Multi-Body Flow Field Calculations with Overlapping-Mesh Method

Takeshi Sakata* and Antony Jameson†

*Department of Mechanical and Aerospace Engineering
Princeton University, Princeton, NJ*

Abstract

An efficient overlapping-mesh method for Euler flow solvers is developed and applied to two-body flow field problems. It features simple procedures to determine the interface cells on which the flow quantities are to be transferred from one mesh system to the other. This allows a great flexibility of mesh arrangements. The method is applied to two cases; one is the transonic flow field around vertically-laid, two NACA0012 airfoils. The present method demonstrates its capability to handle the cases in which strong interaction between the airfoils exists, predicting symmetric flow field in spite of the asymmetric mesh configuration. It is found that the overlapping region can be kept small unless the interaction effect of the flow becomes very large, and there is a large disparity in corresponding cell sizes in the interaction region. The other case is the subsonic flow around Williams airfoil with a flap, to which an exact potential solution is available for comparison. Though the geometry imposes severe restriction for the mesh arrangement, once a minor modification to the meshes is carried out, the method yields an excellent result.

1 Introduction

The application of numerical methods to fluid dynamics for prediction of aerodynamic properties has been becoming more and more popular and important. Especially in the area of aircraft design, large scale numerical applications are essential because of high cost and long lead time required for the traditional, empirical trial-and-error approach. This trend was set by the introduction of panel method in the sixties [1], and since that time, transonic flow calculations have been taking a leading role in computational fluid dynamics, which burst into wide use after Murman and Cole successfully solved the transonic small disturbance equation [2] in the mid-seventies. Together with rapid progress of computers in speed and capacity, newly developed methods have extended the range of feasible calculations.

We are at the onset of an era where the computational methods play an important role in science, offering a useful tool to provide new and better insights into very complex physics like turbulent flow [3]. For wide aspects of practical engineering applications, Euler and Navier-Stokes equations are available and commonly used in many flow solvers. However, those applications have imposed another challenge—generation of a suitable computational grid, which can influence the applicability of the codes and the credibility of their solutions.

It is now common to use boundary conforming grids to ensure accurate imposition of the boundary conditions. In general, there are three major approaches to generate such computational grids. One is the use of the numerical solution of a partial differential equation of the grid coordinates, and another is the construction of the grid by an algebraic expression of the coordinates using surface fitting and interpolations [4]. The third is the unstructured grid, which uses triangular or tetrahedral elements. The former two approaches have been quite widely accepted due to their straightforward formulations. But as the geometry becomes more complicated, much more computational effort is required to generate a suitable mesh system. To cope with the difficulties with complex geometries, several methods have been proposed.

The multiblock or zonal method, in which the physical domain is separated into a number of simpler subdomains, is probably the most accepted one. A typical, conservative implementation uses grid patching with common cells along the grid boundaries. This method ensures the continuity of flow quantities and possibly the conservation laws across the grid patches. However, the imposition of common cells requires additional computational effort and one might end up by introducing new kinds of grid singularities [5]. An alternative is the grid embedding method, which overlays the component subgrids. It is more flexible but requires a proper scheme to transfer the flow quantities among the grids.

An ultimate adaptability to any geometry can be achieved by using the unstructured mesh. It has achieved a remarkable success in calculating the Eu-

*Student Member AIAA.

†Professor, Member AIAA.

ler solution around a complete aircraft [6], which, however, requires vast memory space and complex bookkeeping of data.

This paper explores the capability of the grid embedding method, which we call the overlapping-mesh method. This area has been pursued by Atta and Vadyak [7], and Benek, Buning and Steger [8]. The idea is as follows: for a multi-body problem a grid is generated around each component independently in a simple and suitable way, and they are overlapped. Where and how to transfer the flow quantities is determined according to the overlapping geometries. Then, with a proper interaction among the component grids iterations over all the grids would establish a solution which satisfies the entire flow field of interest. It has flexibility both in generating the grids and in the choice of the flow solver to be used on each grid. It has the potential to ease the solution of complex problems like the store separation process [9]. However, this method has also some drawbacks. Definition of the overlapping region and the interpolation scheme could be inefficient and complex, and the conservation laws may not be strictly enforced in that region, leading to loss of accuracy and a stability. In its initial stage of development the code called Chimera developed by Benek, et al. does not show an excellent agreement with the experimental data [10]. Here we would like to present a new method based on the same idea but with a simpler treatment of the overlapping region and with a robust interpolation scheme.

