YACHT97: A Fully Viscous Nonlinear Free-Surface Analysis Tool for IACC Yacht Design

J. Farmer, L. Martinelli, A. Jameson
Princeton University, Princeton, New Jersey, USA

Proceedings
12th Chesapeake Sailing Yacht Symposium
Annapolis

January, 1995
Abstract

A novel free-surface computer program, used to calculate the viscous flow field about a full configuration sailing yacht, is presented. The numerical techniques incorporated into the computer program - YACHT97 - facilitate accurate quantification of the residuary and the frictional drag components through the use of a "moving" computational grid which conforms to the exact wetted surface area of the hull. Use of a full field approach, whereby the entire flow domain is discretized, permits the simulation of the boundary layer and wavemaking interaction as well as the onset of vortex structures which appear downstream of lift producing appendages. A recently implemented free-surface discretization permits the bow wave evolution to proceed to the near-breaking point. Computed results include comparisons between the inviscid and viscous flowfields about a bare-hull configuration and a full configuration with keel and rudder appendages.

1 Introduction

The prediction of the total drag experienced by an advancing ship is a complicated problem which requires a thorough understanding of the hydrodynamic forces acting on the ship hull, the physical processes from which these forces arise and their mutual interaction. For instance, it is well established that an advancing ship generates a complex flow field which consists of both the wave structure or pattern and the viscous boundary layer and wake. These two features of the flow field result in the well known drag components referred to as the wave resistance and the viscous resistance, respectively. The wave resistance arises from the continuous transfer of energy, from the ship's propulsor mechanism to the fluid, required to maintain the wave structure that extends well aft of the ship. The viscous resistance manifests itself through the stresses which act tangentially on the wetted surface of the ship hull, through the absence of "perfect fluid streamlines" due to boundary layer growth and separation and through the presence of vortices. It is also well established that the boundary layer and wake can influence the resulting wave structure through displacement thickness effects and the wave structure can influence the boundary layer through wave elevation and wave-induced pressure gradient effects [1, 2]. It is this mutual interaction that renders the prediction of the total drag of an advancing ship a challenging task, whether from experimental, analytical or computational standpoints.

The advent of powerful computers - exhibiting both fast processing speed and large storage capabilities - has now made possible computational solutions of the full set of mathematical equations which describe the coupled wave structure and viscous boundary layer interaction. Notable previous computational approaches in this area include: the finite difference, velocity-pressure coupling approach of Hino [3]; the finite volume, velocity-pressure coupling approach of Miyata et. al. [4]; and the interactive approach of Tahara et. al. [5] which com-

---

1 To appear in, Proceedings, 12th Chesapeake Sailing Yacht Symposium, 1993.
* Research Associate.
† Assistant Professor.
‡ Professor.
bines the finite analytic approach of Chen et al. [6] and the “SPLASH” panel method of Rosen et al. [7]. These methods all represent major advances in the computational solution of the coupled wave structure and boundary layer interaction problem as it applies to ship hulls in general. However, they are all computationally intensive, requiring significant amounts of CPU time, and more importantly, they all rely on a linearization of the free surface boundary conditions.

Unlike the above contributions, the task of computationally predicting IACC yacht hull performance has relied primarily through techniques which decouple the wave structure and boundary layer interaction. The work of Boppe et al. [8] combined the state-of-the-art in aerospace CFD techniques with a linearized free surface boundary condition to investigate performance predictions for the 12-meter yacht Stars and Stripes ’87. A major conclusion stated in this work points to the need for a “... simultaneous solution of the lift-induced drag, wave resistance, and viscous/turbulence drag within a single solution...” and suggests that this accomplishment will bring the community a step closer to numerical optimization for yacht design. A significant outgrowth of the work of Boppe et al. is the SPLASH free surface code developed by Rosen et al. [7]. The SPLASH methodology was shown to accurately capture details in wave resistance and induced drag components for full configuration IACC yacht models.

The motivation behind the development of YACHT97 follows directly from Boppe’s suggestion of a “single-shot” solution to the entire problem. YACHT97 is a specialization of our recently developed methodology used to simulate the wave structure and boundary layer interaction for the ship hull problem mentioned above [9, 10, 11]. The method employs a fully nonlinear free surface algorithm which permits the free surface to conform exactly to the hull geometry as the simulation proceeds. This feature is crucial for accurate predictions of stern flow separation and the wave drag component. The method also employs a full field discretization that is linked to the free surface. This feature permits the simulation of both flow separation off the hull surface and the vortex structure which appears downstream off lift producing appendages. The numerical techniques that comprise the majority of the flow solver follow those employed by Jameson [12] for the solution of compressible flows about complete aircraft configurations. These techniques have proven to be both accurate and extremely efficient.

2 Mathematical Model

Figure 1 shows the reference frame and ship location used in this work. A right-handed coordinate system Oxyz, with the origin fixed at midship on the mean free surface is established. The z direction is positive upwards, y is positive towards the starboard side and x is positive in the aft direction. The free stream velocity vector is parallel to the x axis and points in the same direction. The ship hull pierces the uniform flow and is held fixed in place, i.e. the ship is not allowed to sink (translate in z direction) or trim (rotate in x – z plane).

