NASA CR-166,317

NASA CONTRACTOR REPORT 166317

BGRID:A Block-Structured Grid Generation Code For Wing Sections

H.C. Chen and K.D. Lee

CONTRACT NAS2-10676
November 1981

NASA

NASA CONTRACTOR REPORT 166317

BGRID: A Block-Structured Grid Generation Code for Wing Sections

H.C. Chen and K.D. Lee
The Boeing Company
P.O. Box 3707
Seattle, Washington 98124

Prepared for
Ames Research Center
under Contract NAS2-10676

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

N82-74148#

BGRID - A BLOCK-STRUCTURED GRID GENERATION CODE FOR WING SECTIONS

H. C. Chen and K. D. Lee

The Boeing Company
Seattle Washington 98124

Prepared for ARC Under Cotract NAS2-10676

SUMMARY

The operation of the BGRID computer program is described for generating block-structured grids. Examples are provided to illustrate the code input and output. The application of a fully implicit AF (approximation factorization)-based computer code, called TWINGB (Transonic WING), for solving the 3D transonic full potential equation in conservation form on block-structured grids is also discussed.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRUDUCTION

The basic feature of BGRID is that it generates grids with a block-structure (reference 1). An important aspect of BGRID is its adaptability to complex configurations with multiple components. Linear grid generation equations are used for simplicity. The grid produced is verified by computing the Jacobian for each grid cell. A good grid will have positive Jacobian for all grid cells to ensure that the transformation between the physical and the computational spaces is one-to-one. The block-structured grids generated by BGRID have been successfully used with finite volume solution algorithms using SLOR (successive line over-relaxation) in 2D (reference 2) and 3D (reference 1).

This study adapts the TWINGB (reference 3), an fully implicit AF-based computer code to the block-structured grids for the solution of the 3D transonic full potential equation in conservation form. This report describes how to use the modified TWINGB.

The block-structurted grid generation methodology in the BGRID code is summarized in Chapter II. Input data description and output interpretation for BGRID are presented in Chapters III and IV. BGRID program information is given in Chapter V. Applications of the TWINGB code with block-structured grids are discussed in chapter VI. Operation of TWINGB is described in Chapter VII.

II: BLOCK-STRUCTURED GRID

The construction of a suitable grid system for complex 3D configurations, such as a wing/body/nacelle, is a necessary step for computing the corresponding transonic full potential flow. Two approaches have been available based on Thompson's (reference 4) surface-adapted coordinate concept. A limited approach maps the flow domain surrounding a 3D configuration into a single rectangular box. This approach has been successful for simple geometries (reference 5) but cannot be effectively applied to complex configurations with multi-components. A more general approach divides the computational domain into multiple rectangular blocks where the configuration itself is also represented by a set of blocks whose structure follows the nature lines of the configuration (Figure 1). This results in a multi-block grid system which is adaptable to complex configurations with multiple components and can produce good grid quality near physical corners. A comparison of typical single block and multi-block grids is given in Figure 3. In General, a division of the flow field into multiple blocks allows the accommodation of slope discontinuities of the boundary surfaces and provide good boundary fitting behavior (Figure 4).

A typical block-structured grid generation (Figure 5) process is described below:

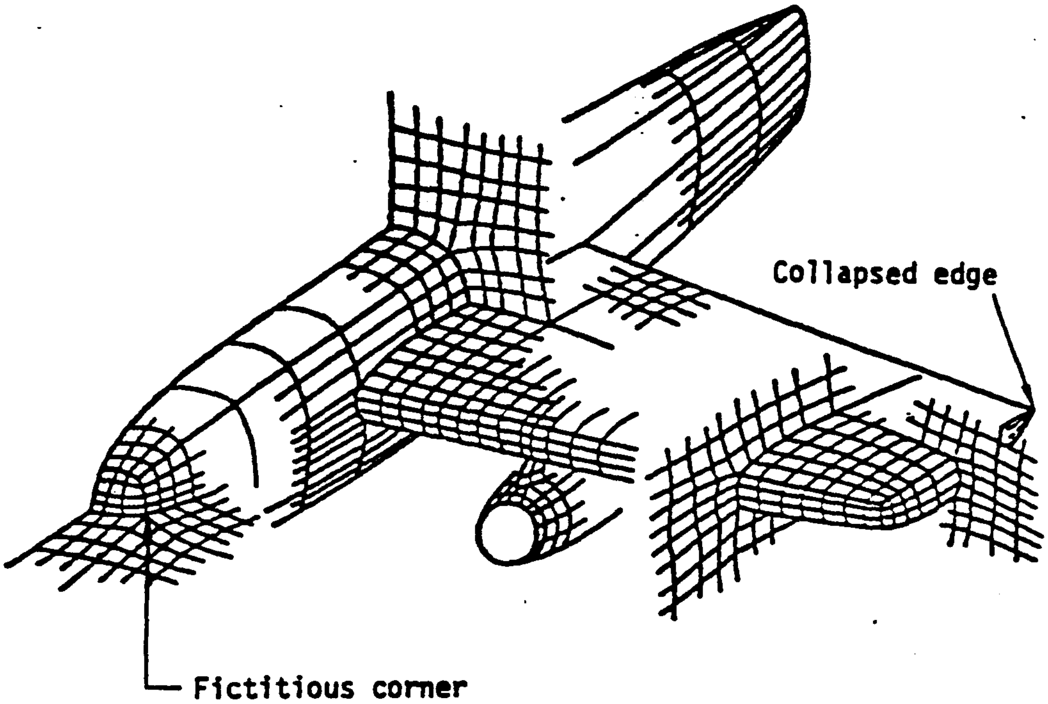
1. Define the overall block structure according to the natural lines of the configuration.

2. Generate 1D grids along the block perimeter (perimeter discretization).

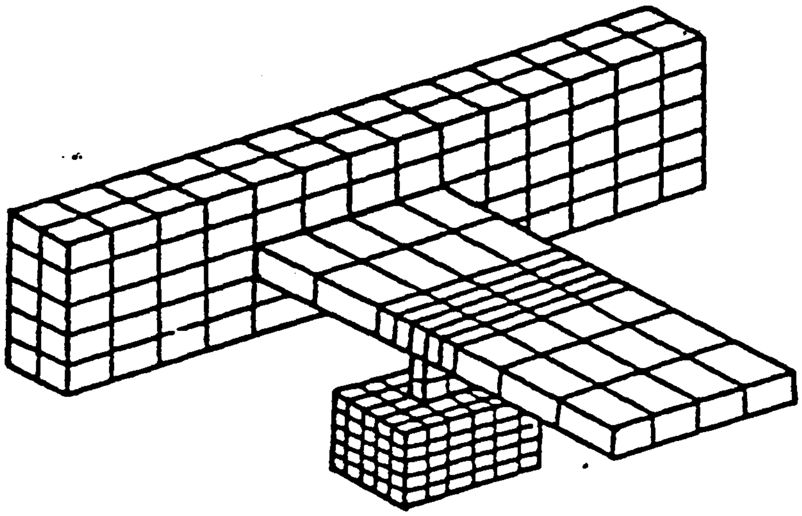
3. Generate 2D grids covering the block surfaces.

4. Generate 3D grid covering the volume grids filling each block using the information obtained in Step 3 as a boundary condition for the 3D grid generation process.

A simplified procedure is used to replace step 4 for wing grid generation. In step 3, 2D grids are generated for each wing span station.

