Double Mesh Method for Efficient Finite Difference Calculations

by Xu Qi*, Member Kazuhiro Mori*, Member Myung-soo Shin*, Member

Summary

The double mesh method (DMM), which is a numerical technique newly developed in the numerical simulation of free surface problems by boundary element method [11, is applied to the finite difference solutions of Navier-Stokes equation. DMM is a numerical method in which two mesh systems are used depending on the characteristics of the equations or terms to be calculated. The viscous flow around a circular cylinder and the free surface flow around the Wigley model are calculated by DMM. It was proved that the method can greatly improve the efficiency of the finite difference method.

1. Introduction

The finite difference method is widely used in the simulations of various physical phenomena. Many numerical techniques have been developed to improve the method, but the method still faces a serious problem that it requires relatively very long CPU time and a huge memory storage for accurate simulations. The improvement of the efficiency of the method has been the focus of researches in recent years.

Various approaches have been investigated about the problem to carry out the finite difference computations more efficiently. For example, the method of IAF, a kind of implicit scheme, the multi-grid method, the method of local time step, etc. Some comparative calculations by these methods have been carried out [2]. It seems that IAF is quite promising to speed up the calculation but its formulation is a little complicated. Moreover the method brings forth some numerical damping. As to the methods of multi-grid and local time step it seems that we can not expect much of them [2].

The mesh size, being a crucial element for efficient computations, is so required that the numerical truncation errors are small enough to have little effect on the physical performance of the object problem. For example, in the finite difference calculations of viscous flow described by the N-S equation, the minimum mesh size is usually so chosen that the numerical dissipation, which comes from the discretization of the convection terms, is much less than the physical dissipation. As discussed in 2., the mesh size must be extremely small for high Reynolds number flows to meet this demand. The computations in such a mesh system require long CPU time and large memory storage.

It is very common that the mesh system is chosen to meet the severest requirement. However, such fine meshes are not always necessary for all the equations and the terms. For example, the truncation errors of the Poisson equation for the pressure or the nonconvective terms in N-S equation do not have much influence on the results as the convection terms do.

A proper choice of the mesh size may make the computations more efficient. One possibility is to employ two different mesh systems depending on the characteristics of the equations or the terms. We call such a method the “double mesh method”, written in short as DMM hereafter. It was first proposed for numerical simulations of 3-D nonlinear free-surface flow problems by boundary element method [11]. In [1], in order to reduce the numerical viscosity as much as possible, a very fine mesh system which contains about 60 grids in one wave length is used in the finite difference calculation concerned with the free surface equations, while the governing Laplace equation is solved on a relatively coarse mesh system which contains about 10 grids in one wave length by boundary element method. The computed results by DMM were of enough accuracy and both the computing time and the size of the memory storage were remarkably reduced.

In the present paper, DMM is introduced into a finite difference solver of the Navier-Stokes equation to improve the calculation efficiency. As mentioned already, in the finite difference method the demands to the mesh size are not the same for all the equations and the terms. So it is expected that improvement, similar to that we have achieved in the simulation of free-surface problem by the boundary element method, may

* Hiroshima University
be made by introducing DMM into the conventional finite difference method.

2. Discretization Errors

Many physical phenomena can be described by a set of differential equations containing convection terms. When the differential equations are solved by the finite difference method, numerical dissipation is unavoidable due to the discretization of the differential equations. The numerical dissipation is sometimes crucial. To observe how the numerical dissipation works we introduce a simple model equation (Burgers Equation) as shown below and try to analyze it.

\[ u_t + a u_x = \nu u_{xx} \quad (1) \]

where \( u \) is an arbitrary physical variable; \( t \) is the time, \( x \) is a coordinate and \( a \) and \( \nu \) are constants. In (1) \( a u_x \) is the convection term and \( \nu u_{xx} \) is the dissipation term.

The equation can be discretized by various difference schemes; for instance, the 1st order explicit upstream scheme, the implicit scheme, the Lax-Friedrichs scheme, etc. All the schemes have truncation errors. They are tabulated in Table 1, where \( \Delta t \) and \( \Delta x \) are the increment of \( t \) and the mesh size respectively.