![Reference Frame and Ship Location](image)

2.1 Bulk Flow

For a viscous incompressible fluid moving under the influence of gravity, the continuity equation and the Reynolds averaged Navier-Stokes equations may be put in the form [3],

\[
 u_x + v_y + w_z = 0 
\]

\[
 u_t + uu_x + vu_y + wu_z = -\psi_x + (Re^{-1} + \nu_1) (\nabla^2 u)
\]

\[
 v_t + uw_x + vv_y + wv_z = -\psi_y + (Re^{-1} + \nu_1) (\nabla^2 v) 
\]

\[
 w_t + uw_x + vw_y + wv_z = -\psi_z + (Re^{-1} + \nu_1) (\nabla^2 w). 
\]

Here, \( u = u(x, y, z, t), v = v(x, y, z, t) \) and \( w = w(x, y, z, t) \) are the mean total velocity components in the \( x, y \) and \( z \) directions. All lengths and velocities are nondimensionalized by the ship length \( L \) and the free stream velocity \( U \), respectively. The
pressure $\psi$ is the static pressure $p$ minus the hydrostatic component $-zFr^{-2}$ and may be expressed as $\psi = p + zFr^{-2}$, where $Fr = \frac{U}{\sqrt{gL}}$ is the Froude number. The pressure variable $\psi$ is non-dimensionalized by $\rho U^2$. The Reynolds number $Re$ is defined by $Re = \frac{UL}{\nu}$ where $\nu$ is the kinematic viscosity of water and is constant. $\nu_t$ is the dimensionless turbulent eddy viscosity, computed locally using the Baldwin-Lomax turbulence model. This set of equations shall be solved subject to the following boundary conditions.

2.2 Boundary Conditions

2.2.1 Free Surface

When the effects of surface tension and viscosity are neglected, the boundary condition on the free surface consists of two equations. The first, the dynamic condition, states that the pressure acting on the free surface is constant. The second, the kinematic condition, states that the free surface is a material surface: once a fluid particle is on the free surface, it forever remains on the surface. The dynamic and kinematic boundary conditions may be expressed as

$$p = \text{constant}$$
$$\frac{d\beta}{dt} = \nu + u\beta_x + v\beta_y$$

where $z = \beta(x, y, t)$ is the free surface location. Equation 3 only permits solutions where $\beta$ is single-valued.

2.2.2 Hull and Farfield

The remaining boundaries consist of the ship hull and the far field of the computational domain. On the ship hull, the condition is that of no-slip for the viscous formulation and flow tangency for the inviscid formulation. These conditions are stated mathematically as

$$u = v = w = 0$$

and

$$\mathbf{q} \cdot \mathbf{n} = un_x + vn_y + wn_z = 0.$$  

The upstream plane has $u = U_o$, $v = V_o$, $w = W_o$ and $\psi = 0$ ($p = -zFr^{-2}$). Similar conditions hold on the outer boundary plane which is assumed far enough away from the hull such that no disturbances are felt. A radiation condition should be imposed on the outflow domain to allow the wave disturbance to pass out of the computational domain. Although fairly sophisticated formulations may be devised to represent the radiation condition, simple extrapolations proved to be sufficient in this work.

2.3 Turbulence Model

To model turbulence in the flow field the laminar viscosity is replaced by

$$\mu = \mu_l + \mu_t$$

where the turbulent viscosity $\mu_t$ is computed using the algebraic model of Baldwin and Lomax [13]. The Baldwin-Lomax model is an algebraic scheme that makes use of a two-layer, isotropic turbulence formulation. In this model the turbulent viscosity is evaluated using

$$\mu_t = \left\{ \begin{array}{ll}
       (\mu_t)_{\text{inner}} y \leq y_{\text{crossover}} \\
       (\mu_t)_{\text{outer}} y > y_{\text{crossover}} 
       \end{array} \right.$$  

where $y$ is the distance measured normal to the body surface and $y_{\text{crossover}}$ is the minimum value of $y$ where both the inner and outer viscosities match. The inner viscosity follows the Prandtl-Van Driest formula,

$$(\mu_t)_{\text{inner}} = l^2|\omega|$$

where

$$l = k y \left[ 1 - \exp(-y^+/A^+) \right]$$

is the turbulent length scale for the inner region, $k$ and $A^+$ are model constants, $|\omega|$ is the vorticity magnitude and $y^+ = (\tau_w/\mu_w)y$ is the dimensionless distance to the wall in wall units.

In the outer region of the boundary layer, the turbulent viscosity is given by

$$(\mu_t)_{\text{outer}} = KC_{cp}F_{\text{wake}}F_{Klep}$$

where $K$ and $C_{cp}$ are model constants, the function $F_{\text{wake}}$ is

$$F_{\text{wake}} = \min \left( \gamma_{\text{max}} F_{\text{max}}, C_{wk}\gamma_{\text{max}} U_{\text{diff}}^2/F_{\text{max}} \right)$$

and the function $F_{Klep}$ is

$$F_{Klep} = \left[ 1 + 5.5 \left( \frac{C_{Klep}\gamma}{\gamma_{\text{max}}} \right)^6 \right]^{-1}.$$  

The quantities $F_{\text{max}}$ and $\gamma_{\text{max}}$ are determined by the value and corresponding location, respectively, of the maximum of the function

$$F = y|\omega| \left[ 1 - \exp(-y^+/A^+) \right].$$

The quantity $U_{\text{diff}}$ is the difference between maximum and minimum velocity magnitudes in the profile and is expressed as

$$U_{\text{diff}} = (u^2 + v^2 + w^2)^{1/2} - (u^2 + v^2 + w^2)^{1/2}_{\text{min}}$$
\( C_{\text{Kleb}} \) and \( C_{\text{wk}} \) are additional model constants. Numerical values for the model constants used in the computations are listed here:

\[
A^+ = 26, \quad k = 0.4, \quad K = 0.0168,
\]

and

\[
C_{cp} = 1.6, \quad C_{\text{wk}} = 1.0, \quad C_{\text{Kleb}} = 0.3.
\]

