6th Australasian Hydraulics and
Fluid Mechanics Conference
Adelaide, Australia, 5-9 December 1977

Transonic Computational Mesh Generation by Means of Panel Methods

P. E. RUBBERT
Head, Aerodynamic Method Branch, Boeing Military Airplane Development, Boeing Airspace Co., Seattle, U.S.A.

SUMMARY: A scheme for generating an ordered field mesh of points or cells about arbitrary geometrical shapes is presented. Panel method technology is used to solve a boundary value problem from which is extracted a mesh in automatic fashion. Grid modifications and topological features are also discussed.

1 INTRODUCTION

Computational algorithm development for the solution of nonlinear transonic and other fluid dynamics problems is advancing at a rapid rate. These algorithms typically are of finite difference/finite volume/finite element type, all of which require the construction of an ordered field mesh of points or cells prior to carrying out the hosted calculations (hereinafter, the terminology "hosted calculations" is used to refer to any such algorithm, which uses a field mesh of this type). A major factor limiting the extension of today's transonic computational algorithms to general three-dimensional applications is the problem of constructing a suitable field mesh.

In two-dimensions the problem is comparatively simple, and the approaches used have involved (1) conformal mapping to a geometrically simple domain wherein the mesh is easily constructed by hand, and (2) the solution of a vector valued Poisson equation relating physical and computational variables as pioneered by Thompson (1974). But even with these methods, 2-D mesh generation becomes complex when dealing with multi-element airfoils and other multiply connected geometries. Approaches used in three-dimensions have mostly been extensions of these two-dimensioned types, as in the work of Jameson who used a rather complicated mapping of very limited generality, and by investigators who have applied Thompson's 2-D method in a series of cross-section planes. These, to date, have not produced a mesh generation capability that can satisfactorily handle cases involving geometrically complex, three-dimensional flows.

2 REQUIREMENTS

What is needed is an automatable, simple-to-use mesh generation scheme with enough control options to produce a mesh containing the global and detailed characteristics wanted for the particular application at hand. Control can be of two types, namely (1) direct interactive control by a person who renders a judgement on the "goodness" of a particular mesh pattern based on his forecast of the global features of the yet-to-be solved hosted calculation, and (2) automated control triggered by error analysis criteria associated with the hosted calculation algorithm such that the mesh can be changed during the solution of the hosted calculation in response to the discretization errors that emerge. The control requirement, from either source, implies the ability to first establish a mesh and then to change it appropriately without undue labor or chronological flow time. It also implies a capability to display a mesh in such a way that a user can ascertain its characteristics (this is not a trivial problem in three dimensions). And finally, the mesh must be of orderly character so as to be compatible with the logical sequences of a hosted computational algorithm.

3 APPROACH

3.1 Basic Method

In the present approach, a field mesh in a prescribed domain is extracted from the solution to a linear boundary value problem associated with Laplace's equation in that domain. The user need input only the boundaries of the domain, which in a typical aerodynamics problem is comprised of the aircraft surface plus some far field boundary. The far field boundary position is selected to be compatible with whatever far field solution algorithm is to be used in conjunction with the hosted near field computational algorithm. Panel method techniques (Johnson and Rubbert, 1975), originally developed for the solution of incompressible flow problems, are used to solve a boundary value problem from which is extracted, in automatic fashion, a complete mesh subject to simple controls executed by a user.

As a first example, we pose the problem of generating a two-dimensional mesh about an airfoil, to be used in a transonic, finite difference method flow analysis. Desirable characteristics of the mesh are that: (1) the mesh density on the airfoil surface and wake should be greatest at the leading and trailing edges (regions of large gradients) and perhaps in the region where shock wave development is anticipated, and (2) the mesh should decrease in density away from the airfoil boundary as the gradients in the flow field attenuate. The boundary value problem selected as the basis for generating such a mesh is shown in Figure 1. It is easily solved using the referenced panel method. The solution is stored for subsequent extraction of the mesh.

