

AD-A127 800

THEORETICAL INVESTIGATIONS OF FLOW FIELDS AND HEAT
TRANSFER IN MODERN GAS..(U) MASSACHUSETTS INST OF TECH
CAMBRIDGE GAS TURBINE AND PLASMA D.. E E COVERT ET AL.

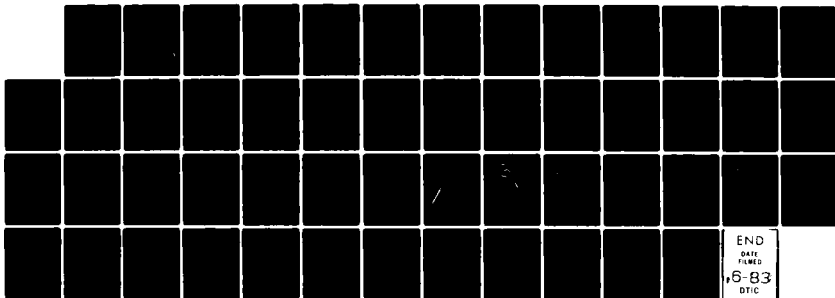
1/1

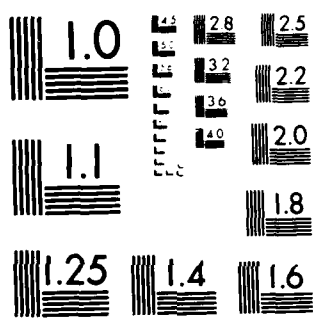
UNCLASSIFIED

31 DEC 82

F/G 20/4

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

ADA 127800

(12)

GAS TURBINE AND PLASMA DYNAMICS LABORATORY
DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MASSACHUSETTS 02139

ANNUAL REPORT NO. 2

on

Contract N00014-81-K-0024

entitled

PART 1: THEORETICAL INVESTIGATIONS OF FLOW FIELDS
AND HEAT TRANSFER IN MODERN GAS TURBINES

prepared for

DEPARTMENT OF THE NAVY
OFFICE OF NAVAL RESEARCH
ARLINGTON, VA 22217
ATTN: MR. KEITH ELLINGSWORTH
MECHANICS DIVISION

PERIOD OF
INVESTIGATION: JANUARY 1, 1982 - DECEMBER 31, 1982

PRINCIPAL INVESTIGATORS:

PROF. EUGENE E. COVERT
PROF. ALAN H. EPSTEIN
PROF. WILLIAM T. THOMPSON, JR.

DTIC
ELECTE
MAY 9 1983
S A D

DTIC FILE COPY

88 04 26 105

This document has been approved
for public release and sale; its
distribution is unlimited.

PART-I MIT/RR/ONR COMPUTATIONAL PROJECT

During the last reporting period most of the effort has been directed to assembling the dedicated computer facility, integrating its use with computational codes, and developing the first version of the high resolution two dimensional Navier-Stokes code. Secondary amounts of effort have gone into algorithm development for the three-dimensional viscous code and defining an improved two dimensional code.

Accession

NRLS

1. 1. 1.

2. 2. 2.

3. 3. 3.

4. 4. 4.

5. 5. 5.

6. 6. 6.

7. 7. 7.

8. 8. 8.

9. 9. 9.

10. 10. 10.

11. 11. 11.

12. 12. 12.

13. 13. 13.

14. 14. 14.

15. 15. 15.

16. 16. 16.

17. 17. 17.

18. 18. 18.

19. 19. 19.

20. 20. 20.

21. 21. 21.

22. 22. 22.

23. 23. 23.

24. 24. 24.

25. 25. 25.

26. 26. 26.

27. 27. 27.

28. 28. 28.

29. 29. 29.

30. 30. 30.

31. 31. 31.

32. 32. 32.

33. 33. 33.

34. 34. 34.

35. 35. 35.

36. 36. 36.

37. 37. 37.

38. 38. 38.

39. 39. 39.

40. 40. 40.

41. 41. 41.

42. 42. 42.

43. 43. 43.

44. 44. 44.

45. 45. 45.

46. 46. 46.

47. 47. 47.

48. 48. 48.

49. 49. 49.

50. 50. 50.

51. 51. 51.

52. 52. 52.

53. 53. 53.

54. 54. 54.

55. 55. 55.

56. 56. 56.

57. 57. 57.

58. 58. 58.

59. 59. 59.

60. 60. 60.

61. 61. 61.

62. 62. 62.

63. 63. 63.

64. 64. 64.

65. 65. 65.

66. 66. 66.

67. 67. 67.

68. 68. 68.

69. 69. 69.

70. 70. 70.

71. 71. 71.

72. 72. 72.

73. 73. 73.

74. 74. 74.

75. 75. 75.

76. 76. 76.

77. 77. 77.

78. 78. 78.

79. 79. 79.

80. 80. 80.

81. 81. 81.

82. 82. 82.

83. 83. 83.

84. 84. 84.

85. 85. 85.

86. 86. 86.

87. 87. 87.

88. 88. 88.

89. 89. 89.

90. 90. 90.

91. 91. 91.

92. 92. 92.

93. 93. 93.

94. 94. 94.

95. 95. 95.

96. 96. 96.

97. 97. 97.

98. 98. 98.

99. 99. 99.

100. 100. 100.

DEDICATED COMPUTER FACILITY

The dedicated computer facility has been completed, and consists of a Perkin-Elmer 3242 host computer with 4 MBYTES main memory, a Floating Point Systems Array Processor, model AP-120B, and a bulk memory system supplied by DATARAM. A block diagram for the system is shown in Figure 1. The bulk memory system is the last hardware item purchased and consists of 32 MBYTES of MOS semiconductor memory in a single unit. This system will be used to store the solution matrix for the three-dimensional viscous code and as a data buffer for the blowdown turbine data acquisition system.

All system components are working as expected with concurrent processing of finite difference solutions on the array processor, turbulence modeling in the host processor, and bulk memory transfers. The interface between the Perkin-Elmer host and the bulk memory device has achieved a transfer rate of 4 MBYTES/sec which is adequate for the computational and experimental projects. The system allows an average computational rate of approximately 3 million floating point operations per second to be achieved which is quite close to the expected computation rate. At present the original reference grid of 50 x 50 x

100 node points fits comfortably in the bulk memory. The maximum grid size which can be run is nearly 350,000 node points, which is about 40% larger than the reference grid. These possible extra node points will be used to increase resolution in shear flow regions.

TWO-DIMENSIONAL VISCOUS CODE DEVELOPMENT

Summary

The first version of the two dimensional, Reynolds averaged, Navier-Stokes code has been completed. An algebraic turbulence model of the Cebeci-Smith type is used for turbulence modeling. The present code uses an unusual finite difference grid which is not simply connected. The grid structure enforces periodicity through an interconnection table rather than having grid lines running in the periodic direction. Use of this grid structure allows a better physical space grid, but at the expense of more complication in code structure, inflow/outflow boundary conditions, and smoothing. While the code with simple sheared grids had become reasonably robust, it is difficult to produce starting solutions for the present code, and some convergence problems now exist when running with large time steps.

Calculations for the T-7 and ACE turbine cascades and a supercritical compressor cascade using the new grid are now being evaluated. Approximately 500 iterations are required for convergence on a 50 x 100 grid using full turbulence models, and this type case requires about 3 hours to complete. When using the non-simply connected grids only a portion of the calculations is presently implemented on the AP (array processor). When coding for this grid type is completed, all floating point calculations, except those in the turbulence model, will be moved to the AP and run times should come down to about 1/2 hour. Present results for the laminar, design incidence, T-7 case are encouraging, but the off-design incidence case and the supercritical compressor results are poor. The sources of these problems are felt to be understood and a new version of the two-dimensional code is being prepared.

DESCRIPTION OF TWO-DIMENSIONAL VISCOUS CODE - FANSI

The present two-dimensional viscous code is a generalization of an algorithm proposed by Beam and Warming [1] for Reynolds averaged, Navier-Stokes equations. The original algorithm has been modified to increase accuracy of computed solutions and decrease total run time. Major modifications are in the areas of inflow/outflow boundary conditions and flux balance approximations. The present algorithm is essentially a finite volume approximation to the steady state equations coupled to a finite difference, time marching integration in time. Time accurate solutions can be obtained, and acceleration to steady state is possible through local time or pseudo-time calculations. If only steady solutions need to be computed, considerable further reductions in run time could be obtained through approximations to the time-marching algorithm.

If we consider the Navier-Stokes equations to be expressed in vector form as:

$$\hat{U}_{,t} + \hat{L}_h(U) = \hat{L}_v(U) \quad (1)$$

where $U = [\rho, \rho u, \rho v, \rho E]$ is the state vector

$$\hat{L}_h(U) = \hat{L}_{hx}(U) + \hat{L}_{hy}(U) \quad (2)$$

represents the inviscid terms or the Euler equations

and

$$\hat{L}_v(U) = \hat{L}_{vx}(U) + \hat{L}_{vy}(U) \quad (3)$$

represents the viscous terms

If we define a Navier-Stokes operator,

$$\hat{L}_{ns}\{U\} = \hat{L}_v\{U\} - \hat{L}_h\{U\} \quad (4)$$

we can write the equations as

$$\hat{U}_{,t} = \hat{L}_{ns}\{U\} \quad (5)$$

The present version of FANSI approximates these equations by either a first order or a second order accurate, in time, approximation expressed as:

$$[I - \theta \Delta t L'_{nsx}][I - \theta \Delta t L'_{nsy}]\{\Delta U^n\} = \Delta t L_{ns}\{U^n\} \quad (6)$$

where L_{ns} is a finite difference operator approximating the steady state equations

L'_{nsx} and L'_{nsy} are linearized approximations to the x and y Navier-Stokes operators

$\theta = 1/2$ for backward Euler and 1 for trapezoidal, implicit
and

$$\Delta U^n = U^{n+1} - U^n$$

For steady state calculations, the backward Euler form is used, and for time accurate calculations the trapezoidal form is recommended. Details of the time marching algorithm, or the left hand side solution of equation (4), are as described in reference [2]. This approximation does not use the explicit mixed derivative terms as suggested in

reference [1], but these terms are easily added if cases in which they are important are encountered.

The L_{ns} finite difference operator uses the grid distribution layout illustrated in Figure 2. The approximation at a node point j,k uses information from the 8 adjacent nodes in a bi-linear approximation for flux vectors at these points. These flux vectors are then summed using trapezoidal integration in physical space. The time marching algorithm uses only central differencing operations on the real node points to compute the new state vector at node j,k . This type of approximation in which the time marching algorithm has a different grid star and approximation procedure than does the steady state solution operator has proved very flexible and powerful.

GRID GENERATION EFFORTS AND RESULTS

Accurate prediction of the flow behavior in a turbomachinery blade row depends both on the flowfield prediction program and the calculation grids used. The accuracy of a good flowfield prediction program can be significantly reduced if the grid generator does not give sufficient grid resolution in regions where flow conditions change rapidly. These regions include the leading and trailing edges of the profile, and the profile boundary layer. This inter-dependency of flowfield prediction programs and grid generators has required the development of a grid generation routine capable of providing the required resolution. This work is being carried out in parallel with the development of the two- and three-dimensional flowfield prediction routines.

This section provides a brief history of the stages passed in reaching the latest grid generator code. This latest code should not be taken as the perfect solution to the problem, but as the best available at the moment. Refinement and generation of new types of grid is expected to continue well into the future.

Boundary Orthogonal Grid Generator (BOGG)

This grid generation routine was developed under the restrictions that :

- a) One set of grid lines should be normal to the blade surfaces. This was required first so that the thin shear layer approximations in the 2-D flowfield prediction code were correctly applied. These assumptions were basically the standard boundary layer equation assumptions, and in order to be valid the grid lines had to be normal to the boundary layer. Secondly this normality at the wall is also important for turbulent boundary layer modeling, where the turbulent

conditions inside the boundary layer must be related to those at the edge of the boundary layer.

- b) The grid lines had to be periodic along the cascade periodic boundaries. If this were not so, some interpolation would have to be undertaken to match the boundary values, and this could lead to significant errors. This non-periodicity may also result in reductions in the stability limit of the code used.
- c) The grid had to be used in an implicit code whose structure specified that any grid node had to be surrounded by four other grid nodes.

A viscous grid for a compressor cascade is shown in Figure 3. It can be seen that these goals have been achieved. This type of grid, although satisfying the above criteria, did have disadvantages as detailed below:

- i) High rates of change of grid shear resulted in total pressure errors in the inviscid free stream. The rapidly changing shear of the grid is readily apparent in this figure.
- ii) This type of grid could not be used for profiles with blunt leading or trailing edges (e.g. turbine profiles) because it is not possible to specify normality of the cross-passage grid lines at the leading edge and still be able to produce throughflow grid lines.

In order to provide good leading edge resolution for blunt leading edge profiles, it was necessary to produce C-type grids. A typical C-type grid is shown in Figure 4. Here the grid lines normal to the

profile surface at the leading edge are extended upstream and around to the other leading edge. If a blunt trailing edge were present, a similar scheme would be used along the exit boundary. This scheme does provide normality of grid lines at the leading edge but suffers from the following disadvantages:

- i) This type of grid still has high rates of change of grid shear.
- ii) C-type grids of this type result in poor inlet (and exit if a blunt trailing edge is present) resolution. This is because the spacing of the grid lines at the inlet is closely linked to the spacing on the profile leading edge. If the inlet were to be extended further forward, the distance between grid nodes would increase. The inlet grid resolution cannot, therefore, be controlled separately from the profile leading edge resolution.

