NUMERICAL GENERATION OF TWO-DIMENSIONAL GRIDS BY

THE USE OF POISSON EQUATIONS WITH GRID CONTROL

AT BOUNDARIES

Reese L. Sorenson
NASA Ames Research Center, Moffett Field, CA 94035

and

Joseph L. Steger[*]
Flow Simulations, Inc., Sunnyvale, CA 94086

Abstract

A new method for generating boundary-fitted, curvilinear, two-dimensional grids by the use of the Poisson equations is presented. Grids of C-type and O-type have been made about airfoils and other shapes, with circular, rectangular, cascade-type, and other outer boundary shapes. Both viscous and inviscid spacings have been used. In all cases two important types of grid control can be exercised at both inner and outer boundaries. First is arbitrary control of the distances between the boundaries and the adjacent lines of the same coordinate family, i.e., "stand-off" distances. Second is arbitrary control of the angles with which lines of the opposite coordinate family intersect the boundaries. Thus, both grid cell size (or aspect ratio) and grid cell skewness are controlled at boundaries. Reasonable cell size and shape are ensured even in cases wherein extreme boundary shapes would tend to cause skewness or poorly controlled grid spacing. An inherent feature of the Poisson equations is that lines in the interior of the grid smoothly connect the boundary points (the grid mapping functions are second-order differentiable).

A user-oriented, well documented, FORTRAN computer program, called GRAPE, has been written to employ this grid generation method. It is available from the Applied Computational Aerodynamics Branch at NASA-Ames Research Center.
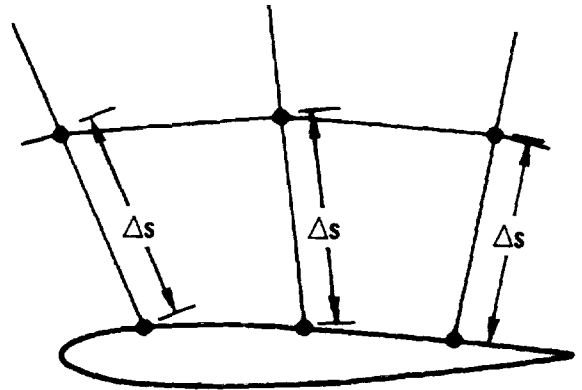
[*]Now with Stanford University.

DESIRED PROPERTIES OF A GRID GENERATOR

- ARBITRARY BOUNDARY SHAPES

- ARBITRARY POINT DISTRIBUTION ON BOUNDARIES

- SMOOTH VARIATION (DIFFERENTIABILITY) IN INTERIOR

- COMPUTATIONALLY FAST
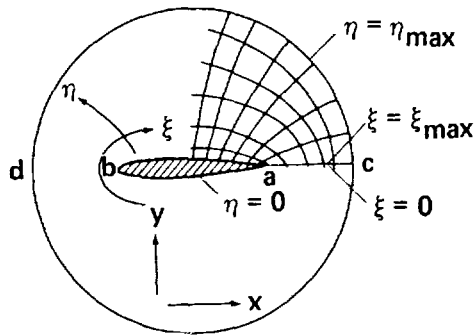
- EASY TO USE

- CONTROL OF ANGLES
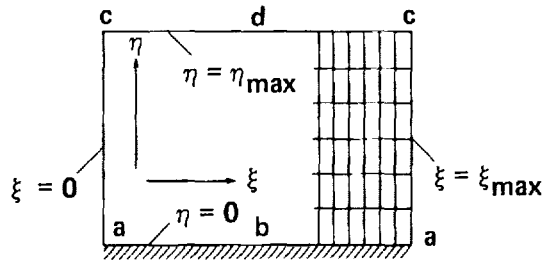 AT BOUNDARIES
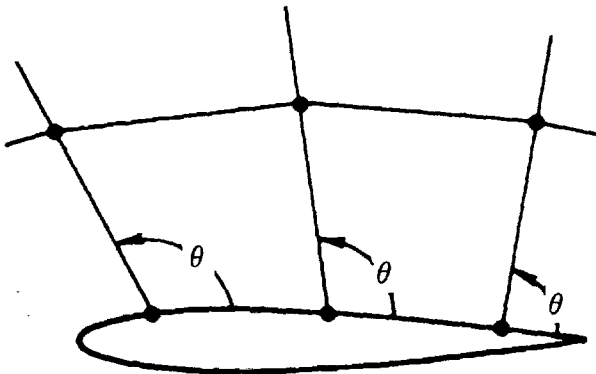
- CONTROL OF SPACING
  NEAR BOUNDARIES