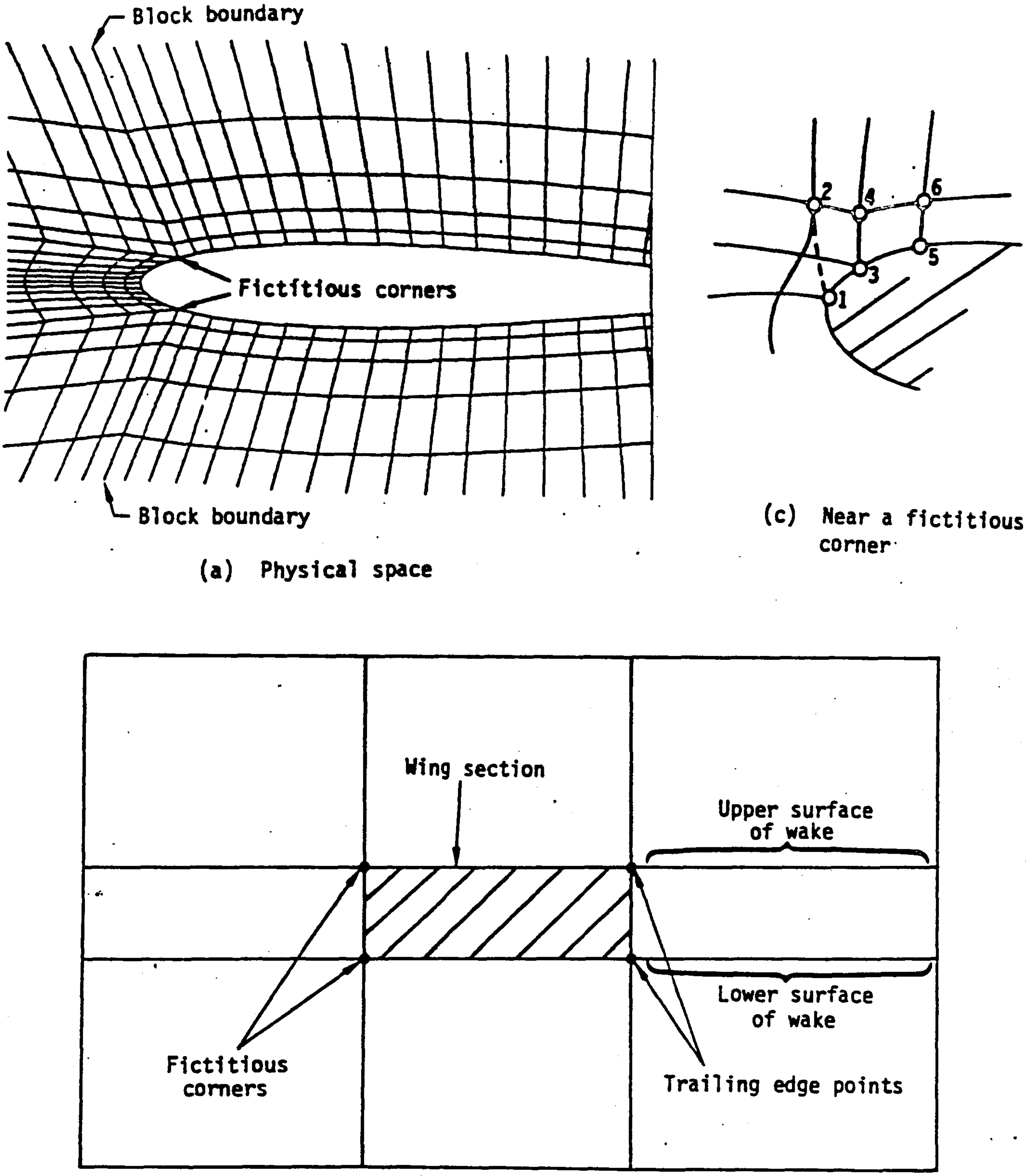


Physical space



Computational space

Figure 1. Block Structuring



(a) Physical space

(c) Near a fictitious corner

(b) Computational space

Figure 2. A Typical Block-Structured Grid.

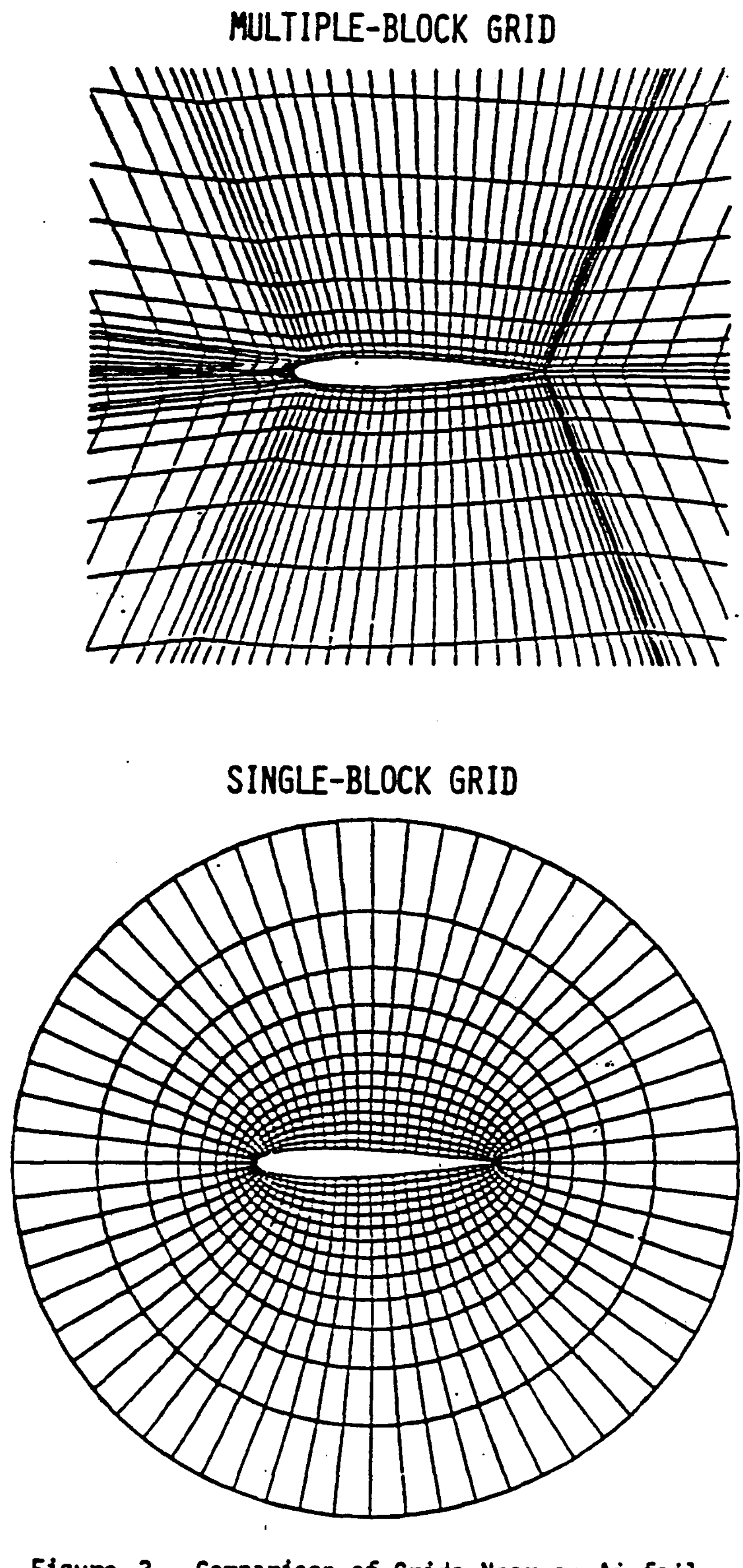


Figure 3. Comparison of Grids Near an Airfoil.