Staggered Grids

The BOGG generated grids were suitable for the early compressor cascade predictions, but the high total pressure losses occurring in turbine predictions resulted in the need for a grid which reduced these losses. It was found that simple staggered grids reduced inlet stagnation pressure loss, which led to their use with inviscid and laminar calculations. Staggered grids are those where the cross-passage lines have the same X-location over the whole of their length. A typical viscous staggered grid for a turbine geometry is shown in Figure 5.

The predicted flowfield behavior using this type of grid was significantly better than that with the BOGG grids due to the reduction in inviscid total pressure losses. This type of grid still had several disadvantages:

- i) The grid lines are no longer normal to the profile surface. This is shown well in Figure 6, which is an enlargement of the leading edge region of the grid shown in Figure 5. The grid lines are now inconsistent with the thin shear layer approximations, but the 2-D code has been converted to full Navier-Stokes form, which has removed the necessity for grid normality to the profile surface for laminar calculations.
- ii) At the trailing edge (shown enlarged in Figure 7), the major problem is that the grid "wake" does not follow the real wake if the trailing edge circle is fully defined (if a cusp were used the grid wake could be put in the correct location, but the use of a cusp would most probably result in incorrect modeling of the flow in the vicinity of the trailing edge). This disagreement of grid and real wake positions means that the wake resolution is not where it is required.
- iii) Non-normality of the grid lines at the profile surface makes implementation of an accurate turbulence model more difficult, as some interpolation may be necessary to determine the correct edge conditions for any point in the boundary layer.

In order to predict turbulent boundary layer flows it was therefore necessary to produce a new type of grid which would have grid lines normal to the surface, but would also have all the good properties of the two previous types of grid (i.e. low inviscid total pressure losses).

Viscous Orthogonal Grids

Before discussing the new types of grid, the goals of the grid generation project should be detailed. These were:

- i) Provide good boundary layer and wake resolution.
- ii) Grid lines must be normal at the walls (for turbulence modeling).
- iii) Ability to provide good leading and trailing edge resolution.
- iv) Grid cells must be as close to rectangular as possible, with no large rates of change of grid shear.
- v) Flexible grid resolution upstream and downstream of the profile.
- vi) Periodicity of grid lines in the inlet and outlet regions.
- vii) Grid must be used with an implicit algorithm.

The grid production routine starts with an X-Y array of profile coordinates defined on the profile suction and pressure surfaces. Grid inlet and exit angles are also defined. These should correspond closely to the profile inlet and exit metal angles since the grids produced are essentially quasi-streamlines, and hence want to follow the expected flow directions.

These X-Y coordinates are then interpolated to 1000 points per surface. The through passage lines, or level lines, are then generated. The boundary layer level lines are generated using simple correlations for laminar or turbulent boundary layers. The user may specify either laminar or turbulent boundary layers on either of the surfaces. The number of grid lines in the boundary layer is also user specified. The profile wake is then merged into the surface boundary layers. Once the specification of the boundary layer level lines is finished, the

"inviscid" level lines are generated by using the BOGG grid generator. The number of inviscid level lines and their spacing distribution are specified by the user. This technique produces smooth level lines which follow the boundary layer edge.

The cross-passage grid lines are then generated using a construction technique, marching from the mid passage level line outwards towards both surfaces. It was necessary to relax orthogonality of the grid lines in the vicinity of the leading and trailing edges. Cross-passage grid lines are automatically generated to ensure periodicity on the inlet and exit periodic boundaries, and the user may manipulate the orthogonals so that they end at any specified location.

A sample of the type of grid initially produced with this grid generator is shown in Figure 8. An enlargement of the leading edge of this grid is presented in Figure 9. It is obvious in this figure that close to the leading edge "grid" stagnation point, orthogonality of the grid lines is relaxed. This was done to ensure that fine resolution could be obtained in this region. Although this does not satisfy one of the generation goals, the dropping of orthogonality is not important in this region since the boundary layer is normally laminar, and the full Navier-Stokes equations are solved.

An enlargement of the trailing edge is shown in Figure 10. Again, the orthogonality condition has been dropped in this region. However, although the grid lines are no longer normal to the surface, they are close to normal to the separated boundary layer streamlines.

The above grids were produced for a turbine profile. This generator can also be used for compressor profiles, and a sample grid is shown for a supercritical compressor in Figure 11.

Although this grid satisfies the goals detailed earlier, it still

has some disadvantages:

- i) The total number of grid nodes is greater than either the BOGG or staggered grids.
- ii) The resolution in the vicinity of the leading and trailing edges results in resolution in regions across the passage where it is not really needed.
- iii) The upstream and downstream grid lines are no longer periodic with themselves (as in the BOGG and staggered grids). This results in the need for a grid end connection table. This increases the complexity of the 2-D flowfield prediction routine.

Turbulence Modeling Status

An algebraic turbulence model of the Cebeci-Smith, reference [3], type is presently used for turbulence modeling. This model includes sections for boundary layers, near wake regions and far wake regions. The model is implemented using the vorticity formulation suggested in reference [4].

DISCUSSION OF PRESENT RESULTS

Present results for the T-7 cascade, ACE cascade, and a supercritical compressor cascade are of mixed quality. Design incidence results for T-7 and ACE appear good while the supercritical stator result is poor. The off-design results for T-7 show that a leading edge, pressure surface separation is missed. On balance it appears that the non-simply connected grids discussed earlier provide adequate resolution of flow feature, but we must be much more sophisticated in our use of the grids. Throughout the discussion of results, areas will be examined where the grids, or our handling of the grids, has resulted in problems.

An example of our good results is provided by the design incidence T-7 cascade calculations for which the finite difference grid was shown

DISCUSSION OF PRESENT RESULTS

Present results for the T-7 cascade, ACE cascade, and a supercritical compressor cascade are of mixed quality. Design incidence results for T-7 and ACE appear good while the supercritical stator result is poor. The off-design results for T-7 show that a leading edge, pressure surface separation is missed. On balance it appears that the non-simply connected grids discussed earlier provide adequate resolution of flow feature, but we must be much more sophisticated in our use of the grids. Throughout the discussion of results, areas will be examined where the grids, or our handling of the grids, has resulted in problems.

An example of our good results is provided by the design incidence T-7 cascade calculations for which the finite difference grid was shown in Figure 8. The grid has 100×49 points and is nearly orthogonal at each node point. Predicted results are quite sensitive to the grid resolution. The turning is 126 degrees with an outflow Mach number of 0.75. The Reynolds number is 680000.

The computational results are presented in Figures 12 through 15. The first figure compares predicted blade surface pressure to the experimental values. Reasonably accurate surface pressures were predicted everywhere including the stagnation point and the trailing edge. We are as yet unable to explain the discrepancy at 70% chord on the suction surface, although we believe it to be linked to an overexpansion of the pressure surface flow around the trailing edge.

The predicted trailing edge flowfield is illustrated in Figure 13 in terms of velocity direction vectors. This calculation shows one small separation zone comparable in size to the trailing edge radius and a migration of suction surface boundary layer flow toward the pressure surface. The velocity pattern here was found to be extremely sensitive to grid resolution. While the predicted flowfield at the trailing edge seems reasonable, we have no way to verify its accuracy.

One major concern when developing the present grids was the effect of the closely spaced throughflow lines in the upstream region. Figures

14 and 15 show the effect to be relatively small, but not necessarily insignificant. Figure 14 is a Mach number contour plot, and Figure 15 is a stagnation pressure error contour plot for the upstream domain. Mach number contours are quite reasonable, and the maximum stagnation pressure error is about 1%. This error probably results from smoothing algorithms and from difference cell definitions on the upstream boundary edges.

The problems with the T-7 results show up quite clearly in the off-design cases as illustrated in Figure 16, which is a blade surface static pressure comparison with experimental results. The experimental results clearly illustrate that the flow separates on the pressure surface near the leading edge. The computation shows no separation and appears to give nearly an inviscid pressure distribution here. This separation might be predicted if more than 3 streamwise node points were in the separation zone, but it proved quite difficult to provide adequate grid resolution in this zone. We regard this result to be a serious failure, and it is the prime driving force behind the major coding changes to FANSI which are in progress.

The second serious problem encountered when implementing the new grids is illustrated in Figures 17 and 18. These figures are a Mach number contour plot and a stagnation pressure error contour plot for the supercritical stator with full turbulence modeling. The predicted suction surface boundary layer and wake are far too thick, and a completely non-physical Mach disturbance occurs upstream which is linked to a 8% stagnation pressure error along the upstream inflow boundary.

The non-simply connected grids require that the line A-B-C in Figure 18 be treated as an inflow boundary and line D-E-F be treated as an outflow boundary. Two problems arise in such a treatment. First, a computational inflow or outflow boundary must actually be a physical inflow or outflow boundary. In the supercritical case, it appears we violated this fundamental constraint by generating the grid lines such that line B-C is aligned with the far upstream flow angle. Locally this boundary is then either an inflow or an outflow boundary depending on

the local flow angle. It appears as if a simple regridding will improve the supercritical results.

A second more fundamental difficulty with the non-simply connected grid in an implicit code is that A-B lies along a computational y coordinate line and B-C lies along a computational x coordinate line. The present version of FANSI uses explicit boundary conditions along these boundaries rather than the implicit conditions which were used with the original staggered grids. As a result, stability constraints associated with boundary conditions have been reintroduced into FANSI. Because of these problems, we have decided to produce an interim version of FANSI for use with only staggered grids, but having the fully implicit boundary condition capabilities restored. The interim version will have consistent, characteristic inflow/outflow boundary conditions. This version plus the algebraic turbulence model and test cases will constitute the Ph.D. thesis for R. H. Bush, who expects to finish by June 1983.

DEFINITION OF FANSI II

The current computational problems associated with T-7 and the supercritical cascade have convinced us that we must be much more sophisticated in our grid generation and handling. These changes amount to a major rewrite of FANSI to produce FANSI-II. This new version will contain new grid structures, new difference cell definitions, new flux definitions, new smoothing, and proper inflow/outflow boundary conditions.

The first step toward FANSI-II is the generation of mixed O type and throughflow type grids. These grids contain most of the grid lines to resolve the boundary layer near the body and eliminate the upstream grid cluster problem. An example of this grid near the leading edge is shown in Figure 19. These grids also allow more control over cell shapes in physical space and better placement of grid lines near leading edges. This last item is particularly important for the turbine cascades.

Since we expect to use this same type grid in three-dimensional problems, a internal grid block mapping technique is being developed. This technique will organize the codes internal grid storage algorithm to operate on blocks or groups of cells which will facilitate development of the three-dimensional viscous code. This feature will be transparent to the grid generation process. A second benefit of the new structure is that local grid refinements and overlapping grid definitions will be much easier to implement.

A change to the flux cell definitions in the 2-D viscous code is to be implemented. The previous version stored the state vectors at the grid nodes. This severely limited the grid choice options. The new scheme has the state variables stored at the centers of the grid "cells", and the grid cells are now to be the flux cells. This will enable the grids to be changed from full throughflow type grids to a mixture of C-type (or O-type) and throughflow grids. An enlargement of the leading edge of this new type of grid was presented in Figure 19. The boundary layer grid lines go fully around the leading edge, thereby

removing the inlet "wake" present in the earlier versions of the grids. This type of grid also ensures that the grid lines are normal to the profile surface all the way around the leading edge. The cross in the center of each grid cell is where the state variable is going to be stored. It can be seen that although the grid nodes may have more than four neighbors, the cells are all four-sided.

At the moment the trailing edge grid scheme is the same as shown in Figure 19. However, this is in the process of being changed to give an O-type boundary layer grid on the profile surface. The use of an O-type or throughflow type of grid at the trailing edge has not yet been fully decided upon.

The new state vector positions are best thought of as cell centered averages. With cell centered averages and better control over grid shapes in physical space, new definitions of flux vectors are possible. The flux vectors are illustrated in Figure 20 and are computed from averages of the cell centered values in a way which minimizes the effect of odd-even decoupling in a central difference scheme.

$$[\rho u]_{j\pm 1/2} = \mu_j^{\pm}(\rho u) = [\overline{\rho u}] \quad (7)$$

$$[\rho]_{j\pm 1/2} = \mu_j^{\pm}(\rho) = [\overline{\rho}] \quad (8)$$

$$[u]_{j\pm 1/2} = [\overline{\rho u}] / [\overline{\rho}] = [\overline{u}] \quad (9)$$

$$[v]_{j\pm 1/2} = [\overline{\rho v}] / [\overline{\rho}] \quad (10)$$

$$[E]_{j\pm 1/2} = [\overline{\rho E}] / [\overline{\rho}] \quad (11)$$

$$[p]_{j\pm 1/2} = (\gamma-1)[\overline{\rho}] \left([\overline{E}] - ([\overline{u}]^2 + [\overline{v}]^2)/2 \right) \quad (12)$$

Viscous flux terms will be defined in terms of the cell centered values

$$[\tau_{xy}]_{j\pm 1/2} = \text{viscosity} [\delta_k^{\pm}(u_{j,k})] \quad (13)$$

With these definitions the FANSI algorithm is to be viewed as a flux

balance method on the cell faces using state vector approximations at the cell centers. A converged steady state solution is obtained when the flux balance becomes zero, and the final solution will be defined in terms of average variables.