2 Mesh generation

An inherent advantage of the overlapping-mesh method is free choice of a mesh type and a flow solver for each component. In order to obtain good results one must consider several aspects of mesh generation. At least one mesh has to be extended sufficiently to the far field. Each component mesh must have appropriate mesh resolution near the component body. And they should have an adequate overlapping region—the meaning of adequate will be explained in the following section.

As the simplest configuration for a two-body flow field, two meshes are generated. The larger mesh around a body that also covers the entire flow field is called *the outer mesh*, and a C-mesh is used. The smaller mesh around the other body is generated locally and submerged completely into the outer mesh, and is called *the inner mesh*, which is chosen to be an O-mesh. The inner mesh is totally overlapped on the outer mesh, but we restrict the term "overlap region" to the designated region where the interaction of the flow quantities between the meshes takes place. This overlap region should not fall on the outer body, because the interpolation scheme does

not work in the interior of the body where no flow exists. It is possible to avoid this additional constraint on mesh generation, but such a remedy would require either an elaborate interpolation scheme or an extensive modification of the flow solver. However, the constraint becomes severe when the method is applied to close-coupled bodies, where the physical space between the bodies is too small to generate a suitable mesh system.

3 Interpolation

3.1 Set up of overlap region

The interpolation scheme between the meshes is crucial for a successful calculation, and most careful attention should be paid to it. Two-way interpolation must be established; the outer-to-inner and the inner-to-outer interpolations.

First, the overlap region has to be defined. For the flow calculation on the inner mesh the flow data of the outer mesh is provided at the outermost cells so that it serves as the far field boundary condition of the inner calculation. Thus, the outer rows of the inner mesh cells are designated as the outer-to-inner interface cells, where the interpolation is carried out.

The inner-to-outer interpolation must pick up valid result of the inner calculation, which means it has to take place a little deep in the inner mesh. For the outer calculation the outer cells on and adjacent to the inner body are of no use, since the interpolation would not work properly and it is too complicated to represent such a local flow field by "foreign" cells. Essentially, the flow quantities in that area are calculated on the inner mesh, so the overlaid outer cells can be neglected. Thus, we have those "hole cells" around the inner body and that hole must be surrounded by the inner-to-outer interface cells. Although such designations are to be made to the outer cells, locating those cells is easier if we start the cell search from the inner mesh, since the inner mesh has an established structure ready to use. The actual procedure to define the inner-to-outer interface cells is as follows; First, the candidates of the interface cells are chosen by seeking the closest outer cell to an inner cell of the specified row or the ring, supposing the inner mesh is of O-type as it is the case. Those candidate interface cells form a near-ring. Then, we have to fill the voids of the ring, if any, with an appropriate rule, so that the interface cells become a complete ring. Storing their locations and setting special flags for both the interface cells and the hole cells completes the procedure.

Now we have several parameters to control the overlap region. One is the extent of the inner mesh that controls how far the outer edge of the inner mesh is from the inner body (I-BOUND). Another is the separation of the overlap region, which is de-

defined as the number of the cells laid between the outer-to-inner and the inner-to-outer interface cells. The third is the depth of the interface cells. We can deepen or thicken the interface cell area or band by adding the rows adjacent to them, inner-to-outer and outer-to-inner, independently.

Whatever the interpolation scheme is, if there is a large disparity among the corresponding cells' volume (area) at the interface, the accuracy of the interpolation degrades. To measure this disparity the inner-to-outer and the outer-to-inner interface cell area ratios are defined as

$$\frac{\text{area of the data giving cell}}{\text{area of the data receiving cell}}$$

Note the smaller the ratio is, the better the quality of the interpolation becomes.

3.2 Interpolation scheme

Several interpolation scheme have been proposed for the overlapping-mesh method. They are; first order Taylor approximation, bilinear (or trilinear for 3-D) interpolation, and higher order schemes. Mastin and McConnaughey [11] compared these schemes applied to the overlapping-mesh method for solving Laplace and Poisson equations, and found that if higher order derivatives are not important, a higher order interpolation scheme results in better accuracy, but if they are important, the higher order approximation suffers more. In our case a shock wave may be involved in the interpolation procedure, and a higher order scheme has a larger stencil which requires a wider overlap region and might cause undesirable oscillations. Thus, a simple and compact, 4-point bilinear scheme is employed for both interpolations, providing second order accuracy.

Using the bilinear scheme does not ensure the conservation. Berger [12] introduced a 2-D conservative interpolation method, which, however, is not readily applicable due to its complexity. Allowing an arbitrary overlay of the meshes results in disparity in the interface cell sizes, with a consequent deterioration of accuracy. Thus, the local lack of the conservation may be less important than accuracy loss at the interpolations.