## 3 Numerical Solution

The formulation of the numerical solution procedure is based on a finite volume method (FVM) for the bulk flow variables \((u, v, w, \psi)\), coupled to a finite difference method for the free surface evolution variables \((\beta \text{ and } \psi)\). Alternative cell-centered and cell-vertex formulations may be used in finite volume schemes [14]. A cell-vertex formulation was preferred in this work because values of the flow variables are needed on the boundary to implement the free surface boundary condition. The bulk flow is solved subject to Dirichlet conditions for the free surface pressure, followed by a free surface update via the bulk flow solution (i.e. constant values for the velocities in equation 3). Each formulation is explicit and uses local time stepping. Both multigrid and residual averaging techniques may be used in the bulk flow to accelerate convergence.

### 3.1 Bulk Flow Solution

Following Chorin [15] the governing set of incompressible flow equations may be written in vector form as

\[
\mathbf{w}_t + (\mathbf{f} - \mathbf{f}_V)_x + (\mathbf{g} - \mathbf{g}_V)_y + (\mathbf{h} - \mathbf{h}_V)_z = 0 \quad (4)
\]

where the vector of dependent variables \(\mathbf{w}\) and inviscid flux vectors \(\mathbf{f}, \mathbf{g}\) and \(\mathbf{h}\) are given by

\[
\begin{align*}
\mathbf{w} &= [\psi, \ u, \ v, \ w]^T \\
\mathbf{f} &= [\Gamma^2 u, \ u^2 + \psi, \ uv, \ uw]^T \\
\mathbf{g} &= [\Gamma^2 v, \ vu, \ v^2 + \psi, \ vw]^T \\
\mathbf{h} &= [\Gamma^2 w, \ wu, \ vw, \ w^2 + \psi]^T.
\end{align*}
\]

The viscous flux vectors \(\mathbf{f}_V, \mathbf{g}_V\) and \(\mathbf{h}_V\) are given by

\[
\begin{align*}
\mathbf{f}_V &= [0, \ \tau_{xx}, \ \tau_{xy}, \ \tau_{xz}]^T \\
\mathbf{g}_V &= [0, \ \tau_{yx}, \ \tau_{yy}, \ \tau_{yz}]^T \\
\mathbf{h}_V &= [0, \ \tau_{zx}, \ \tau_{zy}, \ \tau_{zz}]^T.
\end{align*}
\]

where the viscous stress components are defined as

\[
\begin{align*}
\tau_{xx} &= (Re^{-1} + \nu_t)(2u_x - 2/3(u_x + v_y + w_z)) \\
\tau_{yy} &= (Re^{-1} + \nu_t)(2v_y - 2/3(u_x + v_y + w_z)) \\
\tau_{zz} &= (Re^{-1} + \nu_t)(2w_z - 2/3(u_x + v_y + w_z)) \\
\tau_{xy} &= (Re^{-1} + \nu_t)(u_y + v_x) \\
\tau_{yz} &= (Re^{-1} + \nu_t)(v_z + w_y) \\
\tau_{zx} &= (Re^{-1} + \nu_t)(w_z + u_x).
\end{align*}
\]

\(\Gamma\) is called the "artificial compressibility" parameter due to the analogy that may be drawn between the above equations and the equations of motion for a compressible fluid whose equation of state is given by [15]

\[
\psi = \Gamma^2 \rho.
\]

Thus, \(\rho\) is an artificial density and \(\Gamma\) may be referred to as an artificial sound speed. When the temporal derivatives tend to zero, the set of equations satisfy precisely the incompressible equations 2, with the consequence that the correct pressure may be established using the artificial compressibility formulation. The artificial compressibility parameter may be viewed as a device to create a well posed system of hyperbolic equations that are to be integrated to steady state along lines similar to the well established compressible flow FVM formulation [12]. In addition, the artificial compressibility parameter may be viewed as a relaxation parameter for the pressure iteration. Note that temporal derivatives are now denoted by \(t^*\) to indicate pseudo time; the artificial compressibility, as formulated in the present work, destroys time accuracy.

To demonstrate the effect of \(\Gamma\) on the above set of equations and to establish the hyperbolicity of the set, the convective part of equation 4 may be written in quasi-linear form to determine the eigenvalues [16]. The eigenvalues are found to be

\[
\lambda_1 = U, \quad \lambda_2 = U, \quad \lambda_3 = U + a, \quad \lambda_4 = U - a
\]

where

\[
U = uw_x + vw_y + ww_z
\]

and

\[
a^2 = U^2 + \Gamma^2 (\omega_x^2 + \omega_y^2 + \omega_z^2).
\]

The wave number components \(\omega_x, \omega_y\) and \(\omega_z\) are defined on \(-\infty \leq \omega_x, \omega_y, \omega_z \leq +\infty\). Since the eigenvalues are clearly real for any value of \(\omega_x, \omega_y, \omega_z\), the system of equations 4 is hyperbolic.