FANSI-II will use artificial smoothing operators that are applied to the cell centered values rather than the flux balance operators. The sequence of steps becomes

$$[I - \theta \Delta t L'_{NSX}] \{\Delta U_j^*\} = L_{NS} \{U_j^n\} \quad (14)$$

$$[I - \theta \Delta t L'_{NSX}] \{\Delta U_j^{**}\} = \Delta U_j^* \quad (15)$$

$$U_j^{***} = U_j^n + \Delta U_j^{**} \quad (16)$$

$$U_j' = [I + S_X] \{U_j^{***}\} \quad (17)$$

$$U_j^{n+1} = [I + S_Y] \{U_j'\} \quad (18)$$

where

S_X , S_Y are the smoothing operators

$$S_X U = \alpha_{j+1/2} [U_{j+1} - U_j] + \alpha_{j-1/2} [U_{j-1} - U_j] \quad (19)$$

$$\alpha_{j\pm 1/2} = \text{abs}(\mu_j^\pm p) / (p_{j+1} + p_{j-1} + p_j) \quad (20)$$

$$S_Y U = \alpha_{k+1/2} [U_{k+1} - U_k] + \alpha_{k-1/2} [U_{k-1} - U_k] \quad (21)$$

$$\alpha_{k\pm 1/2} = \text{abs}(\mu_k^\pm p) / (p_{k+1} + p_{k-1} + p_k) \quad (22)$$

The split smoothing simulates both a second order and a fourth order smoothing operator.

FANSI-II will also have a new smoothing operator, which smooths along diagonals in computational space to eliminate an odd-even decoupling mode in these directions which is not presently damped as well as proper inflow/outflow boundary conditions on the non-simply

connected domain.

SUMMARY

Efforts on the three-dimensional code development have centered on selection of an efficient algorithm for the turbine blade row case. The algorithm selected is a hybrid explicit-implicit algorithm which should retain maximum speed and flexibility for both inviscid and viscous calculations. Development of the inviscid section of this code is now about 70% complete, and a first version of the inviscid operator should be available about April 1983. Two or three months of work should then be needed to add the viscous operators to this code. The viscous operators are to be based on the two-dimensional viscous code.

DESCRIPTION OF PROPOSED ALGORITHM

The proposed algorithm relies on splitting the Navier-Stokes equations in an inviscid operator and a viscous operator.

$$U_{,t} + L_h\{U\} = L_v\{U\} \quad (24)$$

where

L_h is a finite difference operator which operates on the state vector to produce the steady state flux balance for the inviscid terms

L_v is a finite difference operator which approximates the viscous steady state flux balance.

The proposed operator split is:

STEP 1

$$U^* = U^n - \Delta t L_h \{U^n\} \quad (25)$$

STEP 2 a)

$$[I - \Delta t L_v] \{U^{n+1} - U^*\} = \Delta t L_v \{U^*\} \quad (26)$$

STEP 2 b)

$$[I - \frac{\Delta t L_v}{2}] \{U^{n+1} - U^*\} = \Delta t [L_v \{U^*\} + \frac{\Delta t L_v}{2} \{L_h \{U^n\}\}] \quad (27)$$

where the linearized viscous operator is defined by

$$L_v^{n+1} \{U^{n+1}\} = L_v \{U^n\} + L_v \{U^{n+1} - U^n\} \quad (28)$$

Step 1 and step 2 b) form a second order accurate time integration scheme, while step 1 and step 2 a) form a first order accurate time integration scheme.

The single step version of either choice is:

$$[I - \Delta t L_V] \{U^{n+1} - U^n\} = \Delta t \{L_V \{U^n\} - L_H \{U^n\}\} \quad (29)$$

as desired.

This operator splitting, which is not the same as direction splitting, has a number of important advantages:

An explicit, fully conservative form in the inviscid regions which is similar to present Euler codes.

Viscous operator already developed as FANSI or FANSI-II.

Implicit viscous operator only need be performed in viscous regions and not in core flow.

Steady state solution acceleration techniques applicable to step 1 but with solutions independent of Δt .

Boundary conditions on intermediate steps are well defined.

STATUS

Coding of the 3-D split operator forms has begun and a "centered" MacCormack operator developed. This operator is being tested against transonic compressor solutions run in the past and a low speed turbine test case. This testing is about 70% complete and it is expected that a centered, second order Runge-Kutta time integration scheme like that of

reference [5] will be developed. This scheme is compatible with the expected FANSI-II developments. This work should be complete by June 1983. Future work might include a Ni type, reference [6], cell structure and multi-grid procedure.

After the Euler operator is sufficiently well developed, work will begin on updating FANSI for use as the viscous operator and the turbulence model. This coding work should be complete by August 1983.

SCHEDULES

INTERIM FANSI COMPUTER CODE

APRIL 1 1983

STAGGERED GRIDS ONLY

FULL ALGEBRAIC TURBULENCE MODEL

CORRECTED INFLOW/OUTFLOW CONDITIONS

FANSI FINAL REPORT

JUNE 1 1983

R.H. BUSH THESIS

FANSI - II COMPUTER CODE, FIRST VERSION

JUNE 1 1983

3-D EULER OPERATOR COMPUTER CODE, FIRST VERSION

APRIL 1 1983

3-D VISCOUS OPERATOR COMPUTER CODE, FIRST VERSION

AUGUST 1 1983

PAPERS AND REPORTS

During the past year a number of papers and reports associated with this project have either been presented, scheduled as invited papers or cleared for journal publication.

Thompkins, W.T., Tong, S.S., Bush, R.H., Usab, W.J., and Norton, R.J.G., "Solution Procedures for Accurate Numerical Simulations of Flow in Turbomachinery Cascades," AIAA paper 83-0257, Jan. 1983.

Presented at AIAA Aerospace Sciences Meeting, Reno Nevada

Expanded version to appear as chapter 4 of "Recent Advances in Numerical Methods in Fluids," Volume IV.

Compressed version submitted to AIAA Journal

Thompkins, W.T. and Haimes R., "A Minicomputer/Array Processor/Memory System for Large-Scale Fluid Dynamic Calculations"

Invited paper for SYMPOSIUM ON IMPACT OF NEW COMPUTING SYSTEMS ON COMPUTATIONAL MECHANICS, ASME Winter Annual Meeting, 1983

Lavante, E.V. and Thompkins, W.T., "An Implicit, Bi-Diagonal Numerical Method for Solving the Navier-Stokes Equations," AIAA paper 82-0063.

Cleared for publication AIAA Journal

Thompkins, W.T. and Bush, R.H., "Boundary Treatments for Implicit Solutions to Euler and Navier-Stokes Equations"

Published Journal of Computational Physics, Nov. 1982

REFERENCE LIST

- [1] Beam, R. M. and Warming, R. F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," AIAA Journal, Vol. 16, No. 4, April 1978, pp. 393-402.
- [2] Bush, R. H., "Time Accurate Internal Flow Solutions of The Thin Shear Layer Equations," Gas Turbine and Plasma Dynamics Laboratory, M.I.T., Report No. 156, February 1981.
- [3] T. Cebeci and A.M.O. Smith, "Analysis of Turbulent Boundary Layers," Academic Press, New York, 1974.
- [4] Baldwin, B.S, and Lomax, H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA Paper #78-257, presented at AIAA 16th Aerospace Sciences Meeting, Huntsville, Alabama January 16 - 18, 1978.
- [5] Jameson, A., Schmidt, W. and Turkel, E., "Numerical Solutions to the Euler Equation by Finite Volume Methods Using Runge-Kutta Time Stepping," AIAA Paper 81-1259, June 1981.
- [6] Ni, R. H., "A Multiple Grid Scheme for Solving the Euler Equation," AIAA paper 81-1025, June 1981.

APPENDIX

Finite Difference Operator Definitions

$$\delta_j^0 u_{j,k} = \frac{1}{2}(u_{j+1,k} - u_{j-1,k})$$

$$\mu_j^+ u_{j,k} = \frac{1}{2}(u_{j+1,k} + u_{j,k})$$

$$\mu_j^- u_{j,k} = \frac{1}{2}(u_{j,k} + u_{j-1,k})$$

$$\delta_j^+ u_{j,k} = u_{j+1,k} - u_{j,k}$$

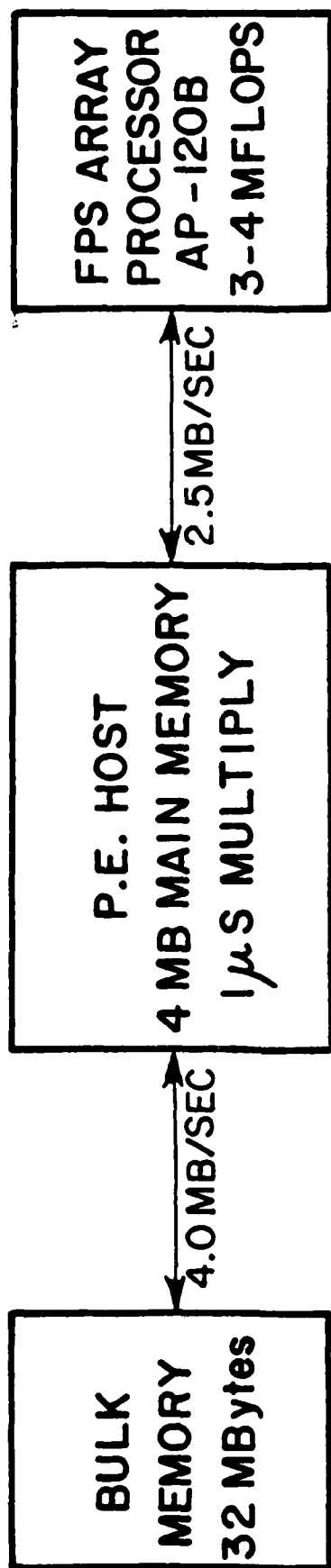
$$\delta_j^- u_{j,k} = u_{j,k} - u_{j-1,k}$$

$$\delta_{j/2}^0 u_{j,k} = u_{j+1/2,k} - u_{j-1/2,k}$$

Note that all coordinate positions at non-integer mesh spacings are to be defined by a simple average

$$y_{j+1/2,k} = \mu_j^+ y_{j,k} .$$

PRESENT COMPUTER FACILITY



CONCURRENT OPERATIONS IN ALGORITHM PROCESSING

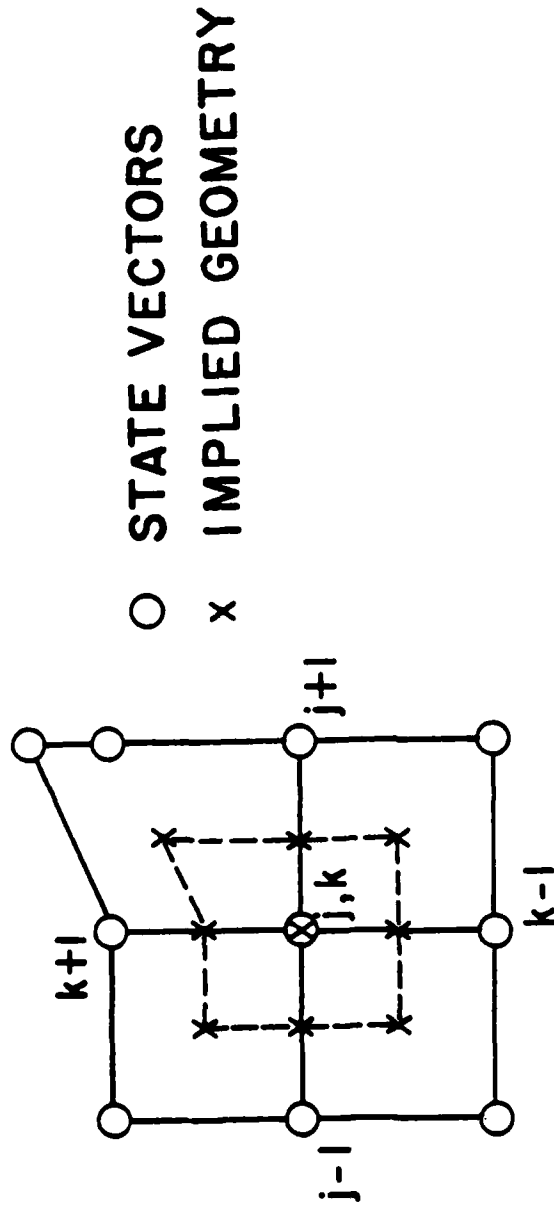
Vector Processing in FPS AP-120B

Scalar Processing in Host-Turbulence Model

Data Movement Between Bulk Memory $\leftrightarrow$ Host $\leftrightarrow$ AP-120B

