE SOLVE.

FORCE. This subroutine is used to compute the wing sectional lift, wave-drag, and moment coefficients by integrating the surface pressure distribution for a given spanwise station. SUBROUTINE FORCE is called from SUBROUTINE OUTPUT.

TFORCE. This subroutine is used to compute the wing total lift and wave-drag coefficients. The moment coefficient at each spanwise station is also computed using the root quarter chord station as the datum about which moments are computed. For this subroutine to execute properly, the relations JSKIP=1 and JCPEND>NJIM must be satisfied. SUBROUTINE TFORCE is called from SUBROUTINE OUTPUT.

READEC. This subroutine is used to transfer data from extended core storage to central memory (replaces CDC SUBROUTINE READEC).

WRITEC. This subroutine is used to transfer data from central memory to extended core storage (replaces CDC SUBROUTINE WRITEC).

READMS. This subroutine is used to transfer data from a mass storage device to central memory (replaces CDC SUBROUTINE READMS).

WRITMS. This subroutine is used to transfer data from central memory to a mass storage device (replaces CDC SUBROUTINE WRITMS).

SAVE. This subroutine is used to save the flow field solution on disk file

TAPE 12 for restarting the program.

TRIB. This subroutine is used to solve a system of simultaneous linear equations with a tridiagonal coefficient matrix and fixed boundary conditions. This routine is used in the AF2 iteration algorithm to obtain a potential function intermediate correction. SUBROUTINE TRIB is called from SUBROUTINES NSOLVE and SOLVE.

TRIP. This subroutine is used to solve a system of simultaneous linear equations with a tridiagonal coefficient matrix and periodic boundary conditions. This routine is used in the AF2 iteration algorithm to obtain a potential function intermediate correction. SUBROUTINE TRIP is called from SUBROUTINES NSOLVE and SOLVE.

SECANT. This subroutine is used to determine the nacelle local internal flow Mach number using one-dimensional gas dynamic formulae which express mass continuity for the captured streamtube. A reference area and Mach number are supplied, and from the local flow area the local Mach number is determined. A secant iteration scheme is used to numerically determine the zero of the governing implicit relation. SUBROUTINE SECANT is called from SUBROUTINE NINIT and determines the compressor face Mach number from the user-supplied capture ratio. It also determines the local Mach number from the compressor face Mach number and the ratio between the local and compressor face areas.

METRIC. This subroutine is used to compute the metric parameters for all points in the nacelle component grid. The metric coefficients define the three-dimensional grid mapping from physical space to computational space. The metric coefficients are computed using second-order accurate finite-difference formulae. The physical grid point locations are entered through a binary read of a disk file. SUBROUTINE METRIC is called from SUBROUTINE NINIT.

NRO. This subroutine is used to calculate the physical density coefficients at the ξ-half points and the ξ-end points in the nacelle computational mesh. From the current potential function field, the derivatives of the velocity potential in the ξ, η, and ζ directions are computed using second-order accurate finite-difference formulae. The potential function derivatives are then used in a form of the isentropic energy equation to determine the density. At the body surface, the flow tangency condition is used in computing the density. At the internal outflow surface, the density is assumed known from the prescribed boundary conditions. At the outer computational boundary, the density is calculated using potential values interpolated from the wing solution. SUBROUTINE NRO is called from SUBROUTINES NSOLVE and NOUTPUT.

NROCO. This subroutine is used to calculate the modified density coefficients at the ξ, η, and ζ half points in the nacelle computational mesh. The modified density coefficients are computed by forming the appropriate averages of the physical density coefficients calculated by SUBROUTINE NRO. If the flow is locally supersonic, then upwinding is effected by forming a weighted average of the physical density at the point in consideration and the density at a point in the upwind

direction along the appropriate curvilinear coordinate. Upwinding is always performed along the ξ-wraparound coordinate. Upwinding is not performed in the η - circumferential direction but can be in the ζ - radial direction. SUBROUTINE NROCO is called from SUBROUTINE NSOLVE

NRESID. This subroutine is used to apply the full-potential equation residual operator to the current velocity potential field to obtain the mass residuals at all points in the nacelle computational mesh except for those points on the outer computational boundary. The mass residual at a given point is computed by forming the sum of the finite-difference derivatives of the mass fluxes in the ξ, η, and ζ coordinate directions. Central difference expressions are used at interior field points while special procedures are used at the body surface and at the internal outflow surface. SUBROUTINE NRESID employs the modified density coefficients calculated by SUBROUTINE NROCO. SUBROUTINE NRESID is called from SUBROUTINE NSOLVE.

NSOLVE. This subroutine is the main nacelle flow field integration control routine and is used to apply the AF2 approximate factorization scheme to determine the inviscid nacelle flow solution. The AF2 scheme is applied in an iterative manner with the iteration sequence being terminated by either achieving convergence or by reaching a maximum permissible number of iterations. Per iteration, a three-step process is used to obtain the potential function correction. The first two steps calculate intermediate potential function corrections arising from solving systems of factorized difference equations in the ξ-wraparound and η-circumferential directions. The third step calculates the actual potential correction by solving a system of factorized difference equations for the ζ-radial coordinate. Steps 1 and 2 require sweeping the computational mesh from the body surface outwards. The third step sweeps from the outer computational boundary towards the body surface. SUBROUTINE NSOLVE calls SUBROUTINES NRO, NROCO, and NRESID for calculation of the physical density coefficients, modified density coefficients, and mass residuals, respectively. SUBROUTINES TRIB and TRIP are called in the course of applying the AF2 scheme to solve systems of tridiagonal simultaneous linear equations with fixed and periodic boundary conditions, respectively. SUBROUTINE NOUTPUT is called to print the computed surface solution. SUBROUTINE NSOLVE is called by routine TWN.

NOUTPUT. This subroutine is used to print the computed nacelle surface solution when convergence has been attained or when the maximum permissible number of iterations has been reached. SUBROUTINE NRO is applied at the body surface to calculate the physical density coefficients at mesh mid-points on the surface. From the computed density distribution, the surface Mach number and pressure coefficient distributions are determined. SUBROUTINE NOUTPUT is called from SUBROUTINE NSOLVE.

METSET. This subroutine is used to transfer property data between central memory and a mass storage device.

SECTION IV

INPUT PARAMETERS AND OUTPUT INTERPRETATION

1. INTRODUCTION

In this section, the input parameters are defined for the GRGEN3 and NGRIDA grid generation programs, the LK grid interface program, and the TWN wing/nacelle flow solution program. Where applicable, both the default value and the typical value of each input parameter are given. Discussions on interpretation of the output, PARAMETER statement specification, and the file usage for each program are also provided.

2. GRGEN3 WING GRID GENERATION PROGRAM INPUT PARAMETERS

The input data required for execution of the GRGEN3 wing grid generation program are entered by namelist input and, in some cases, formatted read statements. In all cases, the two NAMELISTS GRID3D and GRIDIN are entered. Most of the input parameters have assigned default values.

2.1 NAMELIST GRID3D

The input parameters entered in NAMELIST GRID3D specify the number of mesh points in the wing grid, control the wing grid mesh spacing, and identify the location of the engine nacelle.

NI — A positive integer denoting the number of ξ-wraparound stations in the wing computational grid, as shown in Figure 25. The ξ-stations are numbered in a counterclockwise direction starting at the upper surface trailing edge. The specified value of NI must be odd. A default value of 101 is specified for NI.

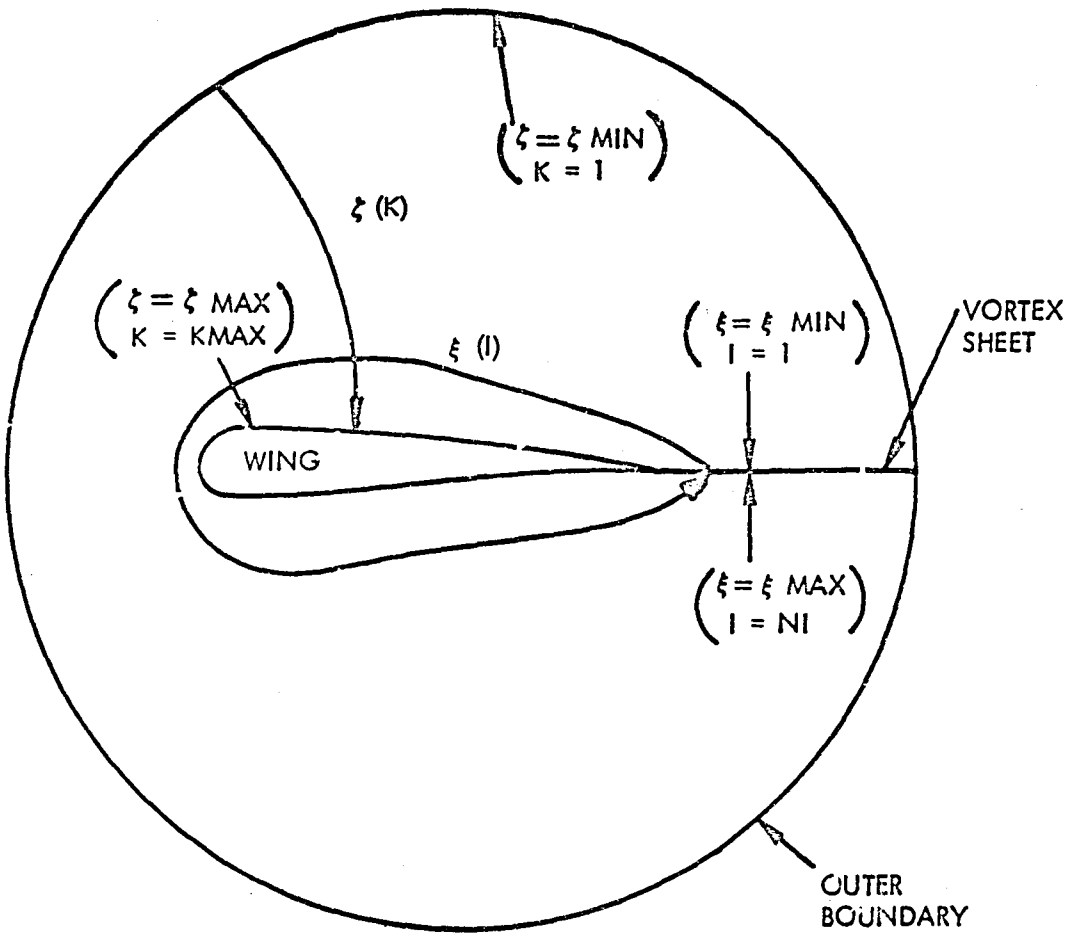
NJ — A positive integer denoting the number of η-spanwise stations in the wing computational grid, as shown in Figure 25. The η-stations are numbered starting at the wing root and increasing in the outward direction. A default value of 26 is specified for NJ.

NK — A positive integer denoting the number of ζ-radial stations in the wing computational grid, as shown in Figure 25. The ζ-stations are numbered starting at the outer boundary and increasing in the inward direction. A default value of 18 is specified for NK.

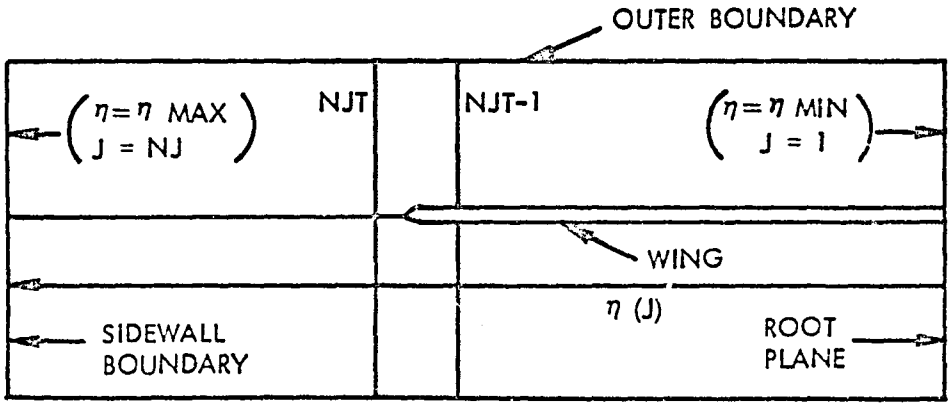
NJT — A positive integer denoting the index of the first η-station that is off the wing tip. A default value of 19 is specified for NJT.

NJC — A positive integer denoting the index of the η-station that contains the nacelle plane of symmetry. A default value of 9 is specified for NJC.

SFACT — A real variable denoting the stretching factor used in controlling the η-spanwise grid spacing beyond the wing tip. A default value of 1.2 is specified for SFACT.



(a) SPAN ' 'ATION GRID TOPOLOGY



(b) GRID TOPOLOGY ALONG SPANWISE DIRECTION

Figure 25. Topology for wing component grid.

SMOOTH A real variable denoting the factor used in smoothing the grid spanwise spacing near the wing tip. A default value of 1.0 is specified for SMOOTH.

SWPANG A real variable denoting the wing sweep angle, in degrees, along the spanwise direction. A default value of 0.0 is specified for SWPANG.

TWANG A real variable denoting the wing twist angle, in degrees, along the spanwise direction. A default value of 0.0 is specified for TWANG.

CEL A real variable denoting the distance between the nacelle plane of symmetry and the wing root station. A default value of 3.0 is specified for CEL.

TIP A real variable denoting the spanwise location of the wing tip (the tip is assumed to lie halfway between the span stations NJT and NJT-1). A default value of 6.0 is specified for TIP.

DYROOT A real variable denoting the spanwise distance between the wing second span station and the root station. If DYROOT = 0.0 is entered, then DYROOT is internally reset to (TIP/(NJT-1.5)). A default value of 0.0 is specified for DYROOT.

DYTIP A real variable denoting the spanwise wing grid spacing near the wing tip. If DYTIP = 0.0 is entered, then DYTIP is internally reset to (TIP/(NJT-1.5)). A default value of 0.0 is specified for DYTIP.

DYNCL A real variable which controls the wing grid spanwise spacing near the nacelle plane of symmetry. A default value of 0.1 is specified for DYNCL.

2.2 NAMELIST GRIDIN

The input parameters entered in NAMELIST GRIDIN specify the wing geometry and control the relaxation solution algorithm used to generate the wing grid.

RADMAX A positive real variable denoting the radius of the circular outer boundary of the wing computational grid. A default value of 5.0 is specified for RADMAX.

NSTATN An integer denoting whether one or more airfoil sections are to be used in defining the wing geometry. If NSTATN = 0, one airfoil section will be used. If NSTATN > 0, more than one airfoil section will be used. A default value of 0 is specified for NSTATN.

IOPT An integer denoting the wing section geometry option. IOPT specifies the following geometry options:

IOPT	Option
1	NACA 00XX type airfoil
2	circular arc airfoil (parabolic leading edge blunting)
3	circular cylinder

4	read in arbitrary airfoil geometry
5	circular arc airfoil (square root leading edge blunting)
6	ONERA M6 airfoil
7	KORN airfoil

A default value of 1 is specified for IOPT.

TMAX A positive real variable denoting the airfoil maximum thickness as a fraction of the chord length. TMAX must be entered only if IOPT = 1, 2, or 5. A default value of 0.12 is specified for TMAX.