Chesshire and Henshaw [13] analyzed the necessary accuracy of the interpolation scheme for the overlapping-mesh method, and showed for 1-D elliptic differential equations that if the separation of the overlap region is dependent on the mesh width, a second order accurate solution requires a third order accurate interpolation scheme. This result seems rather conservative and is not directly applicable to our calculations. The Euler equations are of hyperbolic type, and the accuracy of the solution would depend more on the accuracy of the interpolation in passing disturbance waves from one mesh to the

other. Therefore, we concluded that the bilinear scheme would be sufficient, and opted for its ease of implementation and robustness.

The 4-point bilinear interpolation formula is

$$u = u_1\left(\frac{1}{2} - X\right)\left(\frac{1}{2} - Y\right) + u_2\left(\frac{1}{2} + X\right)\left(\frac{1}{2} - Y\right) \\ + u_3\left(\frac{1}{2} + X\right)\left(\frac{1}{2} + Y\right) + u_4\left(\frac{1}{2} - X\right)\left(\frac{1}{2} + Y\right),$$

where X and Y are mapped coordinates of an interior point (x, y) on physical plane onto a unit square with the local bilinear mapping of the same form (Fig. 1).

On an interface cell the point search procedure is called to find 4 points on the other mesh surrounding that point. This is done by seeking the closest point and determining in which quadrant that point lies. A check of closure is necessary, and, if needed, an additional local point search is carried out automatically. This procedure leads to an efficient and robust interpolation scheme applicable to any mesh arrangement, which is also easily extendible to the three dimensional problems. Note that X, Y and the indices of corresponding points are determined only by the geometry, so they need to be calculated once for all. An example of the inner-to-outer and the outer-to-inner interpolations are shown in Fig. 2.

4 Flow solver

The flow solver employed to this work is an explicit time stepping, finite volume Euler solver. The Euler equation in conservation form is written as

$$\frac{\partial w}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0 \quad (1)$$

where

$$w = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix}, F = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uH \end{pmatrix}, G = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vH \end{pmatrix}.$$

E, H are total energy and total enthalpy, respectively. This is integrated on a finite volume cell to become

$$\frac{\partial}{\partial t} \int_{\Omega} w ds + \int_{\partial\Omega} (F dy - G dx) = 0, \quad (2)$$

which leads to a semi-discrete form

$$\frac{d}{dt} w_{i,j} + R_{i,j} = 0, \quad (3)$$

where

$$R_{i,j} = \frac{1}{S_{i,j}} (Q_{i,j} - D_{i,j}).$$

R, S, Q and D are residual, cell area, Euler flux balance and adaptive dissipation flux balance, respectively. Then, equation(3) is integrated in time by

multi-stage Runge Kutta scheme, until the steady state is achieved. Residual smoothing and enthalpy damping can be employed to allow the use of larger CFL numbers.

Although the same type of flow solver is used on both meshes in this paper, the parameters of the solver can be adjusted independently. The interpolation from one mesh to the other is called after the flow data is updated on the mesh by one time step.

5 Calculated cases and results

5.1 Two NACA0012 airfoils

For transonic flow application two NACA0012 airfoils are laid parallel with a vertical distance. Both airfoils have zero angle of attack and the resultant flow field must be symmetric about the center line between the airfoils, while the mesh arrangement is not symmetric. The 256×32 outer C-mesh is generated around the lower airfoil (airfoil #1) by the conformal mapping method, and the 160×32 inner O-mesh around the upper airfoil (airfoil #2) is also generated by conformal mapping (Fig. 3). M_∞ is set to 0.8 so that a strong interaction effect is expected in the region between the airfoils.

Figs.4-6 show C_p distributions of the upper and the lower airfoils with three different vertical distances (VD) of $3c$, $2c$ and $1c$, respectively, where c means a chord length. The extent of the inner mesh is adjusted according to VD. For $VD=3c$ and $2c$ cases the minimal overlap configuration that consists of one row of the interface cells for each interpolation separated by only one row width gives fairly symmetric, reasonable results. But for $VD=2c$ the shock locations in the interaction region and C_p profiles begin to deviate. For $VD=3c$ case the outer-to-inner interface cell area ratios are about 0.3 in the interaction region, and 12.6 in the non-interaction region. The inner-to-outer ratios are 1.7 in the interaction region and 0.07 in the non-interaction region. Those values for $VD=2c$ case are 0.1, 6.9, 4.3 and 0.1, respectively. Note that the inner-to-outer cell area ratio in the interaction region deteriorates from 1.7 to 4.3. The deviation from the symmetric solutions becomes clearer in Fig. 6(a), where $VD=1c$ with the minimal overlap configuration. That cell area ratio is 5.0. C_p profiles on the interaction sides are quite different and those on the non-interaction sides are also suffering. Fig. 6(b) shows the same $VD=1c$ case, but with deepened inner-to-outer interface region so that most of the interaction region becomes the interface. By increasing the data transfer a good result is restored.