The basis of the present method is to fabricate the mesh as the intersection points of prescribed "streamlines" and equipotential surfaces. (The word "streamlines" might better be replaced by "lines of maximum potential gradient" since these are not streamlines in the conventional sense of following an airfoil surface.) The controls used to fabricate the desired mesh are the following: (1) the circumferential mesh spacing about the airfoil surface and wake are prescribed directly by the user to satisfy whatever spacing criteria is desired on these surfaces. (2) The radial, or outward spacing is prescribed by specification of a tabulated set of values of ϕ lying between ϕ_1, the airfoil surface value, and ϕ_n, the value prescribed at the outer boundary. Control of this radial spacing is provided by the relative spacing given in the table values; thus, if dense radial spacing is desired near the airfoil surface, the elements input to the ϕ_i table ($i=1,\ldots,n$) should be bunched together near the value ϕ_1.

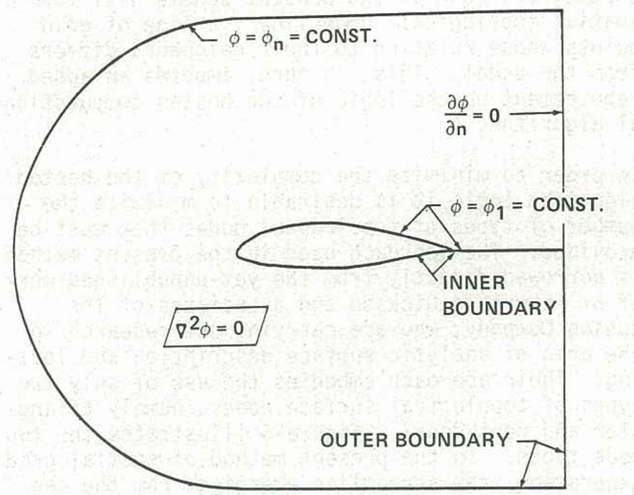

Figure 1 Boundary value problem

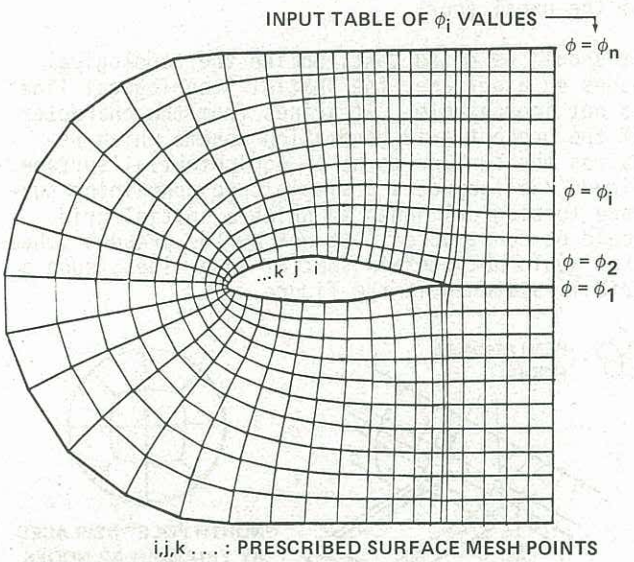

Figure 2 Automatically generated mesh

Figure 2 displays a mesh generated by this means. It displays the desired characteristics of mesh spacing, and is comprised of a logically simple, rectangular array of mesh points. In viewing the results it is apparent that the process produced, in automatic fashion, a "good" mesh from a very simple set of input data, with very little effort by the user.

In this and other examples to be given in this paper, the total number (global density) of mesh lines displayed is fewer than would normally be used to perform a hosted calculation. This is adequate to demonstrate the method and the grid and was done to avoid visual smearing of closely-spaced grids in dense regions.

The approach used above is equally applicable in three dimensions. In Figure 3 is shown a portion of a surface. The user prescribes a two-dimensional mesh pattern (s, t variables) on the surface in accordance with his relevant mesh spacing criteria. He sets up a boundary value problem analagous to that of Figure 1, solves it using a panel method, and prescribes a table of ϕ_i values to dictate the outward (n variable) spacing in complete analogy with the two-dimensional case. Streamlines are then calculated that emerge from the s, t intersections and travel in the n direction. Points along the streamlines where they pass through the tabulated ϕ_i values are identified as mesh corner points, thus completing the definition of the mesh as an ordered set of (s,t,n) coordinates.