The choice of \(\Gamma\) is crucial in determining convergence and stability properties of the numerical scheme. Typically, the convergence rate of the scheme is dictated by the slowest system waves and the stability of the scheme by the fastest. In
the limit of large \( \Gamma \) the difference in wave speeds can be large. Although this situation would presumably lead to a more accurate solution through the “penalty effect” in the pressure equation, very small time steps would be required to ensure stability. Conversely, for small \( \Gamma \), the difference in the maximum and minimum wave speeds may be significantly reduced, but at the expense of accuracy. Thus a compromise between the two extremes is required. Following the work of Dreyer [17], the choice for \( \Gamma \) is taken to be

\[
\Gamma^2 = \gamma (u^2 + v^2 + w^2)
\]

where \( \gamma \) is a constant of order unity. In regions of high velocity and low pressure where suction occurs, \( \Gamma \) is large to improve accuracy, and in regions of lower velocity, \( \Gamma \) is correspondingly reduced.

The choice of \( \Gamma \) also influences the outflow boundary condition, or radiation condition. If it can be demonstrated that all system eigenvalues are both real and positive, then downstream or outflow boundary points may be extrapolated from the interior upstream flow. Even though an examination of the eigenvalues reveals that this can never be the case, the condition can be approached by a judicious choice of \( \Gamma \). If \( \Gamma \) is large, extrapolation fails because the flow has both downstream and upstream dependence. As \( \Gamma \) is reduced, the upstream dependence becomes more pronounced and the downstream is reduced. Eventually, the upstream dependence is sufficiently dominant to allow extrapolation. Hence, all outflow variables are updated using zero gradient extrapolation.

Following the general procedures for FVM, the governing equations may be integrated over an arbitrary volume \( \Omega \). Application of the divergence theorem on the convective and viscous flux term integrals yields

\[
\frac{\partial}{\partial t} \int_{\Omega} w d\Omega + \int_{\partial \Omega} \left( \mathbf{f} dS_x + g dS_y + h dS_z \right) - \int_{\partial \Omega} \left( \mathbf{f} \mathbf{v} dS_x + g \mathbf{v} dS_y + h \mathbf{v} dS_z \right) = 0
\]

where \( S_x, S_y \), and \( S_z \) are the projections of the area \( \partial \Omega \) in the \( x, y \), and \( z \) directions, respectively. The computational domain is divided into hexahedral cells. Application of FVM to each of the computational cells results in the following system of ordinary differential equations,

\[
\frac{d}{dt^*} (\Lambda_{ijk} w) + C_{ijk} - V_{ijk} = 0
\]

where \( C_{ijk} \) and \( V_{ijk} \) are the discretized evaluations of the convective and viscous flux surface integrals appearing in equation 5 and \( \Lambda_{ijk} \) is the volume of the computational cell. For the vertex based scheme, the computational cell is defined as the summation of the eight cells surrounding node \( i,j,k \). The control volume for the \( \Lambda_{ijk} \) is shown in figure 2. Because the convective and viscous flux evaluations are performed differently, they are discussed separately in the next two subsections.

3.1.1 Convective Flux Discretization

The convective flux \( C_{ijk}(w) \) is defined as

\[
C_{ijk}(w) = \sum_{k=1}^{n} (\mathbf{f} S_x + g S_y + h S_z)_k
\]

where the summation is over the \( n = 24 \) faces surrounding \( \Lambda_{ijk} \).

Evaluation of equation 6 is straightforward and is performed as follows. On each of the \( n \) faces which comprise the cell boundary the convective fluxes are assumed constant and evaluated by averaging the flux values from the four nodes which define the face. The projected areas are computed by taking the cross product of the two vectors joining opposite corners of each cell face in the physical coordinate system. The summation is carried out with the result that \( C_{ijk} \) equals the net flux out of the cell.

Evaluation of the flux terms in this fashion may be performed directly, without direct differentiation and without the need to handle grid singularities in a special fashion. Note that if the projected areas are identified with the grid metrics \( J_{\xi_x}, J_{\xi_y}, J_{\xi_z} \), etc. appearing in a transformation to a curvilinear coordinate system \( \xi = \xi(x,y,z), \eta = \eta(x,y,z) \) and
\( \zeta = \zeta(x, y, z) \) where \( J \) is the Jacobian of the transformation, then it can be shown that the above method of discretization is equivalent to a finite element discretization using linear basis functions and one-point quadrature as well as a finite difference discretization using second order central differences, provided the grid is Cartesian.

3.1.2 Viscous Discretization

The viscous flux \( V_{ijk}(w) \) is defined as

\[
V_{ijk}(w) = \sum_{k=1}^{n} (f_v S_x + g_v S_y + h_v S_z)_k
\]

(7)

where the summation is over the \( n \) faces surrounding \( A_{ijk} \).

The evaluation of the viscous flux is slightly different than the convective flux counterpart. The control volume for the \( A_{ijk} \) is shown in figure 3. The discretization for the viscous fluxes follows the

![Figure 3: Control volume for viscous flux discretization](image)

guidelines originally proposed in [18, 19] for the simulation of two dimensional viscous flows. The components of the stress tensor are computed at the centroids of the eight cells (small dots) surrounding node \( i, j, k \) (large dot) with the aid of Gauss' formula. The viscous fluxes are then computed, in a similar fashion, by making use of an auxiliary cell bounded by the faces lying on the planes containing the centers of the cells surrounding node \( i, j, k \) and the mid-lines of the cell faces. For example, using Gauss' formula, the \( u_x \) term in \( \tau_{xx} \) may be computed from

\[
u_x A = \int_A u_x dA = \int_S u dS_x \approx \sum_{k=1}^{6} u_k S_{x_k}
\]

where \( k = 1, 6 \) are the six faces surrounding a particular cell, \( u_k \) is an average of the velocities from the nodes that define the \( k^{th} \) face and \( S_{x_k} \) are the projected areas in the \( x \) direction corresponding to each face. Once the components of the complete stress tensor are computed at the centroids of the cells then the same method of evaluation may be used to compute the viscous fluxes at the vertex through use of equation 7. For this purpose the control volume is now constructed by assembling \( \frac{1}{8} \) fractions of each of the eight cells surrounding a particular vertex. This discretization procedure is designed to minimize the error induced by a kink in the grid.

