CALCULATION OF VISCOUS FLOW ABOUT A PROPELLER DRIVEN AIRPLANE

Dieqian Wang

The Aeronautical Research Institute of Sweden, FFA
Stockholm, Sweden

Abstract

Systematic application of Navier-Stokes solvers to the design of new aircraft is still limited by the extremely high costs of these methods. Therefore, there is a need to couple efficient boundary-layer solvers with inviscid flow solvers. This results in a substantial reduction of the computational effort. In the commercial airplane design environment, with attached flow and week shocks, even no considerable difference between inviscid flow and equivalent inviscid flow is generally supposed, there is a need for the use of the boundary-layer equations together with the Euler equations or the potential flow equations to predict the viscous flow. A multigrid-multiblock full potential solver with a robust algorithm code cooperating with a boundary-layer code is used here. The calculations have shown good agreement with experimental data on velocity profils, distribution of skin friction coefficient, displacement thickness and momentum thickness on a wing at a transonic Mach number with angle of attack. The application to a commercial airplane will be shown, and different topologies for calculations of a propeller driven airplane configuration will be discussed herein.

Summary

The coupling of efficient boundary-layer solvers with inviscid flow methods is part of the more general zonal calculation method in which the Navier-Stokes equations are solved in the regions of strong viscous-inviscid interaction only, most of the flow field being predicted by the coupling of efficient boundary-layer solvers with inviscid flow methods.
In a commercial airplane design environment, with attached flow and week shocks, even no considerable difference between inviscid flow and equivalent inviscid flow, there is a need for the use of the boundary-layer equations together with the Euler equations or the potential flow equations to predict the viscous flow in the near wall region.

Compared with the more general Navier-Stokes equations, the boundary-layer equations have the advantage of being of parabolic type. Therefore, they can be solved by numerical space-marching methods, while the Navier-Stokes equations have to be integrated with time-marching methods, which require enormous amounts of computer time and storage. For this reason, systematic application of Navier-Stokes solvers to the design of new aircraft is still limited by the extremely high costs and big computational effort of these methods. In practical applications, the flow over three-dimensional configurations is predicted by the use of the boundary-layer equations together with the Euler equations or the potential flow equations. The influence of the boundary-layer on the inviscid flow is then accounted for by a distribution of equivalent sources at the wall. Since the equivalent inviscid flow does not present a very steep gradient in the vicinity of the wall, the Euler or potential grid can be coarser then a Navier-Stokes grid. This results in a substantial reduction of the computational effort. On the other hand, the boundary-layer equations describing the near-wall viscous flow can be solved on very fine meshes and the CPU time is negligible compared with a Navier-Stokes solver. Therefore, there is a need to couple efficient boundary-layer solvers with inviscid flow solvers in applications.
Commercial airplanes are designed to avoid strong viscous-inviscid interaction regions, even there is not a considerable difference between the inviscid flow and the equivalent inviscid flow, it is necessary for such complicated configurations to make use of the boundary-layer equations together with the Euler equations or the potential flow equations to predict the viscous flow in the near wall region. Potential flow calculations require far less then a tenth of the CPU time of Euler calculations, and for a commercial aircraft potential model they give a good simulation.
A potential code has been developed at FFA for simulation of transonic flow around wing-body-nacelle configurations. The propeller is modelled

as an actuator disc, and the effect of the slipstream is considered in an approximate way. It is a multigrid-multiblock solver using both the incomplete lower upper decomposition and the strongly implicit procedure as the smoothing algorithm. It is fully implicit not only in all the three coordinate directions but also between the blocks. The multigrid method with the implicit algorithm brings robustness, effectiveness and good convergence. The applications have shown good agreement with wind tunnel tests for the propeller driven airplane in subsonic climb and in transonic cruise. This code gives the outer boundary condition for the boundary-layer calculations.

The boundary-layer code used here is based on the experience gained from the application of several existing three-dimensional boundary-layer solvers. Performance improvement is achieved by the choice of numerical method. The finite difference solution method is used. In the space marching at each sweep in the downstream direction the most suitable scheme is selected at each node, and the sequence of calculation is determined so as to achieve accuracy of the results and efficiency of the computer program. The resulting algorithm is very flexible, because the problem of solving the boundary-layer on the whole configuration reduces to a succession of solutions at one single normal at a time. Therefore in this procedure it is possible to continue the process around normals, where the calculation has broken down. This feature makes it very suitable for applications to three-dimensional complex configurations.

The boundary-layer calculations have shown good agreement with experimental data on the velocity profiles, distributions of skin friction coefficient, displacement thickness and momentum thickness on a wing at transonic Mach number and angle of attack. The boundary-layer calculations on the wing for a wing-body-nacelle configuration will be shown here. The results for the other components will be decribed.

The different topologies for viscous flow calculations around wing-body-nacelle configurations will be disscussed.

Numerical Method

Boundary-Layer Numerical Method

The brief description here concerns aeronautical applications.

The boundary-layer code (1) is a second order boundary-layer theory method. The code can also be used for first order boundary-layer calculations. The finite difference solution method is used. The code was devoloped for calculation of three-dimensional attached viscous flow. For turbulent flow an algebraic Baldwin-Lomax model or the Cebeci-Smith model are available.

As the system of governing equations has a dominant parabolic character, and all surfaces normal to the body surface are characteristic surfaces, the three-dimensional boundary-layer equations are solved by a space marching integration process. The three-dimensional boundary-layer equations differ from the two-dimensional boundary-layer equations in the occurrence of two coordinate directions to describe the boundary-layer development, which introduces a hyperbolic character as the subcharacteristic to the problem. Therefore, an important feature of the governing differential equations is their domain of dependence bounded by characteristics corresponding to the direction normal to the wall, the directions of the limiting surface streamline and of the external streamline (2), see Fig. 1. Following the Courant-Friedrichs-Lewy (CFL) stability condition, a stable integration can only be achieved, if the numerical domain of dependence involves the domain of dependence of the boundary-layer equations. This condition has importance for the choice of grid and of suitable numerical schemes.