XC A real variable denoting the leading edge bluntness parameter. XC must be entered only if IOPT = 2 or 5, and must be in the range of $0.0 < XC < 0.4$. A default value of 0.2 is specified for XC.

XCN A real variable denoting the x-coordinate, in terms of fraction of the chord, of the point to which the origin is shifted. A default value of 0.0 is specified for XCN.

BINN A real variable denoting the stretching parameter which controls the grid point distribution on the wing surface. BINN controls the surface grid point distribution as follows:

BINN	Option
0.0	the grid points are distributed evenly in terms of the arc length
-1.0	the grid point distribution is established from a data statement which was developed by using a circle plane conformal mapping around a NACA 0012 airfoil
> 1.0	the grid point distribution is established from an exponential clustering formula with a BINN value closer to 1.0 clustering more points at the leading and trailing edges of the wing sections

A value of 1.05 is suggested for BINN. A default value of -1.0 is specified for BINN.

ERGRID A real variable denoting the convergence tolerence used in the wing grid generation solution algorithm. A default value of 0.001 is specified for ERGRID.

MAXIT An integer denoting the maximum number of iterations permissible to attain convergence in the wing grid generation algorithm. A default value of 200 is specified for MAXIT.

AKGRID A real variable denoting the largest value in the alpha acceleration

parameter sequence used in the wing grid generation relaxation algorithm. A default value of 0.01 is specified for AHGRID.

ALGRID A real variable denoting the smallest value in the alpha acceleration parameter sequence used in the wing grid generation relaxation algorithm. A default value of 0.00006 is specified for ALGRID.

KGRID An integer denoting the starting element in the alpha sequence used for the wing grid generation. A default value of 0 is specified for KGRID.

MGRID An integer denoting the number of elements in the alpha sequence used for the wing grid generation. A default value of 8 is specified for MGRID.

OMEG A positive real variable denoting the relaxation parameter used in the wing grid generation algorithm. A default value of 2.0 is specified for OMEG.

NOUT5 An integer which controls the amount of grid output information. If NOUT5=1, the grid point coordinates are printed. If NOUT5=0, the grid point coordinates are not printed. A default value of 0 is specified for NOUT5.

NOUT6 An integer denoting the increment in iteration count at which the grid point coordinates are to be printed during the course of the relaxation solution. A default value of 201 is specified for NOUT6. (If convergence is attained before NOUT6 iterations, the NOUT5 print option is invoked.)

IINCR An integer which controls the grid solution output frequency in the ξ - direction. The wing grid coordinates will be printed with an IINCR increment. A default value of 3 is specified for IINCR.

JINCR An integer which controls the grid solution output frequency in the η - direction. The wing grid coordinates will be printed with a JINCR increment. A default value of 6 is specified for JINCR.

2.3 INPUT DATA FOR WING MULTIPLE STATION DEFINITION OPTION

The following input parameters are entered by formatted read statements only for the IOPT=4 wing multiple station definition option. No default values are specified for the following parameters.

CARD 1 [A-Format]:

CHAR Title card (any alphanumeric characters).

CARD 2 [Format (7F10.6)]:

YPLANE A real variable denoting the local wing section span position (y coordinate) normalized by the root chord.

CHORD A real variable denoting the local wing section chord normalized by

the root chord.

THICK A real variable denoting the local wing section thickness in terms of fraction of the root chord (if the wing has the same section but with varying thickness).

ALPHA A real variable denoting the local wing section twist angle, in degrees.

ALAMB A real variable denoting the local wing section sweep angle, in degrees.

XCEN A real variable denoting the x-coordinate, in terms of the fraction of the chord, to which the origin is shifted.

RADIUS A real variable denoting the radius of the circular outer boundary. For highly tapered wings, the values of radius should be decreased in the spanwise direction.

CARD 3 [Format (I5)]:

NPTS An integer denoting the number of points in the wraparound direction used to define the local wing section. The points are numbered from the trailing edge counterclockwise and do not include the redundant trailing edge point.

CARD 4 [Format (2F10.5)]:

XB A real variable denoting the wing section chordwise coordinate, in terms of fraction of the local chord.

YB A real variable denoting the wing section normal coordinate, in terms of fraction of the local chord.

Note there are NPTS CARD 4 cards for each wing defining station. CARDS 1-4 must be supplied for each of the NSTATN defining stations.

2.4 PARAMETER STATEMENT SPECIFICATION

The GRGEN3 wing grid generation program uses variable array dimensions. The respective array sizes are fixed by specification of the following dimension parameters on the PARAMETER statement:

Parameter		Allowed Values
IZZ	$\geq$	NI
IZ2	$\geq$	IZZ+2
JZ	$\geq$	NJ
KZ	$\geq$	NK

The array dimensions will be fixed at the time of compilation.

2.5 FILE USAGE

The following files are used by the GRGEN3 wing grid generation code.

File No.	Usage
TAPE 5	input file
TAPE 6	printed output file
TAPES 9,10	output files containing the wing grid coordinates

3. GRGEN3 PROGRAM OUTPUT INTERPRETATION

The initial portion of the computer printout for the GRGEN3 wing grid generation program consists of the supplied input parameters. The remaining output parameters are defined below.

J	index of the spanwise wing sections
Y	final spanwise coordinate distribution
I	index of points in the wraparound direction (starting from the upper trailing edge point)
XB,YB	wing section coordinates (the coordinates are printed twice, first the input coordinates are printed and then the final coordinates after interpolation and clustering are printed)
S2	arc length normalized by section chord
SX	first difference of the arc length

The following parameters are associated with the convergence history for the ADI numerical grid generation scheme.

N	iteration number in the numerical grid generation scheme
CXMAX	maximum correction (chordwise direction)
I,J	position of the maximum correction CXMAX
CYMAX	maximum correction (normal direction)
I,J	position of the maximum correction CYMAX
RXMAX	maximum residual (chordwise direction)
I,J	position of the maximum residual RXMAX
RYMAX	maximum residual (normal direction)
I,J	position of the maximum residual RYMAX
ALPHA	the acceleration parameter used for the ADI scheme

4. NGRIDA NACELLE GRID GENERATION PROGRAM INPUT PARAMETERS

The input data required for execution of the NGRIDA nacelle grid generation computer program are entered by namelist input. In all cases, the four NAMELISTS LIST1, LIST2, LIST3, and LIST4 are entered. All of the input parameters have assigned default values. A suitable grid can often be generated by retaining many of the input parameters at their default values.

4.1 NAMELIST LIST1

The input parameters specified in NAMELIST LIST1 specify the number of mesh points in the nacelle grid, outer boundary shape, grid point distribution stretching factors, and print and plot options.

The following three parameters specify the number of grid points.

IMAX — A positive integer denoting the number of ξ-wraparound stations in the nacelle computational grid, as shown in Figure 26. The specified value of IMAX must be less than or equal to 140, and should be of the form (3n+1) if grid sequencing is to be used in the GRAPE algorithm, where n is an integer. A default value of 67 is specified for IMAX.

JMAX — A positive integer denoting the number of circumferential stations in the nacelle computational grid, as shown in Figure 27. Stations J=1 and J=JMAX are coincident. The remaining meridional stations are spaced in equal angular increments around the body. The specified value of JMAX must be odd. A default value of 13 is specified for JMAX.

KMAX — A positive integer denoting the number of ζ-radial stations in the nacelle computational grid, as shown in Figure 26. The specified value of KMAX must be less than or equal to 60, and should be of the form (3n+1) if grid sequencing is to be used in the GRAPE algorithm, where n is an integer. A default value of 13 is specified for KMAX.

The next three parameters specify the nacelle grid outer computational boundary.

XLEFT — A real variable denoting the x-coordinate value of the left-hand side of the nacelle grid outer computational boundary, as shown in Figure 28. XLEFT is specified in the units of the input geometry, and typically is negative. A default value of -30.0 is specified for XLEFT.

RADOUT — A positive real variable denoting the radius of the nacelle grid outer computational boundary, as shown in Figure 28. RADOUT is specified in the units of the input geometry. A default value of 50.0 is specified for RADOUT.

DELTA — A positive real variable denoting the offset distance between the centerline computational boundary and the x-axis, as shown in Figure 28. DELTA is specified in the units of the input geometry and must not be zero. A value of approximately 5% of the compressor face radius is recommended for DELTA. A default value of 0.5 is specified for DELTA.

The next four parameters control the grid point distribution on the nacelle surface.

IOUTER — A positive integer denoting the number of grid points desired on the outer portion of the nacelle surface ranging from the hilite point to the external outflow surface, as shown in Figure 28. A default value of 38 is specified for IOUTER.

IINNER — A positive integer denoting the number of grid points desired on the

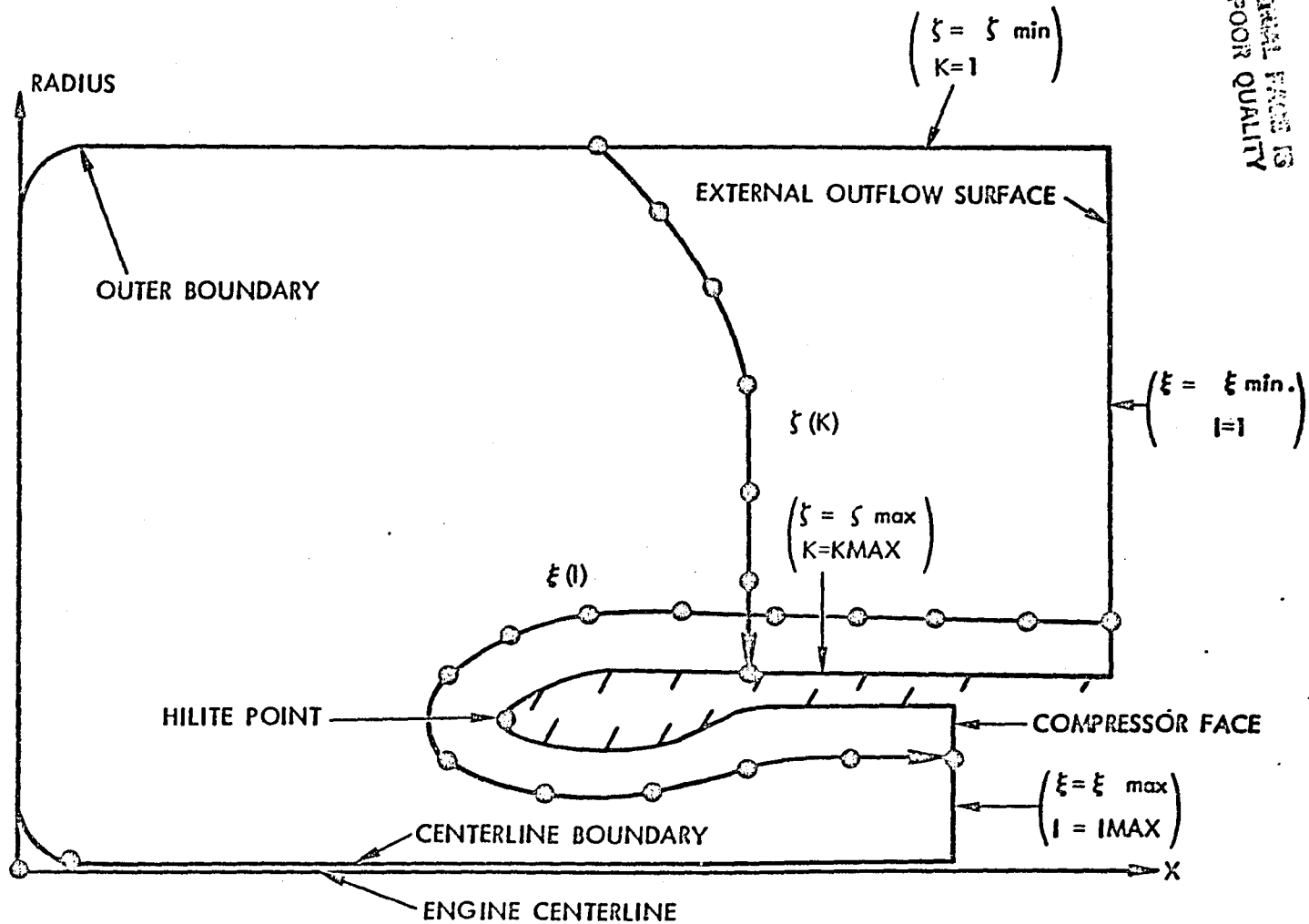


Figure 26. Topology for nacelle component grid (meridional plane).

Figure 27. Topology for nacelle component grid (view along axis of inlet).

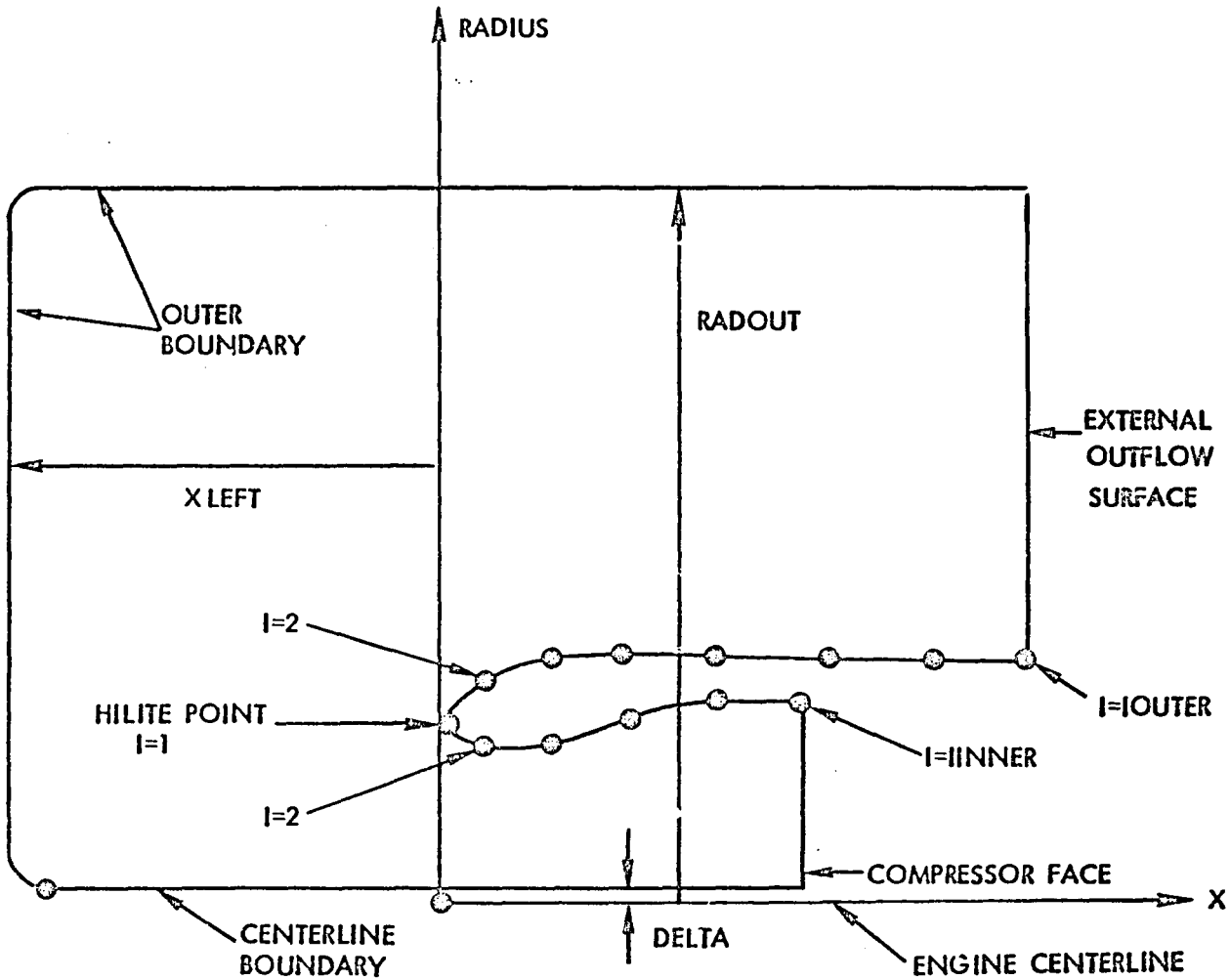


Figure 28. Specification of nacelle grid outer boundary and point distribution on surface.

inner portion of the nacelle surface ranging from the hilite point to the compressor face outflow surface, as shown in Figure 28. The specified values for IOUTER and IINNER must satisfy the relation

$$IMAX = IINNER + IOUTER - 1 \qquad (29)$$

A default value of 30 is specified for IINNER.