Figure 1

FANSI CELL AND FLUX DEFINITIONS

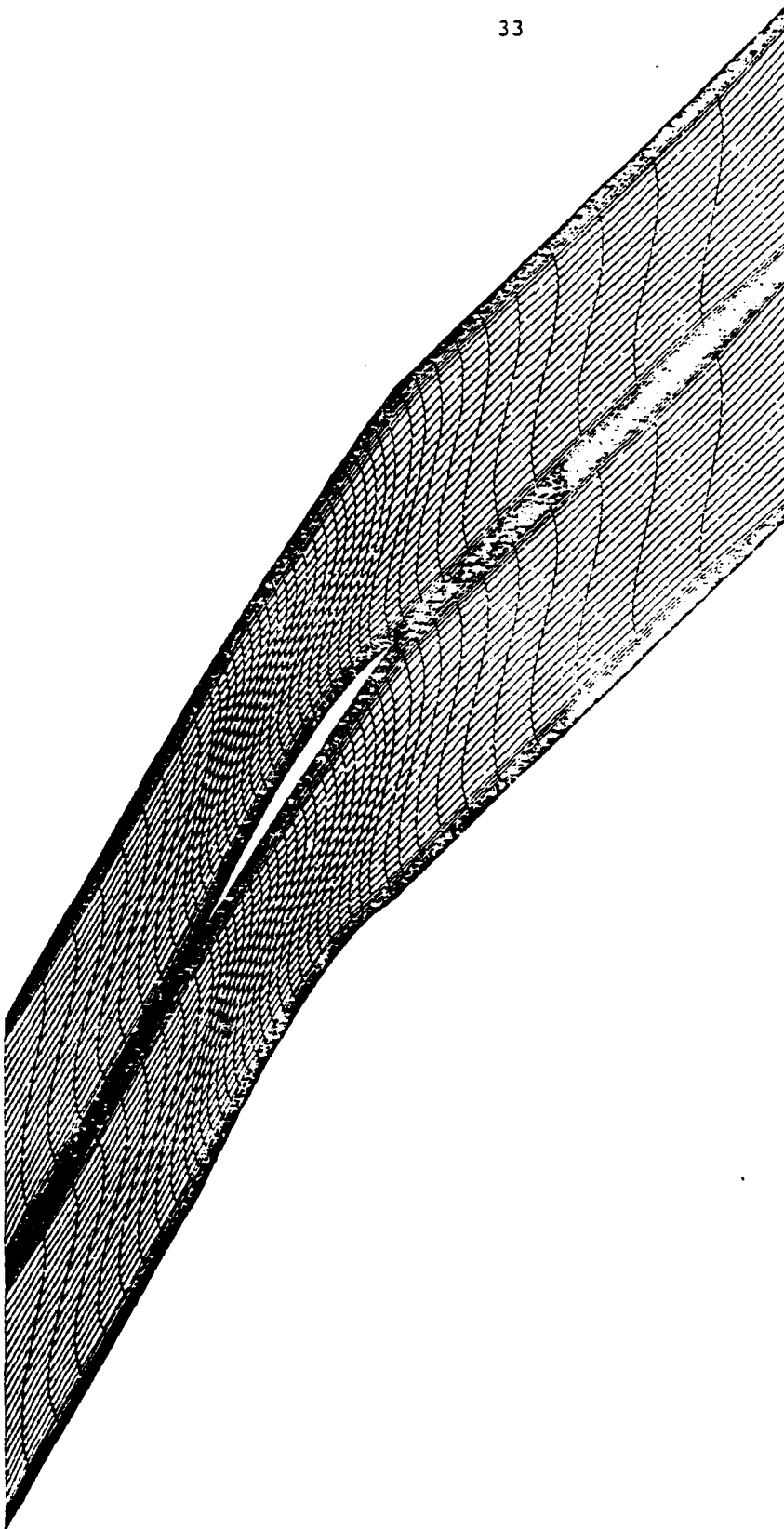


$$(\rho u)_{j+1/2} = 1/2 [(\rho u)_{j+1} + (\rho u)_j] = \mu_j^+ (\rho u)$$

$$(\rho u^2 + P)_{j+1/2} = \mu_j^+ (\rho u^2 + P)$$

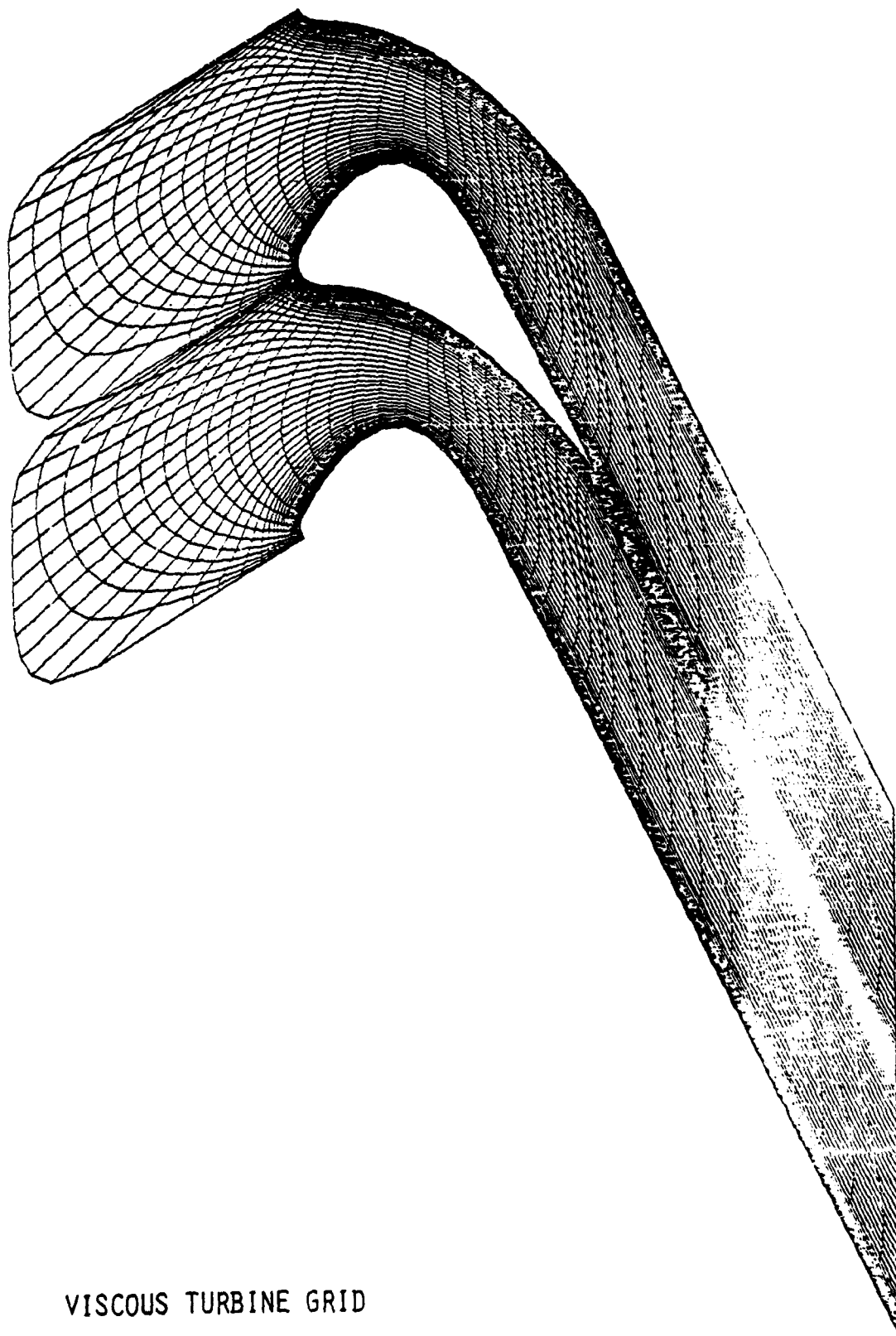
$$(\rho u)_{j+1/2, k+1/2} = \mu_k^+ \mu_j^+ (\rho u)$$