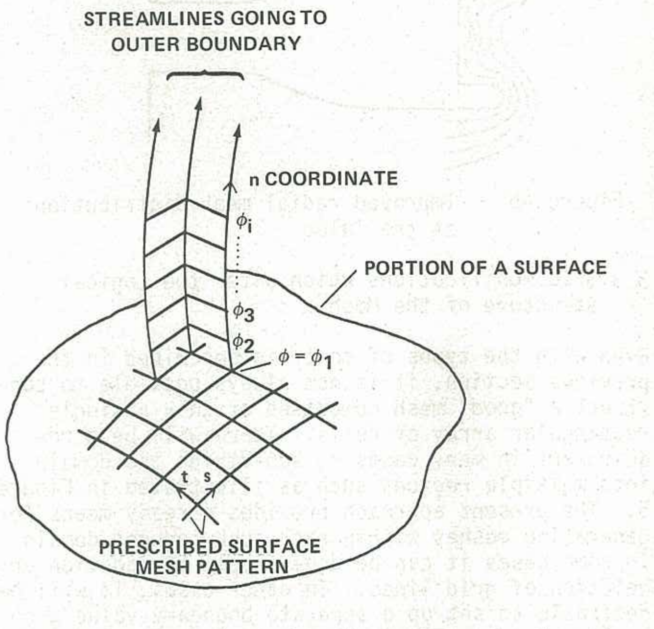

Figure 3 Portion of a three-dimensional mesh

3.2 Simple Grid Improvements Not Affecting the Logical Ordering of a Mesh

There is no a priori guarantee that a mesh thus produced will display all of the desired characteristics, so additional degrees of control are needed. Figure 4a shows a mesh thus calculated about a nacelle which displays some deficiencies. One deficiency is that the outward, or n, spacing is too coarse in the region near the surface between the points S_i and S_j shown in Figure 4a.

This is easily cured as follows. A new table of ϕ_i values is prescribed along the streamline emanating from S_j in the n direction to produce a more dense mesh spacing in n near the nacelle surface. A smooth interpolation function (a number of these can be built into the code) is then applied between the original n variable spacing (the original table of ϕ_i values) emanating from S_i and the new spacing (the new ϕ_i table) prescribed along n emerging from S_j. The result is the mesh shown in Figure 4b, an obvious improvement.

A second type of control, not yet implemented, may at times be desired to alter the lateral spacing of adjacent n lines away from the surface. In Figures 4a,b the n lines appear to be bunched too closely together in the region in front of the inlet. This type of control can be achieved by the addition of source-type Poisson terms to the linear differential equation being solved.

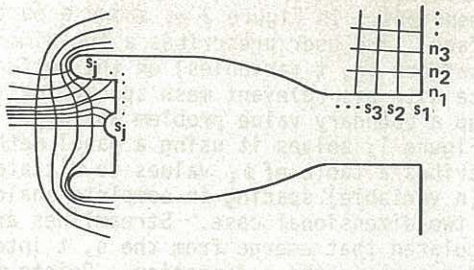

Figure 4a A deficient mesh about a nacelle

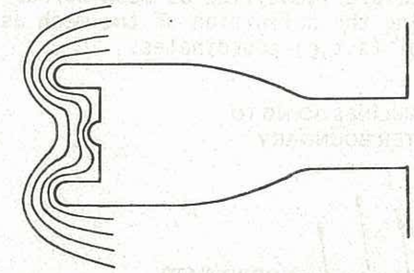

Figure 4b Improved radial mesh distribution at the inlet

3.3 Grid Modifications which alter the Logical Structure of the Mesh

Even with the types of controls described in the previous section, it is not always possible to construct a "good" mesh comprised of only a single rectangular array of cells. There will be a requirement in many cases to sub-divide the domain into multiple regions such as illustrated in Figure 5. The present approach provides a ready means for generating meshes within each such reduced domain. In some cases it can be done simply by addition or deletion of grid lines. In other cases, it will be desirable to set up a separate boundary value problem, analogous to that of Figure 1, in each reduced domain.