ALPHAO A positive real variable denoting the stretching factor used in determining the grid point distribution on the nacelle external surface. Entering ALPHAO=1.0 produces a point distribution with uniform arc length between the points. Entering ALPHAO > 1.0 clusters points near the nacelle hilite. A default value of 1.1 is specified for ALPHAO.

ALPHAI A positive real variable denoting the stretching factor used in determining the grid point distribution on the nacelle internal surface. Entering ALPHAI=1.0 produces a point distribution with uniform arc length between the points. Entering ALPHAI > 1.0 clusters points near the nacelle hilite. A default value of 1.1 is specified for ALPHAI.

The next five parameters determine the outer boundary grid point distribution. Generally, the outer boundary points near both the external and internal outflow surfaces are found by equating the axial coordinate of a given boundary point to the axial coordinate of the corresponding body point (see Figure 29). The remaining boundary points are found by either using an angular distribution or an arc-length distribution.

IAA A positive integer denoting the wraparound coordinate station up to which the points on the computational boundary near the centerline have their axial coordinates equated to the respective values of the corresponding points on the internal nacelle surface, as illustrated in Figure 29. IAA is specified in terms of the GRAPE algorithm point-ordering scheme in which the wraparound coordinate initiates at the compressor face and terminates at the external outflow surface (see Figure 29). A default value of 12 is specified for IAA.

IDD A positive integer denoting the wraparound coordinate station after which the points on the outer computational boundary have their axial coordinates equated to the respective values of the corresponding points on the nacelle external surface, as shown in Figure 29. IDD is specified in terms of the GRAPE algorithm point-ordering scheme in which the wraparound coordinate initiates at the compressor face and terminates at the external outflow surface (see Figure 29). Note that IDD must satisfy the relation IAA < IDD < IMAX. A default value of 57 is specified for IDD.

KOUTER An integer denoting whether the outer boundary grid points between points IAA and IDD are to be determined by an angular distribution or by an arc-length distribution. If KOUTER=0, the grid points are determined by an angular distribution with the grid points being spaced at equal angular increments around the nacelle hilite. If KOUTER=1, the grid points are determined by an arc-length distri-

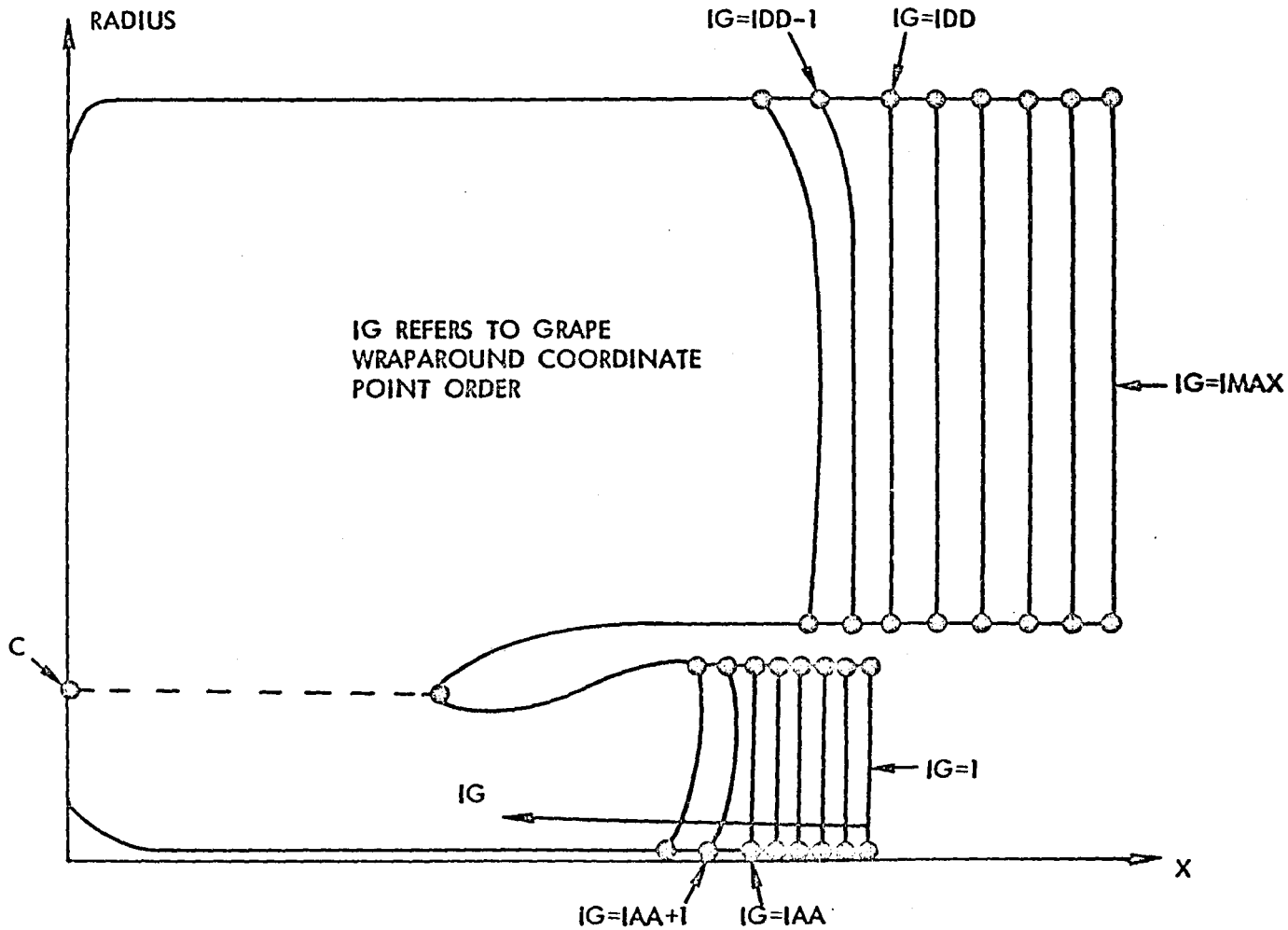


Figure 29. Definition of points IAA and IDD.

bution with respective stretching factors being specified by the user. A default value of 0 is specified for KOUTER.

If KOUTER=1 is specified, then the following two parameters must be entered.

ALPHBO A positive real variable denoting the stretching factor used in determining the nacelle outer boundary grid point distribution for the portion of the boundary between points C and IDD, as shown in Figure 29. Entering ALPHBO=1.0 produces a point distribution with uniform arc length between the points. Entering ALPHBO > 1.0 clusters points near point C. A default value of 1.0 is specified for ALPHBO.

ALPHBI A positive real variable denoting the stretching factor used in determining the nacelle outer boundary grid point distribution for the portion of the outer boundary between points C and IAA, as shown in Figure 29. Entering ALPHBI=1.0 produces a point distribution with uniform arc length between the points. Entering ALPHBI > 1.0 clusters points near point C. A default value of 1.0 is specified for ALPHBI.

The next three parameters control the grid spacing adjacent to the body surface in the direction normal to the surface. Generally, the normal distance DS, as shown in Figure 30, is computed by the relation

$$DS = (RCF - DELTA)/(KMAX - 1) \qquad (30)$$

where RCF is the compressor face radius. For isolated nacelle grids, it is generally preferable to adjust DS near the external outflow surface to make the radial point distribution uniform as shown in Figure 30. This point distribution produces a more favorable cell aspect ratio at the outflow surface, and thereby enhances stability and convergence in the flow solution algorithm.

DSFACT A positive real variable used as a multiplier for DS which is defined by equation (30). A default value of 1.0 is specified for DSFACT.

KRAMP An integer denoting whether or not the DS distribution is to be altered to produce a uniform radial point distribution at the external outflow surface. If KRAMP=1, the DS distribution is altered to produce the uniform radial point distribution. If KRAMP=0, the DS distribution is not altered, and the grid points will be packed towards the nacelle surface. If an isolated nacelle grid is being generated, then the KRAMP=1 option is recommended. A default value of 1 is specified for KRAMP.

IRAMP A positive integer denoting the wraparound station after which the DS distribution is altered to produce a uniform radial point distribution at the external outflow surface, as shown in Figure 30. IRAMP must be entered only if KRAMP=1. IRAMP is specified in terms of the GRAPE algorithm point-ordering scheme in which the wraparound coordinate initiates at the compressor face and terminates at the external outflow surface. Note that IRAMP must satisfy the relation IRAMP < IMAX. A default value of 57 is specified for IRAMP.

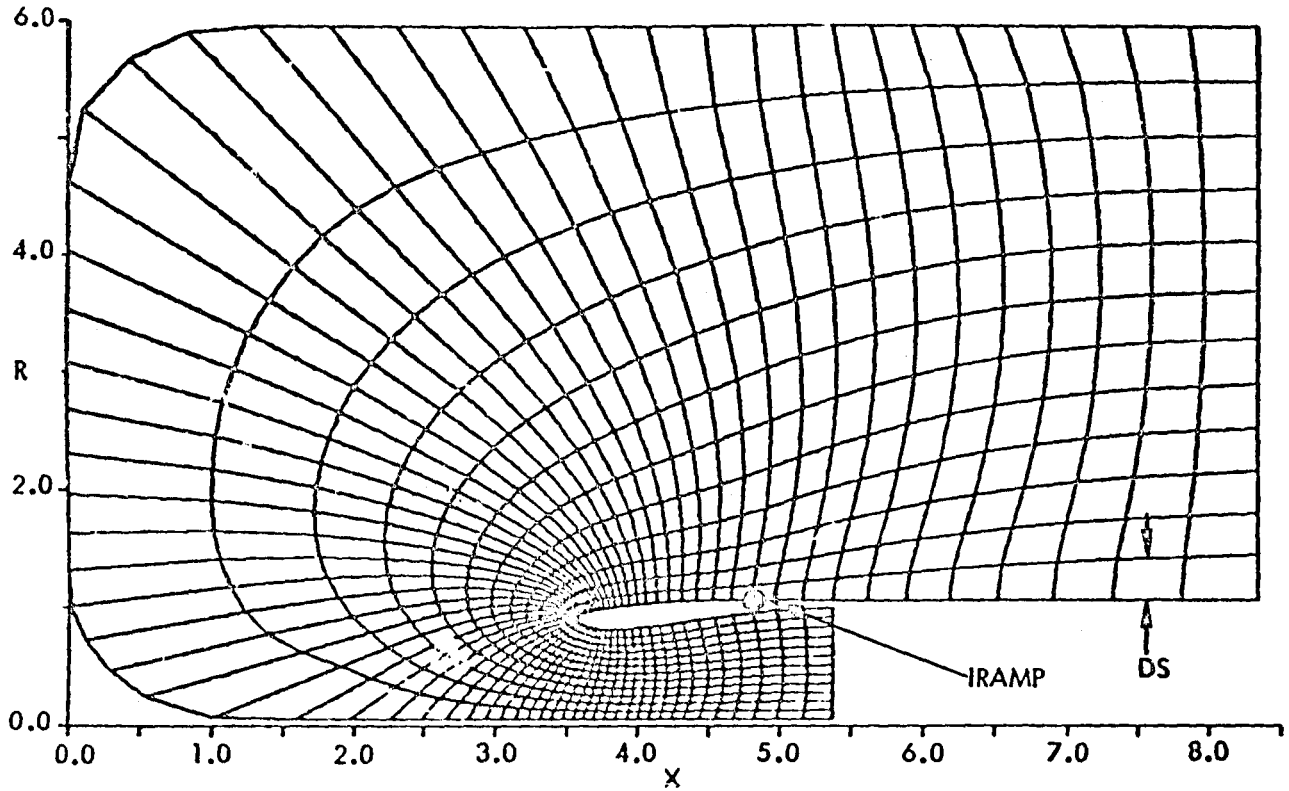


Figure 30. Nacelle grid normal mesh spacing and definition of point IRAMP.

The next three parameters specify the scaling and smoothing function parameters.

SCALE A positive real variable denoting the scaling multiplier used in obtaining the nacelle final grid. Entering SCALE=1.0 produces a grid in terms of the original input units. Setting SCALE=1.0/RCF, where RCF is the compressor face radius, produces a grid with a compressor face radius of 1.0. This option is recommended as it allows for more easily determining the optimum acceleration parameters in the flow solution code. A default value of 0.11904762=1.0/8.4 is specified for SCALE.

KTIME An integer denoting the number of times smoothing polynomials are applied in determining the nacelle grid outer boundary axial coordinate distribution in the vicinity of point IDD, and the DS distribution in the vicinity of point IRAMP if the KRAMP=1 option is specified. If KTIME=0 is entered, no smoothing is performed. The default and recommended value for KTIME is 3.

NDEL A positive integer denoting the number of points to the left of point IDD at which the smoothing is initiated for the nacelle grid outer boundary point axial locations if KTIME > 0 is specified. NDEL must be entered only if KTIME > 0 is specified. A default value of 5 is specified for NDEL.

The next two parameters specify the print options.

KDUMP An integer denoting whether or not detailed nacelle surface and outer boundary coordinate data are to be printed. If KDUMP=0, the data are not printed. If KDUMP=1, the printout is performed. A default value of 1 is specified for KDUMP.

KPRINT A one-dimensional integer array consisting of JMAX elements. Each element of KPRINT specifies whether or not a meridional plane grid point distribution is to be printed. Specifying KPRINT(J)=1 (J=1,...,JMAX) causes the grid point coordinates to be printed for the Jth meridional plane. If KPRINT(J)=0 is entered, the printout is skipped for the Jth plane. A default value of KPRINT(1)=1 is specified, while the remaining elements of KPRINT are specified as 0.

The next two parameters specify the plot options.

KPLOT An integer denoting whether or not the J=1 meridional plane grid is to be plotted. Specifying KPLOT=1 causes the grid to be plotted for the J=1 meridional plane. If KPLOT=0 is entered, the plotting is not performed. A default value of KPLOT=1 is specified.