3.1.3 Artificial Dissipation

The discretization scheme just described reduces to a second order accurate, nondissipative central difference approximation to the bulk flow equations on sufficiently smooth grids. A central difference scheme permits odd-even decoupling at adjacent nodes which may lead to oscillatory solutions. To prevent this "unphysical" phenomena from occurring, a dissipation term is added to the system of equations such that the system now becomes

\[
d \frac{d}{dt^*} (A_{ijk} w) + [C_{ijk}(w) - V_{ijk}(w) - D_{ijk}(w)] = 0.
\]

(8)

For the present problem a fourth derivative background dissipation term is added. The dissipative term is constructed in such a manner that the conservation form of the system of equations is preserved. The dissipation has the form

\[
D_{ijk}(w) = D_\xi + D_\eta + D_\zeta
\]

(9)

where

\[
D_{\xi_{ijk}} = d_{\xi_{i+1,j,k}} - d_{\xi_{i,j,k}}
\]

and

\[
d_{\xi,i,j,k} = \lambda_\xi \delta_x^2 \text{w} \theta_{i+1,j,k} - \text{w}_{i,j,k}.
\]

(10)

Similar expressions may be written for the \( \eta \) and \( \zeta \) directions with \( \delta_x^2, \delta_y^2 \) and \( \delta_z^2 \) representing second difference central operators.

In equation 10, the dissipation coefficient \( \lambda_\xi \) is a scaling factor proportional to the local wave speed, and renders equation 9 third order in truncation terms so as not to detract from the second order accuracy of the flux discretization. The actual form for the coefficient is based on an estimate of the spectral radius of the convective flux Jacobian matrices and is given in the \( \xi \) direction as

\[
\lambda_\xi = \epsilon | \hat{u} | + \Gamma (S_x^2 + S_y^2 + S_z^2)^{1/2}
\]

where \( \hat{u} \) is the contravariant velocity component

\[
\hat{u} = u\xi_x + v\xi_y + w\xi_z.
\]
Similar dissipation coefficients are used for the \( \eta \) and \( \zeta \) components in equation 9. The \( \varepsilon \) term is used to manually adjust the amount of dissipation.

To account for the high aspect ratio grids required for viscous simulations, the dissipative coefficients calculated above are rescaled following Martinelli’s previous work \[18\]. The coefficients are rescaled according to the following expression for the \( \xi \) direction:

\[
\lambda_\xi = \lambda_\xi \left( 1 + \left( \frac{\lambda_\eta}{\lambda_\xi} \right)^\omega + \left( \frac{\lambda_\zeta}{\lambda_\xi} \right)^\omega \right)
\]

where \( \omega \) is the rescaling parameter usually chosen equal to 2/3.

### 3.1.4 Time Integration

Equation 8 is integrated in time by an explicit multistage scheme. For each bulk flow time step, the grid, and thus \( \Lambda_{ijk} \), is independent of time. Hence equation 8 can be written as

\[
\frac{dw_{ijk}}{dt^*} + R_{ijk}(w) = 0
\]

where the residual is defined as

\[
R_{ijk}(w) = C_{ijk}(w) - V_{ijk}(w) - D_{ijk}(w)
\]

and the cell volume \( \Lambda_{ijk} \) is absorbed into the residual for clarity. If one analyzes a linear model problem corresponding to (11) by substituting a Fourier mode \( \hat{w} = e^{ipx} \), the resulting Fourier symbol has an imaginary part proportional to the wave speed, and a negative real part proportional to the diffusion. Thus the time stepping scheme should have a stability region which contains a substantial interval of the negative real axis, as well as an interval along the imaginary axis. To achieve this it pays to treat the convective and dissipative terms in a distinct fashion. Thus the residual is split as

\[
R_{ijk}(w) = C_{ijk}(w) + D_{ijk}(w)
\]

where \( C_{ijk}(w) \) is the convective part and \( D_{ijk}(w) = -(V_{ijk} + D_{ijk}) \) the dissipative part. Denote the time level \( n \Delta t \) by a superscript \( n \), and drop the subscript for clarity. Then the multistage time stepping scheme is formulated as

\[
\begin{align*}
   w^{(n+1,0)} & = w^n \\
   \quad \quad \quad \quad \ldots \\
   w^{(n+1,k)} & = w^n - \alpha_k \Delta t \left( C^{(k-1)} + D^{(k-1)} \right) \\
   \quad \quad \quad \quad \ldots \\
   w^{n+1} & = w^{(n+1,m)}
\end{align*}
\]

where the superscript \( k \) denotes the \( k \)-th stage, \( \alpha_m = 1 \), and

\[
\begin{align*}
   C^{(0)} & = C(w^n), \quad D^{(0)} = D(w^n) \\
   \quad \quad \quad \quad \ldots \\
   C^{(k)} & = C(w^{(n+1,k)}) \\
   D^{(k)} & = \beta_k D(w^{(n+1,k)}) + (1 - \beta_k) D^{(k-1)}
\end{align*}
\]