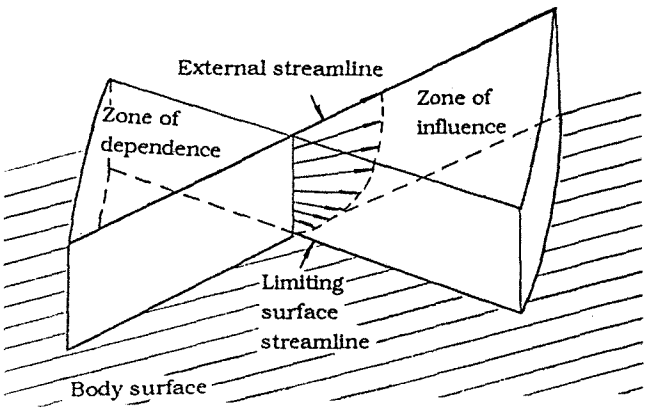

Fig. 1. Zone of dependence and zone of influence

Computationally the marching process is performed in the increasing x_1 direction, Fig. 2. As the CFL condition should be met, the x_1 coordinate lines should be aligned between the directions of the limiting surface streamline and the external streamline. Since the limiting surface streamlines are unknown before the calculation, the most convenient coordinate systems are then the ones based on the inviscid streamlines. Surface fitted coordinate systems should be used, in which x_1 and x_2 are locally parallel to the surface, and x_3 is normal

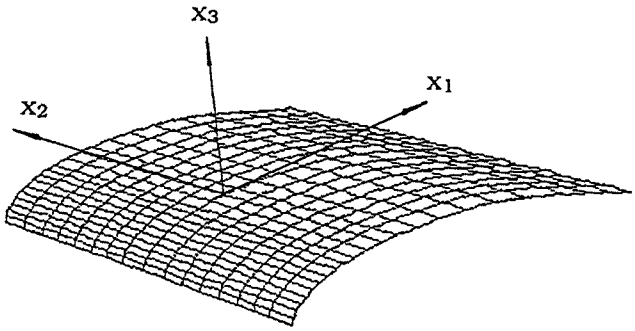

Fig. 2. Coordinate system

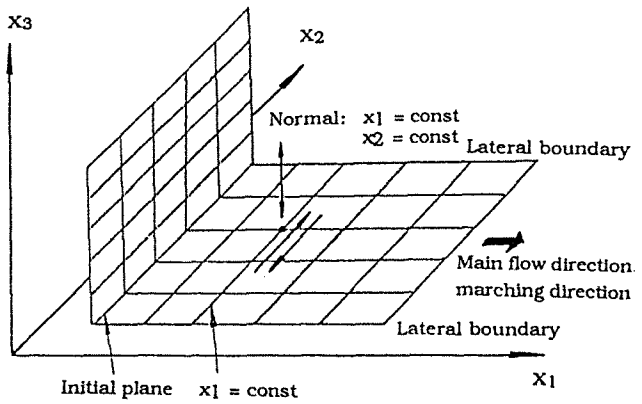

Fig. 3. Marching process

to it, see Fig. 2. In these systems the velocity coordinate v_1 should always be positive i.e. the x_1 lines are roughly aligned with the inviscid flow direction. The numerical schemes used should be stiff enough to deal with locally large crossflow components. The equations are solved at the nodes of a grid as described in Fig. 3 in the computational domain. The marching process is performed in the increasing x_1 direction, starting from an initial solution at the first plane x_1=constant, the x_1 direction referred to as the mainstream direction.

At each x_1 = constant plane, the integration is performed in the crossflow direction, i.e. along x_2. According to the flow characteristics, the numerical scheme should be able to get information in positive, negative or even both crossflow directions. Apparently, it would be very efficient to solve the equations at all the nodes simultaneously in a x_1 = constant plane by an implicit solution process. The deficiency of an implicit solution process is that the process would be terminated once breakdown of the calculation happened at one normal (x_1 = const, x_2 = const). Such a termination would not be expected especially for a complicated configuration. For this reason, in this code in the crossflow marching process, along x_2, the normals are treated successively with numerical schemes satisfying the local CFL condition. Therefore, there should be a calculation strategy to organize the crossflow marching. The solution for x_1 = const plane is then computed along each normal, i.e. along x_3. As the diffusion terms are dominant along normals, this means that information must be able to travel in both positive and negative normal directions simultaneously. The scheme therefore must be implicit, and the process solves the boundary-layer equations at all the nodes along a normal simultaneously.

The numerical schemes used are second-order accurate in discretization error using a minimum number of neighbouring nodes for tangential derivative approximations. The zig-zag, rectangular and double zig-zag schemes are used. They have different domains of stability.

The general space marching integration process has been described above. One can see that the calculation strategy to organize the crossflow marching is of importance for the efficiency and feasibility of the method. The choice of the most suitable numerical scheme at each normal determines the convergence of the iterative solution of the governing equations. Whereas, the sequence of calculation of the normals can not be chosen arbitrarily, since information on one neighbouring normal lying on the new x_1 coordinate is needed for zig-zag and rectangular schemes. The calculation strategy is then defined as follows; determining the stable direction of cross flow integration for each normal, once the necessary information from the neighbouring nodes being available, selecting the suitable numerical scheme at each node, optimizing the sequence, and organizing the calculation. This procedure permits to circumvent as much as possible limitations of the step size in marching direction, since the CFL numerical stability criteria effectively limit the relative size of Δx_1 and Δx_2.

In conclusion, the resulting algorithm is very flexible because the problem of solving the boundary-layer on the whole configuration reduces to a succession of solutions at one single normal at a time. For this reason, it is possible to continue the process around normals where the calculation has broken down. This feature gives its feasibility to be applied on a complex configurations.