KDEV An integer denoting the plotting device. KDEV must be specified only if KPLOT=1 is entered. If KDEV=1, the plot device is the VERSATEC electrostatic plotter. If KDEV=2, the plot device is the HP ink plotter. A default value of 1 is specified for KDEV. (Note that the plot device specification can be altered by modifying SUBROUTINE CONVERT).

The next parameter specifies the tape number (disk file) on which the grid coordinate data are written.

ITAPE An integer denoting the tape (file) number on which the grid coordinate data are written. ITAPE can be entered as either 10 or 14. For isolated nacelle computations, ITAPE should be retained at its default value of 10. For use of the nacelle grid in the TWN code, ITAPE = 14 should be entered.

4.2 NAMELIST LIST2

The input parameters entered in NAMELIST LIST2 specify the nacelle/inlet geometry which is described by tabular input. A default axisymmetric nacelle/inlet geometry has been loaded onto DATA statements within the program. The default geometry data are for the Lockheed-Georgia GELAC1 axisymmetric nacelle/inlet configuration.

The following three parameters specify the internal surface contour for axisymmetric nacelles.

NI A positive integer denoting the number of tabular data points used in defining the internal surface contour, as shown in Figure 31. A default value of 60 is specified for NI.

XIN A one-dimensional real variable array consisting of NI elements. XIN(I) (I=1,...,NI) denotes the axial coordinate of the Ith point used in defining the internal surface contour, as shown in Figure 31. Successive elements of XIN must be monotonically increasing. XIN(1) denotes the axial station of the nacelle hilite, and XIN(NI) denotes the compressor face axial station. Default values for XIN(I) (I=1,...,60) are specified.

RIN A one-dimensional real variable array consisting of NI elements. RIN(I) (I=1,...,NI) denotes the radius of the Ith point used in defining the internal surface contour, as shown in Figure 31. RIN(1) denotes the nacelle hilite radius, and RIN(NI) denotes the compressor face radius. Default values for RIN(I) (I=1,...,60) are specified.

The next three parameters specify the external surface contour for axisymmetric nacelles.

NO A positive integer denoting the number of tabular data points used in defining the external surface contour, as shown in Figure 31. A default value of 100 is specified for NO.

XOUT A one-dimensional real variable array consisting of NO elements. XOUT(I) (I=1,...,NO) denotes the axial coordinate of the Ith point used in defining the external surface contour, as shown in Figure 31. The successive elements of XOUT must be monotonically increasing. XOUT(1) denotes the nacelle hilite axial station, and XOUT(NO) denotes the axial station of the external outflow surface. The relation XOUT(1) = XIN(1) must be satisfied. Default values for XOUT(I) (I=1,...,100) are specified.

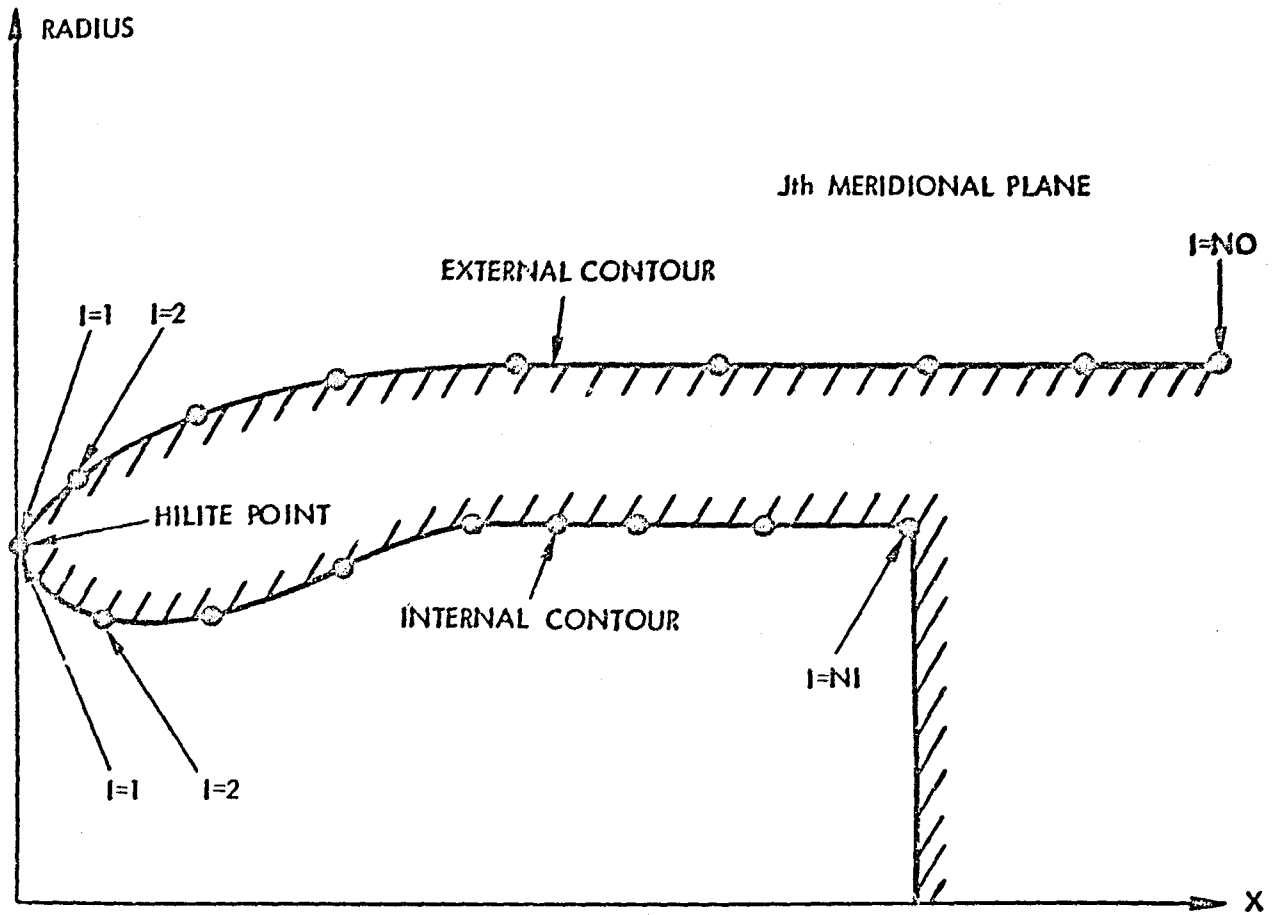


Figure 31. Nacelle geometry specification.

ROUT — A one-dimensional real variable array consisting of NO elements. ROUT(I) (I=1,...,NO) denotes the radius of the Ith point used in defining the external surface contour, as shown in Figure 31. ROUT(1) denotes the nacelle hilite radius, and ROUT(NO) denotes the nacelle radius at the external outflow surface. The relation ROUT(1)=RIN(1) must be satisfied. Default values for ROUT(I) (I=1,...,100) are specified.

4.3 NAMELIST LIST3

The input parameters entered in NAMELIST LIST3 specify the convergence criteria, iteration limits, and print options used in the GRAPE algorithm. These parameters are given only cursory descriptions herein, but are more fully discussed in Reference 6. Generally, the program is executed by retaining all of the parameters in NAMELIST LIST3 at their default values.

NORDA — A one-dimensional integer array consisting of two elements which denote the convergence criteria for the coarse-mesh and the fine-mesh GRAPE solutions. NORDA(1) and NORDA(2) denote the numbers of order of magnitude by which the maximum correction is to be reduced for the coarse-mesh and fine-mesh solutions, respectively. Default values of NORDA(1)=6 and NORDA(2)=3 are specified.

MAXITA — A one-dimensional integer array consisting of two elements which denote the iteration limits for the coarse-mesh and fine-mesh GRAPE solutions. MAXITA(1) and MAXITA(2) denote the iteration limits for the coarse-mesh and fine-mesh solutions, respectively. The MAXITA array is used in conjunction with the NORDA array. Default values of MAXITA(1)=400 and MAXITA(2)=200 are specified.

JPRT — An integer denoting the print option to be used in the GRAPE algorithm. If JPRT < 0, no printing is performed except for error warning messages. If JPRT=1, a detailed printout will be performed. A default value of -1 is specified for JPRT.

4.4 NAMELIST LIST4

The input parameters entered in NAMELIST LIST4 specify the grid control functions and convergence parameters used in the GRAPE algorithm. Again, only cursory descriptions are provided herein, with detailed definitions being available in Reference 6. Generally, the program is executed by retaining all of the parameters in NAMELIST LIST4 at their default values.

AAAI, BBBI — Positive real variables which control the enforcement of grid orthogonality and normal mesh spacing at the nacelle surface. Small values (i.e., 0.2) cause these effects to be propagated far into the field, whereas larger values (i.e., 0.6) cause these effects to diminish more rapidly. A default value of 0.6 is specified for both AAAI and BBBI.

CCCI, DDDI — Positive real variables which control the enforcement of grid orthogonality and normal mesh spacing at the outer boundary. Small values (i.e., 0.2) cause these effects to be propagated far into the field, whereas larger values (i.e., 0.6) cause these effects to

diminish more rapidly. A default value of 0.6 is specified for both CCCI and DDDI.

OMEGA — A positive real variable used in the GRAPE successive-line-over-relaxation scheme to obtain the grid point distribution. OMEGA must be in the range of 0.0 < OMEGA < 2.0. A default value of 1.3 is specified for OMEGA.

OMEGP, OMEGQ — Real variables used as relaxation parameters in obtaining the body surface forcing functions that are used in the Poisson equations. The effects of controlling grid spacing and orthogonality at the nacelle surface can be eliminated by entering OMEGP = OMEGQ = 0.0. OMEGP and OMEGQ must be in the range of 0.0 to 2.0. A default value of 0.1 is specified for both OMEGP and OMEGQ.

OMEGR, OMEGS — Real variables used as relaxation parameters in obtaining the outer boundary forcing functions that are used in the Poisson equations. The effects of controlling grid spacing and orthogonality at the outer boundary can be eliminated by entering OMEGR = OMEGS = 0.0. OMEGR and OMEGS must be in the range of 0.0 to 2.0. A default value of 0.0 is specified for both OMEGR and OMEGS.

PLIM, QLIM, RLIM, SLIM — Real variables used as limitation factors in solving for the Poisson equation forcing functions. The range for each of these parameters is 0.0 to 100.0. A default value of 0.5 is specified for PLIM, QLIM, RLIM, and SLIM.

DSOBI — A positive real variable denoting the normal distance to be imposed between the nacelle radial stations KMAX and KMAX-1. DSOBI has units of the input geometry. A default value of 0.2 is specified for DSOBI.

4.5 PARAMETER STATEMENT SPECIFICATION

The NGRIDA nacelle grid generation program uses variable array dimensions. The respective array sizes are fixed by specification of the following dimension parameters on the PARAMETER statement:

Parameter		Allowed Values
NX	≥	IMAX
NY	=	1
NYG	≥	JMAX
NZ	≥	KMAX
NXO	≥	NO
NXI	≥	NI
NXON	≥	IOUTER
NXIN	≥	IINNER
NXO2	≥	NO*NO
NXI2	≥	NI*NI

The array dimensions will be fixed at the time of compilation.

4.6 FILE USAGE

The following files are used by the NGRIDA nacelle grid generation code.

File No.	Usage
TAPE 5	input file
TAPE 6	printed output file
TAPE 10 or TAPE 14	output files containing the nacelle grid coordinates
TAPES 1,4,7	scratch files (should be deleted after computation)

5. NGRIDA PROGRAM OUTPUT INTERPRETATION

The initial portion of the computer printout for the NGRIDA program consists of the NAMELIST input data. Then if the KDUMP=1 option is specified, detailed nacelle surface and outer boundary coordinate data are listed. After this, the GRAPE algorithm output is printed. The finished grid point coordinates are then printed for selected meridional planes as specified by the KPRINT array. Finally, certain index parameters are printed.

The output parameters for the GRAPE algorithm are discussed in Reference 6. The remaining output parameters are defined below.

I	ξ-wraparound point index
J	η-circumferential point index
K	ζ-radial point index
X	x-coordinate of point in finished grid, (scaled units)
Y	y-coordinate of point in finished grid, (scaled units) *
Z	z-coordinate of point in finished grid, (scaled units) *
XOUT	x-coordinate of input point on external surface, (original units)
ROUT	radius of input point on external surface, (original units)
XIN	x-coordinate of input point on internal surface, (original units)
RIN	radius of input point on internal surface, (original units)
ARC-LENGTH	arc length measured from hilite point, (original units)
XBO	x-coordinate of redistributed point on external surface, (original units)
RBO	radius of redistributed point on external surface, (original units)
XBI	x-coordinate of redistributed point on internal surface, (original units)
RBI	radius of redistributed point on internal surface, (original units)
XSURF	surface point x-coordinate used as input for GRAPE code, (original units)

* (Note - the y and z axes are along the wingspan for the wing and nacelle algorithms, respectively; the z and y axes are normal to the wing for the wing and nacelle algorithms, respectively)

RSURF	surface point radius used as input for GRAPE code, (original units)
XBOUND	outer boundary point x-coordinate used as input for GRAPE code, (original units)
RBOUND	outer boundary point radius used as input for GRAPE code, (original units)

6. LK GRID INTERFACE PROGRAM INPUT PARAMETERS

The input data required for execution of the LK grid interface program are entered by namelist input. In all cases, the single NAMELIST NACEL must be entered.

6.1 NAMELIST NACEL

The input parameters entered in NAMELIST NACEL specify the location of the nacelle with respect to the wing.

XR — A real variable denoting the distance, in terms of the fraction of the local wing section chord, between the wing leading edge and the nacelle hilite, as shown in Figure 32. A default value of 0.7 is specified for XR.

ZT — A real variable denoting the distance, in terms of the fraction of the local wing section chord, between the top of the nacelle and the point on the wing leading edge, as shown in Figure 32. A default value of 0.4 is specified for ZT.

SCALE — A real variable denoting the multiplying factor used to scale the nacelle and its grid. If the nacelle and wing reference lengths are the same, then SCALE=1.0. A default value of 0.22 is specified for SCALE.

6.2 PARAMETER STATEMENT SPECIFICATION

The LK grid interface program uses variable array dimensions. The respective array sizes are fixed by specification of the following dimension parameters on the PARAMETER statement.

Parameter		Allowed Values (see documentation for)
IZZ	$\geq$	NI (GRGEN3)
IZ2	=	IZZ*2
JZ	$\geq$	NJ (GRGEN3)
KZ	$\geq$	NK (GRGEN3)
IZJZ	=	IZZ*JZ
TSSZZZ	=	IZZ*JZ*KZ*6
IZT	$\geq$	estimated number of points in the grid overlap region
IZB	$\geq$	estimated number of points in the exhaust jet region at the nacelle aft station
NX	$\geq$	IMAX (NGRIDA)

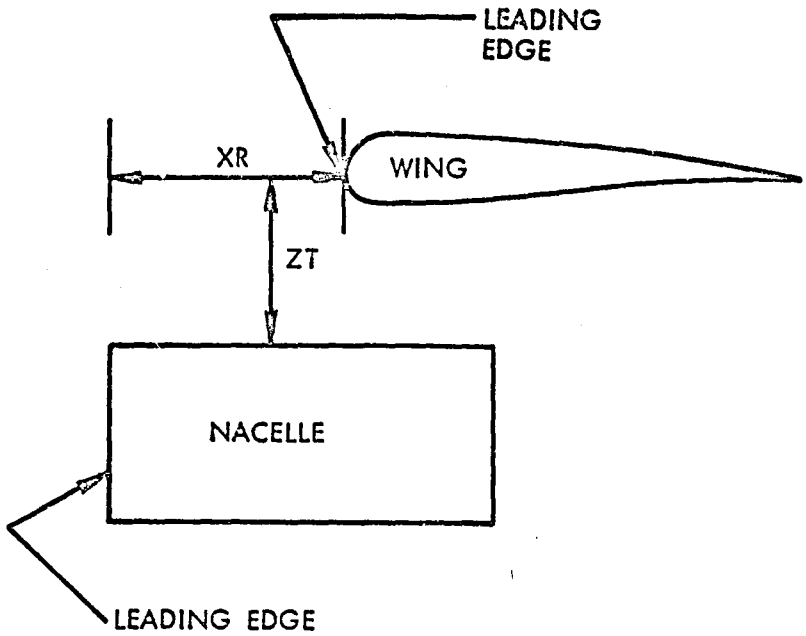


Figure 32. Definition of nacelle position parameters XR and ZT.