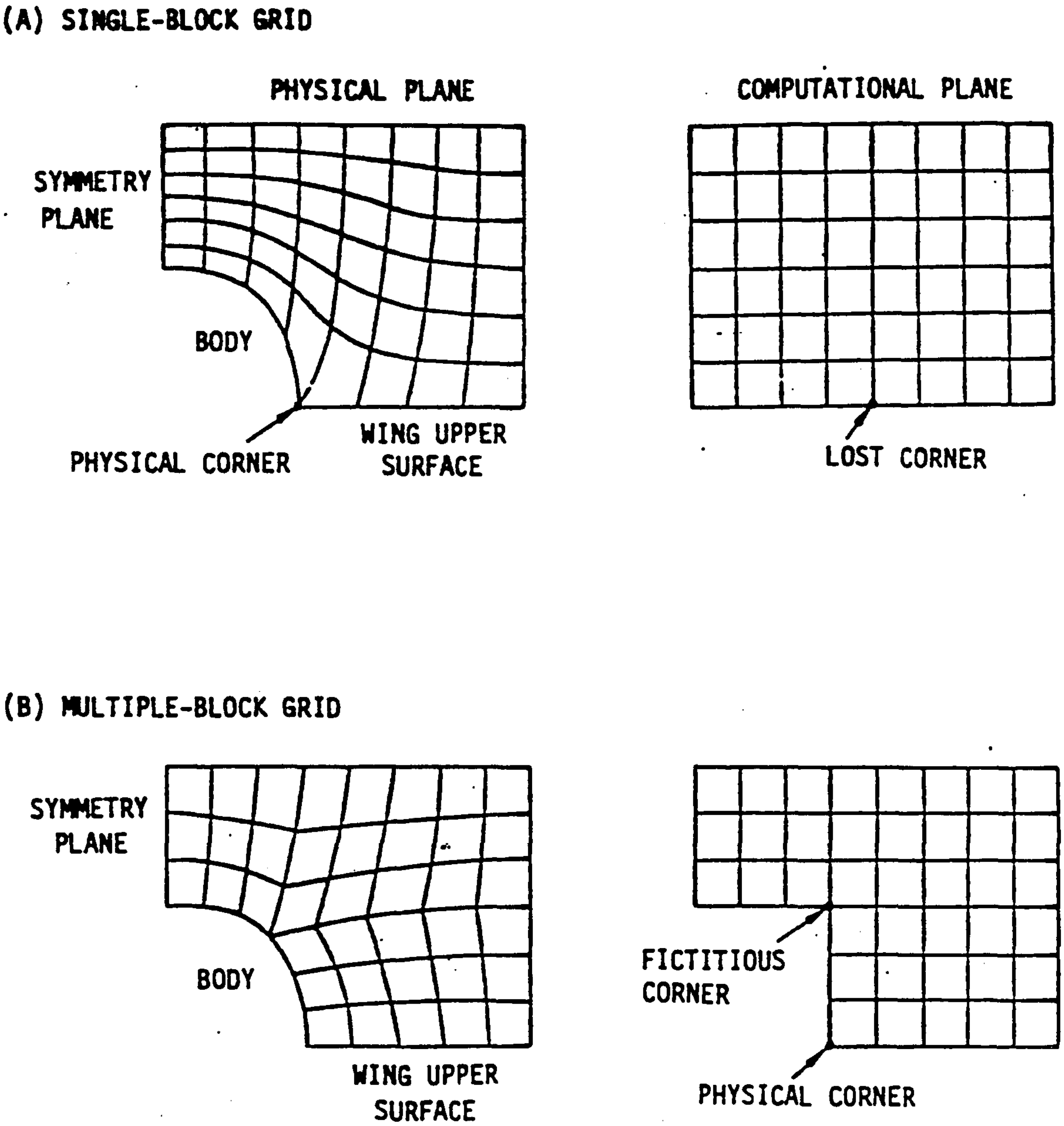


Figure 4. Comparison of Grid Structure.

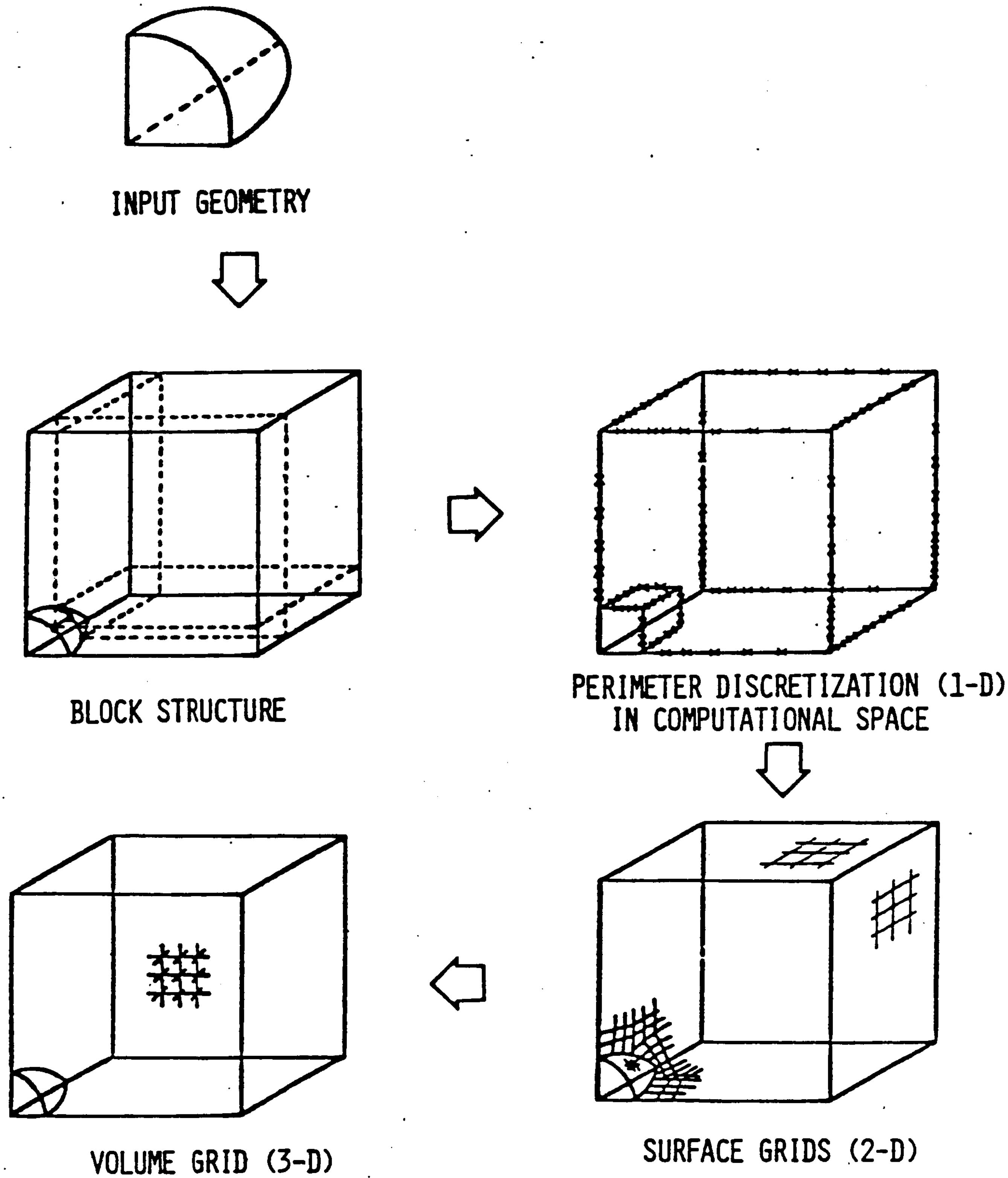


Figure 5. Block-Structured Grid Generation Process.

Connecting the cooresponding grid points between all neighboring span stations then produces a 3D grid. This code further assumes constant wing section so that the same 2D grid can be used for all span stations.

Thompson's concept of using a boundary-fitted curvilinear coordinate system requires that the curvilinear coordinates be solutions of a system of elliptic partial differential equations in the physical space subject to Dirichlet boundary conditions on all boundaries. One (curvilinear) coordinate is specified to be constant on each of the boundaries and a monotonic variation of the other coordinate along each boundary is specified. By interchanging the role of the Cartesian coordinates and the transformed coordinates, one arrives at a quasi-linear elliptic system for the Cartesian coordinates in the transformed space.

The present approach simplifies the formulation by requiring the Cartesian coordinates $x = (x, y)$ to be the solutions of three independent linear equations defined in the computational coordinates (ξ , η):

$$Ax_{\xi\xi} + Bx_{\eta\eta} + Dx_{\xi} + Ex_{\eta} = 0 \tag{1}$$

where the coefficients A to E are constants or specified functions used for grid control. The coefficients A and B provide means to locally rescale the computational coordinates. Along the block boundaries, these two coefficients may be evaluated knowing the grid distributions. The coefficients D and E are related to the source terms in Thompson's nonlinear approach (reference 4). These two coefficients can be extracted effectively from the perimeter discretization of the block boundary (reference 6) by taking the limiting form of equation (1). For example, D may be extracted from the relation.

$$Ax_{\xi\xi} + Dx_{\xi} = 0 \tag{2}$$

The coefficients D and E are linearly interpolated on a block boundary surface using the block perimeter values. The grid control coefficients A to E are then linearly interpolated inside a block to match up with the boundary surface values.

III. INPUT DATA CARDS FOR BGRID

This section describes the preparation of input data cards for the grid generation program BGRID. Assuming a user wants to analyze flow over a wing, he should provide a set of parameters for a block-structured grid. The user should also define the planform and section geometries of the wing. More information about the input variables will now be discussed.

The case title card is read in with a 10A8 FORMAT. A set of data should be prepared for the NAMELIST GRIDIN if the user wants to override the program defaults. The variable heading cards are included for clarification purposes; they may be replaced by blank cards if the user desires to do so. All the integer variables are read in with a 8I10 FORAMT and all the floating point variables are read in with a 8F10.0 FORMAT.