The principal contribution of this work
is that the angles and spacing at the
boundaries are input. Thus, one need
not try to implement pre-determined
angles and spacing by trial-and-error
tuning of other parameters.
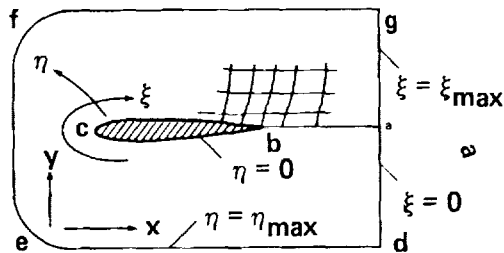
TOPOLOGY OF GRID MAPPINGS
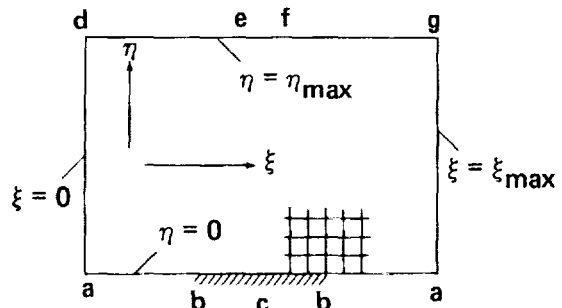

PHYSICAL SPACE

COMPUTATIONAL SPACE

O-TYPE GRIDS

PHYSICAL SPACE

COMPUTATIONAL SPACE

C-TYPE GRIDS

Topology and notation for O-type and
C-type grids are shown here. The
independent variables in the physical
space are x and y, while ξ and η are the
independent variables in the cartesian
computational space.

POISSON EQUATIONS

$$\xi_{xx} + \xi_{yy} = P(\xi, \eta)$$

$$\eta_{xx} + \eta_{yy} = Q(\xi, \eta)$$

OR, WITH DEPENDENT AND INDEPENDENT VARIABLES INTERCHANGED:

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = -J^2 (Px_\xi + Qx_\eta)$$

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = -J^2 (Py_\xi + Qy_\eta)$$

WHERE:
$$\alpha = x_\eta^2 + y_\eta^2$$

$$\beta = x_\xi x_\eta + y_\xi y_\eta$$

$$\gamma = x_\xi^2 + y_\xi^2$$

$$J = x_\xi y_\eta - x_\eta y_\xi$$

Basic to the method is that the grid transformations $\xi = \xi(x,y)$, $\eta = \eta(x,y)$ must satisfy the Poisson equations. The equations are solved with dependent and independent variables interchanged to facilitate numerical integration and the application of boundary conditions. The equations with variables thusly interchanged are sometimes referred to as the "transformed" Poisson equations.
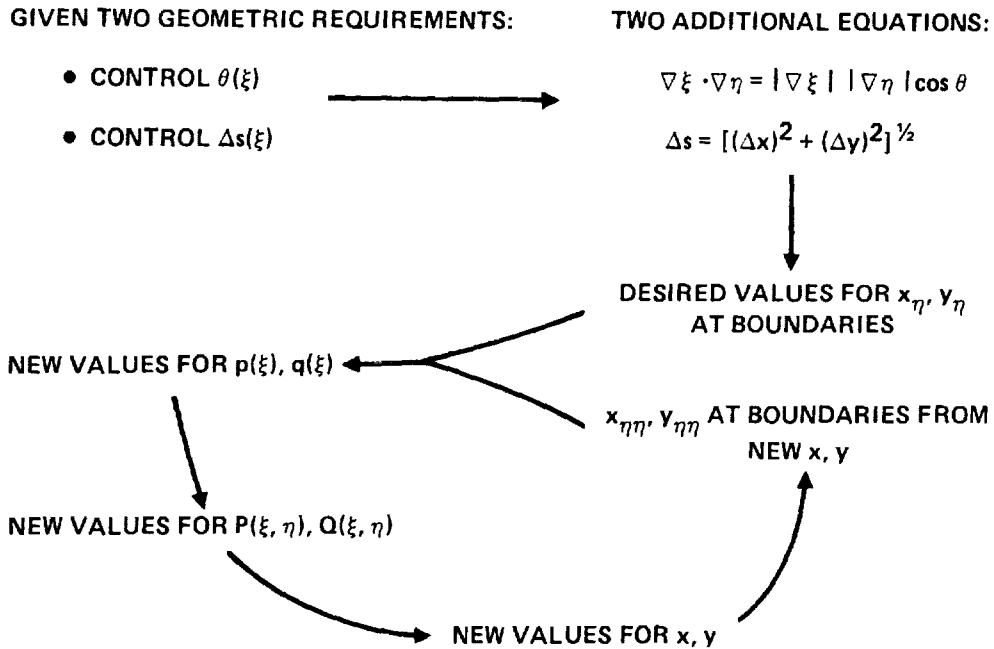
CHOICE OF INHOMOGENEOUS TERMS

$$P(\xi, \eta) = p(\xi)e^{-a\eta} + r(\xi) e^{-c(\eta_{max}-\eta)}$$

$$Q(\xi, \eta) = q(\xi) e^{-b\eta} + s(\xi) e^{-d(\eta_{max}-\eta)}$$

Inhomogeneous, or right-hand-side, or P and Q terms in the Poisson equations determine the character of the grid. Different choices for P and Q produce different grids. In this method P and Q are chosen as shown here with a,b,c, and d positive Note that the inner ($\eta = 0$) boundary $P(\xi,\eta)$ reduces to $p(\xi)$, and that at the outer ($\eta = \eta_{max}$) boundary $P(\xi,\eta)$ becomes $r(\xi)$, and similarly for $Q(\xi,\eta)$. The approach is to assume that the geometric input requirements (control of angles and spacing at boundaries) are satisfied along with the Poisson equations at the boundaries, then back-solve for $p(\xi)$, $q(\xi)$, $r(\xi)$, and $s(\xi)$. Then $P(\xi,\eta)$ and $Q(\xi,\eta)$ can be calculated for every point in the field.

ITERATIVE UPDATE OF INHOMOGENEOUS TERMS

GIVEN TWO GEOMETRIC REQUIREMENTS:

- CONTROL $\theta(\xi)$

- CONTROL $\Delta s(\xi)$

TWO ADDITIONAL EQUATIONS:

$$\nabla \xi \cdot \nabla \eta = |\nabla \xi| \, |\nabla \eta| \cos \theta$$

$$\Delta s = [(\Delta x)^2 + (\Delta y)^2]^{1/2}$$

DESIRED VALUES FOR x_η, y_η AT BOUNDARIES

NEW VALUES FOR $p(\xi)$, $q(\xi)$

$x_{\eta\eta}$, $y_{\eta\eta}$ AT BOUNDARIES FROM NEW x, y

NEW VALUES FOR $P(\xi, \eta)$, $Q(\xi, \eta)$

NEW VALUES FOR x, y

For each boundary (inner and outer) the two geometric control requirements can be re-cast as the two additional equations shown on the upper right. These two equations can be solved for the two derivatives x_η and y_η. Derivatives x_ξ, y_ξ, $x_{\xi\xi}$, and $y_{\xi\xi}$ can be found by differencing known, fixed boundary data. Derivatives $x_{\xi\eta}$ and $y_{\xi\eta}$ are found by differencing the x_η and y_η, just found, with respect to ξ. Thus, to back-solve the transformed Poisson equations for $p(\xi)$ and $q(\xi)$, two derivatives remain to be found: $x_{\eta\eta}$ and $y_{\eta\eta}$.