NY	$\geq$	JMAX (NGRIDA)
NZ	$\geq$	KMAX (NGRIDA)
NXP	=	NX+1
TSS	=	NX*NY*NZ*7

The array dimensions will be fixed at the time of compilation.

6.3 FILE USAGE

The following files are used by the LK grid interface program.

File No.	Usage
TAPE 5	input file
TAPE 6	printed output file
TAPES 9,10	input files from the GRGEN3 wing grid generation code containing the wing grid coordinates
TAPE 14	input file from the NGRIDA nacelle grid generation code containing the nacelle grid coordinates
TAPES 19,44	output files from the LK grid interface program to be used as input to the TWN code

7. LK PROGRAM OUTPUT INTERPRETATION

The initial portion of the computer printout for the LK grid interface program consists of the NAMELIST input data. The remaining output parameters are defined below.

XR	the horizontal distance from the nacelle hilite to the wing leading edge (expressed in the wing coordinate system)
XL	the horizontal distance from the nacelle aft station to the wing leading edge (expressed in the wing coordinate system)
ZT	the vertical distance from the nacelle hilite (top meridional plane) to the wing leading edge (expressed in the wing coordinate system)
ZB	the vertical distance from the nacelle hilite (bottom meridional plane) to the wing leading edge (expressed in the wing coordinate system)
XWOR	the x coordinate of the nacelle origin (expressed in the wing coordinate system)
YWOR	the y coordinate of the nacelle origin (expressed in the wing coordinate system)
ZWOR	the z coordinate of the nacelle origin (expressed in the wing coordinate system)
J1,J2	the J indices (spanwise direction) of the overlap region inner boundary
NJC	the J index of the wing span station that contains the nacelle plane of symmetry
K1,K2	the K indices (normal direction) of the overlap region inner boundary
KJ1,KJ2	the K indices of the exhaust region at the aft station of the nacelle
N	number of points in the overlap region (J1 $>$ J $<$ J2)
NS	number of points in the overlap region (J=J1 or J=J2)

NN number of points in the exhaust jet region at the aft station of the nacelle

8. TWN FLOW SOLUTION PROGRAM INPUT PARAMETERS

The input data required for execution of the TWN flow solution program are entered by namelist input. In all cases, the two NAMELISTS FLOWIN and FLOWC are entered. Most of the input parameters have assigned default values.

8.1 NAMELIST FLOWIN

The input parameters specified in NAMELIST FLOWIN control the computation of the wing component flow field.

MODE An integer which controls the program mode of operation for the wing component flow field calculation. If MODE=1, the mode of operation is for a wing/nacelle solution. If MODE=0, the mode of operation is for an isolated wing solution. A default value of 1 is specified for MODE.

KCX An integer denoting the number of solution cycles between the wing and nacelle algorithms that are to be performed for wing/nacelle flow field computations. KCX =1 must be entered if MODE=0 is specified. A default value of 4 is specified for KCX.

FMACH A real variable denoting the free-stream Mach number. A default value of 0.75 is specified for FMACH.

ALPHAW A real variable denoting the angle of attack, in degrees. A default value of 0.0 is specified for ALPHAW.

ARAT A real variable denoting the wing aspect ratio based on a simple trapezoidal planform. If ARAT=0.0 is entered, the aspect ratio is computed internally. A default value of 0.0 is specified for ARAT.

TIP A real variable denoting the span location of the wing tip. If TIP=0.0 is entered, then TIP is internally reset to 0.5* (Y(NJT)+Y(NJT-1)). A default value of 0.0 is specified for TIP.

ERROR A real variable denoting the convergence tolerance used for terminating the wing flow field calculation. Convergence is attained if the following relation is satisfied

$$|R^{n}_{max}| / |R^{1}_{max}| \leq \text{CRIT} \qquad (31)$$

where R^{1}_{max} is the maximum residual on the first iteration and R^{n}_{max} is the maximum residual on the nth iteration. A default value of 0.001 is specified for ERROR.

MAXIT A positive integer denoting the maximum number of iterations per cycle permissible in calculating the wing component flow field. A default value of 40 is specified for MAXIT.

The next four parameters specify the acceleration constants used in the

AF2 iteration scheme for the wing algorithm. One method has been incorporated into the program for calculating the acceleration parameter sequence. The method employs the sequence

$$\alpha_K = \alpha_H \, (\alpha_L / \alpha_H)^{\frac{K-1}{M-1}} \qquad (K=1,2,3,\ldots,M) \qquad (32)$$

where α_L is the lower limit for α, α_H is the upper limit for α, K is the sequence element number, and M is the number of elements in the sequence.

ALPHAL A positive real variable denoting the α lower limit α_L. A default value of 0.4 is specified for ALPHAL.

ALPHAH A positive real variable denoting the α upper limit α_H. A default value of 4.0 is specified for ALPHAH.

M A positive integer denoting the number of elements M of the sequence in equation (32). A default value of 8 is specified for M.

KKK An integer denoting the starting element of the sequence given in equation (32). KKK must satisfy the relation $0 \leq KKK \leq M$. A default value of 0 is specified for KKK.

The following parameter specifies the standard relaxation factor used in the AF2 iteration scheme for the wing algorithm.

OMEGA A positive real variable denoting the wing algorithm relaxation factor. For stability, OMEGA must be in the range $0.0 < OMEGA < 2.0$. The default and recommended value for OMEGA is 1.8.

The next three parameters control upwinding of the density and the amount of artificial viscosity and damping used in stabilizing the computations in regions of supersonic flow in the wing flow field solution. Note that upwinding is always applied in the ξ-wraparound coordinate direction.

NDIF An integer which controls the upwinding of the density coefficients in the wing algorithm for regions of supersonic flow. NDIF controls the upwinding options as follows:

NDIF	Option
0	upwinding in the wraparound direction
1	upwinding in the wraparound and spanwise directions
2	upwinding in the wraparound and normal directions
3	upwinding in the wraparound, spanwise, and normal directions

In most cases, upwinding is not required in all three directions. A default value of 1 is specified for NDIF.

CON A positive real variable denoting the artificial viscosity coefficient used in performing upwinding of the density in regions of supersonic flow. CON generally takes the values $1.0 \leq CON \leq 2.0$,

with higher values of CON producing greater upwinding. High values of CON can decrease shock wave resolution. A default value of 1.2 is specified for CON.

BXI A positive real variable denoting the timelike dissipation coefficient which is used in the AF2 algorithm to produce diagonal dominance in the ξ-difference equations and thereby enhance stability in regions of supersonic flow. Increasing BXI increases the amount of dissipation. A default value of 0.1 is specified for BXI.

RGAM A real variable used as a circulation relaxation parameter in the wing flow field solution algorithm. Low values of RGAM retard the build up of circulation whereas higher values have the opposite effect. Too high of values of RGAM can cause divergent oscillations in the iterative solution process. Best results are generally achieved using a high RGAM value (1.3 ~ 1.5) for the first several iterations and then reducing RGAM to a lower value (0.7 ~ 0.9) for the remaining iterations. A default value of 0.7 is specified for RGAM.

INTOUT An integer which denotes the interval in iteration number at which the wing solution is printed. If INTOUT=0, the printing of the solution is performed only at the termination of the calculation. A default value of 0 is specified for INTOUT.

IOUT An integer which controls the amount of printout for the wing flow field solution algorithm. IOUT controls the printout as follows:

IOUT	Print/Plot Option
0	density, Mach number and pressure coefficient plots
1	density and Mach number plots
2	density and pressure coefficient plots
3	density plot
4	pressure coefficient plot
5	Mach number and pressure coefficient plots
6	Mach number plot
7	skip all print routines

A default value of 4 is specified for IOUT.

JCPEND An integer denoting the last η-spanwise wing station up to which the wing solution is printed. If JCPEND=0 is entered, then JCPEND is internally reset to NJT-1. A default value of 0 is specified for JCPEND.

JSKIP An integer denoting the increment in η-spanwise station number at which the wing solution is output. A default value of 1 is specified for JSKIP.

MCP An integer denoting whether or not to store the computed surface pressure coefficient data on disk file TAPE 20. If MCP≠0, the file write is performed. If MCP=0, the file write is not performed. A

default value of 0 is specified for MCP.

MORIN An integer employed in restarting the program from a preceding calculation. If MORIN≠0, the potential field is read from disk file TAPE 11. If MORIN=0, the file read is not performed. A default value of 0 is specified for MORIN.

MOROUT An integer employed in storing the potential field for post-processing or a restart calculation. If MOROUT≠0, the potential field is written onto disk file TAPE 12. If MOROUT=0, the file write is not performed. A default value of 0 is specified for MOROUT.

After NAMELIST FLOWIN is entered, a title card is then entered.

8.2 NAMELIST FLOWC

The input parameters specified in NAMELIST FLOWC control the computation of the nacelle component flow field.

XMCF A real variable denoting the effective Mach number at the compressor face. XMCF must be entered only if the inlet capture ratio (CRATIO) is not entered. A default value of 0.3516 is specified for XMCF.

CRATIO A real variable denoting the inlet capture ratio, which is defined by

$$CRATIO = AINF/AHL \tag{33}$$

where AINF is the cross-sectional area of the capture streamtube far upstream of the inlet, and AHL is the effective hilite area (see Figure 33). If CRATIO≠0.0, then the compressor face Mach number XMCF will be internally computed from the supplied value of CRATIO. If CRATIO = 0.0, then XMCF must be entered directly. A default value of 0.0 is specified for CRATIO.

KINIT An integer denoting the nacelle grid potential function initialization option. If KINIT = 0, the potential function initialization is performed using free-stream velocity components for each point in the computational mesh. If KINIT = 1, the potential function initialization is performed using free-stream velocity components for the external flow mesh points, and using velocity components calculated from one-dimensional gas dynamic formulae for the internal flow mesh points. The local internal flow Mach number is calculated using the compressor face Mach number and the ratio between the local flow area and the compressor face area. Setting KINIT = 1 generally significantly improves convergence speed. The default and recommended value for KINIT is 1.

XMULM A positive real variable used as a Mach number multiplier for the nacelle grid internal flow points if the KINIT = 1 initialization option is specified. XMULM must be entered only if KINIT = 1, and generally is in the range of 0.6 to 0.8. The default value for XMULM is 0.7.

CRIT A real variable denoting the convergence criterion used for terminating the nacelle flow field calculation. Convergence is

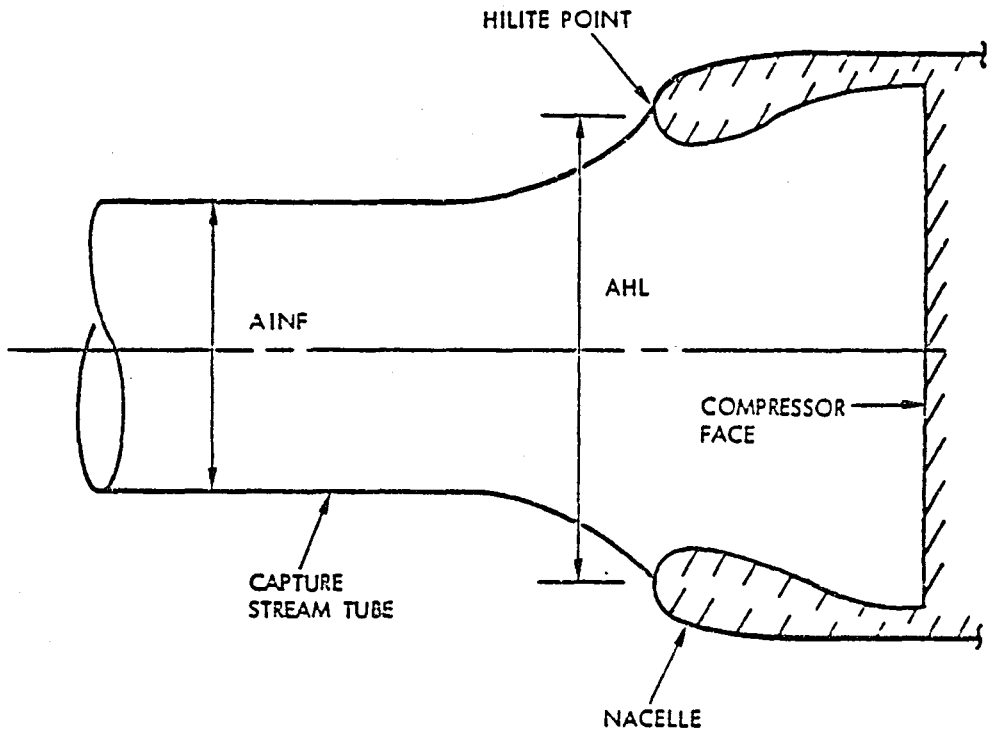


Figure 33. Definition of nacelle AINF and AHL areas.

attained if the following relation is satisfied

$$|R^n_{max}| / |R^1_{max}| \leq CRIT \tag{34}$$

where R^1_{max} is the maximum residual on the first iteration, and R^n_{max} is the maximum residual on the nth iteration. A default value of 0.001 is specified for CRIT.

ITMAX A positive integer denoting the maximum number of iterations per cycle permissible in calculating the nacelle component flow field. A default value of 50 is specified for ITMAX.

The next four parameters specify the acceleration constants used in the AF2 iteration scheme for the nacelle algorithm. Two options have been incorporated into the nacelle program for calculating the acceleration parameter sequence. The first option employs the sequence

$$\alpha_K = \alpha_H \left(\alpha_L / \alpha_H\right)^{\frac{K-1}{M-1}} \qquad (K=1,2,3,\ldots,M) \tag{35}$$

where α_L is the lower limit for α, α_H is the upper limit for α, K is the sequence element number, and M is the number of elements in the sequence. The second option employs the sequence

$$\alpha_1 = \alpha_H$$

$$\alpha_K = \alpha_{K-1} - \left[\frac{\alpha_H - \alpha_L}{28}\right] (9-K) \quad (K=2,3,\ldots,7,8) \tag{36}$$

in which case M equals 8.

KALPHA An integer denoting the α-sequence to be used in the computation. If KALPHA=1, the sequence given by equation (35) is used. If KALPHA=2, the sequence given by equation (36) is used. The default and recommended value for KALPHA is 1.

