A Study of Grid Dependence in Navier-Stokes Solutions for Free Surface Flows around a Ship Hull

by Takanori Hino*, *Member*

Summary

Grid dependence in Navier-Stokes solution for free surface flows around a ship hull is investigated through numerical experiments. A Navier-Stokes solver used is a multigrid, finite-volume code with the artificial compressibility approach. Convection terms are evaluated by the third order upwind scheme based on the flux difference splitting. Time integration is carried out by the Runge-Kutta scheme. Nonlinear free surface conditions are implemented and a computational grid is re-generated to follow the free surface deformation. Free surface viscous flows around a Series 60 ($C_b=0.6$) hull are computed by the solver using three grids with different grid density. Solutions are compared with each other and with experimental data and grid dependence of solutions is discussed.

1. Introduction

CFD in ship hydrodynamics is expanding its applicability from the simplest case of a double model flow around a ship to various other targets, such as a self-propelled condition, a free surface flow or an oblique tow condition. Among them, free surface flows are taken into account in the present study. Free surface problems have been one of the main subjects in ship hydrodynamics. Recently a number of CFD codes have been developed which can cope with free surface flows. At the recent CFD Workshop 1994¹⁾, a free surface viscous flow around a ship is selected as one of three programs for comparative computations. The results computed by several Navier-Stokes solvers were collected and compared at the Workshop. It turned out that though the wave profiles on a ship hull were well simulated by some solvers, predictions of wave fields away from a ship were not so good as those obtained by potential flow codes. This inaccuracy is thought to come from poor grid resolution away from a hull.

In the present study, grid dependence of free surface viscous flow solutions is examined through numerical experiments. A Navier-Stokes solver used is a finite-volume, multigrid code which can compute a free surface flow efficiently. A Series 60 ($C_b=0.60$) is selected as a test model due to its extensive experimental²⁾ and computational data¹⁾. With three grids with different grid density, free surface viscous flows around this model are computed. Solutions are compared with each

other and with experimental data and grid dependence of solutions is discussed.

2. Computational Method

2.1 Governing Equations

The governing equations are the three-dimensional incompressible Navier-Stokes equations. With the introduction of artificial compressibility, they can be written as

$$\frac{\partial q}{\partial t} + \frac{\partial(e - e^v)}{\partial x} + \frac{\partial(f - f^v)}{\partial y} + \frac{\partial(g - g^v)}{\partial z} = 0 \quad (1)$$

where

$$q = \begin{bmatrix} p \\ u \\ v \\ w \end{bmatrix}, \quad e = \begin{bmatrix} \beta u \\ u^2 + p \\ uv \\ uw \end{bmatrix},$$

$$f = \begin{bmatrix} \beta v \\ vu \\ v^2 + p \\ vw \end{bmatrix}, \quad g = \begin{bmatrix} \beta w \\ wu \\ wv \\ w^2 + p \end{bmatrix} \quad (2)$$

$$e^v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{zx} \end{bmatrix}, \quad f^v = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{yz} \end{bmatrix}, \quad g^v = \begin{bmatrix} 0 \\ \tau_{zx} \\ \tau_{yz} \\ \tau_{zz} \end{bmatrix} \quad (3)$$

and

$$\tau_{xx} = \left(\frac{1}{R} + \nu_t\right) \left(2 \frac{\partial u}{\partial x}\right), \quad \tau_{yy} = \left(\frac{1}{R} + \nu_t\right) \left(2 \frac{\partial v}{\partial y}\right),$$

$$\tau_{zz} = \left(\frac{1}{R} + \nu_t\right) \left(2 \frac{\partial w}{\partial z}\right), \quad \tau_{xy} = \left(\frac{1}{R} + \nu_t\right) \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right),$$

$$\tau_{yz} = \left(\frac{1}{R} + \nu_t\right) \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right), \quad \tau_{zx} = \left(\frac{1}{R} + \nu_t\right) \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right)$$

(x, y, z) are the Cartesian coordinates and (u, v, w) are the velocity components. p is the modified pressure defined by

$$p = p^* + z/F^2 \quad (4)$$

* Ship Research Institute.

where p^* is the original pressure and F is the Froude number. R is the Reynolds number and ν_t is kinematic eddy viscosity determined from a turbulence model. β is the parameter of artificial compressibility. Artificial compressibility makes the system of equations (1) hyperbolic and advanced techniques developed for compressible flow solvers can be applied to the present incompressible flow solver with the expense of time accuracy.