The coefficients \( \alpha_k \) are chosen to maximize the stability interval along the imaginary axis, and the coefficients \( \beta_k \) are chosen to increase the stability interval along the negative real axis. A five-stage scheme with three evaluations of dissipation has been found to be particularly effective. Its coefficients are

\[
\begin{align*}
   \alpha_1 & = 1/4 \quad \beta_1 = 1 \\
   \alpha_2 & = 1/6 \quad \beta_2 = 0 \\
   \alpha_3 & = 3/8 \quad \beta_3 = 0.56 \\
   \alpha_4 & = 1/2 \quad \beta_4 = 0 \\
   \alpha_5 & = 1 \quad \beta_5 = 0.44
\end{align*}
\]

The stability region for this scheme is shown in figure 4. The region of stability extends to a CFL number of 4 along the imaginary axis and about 9 along the real axis. The dark line to the immediate left of the imaginary axis represents the locus of points of the amplification factor for CFL number equal to 3 and a diffusion coefficient equal 1/32 \[19\].

![Figure 4: Stability region of the five-stage scheme with three evaluations of dissipation.](image)

The actual time step \( \Delta t \) is limited by the Courant number (CFL), which states that the fastest waves in the system may not be allowed to propagate further than the smallest grid spacing over the course of a time step. In this work, local time stepping is used such that regions of large grid spacing are permitted to have relatively larger time steps than.
regions of small grid spacing. Of course the system wave speeds vary locally and must be taken into account as well. The final local time step is thus computed as,

\[ \Delta t^*_{ijk} = \frac{(CFL) \lambda_{ijk}}{\lambda_{ijk}} \]

where \( \lambda_{ijk} \) is the sum of the spectral radii of both the convective and viscous flux Jacobian matrices in the \( \xi, \eta \) and \( \zeta \) directions. In regions of small grid spacing and/or regions of high characteristic wave speeds, the time step will be smaller than elsewhere.

### 3.1.5 Multigrid Scheme

Very rapid convergence to a steady state is achieved with the aid of a multigrid procedure. The idea behind the multigrid strategy is to accelerate evolution of the system of equations on the fine grid by introducing auxiliary calculations on a series of coarser grids. The coarser grid calculations introduce larger scales and larger time steps with the result that low-frequency error components may be efficiently and rapidly damped out. Auxiliary grids are introduced by doubling the grid spacing, and values of the flow variables are transferred to a coarser grid by the rule

\[ w_{2h}^{(0)} = T_{2h,h} w_h \]

where the subscripts denote values of the grid spacing parameter (i.e., \( h \) is the finest grid, \( 2h \), \( 4h \), ... are successively coarser grids) and \( T_{2h,h} \) is a transfer operator from a fine grid to a coarse grid. The transfer operator picks flow variable data at alternate points to define coarser grid data as well as the coarser grid itself. A forcing term is then defined as

\[ P_{2h} = \sum R_h(w_h) - R_{2h}(w_{2h}^{(0)}) \]

where \( R \) is the residual of the difference scheme. To update the solution on the coarse grid, the multistage scheme is reformulated as

\[
\begin{align*}
    w_{2h}^{(1)} &= w_{2h}^{(0)} - \alpha_1 \Delta t^*(R_{2h}^{(0)} + P_{2h}) \\
    &\quad \cdots \\
    w_{2h}^{(q+1)} &= w_{2h}^{(0)} - \alpha_q \Delta t^*(R_{2h}^{(q)} + P_{2h}) \\
    &\quad \cdots
\end{align*}
\]

where \( R^{(q)} \) is the residual of the \( q^{th} \) stage. In the first stage, the addition of \( P_{2h} \) cancels \( R_{2h}(w_{2h}^{(0)}) \) and replaces it by \( \sum R_h(w_h) \), with the result that the evolution on the coarse grid is driven by the residual on the fine grid. The result \( w_{2h}^{(m)} \) now provides the initial data for the next grid \( w_{4h}^{(0)} \) and so on. Once the last grid has been reached, the accumulated correction must be passed back through successively finer grids. Assuming a three grid scheme, let \( w_{4h}^{(+)} \) represent the final value of \( w_{4h}^{(0)} \). Then the correction for the next finer grid will be

\[ w_{2h}^{(+)} = w_{2h}^{(m)} + I_{2h,4h}(w_{4h}^{(+) - w_{4h}^{(0)}}) \]

where \( I_{a,b} \) is an interpolation operator from the coarse grid to the next finer grid. The final result on the fine grid is obtained in the same manner:

\[ w_h^{(+)} = w_h^{(m)} + I_{h,2h}(w_{2h}^{(+)} - w_{2h}^{(0)}) \]

The process may be performed on any number of successively coarser grids. The only restriction in the present work being use of a structured grid whereby elements of the coarsest grid do not overlap the ship hull. A 5-level "W-cycle" is used in the present work for each time step on the fine grid [12].

![Multigrid W cycle for managing the grid calculation.](image)

Figure 5: Multigrid W cycle for managing the grid calculation.
3.1.6 Grid Refinement

The multigrid acceleration procedure is embedded in a grid refinement procedure to further reduce the computer time required to achieve steady state solutions on finely resolved grids. In the grid refinement procedure the flow equations are solved on coarse grids in the early stages of the simulation. The coarse grids permit large time steps, and the flow field and the wave pattern evolve quite rapidly. When the wave pattern approaches a steady state, the grid is refined by doubling the number of grid points in all directions and the flow variables and free surface location are interpolated onto the new grid. Computations then continue using the finer grid with smaller time steps. The multigrid procedure is applied at all stages of the grid refinement to accelerate the calculations on each grid in the sequence, producing a composite “full multigrid” scheme which is extremely efficient.