Potential Flow Numerical Method

Potential flow calculations take far less then a tenth of the CPU time of Euler calculations, and they give good simulations for commercial aircraft applications.

A potential flow code (3)(4) WBNFLOW has been developed at FFA for simulation of transonic

flow around wing-body-nacelle configurations. The propeller is modelled as an actuator disc, and the effect of the slipstream is considered in an approximate way. The finite volume formulation is used here for the discretization. It is a multigrid-multiblock solver using both the incomplete lower upper decomposition and the strongly implicit procedure as the smoothing algorithm. It is fully implicit not only in all the three coordinate directions but also between the blocks.

The multigrid method with the implicit algorithm brings robustness, effectiveness and good convergence. The applications have shown excellent agreement with wind tunnel test data for a propeller driven airplane in subsonic climb and in transonic cruise [3][4]. This code is used here to give the outer boundary condition for the boundary-layer calculations.

The code is cost effective as a result of the multigrid method with implicit algorithm. The concept for the multigrid method as an acceleration technique is to eliminate efficiently each Fourier component of the error spectrum on the coarsest possible grid. This concept relies on the use of relaxation algorithms that are very efficient in damping those wavelength components of the error, in at least one of the coordinate directions, with the wavelength comparable to the mesh size. The incomplete lower upper decomposition and the strongly implicit procedure[5][6] is used as the smoothing algorithm. The potential equation in the computational domain after the finite volume discretization for the fixed grid level may be written as

$$L(\phi)\,\phi = f \qquad (1)$$

where ϕ is the potential, which is unknown at the nodes. L contains the algebraic coefficients arising from discretization, depending on the solution ϕ itself, and can be given by the previous iteration results in the iteration procedure. f is made up of algebraic coefficients associated with discretization and known values of ϕ, e.g. that are from the boundary conditions. First, equation (1) is linearized using the previous iteration results

$$\boldsymbol{L}\,\phi = f \qquad (2)$$

In the iteration procedure (2) can be rewritten as

$$A\,\phi^{n+1} = B\,\phi^{n} + f \qquad (3)$$

where $A = \boldsymbol{L} + B$ is close to $\boldsymbol{L}$, but computationally efficient to factorise. This means that the error matrix B should be chosen such that the iteration matrix A is easy to eliminate, which yields rapid convergence of the iteration procedure without too much memory storage, and is a good approximation of the system matrix $\boldsymbol{L}$. And (3) can be rewritten as

$$\phi^{n+1} = A^{-1} B\,\phi^{n} + A^{-1} f$$

or

$$\phi^{n+1} = \phi^{n} - A^{-1} R^{n}$$

where

$$R^{n} = -f + \boldsymbol{L}\,\phi^{n}$$

For the case of incomplete lower upper decomposition and the strongly implicit procedure, an incomplete lower upper decomposition of the system matrix is carried out

$$A \equiv \boldsymbol{L}\,\boldsymbol{U} = \boldsymbol{L} + B$$

The algorithm is implemented in two stages. First, decomposition of the system matrix into a lower and an upper triangular matrix in the way as briefly described above, a forward sweep is done and the matrix is now upper triangular. Then is followed a back-substitution to solve $\Delta\phi$.

The algorithm is coded as a plane-by-plane algorithm, and it is necessary to store the entire (sparse) upper triangular matrix $\boldsymbol{U}$. The important aspects of incomplete lower upper decomposition and the strongly implicit procedure algorithm are its full implicitness and the absence of a preferred sweep direction. It has been shown that use of incomplete lower upper decomposition and the strongly implicit procedure in the multigrid method leads to a fast and robust process and a well converged solution. The price for the full implicitness is the need to store the entire upper triangular matrix $\boldsymbol{U}$. The storage required is not restrictive. However, this algorithm is not so flexible to be easily applied to any kind of multiblock organization.

This is a two block code. In order to be implicit between the blocks, as the algorithm is coded as a plane-by-plane algorithm, it is mainly needed to store the entries of two planes for the upper triangular matrix at the first block for second block use. The alternative is, of course, to use an algorithm which is restricted to be implicit per block. Such an algorithm is in fact for the block boundaries a local two-dimensional SLOR algorithm for a three-dimensional gener-

alization. Therefore, it degrades the incomplete lower upper decomposition and the strongly implicit procedure algorithm completely.

Computational Results and Choice of Topology

According to Van Dyke[7], a second order boundary-layer theory differs from a first order not only with the difference in the boundary-layer equations but also in the interaction between the external equivalent inviscid flow and inner viscous flow taken into account by the matching condition at the edges. With the first order or the second order in this paper is meant only the boundary-layer calculation. As there is no considerable difference between the inviscid flow results and the equivalent inviscid flow results in the application cases within this paper, the first order boundary-layer calculations are used here.

In order to validate the method, calculation has been done on a wing in transonic flow. The results are compared with experimental data [8] and shown in Fig. 4. The configuration was an aspect ratio 20 rectangular wing with an RAE 2822 airfoil, and the flow is given through Mach number 0.676, angle of attack of 1.98 degree and $Re=5.7*10^6$. The transition was fixed at $x/c=0.11$ according to the experiments in [8]. The outer edge boundary condition for the boundary-layer calculation was obtained from a WBNFLOW calculation using a grid of C-H topology [9]. The calculation results on the symmetrical section are compared with the RAE 2822 airfoil 2D experimental data [8]. In [8] the test was performed at an angle of attack of 2.4 degree. However, in the calculation the angle of attack is taken as 1.98 degree because of the wind tunnel wall disturbances according to [10], the correction taken into account. Fig. 4 shows that the calculation results are in good agreement with the experimental data on the velocity profiles, the distribution of skin friction coefficient, displacement thickness and momentum thickness for this wing at a transonic Mach number at an angle of attack.