We take the 1st order explicit upstream scheme as an example to observe the effect of the numerical dissipation.

The term \( \frac{a}{2} u_{xx} \Delta x \) is the numerical dissipation term.

So the numerical viscosity coefficient is \( \nu_{num} = \frac{a}{2} \Delta x \).

To get reliable calculation results the numerical viscosity should be far less than the physical viscosity. Hence the following inequality should be satisfied when we select a mesh system for the finite difference calculation.

\[ \nu_{num} = \frac{a \Delta x}{2} < \nu \quad (2) \]

Here \( \frac{\nu_{num}}{\nu} \) is defined as mesh Reynolds number \( R(\Delta x) \) which can be expressed in the following form:

\[ R(\Delta x) = Re \frac{\Delta x}{2L} \quad (3) \]

Table 1 Finite Difference Schemes and Truncation Errors

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Truncation Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st order upstream</td>
<td>[ u^{n+1} = u^n + \frac{a}{2} \left( u_{n+1} - u_{n-1} \right) ]</td>
</tr>
<tr>
<td>Implicit</td>
<td>[ u^{n+1} - u^n = -\frac{1}{2} \Delta t \left( \frac{\Delta x}{\Delta t} \right)^2 ]</td>
</tr>
<tr>
<td>Lax-Friedrichs</td>
<td>[ u^{n+1} - \frac{1}{2} \left( u^n + u^n \right) ]</td>
</tr>
</tbody>
</table>

where \( Re = \frac{aL}{\nu} \) is the Reynolds number, \( L \) is the length scale.

It can be seen clearly that the mesh Reynolds number should be far less than one.

From the above analysis it can be seen that a very fine mesh system is required to solve the problems with a relatively high Reynolds number.

On the other hand, the truncation errors which come from the dissipation terms, gradient of pressure and the Poisson equation are of little influence on the accuracy of the computations. According to the experience of the authors, the use of the fourth order central difference scheme in the Poisson equation has not improved the accuracy much. The situation of the dissipation terms and gradient of the pressure are more or less the same.

Thus it may be concluded that the fine mesh system is necessary only for the convection terms whereas the other equations and terms can be discretized on a relatively coarse mesh system.

3. Computations by DMM

In this section, it is demonstrated how the DMM can be introduced into the finite difference solution of the Navier-Stokes equation based on the MAC method\(^6\). The 2-D case of the problem is considered, the governing equations for which are given as follows.

\[ \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \nabla^2 u \quad (4) \]

\[ \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \nu \nabla^2 v \quad (5) \]

Poisson Equation:

\[ \nabla^2 p = R \quad (6) \]

where

\[ R = -\left( \frac{\partial u}{\partial x} \right)^2 \left( \frac{\partial v}{\partial y} \right)^2 - 2 \left( \frac{\partial u}{\partial x} \frac{\partial v}{\partial y} \right) \frac{\partial D}{\partial t} \]

\[ -u \frac{\partial D}{\partial x} - v \frac{\partial D}{\partial y} + \frac{1}{Re} \nabla^2 D \]

\[ D = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \quad (8) \]

The underlined terms in the N-S equation are computed on a fine mesh system and the remains in the N-S equation and the Poisson equation are computed on a relatively coarse mesh system. The procedure of the computation is shown in Fig.1 together with the procedure of the conventional single mesh method, written in short as SMM hereafter, for comparison.

According to the continuity equation, the term \( D = (\partial u/\partial x) + (\partial v/\partial y) \) should be equal to zero. But actual computations are always accompanied by some errors. In order to avoid the accumulation of the errors we keep this term in the solution of the Poisson equation as a modification to the flow field.

For simplicity, initially a coarse mesh system is generated and is then refined by the interpolation of the grids to obtain the fine mesh system. Thus the grids of the coarse mesh system form a subset of the grids of the
fine mesh system. Physical values of the fine mesh system can be obtained by the interpolation of the values of the coarse mesh system.

4. Numerical Example

4.1 Viscous Flow Past A Circular Cylinder

The calculations around a circular cylinder are carried out by both SMM and DMM.