Fig. 7 shows the Mach contours of the case of $VD=2c$. A strong shock due to the interaction effect is apparent in that region. The shock is only little smeared in the middle. Also, the good matching

of the contour lines at the mesh interface indicates satisfactory accuracy of the interpolation scheme.

Concerning the mesh arrangement, a question may arise as to what happens if the inner mesh extends so much that it completely includes the outer body. Fig. 8 shows one of those mesh arrangements. The outer-to-inner interpolation is carried out along the outer edge of the inner mesh, which encloses the lower airfoil. The inner-to-outer interpolation is done in the middle of the inner mesh. The result shown in Fig. 9 does not seem promising. Although this is also a $VD=3c$ case, the upper airfoil shows much less interaction effect, and the lower airfoil indicates the reversed effect. The breakdown of the present method could be attributed to the improper boundary conditions set by such an inappropriate mesh arrangement. The inner calculation is provided with its outer edge flow data from the outer calculation. Let us consider the circulation in the flow. Since the entire flow field is symmetric, if one takes a path that encloses both airfoils, the circulation along it must vanish. Now in this case, that zero circulation information is provided to the inner calculation, which does not "see" the existence of the lower airfoil. Thus, the inner calculation goes toward the flow field with only one isolated airfoil. The outer calculation degrades accordingly. This case shows that some important information can be lost at the interpolation with an inappropriate mesh arrangement, even though the interpolation scheme is accurate enough.

5.2 Williams airfoil with a flap

In order to further verify the present method another case has been studied. Williams [14] generated an airfoil with a flap from two circles by a sequence of Karman-Trefftz transform and provided an exact potential flow solution. Since that solution is for incompressible flow, M_∞ is set to 0.25 for the present calculations, and the Prandtl-Glauert transform is applied to the exact solution for comparison. This is a very severe test because of the geometry. Because of the narrow space between the main airfoil and the flap, it is difficult to generate a sufficiently large inner mesh without impinging on the main airfoil.

Preliminary calculations with the conformally generated C- and O-mesh combination revealed the other difficulty. Due to the shape of the flap, the conformally generated O-mesh around it has too small cells near the leading and the trailing edges. Combined with the fairly large outer cells generated near the trailing edge of the main airfoil, the disparity in the cell sizes is aggravated. To improve the situation, modifications on both meshes were undertaken. For the inner mesh an expansion function is added to the generation process to control the cell sizes in the region of interest. For the outer C-mesh

a hyperbolic type mesh generator is used to obtain a better cell size distribution around the trailing edge while the cell number is retained.

Fig. 10 shows a close-up view of the mesh system, 256×32 outer and 160×7 inner meshes. The result is compared with the exact solution in Fig. 11. C_p profiles of both the main airfoil and the flap agree with the exact solution very well. Table 1 compares the aerodynamic coefficients, which show good agreement, too. One drawback is relatively slow convergence due to the existence of high aspect ratio cells.

Table 1. Comparison of aerodynamic coefficients

Case	Main airfoil		Flap	
	C_l	C_d	C_l	C_d
160×7	1.7937	-.0950	.9945	-.2630
Exact	1.7470	-.0927	.9592	-.2556

Note that C_d of the flap is based on the flap chord length. If the chord of the main airfoil is used, then

$$C_{d_{Main}} + C_{d_{Flap}} = 0$$

for the exact solution.

6 Concluding remarks

An efficient and easily implementable overlapping-mesh method is presented. Meshes are generated separately around the bodies with few restrictions, an outer mesh extended to the far field and an inner mesh submerged into the outer mesh, and the interface cells are determined for the inner-to-outer and the outer-to-inner interpolations with the simple procedures. Due to the robust and compact bilinear interpolation scheme, a minimal overlap configuration can keep the computational effort small, provided that the interface cell area does not become too large.

Transonic cases for two NACA0012 airfoils laid parallel show the capability of the method to deal with strong shock interactions. Though the computational domain is not symmetric because of the meshes, it successfully reproduces the symmetric flow field, with deepening of the interface cell rows if necessary, as indicated by the cell area ratios. Also, it is shown that an inappropriate mesh arrangement could degrade the solution badly.