2.2 Spatial Discretization

A finite-volume approach is adopted for spatial discretization, in which a computational domain is divided into small hexagonal cells. Following Kodama³⁾, the cell-centered layout is employed, that is, flow variables (pressure, velocity and eddy viscosity) are stored in the center of each cell. Integration of the governing equation (1) over a cell yields

$$\frac{\partial}{\partial t} \int_V \mathbf{q} dV + \oint_{\partial V} ((\mathbf{e} - \mathbf{e}^v) dS_x + (\mathbf{f} - \mathbf{f}^v) dS_y + (\mathbf{g} - \mathbf{g}^v) dS_z) = 0 \quad (5)$$

where the divergence theorem is applied to the momentum and the mass flux integral. V is the volume of a cell and ∂V is its boundary. (S_x, S_y, S_z) are the area vector component of the cell boundaries.

The control volume is time dependent when the moving free surface boundary is taken into account. However, since the present artificial compressibility formulation aims to obtain a steady state solution, all terms associated with cell volume change and grid movement velocity which should appear in the above integration (5) are neglected. This reduces a computational cost significantly and upon convergence to steady state, this simplification does not affect accuracy of solutions.

In the discretized form, the above equation is written as

$$\frac{d}{dt} (\mathbf{q}_{ijk} V_{ijk}) + \mathbf{C}_{ijk} - \mathbf{D}_{ijk} = 0 \quad (6)$$

where ijk is the cell numbering and $\mathbf{C}_{ijk}$ and $\mathbf{D}_{ijk}$ are the convection and viscous terms which can be expressed as

$$\mathbf{C}_{ijk} = \mathbf{E}_{i+1/2} - \mathbf{E}_{i-1/2} + \mathbf{F}_{j+1/2} - \mathbf{F}_{j-1/2} + \mathbf{G}_{k+1/2} - \mathbf{G}_{k-1/2} \quad (7)$$

and

$$\mathbf{D}_{ijk} = \mathbf{E}_{i+1/2}^v - \mathbf{E}_{i-1/2}^v + \mathbf{F}_{j+1/2}^v - \mathbf{F}_{j-1/2}^v + \mathbf{G}_{k+1/2}^v - \mathbf{G}_{k-1/2}^v \quad (8)$$

where $\pm 1/2$ of the subscripts correspond to cell interfaces. $\mathbf{E}$, $\mathbf{F}$ and $\mathbf{G}$ are the convective fluxes and $\mathbf{E}^v$, $\mathbf{F}^v$ and $\mathbf{G}^v$ are the viscous fluxes on cell interfaces.

The convective flux $\mathbf{E}$ is expressed as

$$\mathbf{E} = \begin{bmatrix} \beta U \\ uU + pS_x \\ vU + pS_y \\ wU + pS_z \end{bmatrix} \quad (9)$$

where

$$U = S_x u + S_y v + S_z w \quad (10)$$

The convective fluxes $\mathbf{E}_{i+1/2}$ etc. appeared in Eq. (7) are evaluated by the upwind scheme based on the flux

difference splitting as follows:

$$\mathbf{E}_{i+1/2} = \frac{1}{2} [\mathbf{E}(\mathbf{q}^L) + \mathbf{E}(\mathbf{q}^R) - |\mathbf{A}|(\mathbf{q}^L - \mathbf{q}^R)]_{i+1/2} \quad (11)$$

where $\mathbf{q}_{i+1/2}^L$ and $\mathbf{q}_{i+1/2}^R$ are the left and right interface values of $\mathbf{q}$. $\mathbf{E}(\mathbf{q}^L)$ and $\mathbf{E}(\mathbf{q}^R)$ are evaluated from Eq. (9) with $\mathbf{q}^L$ or $\mathbf{q}^R$ and the area vectors (S_x, S_y, S_z) at the interface $(i+1/2)$. The matrix $|\mathbf{A}|$ is defined as follows. First, the flux Jacobian matrix $\mathbf{A}$ of $\mathbf{E}$ is defined as

$$\mathbf{A} = \frac{\partial \mathbf{E}}{\partial \mathbf{q}} = \mathbf{R} \mathbf{A} \mathbf{L} \quad (12)$$

where $\mathbf{A}$ is

$$\mathbf{A} = \text{diag}(\lambda_1, \lambda_2, \lambda_3, \lambda_4) \quad (13)$$