ALPHAL A positive real variable denoting the α lower limit α_L. A default value of 1.0 is specified for ALPHAL.

ALPHAH A positive real variable denoting the α upper limit α_H. A default value of 20.0 is specified for ALPHAH.

MD A positive integer denoting the number of elements M of the sequence in equation (35). A default value of 8 is specified for MD. MD must be entered as 8 if the KALPHA=2 option is specified.

The following parameter specifies the standard relaxation factor used in the AF2 iteration scheme for the nacelle algorithm.

OMEGA A positive real variable denoting the nacelle algorithm relaxation factor. For stability, OMEGA must be in the range $0.0 < OMEGA < 2.0$.

The default and recommended value for OMEGA is 1.0.

The next three parameters control the upwinding of the density and the amount of artificial viscosity and dissipation used in stabilizing the computations in regions of supersonic flow in the nacelle flow field solution. Note that upwinding is always applied in the ξ-wraparound coordinate direction.

NKVIS — An integer denoting whether or not upwinding in regions of supersonic flow is to be applied in the ζ-radial coordinate direction. If NKVIS=0, the upwinding is not applied. If NKVIS=1, upwinding of the density is applied in the ζ-direction. A default value of 0 is specified for NKVIS.

CFACT — A positive real variable denoting the artificial viscosity coefficient used in performing upwinding of the density in regions of supersonic flow. CFACT generally takes the values $1.0 \leq$ CFACT ≤ 2.0, with higher values of CFACT producing greater upwinding. High values of CFACT can decrease shock wave resolution. A default value of 1.2 is specified for CFACT.

BXIE — A positive real variable denoting the timelike dissipation coefficient which is used in the AF2 algorithm to produce diagonal dominance in the ξ-difference equations and thereby enhance stability in regions of supersonic flow. Increasing BXIE increases the amount of dissipation. A default value of 0.1 is specified for BXIE.

The next three parameters control the smoothing operations performed in the computations.

KSMTH — A positive integer denoting the ζ-radial station index K up to which smoothing is performed for the computed density in the ξ-wraparound coordinate direction in the vicinity of the ITRAN2 ξ-station (see Figure 34). The density smoothing generally improves convergence speed. A default value of 4 is specified for KSMTH.

KTIME — An integer denoting the number of times a smoothing polynomial is applied in smoothing the density values in the ξ-direction in the vicinity of the ITRAN2 ξ-station for the ζ-stations ranging from 1 to KSMTH. If KTIME=0 is entered, no smoothing is performed. The density smoothing generally improves convergence speed. A default value of 3 is specified for KTIME.

NDEL — A positive integer denoting the number of ξ-stations to the left and to the right of point ITRAN2 at which the density is smoothed. Smoothing is performed from point (ITRAN2-NDEL) to point (ITRAN2+NDEL) for the ζ-stations ranging from 2 to KSMTH, and from point (ITRAN2+1) to point (ITRAN2+2*NDEL) for the 1st ζ-station. A default value of 3 is specified for NDEL.

The next two parameters specify the potential function under-relaxation factors.

CORFAT — A one-dimensional real variable array consisting of KMAX elements.

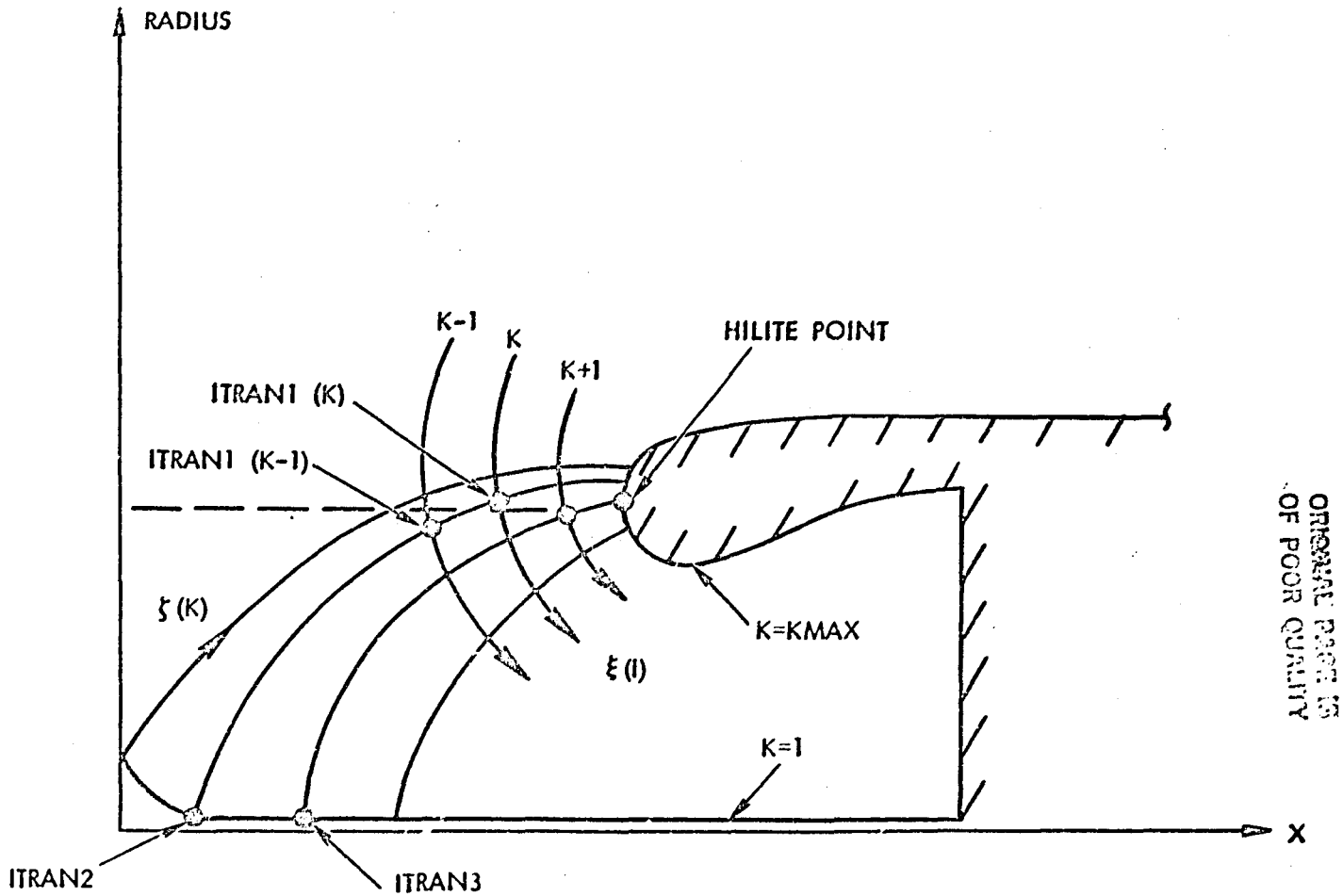


Figure 34. Definition of ITRAN1, ITRAN2, and ITRAN3 points.

CORFAT(K) (K=1,...,KMAX) specifies the potential function under-relaxation factor for the Kth ζ-station. This under-relaxation is necessary to maintain stability and enhance convergence for mesh points near the inlet centerline. Under-relaxation of the potential function is performed only for the ξ-stations which are in the range ITRAN3 $\leq$ I $\leq$ IMAX (see Figure 34). The elements of CORFAT generally range from a low value at K = 1 to a value of 1.0 (no under-relaxation) as K approaches KMAX. The following default values have been assigned to the elements of CORFAT: CORFAT(K) = 0.2, 0.40, 0.60, 0.80, and 1.0 for K = 1 to 5, respectively.

ITCOR — A positive integer denoting the iteration number at which the potential function under-relaxation is terminated. For iteration numbers greater than ITCOR, the standard algorithm with no under-relaxation is applied. Generally, fastest convergence is obtained by using under-relaxation for all iterations. The default and recommended value for ITCOR is 10,000.

VJT — A real variable denoting the exhaust jet velocity normalized by the critical sonic speed. A default value of 0.9 is specified for VJT.

JNPR — An integer which controls the meridional station output frequency when the nacelle surface solution is printed. Setting JNPR=1 prints every circumferential station. A default value of 3 is specified for JNPR.

KDUMP — A one-dimensional integer array consisting of seven elements. Each element of KDUMP specifies whether or not a particular nacelle algorithm subroutine is to have debug output printed. Specifying KDUMP(I)=1 (I=1,...,7) activates the debug output option for the corresponding subroutine. Specifying KDUMP(I) = 0 causes the debug printout to be skipped. The elements of KDUMP activate the debug printout for the following subroutines/computations and have the following default values.

KDUMP(I)	Activates Debug Output for (SUBROUTINE)	Default Value
KDUMP(1)	metric calculations (METRIC)	0
KDUMP(2)	initialization (NINIT)	0
KDUMP(3)	not presently used	0
KDUMP(4)	AF2 solution scheme (NSOLVE)	0
KDUMP(5)	physical density calculation (NRO)	0
KDUMP(6)	residual calculation (NRESID)	0
KDUMP(7)	modified density calculation (NROCO)	0

ITSTRT — An integer denoting the iteration number at which debug output is to be initiated. A default value of 1 is specified for ITSTRT.

8.3 PARAMETER STATEMENT SPECIFICATION

The TWN flow solution program uses variable array dimensions. The respective array sizes are fixed by specification of certain parameters on a PARAMETER statement. The PARAMETER statement for the TWN code is identical to

that used in the LK grid interface code. Hence, the user should refer to Section 6.2 of this chapter for a definition of the dimension parameters for the TWN code.

8.4 FILE USAGE

The following files are used by the TWN flow solution code.

File No.	Usage
TAPE 5	input file
TAPE 6	printed output file
TAPES 9,10	input files from the GRGEN3 wing grid generation code containing the wing grid coordinates
TAPE 11	input file containing the potential field from a previous wing solution (for restart option)
TAPE 12	output file on which the current wing potential field is written (for restart option)
TAPE 19	input file from the LK grid interface code containing the grid overlap region parameters
TAPE 20	output file containing the wing surface pressure coefficient field
TAPE 44	input file from the LK grid interface code containing the scaled nacelle grid coordinates

9. TWN PROGRAM OUTPUT INTERPRETATION

The initial portion of the computer printout for the TWN program consists of the NAMELIST input data. Then the preliminary information listed below is output.

RHOSTR	critical density normalized by stagnation density
RHOINF	free-stream density normalized by stagnation density
QINF	free-stream velocity magnitude normalized by critical sonic speed
NI	maximum number of wing grid points in the wraparound direction
NJ	maximum number of wing grid points in the spanwise direction
NK	maximum number of wing grid points in the normal direction
IHALF	equals NI/2+1

The wing algorithm output parameters are defined below.

ITER	iteration number
ICMAX	location of maximum correction (ξ-index)
JCMAX	location of maximum correction (η-index)
KCMAX	location of maximum correction (ζ-index)
CMAX	maximum correction
IRMAX	location of maximum residual (ξ-index)

JRMAX	location of maximum residual (η-index)
KRMAX	location of maximum residual (ζ-index)
RMAX	maximum residual
NSP	number of supersonic points
ALPHA	acceleration parameter
GAM	an array containing circulation values for four consecutive iterations
IU	grid point number on wing surface
XU	x-coordinate for wing upper surface
ZU	z-coordinate for wing upper surface
XL	x-coordinate for wing lower surface
ZL	z-coordinate for wing lower surface
XUC	x-coordinate for wing upper surface in % of chord
XLC	x-coordinate for wing lower surface in % of chord
CPU	pressure coefficient on wing upper surface
CPL	pressure coefficient on wing lower surface
RHOU	density on wing upper surface normalized by stagnation density
RHOL	density on wing lower surface normalized by stagnation density
MACHU	Mach number on wing upper surface
MACHL	Mach number on wing lower surface
I	wing span station number
Y	y-coordinate of wing span station
Y/TIP	y-coordinate of wing span station in terms of fraction of wing semispan
CL/C	local wing section lift coefficient scaled by local chord
CD/C	local wing section wave drag coefficient scaled by local chord
CM(RTC/4)	local wing section moment coefficient about the root quarter chord

The nacelle algorithm output parameters are defined below.

I	ξ-wraparound point index
J	η-circumferential point index
X	x-coordinate of surface solution point
Y	y-coordinate of surface solution point *
Z	z-coordinate of surface solution point *
DX	axial (x) distance between surface solution point and hilite point for a given meridional plane
RHO	density coefficient defined as the local density normalized by stagnation density
MACH	Mach number
CP	pressure coefficient
NSP	number of supersonic points
RMAX	maximum residual
RAVG	average residual
IRMAX	ξ-location of maximum residual
JRMAX	η-location of maximum residual
KRMAX	ζ-location of maximum residual

* (Note - the y and z axes are along the wingspan for the wing and nacelle algorithms, respectively; the z and y axes are normal to the wing for the wing and nacelle algorithms, respectively)

SECTION V

SAMPLE CASES AND SUGGESTIONS FOR USAGE

1. INTRODUCTION

In this section, sample cases are presented to illustrate application of both the GRGEN3 and NGRIDA grid generation programs, the LK grid interface program, and the TWN flow analysis program. For each sample case, a discussion of the problem is given, the required input data are presented, and selected portions of the computer output are listed. The input parameter discussions follow the order in which the input parameters are presented in Section IV. At the end of this section, suggestions are provided for use of the computer programs.

2. GRGEN3 WING GRID GENERATION PROGRAM SAMPLE CASE

This sample case is presented to illustrate application of the GRGEN3 wing grid generation program to determining the computational grid about a NACA 0012 unswept and untapered wing configuration. This case represents the case being considered when the default wing geometry contained within the GRGEN3 code is used.

The data deck for this sample case is illustrated in Figure 35. All input parameters in NAMELIST GRID3D retain their default values except for NJ, NK, NJT, NJC, DYTIP, DYNCL, and DYROOT. The number of spanwise and normal stations to be used in the grid are 30 and 22, respectively; hence, NJ = 30 and NK = 22 are entered in NAMELIST GRID3D. The spanwise indices denoting the first station off the wing tip and the nacelle location are 25 and 13, respectively; hence, NJT = 25 and NJC = 13 are specified. The spanwise spacings near the wing tip and wing root are 0.4 and 0.2, respectively; hence, DYTIP = 0.4 and DYROOT = 0.2 are specified. The spanwise grid spacing near the nacelle symmetry plane is controlled by entering DYNCL=0.20.

All input parameters in NAMELIST GRID3N retain their default values except for the grid point distribution parameter BINN. BINN = 1.08 is entered which clusters points near the leading and trailing edges of the wing surface.

Selected portions of the computer output for this sample case are presented in Figure 36.

3. NGRIDA NACELLE GRID GENERATION PROGRAM SAMPLE CASE

This sample case is presented to illustrate application of the NGRIDA nacelle grid generation program to determining the interface computational grid about the Lockheed-Georgia GELAC1 axisymmetric nacelle/inlet configuration. This case represents the case being considered when the default nacelle geometry contained within the NGRIDA code is used.