In the solution procedure, each iteration step is in two parts. First, the x and y from the previous iteration (or initial conditions) are differenced to find $x_{\eta\eta}$ and $y_{\eta\eta}$ at the boundaries. These are combined with all the other derivatives discussed above, which are fixed for all iteration levels, to form the transformed Poisson equations at the boundary. These are back-solved for new values of $p(\xi)$ and $q(\xi)$. Terms $r(\xi)$ and $s(\xi)$ are similarly found. New values for $P(\xi,\eta)$ and $Q(\xi,\eta)$ can then be calculated. The second part of each solution step is to perform one iteration of some solution procedure, such as SLOR. The above is iterated to convergence, producing a grid that satisfies the given geometric requirements. Inhomogeneous terms which yield the desired grid control are thus found automatically as the solution proceeds.

454

# COARSE-FINE SEQUENCING SOLUTION PROCEDURE
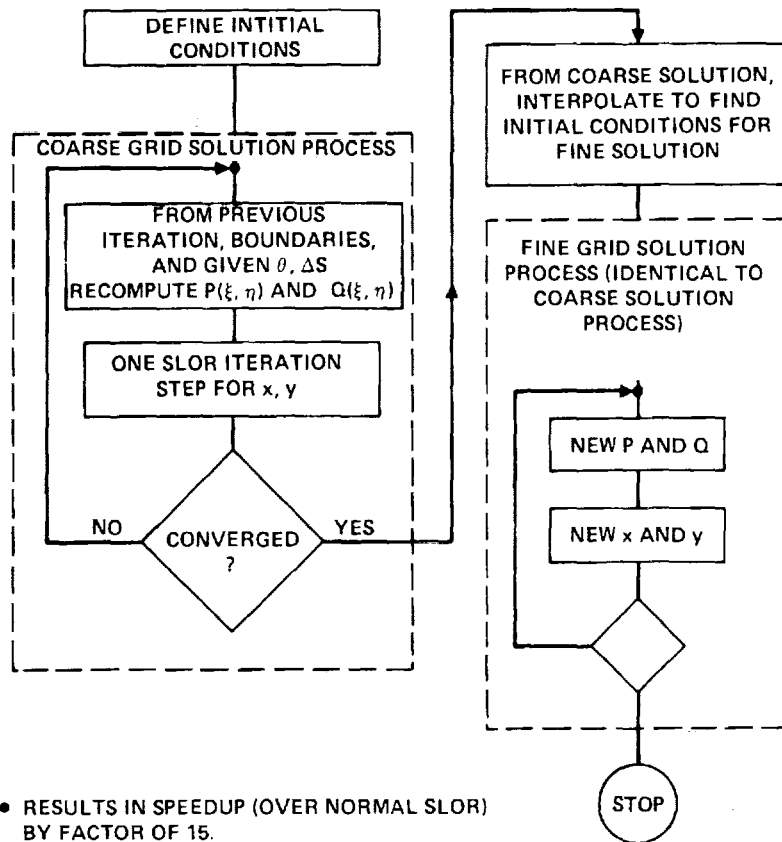
- **RESULTS IN SPEEDUP (OVER NORMAL SLOR) BY FACTOR OF 15.**