The application to a propeller driven airplane configuration, which consists of wing, body and nacelle, Fig. 5, at a subsonic Mach number 0.158, an angle of attack of 0 degree, and $Re=4.83*10^6$, is shown in Fig. 6 to Fig. 14. The outer edge boundary condition was obtained from the WBNFLOW calculation, using C-H topology and two blocks for the grid. In the boundary-layer calculation the flow was assumed to be turbulent. No considerable difference in results was observed when using the turbulence model of Baldwin-Lomax or of Cebeci-Smith.

Figs. 6 to 9 show characteristics of the boundary-layer for the upper surface of the inboard wing, such as external streamlines and skin friction lines, as well as isolines for boundary-layer thickness and skin friction. The flow is dominantly two-dimensional, except in the region close to the nacelle and the wing trailing edge. Fig. 10 shows the boundary-layer development and the velocity distribution at a wing section close to the inboard side of the nacelle.

The remaining figures apply to the upper surface of the outboard wing. Figs. 11 and 12 contain patterns of streamlines and skin friction lines. They show that only close to inboard and outboard ends respectively are any visible three-dimensional flow effects. The two last pictures show isolines for boundary-layer thickness and skin friction.

The full potential calculations take far less than a tenth of the CPU time of Euler calculations. The multigrid method together with the implicit smoothing algorithm brings robustness, effectiveness, and good convergence, therefore the code WBNFLOW is cost effective. In the case of a C-H topology grid, certainly one can get very good resolution for the wing and also for the region of the whole wing-body-nacelle configuration, which is rather close to the wing, see [3] and[4]. For this commercial airplane configuration the surface grid on the wing can be directly used for the boundary-layer calculations without any extra work, e.g. generating surface grid, interpolating data from inviscid calculations for the outer edge boundary condition in boundary-layer calculations. It is important that the surface grid is suitable for boundary-layer calculations. Then it is unnecessary to make interpolations, that have no physical meaning, for the boundary-layer outer edge conditions, as such an interpolation often damages the accuracy in the boundary-layer calculation, especially in regions where the geometry changes rapidly and steep gradients occur in the field.

The multigrid method together with the implicit smoothing algorithm makes the code WBNFLOW cost effective. On the other hand, the implicit smoothing algorithm used here is difficult in the coding respect to implement for any number and type of block arrangement. For the wing-body-nacelle configuration using two blocks of C-H topology, the sweep is made roughly in the spanwise direction. The code is

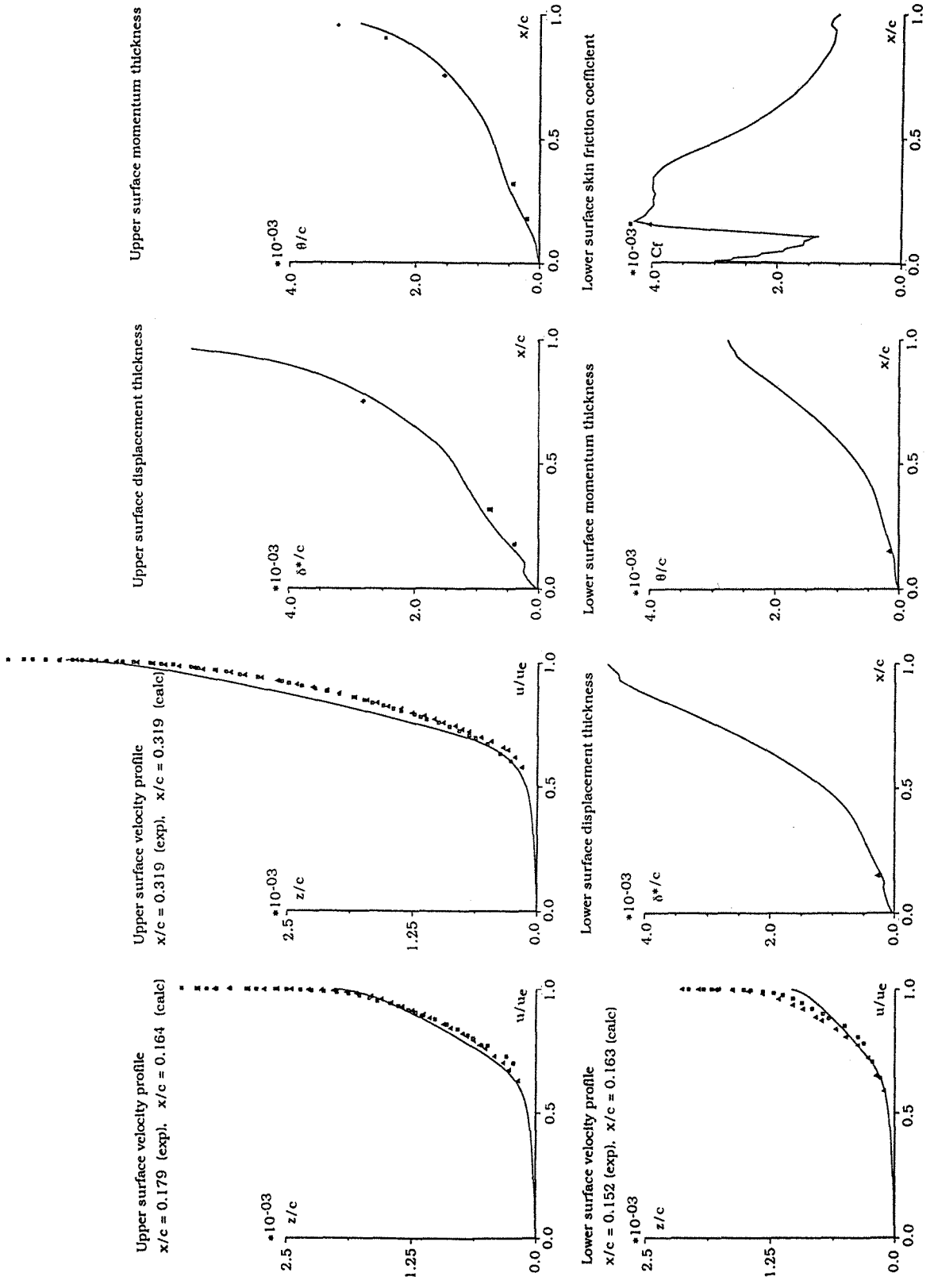

Fig. 4. Comparison between calculated results at the symmetry section on an aspect ratio 20 wing with RAE2822 airfoil and two-dimensional experimental data[8].
M_∞= 0.676, α = 1.98°, Re=5.7*10^6.