Fig. 2 shows the two mesh systems used in the calculations; MESH-1 (161 × 121), the fine mesh and MESH-2 (81 × 61), the coarse one. The 2nd order up-stream difference scheme is used for the convection terms in (1) and (2). $P P$ and $P^2 V$ are calculated by the 2nd order central difference scheme. The Poisson equation is solved by the SOR method. The maximum number of iterations for the Poisson equation is 10.

Three calculations are carried out to verify the efficiency of DMM which are:

CASE-1: SMM on MESH-1
CASE-2: SMM on MESH-2
CASE-3: DMM on MESH-1 & MESH-2

The Reynolds number is 2000 for all the calculations.

The calculations are carried out by EWS Sun-4 computer.

First we compare the results of CASE-1 and CASE-3. Fig. 3 shows the velocity vectors and Fig. 4 the pressure contours averaged from $T = 24$ to $T = 27$, where $T$ is the nondimensional time. Although a small phase shift in time can be observed in the flow patterns between CASE-1 and CASE-3, the two results are almost the same. No significant difference can be seen in the pressure distributions also.

Fig. 5 and Fig. 6 show the time histories of the drag and the lift coefficients of CASE-1 and CASE-3. Although both are relatively large compared to the experimental value, the average values of drags in the
Fig. 3 Velocity Vectors Around the Circular Cylinder, A : CASE-1, B : CASE-3

Fig. 4 Average Pressure Contour Around the Circular Cylinder, A : CASE-1, B : CASE-3

Fig. 5 Time History of Drag Coefficient, A : CASE-1, B : CASE-3
Double Mesh Method for Efficient Finite Difference Calculations

A slight phase shift can be observed in the lift coefficients.

Through the comparisons between CASE-1 and CASE-3 it can be concluded that the computation by DMM, though a relatively coarse mesh is used for the dissipation terms, the gradient of pressure and the Poisson equation, can give almost the same results as those by SMM.

Contrary to the negligible differences in the computed results, the CPU time and the memory size, which are tabulated in Table 2, are remarkably reduced by DMM. The CPU time needed for DMM is only about one third of that in CASE-1 and the memory size is reduced to 50% of that in CASE-1.

The calculation by SMM on MESH-2, the coarser mesh system, is also carried out as CASE-2. The computed drag and lift coefficients are shown in Fig. 7 and the pressure contour averaged from $T = 24$ to $27$ in Fig. 8. It is found that the amplitude of drag is less than those of CASE-1 and CASE-2 shown in Fig. 5 and Fig.

Table 2 Comparisons of CASE-1, CASE-2 and CASE-3

<table>
<thead>
<tr>
<th></th>
<th>CASE-1</th>
<th>CASE-2</th>
<th>CASE-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numerical Method</td>
<td>SMM</td>
<td>SMM</td>
<td>DMM</td>
</tr>
<tr>
<td>Mesh System</td>
<td>MESH-1</td>
<td>MESH-2</td>
<td>MESH-1 MESH-2</td>
</tr>
<tr>
<td>CPU time</td>
<td>16.10 s/step</td>
<td>4.11 s/step</td>
<td>5.28 s/step</td>
</tr>
<tr>
<td>Memory size</td>
<td>$\approx 3.0$ MB</td>
<td>$\approx 0.75$ MB</td>
<td>$\approx 1.5$ MB</td>
</tr>
</tbody>
</table>

two cases are almost the same. A slight phase shift can be observed in the lift coefficients.

Through the comparisons between CASE-1 and CASE-3 it can be concluded that the computation by DMM, though a relatively coarse mesh is used for the dissipation terms, the gradient of pressure and the Poisson equation, can give almost the same results as those by SMM.

Contrary to the negligible differences in the computed results, the CPU time and the memory size, which are tabulated in Table 2, are remarkably reduced by DMM. The CPU time needed for DMM is only about one third of that in CASE-1 and the memory size is reduced to 50% of that in CASE-1.

The calculation by SMM on MESH-2, the coarser mesh system, is also carried out as CASE-2. The computed drag and lift coefficients are shown in Fig. 7 and the pressure contour averaged from $T = 24$ to $27$ in Fig. 8. It is found that the amplitude of drag is less than those of CASE-1 and CASE-2 shown in Fig. 5 and Fig.