Numerical convergence is greatly accelerated by an additional feature: coarse-fine sequencing. The solution is first iterated to convergence on a coarse grid consisting of every third point in the ξ direction and every third point in the η direction. This convergence requires relatively little computer time since the amount of arithmetic being done per step is one-ninth that which would otherwise be done. The coarse solution is then interpolated to provide initial conditions for a fine solution using all of the points. Coarse-fine sequencing produces a speed-up over normal SLOR by a factor of up to 15. Grids have been generated, for simple cases, in as little as two-thirds of a second of CPU time per thousand grid points on a CDC 7600 computer, including "set-up" overhead.
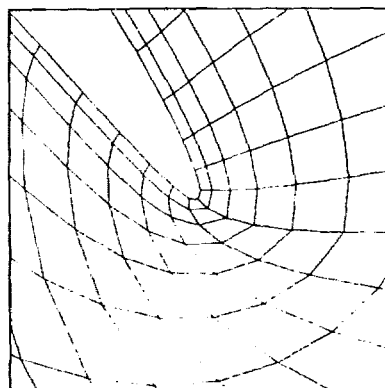
GRIDS ABOUT HIGHLY CAMBERED ELLIPSE
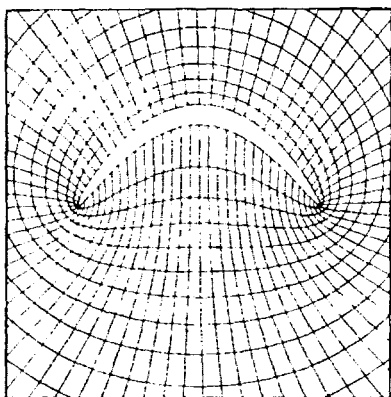
GENERATED BY
LAPLACE
EQUATION,
SHOWING POOR
CONTROL OF CELL
SIZE AND
SKEWNESS
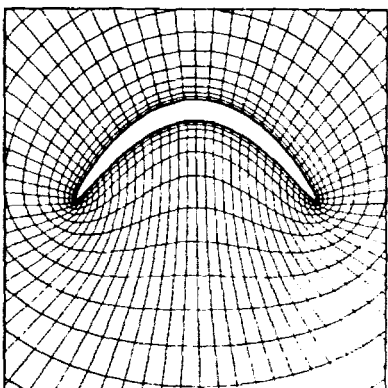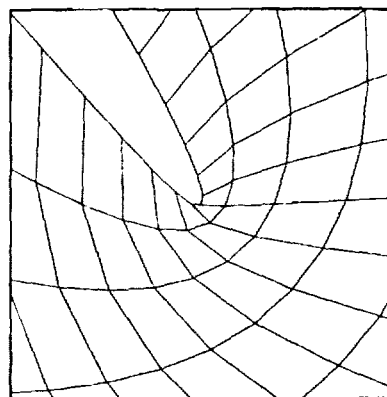
GENERATED BY
POISSON
EQUATION
WITH CONTROL
AT BOUNDARIES

The effectiveness of the grid control is demonstrated in
this comparison of two grids about a highly cambered 12:1
elliptical airfoil. In the two top figures is seen a grid
generated by the Laplace equations--like the Poisson equa-
tions but with P = Q = 0. Uncontrolled cell size and skew-
ness are clearly seen. The figures on the bottom were
generated by the present method with the grid control at
the boundaries. It was required that the lines intersect
the airfoil at 90° and that the standoff distance be 0.005
chord lengths at all points on the airfoil surface. The
angle requirement was satisfied to a tolerance of ± 0.1°
and the distance to a tolerance of ± 0.00001 chord lengths.