and λ_m ($m=1, 4$) are the eigenvalues of $\mathbf{A}$ expressed as

$$\lambda_1 = \lambda_2 = U$$

$$\lambda_3 = U + \sqrt{U^2 + \beta(S_x^2 + S_y^2 + S_z^2)}$$

$$\lambda_4 = U - \sqrt{U^2 + \beta(S_x^2 + S_y^2 + S_z^2)}$$

$\mathbf{R}$ and $\mathbf{L}$ are the similarity transformation matrices of the eigenvectors of $\mathbf{A}$. $|\mathbf{A}|$ is expressed as

$$|\mathbf{A}| = \mathbf{R} |\mathbf{A}| \mathbf{L} \quad (14)$$

where

$$|\mathbf{A}| = \text{diag}(|\lambda_1|, |\lambda_2|, |\lambda_3|, |\lambda_4|) \quad (15)$$

In the evaluation of $|\mathbf{A}|$, the metrics at the interface $(i+1/2)$ and $\mathbf{q}_{i+1/2} = 1/2(\mathbf{q}_{i+1/2}^L + \mathbf{q}_{i+1/2}^R)$ are used.

Finally, the interface values can be computed by the MUSCL approach as

$$\begin{aligned} \mathbf{q}_{i+1/2}^L &= \frac{2}{6} \mathbf{q}_{i+1} + \frac{5}{6} \mathbf{q}_i - \frac{1}{6} \mathbf{q}_{i-1}, \\ \mathbf{q}_{i+1/2}^R &= \frac{2}{6} \mathbf{q}_i + \frac{5}{6} \mathbf{q}_{i+1} - \frac{1}{6} \mathbf{q}_{i+2} \end{aligned} \quad (16)$$

The above expression corresponds to the third-order upwind scheme. $\mathbf{F}$ and $\mathbf{G}$ are evaluated in the similar manner.

The viscous flux $\mathbf{E}^v$ can be written as

$$\mathbf{E}^v = S_x \mathbf{e}^v + S_y \mathbf{f}^v + S_z \mathbf{g}^v \quad (17)$$

Application of the divergence theorem to a control volume surrounding a cell interface gives velocity gradient at a cell interface which is required in the evaluation of the above flux. $\mathbf{F}^v$ and $\mathbf{G}^v$ are evaluated in the similar manner.

Turbulence model used is the algebraic model by Baldwin and Lomax⁴⁾. To apply this model to free surface flow problems, modification is made on a search of the function $F_{\max}$ in a velocity profile. Near free surface, the search of $F_{\max}$ is limited to proximity of a solid wall or a wake centerline, which prevents the wrong choice of $F_{\max}$ due to free surface waves.

2.3 Free Surface Conditions

Free surface conditions, both dynamic and kinematic, are implemented in the scheme as follows. If both stress of air and surface tension of water are neglected, the dynamic condition becomes no stress on free surface. From zero tangential stress condition, the viscous flux $\mathbf{F}^v$ vanishes on free surface and the velocity boundary condition on a free surface can be approximated by zero gradient extrapolation. The condition of zero normal stress on a free surface gives the Dirichlet condition for

pressure on a free surface, which is approximated by

$$p = p_{air} + h/F^2 \quad (18)$$

where p_{air} ($=0$) is atmospheric pressure.

The kinematic condition is used to update free surface height. If free surface height is written as $H = h(x, y; t) - z = 0$, the kinematic condition becomes

$$\frac{DH}{Dt} = \frac{\partial h}{\partial t} + u \frac{\partial h}{\partial x} + v \frac{\partial h}{\partial y} - w = 0 \quad (19)$$

In the present scheme, the kinematic free surface condition is formulated based on the mass conservation. Let the area vector of free surface control volume be $\mathbf{S}^{FS}$ and the velocity vector on free surface be $\mathbf{u}^{FS}$. The change of volume due to the free surface movement in the time increment Δt is expressed as

$$\Delta t \mathbf{S}^{FS} \cdot \mathbf{u}^{FS}$$

This change of volume corresponds to the change of volume due to wave height movement Δh in the direction of $\mathbf{r}$ which is arbitrary unless $\mathbf{S}^{FS} \cdot \mathbf{r} = 0$, i. e.,

$$\Delta h \mathbf{S}^{FS} \cdot \mathbf{r} = \Delta t \mathbf{S}^{FS} \cdot \mathbf{u}^{FS} \quad (20)$$