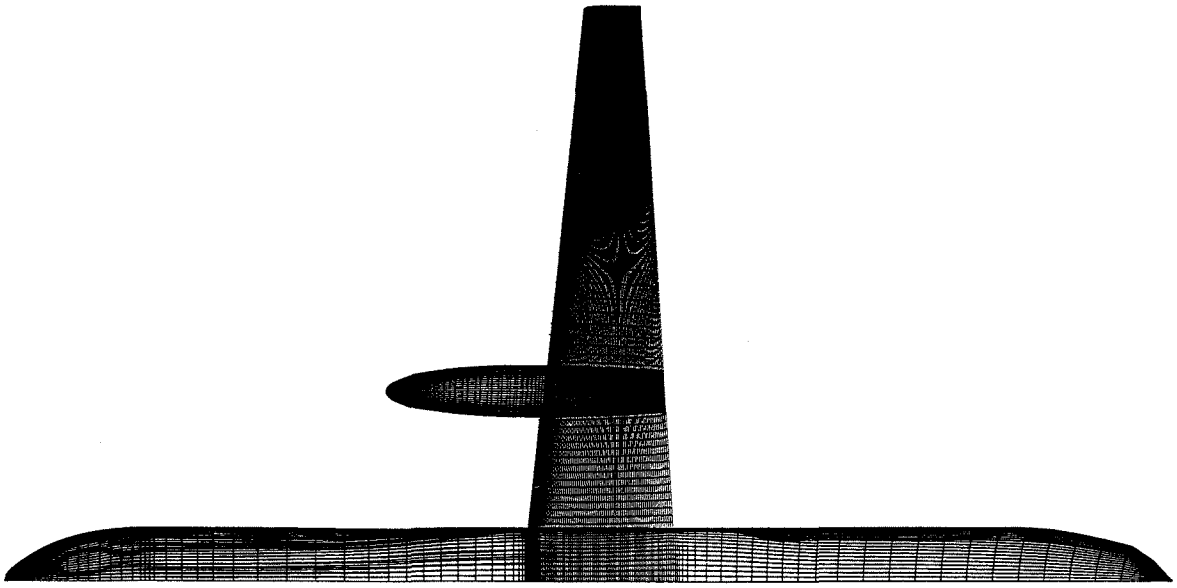

Fig. 5. Wing-body-nacelle configuration

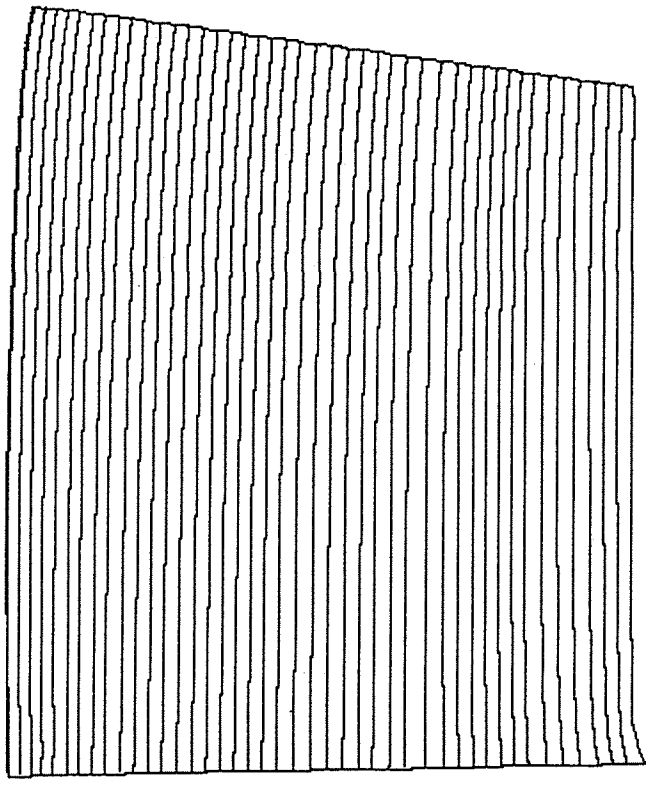

Fig. 6. Streamlines for inboard wing, upper surface

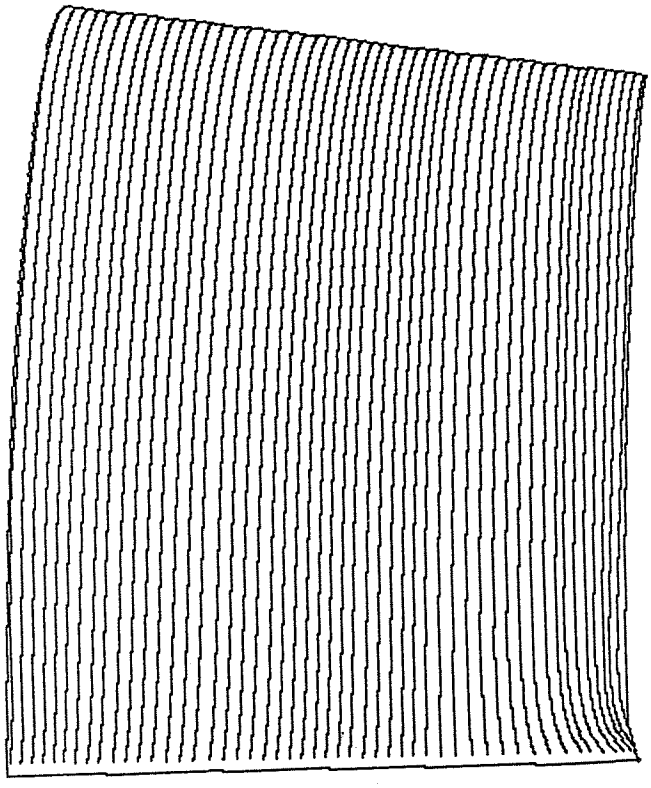

Fig. 7. Skin friction lines for inboard wing, upper surface

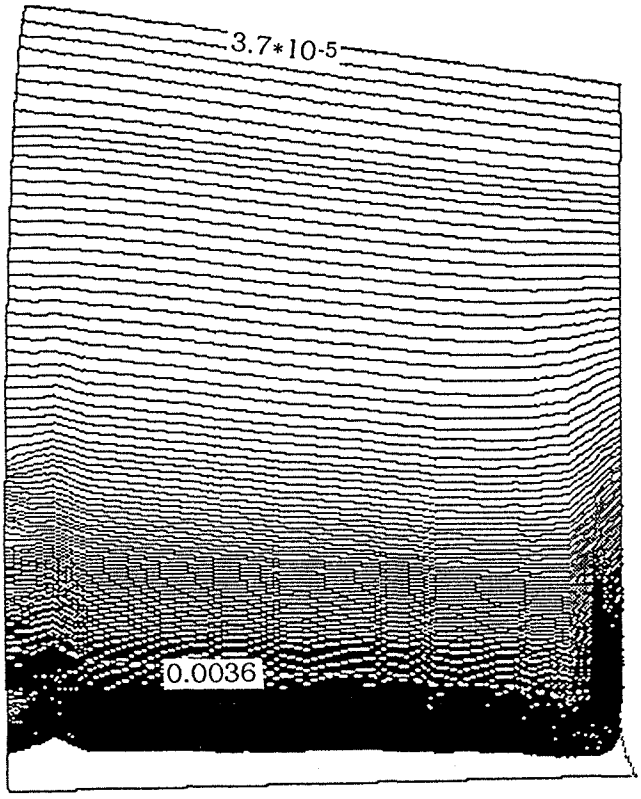

Fig. 8. Boundary layer thickness isolines inboard wing, upper surface

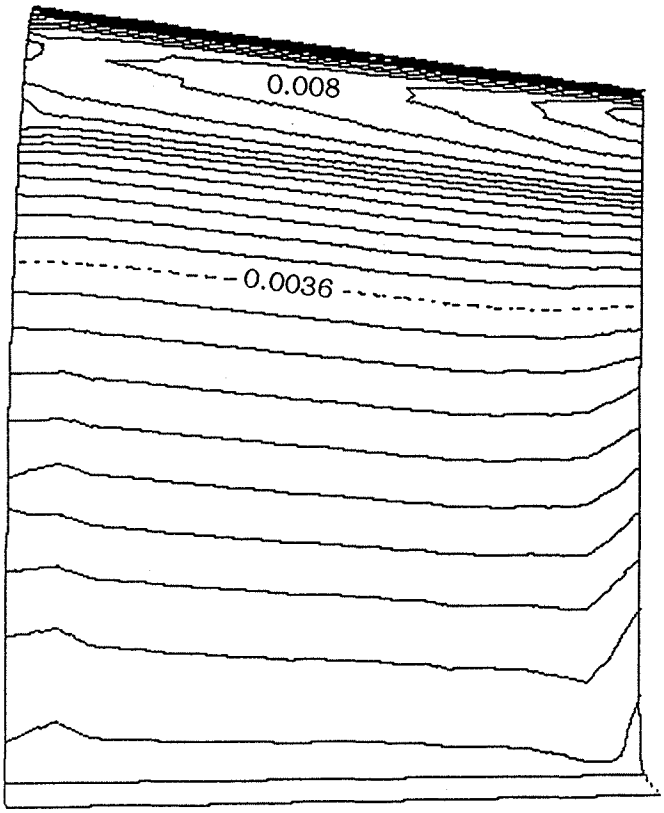

Fig. 9. Skin friction coefficient isolines for for inboard wing, upper surface

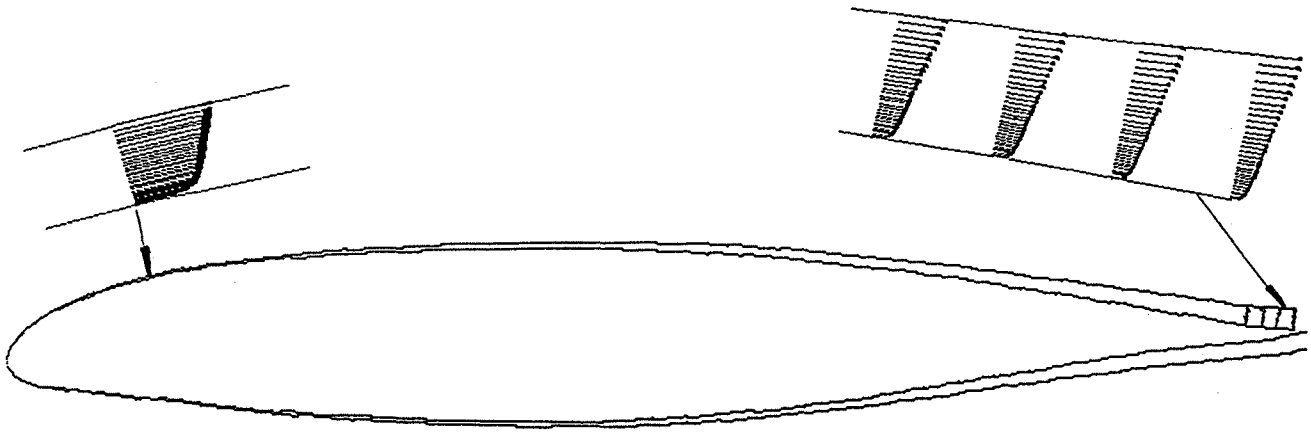

Fig. 10. Boundary layer at wing section, just inboard of nacell

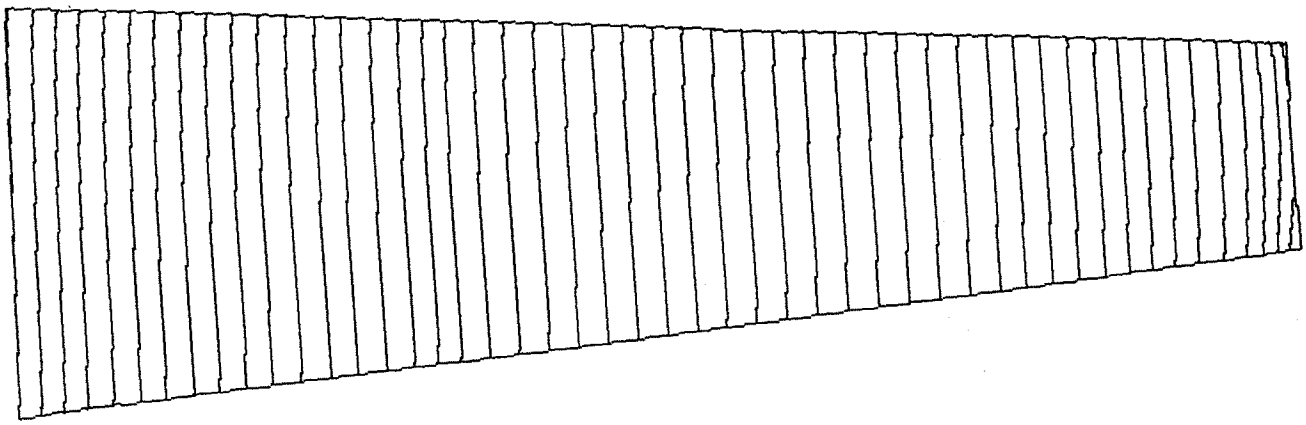

Fig. 11. Streamlines for outboard wing, upper surface

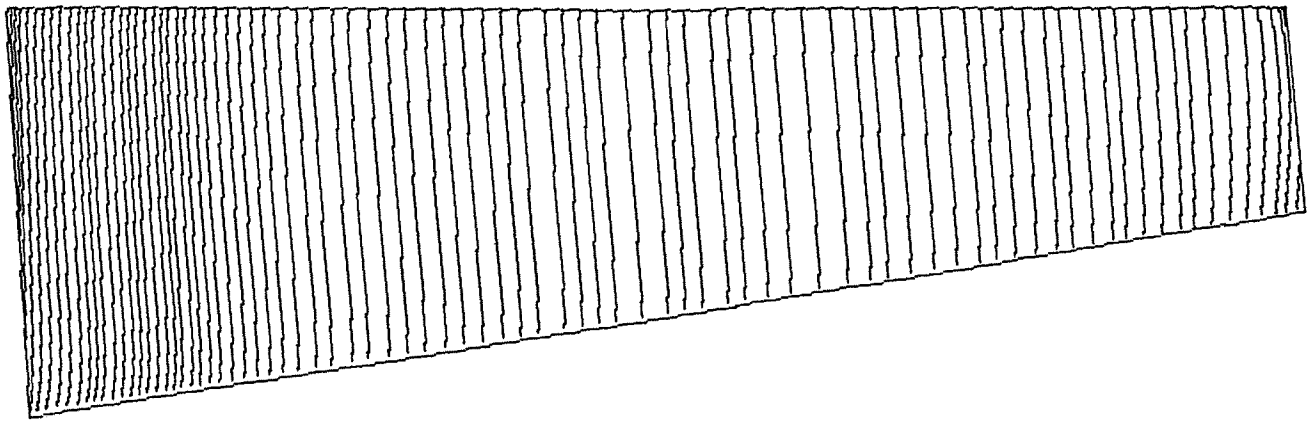

Fig. 12. Skin friction lines for outboard wing, upper surface

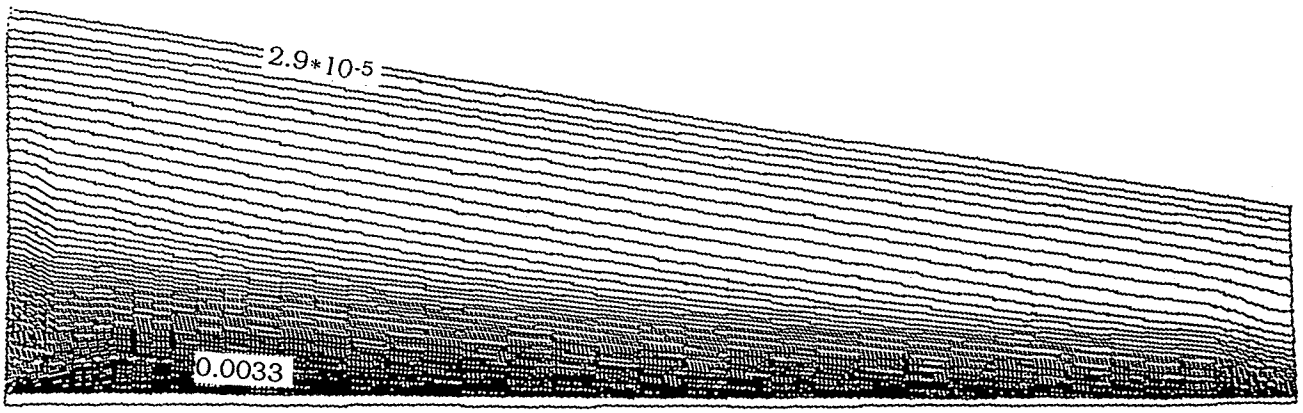

Fig. 13. Boundary layer thickness isolines for outboard wing, upper surface

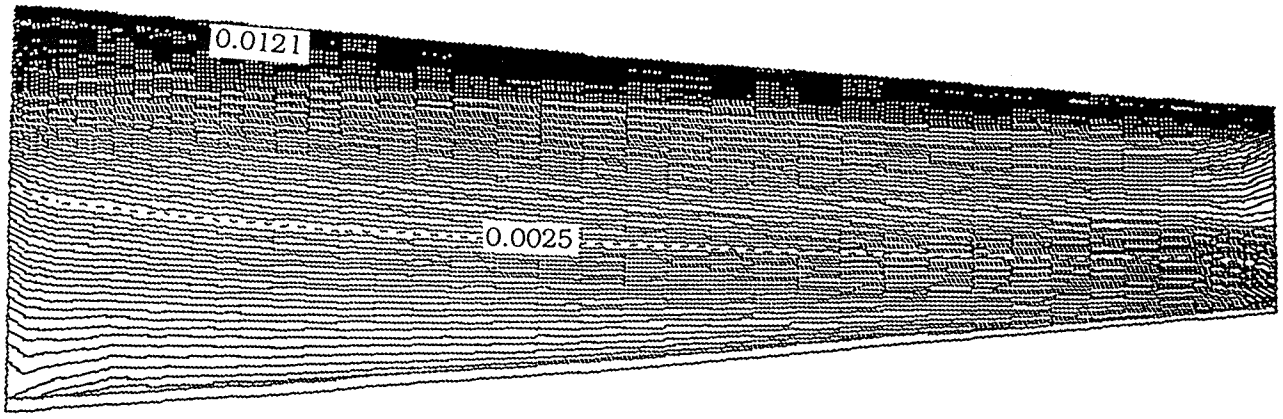

Fig. 14. Skin friction isolines for outboard wing, upper surface

organized plane-by-plane. For any block arrangement if one has to break the sweep in one plane into several pieces, it would be difficult in coding respect.