NEARLY SQUARE GRID CELLS ALONG SURFACE IN LEADING
EDGE REGION OF NACA 0012 AIRFOIL
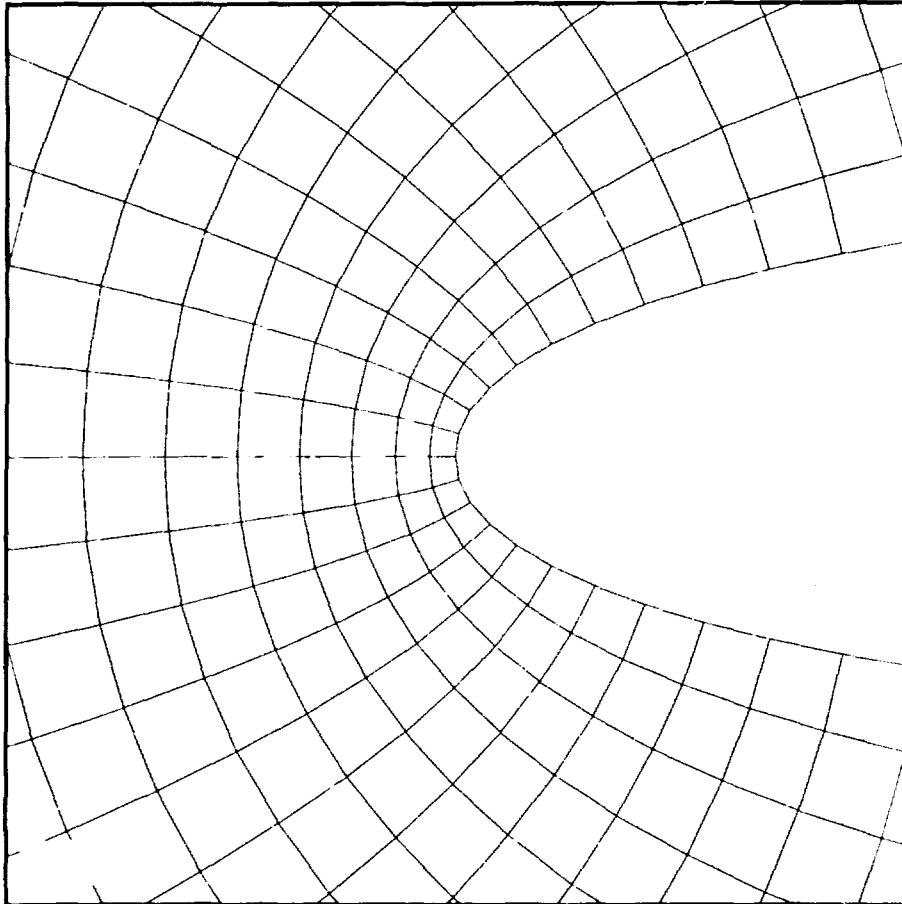
GRID SPACING <u>NORMAL</u> TO SURFACE (Δs) SPECIFIED AS
EQUAL TO LOCAL VALUE OF ARC-LENGTH
<u>ALONG</u> SURFACE

An interesting capability of this method is seen in this
close-up view of the leading edge region of a grid about
an NACA 0012 airfoil. The angle requirement need not be
90° and it need not be equal at all points on the airfoil
surface. Likewise, the standoff spacing need not be con-
stant. In this grid the angle requirement was chosen as
90° everywhere on the airfoil, but the standoff spacing
requirement (normal to the airfoil surface) was taken to
be equal to the local value of the arc-length along the
surface. This produces grid cells along the surface that
are nearly square, despite greatly varying surface dis-
tribution.