Figure 2



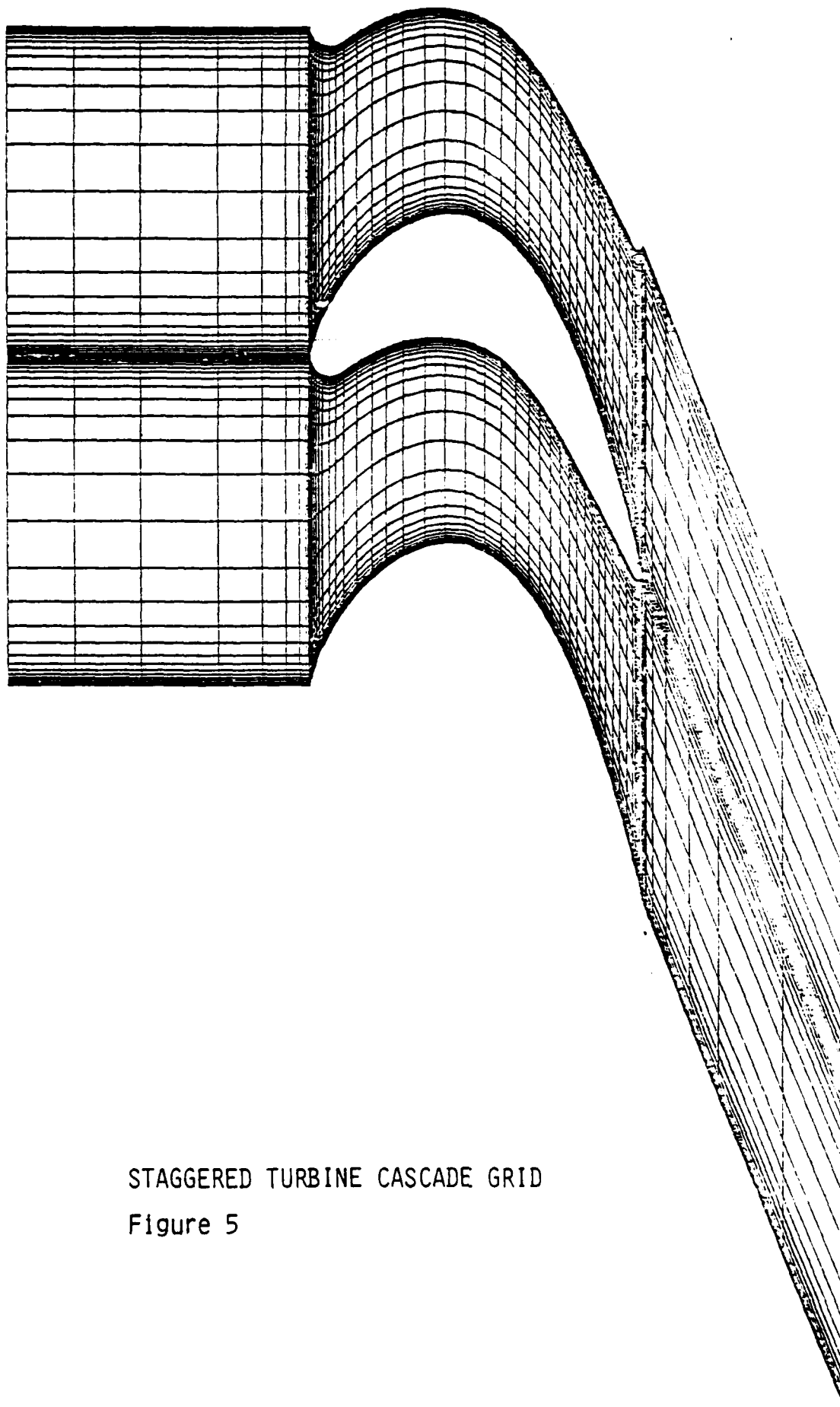
VISCOUS COMPRESSOR GRID

Figure 3



VISCOUS TURBINE GRID

Figure 4



STAGGERED TURBINE CASCADE GRID
Figure 5

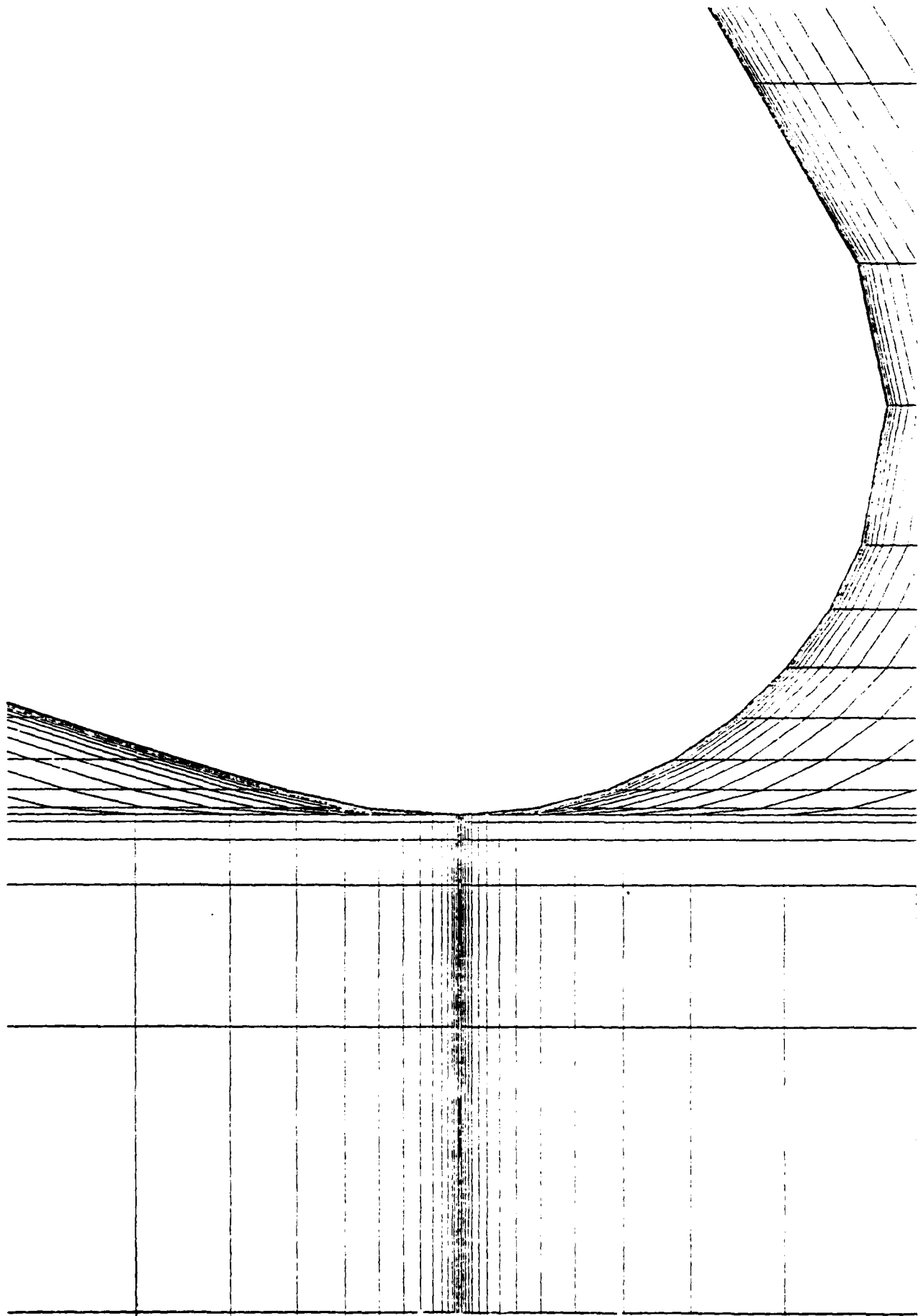


Figure 6

LEADING EDGE BLOWUP, STAGGERED TURBINE GRID

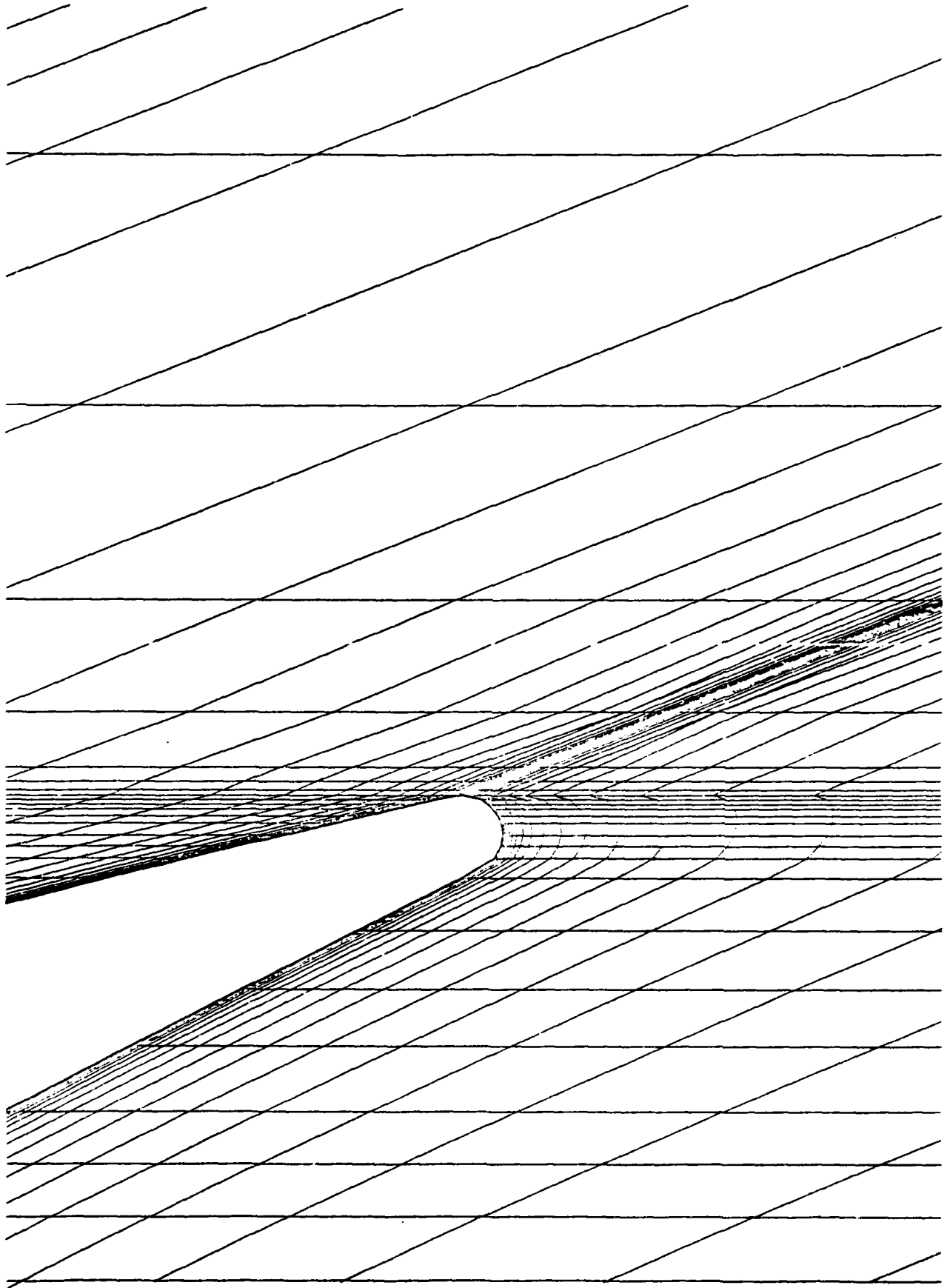
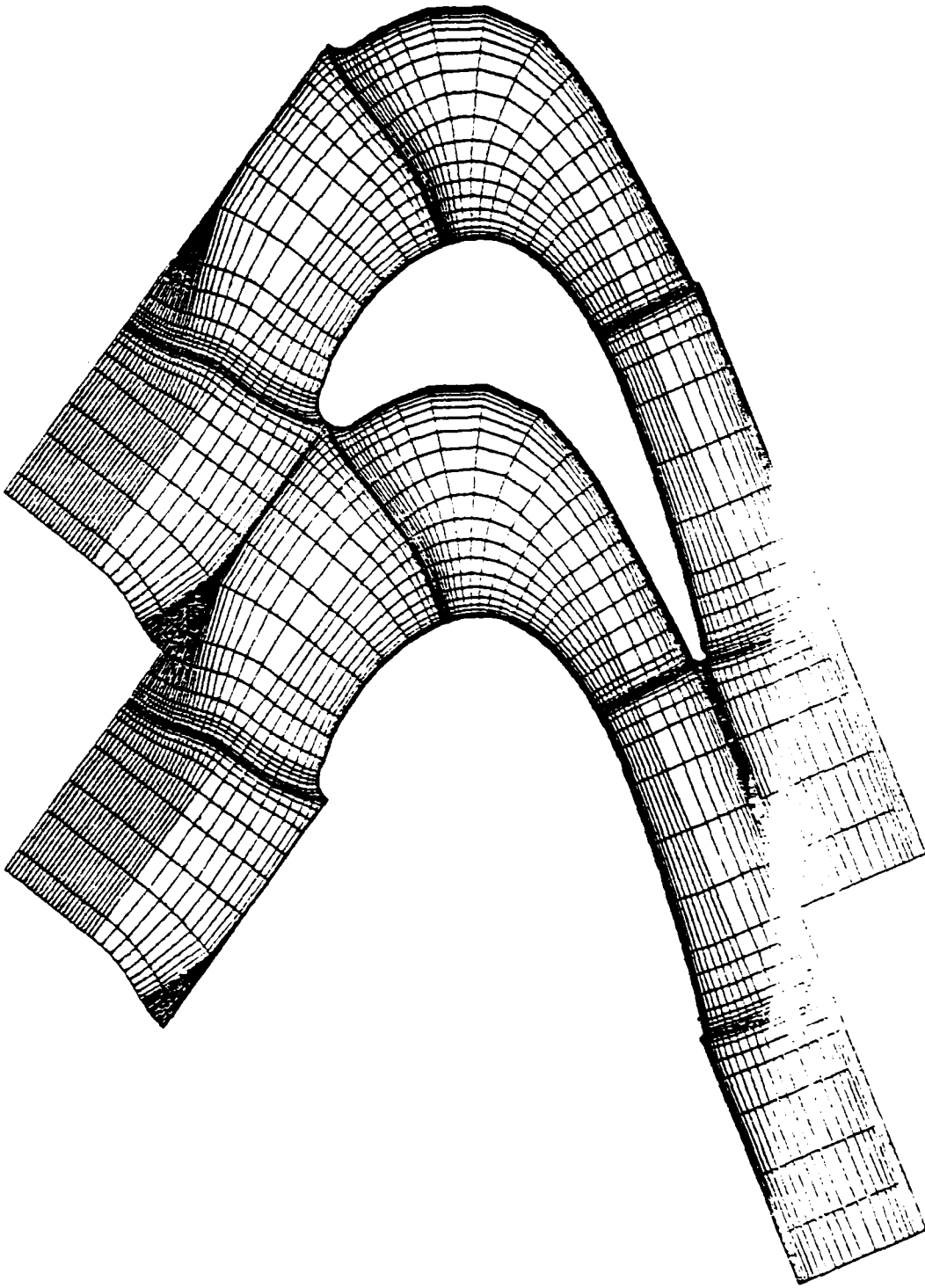