Table 1 Data Deck for Blocked Grid Generation

Columns Cards	1-10	11-20	21-30	31-40	41-50	51-60	61-70	71-80
Title of Case								
				Data Cards for NAMELIST GRIDIN				
Title Card 1	IBLOCK	JBLOCK	LINEAR	NTHREE	NPRINT			
Data Card 1	4	4	1	1	1			
Title Card 2	IBB							
Data Card 2	1	9	31	43				
Title Card 3	JBB							
Data Card 3	1	9	13	21				
Title Card 4	NBLOCK							
Data Card 4	2	2	2					
	2	2	-10					
	2	2	2					
Title Card 5	NCONTL							
Data Card 5	4	2	2					
	4	1	0					
	3	1	1					
Title Card 6	ILE	ITE	JLS	JUS				
Data Card 6	9	33	9	13				
Tilte Card 7	KMIN	KTIP	KMAX					
Data Card 7	1	11	17					

Table 1 (Continue)

Tilte Card 8	SWEEP	ZTIP	ZFAR	
Data Card 8	30.0	3.0	9.0	
Tilte Card 9	CHORD	SPAN		
Data Card 9	1.0	0.0		
	1.0	0.1		
	1.0	0.2		
	1.0	0.3		
	1.0	0.4		
	1.0	0.5		
	1.0	0.6		
	1.0	0.7		
	1.0	0.8		
	1.0	0.9		
	1.0	1.0		
Title Card 10	FSYM	FNU	FNL	FMESH
Data Card 10	1.0	72.0	72.0	1.0
Title Card 11	TRIAL	SLOPT	XSING	YSING
Data Card 11	14.12	0.0	0.07	0.0
Title Card 12	XB	YB		
Data Card 12	72 cards for upper surface coordinates			
Title Card 13	Not used			
Data Card 13	Not used			
Title Card 14	XX			
Data Card 14	-10.0	1.5	1.5	11.0

Table 1 (Continue)

Title Card 15	YY			
Data Card 15	-10.0	-0.3	0.3	10.0
Title Card 16	DELI	DELJ		
Data Card 16	0.03	0.015		

Description of NAMELIST GRIDIN Parameters

MAXIT	Maximum number of iterations in the grid generation. Default value is 100.
NOUT6	Interval of iterations for printout of transitory evolution of coordinates. Generally, the additional information will not be needed and the recommended value for NOUT6 is MAXIT+1. Default value is 101.
IINCR	Print the coordinates in the I-direction for every IINCR intervals. Default value is 4.
JINCR	Print the coordinates in the J-direction for every JINCR intervals. Default value is 4.
LPLOT	When generating the grid plot file, the program stores the coordinates for every LPLOT point in both the I- and J-directions. Default value is 1.
OMEG	The overrelaxation factor in the relaxation of the grid generation equations. Default value is 1.5
ERR	The convergence tolerance for the maximum corrections in the x-coordinates and the y-coordinates. Default value is 0.001.

Data Card 1

Column	Variable	Format	Explanation
1-10	IBLOCK	I10	Number of block boundaries in I-direction.
11-20	JBLOCK	I10	Number of block boundaries in J-direction.
21-30	LINEAR	I10	Parameter to select grid generation equations 1 use linear grid generation equations 0 use nonlinear grid generation equations

31–40	NTHREE	I10	Parameter to select 2D or 3D grids 1 generate 3D wing grid 0 generate 2D airfoil grid
41–50	NPRNT	I10	Parameter to control print out 1 print convergence history 2 print convergence history and block boundary grids 3 print convergence history and block boundary and 2D grids

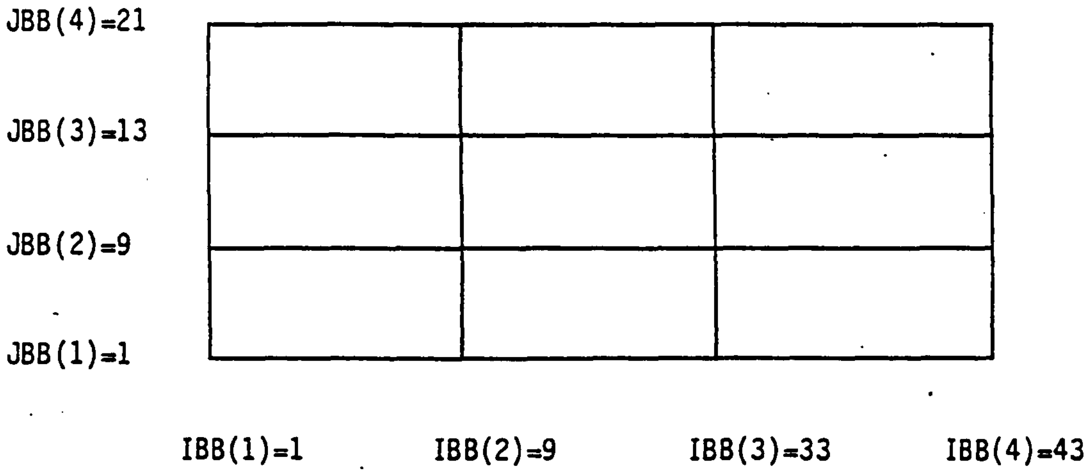
Data Card 2

Column	Variable	Format	Explanation
1–11 . . 71–80	IBB	I10	Grid point indices of the block boundaries in I-direction

Data Card 3

Column	Variable	Format	Explanation
1–11 . . 71–80	JBB	I10	Grid point indices of the block boundaries in J-direction.

Example



This sketch illustrates how to use the IBB and JBB arrays to identify the I,J indices of the block boundaries in the global grid systems.

Data Card 4

Column	Variable	Format	Explanation
1–10	NBLOCK	I10	NBLOCK denotes the block type I.D. number. The domain is filled with blocks and only some of them are filled with flow or wetted by the flow.
.			1 block inside body (wetted by the flow)
.			2 flow field block (filled by the flow)
71–80			–10 block inside wake

Example

2	2	2
2	1	–10
2	2	2

This sketch illustrates typical block I.D. to be used for flow over an airfoil.

Data Card 5

Column	Variable	Format	Explanation
1-10 . . 71-80	NCONTL	I10	Index to select the grid control parameters for the interior grid points using block boundary discretizations 0 linear variation between boundaries 1 left and lower boundaries 2 left and upper boundaries 3 right and lower boundaries 4 right and upper boundaries

Data Card 6

Column	Variable	Format	Explanation
1-10	ILE	I10	I-index for wing leading edge
11-20	ITE	I10	I-index for wing trailing edge
21-30	JLS	I10	J-index for wing lower surface
31-40	JUS	I10	J-index for wing upper surface

Data Card 7

Column	Variable	Format	Explanation
1-10	KMIN	I10	K-index (spanwise) for wing root station
11-20	KTIP	I10	K-index for wing tip station

21-30	KMAX	I10	K-index for far field station in the spanwise direction

Data Card 8

Column	Variable	Format	Explanation
1-10	SWEEP	F10.0	Wing sweep angle in degree
11-20	ZTIP	F10.0	z-coordinate (spanwise) for wing tip station
21-30	ZFAR	F10.0	z-coordinate (spanwise) for far field station

Data Card 9

Column	Variable	Format	Explanation
1-10	CHORD	F10.0	Local chord length
11-20	SPAN	F10.0	Normalized spanwise coordinates

Data Card 10

Column	Variable	Format	Explanation
1-10	FSYM	F10.0	Indicator to indicate whether or not the wing section is symmetrical 0.0 nonsymmetry 1.0 symmetry
11-20	FNU	F10.0	Number of points to define the wing section upper surface
21-30	FNL	F10.0	Number of points to define the wing section lower surface
31-40	FMESH	F10.0	Use 1.0

Data Card 11

Column	Variable	Format	Explanation
1-10	TRAIL	F10.0	Trailing edge closure angle
11-20	SLOPT	F10.0	Trailing edge bisector slope
21-30	XSING	F10.0	x-coordinate of the leading edge singular point, consistent with the wing-section defining data in data card 8
31-40	YSING	F10.0	y-coordinate of the leading edge singular point, consistent with the wing-section defining data in data card 8

Data Card 12

Data card 12 contains as many cards as needed to define the wing upper surface section geometries.

Column	Variable	Format	Explanation
1-10 21-30 41-50 61-70	XB	F10.0	x-coordinate of the wing section defining points (upper surface)
11-20 31-40 51-60 71-80	YB	F10.0	y-coordinate of wing section defining points (upper surface)

Data Card 13

Data Card 13 should be included only if the corresponding wing section is unsymmetrical (FSYM = 0.0 in data card 10). It should contain as many cards as needed to define the wing lower surface section geometries.

Column	Variable	Format	Explanation
1-10 21-30 41-50 61-70	VAL	F10.0	x-coordinates of the wing section defining points (lower surface)
11-20 31-40 51-60 71-80	DUM	F10.0	y-coordinate of the wing section defining points (lower surface)

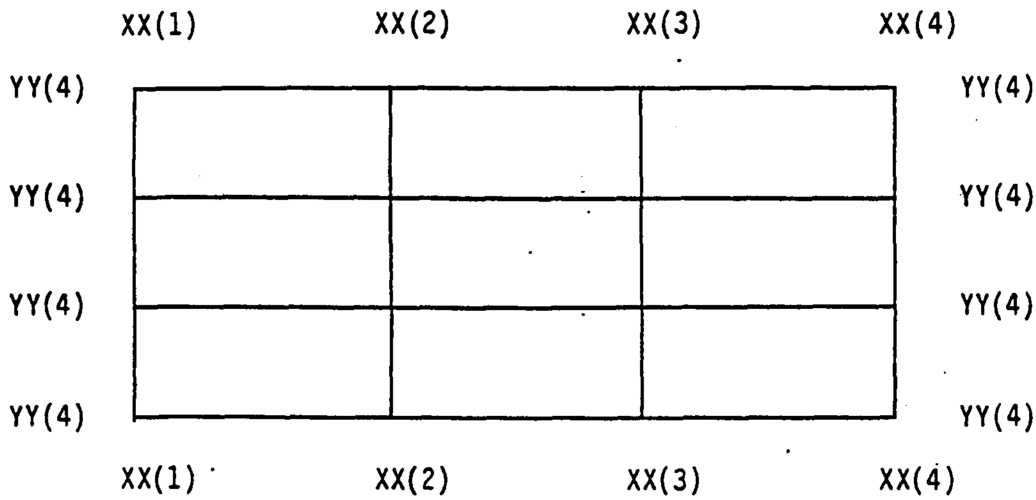
Data Card 14

Column	Variable	Format	Explanation
1-11 . . 71-80	XX	F10.0	Block boundary x-coordinates at a far field boundary

Data Card 15

Column	Variable	Format	Explanation
1-11 . . 71-80	YY	F10.0	BLock boundary y-coordinates at a far field boundary

Example



This sketch illustrates how to use the XX and YY arrays to define the block boundary coordinates along far field boundaries.