6. It can also be seen that the pressure rise in the upstream is moderated compared with those of CASE-1 and CASE-3. These are all due to the undesirable numerical damping caused by the coarse mesh system. The Strouhal number, which can be estimated from the period of the lift coefficient, is a little larger than those of CASE-1 and CASE-3. It is a general trend for the Strouhal number to increase as the mesh size increases.

Through these comparative calculations, it can be concluded that DMM, where a coarse mesh system is used in the calculation of Poisson equation, the dissipation terms and the gradient of the pressure, does not introduce any additional numerical damping. On the other hand, it improves the calculation efficiency considerably.
4. 2 Free-Surface Flow Around Wigley Model

As the second example, DMM is applied to the three-dimensional flows with free-surface. As pointed out in Kwag, et al. [4], the grid size is primarily important in the free-surface flow problems to obtain well-developed wave elevations.

The computing scheme developed in [4] is used here which is based on the MAC method. The free surface elevation is calculated by moving marker particles as

\[
\begin{align*}
x_n^{i+1} &= x_n^i + u_n^i \Delta t \\
y_n^{i+1} &= y_n^i + v_n^i \Delta t \\
z_n^{i+1} &= z_n^i + w_n^i \Delta t
\end{align*}
\]

where \(x_n^i, y_n^i\), and \(z_n^i\) are the position of the \(i\)-th particle at the time step \(n\), \(u_n^i, v_n^i\), and \(w_n^i\) are the velocity components and \(\Delta t\) is the time step.

(9) is the Lagrangian expression of the kinematic condition on the free-surface. The condition can also be expressed in the Euler form as follows;

\[
\frac{\partial \zeta}{\partial t} = -u \frac{\partial \zeta}{\partial x} - v \frac{\partial \zeta}{\partial y} + w
\]

where \(\zeta\) and \(t\) are the free-surface elevation and the time respectively. Numerically (9) is equivalent to (10) if the 1st order upstream difference scheme is used in (10). (See Appendix for detailed description.) Therefore (9) should be carried out on the fine mesh when DMM is applied, for the first two terms on rhs of (10) are the convection terms of \(\zeta\).

Two comparative calculations, CASE-4 and CASE-5, are carried out for the flow around the Wigley model. CASE-4 is carried out by SMM while CASE-5 by DMM. The Froude and the Reynolds numbers are 0.316 and \(10^4\) respectively. The calculations are carried out by the HITAC-680H computer of Information Processing Center of Hiroshima University.

The computational conditions are tabulated in Table 3. The number of grids of CASE-5 is about twice that of CASE-4.

Fig. 9 shows the wave patterns calculated by the two methods. Both are at the non-dimensional time \(T=3\). As can be seen clearly, the wave of CASE-4 is far from fully developed due to the numerical viscosity, but the wave of CASE-5 has much more developed than that of CASE-4 because the use of the fine mesh system.

Table 3 Comparisons of CASE-4 and CASE-5

<table>
<thead>
<tr>
<th>Method</th>
<th>CASE - 4</th>
<th>CASE - 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of grids</td>
<td>81,000 (120 X 45 X 15)</td>
<td>160,000 (200X50X15), 37,500 (100X25X15)</td>
</tr>
<tr>
<td>Memory size</td>
<td>22 MB</td>
<td>29 MB</td>
</tr>
<tr>
<td>CPU time</td>
<td>13.2 sec/step</td>
<td>11.1 sec/step</td>
</tr>
<tr>
<td>(C_p)</td>
<td>1.67 X 10^-3</td>
<td>1.80 X 10^-3</td>
</tr>
</tbody>
</table>

![Fig. 9 Comparison of Wave Patterns of Wigley Model (\(F_n=0.316, T=3.0\), Contour interval is 0.02 \(x Uo I2g\) and solid lines denote positive values). A: CASE-4, B: CASE-5](image)

Fig. 10 shows the comparison of the pressure on the hull. It is seen that the pressure has also much developed and the peak values are larger in CASE-5 than those in CASE-4. The pressure drag coefficients are compared in Table 3. The result of CASE-5 is larger than that of CASE-4 by 8 %. It is closer to the experimental value than that of CASE-4.