Figure 7

TRAILING EDGE BLOWUP, VISCOUS TURBINE CASCADE



NON-SIMPLY CONNECTED TURBINE CASCADE GRID

Figure 8

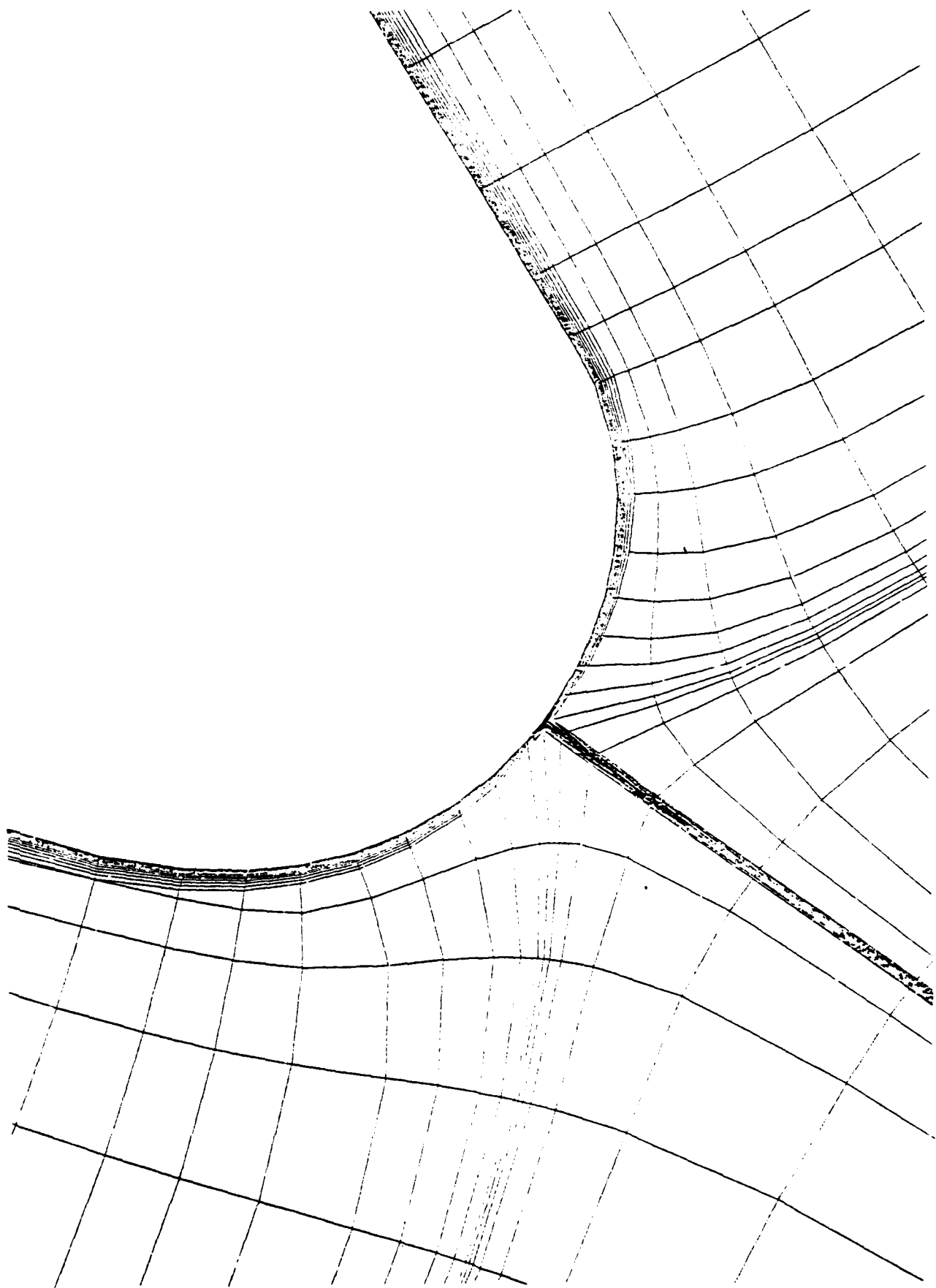


Figure 9

LEADING EDGE BLOWUP, NON-SIMPLY CONNECTED GRID

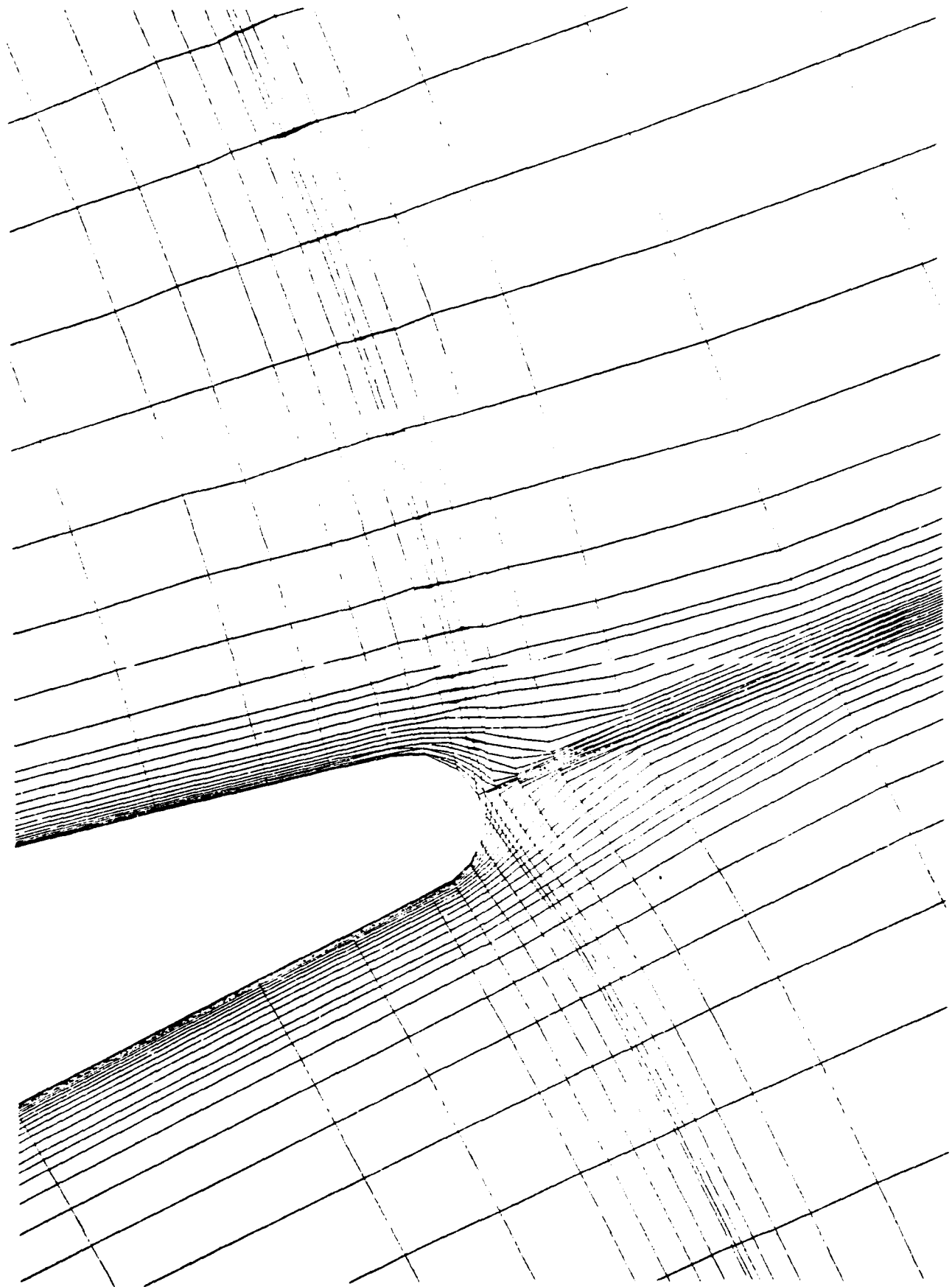
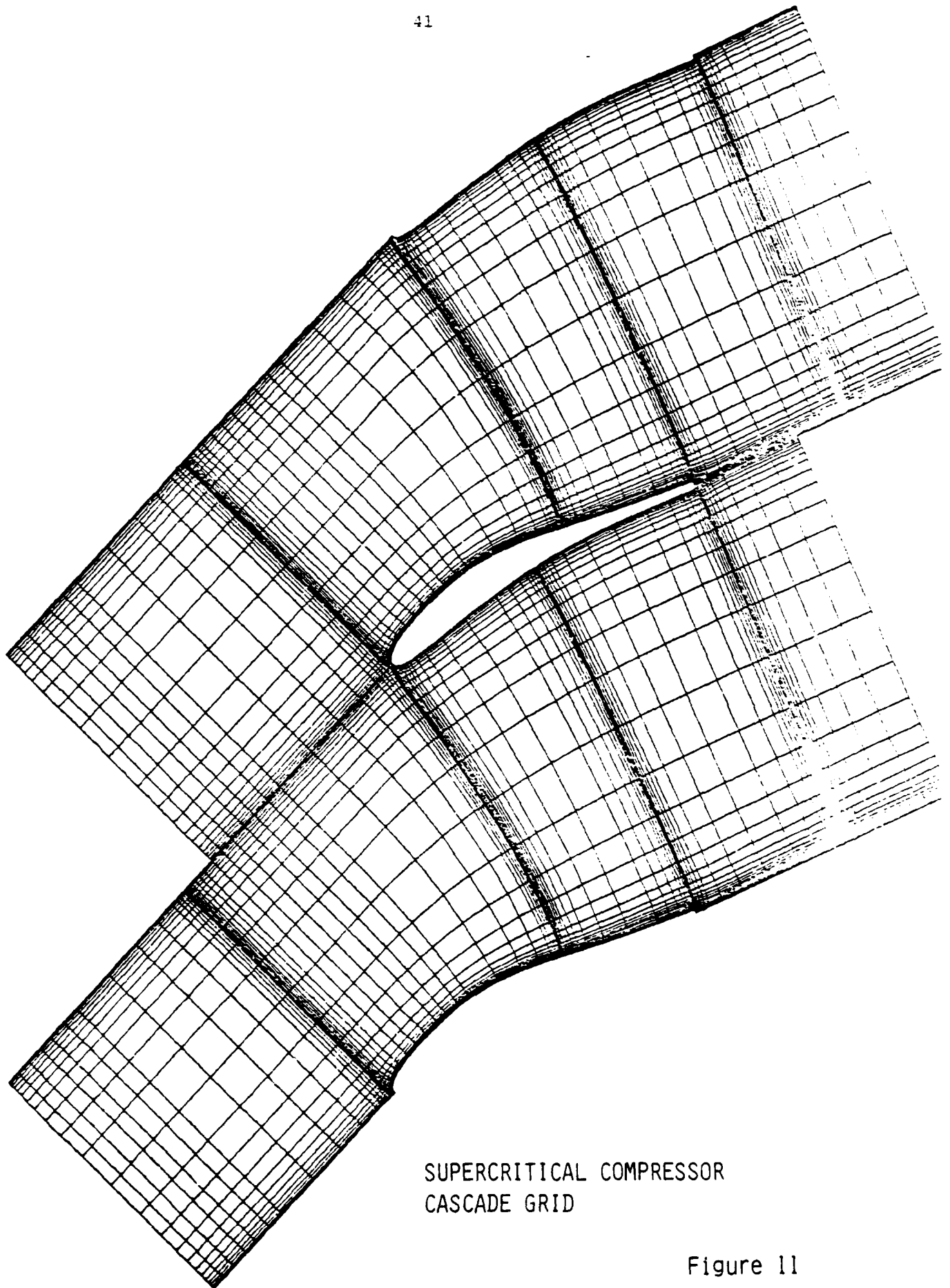