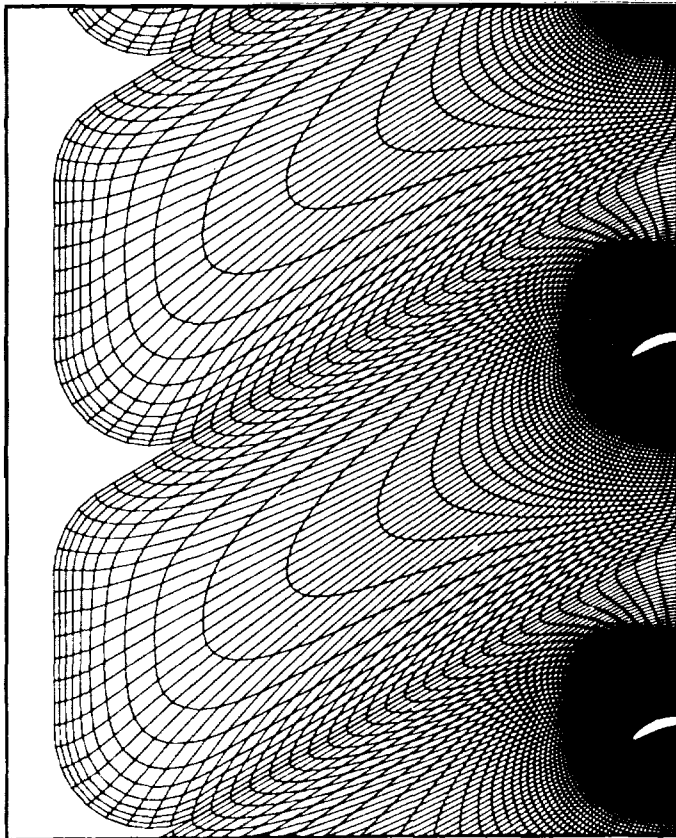# C-TYPE GRID FOR MODELING WIND TUNNEL WALLS
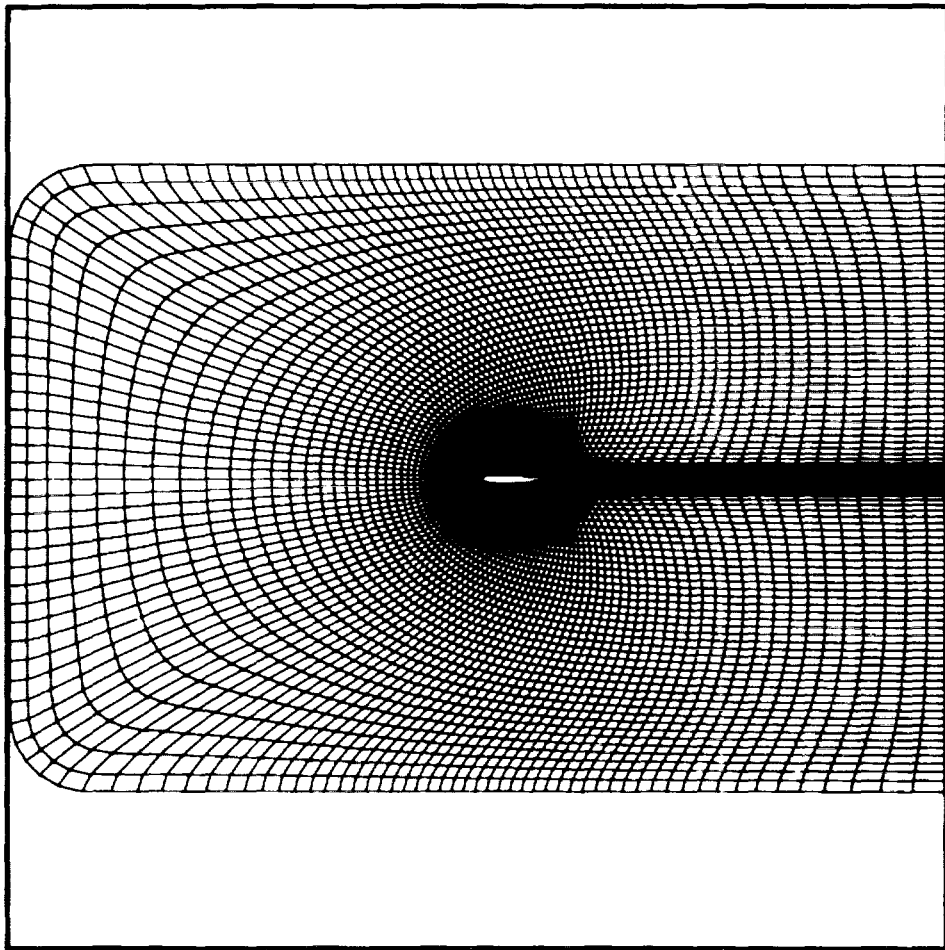
**GRID SPACING AND ANGLES CONTROLLED AT
OUTER BOUNDARY**

A C-type grid for modeling flow through
a wind tunnel is seen here. Grid spac-
ing and angles are controlled at the
outer boundary.