The data deck for this sample case is illustrated in Figure 37. All input parameters in NAMELIST LIST1 retain their default values except for KMAX, RADOUT, XLEFT, ALPHA0, ALPHAI, and ITAPE. Five radial stations will be

```
$GRID3D NJ=30, NK=22, NJT=25, NJC=13, DYTIP=0.4,
 DYNCL=0.20, DYROOT=0.2 $END
$GRIDIN BINN=1.08 $END
```

Figure 35. Data deck for GRGEN3 sample case.

```
GRGEN3, THREE DIMENSIONAL WING GRID GENERATOR USING A TWO DIMENSIONAL GRID AT EACH SPAN STATION.
WRITTEN BY TERRY HOLST (APPLIED COMPUTATIONAL AERODYNAMICS BRANCH), ALAN FERNQUIST, AND SCOTT D.
*** NASA AMES RESEARCH CENTER, MOFFETT FIELD, CALIFORNIA 94035 ***

THE MAXIMUM DIMENSIONS ARE 151 BY 30 BY 30.
$GRID3D
DYROOT  =  0.2000000000000000    ,
DYTIP   =  0.4000000000000000    ,
DYNCL   =  0.2000000000000000    ,
NI      =          101,
NJ      =           30,
NJC     =           13,
NJT     =           25,
NK      =           22,
SFACT   =   1.200000000000000    ,
SMOOTH  =   1.000000000000000    ,
SWPANG  =  0.0000000000000000E+00,
TIP     =   6.000000000000000    ,
TWANG   =  0.0000000000000000E+00,
CEL     =   3.000000000000000
$END

NOTE THAT THERE ARE 101* 30= 3030 POINTS IN EACH K-PLANE, AND 101* 30* 22= 66660 TOTAL POINTS.
INCIDENTALLY,101* 30* 22*6= 399960

        INTERNALLY GENERATED Y DISTRIBUTION

        FOR TIP AT Y =6.0000, AND NJT = 25

                J           Y(J)

                1       0.000000
                2       0.200000
                3       0.411177
                4       0.647392
                5       0.911773
```

Figure 36. Output for GRGEN3 sample case.

```
                    27        7.080000
                    28        7.656000
                    29        8.347200
                    30        9.176640
 $GRIDIN
 AHGRID  =  1.0000000000000000E-02,
 ALGRID  =  6.0000000000000000E-05,
 BINN    =   1.000000000000000    ,
 ERGRID  =  9.9999999999999999E-04,
 IINCR   =             3,
 IOPT    =             1,
 JINCR   =             6,
 KGRID   =             0,
 MAXIT   =           200,
 MGRID   =             8,
 NOUT5   =             0,
 NOUT6   =           201,
 NSTATN  =             0,
 ONEG    =   2.000000000000000    ,
 RADMAX  =   6.000000000000000    ,
 TMAX    =  0.12000000000000000   ,
 XC      =  0.20000000000000000   ,
 XCN     =  0.00000000000000000E+00
 $END

 NOTE THAT FOR THE NACA00XX OPTION THE THICKNESS-TO-CHORD RATIO IS ACTUALLY 0.1109.
```

Figure 36. Continued.

*** FINAL AIRFOIL COORDINATE DISTRIBUTION, Y = 0.000000 ***

I	XB	YB	S2	SX	I	XB	YB	S2	SX	I	XB	YB
1	1.00000	0.00000	0.00000	0.00267	35	0.21523	0.00000	0.78477	0.02540	69	0.26946	0.00000
2	0.99467	0.00000	0.00533	0.00567	36	0.19071	0.00000	0.80929	0.02364	70	0.29901	0.00000
3	0.98866	0.00000	0.01134	0.00639	37	0.16795	0.00000	0.83205	0.02189	71	0.33003	0.00000
4	0.98189	0.00000	0.01811	0.00718	38	0.14694	0.00000	0.85306	0.02015	72	0.36235	0.00000
5	0.97430	0.00000	0.02570	0.00806	39	0.12764	0.00000	0.87236	0.01847	73	0.39576	0.00000
6	0.96578	0.00000	0.03422	0.00902	40	0.11000	0.00000	0.89000	0.01685	74	0.43002	0.00000
7	0.95625	0.00000	0.04375	0.01008	41	0.09395	0.00000	0.90605	0.01531	75	0.46486	0.00000
8	0.94561	0.00000	0.05439	0.01124	42	0.07939	0.00000	0.92061	0.01386	76	0.50000	0.00000
9	0.93376	0.00000	0.06624	0.01250	43	0.06624	0.00000	0.93376	0.01250	77	0.53514	0.00000
10	0.92061	0.00000	0.07939	0.01386	44	0.05439	0.00000	0.94561	0.01124	78	0.56998	0.00000
11	0.90605	0.00000	0.09395	0.01531	45	0.04375	0.00000	0.95625	0.01008	79	0.60424	0.00000
12	0.89000	0.00000	0.11000	0.01685	46	0.03422	0.00000	0.96578	0.00902	80	0.63765	0.00000
13	0.87236	0.00000	0.12764	0.01847	47	0.02571	0.00000	0.97430	0.00806	81	0.66997	0.00000
14	0.85306	0.00000	0.14694	0.02015	48	0.01809	0.00000	0.98189	0.00718	82	0.70099	0.00000
15	0.83205	0.00000	0.16795	0.02189	49	0.01139	0.00000	0.98866	0.00639	83	0.73054	0.00000
16	0.80929	0.00000	0.19071	0.02364	50	0.00523	0.00000	0.99467	0.00567	84	0.75850	0.00000
17	0.78477	0.00000	0.21523	0.02540	51	0.00000	0.00000	1.00000	0.00533	85	0.78477	0.00000
18	0.75850	0.00000	0.24150	0.02711	52	0.00523	0.00000	1.00533	0.00567	86	0.80929	0.00000
19	0.73054	0.00000	0.26946	0.02876	53	0.01139	0.00000	1.01134	0.00639	87	0.83205	0.00000
20	0.70099	0.00000	0.29901	0.03029	54	0.01809	0.00000	1.01811	0.00718	88	0.85306	0.00000
21	0.66997	0.00000	0.33003	0.03167	55	0.02571	0.00000	1.02570	0.00806	89	0.87236	0.00000
22	0.63765	0.00000	0.36235	0.03286	56	0.03422	0.00000	1.03422	0.00902	90	0.89000	0.00000
23	0.60424	0.00000	0.39576	0.03383	57	0.04375	0.00000	1.04375	0.01008	91	0.90605	0.00000
24	0.56998	0.00000	0.43002	0.03455	58	0.05439	0.00000	1.05439	0.01124	92	0.92061	0.00000
25	0.53514	0.00000	0.46486	0.03499	59	0.06624	0.00000	1.06624	0.01250	93	0.93376	0.00000
26	0.50000	0.00000	0.50000	0.03514	60	0.07939	0.00000	1.07939	0.01386	94	0.94561	0.00000
27	0.46486	0.00000	0.53514	0.03499	61	0.09395	0.00000	1.09395	0.01531	95	0.95625	0.00000
28	0.43002	0.00000	0.56998	0.03455	62	0.11000	0.00000	1.11000	0.01685	96	0.96570	0.00000
29	0.39576	0.00000	0.60424	0.03383	63	0.12764	0.00000	1.12764	0.01847	97	0.97430	0.00000
30	0.36235	0.00000	0.63765	0.03286	64	0.14694	0.00000	1.14694	0.02015	98	0.98189	0.00000
31	0.33003	0.00000	0.66997	0.03167	65	0.16795	0.00000	1.16795	0.02189	99	0.98866	0.00000
32	0.29901	0.00000	0.70099	0.03029	66	0.19071	0.00000	1.19071	0.02364	100	0.99467	0.00000
33	0.26946	0.00000	0.73054	0.02876	67	0.21523	0.00000	1.21523	0.02540			
34	0.24150	0.00000	0.75850	0.02711	68	0.24150	0.00000	1.24150	0.02711			

Figure 36. Continued.

```
$LIST1 KMAX=5, RADOUT=16.0, XLEFT=-12.0, ALPHAO=1.09,
ALPHAI=1.09, ITAPE=14 $END
$LIST2 $END
$LIST3 $END
$LIST4 $END
```

Figure 37. Data deck for NGRIDA sample case.

used in the interface grid, hence KMAX=5 is specified in NAMELIST LIST1. Because an interface grid is being generated, the magnitudes of the outer boundary parameters RADOUT and XLEFT are reduced from their default values to 16.0 and -12.0, respectively. The ALPHAO and ALPHAI grid stretching function parameters are changed slightly from their default values to a value of 1.09 each. Since this is an interface grid generation, the nacelle grid coordinate data must be loaded onto disk file TAPE 14; hence, ITAPE = 14 is specified.

All input parameters entered in NAMELISTS LIST2, LIST3, and LIST4 retain their default values.

Selected portions of the computer output for this sample case are presented in Figure 38.

4. LK GRID INTERFACE PROGRAM SAMPLE CASE

This sample case is presented to illustrate application of the LK grid interface program. The case considered interfaces the NACA 0012 wing grid and the GELAC1 nacelle grid generated by the sample cases of the GRGEN3 and NGRIDA programs, respectively.

The data deck for this sample case is illustrated in Figure 39. All input parameters in NAMELIST NACEL retain their default values except for XR. The distance, expressed in terms of the fraction of the local chord, between the nacelle hilite point and the wing leading edge point is 1.0. Hence, XR = 1.0 is specified in NAMELIST NACEL.

Selected portions of the computer output for this sample case are presented in Figure 40.

5. TWN FLOW SOLUTION PROGRAM SAMPLE CASE

This sample case is presented to illustrate application of the TWN flow analysis program. This case considers the multicomponent flow field computation about the NACA0012-wing/GELAC1-nacelle configuration. This case employs the grids and interface parameters generated by the GRGEN3, NGRIDA, and LK sample case executions.

The data deck for this sample case is illustrated in Figure 41. All input parameters in NAMELIST FLOWIN retain their default values except for ALPHAW, NDIF, RGAM, and JSKIP. For this case, the free-stream Mach number is retained at its default value of 0.75. The angle of attack considered in this case equals 1.0 degree, hence ALPHAW=1.0 is entered in NAMELIST FLOWIN. Upwinding in regions of supersonic flow will be applied only in the wraparound direction, hence NDIF=0 is entered. The wing circulation relaxation parameter RGAM is entered as 1.1. The spanwise output frequency for the wing solution is specified as JSKIP=12. A title card is entered after NAMELIST FLOWIN.

All input parameters in NAMELIST FLOWC retain their default values except for CRATIO and JNPR. For this case, the inlet capture ratio CRATIO is specified as 0.7. The circumferential output frequency for the nacelle solution is now specified as JNPR=6.

Selected portions of the computer output for this sample case are

THREE-DIMENSIONAL NACELLE GRID GENERATION PROGRAM

LISTING OF INPUT DATA

```
$LIST1
IMAX    =          67,
JMAX    =          13,
KMAX    =           5,
XLEFT   =  -12.00000    ,
RADOUT  =   16.00000    ,
DELTA   =  0.5000000    ,
IOUTER  =          38,
IINNER  =          30,
ALPHAI  =   1.090000    ,
ALPHAO  =   1.090000    ,
KOUTER  =           0,
ALPHBI  =   1.000000    ,
ALPHBO  =   1.000000    ,
SCALE   =  0.1190476    ,
ITAPE   =          14,
KPLOT   =           1,
KDEV    =           1,
KDUMP   =           1,
IAA     =          12,
IDD     =          57,
KPRINT  =           1, 24*0,
KTIME   =           3,
NDEL    =           5,
DSFACT  =   1.000000    ,
KRAMP   =           1,
IRAMP   =          57
$END
$LIST2
NO      =         100,
XOUT    =  0.0000000E+00,  4.9999999E-03,  9.9999998E-03,  1.5000000E-02,
 2.0000000E-02,  2.5000000E-02,  2.9999999E-02,  3.5000000E-02,  3.9999999E-02,
 4.5000002E-02,  5.0000001E-02,  5.5000000E-02,  5.9999999E-02,  6.4999998E-02,
 7.0000000E-02,  7.9999998E-02,  9.0000004E-02,  0.1000000    ,  0.1100000    ,
```

Figure 38. Output for NGRIDA sample case.

GRID POINT COORDINATES FOR SELECTED MERIDIONAL PLANES

I=	1	J=	1	K=	1	X=	0.619048E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	2	J=	1	K=	1	X=	0.581150E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	3	J=	1	K=	1	X=	0.544038E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	4	J=	1	K=	1	X=	0.509019E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	5	J=	1	K=	1	X=	0.476676E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	6	J=	1	K=	1	X=	0.446923E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	7	J=	1	K=	1	X=	0.419443E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	8	J=	1	K=	1	X=	0.393825E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	9	J=	1	K=	1	X=	0.369473E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	10	J=	1	K=	1	X=	0.345491E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	11	J=	1	K=	1	X=	0.320696E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	12	J=	1	K=	1	X=	0.294156E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	13	J=	1	K=	1	X=	0.266812E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	14	J=	1	K=	1	X=	0.241706E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	15	J=	1	K=	1	X=	0.220771E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	16	J=	1	K=	1	X=	0.203231E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	17	J=	1	K=	1	X=	0.187544E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	18	J=	1	K=	1	X=	0.175016E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	19	J=	1	K=	1	X=	0.163354E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	20	J=	1	K=	1	X=	0.152214E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	21	J=	1	K=	1	X=	0.141303E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	22	J=	1	K=	1	X=	0.130354E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	23	J=	1	K=	1	X=	0.119096E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	24	J=	1	K=	1	X=	0.107226E+01	Y=	0.190476E+01	Z=	0.000000E+00
I=	25	J=	1	K=	1	X=	0.943739E+00	Y=	0.190476E+01	Z=	0.000000E+00
I=	26	J=	1	K=	1	X=	0.800539E+00	Y=	0.190476E+01	Z=	0.000000E+00
I=	27	J=	1	K=	1	X=	0.635768E+00	Y=	0.190476E+01	Z=	0.000000E+00
I=	28	J=	1	K=	1	X=	0.439106E+00	Y=	0.190476E+01	Z=	0.000000E+00
I=	29	J=	1	K=	1	X=	0.251400E+00	Y=	0.188206E+01	Z=	0.000000E+00
I=	30	J=	1	K=	1	X=	0.109190E+00	Y=	0.179077E+01	Z=	0.000000E+00

Figure 38. Continued.

```
$NACEL XR=1.0 $END
```

Figure 39. Data deck for LK sample case.

```
$NACEL
XR      =   1.000000    ,
ZT      =  0.4000000    ,
SCALE   =  0.2200000
$END

    XR=  -1.00000    XL=   0.04762    ZT=  -0.40000    ZB=  -0.87143

    XWOR=  -1.00000    YWOR=   3.00000    ZWOR=  -0.63571

    NJC=13      J1=11      J2=15

         NUMBER OF POINTS ON ZIGZAG BOUNDARY N =     44

    K1=  9      K2= 17

         NUMBER OF POINTS ON ZIGZAG SURFACE NS =     94

    NO.OF EXHAUST POINTS NN =    5
    KJ1=  14  KJ2=  16
```

Figure 40. Output for LK sample case.

```
 $FLOWIN ALPHAW=1.0, NDIF=0, RGAM=1.1, JSKIP=12 $END
NACA0012/GELAC1
 $FLOWC CRATIO=0.7, JNPR=6 $END
```

Figure 41. Data deck for TWN sample case.

presented in Figure 42.