A decision to subdivide the domain into multiple regions imposes additional requirements on the hosted computational algorithms. They must be able to correctly "match" or transmit the hosted computation across domain boundaries where the grids do not interface in a one-on-one fashion. This is a problem to be addressed in the context of the hosted computational scheme and is thus beyond the scope of this paper. (Several procedures (Magnus and Yoshihara, 1970 and Chen, Dickson and Rubbert, 1977) for matching or transmitting solutions across domain boundaries have been used successfully with transonic finite difference problems.)

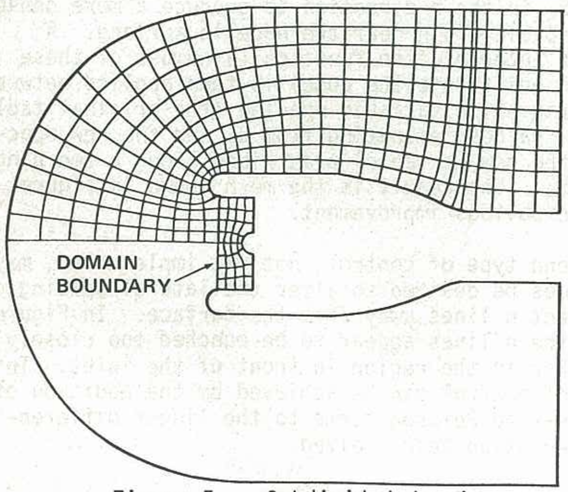

Figure 5 Subdivided domain

3.4 Topological Features

Other features that must be dealt with, particularily in three-dimensional mesh generation, are those imposed by the topological character of the boundary surfaces. It can be proved that the closed surface of a non-toroidal three-dimensional object cannot be covered everywhere by a simple, rectangular surface grid. The grid must exhibit "poles" (as at the poles on a world globe), or other non-standard "topological" nodes at one or another points on the surface. Those spatial grid points lying along a streamline emanating from a surface topological node in the present scheme will form a spatial topological "node line", a line of grid points whose relation to their neighbors differs from the usual. This, in turn, imposes an added requirement on the logic of the hosted computational algorithm.

In order to minimize the complexity of the hosted algorithm logic it is desirable to minimize the number of types of topological nodes that must be provided. The approach used in the present method is borrowed directly from the yet-unpublished work of Dr. Lawrence Dickson and associates of The Boeing Company, who are carrying out research in the area of analytic surface description and lofting. Their approach embodies the use of only two types of topological surface nodes, namely triangular and pentagonal. Figure 6 illustrates the two node types. In the present method of spatial grid generation, the streamline emerging from the center of a node would form a spatial node line where three or five surrounding grid cells meet instead of the usual four.

It should be noted that, unlike the topological nodes on a surface, the spatial "topological line" is not inescapable. It arises from the character of the present grid generation scheme which requires the surface to be an equipotential surface. Figure 7 illustrates a simple case containing surface topological nodes in which a spatial grid could be constructed (but not by the present scheme) that would not contain spatial node lines; such a grid is sketched in the figure.

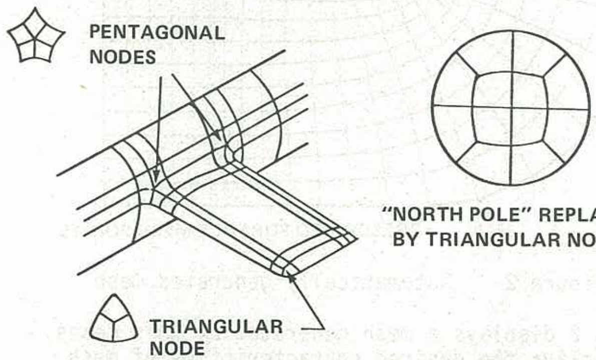

Figure 6 Topological nodes

A second type of topological phenomenon occurs in multiple-body problems. Figure 8 illustrates a mesh generated by the present method for the problem of two airfoils. Note the occurrence of two five-sided cells which are shaded in the figure. The location of such odd cells and the global character of the mesh can be adjusted at will by changing the boundary conditions, but the presence of such cells must be accounted for in the logic of the hosted algorithm. They could be eliminated in this case by setting the potentials of the two airfoils equal to one another, but the resulting mesh grid would not be a desirable one for transonic finite difference computations.