Data Card 16

Column	Variable	Format	Explanation
1-10	DELI	F10.0	Minimum grid spacing in I-direction
11-20	DELJ	F10.0	Minimum grid spacing in J-direction

IV. OUTPUT INTERPRETATION FOR BGRID

The first page of output lists the parameters of NAMELIST GRIDIN. The second page prints the block structuring information. The third page prints wing planform definition and the next three pages print the wing section defining geometries. The mapped coordinates are printed in the next two pages. Information about control parameters and grid convergence histories and coordinates are printed for each block in the following pages. An example for a block (IB=4, JB=4) is demonstrated and grid information for other blocks is omitted because of its length.

The grid is stored on TAPE4 for transonic potential flow calculations and a plot file of the grid is stored on TAPE7 for graphical displays. Figure 6 illustrates a block-structured grid for a 3D wing.

```
$GRIDIN
MAXIT   = 200,
NOUT6   = 201,
IINCR   = 1,
JINCR   = 1,
LPLOT   = 1,
OMEG    = .17+01,
ERR     = .1E-03,
$END
```

IBLOCK	JBLOCK	LINEAR	NTHREE	PRRNT
4	4	1	1	1

*** NBLOCK ***

		IBB	IBB	IBB	IBB
	1	7	33	43	
JBB	[illegible]				
JBB	[illegible]	2	2	2	
JBB	13	2	2	-10	
JBB	21	2	2	2	

*** NCONTL ***

		IBB	IBB	IBB	IBB
	1	7	33	43	
JBB	1				
JBB	[illegible]	4	2	2	
JBB	13	4	1	0	
JBB	21	3	1	1	

23

IMIN	IMAX	JMIN	JMAX
1	43	1	[illegible]

ILS	ITE	JLS	JUS
[illegible]	33	9	13

KMIN	KTIP	KMAX
1	11	17

SWEEP	Z TIP	Z FAR
30.000	3.00000	[illegible].00000

K	CHORD	SPAN
1	1.00000	0.00000
2	1.00000	.10000
3	1.00000	.20000
4	1.00000	.30000
5	1.00000	.40000
6	1.00000	.50000
7	1.00000	.60000
8	1.00000	.70000
9	1.00000	.80000
10	1.00000	.90000
11	1.00000	1.00000

.015617	-.020622
.012443	-.018754
.009391	-.016898
.007711	-.015095
.005775	-.013371
.004570	-.011742
.003433	-.010216
.002544	-.008796
.001456	-.007478
.001217	-.006260
.000766	-.005134
.000551	-.004096
.000323	-.003138
.000169	-.002255
.000070	-.001442
.000017	-.000691
0.000000	0.000000
.000017	.000691
.000070	.001442
.000169	.002255
.000323	.003138
.000551	.004096
.000766	.005134
.001217	.006260
.001456	.007478
.002544	.008796
.003433	.010216
.004570	.011742
.005775	.013371
.007711	.015095
.009391	.016898
.012443	.018754
.015617	.020622
.019351	.022455
.024177	.024299
.029701	.025825
.036476	.027332
.044415	.028791
.054125	.030324
.065650	.032014
.079337	.034837
.095535	.036774
.114799	.037792
.137976	.039852
.164919	.041907
.191335	.044621
.218710	.045051
.245331	.046236
.271799	.047199
.298111	.047949
.324273	.048490
.350213	.048911
.376145	.048930
.401457	.048820
.427422	.048488
.452944	.047935
.478120	.047166
.503251	.046193
.52[illegible]	.044021
.55[illegible]	.043673

.015617	-.021622
.012443	-.019754
.009341	-.016898
.007711	-.015095
.005775	-.013371
.004570	-.011742
.003423	-.010216
.002544	-.008796
.001836	-.007478
.001237	-.006260
.000866	-.005134
.000551	-.004096
.000323	-.003138
.000169	-.002255
.000070	-.001442
.000017	-.000691
0.000000	0.000000
.000017	.000691
.000070	.001442
.000169	.002285
.000323	.003138
.000551	.004096
.000866	.005134
.001237	.006260
.001836	.007478
.002544	.008796
.003423	.010216
.004570	.011742
.005775	.013371
.007711	.015095
.009341	.016898
.012443	.019754
.015617	.021622
.019351	.022455
.024177	.023200
.029701	.025925
.036476	.027332
.044415	.028791
.054175	.030329
.065611	.032014
.079337	.034837
.095635	.035779
.114939	.037792
.137976	.039852
.164979	.041907
.191533	.043621
.219710	.045051
.245331	.046236
.271739	.047199
.298111	.047949
.324273	.048490
.350273	.048813
.376195	.048939
.401457	.048820
.427492	.048489
.452844	.047935
.478120	.047166
.503251	.046190
.528233	.045021
.553[illegible]	.043674

.577[illegible]	.[illegible]216
.[illegible]0[illegible]	.[illegible]0 524
.[illegible]2[illegible]	.[illegible]761
.[illegible]511[illegible]	.[illegible]5[illegible]
.[illegible]75[illegible]	.[illegible]34954
.[illegible]9[illegible]	.[illegible]2941
.723[illegible]	.[illegible]0966
.744[illegible]	.[illegible] 737
.77[illegible]	.[illegible]6551
.7591[illegible]	.[illegible]4305
.[illegible]174[illegible]	.[illegible]1[illegible]
.[illegible]4[illegible]	.[illegible]5[illegible]
.[illegible]65[illegible]	.[illegible]7092
.[illegible]86[illegible]	.[illegible]14505
.[illegible]061 1	.[illegible]1223[illegible]
.[illegible]27[illegible]	.[illegible]10275
.[illegible]36[illegible]	.[illegible]543
.[illegible]47[illegible]	.[illegible]7142
.[illegible]57[illegible]1	.0[illegible]5922
.[illegible]66[illegible]	.0[illegible]4891
.[illegible]73[illegible]	.[illegible]01[illegible]
.[illegible]7[illegible]	.[illegible]328[illegible]
.[illegible]84[illegible]1	.[illegible]655
.[illegible]8[illegible]	.[illegible]2126
.[illegible]92144	.[illegible]167
.[illegible]95[illegible]	.0 123[illegible]
.[illegible]97[illegible]	.0 1577
1.[illegible]	.[illegible]07[illegible]

[illegible]SIN	[illegible]Y ING	[illegible]TE [illegible]LOPE	[illegible]E ANGLE
.[illegible]07[illegible]	0.[illegible]00[illegible]	[illegible].0[illegible]0[illegible]	14.120000