For the commercial airplane there is often a long fuselage part ahead of the wing, and a C-H topology grid would be too coarse at the nose of such a fuselage, and the surface grid on the fuselage nose would not be suitable for boundary-layer calculations. In this case the interpolated results might not be very suitable for boundary-layer calculation, and the outer edge matching condition has to be set lower. The results on the fuselage would not be as accurate as on the wing and it would influence the flow in the junction. Otherwise the situation would be misinterpreted as flow separation. The situation is similar for the nacelle. As the two blocks C-H topology is used in the inviscid calculation, the calculated flow results on fuselage and nacelle are not as accurate as on the wing.

Further, in the full potential calculations one has to specify the wake position. This makes it difficult to have a completely body-conforming grid on the rear portion of fuselage. The difficulty is the same for the nacelle grid. This influences the accuracy in the boundary-layer calculation.

The best choice of grid topology for viscous flow calculations is an efficient multiblock grid for the whole wing-body-nacelle configuration, in which the surface grid is suitable for boundary-layer calculations. Then it is unnecessary to make any interpolation for the boundary-layer outer edge condition, as such an interpolation often damages the accuracy in the boundary-layer calculation. In this respect an Euler code would be very flexible and straight forward, because it can utilize any multiblock grid topology, as a Runge Kutta time stepping scheme is used for the flow field solution.

Conclusions

The wing is aerodynamically the most important component of the airplane. The full potential code together with the boundary-layer code is very cost effictive for the prediction of the viscous flow at least about the wing of a wing-body-nacelle configuration. Valid results from such calculations were shown in this report. Therefore it is a useful tool for the airplane design.

However, for calculations of viscous flow about the whole configuration it appears necessary to to use a more complex multiblock grid.

References

1. Monnoyer F.,
   "Calculation of Three-dimensional Attached Viscous Flow on General Configurations Using Second-order Boundary Layer Theory", Z. Flugwiss. Weltraumforsch. 14, 1990, pp.95-108
2. Wang K. C.,
   "On the determination of the zones of influence and dependence for three-dimensional boundary-layer equations", J. Fluid Mech. (1971), vol.48, part 2, pp.379-404
3. Wang D., Hedman S. G.,