Thus,

$$\frac{\Delta h}{\Delta t} = \frac{\mathbf{S}^{FS} \cdot \mathbf{u}^{FS}}{\mathbf{S}^{FS} \cdot \mathbf{r}} \quad (21)$$

If $\mathbf{r}$ is $(0, 0, 1)^T$, that is, the wave height is defined as $z = h$,

$$\begin{aligned} \frac{\Delta h}{\Delta t} &= \frac{\mathbf{S}^{FS} \cdot \mathbf{u}^{FS}}{\mathbf{S}^{FS} \cdot \mathbf{r}} \\ &= \frac{S_x^{FS}}{S_z^{FS}} u + \frac{S_y^{FS}}{S_z^{FS}} v + w \end{aligned} \quad (22)$$

where

$$\begin{aligned} \mathbf{S}^{FS} &= (S_x^{FS}, S_y^{FS}, S_z^{FS})^T \\ &= \frac{|\mathbf{S}^{FS}|}{\sqrt{1 + \left(\frac{\partial h}{\partial x}\right)^2 + \left(\frac{\partial h}{\partial y}\right)^2}} \left(-\frac{\partial h}{\partial x}, -\frac{\partial h}{\partial y}, 1 \right)^T \end{aligned} \quad (23)$$

This yields

$$\frac{\Delta h}{\Delta t} = -u \frac{\partial h}{\partial x} - v \frac{\partial h}{\partial y} + w \quad (24)$$

and it is identical to Eq. (19).

In the present scheme, the control volume for Eq. (21) is constructed around a grid point on a free surface as shown in Fig. 1(a), by which wave height at a grid point is obtained without interpolation. To compute area vectors of the control volume around the point $(i+1/2, k+1/2)$, the coordinates of the midpoints, i. e., $(i, k+1/2)$, $(i+1, k+1/2)$, $(i+1/2, k)$ and $(i+1/2, k+1)$

Fig. 1 Control volumes for wave height computation. (a) internal point: (b) body boundary point

+1) are used (see Fig. 1(a)). x and y coordinates are obtained by the averaging of the coordinates of the end points, while z coordinate, wave height, at the midpoint is evaluated in the third-order upwind manner similar to the momentum flux. Velocity at $(i+1/2, k+1/2)$ is obtained by the averaging. The vector $\mathbf{r}$ in Eq. (21) is taken as the direction of the grid line of the background grid, which is used as the reference grid for the grid generation at each time step. The background grid coordinates are given to the solver at the beginning of computation. This grid covers the domain below and above the still water level. When wave height distribution is obtained as a part of solution, the actual computational grid is generated from the background grid by the spline interpolation along each grid line in the girth direction.

It should be noted here that the wave height along a ship hull is not extrapolated but computed by the above procedure using the control volume shown in Fig. 1(b) and velocity of a cell adjacent to a solid wall. It can be shown by the summation of all the control volumes on a free surface that the global mass conservation is approximated by this procedure.

2.4 Boundary Conditions

The discrete boundary conditions are summarized in Table 1. It should be noted that the outflow and outer (side) boundary conditions for wave height is to give the value of zero. To avoid the reflection of free surface waves in the boundaries, various methods have been investigated⁹⁾. Among them, Hinatsu⁹⁾ showed wave damping is effective in his simulation of progressive waves. Hinatsu added damping terms to the vertical momentum equations in such a way that vertical velocity is suppressed near outer boundaries. The present scheme adopts the similar but simpler approach. The damping term is added to the free surface kinematic condition to reduce the amplitude of waves approaching outer boundaries. The performance of this method has been proven in 2-D case⁷⁾. In 3-D, the kinematic condition (21) is modified as follows:

$$\frac{\Delta h}{\Delta t} = \frac{\mathbf{S}^{FS} \cdot \mathbf{u}^{FS}}{\mathbf{S}^{FS} \cdot \mathbf{r}} - W(x, y)h \quad (25)$$

where h is wave height and $W(x, y)$ is a weight function defined as

$$\begin{aligned} W(x, y) &= A \times \max(W_x(x), W_y(y)) \\ W_x(x) &= \begin{cases} \left(\frac{x - x_d}{x_a - x_d} \right)^2 & \text{if } x_d \leq x \leq x_a \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

Table 1 Boundary conditions

Boundary	(u, v, w)	p	h (wave height)
Inflow	$(1, 0, 0)$	zero gradient	0
Outflow	zero gradient	0	0
Outer	zero gradient	0	0
Centerplane	symmetry	symmetry	symmetry
Body	$(0, 0, 0)$	zero gradient	computed
Free Surface	zero gradient	h/F^2	—

$$W_y(y) = \begin{cases} \left(\frac{y - y_d}{y_o - y_d} \right)^2 & y_d \leq y \leq y_o \\ 0 & \text{otherwise} \end{cases}$$

where x_d, y_d is the coordinate from which the damping region starts and x_o, y_o is the location of outflow and side boundaries. A is the parameter which controls the amount of damping.