C-6

UPSTREAM REGION IN O-TYPE CASCADE GRID

**CONTROL OF SPACING AND
ANGLES AT OUTER BOUNDARY
ENSURES SMOOTH TRANSITION
BETWEEN CASCADE ELEMENTS**

Control of spacing and angles at outer
boundary is applied here to a cascade.
It was specified that the lines of con-
stant n which intersect the top and
bottom parts of the outer boundary of
each cascade element do so vertically.
Thus, the application of periodic bound-
ary conditions between cascade elements
is facilitated.

This same capability ensures smooth ver-
tical transition across the branch-cut
in the wake region of a C-type grid.

GRID USED IN STUDY OF BLAST WAVE ENCOUNTERING STATIONARY PICKUP TRUCK
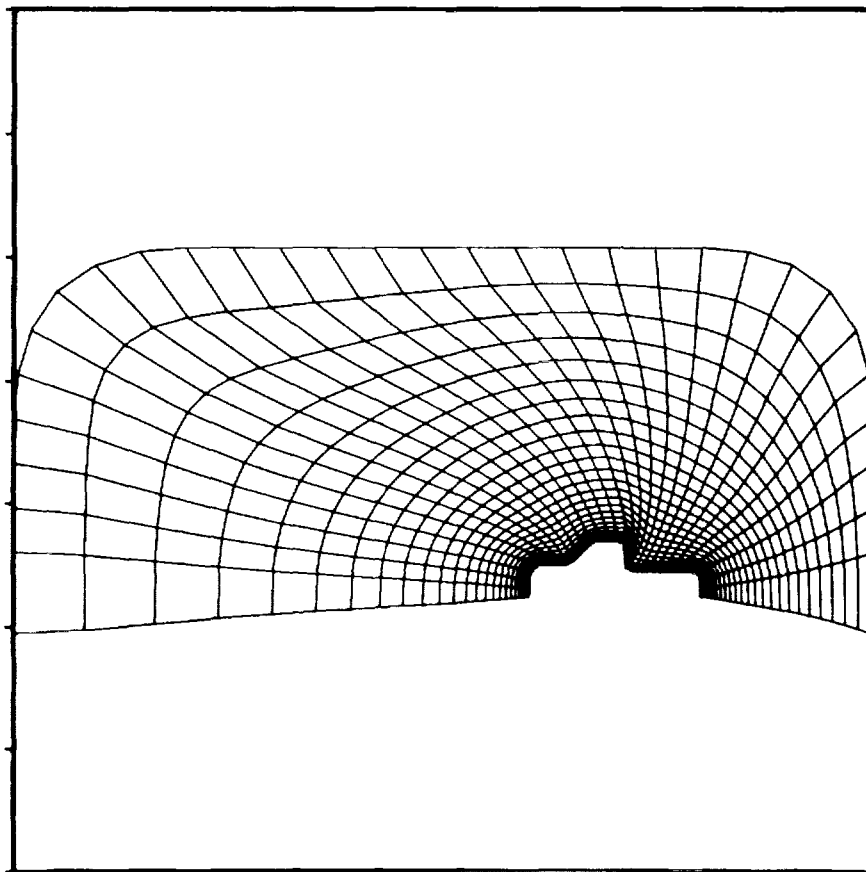
The versatility of the method is illustrated here.

FEATURES OF GRAPE (GRIDS ABOUT AIRFOILS USING POISSON'S EQUATION), A USER-ORIENTED FORTRAN COMPUTER PROGRAM

- θ, Δs ARE INPUT

- CODING IS MODULAR, GENEROUSLY COMMENTED, SYNTACTICALLY CONSERVATIVE

- BUILT-IN DEFAULT CASE, AND SIMPLE, WELL THOUGHT-OUT INPUT WHICH IS CHECKED BEFORE USE

- VERSATILE: C-TYPE OR O-TYPE, FREE-STREAM OR WIND TUNNEL OR CASCADE, VISCOUS OR INVISCID

- GRAPHICAL OUTPUT

- FAST

- WELL DOCUMENTED AND ACTIVELY SUPPORTED

A user-oriented, well documented, FORTRAN computer program, called GRAPE, has been written to employ this grid generation method. It is available from the Applied Computational Aerodynamics Branch at NASA-Ames Research Center.