Figure 10

TRAILING EDGE BLOWUP, NON-SIMPLY CONNECTED GRID



SUPERCRITICAL COMPRESSOR
CASCADE GRID

Figure 11

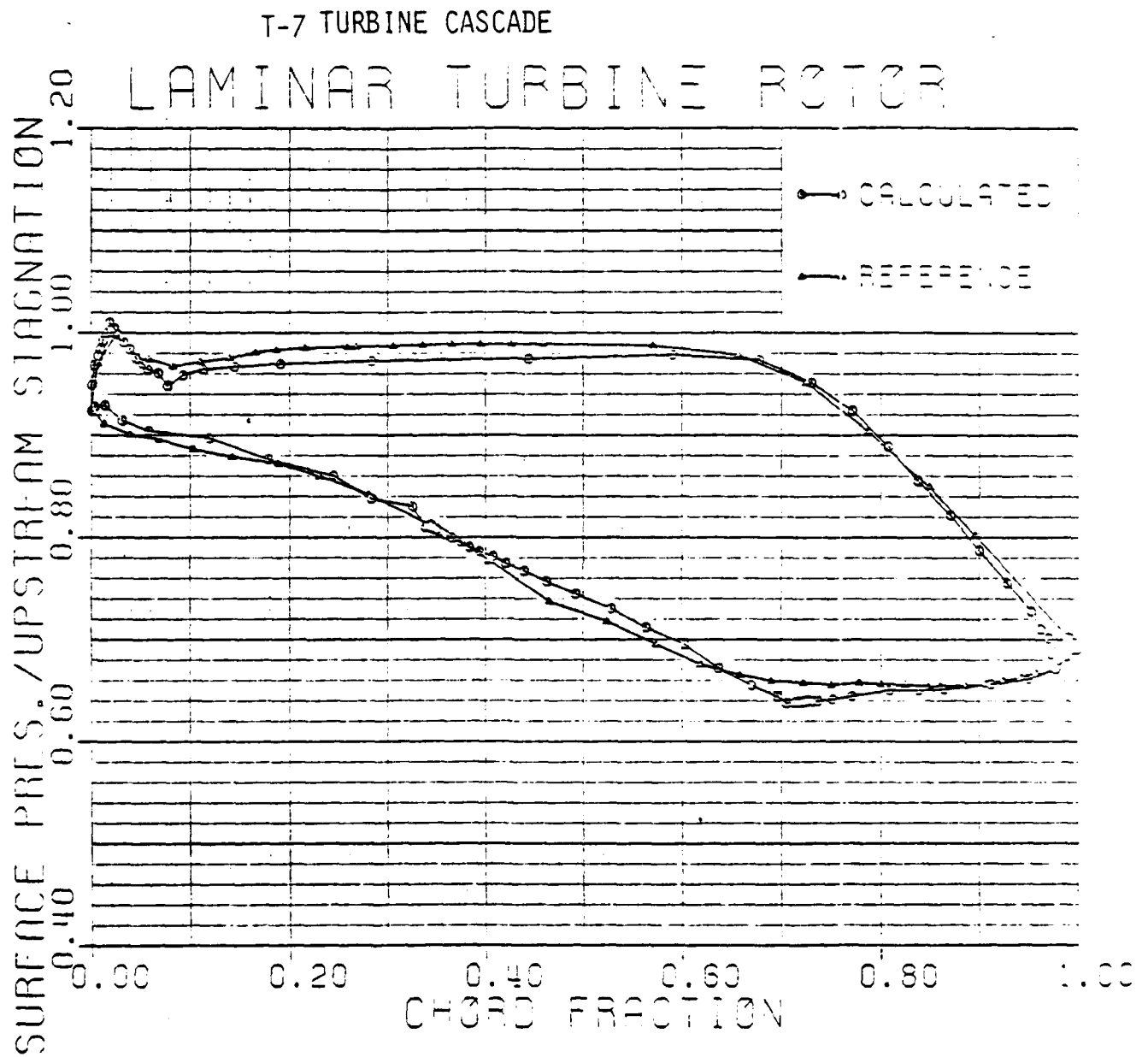


Figure 12

T-7 TURBINE CASCADE
DIRECTION VECTOR PLOT
LAMINAR T.E., TURBINE ROTOR



Figure 13

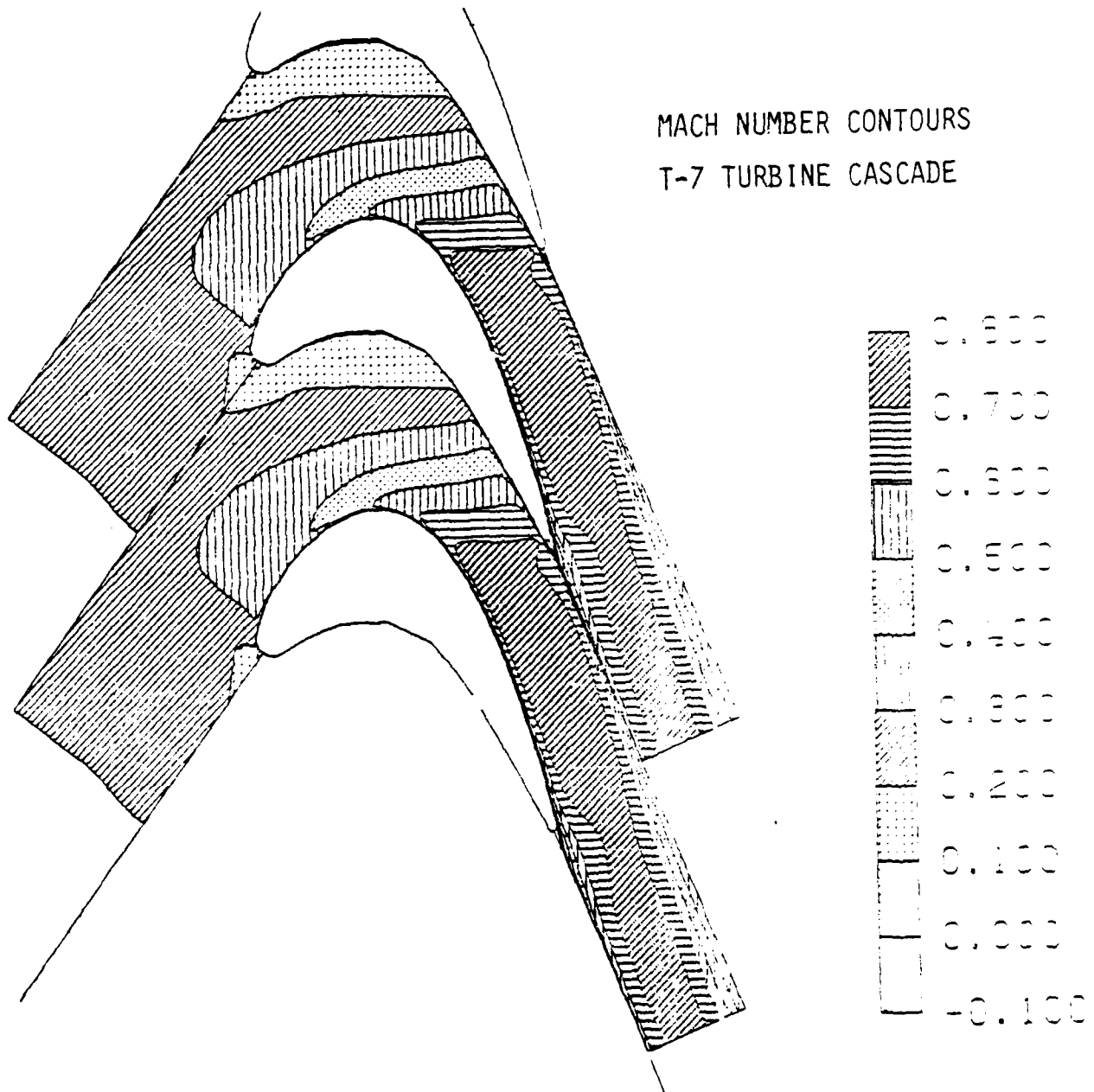


Figure 14

T-7 TURBINE CASCADE

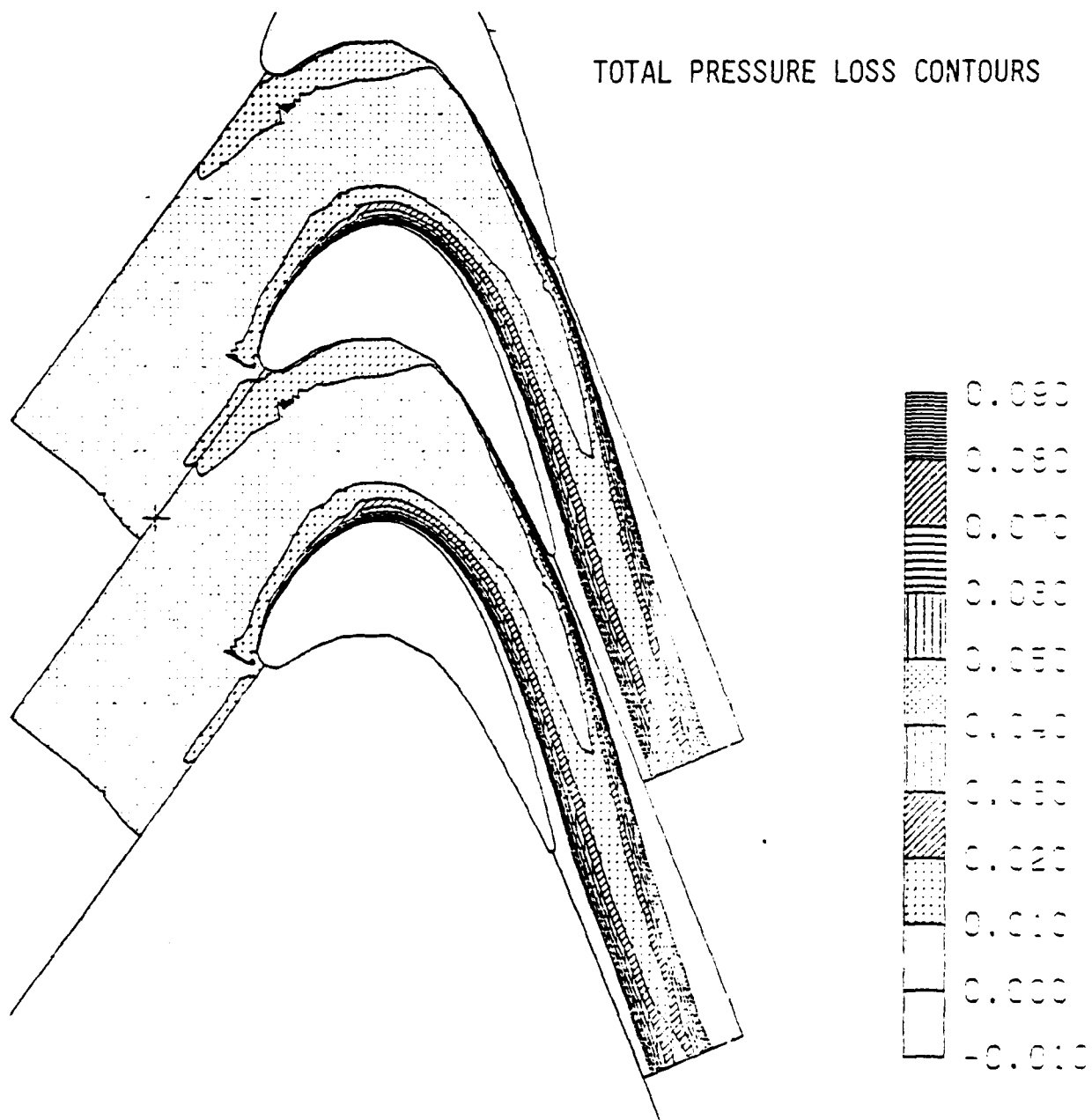


Figure 15

T-7 TURBINE CASCADE, LAMINAR
-10 DEGREES INCIDENCE

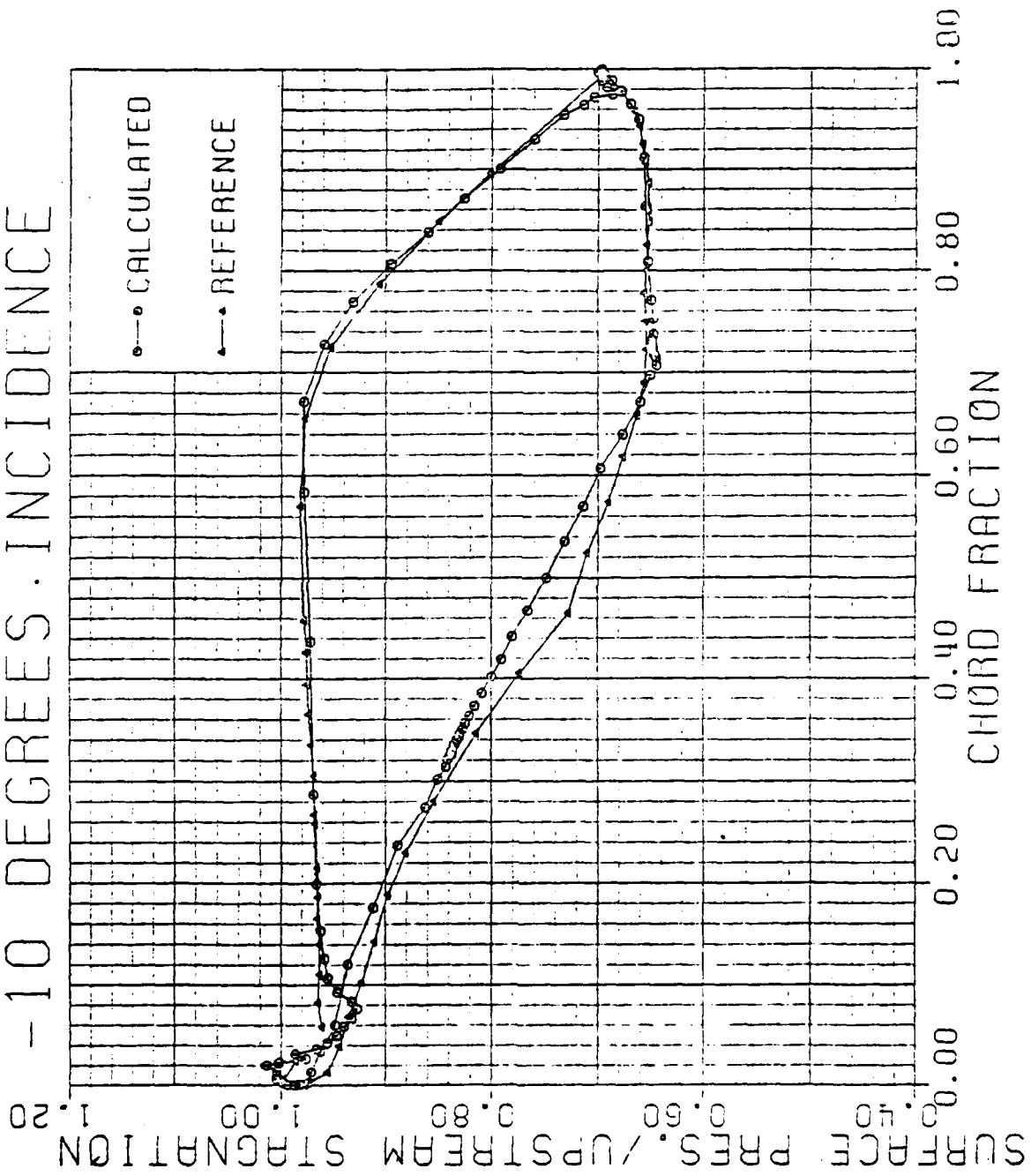


Figure 16

SUPERCritical COMPRESSOR CASCADE
TURBULENCE MODEL

MACH NUMBER

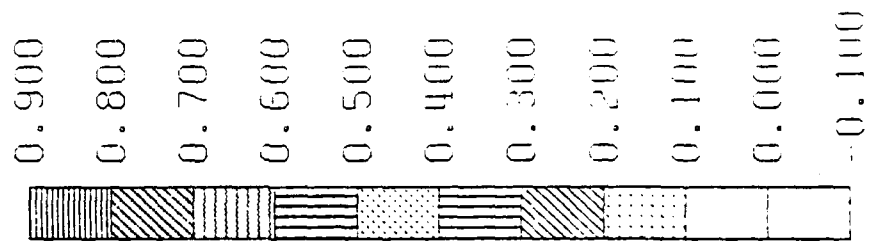
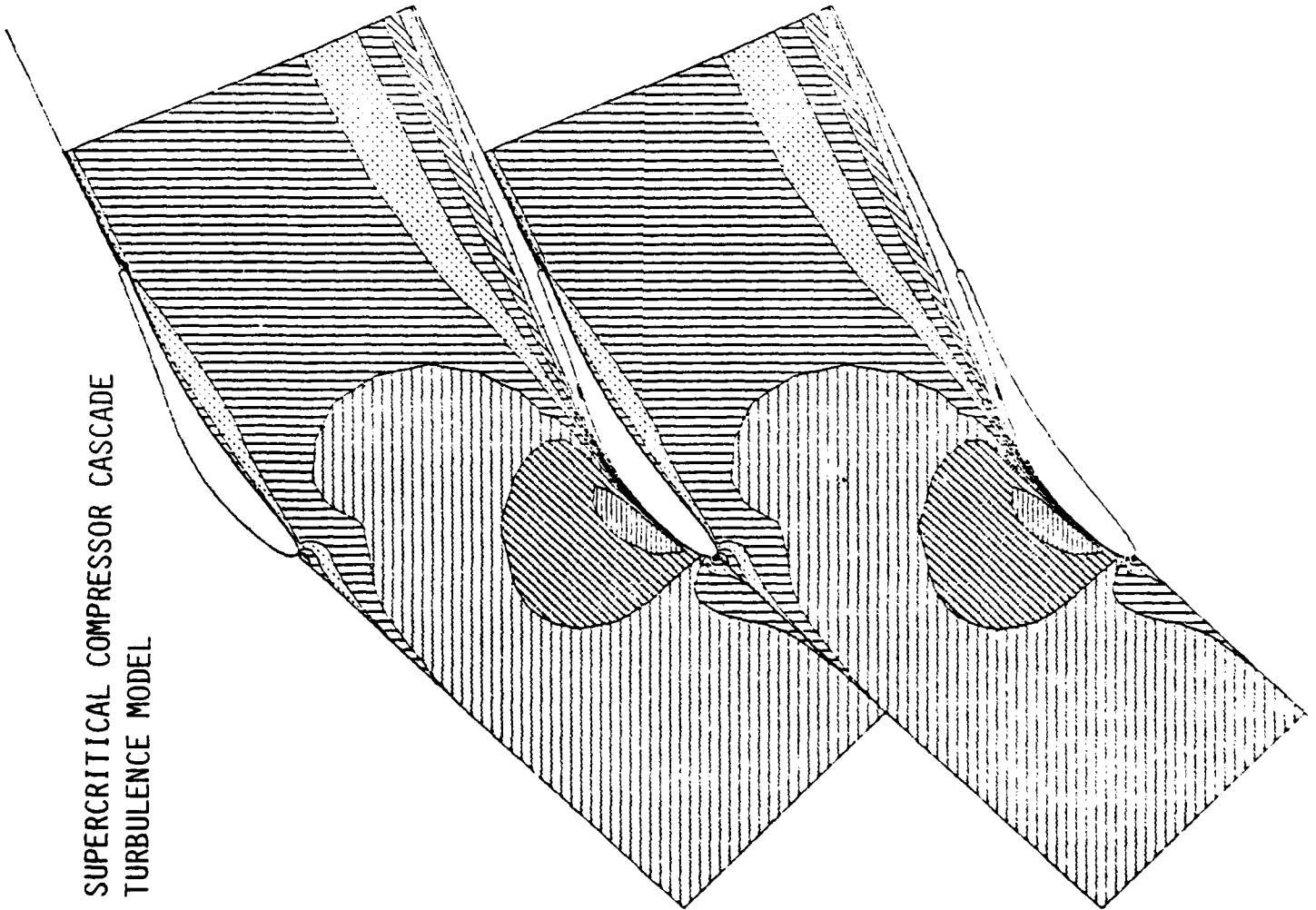