2.5 Time Integration

Since the present scheme uses artificial compressibility, time accuracy of a solution is lost and time integration is just a way to drive a solution to a steady state. The explicit multi-stage scheme based on the Runge-Kutta method first developed by Jameson⁹⁾ is used here.

The continuity and momentum equations (6) is rewritten as

$$\frac{d\mathbf{q}_{ijk}}{dt} = -\mathbf{R}_{ijk}(\mathbf{q}) \quad (26)$$

where

$$\mathbf{R}(\mathbf{q}) = \frac{1}{V_{ijk}} [\mathbf{C}_{ijk}(\mathbf{q}) - \mathbf{D}_{ijk}(\mathbf{q})] \quad (27)$$

The five-stage scheme used here to integrate Eq. (26) from the (p) -th time step to $(p+1)$ -th step can be expressed as

$$\begin{aligned} \mathbf{q}^{(0)} &= \mathbf{q}^p \\ \mathbf{q}^{(1)} &= \mathbf{q}^{(0)} - \alpha_1 \Delta t \mathbf{R}^{(0)}(\mathbf{q}) \\ \mathbf{q}^{(2)} &= \mathbf{q}^{(0)} - \alpha_2 \Delta t \mathbf{R}^{(1)}(\mathbf{q}) \\ &\dots \\ \mathbf{q}^{(5)} &= \mathbf{w}^{(0)} - \alpha_5 \Delta t \mathbf{R}^{(4)}(\mathbf{q}) \\ \mathbf{q}^{p+1} &= \mathbf{q}^{(5)} \end{aligned} \quad (28)$$

where

$$\alpha_1 = 1/4 \quad \alpha_2 = 1/6 \quad \alpha_3 = 3/8 \quad \alpha_4 = 1/2 \quad \alpha_5 = 1$$

$\mathbf{R}^{(m)}(\mathbf{q})$ is the residual evaluated at the m -th stage and is expressed as

$$\mathbf{R}^{(m)}(\mathbf{q}) = \frac{1}{V_{ijk}} [\mathbf{C}_{ijk}^{(m)}(\mathbf{q}) - \mathbf{D}_{ijk}^{(m)}(\mathbf{q})] \quad (29)$$

where

$$\begin{aligned} \mathbf{C}_{ijk}^{(m)} &= \mathbf{C}_{ijk}(\mathbf{q}^{(m)}) \\ \mathbf{D}_{ijk}^{(m)} &= \begin{cases} \mathbf{D}_{ijk}(\mathbf{q}^{(0)}) & \text{if } m=0 \\ \gamma_m \mathbf{D}_{ijk}(\mathbf{q}^{(m)}) + (1-\gamma_m) \mathbf{D}_{ijk}^{(m-1)} & \text{if } m \geq 1 \end{cases} \end{aligned}$$

The values of γ_m are

$$\gamma_1 = 0 \quad \gamma_2 = 0.56 \quad \gamma_3 = 0 \quad \gamma_4 = 0.44$$

These particular coefficients are chosen in such a way that the scheme has an excellent stability property.

2.6 Convergence Acceleration

Convergence to steady state can be accelerated by various ways when time accuracy of a scheme is not required as is the case of the present solver. Among them, local time stepping, implicit residual smoothing and a multigrid method are employed in the present scheme following Jameson⁹⁾.

Since the time integration is made by the explicit scheme, the time step Δt is restricted from the Courant-Friedrichs-Lewy (CFL) condition. The local time stepping is the method in which the solution at each point proceeds in time with the time step defined locally from the local stability limit. Thus, the time step is defined as

$$\Delta t_{ijk} = \left[\frac{\Delta t^v \Delta t^c}{\Delta t^v + \Delta t^c} \right]_{ijk} \quad (30)$$

where

$$\Delta t_{ijk}^c = CFL \frac{V_{ijk}}{|\lambda_{ijk}^c|} \quad \Delta t_{ijk}^v = VSL \frac{V_{ijk}}{|\lambda_{ijk}^v|}$$

where λ_{ijk}^c and λ_{ijk}^v are the sum of the maximum wave speeds in each spatial direction for convection terms and for viscous terms, respectively. CFL and VSL are the Courant number and the viscous stability limit determined by the particular time stepping scheme.