The grid refinement procedure is also extremely important in evaluating the consistency of the computational scheme. As the grid is continually refined a point should be reached whereby further refinement produces no change in the computed results. This method gives an indication of the resolution required to sufficiently capture the details of the particular problem being addressed; if upon further refinement of the grid global quantities such as drag or lift change, then the resolution is probably insufficient and further refinement may be necessary.

3.2 Free Surface Solution

Both a kinematic and dynamic boundary condition must be imposed at the free surface. For the fully nonlinear condition, the free surface must move with the flow (i.e., up or down corresponding to the wave height and location) and the boundary conditions applied on the distorted free surface. Equation 3 can be cast in a form more amenable to numerical computations by introducing a curvilinear coordinate system that transforms the curved free surface $\beta(x, y)$ into computational coordinates $\xi(\xi, \eta)$. This results in the following transformed kinematic condition

$$\frac{d\beta_j}{d\xi} + \frac{\beta_j}{\xi} + \frac{\beta_j}{\eta} = \omega$$

(12)

where $\omega$ and $\gamma$ are contravariant velocity components given by

$$\omega = u\xi_x + v\xi_y$$

$$\gamma = u\eta_x + v\eta_y.$$

The free surface kinematic equation may now be expressed as

$$\frac{d\beta_j}{d\xi} + Q_{ij}(\beta) = 0$$

where $Q_{ij}(\beta)$ consists of the collection of velocity and spatial gradient terms which result from the discretization of equation 12. Throughout the interior of the $(x, y)$ plane, all derivatives are computed using the second order centered difference stencil in computational coordinates $\xi$ and $\eta$. On the boundaries a second order centered stencil is used along the boundary tangent and a first order one sided difference stencil is used in the boundary normal direction.

As was necessary in the FVM formulation for the bulk flow, background dissipation must be added to prevent decoupling of the solution. The method used to compute the dissipative terms borrows from a two dimensional FVM formulation and appears as follows:

$$D_{ij} = D_{\xi} + D_{\eta}$$

where

$$D_{\xi} = d_{\xi_{i+1,j}} - d_{\xi_{i,j}}$$

and

$$d_{\xi_{i,j}} = \alpha \delta^2 \beta_{i+1,j} - \beta_{i,j}.$$ 

The expression for $\alpha$ may be written as

$$\alpha = \epsilon(|\tilde{u}_{i+1,j}| + |\tilde{u}_{i,j}|)J$$

where $J$ is the sum of the cell Jacobians and $\epsilon$ is used to manually adjust the amount of dissipation. Hence the system of equations for the free surface is expressed as

$$\frac{d\beta_j}{d\xi} + R_{ij}(\beta) = 0$$

where

$$R_{ij} = Q_{ij} - D_{ij}.$$ 

The same scheme used to integrate equation 11 is also used here. Once the free surface update is accomplished the pressure is adjusted on the free surface such that

$$\psi^{(n+1)} = \beta^{(n+1)} F_r^{-2}.$$ 

The free surface and the bulk flow solutions are coupled by first computing the bulk flow at each time step, and then using the bulk flow velocities to calculate the movement of the free surface. After the free surface is updated, its new values are used as a boundary condition for the pressure on the bulk flow for the next time step. The entire iterative
process, in which both the bulk flow and the free surface are updated at each time step, is repeated until some measure of convergence is attained; usually steady state wave profile and wave resistance coefficient.

Since the free surface is a material surface, the flow must be tangent to it in the final steady state. During the iterations, however, the flow is allowed to leak through the surface as the solution evolves towards the steady state. This leakage, in effect, drives the evolution equation. Suppose that at some stage, the vertical velocity component \( w \) is positive (cf. equation 3 or 12). Provided that the other terms are small, this will force \( \beta^{n+1} \) to be greater than \( \beta^n \). When the time step is complete, \( \psi \) is adjusted such that \( \psi^{n+1} > \psi^n \). Since the free surface has moved farther away from the original undisturbed upstream elevation and the pressure correspondingly increased, the velocity component \( w \) (or better still \( \mathbf{q} \cdot \mathbf{n} \)) where \( \mathbf{n} = \frac{\nabla F}{|\nabla F|} \) and \( F = z - \beta(x, y) \) will then be reduced. This results in a smaller \( \Delta \beta \) for the next time step. The same is true for negative vertical velocity, in which case there is mass leakage into the system rather than out. Only when steady state has been reached is the mass flux through the surface zero and tangency enforced. In fact, the residual flux leakage could be used in addition to drag components and pressure residuals as a measure of convergence to the steady state.

This method of updating the free surface works well for the Euler equations since tangency along the hull can be easily enforced. However, for the Navier-Stokes equations the no-slip boundary condition is inconsistent with the free surface boundary condition at the hull/waterline intersection. To circumvent this difficulty the computed elevation for the second row of grid points away from the hull is extrapolated to the hull. Since the minimum spacing normal to the hull is small, the error due to this should be correspondingly small, comparable with other discretization errors. The treatment of this intersection for the Navier-Stokes calculations, should be the subject of future research to find the most accurate possible procedure.