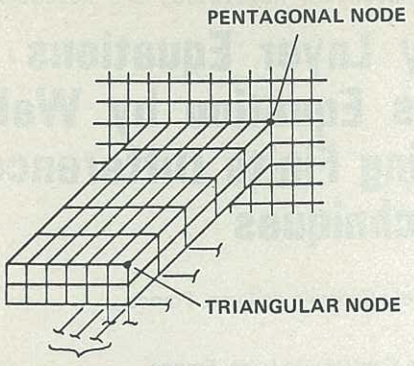

Figure 7 Regular spatial grid with surface topological nodes

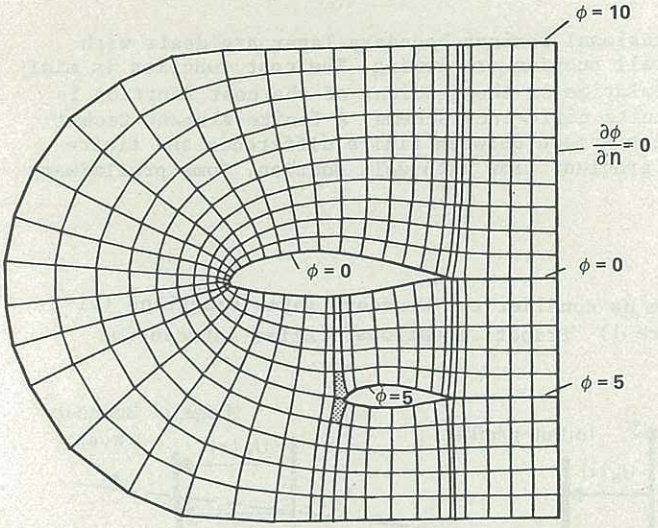

Figure 8 Topological cell in a multiple body problem

4 CONCLUDING REMARKS

The problem of generating an ordered field mesh of points or cells about arbitrary geometrical shapes has been addressed. The viewpoint taken has been that the method of mesh generation must be automatable and easily controllable by a user having an a priori conception of what constitutes a "good" mesh. The approach used was to generate the mesh as the solution to a Laplace boundary value problem, a simple, automatic process with today's panel methods. The conclusions reached were:

1. Mesh generation must be treated as an iterative, interactive process wherein a mesh is generated and displayed, judgements rendered, and modifications commanded, executed and displayed until a mesh judged to be satisfactory is produced.

2. The present method provides the ease-of-use and flexibility necessary to (a) generate a "first cut" mesh displaying most, and sometimes all of the characteristics desired, and (b) to perform commanded mesh modifications improving the character of the mesh.

3. In many cases a simple rectangular mesh is unsuitable for the entire domain; the domain must be subdivided into multiple sub-regions, each containing a rectangular mesh with interfaces along the subregion boundaries. Phenomena requiring subdivision include:

   (i) Geometrical shapes requiring subdivision to achieve desired mesh densities.

   (ii) The topological character of the body surface geometry and the requirements for surface mesh layouts to follow the natural lines of the object. Topological surface nodes create topological lines in the mesh field in the present method.

   (iii) Most multiple body problems.

4. The requirements of the preceding paragraph imply a close coupling between mesh generation options and the logical structure of the hosted computational algorithm. The two are interwoven! It is clear that the hosted computational algorithm logic must be structured to handle the logical irregularities imposed by the character of the mesh.

5 REFERENCES

CHEN, A.W., DICKSON, L.J., and RUBBERT, P.E. (1977). A far field matching method for transonic computations. AIAA Paper No. 77-208.

JOHNSON, F.T. and RUBBERT, P.E. (1975). Advanced panel-type influence coefficient methods applied to subsonic flows. AIAA Paper No. 75-50.

MAGNUS, R. and Yoshihara, H. (1970). Inviscid transonic flow over airfoils, AIAA Journal, Vol. 8, No. 12, PP. 2157-2162.

THOMPSON, J.F., THAMES, F.C., and MASTIN, C.W. (1974). Automatic numerical generation of body-fitted curvilinear coordinate system for field containing any number of arbitrary two-dimensional bodies, J. of Comp. Phys. Vol. 15, No. 3, PP. 299-319.