The residual smoothing is used to increase the bound of the stability limit of the time stepping scheme⁹⁾, in which the residual $\mathbf{R}_{ijk}$ is replaced by the solution of the following equation

$$(1 - \varepsilon_k \delta_k^2)(1 - \varepsilon_r \delta_r^2) \tilde{\mathbf{R}}_{ijk} = \mathbf{R}_{ijk} \quad (31)$$

where δ_k^2 is a central difference operator of second derivative in k -direction and ε_k is the smoothing coefficient evaluated locally by using the maximum wave speeds¹⁰⁾.

A multigrid method is known as the extremely efficient way to get fast convergence. The concept of the multigrid time stepping applied to the solution of hyperbolic equations by Jameson⁹⁾ is to compute corrections to the solution on a fine grid by the time-stepping on a coarser grid. The successively coarser grids can be generated by deleting the alternate points along each grid line of the finer grid.

The procedure of the multigrid method is as follows. Equations to be solved is written as

$$\frac{d\mathbf{q}}{dt} = -\mathbf{R}(\mathbf{q}) \quad (32)$$

and the subscript k denotes the grid index.

First, the solution $\mathbf{q}_k$ is obtained in the fine grid (k) by solving

$$\frac{d\mathbf{q}_k}{dt} = -\mathbf{R}_k(\mathbf{q}_k) \quad (33)$$

by the Runge-Kutta scheme described above. The solution is transferred from the fine grid (k) to the next coarser grid ($k+1$) by

$$\mathbf{q}_{k+1}^{(0)} = T_k^{k+1} \mathbf{q}_k \quad (34)$$

T_k^{k+1} is a transfer operator for a solution defined as

$$T_k^{k+1} \mathbf{q}_k = (\sum \mathbf{q}_k V_k) / V_{k+1} \quad (35)$$

where the sum is over eight cells to be merged to construct a cell in the coarser grid and V is a cell volume. The solution in the coarse grid is updated by solving the equation

$$\frac{d\mathbf{q}_{k+1}}{dt} = -\mathbf{R}_{k+1}(\mathbf{q}_{k+1}) - \mathbf{P}_{k+1} \quad (36)$$

with the same Runge-Kutta scheme and $\mathbf{q}_{k+1}^*$ is obtained. $\mathbf{P}$ in the above equation is the forcing terms in the coarse grid ($k+1$) defined as

$$\mathbf{P}_{k+1} = Q_k^{k+1} \mathbf{R}_k(\mathbf{q}_k) - \mathbf{R}_{k+1}(\mathbf{q}_{k+1}^{(0)}) \quad (37)$$

where Q_k^{k+1} is another transfer operator for a residual and is the sum of the residuals of eight cells to be merged to construct a cell in the coarser grid

$$Q_k^{k+1} \mathbf{R}_k(\mathbf{q}_k) = \sum (\mathbf{R}_k) \quad (38)$$

$\mathbf{q}_{k+1}^* - \mathbf{q}_{k+1}^{(0)}$ gives the correction of the solution at the grid ($k+1$) which reduces the residual in the finer grid

due to the forcing terms. Finally, the correction is transferred back from the coarse grid ($k+1$) to the fine grid (k) by

$$\mathbf{q}_k^* = \mathbf{q}_k + I_{k+1}^k(\mathbf{q}_{k+1}^* - \mathbf{q}_{k+1}^{(0)}) \quad (39)$$

where I_{k+1}^k is an interpolation operator for a correction. Trilinear interpolation in the computational space is used for I_{k+1}^k in the present scheme. A multigrid cycle employed here is V-cycle in which the equations are first solved at the finest grid and the solution moves down to the coarsest grid with an update of a solution at each grid and the interpolation is used in the transfer of correction from the coarsest grid to the finest one.

Also, the full multigrid strategy (FMG) is used. In the FMG, the solution procedure starts with the coarsest grid and after some cycles in the coarsest grid, the solution is interpolated into the finer grid. Then the solution in this grid level is obtained by the two-level multigrid method. After several multigrid cycles, the solution is again interpolated into the next finer grid. This time, the solution is obtained by three-level multigrid method. The procedure repeats until the solution of the finest grid is obtained.