Figure 17



SUPERCRITICAL COMPRESSOR CASCADE
FULL TURBULENCE MODEL

TOTAL PRESSURE LOSS

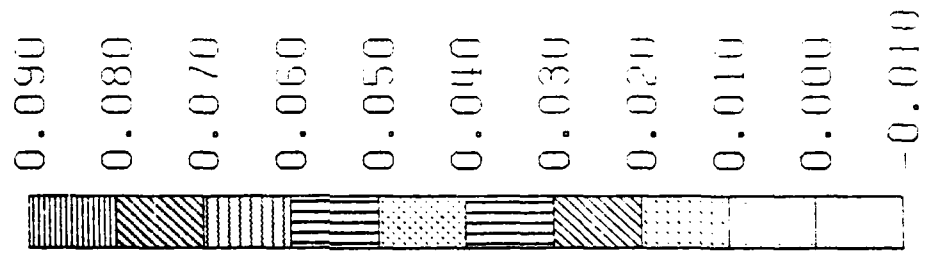


Figure 18

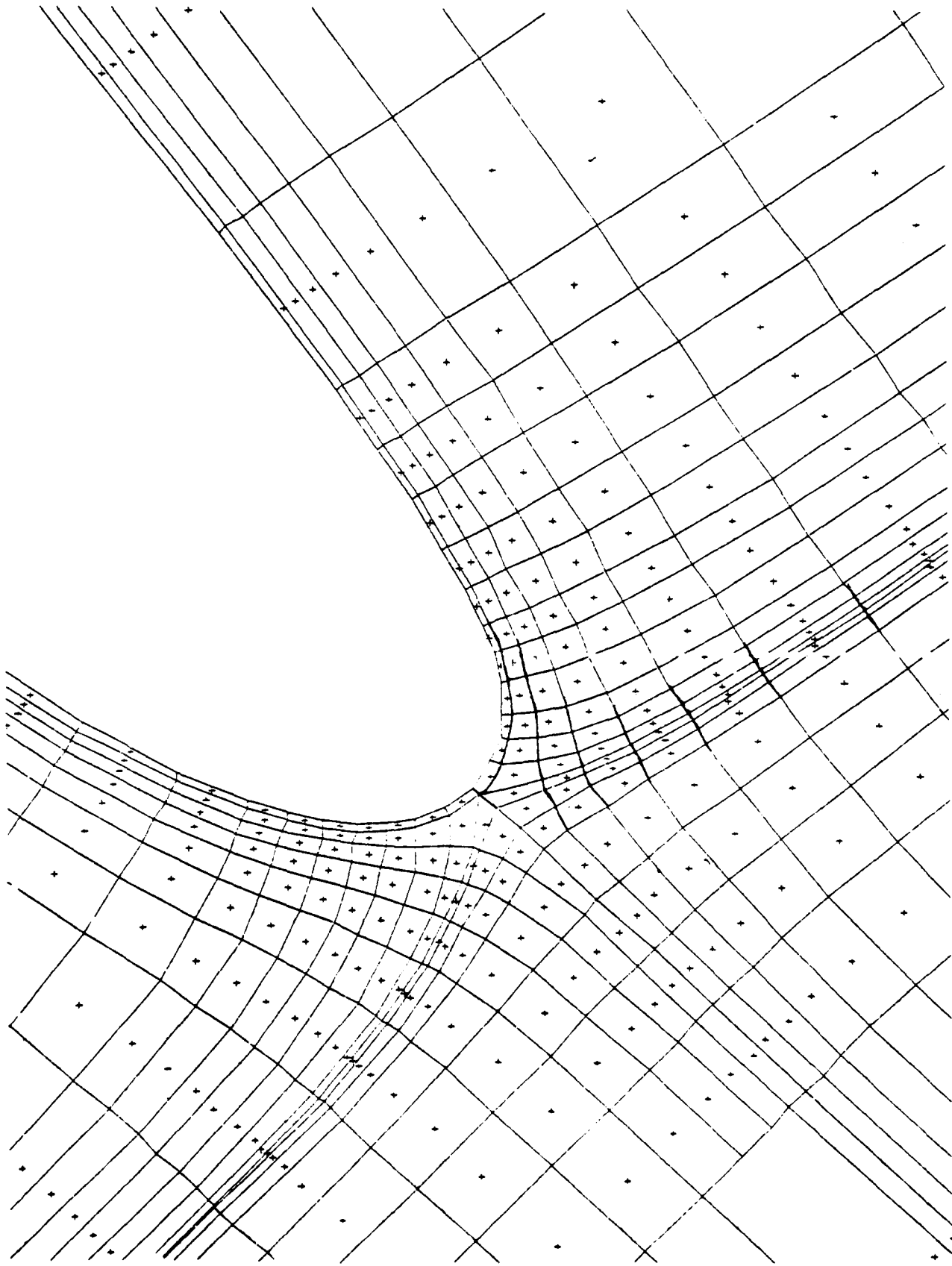


Figure 19

FANSI-II LEADING EDGE CELLS

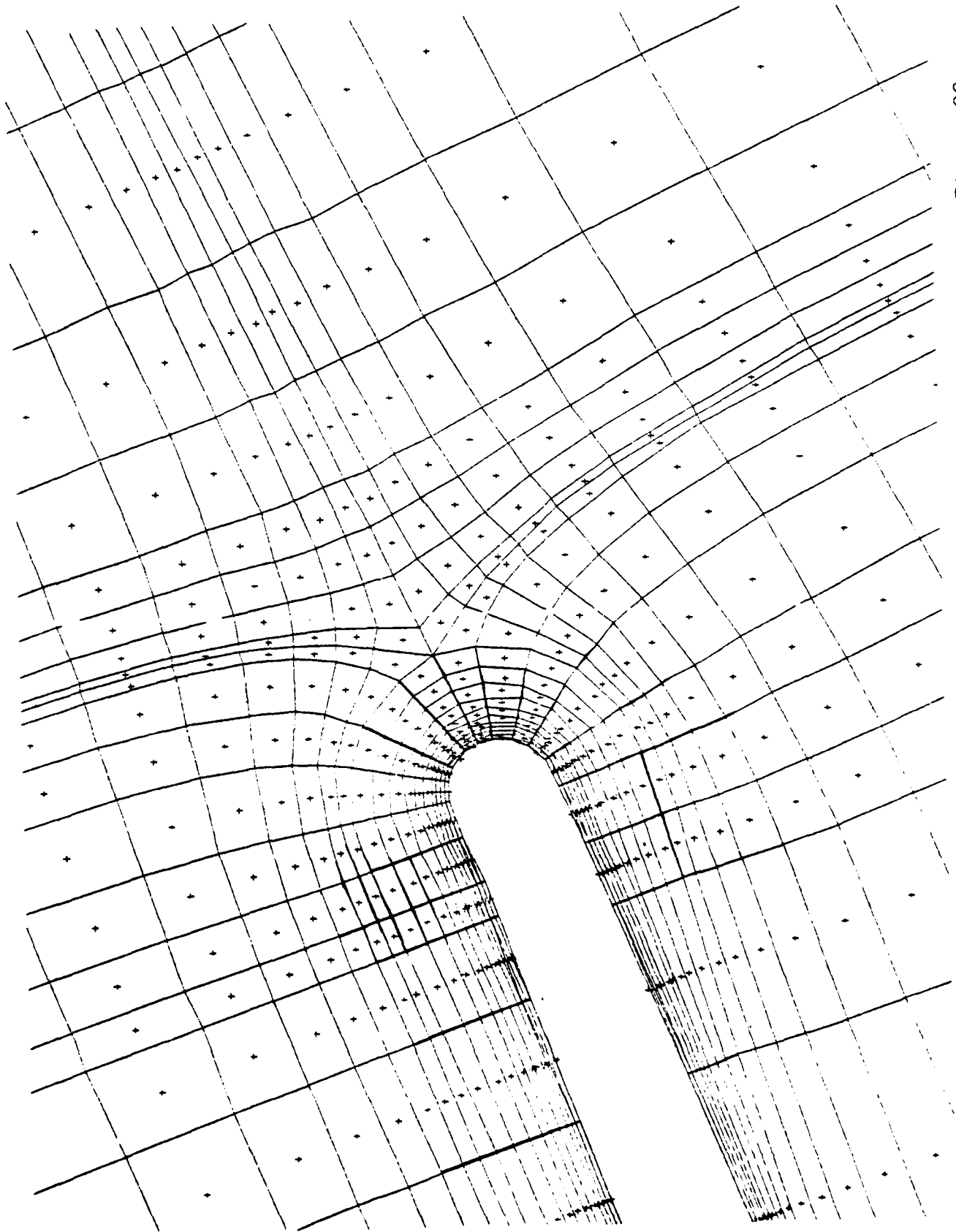
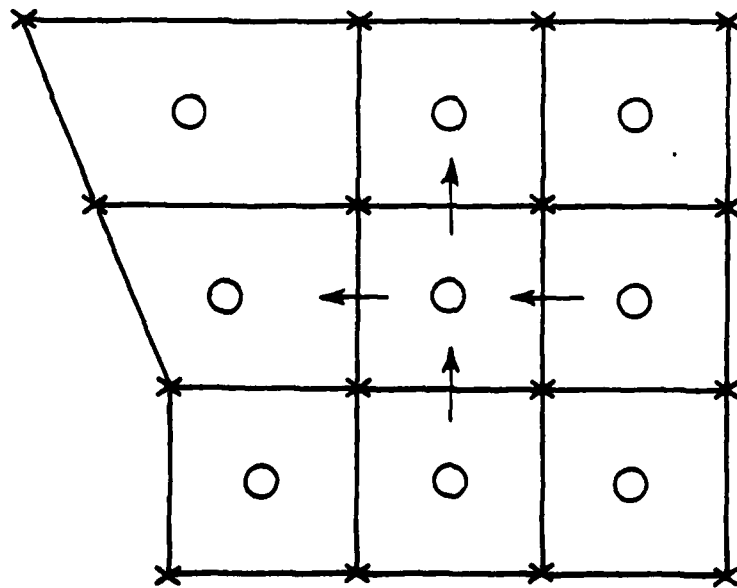


Figure 20

FANSI-II TRAILING EDGE CELLS

FANSI-2 CELL AND FLUX DEFINITIONS

51



O STATE VECTORS

x GEOMETRY

$$(\rho u)_{j+1/2} = \mu_j^+(\rho u)$$

$$(u)_{j+1/2} = [\mu_j^+(\rho u)] / \mu_j^+(\rho)$$

$$(\rho u^2 + P)_{j+1/2} = (\rho u)_{j+1/2} u_{j+1/2} + P_{j+1/2}$$

Flux And Damping Calculated
At Side Of Cell

Figure 21