6. SUGGESTIONS FOR USAGE

The GRGEN3 and NGRIDA grid generation programs, the LK grid interface program, and the TWN flow analysis program have been executed on the VAX-11/780 and CRAY-1 computers. All of these programs employ PARAMETER statements which fix the respective array dimensions at the time of compilation. This provides an effective means of changing program core storage requirements. The VAX version of each program employs an INCLUDE statement which inserts the respective PARAMETER statement at the appropriate locations within the code. The GRGEN3 wing grid generation program employs the 'INCLUDE 'GRGEN3.DIM" statement. The NGRIDA nacelle grid generation program employs the 'INCLUDE 'NGRID.DIM" statement. The LK grid interface program employs the 'INCLUDE 'TWING.DIM" and 'INCLUDE 'NACELLE.DIM" statements. The TWN flow analysis program employs the same INCLUDE statements as does the LK program. If the codes are executed on a VAX, then the GRGEN3.DIM, NGRID.DIM, TWING.DIM, and NACELLE.DIM files must reside in the user's directory. For execution on the CRAY-1 or CDC computers, the PARAMETER statement is edited into the respective code by use of a COMDECK. For use on other computer systems, the PARAMETER statements will have to be inserted in the respective programs at the locations noted by the INCLUDE statements.

```
$FLOWIN
ALPHAH  =   4.000000     ,
ALPHAL  =  0.4000000     ,
ALPHAW  =   1.000000     ,
ARAT    =  0.0000000E+00,
BXI     =  0.1000000     ,
CON     =   1.200000     ,
ERROR   =  1.0000000E-03,
FMACH   =  0.7500000     ,
INTOUT  =             0,
IOUT    =             4,
JCPEND  =             0,
JSKIP   =            12,
KKK     =             0,
MAXIT   =            40,
M       =             8,
MCP     =             0,
MORIN   =             0,
MOROUT  =             0,
NDIF    =             0,
OMEGA   =   1.800000     ,
RGAM    =   1.100000     ,
TIP     =  0.0000000E+00,
MODE    =             1,
KCX     =             4
$END
```

Figure 42. Output for TWN sample case.

3D TRANSONIC FULL POTENTIAL WING-NACELLE PROGRAM WITH MAXIMUM DIMENSIONS 121 BY 30 BY 25.

NACA0012/GELAC1

NI= 101 NJ= 30 NK= 22 NJT= 25

FMACH	ALPHAW	ALPHAH	ALPHAL	RHOSTR	GAMMA
0.750	1.000	4.000	0.400	0.634	1.400

BXI	OMEGA	RHOINF	QINF	NJT
0.100	1.800	0.766	0.779	25

NI	NJ	NK	IHALF
101	30	22	51

Figure 42. Continued.

TRANSONIC NACELLE/INLET FLOW VARIABLES

LISTING OF INPUT DATA

```
$FLOWC
XMCF    =  0.3516000    ,
CRATIO  =  0.7000000    ,
KINIT   =           1,
ITMAX   =          50,
CRIT    =  1.0000000E-03,
XMULM   =  0.7000000    ,
KALPHA  =           1,
ALPHAL  =   1.000000    ,
ALPHAH  =   20.00000    ,
MD      =           0,
OMEGA   =   1.000000    ,
NKVIS   =           0,
CFACT   =   1.200000    ,
BXIE    =  0.1000000    ,
KSMTH   =           4,
NDEL    =           3,
KTIME   =           3,
VJT     =  0.9000000    ,
ITCOR   =       10000,
CORFAT  =  0.2000000    , 0.4000000    , 0.6000000    , 0.8000000    ,
  1.000000    , 5*0.0000000E+00,
KDUMP   = 20*0,
ITSTRT  =           1,
JNPR    =           6
$END
```

Figure 42. Continued.

SPAN STATION Y(13) = 3.000 IS 50.00 PERCENT OF SEMISPAN.

IU	XU	ZU	XL	ZL	%XUC	%XLC	CPU	CPL	RHOU	RHOL	MACHU	MACHL
50	0.000	0.003	0.000	-0.003	0.0	0.0	1.04533	1.14357	0.9799	0.9990	0.2019	0.0444
49	0.002	0.008	0.002	-0.008	0.2	0.2	0.69369	0.96144	0.9103	0.9635	0.4370	0.2738
48	0.006	0.013	0.006	-0.013	0.6	0.6	0.28737	0.65560	0.8270	0.9026	0.6283	0.4574
47	0.012	0.018	0.012	-0.018	1.2	1.2	-0.02445	0.38247	0.7608	0.8468	0.7602	0.5865
46	0.019	0.023	0.019	-0.023	1.9	1.9	-0.22963	0.18044	0.7159	0.8045	0.8457	0.6741
45	0.027	0.027	0.027	-0.027	2.7	2.7	-0.36646	0.03341	0.6853	0.7732	0.9032	0.7360
44	0.036	0.031	0.036	-0.031	3.6	3.6	-0.46499	-0.07799	0.6630	0.7492	0.9452	0.7025
43	0.047	0.034	0.047	-0.034	4.7	4.7	-0.54286	-0.16625	0.6451	0.7299	0.9760	0.8192
42	0.059	0.038	0.059	-0.038	5.9	5.9	-0.61508	-0.23857	0.6284	0.7139	1.0105	0.8404
41	0.073	0.041	0.073	-0.041	7.3	7.3	-0.66221	-0.29875	0.6173	0.7005	1.0315	0.8746
40	0.088	0.044	0.088	-0.044	8.8	8.8	-0.70372	-0.34882	0.6076	0.6893	1.0502	0.8957
39	0.105	0.047	0.105	-0.047	10.5	10.5	-0.74127	-0.39030	0.5987	0.6800	1.0672	0.9133
38	0.124	0.050	0.124	-0.050	12.4	12.4	-0.77581	-0.42347	0.5904	0.6725	1.0831	0.9274
37	0.144	0.052	0.144	-0.052	14.4	14.4	-0.80010	-0.44924	0.5846	0.6666	1.0943	0.9384
36	0.166	0.055	0.166	-0.055	16.6	16.6	-0.81403	-0.46697	0.5813	0.6621	1.1008	0.9469
35	0.190	0.056	0.190	-0.056	19.0	19.0	-0.81722	-0.48340	0.5805	0.6588	1.1023	0.9531
34	0.216	0.058	0.216	-0.058	21.6	21.6	-0.80697	-0.48966	0.5830	0.6574	1.0975	0.9558
33	0.244	0.059	0.244	-0.059	24.4	24.4	-0.77543	-0.48279	0.5905	0.6589	1.0829	0.9528
32	0.273	0.059	0.273	-0.059	27.3	27.3	-0.70482	-0.46268	0.6073	0.6635	1.0507	0.9442
31	0.304	0.059	0.304	-0.059	30.4	30.4	-0.56338	-0.43365	0.6404	0.6701	0.9870	0.9310
30	0.336	0.059	0.336	-0.059	33.6	33.6	-0.50872	-0.40035	0.6530	0.6777	0.9640	0.9176

Figure 42. Continued.

29	0.370	0.058	0.370	-0.058	37.0	37.0	-0.48208	-0.36509	0.6591	0.6857	0.9525	0.9026
28	0.404	0.057	0.404	-0.057	40.4	40.4	-0.43954	-0.32972	0.6668	0.6936	0.9343	0.8877
27	0.440	0.056	0.440	-0.056	44.0	44.0	-0.39680	-0.29516	0.6785	0.7013	0.9161	0.8731
26	0.475	0.054	0.475	-0.054	47.5	47.5	-0.35581	-0.26186	0.6877	0.7087	0.8987	0.8592
25	0.511	0.051	0.511	-0.051	51.1	51.1	-0.31678	-0.23022	0.6965	0.7158	0.8822	0.8459
24	0.547	0.049	0.547	-0.049	54.7	54.7	-0.27996	-0.20025	0.7047	0.7224	0.8660	0.8334
23	0.582	0.046	0.582	-0.046	58.2	58.2	-0.24523	-0.17182	0.7124	0.7287	0.8522	0.8216
22	0.616	0.043	0.616	-0.043	61.6	61.6	-0.21249	-0.14497	0.7197	0.7345	0.8385	0.8104
21	0.649	0.040	0.649	-0.040	64.9	64.9	-0.18155	-0.11942	0.7265	0.7401	0.8256	0.7997
20	0.682	0.037	0.682	-0.037	68.2	68.2	-0.15207	-0.09499	0.7330	0.7455	0.8133	0.7896
19	0.712	0.034	0.712	-0.034	71.2	71.2	-0.12406	-0.07158	0.7391	0.7506	0.8017	0.7798
18	0.741	0.032	0.741	-0.032	74.1	74.1	-0.09705	-0.04890	0.7450	0.7555	0.7904	0.7704
17	0.769	0.029	0.769	-0.029	76.9	76.9	-0.07100	-0.02673	0.7507	0.7603	0.7796	0.7611
16	0.795	0.026	0.795	-0.026	79.5	79.5	-0.04553	-0.00493	0.7562	0.7650	0.7690	0.7521
15	0.819	0.023	0.819	-0.023	81.9	81.9	-0.02050	0.01683	0.7616	0.7697	0.7586	0.7430
14	0.841	0.021	0.841	-0.021	84.1	84.1	0.00425	0.03845	0.7670	0.7743	0.7482	0.7339
13	0.861	0.018	0.861	-0.018	86.1	86.1	0.02864	0.06039	0.7722	0.7790	0.7380	0.7248
12	0.880	0.016	0.880	-0.016	88.0	88.0	0.05377	0.08265	0.7776	0.7838	0.7275	0.7154
11	0.897	0.014	0.897	-0.014	89.7	89.7	0.07898	0.10559	0.7830	0.7887	0.7170	0.7058
10	0.912	0.012	0.912	-0.012	91.2	91.2	0.10479	0.12936	0.7885	0.7937	0.7061	0.6958
9	0.926	0.010	0.926	-0.010	92.6	92.6	0.13173	0.15440	0.7942	0.7990	0.6948	0.6852
8	0.939	0.008	0.939	-0.008	93.9	93.9	0.16001	0.18081	0.8002	0.8046	0.6828	0.6740
7	0.950	0.007	0.950	-0.007	95.0	95.0	0.18992	0.20921	0.8065	0.8106	0.6701	0.6619
6	0.961	0.005	0.961	-0.005	96.1	96.1	0.22126	0.23938	0.8131	0.8169	0.6567	0.6490
5	0.970	0.004	0.970	-0.004	97.0	97.0	0.25279	0.26969	0.8198	0.8233	0.6432	0.6359
4	0.978	0.003	0.978	-0.003	97.8	97.8	0.27785	0.29448	0.8250	0.8285	0.6324	0.6252
3	0.985	0.002	0.985	-0.002	98.5	98.5	0.30298	0.31932	0.8303	0.8337	0.6215	0.6143
2	0.992	0.001	0.992	-0.001	99.2	99.2	0.32817	0.34423	0.8355	0.8388	0.6105	0.6034
1	0.997	0.000	0.997	0.000	99.7	99.7	0.35343	0.36919	0.8408	0.8440	0.5994	0.5924

*** LIFT, DRAG AND MOMENT COEFFICIENTS FOR Y= 3.000 ***

CL = 0.144068E+00 CD = 0.138965E-02 CM = -0.206608E-02

TOTAL NUMBER OF CYCLES = 5

Figure 42. Continued.

SECTION VI

CONCLUSIONS AND RECOMMENDATIONS

An efficient grid interfacing zonal algorithm has been developed to compute the three-dimensional transonic flow field about wing/nacelle multicomponent configurations. In the present approach, a component-adaptive grid embedding scheme has been employed in which the global computational grid is composed of a series of overlapped component grids, where each component grid is optimized for a particular geometry such as the wing or nacelle. The AF2 approximate factorization algorithm is used to determine the full-potential equation solution on each component grid with trivariate interpolation being used to transfer property information between the grids. The optimized grid system produces the desired balance between convergence speed and accuracy of the flow field solution. Numerical results using the overlapped grid scheme indicate that the present algorithm promises to be effective in computing the flow field about multicomponent configurations.

To improve the accuracy of the present algorithm, it is suggested that a viscous correction capability be added to the inviscid flow prediction algorithm. This can be accomplished by either adding a boundary layer displacement thickness distribution onto the original body contours, or by using surface transpiration velocity boundary conditions in which the effective transpiration velocity is calculated from the computed boundary layer growth. Although the boundary layer flows on both the wing and nacelle are three-dimensional, using a two-dimensional boundary layer analysis applied in a stripwise sense at span stations on the wing and at meridional stations on the nacelle may prove to be adequate for effecting a viscous correction to the inviscid algorithm.

In order to correctly predict the wing lower surface pressure distribution near the wing leading edge, it will be necessary to complete the pylon modeling. To model a pylon of realistic thickness, it will probably be necessary to introduce a third component C-type grid around the pylon or to at least adjust the wing and nacelle component grids to have grid lines aligned with the pylon leading and trailing edges.

It also appears that using an H-type wing grid may prove beneficial in attempting to interface the wing and nacelle flow solution algorithms. In an H-type wing grid, the mesh has greater uniformity and the mesh cells are more Cartesian-like underneath the wing surface in the vicinity of the nacelle. This should aid in interpolating flow properties from one component grid to another.

REFERENCES

1. Ballhaus, W. F., Jameson, A., and Albert, J., "Implicit Approximate-Factorization Schemes for Steady Transonic Flow Problems," AIAA Journal, Vol. 16, June 1978, pp. 573-579.

2. Holst, T. L., "An Implicit Algorithm for the Conservative Transonic Full Potential Equation Using an Arbitrary Mesh," AIAA Journal, Vol. 17, Oct. 1979, pp. 1038-1045.

3. Holst, T. L., and Thomas, S. D., "Numerical Solution of Transonic Wing Flow Fields," AIAA Paper No. 82-0105, Jan. 1982.

4. Vadyak, J., and Atta, E. H., "A Computer Program for the Calculation of Three-Dimensional Transonic Nacelle/Inlet Flowfields", NASA CR-166528 Nov., 1983.

5. Atta, E. H., and Vadyak, J., "A Grid Interfacing Zonal Algorithm for Three-Dimensional Transonic Flows About Aircraft Configurations," AIAA Paper No. 82-1017, June 1982; see also: "A Grid Overlapping Scheme for Flowfield Computations About Multicomponent Configurations," AIAA Journal, Vol. 21, Sept. 1983, pp. 1271-1277.

6. Sorensen, R. L., "A Computer Program to Generate Two-Dimensional Grids About Airfoils and Other Shapes by Use of Poisson's Equation," NASA TM-81198, 1980.

7. Atta, E. H., "Component-Adaptive Grid Interfacing", AIAA Paper No. 81-0382, Jan. 1981.

8. Harris, A. E., and Pauley, G. I., "Simulation Techniques for Pylon-Mounted Turbo-Fan Engines", Report No. 36, Aircraft Research Association Limited, Bedford, England, Oct. 1975.

9. Flechner, S. G., and Patterson, J. C., "Tabulated Pressure Measurements on a Large Subsonic Transport Model Airplane with High-Bypass-Ratio, Powered, Fan-Jet Engines", NASA TM X-2530, May 1972.

10. DISSPLA MANUAL, Integrated Software Systems Corporation, San Diego, California, Dec. 1980.

END

DATE

FILMED

FEB 10 1984

End of Document