3. Results and Discussions

3.1 Computational Conditions

A ship model used here is a Series 60 ($C_b=0.6$) hull. Following the computational condition used in the CFD Workshop 1994¹¹ which corresponds to the measurement done by Toda et al.²¹, the Froude number is set 0.316 and the Reynolds number is set 4×10^6 . All the experimental data used for comparison here are by Toda et al.²¹. The computations are carried out using three different grids. The coarse grid consists of $105 \times 25 \times 57$ grid points in the streamwise, the girth and the normal directions, while the medium grid has $105 \times 33 \times 105$ and the fine grid has $153 \times 33 \times 105$. The solution domain for three cases is

$$-1 \leq x/L \leq 3 \quad 0 \leq y/L \leq 2$$

$$-2 \leq z/L \leq (\text{the maximum wave height})$$

where the origin of the coordinates is the intersection of F. P. and still water surface. The wave damping zone is set

$$2 \leq x/L \leq 3 \quad 1 \leq y/L \leq 2$$

The minimum spacing in each direction is listed in Table. 2.

The transverse sections of three grids are shown in Fig. 2. (The horizontal sections can be seen in Fig. 3.) Between the coarse and the medium grids, the streamwise grid density is kept unchanged and grid points in the girthwise and the normal directions are increased. The fine grid is obtained to distribute more

grid points along streamwise direction than the medium grid. The grid is generated in such a way that the points are clustered in the region where waves develop as well as in the boundary layer and the wake region.

3.2 Results

Fig. 3 shows the comparison of wave contours and the grids on a free surface. Three computational results simulate the bow wave and the first trough fairly well. The differences are apparent in the region away from the hull. The medium grid case improves the resolution of the fore and the aft shoulder waves which are not clearly captured in the coarse grid case. As pointed out by Mori and Hinatsu¹¹, the grid spacing in the y -direction is as important as in the x -direction for the accurate simulation of a wave field. The finest grid case improves the results further, particularly in the wake

Fig. 2 Transverse sections of computational grids

Table 2 Minimum spacing of grids

Grid	Streamwise dir.	Girthwise dir.	Normal dir.
$105 \times 25 \times 57$	0.01	0.001	0.00005
$105 \times 33 \times 105$	0.01	0.00075	0.00005
$153 \times 33 \times 105$	0.0075	0.00075	0.00005

Fig. 3 Comparison of wave contours and free surface grids

Fig. 4 Comparison of wave profiles on a ship hull

Fig. 5 Comparison of longitudinal wave cuts

region.

Fig. 4 is the comparison of the wave profiles on a ship hull. The computed results are close to each other and agree well the experimental data. This shows the wave profile on a ship hull is not so sensible to the grid density change in y -direction as the waves away from a hull.

Fig. 5 depicts the comparison of the longitudinal wave cuts at $y=0.1739$ and 0.2723 . In both sections, waves are well simulated by the present solver in the region $x \leq 1$ where the grid resolution seems to be sufficient. Note that in the $y=0.1739$ section, the measured data shows the irregular wave form around $x=0.75$, this is due to the presence of a crest of the fore shoulder wave²⁾. In the computations, however, this wave does not appear maybe because of the difference of a phase and an amplitude of the fore shoulder wave.

Fig. 6 is the similar comparison with the transverse wave cuts. In the $x=0.8$ section, the coarse grid case fails to predict the small wave hump located around $y=0.18$ which corresponds to the shoulder wave. Also, the peak of the wave crest is poorly simulated. The grid refinement reduces the above two discrepancies.

However, even in the finest grid case, there still remains the difference from the experimental data. The similar tendency can be seen in the $x=1.0$ section. The numerical results show dissipated wave amplitude and the inwardly shifted phase compared with the measured data.

Fig. 7 shows the wake distributions at $x=1.2$. As was shown in the CFD Workshop Tokyo¹⁾, the predictions of a wake distribution are poor. It turned out that the present grid refinement does not improve the quality of a solution in terms of a wake distribution. More efforts should be made to overcome this problem.

Fig. 8 depicts the pressure distributions on a hull. Three cases show almost the same result except near the bow end, where the finest grid case shows better resolution. To capture the steep pressure gradient at the bow, smaller grid spacing may be required.