The CPU time and the memory size are compared in Table 3. In spite of the increment of the grid number to twice, the memory size increases only by 26 %. On the other hand, the CPU time is contrarily reduced by 16 %.

Through these comparative computations it is clearly

![Fig. 10 Comparison of Pressure Contours on Wigley Hull (\(F_n=0.316, T=3.0\), Contour interval is 0.02 \(C_p\) and solid lines denote positive values. Vertical exaggeration is 5 : 1), A: CASE-4, B: CASE-5](image)
proved that the fine mesh is necessary for a sound development of the free-surface elevation and the hull pressure and, to be more important, it is proved in this case also that the fine mesh is necessary only for the convection terms in the equations describing the problem.

It is worth while to point out here that the use of the third mesh system, which is even finer than the fine mesh in DMM, for the free-surface condition (9) or (10) may make it possible to have much more developed free-surface elevation for the numerical dissipation concerned with the free-surface elevation comes from the discretization of the convection terms in (10).

5. Concluding Remarks

The double mesh method (DMM) is introduced into the finite difference solution of the N-S equation to improve the computation efficiency. The method is to use more than one mesh system in the computation depending on the characteristics of the terms or the equation. Through several comparative computations of the flows around the circular cylinder and the Wigley model with the free-surface, following findings can be summarized.

1) The fine grid system necessary for the convection terms is not always necessary for the non-convective terms and equations such as the gradient of the pressure and the Poisson equation.

2) DMM is significantly effective for efficient finite difference calculation of the N-S equation. Both the CPU time and the memory size can be considerably reduced.

3) The use of DMM provides us more rational computations where grids are as fine as necessary while the CPU time and the memory size are not much increased, even decreased.

4) It is believed that the method can be applied to other problems including convection phenomena.

The authors wish to express their thanks to Prof. Y. Doi and Mr. S. H. Kwag at Hiroshima University for their discussions and helps.

References:


Appendix: Discretization Error of the Lagrangian Expression of Kinematic Free Surface Condition

Consider the two-dimensional case. Suppose \( P_{n}(X_{n-1}, Z_{n-1}) \) and \( P_{n}(X_{n}, Z_{n}) \) are two grids on the free-surface at \( t=n \) as shown in Fig. 11. At the next step \( t=n+1 \), these mark points move to \( P'_{n+1}(X'_{n-1}, Z'_{n-1}) \) and \( P'_{n}(X'_{n}, Z'_{n}) \) respectively. In the present research, the \( x, y \) coordinates of the grids on the free-surface are fixed and the \( z \) coordinate moves freely. The elevation of the new free-surface grid \( Q_{n} \) can be determined by \( P_{n} \) and \( P'_{n} \) as follows:

\[
\zeta_{n+1} = \zeta_{n} + (X_{n} - X'_{n}) \]

where

\[
k = \frac{Z_{n} - Z'_{n-1}}{X_{n} - X'_{n-1}} = \frac{Z_{n} - Z_{n-1} + (w_{n} - w_{n-1}) \Delta t}{x_{n} - x_{n-1} + (u_{n} - u_{n-1}) \Delta t}
\]

\[
\zeta_{n+1} = \zeta_{n} + w_{n} \Delta t - (u_{n} \Delta t) \frac{\zeta_{n} - \zeta_{n-1}}{x_{n} - x_{n-1}} + \frac{1}{1 + \Delta u \Delta t}
\]

or

\[
\frac{\zeta_{n+1} - \zeta_{n}}{\Delta t} = w_{n} - u_{n} \frac{\zeta_{n} - \zeta_{n-1}}{\Delta x} + \frac{1}{1 + \Delta u \Delta t}
\]

where

\[
\Delta w = w_{n} - w_{n-1}, \quad \Delta u = u_{n} - u_{n-1}
\]

(14) is the finite difference expression of (9) and it can be seen that (9) is equivalent to (10) in which the 1st order upstream difference scheme is used.