27

MAPPED COORDINATES

A	S	XR	YR
-.625[illegible]	.0[illegible]0222	1.000000	-.000705
-.600[illegible]62	.0[illegible]3262	.925076	-.009963
-.576[illegible]3	.0[illegible]6230	.853045	-.018270
-.552[illegible]5	.0[illegible]899[illegible]	.783915	-.025287
-.528[illegible]46	.011663	.717690	-.031350
-.504[illegible]	.01[illegible]2[illegible]	.654369	-.036642
-.480[illegible]	.016820	.553957	-.041103
-.456[illegible]31	.01920[illegible]	.536464	-.044595
-.432[illegible]2	.021385	.481903	-.047032
-.40[illegible]	.023319	.430283	-.048437
-.384615	.025026	.381611	-.048925
-.360[illegible]	.026552	.335835	-.048663
-.336[illegible]	.027957	.293102	-.047822
-.312[illegible]	.029304	.253255	-.046546
-.288462	.030646	.216336	-.044934
-.264[illegible]	.032027	.182340	-.043045
-.240[illegible]5	.033484	.151257	-.040912
-.216346	.035064	.123079	-.038558
-.192[illegible]	.036843	.097788	-.036013
-.16[illegible]	.038965	.075351	-.033327
-.144[illegible]1	.041695	.055700	-.030567
-.120[illegible]2	.045461	.039712	-.027773
-.0961[illegible]	.049751	.024457	-.024315
-.0[illegible]	.0[illegible]27[illegible]	.013397	-.019347
-.04[illegible]	.0[illegible]73[illegible]	.005769	-.013147
-.024[illegible]	.053627	.001410	-.006552
0.00[illegible]	.052411	0.000000	0.000000
.024[illegible]	.053627	.001410	.006552
.04[illegible]	.052738	.005769	.013147
.072115	.052750	.013397	.019347
.0961[illegible]4	.049751	.024457	.024315
.120[illegible]2	.045461	.03[illegible]712	.027773
.144231	.041695	.055700	.030567
.16[illegible]	.038965	.075351	.033327
.192[illegible]	.036843	.097788	.036013
.216346	.035064	.123079	.038558
.240[illegible]5	.033484	.151257	.040912
.264423	.032027	.182340	.043045
.288462	.030646	.216336	.044934
.312[illegible]	.029304	.253255	.046546
.336[illegible]	.027957	.293102	.047822
.36[illegible]	.026552	.335885	.048663
.384615	.025025	.381611	.048925
.40[illegible]54	.023319	.430283	.048437
.432[illegible]2	.021385	.481903	.047032
.456731	.019209	.536464	.044595
.480769	.016820	.593957	.041103
.504[illegible]	.014280	.654369	.036642
.528[illegible]46	.011663	.717690	.031350
.552[illegible]5	.008998	.783915	.025287
.576[illegible]	.006230	.853045	.018270
.600[illegible]2	.003262	.925076	.009963
.625[illegible]0	.0[illegible]0222	1.000000	.000705

```
TIME BEFORE  ETU[illegible]  =     3.7[illegible]8   SECONDS
TIME AFTER   BOUND   =     3.[illegible]5   SECONDS
TIME USED IN BOUND   =      .0[illegible]7   SECONDS
```

29

```
FOR BLOCK: IB= [illegible]     JB= 4
       I1= 33    I2= 43          J1= 13    J2= 21

CONTROL PARAMETERS :

SI (I,1) FOR I1P,I2M   34   42
   -.00010    -.24239    -.24239    -.24239    -.24239    -.24239    -.24239    -.24239    -.24239

SI (I,2) FOR I1P,I2M   34   42
   -.00009    -.24239    -.24239    -.24239    -.24239    -.24239    -.24239    -.24239    -.24239

SJJ(I,1) FOR I1P,I2M   34   42
    .02245     .03912     .10520     .28291     .76081    2.04598    5.50206   14.79618   39.78973

SJJ(I,2) FOR I1P,I2M   34   42
    .03151     .13343     .36849     .74027    1.48713    2.98752    6.00166   12.05682   24.22104

SJ (J,1) FOR J1P,J2M   14   20
   -.40085    -.40085    -.40015    -.40085    -.40085    -.40085    -.40085

SJ (J,2) FOR J1P,J2M   14   20
   -.40085    -.40015    -.40015    -.40015    -.40015    -.40085    -.40015

SII(J,1) FOR J1P,J2M   14   20
    .00351     .01374     .07511     .41057    2.24436   12.26882   67.06759

SII(J,2) FOR J1P,J2M   14   20
    .01769     .06755     .25810     .98620    3.76819   14.39798   55.01353
```

30

CONVERGENCE HISTORY OF SURFACE GRID
I1,I2,J1,J2 = 33 43 13 21

IT	CXMAX	I	J	RXMAX	I	J	CYMAX	I	J	RYMAX	I	J
1	-.16163E+00	39	20	-.80636E+01	42	20	.22031E+00	42	20	.34264E+02	42	20
2	-.53943E-01	41	20	-.31547E+01	42	20	.54733E-01	42	19	.74706E+01	42	20
3	-.23343E-01	41	19	-.13242E+01	42	20	.37967E-01	42	19	.31263E+01	42	19
4	-.13573E-01	41	19	-.83910E+00	42	19	.22354E-01	42	19	.16461E+01	42	19
5	-.10060E-01	41	19	-.52932E+00	42	19	.13878E-01	42	19	.92995E+00	42	19
6	-.66968E-02	41	19	-.34125E+00	42	19	.85847E-02	42	18	.63935E+00	42	18
7	-.46073E-02	41	19	-.22774E+00	42	19	.71853E-02	42	18	.45725E+00	42	18
8	-.33073E-02	41	18	-.16473E+00	42	19	.54335E-02	42	18	.33408E+00	42	18
9	-.25900E-02	41	18	-.12632E+00	42	18	.41852E-02	42	18	.24953E+00	42	18
10	-.20331E-02	41	18	-.97910E-01	42	18	.32720E-02	42	18	.19019E+00	42	18
11	-.16193E-02	41	18	-.76803E-01	42	18	.25923E-02	42	18	.14756E+00	42	18
12	-.13010E-02	41	18	-.60960E-01	42	18	.20775E-02	42	18	.11625E+00	42	18
13	-.10555E-02	41	18	-.48917E-01	42	18	.16814E-02	42	18	.93853E-01	42	17
14	[illegible]	41	18	-.39640E-01	42	18	.13720E-02	42	18	.77193E-01	42	17
15	[illegible]	41	18	-.32400E-01	42	13	.11273E-02	42	18	.63809E-01	42	17
16	[illegible]	41	18	-.26673E-01	42	13	.93145E-03	42	18	.52962E-01	42	17

17	-.[illegible]E-03	41	18	-.[illegible]10[illegible]E-01	42	[illegible]	.77326E-03	42	18	.44114E-01	42	17
18	-.41022[illegible]E-03	41	18	-.1[illegible]429E-01	42	18	.64440E-03	42	18	.36853E-01	42	17
19	-.3442[illegible]E-03	41	18	-.15416E-01	42	18	.53870E-03	42	18	.30861E-01	42	17
20	-.[illegible]E-03	41	18	-.1235[illegible]E-01	42	18	.45150E-03	42	18	.25896E-01	42	17
21	-.[illegible]434E-03	41	18	-.1[illegible]912E-01	42	18	.37921E-03	42	18	.21766E-01	42	17
22	-.[illegible]E-03	41	18	-.[illegible]2187E-02	42	18	.31904E-03	42	18	.18319E-01	42	17
23	-.1[illegible]477[illegible]E-03	41	18	-.7[illegible]041E-02	42	18	.26887E-03	42	17	.15436E-01	42	17
24	-.1[illegible]E-03	41	18	-.6[illegible]177E-02	42	18	.22700E-03	42	17	.13018E-01	42	17
25	-.1[illegible]565[illegible]E-03	41	18	-.56226E-02	42	17	.19175E-03	42	17	.10989E-01	42	17
26	-.1[illegible]E-03	41	18	-.47947E-02	42	17	.16205E-03	42	17	.92801E-02	42	17
27	-.[illegible]E-04	41	18	-.4[illegible]904E-02	42	17	.13700E-03	42	17	.78422E-02	42	17
28	-.[illegible]E-04	41	18	-.31907E-02	42	17	.11586E-03	42	17	.66301E-02	42	17
29	-.[illegible]E-04	41	18	-.2[illegible]79[illegible]E-02	42	17	.98010E-04	42	17	.56076E-02	42	17

```
TIME BEFORE MESH  =     3.[illegible]95   SECONDS
TIME AFTER  MESH  =     4.564   SECONDS
TIME USED IN MESH =      .669   SECONDS
```

GOOD GRID - NO CROSS-OVER

```
TIME BEFORE MARCO  =    4.[illegible]   SECONDS
TIME AFTER  MARCO  =    9.106   SECONDS
TIME USED IN MARCO =    4.542   SECONDS
```