The resistance coefficients are summarized in Table. 3. Three cases give almost the same resistance values. This is consistent with the fact that the wave profile and

the pressure distribution on a ship hull is almost the same among three cases. However, there is a difference between the measured data and the computed one and the major part of this difference comes from the pressure component (See Table. 3.). This may suggest that although the resistance computed here seems to be somewhat grid-independent, the resistance, particularly pressure resistance, is not computed well. More efforts should be made to enable accurate prediction of resistance values as well as wake distributions.

Conclusions

A finite-volume multigrid Navier-Stokes solver is used to study grid dependence of free surface flow solutions around a ship hull. A wave field away from a hull is significantly influenced by grid spacing. With sufficiently small spacing, free surface viscous flows can be simulated accurately. On the other hand, a wave profile along a ship hull and a resistance value do not change drastically by the change of grid density away from a hull.

Therefore, when resistance value of a single body is of interest, the grid density away from a hull is not so critical. However, in the situations like a flow analysis of a catamaran or tandem hydrofoils, where waves generated by one body interact with another body, the grid density away from a hull should be chosen more carefully.

Acknowledgments

The author is grateful to Profs. A. Jameson and L. Martinelli at Princeton University for their guidance and help while the author studied the numerical method of this research at Princeton. The author also would like to thank the members of CFD group at SRI for their valuable discussions.

References

- 1) —, "Proceedings CFD Workshop Tokyo 1994", Vols. 1 and 2, (1994).
- 2) Toda, Y., Stern, F. and Longo, J., "Mean-Flow Measurements in the Boundary Layer and Wake and Wave Field of a Series 60 $C_b=0.6$ Ship Model — Part 1: Froude Numbers 0.6 and 0.316.", J. Ship Research, Vol. 36, No. 4, (1992).
- 3) Kodama Y., "Computation of Ship's Resistance Using an NS Solver with Global Conservation — Flat Plate and Series 60 ($C_b=0.6$) Hull—", J. Soc. Naval Archt. Jpn, Vol. 172, (1992).
- 4) Baldwin B. S. and Lomax H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows", AIAA Paper 78-257, (1978).
- 5) Romante, J. E., "Absorbing Boundary Conditions for Free Surface Waves", J. Comput. Phys., Vol. 99, (1992).
- 6) Hinatsu, M., "Numerical Simulation of Unsteady Viscous Nonlinear Waves Using Moving Grid System Fitted on a Free Surface", J. Kansai Soc. Naval Arch. Jpn., Vol. 217, (1992).

Fig. 6 Comparison of transverse wave cuts

EXP(IIHR).

153 × 33 × 105 Grid.

105 × 33 × 105 Grid.

105 × 25 × 57 Grid.

Fig. 7 Comparison of wake contours and cross flow vectors at $x/L=1.2$

Fig. 8 Comparison of pressure distribution on a ship hull. Contour interval is 0.01.

Table 3 Resistance coefficients

Grid	C_T ($= R_T / 0.5 \rho U^2 S$)	C_F ($= R_F / 0.5 \rho U^2 S$)	C_p ($= C_T - C_F$)
105 × 25 × 57	5.48×10^{-3}	3.61×10^{-3}	1.87×10^{-3}
105 × 33 × 105	5.50×10^{-3}	3.64×10^{-3}	1.86×10^{-3}
153 × 33 × 105	5.53×10^{-3}	3.65×10^{-3}	1.88×10^{-3}
Experiment	5.96×10^{-3}	3.52×10^{-3}	2.44×10^{-3}

Experiment: $F = 0.3089$, $R = 4.1 \times 10^6$.

- 7) Hino, T., Martinelli, L. and Jameson A., "A Finite-Volume Method with Unstructured Grid for Free Surface Flow Simulations", Proc. 6th Int. Conf. on Num. Ship Hydro., (1993).
- 8) Farmer, J., Martinelli L. and Jameson A., "A Fast Multigrid Method for Solving the Nonlinear Ship Wave Problem with a Free Surface", Proc. 6th Int. Conf. on Num. Ship Hydro., (1993).
- 9) Jameson, A., "Computational Transonics", Comm. Pure Appl. Math., Vol. XLI, (1988).
- 10) Liu F. and Jameson A., "Multigrid Navier-Stokes Calculations for Three-Dimensional Cascades", AIAA J. Vol. 31, No. 10, (1993).
- 11) Mori K. and Hinatsu M., "Review of Program 1 Viscous Flow around Series 60 with Free-Surface", Proc. CFD Workshop Tokyo 1994, Vol. 2, (1994).