4 Results

4.1 Bare Hull Viscous and Inviscid Calculations

Figure 6 shows a portion of the fine grid used for the Euler calculations. The number of grid points is 225, 65 and 49 in the x, y and z-directions respectively, and the \( H-O \) type grid is used. Grid points are clustered near the bow and stern with a minimum spacing of 0.001 dimensionless units based on the hull length. The grid extends 1 ship length upstream from the bow, 2 ship lengths downstream from the stern, 2 ship lengths to starboard, and 2 ship lengths down below the undisturbed free surface. The minimum spacing in the y-direction, normal to the hull surface, is 0.00005 for the Navier-Stokes computations and 0.001 for the Euler computations. The resolution on the hull surface is 129 by 49 for both Euler and Navier-Stokes calculations. Note that the 225 by 65 by 49 grid topology is for the one side of the computational domain only (e.g. port or starboard). For zero angle-of-attack calculations, symmetry may be invoked on the center plane of the hull. For non-zero angle-of-attack calculations, both port and starboard sides must be discretized resulting in double the number of grid points.

Figure 6: Upstream view of bare hull discretization and one cross plane.

Three trial simulations were performed for the bare hull. These included both Euler and Navier-Stokes calculations, each run at 7, 9 and 11 knots; the Navier-Stokes calculations were all run using a Reynolds number equal to 5 million. Overhead views of the resulting wave patterns are shown in figures 7 through 9. The significant difference between the three Euler and Navier-Stokes comparisons is the flow behavior near the stern. Note that the occurrence of separation near the stern for the viscous simulations results in a dampening of the wave pattern when compared to the inviscid counterpart. This is an example of the effect of the interaction between wavemaking and the viscous boundary layer.

Figure 10 shows the computed wave elevation at the hull/waterline intersection for both the Euler and Navier-Stokes runs and the computed total
drag for the Navier-Stokes runs. Note the similarity in the wave profiles for the inviscid and viscous runs for each of the three cases. The difference near the bow is due to taking elevation measurements precisely at the hull/waterline intersection for both inviscid and viscous simulations. If the measurements for the viscous case are taken slightly away from the intersection (i.e. away from the boundary layer) then the comparison is much better. However, the stern region again displays the effect of separation. The inviscid waterline tends to rise much higher than the viscous counterpart.

Figure 11 shows the resulting wave drag vs. hull speed curve for the bare hull run at several speeds between 6 and 11 knots. The small hump at the low speed end and the sharp rise at the high speed end are consistent with what is expected due to constructive and destructive interference between the bow and stern wave structures.

Figures 12 and 13 show overhead and frontal views of the computed bow wave structure for the Navier-Stokes simulation at 11 knots. These figures show the degree to which the bow wave has approached the breaking condition. The free surface techniques used in YACHT97 do not permit the bow wave to break and spill in front of itself. Instead, the wave elevation is treated similar to a shock discontinuity in compressible flows. Figure 13 is also a good example which displays how the grid moves to conform to the geometry.

4.2 Complete IACC Yacht Configuration

Figures 14 and 15 show portions of the fine grid used for the Euler calculations. The number of grid points is 289, 97 and 49 in the x, y and z-diretions respectively, and the H – O type grid is
used. Grid points are clustered near the bow and stern with a minimum spacing of 0.001 dimensionless units based on the hull length. The grid extends 1 ship length upstream from the bow, 3 ship lengths downstream from the stern, 2 ship lengths to starboard, and 2 ship lengths down below the undisturbed free surface. The minimum spacing in the y-direction, normal to the hull surface, is 0.0001 for the Navier-Stokes computations and 0.001 for the Euler computations. The resolution is 209 by 49 on the hull surface, 97 by 33 on the keel, 97 by 33 on the bulb and 33 by 49 on the rudder for both Euler and Navier-Stokes calculations. Since the calculation for the full configuration yacht were run at a five degree angle-of-attack, both port and starboard grid are required. This results in a total number of grid points equal approximately 2.75 million.

Figure 16 is an underwater view of the hull and surface wave elevation contours. The angle-of-attack is 5 degree and the speed is 9 knots. Figure 17 shows the streamlines aft of the bulb and rudder for the same case with Reynolds number equal to 2 million.

5 Conclusions and Future Work

The ground work has been laid for a fully nonlinear free surface analysis tool for full configuration IACC yacht hulls. The numerical techniques are efficient enough to run the bare hull calculations on an IBM RISC 580 workstation with 256 Megabytes of main memory; the required CPU time is between 1 and 10 hours depending on the level of grid refinement one desires. The full configuration yacht at an angle-of-attack (and angle-of-heel as well) require significantly more resources: approaching 1 Gigabyte of main memory and 5 to 30 hours of CPU. However, with the advent of parallel architectures these numbers are expected to drop substantially in
the near future.

Work which is currently under way at Princeton includes further refinement studies for the computational grid. It is well recognized that the number of grid points required for adequate RANS simulations about full three dimensional configurations is of the order of ten million. This number is clearly beyond the current capability of workstations, and renders unpractical analyses which require numerous simulations. Our direction in this regard is to determine the optimal distribution of grid points such that meaningful RANS simulations can be performed on workstations with acceptable turnaround time. A companion effort to the grid study is the development of a multiple-block algorithm which results in greatly improved grid quality through enhanced control of point placement. A validated multiple-block algorithm for the free-surface problem will also facilitate exploitation of recently developed and affordable parallel architectures.

Another direction of research being pursued at Princeton is an investigation of new free surface routines. It has long been apparent that the weakest link in our overall approach is the coupling of the free surface routine to the bulk flow solver. The bulk flow solver is extremely robust, fast and accurate, and is capable of producing fixed-grid zero-Froude number solutions in roughly ten percent of the time required for moving grid solutions. We are confident that further investigation into the free surface routine and its coupling to the bulk flow solver will result in a substantial decrease in the required CPU time.

References