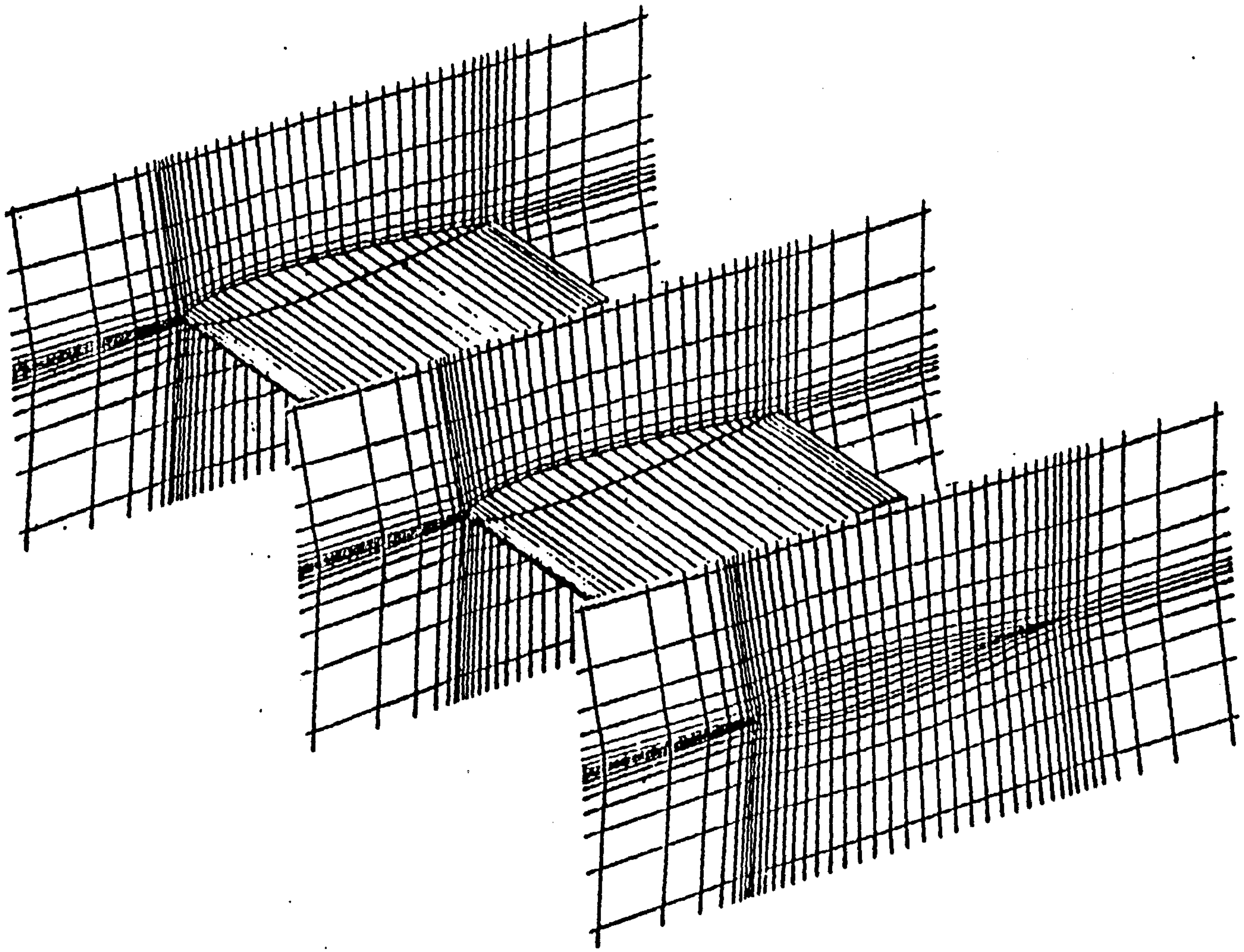


Figure 6. A Block-Structured Grid for a 3-D Wing.

V BGRID PROGRAM INFORMATION

This chapter contains description of the major program modules of the computer program BGRID for blocked grid generation. BGRID consists of 15 program modules: a main program (GRID) which contains the block-structured grid generation logic; several major subroutines which provide for block structuring, geometry specifications, grid initialization, grid relaxation, grid checking and grid output (SETUP, BOUND, MESH, MAPCO, STORE, DISC, INIGRID, CONTROL, RELGRID, SWEEP, GOUT); and three minor subroutines (INTPL, SPLIT, TRIMAT) which perform simpler functions, namely; interpolation, spline fitting, and solution of a system of tridiagonal equations. Each of these routines along with a brief description of its function is listed in table 2. A subroutine tree showing the relation of each routine to its called and calling routines is displyed in Figure 5. This information should be referred to as needed during the following discussion.

Description of MAIN Program GRID

This is the driver program for wing block-structured grid generation using linear equations. The five subroutines, SETUP, BOUND, MESH, MAPCO and STORE are called by the main program sequentially. These five subroutines will be described in the following.

Description of SUBROUTINE SETUP

This subroutine reads in information for block-structuring. It also reads in geometric data for wing definition. The overall block-structuring should be set up by the user following the natural lines of the configuration.

Description of SUBROUTINE BOUND

This subroutine computes the block perimeters based on the information obtained from subroutine SETUP. The wing data is used to discretize wing surface using a parabolic transformation. SUBROUTINE DISC is called to discretize other block perimeter beside the wing surface.

Description of SUBROUTINE MESH

This subroutine generates the computational grids in a block-by-block manner. For each block, SUBROUTINE INIGRID is called to provide an initial guess for the cartesian coordinates of the grid points. SUBROUTINE CONTROL is then called which computes the control parameters in the grid generation equations. Finally, the equations are solved through successive line overrelaxation (SLOR) by calling the SUBROUTINE RELGRID.

Description of SUBROUTINE MAPCO

This subroutine computes the Jacobian of each grid cell. MAPCO then checks to see if all the Jacobian are positive. Affirmative results indicates a useful grid with no cross-over. MAPCO will generate a warning message of BAD GRID-CROSS-OVER if any Jacobian becomes zero or negative.

Description of SUBROUTINE STORE

This subroutine saves the cartesian coordinates of the grid points in TAPE4 to be used by the TWINGB code. The subroutine also saves a plot file for the grid in TAPE7.

Table 2 Subroutine Reference Chart

Routine name	Calling routine	Routines called	Description
BOUND	GRID	INTPL, SPLIF, EDGER	Calculates block perimeters
CONTROL	MESH		Computes control parameters
DISC	BOUND		Discretizes one block perimeter
GOUT	RELGRID		Prints grids
INIGRID	MESH		Initiates grids
INTPL	BOUND		Interpolation
MAPCO	GRID		Checks the Jacobian of the grids
MESH	GRID	INIGRID,CONTROL RELGRID	Generates grids
RELGRID	MESH	SWEEP, GOUT	SLOR of grid equations
SETUP	GRID		Reads block structure
SPLIF	BOUND		Spline fitting
STORE	GRID		Saves grid information
SWEEP	RELGRID		Performs one sweep in the SLOR of grid equations
TRIMAT	SWEEP		Tridiagonal solver