   "Multigrid-Multiblock Method for Potential Flow around Wing-Body-Nacelle Configurations with Slipstream", Advances in Computer Methods for Partial Differential Equations - VII, Robert Vichnevetsky et al. (Editors)
4. Wang D., Hedman S. G.,
   "Multigrid-Multiblock Method for Potential Flow around Wing-Body-Nacelle Configurations Including a Slipstream", Computational Fluid Dynamics '92, Volume 2, Ch. Hirsch et al. (Editors)
5. Stone H. L.,
   "Iterative Solution of Implicit Approximations of Multi-dimensional Partial Difference Equations", SIAM J Numer Anal, vol 5, no 3, Sep 1968
6. Meijerink J. A., v d Vorst H. A .,
   "An Iterative Solution Method for Linear Systems of which the Coefficient Matrix is a Symmetric M-Matrix", Mathematics of Computation, vol 31, no 137, Jan 1977
7. Van Dyke, M.,
   "Higher-order Boundary-layer Theory", Ann. Rev. Fluid Mech. 1(1969), 265-292
8. Cook P. H., McDonald M. A., Firmin M. C. P.,
   "Airfoil RAE 2822 - Pressure Distributions, and Boundary Layer and Wake Measurements", AGARD-AR-138, 1979
9. Hedman S. G. and Tysell L. G.,
   "WBNGRID, A Program for Generation of C-H and C-O Topology Grids around Wing/Body/Nacelle Configurations, User's Guide", FFA TN 1992-09
10. Firmin M. C. P. and Cook P. H.,
    "Disturbances from Ventilated Tunnel Walls in Aerofoil Testing", AGARD Conference Proceedings No.348