The subsonic case for Williams airfoil with a flap presents severe geometrical difficulties, but with minimal modifications to the meshes the present method yields excellent agreement with the exact solution, which demonstrates its flexibility and potential for further applications.

References

- [1] Jameson, A., *Transonic Flow Calculations*, MAE Report No. 1651, Princeton University, 1983.
- [2] Murman, E. M. and Cole, J. D., "Calculation of Plane Steady Transonic Flows", *AIAA J.*, Vol. 12, 1974, pp.626-633.
- [3] Jameson, A., "Successes and Challenges in Computational Aerodynamics", *AIAA Paper 87-1184-CP*, 1987.
- [4] Thompson, J. F., Warsi, Z. U. A. and Mastin, C. W., *Numerical Grid Generation, Foundations and Applications*, North-Holland, 1985.
- [5] Lee, K. D., "3-D Transonic Flow Computations using Grid Systems with Block Structure", *AIAA Paper 81-0998*, 1981.
- [6] Jameson, A., Baker, T. J. and Weatherill, N. P., "Calculation of Inviscid Transonic Flows over a Complete Aircraft", *AIAA Paper 86-0103*, 1986.
- [7] Atta, D. H. and Vadyak, T., "A Grid Interfacing Zonal Algorithm for Three Dimensional Transonic Flows about Aircraft Configurations", *AIAA Paper 82-1017*, 1982.
- [8] Benek, J. A., Buning, P. G. and Steger, J. L., "A 3-D Chimera Grid Embedding Technique", *AIAA Paper 85-1523*, 1985.
- [9] Dougherty, F. C. and Kuan, J-H, "Transonic Store Separation Using a Three-Dimensional Chimera Grid Scheme", *AIAA Paper 89-0637*, 1989.
- [10] Benek, J. A., Steger, J. L. and Dougherty, F. C., "A Flexible Grid Embedding Technique with Application to the Euler Equations", *AIAA Paper 83-1944*, 1983.
- [11] Mastin, C. W. and McConnaughey, H. V., "Computational Problems on Composite Grids", *AIAA Paper 84-1611*, 1984.
- [12] Berger, M. J., "On Conservation at Grid Interfaces", *SIAM J. Numerical Analysis*, Vol. 24, 1987, pp.967-984.
- [13] Chesshire, G. and Henshaw, B., *Composite Overlapping Meshes for the Solution of Partial Differential Equations*, Attached document to CMPGRD computer code, September 1988.
- [14] Williams, B. R., "An Exact Test Case for the Plane Potential Flow about Two Adjacent Lifting Aerofoils", *Aeronautical Research Council, R & M No. 3717*, 1973, England.

Fig. 1 Local bilinear transform

Fig. 2 Examples of the mesh arrangement and the interpolations

Fig. 3 Mesh system for two NACA0012 airfoils
placed parallel
 $VD = 3c$, $I-BOUND = 0.85$
inner mesh 160×32 , outer mesh 256×32

Note: Because of the index shift due to the cell-centered scheme
some of the interface cells are blanked out.

Fig. 4 C_p distributions of two NACA0012 airfoils
placed parallel
 $M_\infty = 0.8$, $\alpha = 0^\circ / 0^\circ$, $VD = 3c$

Fig. 5 C_p distributions of two NACA0012 airfoils
placed parallel
 $M_\infty = 0.8$, $\alpha = 0^\circ / 0^\circ$, $VD = 2c$

(a) with minimal overlap

(b) with deepened inner-to-outer interface cells

Fig. 6 C_p distributions of two NACA0012 airfoils
placed parallel
 $M_\infty = 0.8$, $\alpha = 0^\circ / 0^\circ$, $VD = 1c$

Fig. 7 Mach number contours of two NACA0012 airfoils placed parallel
 $M_\infty = 0.8$, $\alpha = 0^\circ/0^\circ$, $VD = 2c$

Fig. 8 Mesh arrangement for the outer body inclusion case
 $VD = 3c$, $I\text{-BOUND} = 0.94$

Note: Because of the index shift due to the cell-centered scheme some of the interface cells are blanked out.

Fig. 9 C_p distributions for the outer body inclusion case
 $M_\infty = 0.8$, $\alpha = 0^\circ/0^\circ$, $VD = 3c$

Fig. 10 Mesh arrangement for Williams airfoil with the flap
inner mesh 160×7 , outer mesh 256×32

Note: Because of the index shift due to the cell-centered scheme
some of the interface cells are blanked out.

Fig. 11 C_p distributions of Williams airfoil
with the flap
 $M_\infty = 0.25$