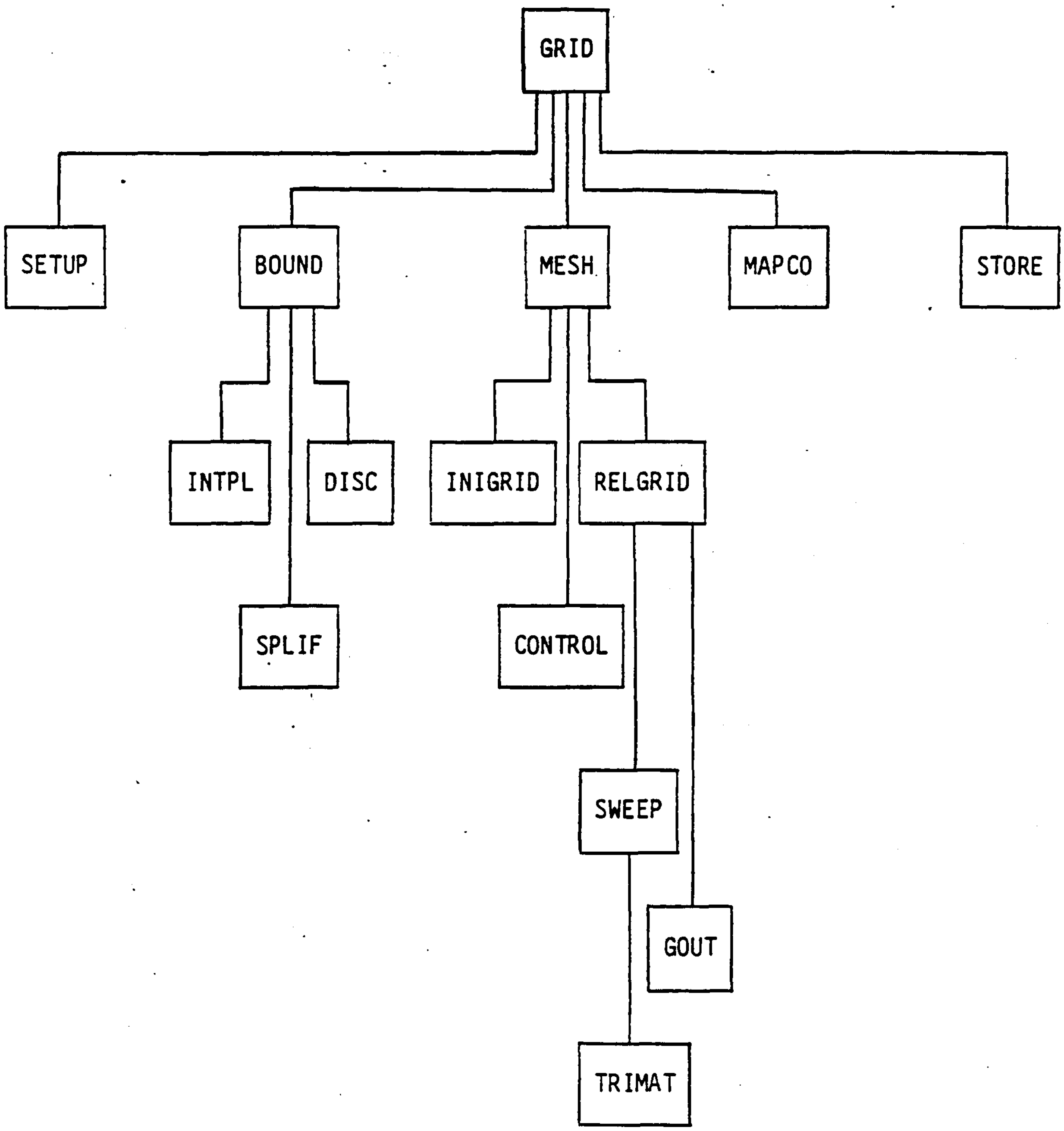


Figure 5. BGRID SUBROUTINE Tree

VI APPLICATION OF THE TWINGB CODE TO BLOCK-STRUCTURED GRIDS

The computer program TWINGB (reference 3) uses a fast, fully implicit AF algorithm to solve the conservative full potential equation for transonic flow about arbitrary wings between walls. The basic TWINGB code uses single-block ring-grid. Current study adapts TWINGB to block-structured grids.

In the ring-grid I = 1 and I = NI corresponding to the upper and lower surfaces of the vortex sheet, respectively. A periodic boundary condition is imposed to equate the values of the flow variables on the two sheets. In the block-strucutred grid I = 1 and I = NI corresponding to the upstream farfield and downstream farfield boundaries. Dirichlet boundary conditions are imposed along these farfield boundaries.

Consideration of solutions accuracy requires that the mapping procedure between the physical and computational spaces be fully consistent. Specifically, the same approximation formulas should be used for the differencing of x,y,z and ϕ in the computational space. The basic TWINGB uses second order formulas to difference ϕ and uses fourth order formulas to difference x, y and z. This inconsistent procedure could provide satisfactory results for smoothly varying grids such as a simple ring-grid. But it will produce problems for general grids such as block-structured grids. The modified TWINGB uses the same second order formulas to difference x, y, z and ϕ.

Following the concept of isoparametric mapping, the difference formulas for x, y, z and ϕ may be derived by differentiating the interpolators for x, y, z and ϕ. Same subdomain definition and interpolation formulas should be used for x, y, z and ϕ. This fully consistent isoparametric mapping procedure will further improve the accuracy in the TWINGB flow solution algorithm but this procedure has not yet been implemented into the program.

The complexity in the general block-structured grids calls for more program logics in the flow solver. Therefore, it is desirable to simplify the TWINGB code whenever possible. To maintain high computational efficiency including program vectorizability, it is necessary to remove the IF statement from a DO-LOOP. To achieve this goal, calculations of flux quantities at different types of boundaries should be treated differently by a separate section of coding. This generates a large number of program instructions in the basic TWINGB code. Many of these instructions are only slightly different and repetition of nearly identical statements is error prone. In the modified TWINGB, these program instructions are packaged as statement functions. The use of nearly identical statements are eliminated by calling the required statement function. This also provides a better clarification of the program. Since the meaning of a statement function call can easily be associated with its name. An advantage in using the statement function over the conventional subroutine is that the statement function is an inline function. This means that at the compiling time, the effect of a statement function call is to replace the call statement by the corresponding program instructions which define the statement function. This would prevent the program from consuming CPU time in address transferring associated with a typical non-inline function call. Thus the modified TWINGB code maintains both computational efficiency and program simplification and clarification.

VII OPERATION OF TWINGB ON BLOCK-STRUCTURED GRIDS

This chapter discusses the use of the modified TWINGB computer program on a block-structured grid including an input description for the program. Most of the parameter inputs to the modified TWINGB are contained in NAMELIST INP (defined in SUBROUTINE INPUT). These data include the flow parameters (freestream Mach number, angle of attack), and the aerodynamic and flow fiel convergence parameters. Default values for the NAMELIST parameters are intialized in INPUT. With the NAMELIST format, the user needs to change only the values that differ from the default values. The grid information is read in from TAPE10.

Description of NAMELIST INP Parameters

ALPHAN	Wing angle of attack in degree. Default value is 0.
MACH	Freestream Mach number. Default value is 0.72.
ALPHAH	Upper bound for the sequence of acceleration parameters in the AF scheme. Default value is 4.0.
ALPHAL	Lower bound for the sequence of acceleration parameters in the AF scheme. Default value is 0.40.
BETA	Temporal damping coefficient in the ETA direction. Default value is 0.0.
BXI	Temporal damping coefficient in the XI direction. Default value is 0.0.
MAXIT	Maximum number of iterations allowed per case. No default value given. It must be supplied by users.
OMEGA	Relaxation parameter in the iteration scheme. OMEGA should be bounded between 1 and 2 inclusively. Default value is 1.8.

CON	Parameter controlling upwind bias of density. CON should be bounded between 1 and 2 inclusively. Default value is 1.5.
M	Number of elements in ALPHA sequence. M must be no less than 2. Default value is 8.
KKK	Starting value of ALPHA sequence. KKK may vary from 0 to M. Default value is 0.
BB	Wing span calculated from the grid.
S	Same as BB.
IOUT	Output option parameter. Should set a default value of 0 to print the density and Mach number and plot C_p.
JOKIP	Skip parameter in space for output plots.
INTOUT	Skip parameter in time for output plots. Printed after convergence only if set to 0. Default value is 0.

IX. REFERENCES

1. Lee, K. D., "3-D Transonic Flow Computations Using Grid Systems with Block Structure", AIAA, Fifth Computational Fluid Dynamic Conference, Palo Alto, California, June 22-23, 1981.

2. Lee, K. D., and Rubbert, P. E., "Transonic Flow Computations Using Grid Systems with Block Structure", Proceedings, the 7th International Conference on Numerical Methods in Fluid Dynamics, Stanford University, California, June, 1980.

3. Holst, T. L., 1979, "A Fast, Conservative Algorithm for Solving the Transonic Full-Potential Equation", AIAA, Vol. 18, 1980, pp. 1431-1439.

4. Thompson, J. F., Thames, F. C., and Mastin, C. M., "Automatic Numerical Generation of Body-Fitted Curvilinear Coordinate System for Field Containing Any Number of Arbitrary Two Dimensional Bodies", Journal of Computational Physics,, Vol. 15, 1974, pp 299-319.

5. Yu, N. J., "Grid Generation and Transonic Flow Calculations for Three-Dimensional Configurations", AIAA Paper 80-1391, AIAA 13th Fluid and Plasma Dynamics Conference, Snowmass, Colorado, July, 1980.

6. Middlecoff, J. F. and Thomas, P. D., "Direct Control of the Grid Point Distribution in Meshes Generated by Elliptic Equations", AIAA Paper 79-1462, 1979.

1. Report No.: NASA CR-166317

2. Government Accession No.:

3. Recipient's Catalog No.:

4. Title and Subtitle: BGRID : A Block-Structured Grid Generation Code For Wing Sections

5. Report Date: November 1981

6. Performing Organization Code:

7. Author(s): H. C. Chen and K. D. Lee

8. Performing Organization Report No.:

9. Performing Organization Name and Address: The Boeing Company, P.O. Box 3707, Seattle, WA 98124

10. Work Unit No.:

11. Contract or Grant No.: NAS2-10676

12. Sponsoring Agency Name and Address: National Aeronautics & Space Administration, Washington, DC 20546

13. Type of Report and Period Covered: Contractor Report

14. Sponsoring Agency Code:

15. Supplementary Notes: ARC Technical Monitor: Dochan Kwak, 202A-14, Applied Computational Aerodynamics Branch, NASA Ames Research Center, Moffett Field, CA 94035 (415) 965-6415 FTS 448-6415 See also NASA CR-166316

16. Abstract

The operation of the BGRID computer program is described for generating block-structured grids. Examples are provided to illustrate the code input and output. The application of a fully implicit AF (approximation factorization)-based computer code, called TWINGB (Transonic WING), for solving the 3D transonic full potential equation in conservation form on block-structured grids is also discussed.

17. Key Words (Suggested by Author(s)): Block-Structured Grid Generations; Approximation Factorization Schemes; Transonic Computational Aerodynamics

18. Distribution Statement: ~~FEDD Distribution~~ Review Date November 1984

19. Security Classif. (of this report): Unclassified

20. Security Classif. (of this page): Unclassified

21. No. of Pages: 45

22. Price*:

Avail NASA's Industrial Application Centers

End of Document