$9^{th}$ International Conference on Numerical Ship Hydrodynamics
Ann Arbor, Michigan, August 5-8, 2007

# AN ADJOINT METHOD FOR DESIGN OPTIMIZATION OF SHIP HULLS

LUIGI MARTINELLI[1] AND ANTONY JAMESON[2]
([1]PRINCETON UNIVERSITY, [2]STANFORD UNIVERSITY)

## Abstract

We present a design optimization method for reducing the wave resistance (drag) of surface piercing hulls. Since this component of the drag is directly related to the energy contained in the wave structure that extends behind the ship, a fully coupled calculation of the flow field and wave evolution equation is required. The method combines the authors' efficient automatic shape optimization technique based on control theory, and non-linear free surface solver. In this approach an evaluation of the gradient of a cost function is computed by solving an adjoint system of partial differential equations, which enables to account for an arbitrary number of design variables (degrees of freedom).

## Introduction

In hydrodynamics, as in aerodynamics, the design space of shape optimization problems is essentially infinitely dimensional, and the surface of the solid body can be thought as a mean to control the flow. It is therefore natural to approach shape optimization problems by borrowing from control theory of systems constrained by partial differential equations. This approach pioneered by Pironneau [15] and Jameson [9] in the late eighties has, by now, become a powerful tool for aerodynamic optimization of transonic wings [16, 12]. In the late nineties this approach has been extended to incompressible flow by Cowles and Martinelli [14], and, more recently, it has successfully been applied to shape optimization of marine propulsors by Dreyer and Martinelli [5]. The present work develops our method further, enabling shape optimization for problems in which non-linear wave dynamics are important, and the calculation of the evolution of the wave fields is essential to the determination of the performance.

In most practical application, the merits of a particular design can be generally quantified by using a cost function. Suppose that the performance of a system design can be measured by a cost function $I$ which depends on a representation of the shape $\mathcal{F}(x)$, where under an arbitrary variation of the body shape, $\delta\mathcal{F}(x)$, the variation of the cost is $\delta I$. To first order $\delta I$ can be expressed as

$$\delta I = \int \mathcal{G}(x)\delta\mathcal{F}(x)dx$$

where $\mathcal{G}(x)$ is the gradient. Then by setting

$$\delta\mathcal{F}(x) = -\lambda\mathcal{G}(x)$$

one obtains an improvement

$$\delta I = -\lambda \int \mathcal{G}^2(x)dx$$

unless $\mathcal{G}(x) = 0$. Thus the vanishing of the gradient is a necessary condition for a local minimum.

Computing the gradient of a cost function for a complex system can be a numerically intensive task, especially if the number of design parameters is large and the cost function is an expensive evaluation. One could discretize the body using some appropriate basis functions, and attempt to compute the gradient using finite differences, in which case the number of flow calculations needed to estimate the gradient is proportional to the number of design variables. Similarly, if one resorts to direct code differentiation (ADIFOR) [1], or complex-variable perturbations, the cost of determining the gradient would also be directly proportional to the number of variables used to define the design. Thus, even small problems of hydrodynamic shape optimization based on these approaches can require computational resources that are measured in CPU-Years, which can only be completed in reasonable elapsed time through utilization

of massively-parallel computers costing millions of dollars.

The most cost effective technique is to compute the gradient through the solution of an adjoint problem, such as that developed by the authors [10]. The essential idea may be summarized as follows. For flow about an arbitrary body, the hydrodynamic properties that define the cost function are functions of the flow-field variables ($w$) and the physical shape of the body, which may be represented by the function $\mathcal{F}$. Then

$$I = I(w, \mathcal{F})$$

and a change in $\mathcal{F}$ results in a change of the cost function

$$\delta I = \frac{\partial I^T}{\partial w}\delta w + \frac{\partial I^T}{\partial \mathcal{F}}\delta \mathcal{F}.$$

Using a technique drawn from control theory [11], the governing equations of the flowfield are introduced as a constraint in such a way that the final expression for the gradient does not require reevaluation of the flowfield, but only the solution of an adjoint system, which has the same mathematical complexity of the model used for the flow. In the case that the equation governing the flow is a partial differential equation (or a system), the adjoint equation is also a partial differential equation (or system). The cost of solving the adjoint equation is comparable to that of solving the flow equation. Hence, the cost of obtaining the gradient is comparable to the cost of two flow evaluations, regardless of the dimension of the design space.

Practical implementation of the control based design method relies heavily upon fast and accurate solvers for both the state ($w$) and co-state ($\psi$) systems. Starting from the early nineties, in collaboration with Farmer [6] and, later with Cowles [7], we have developed fast and robust multigrid methods capable of predicting the the flow around surface piercing hulls, including non-linear wave dynamics. These methods and their implementation in software have been extensively validated in comparison with available experimental data, and provide accurate predictions of both wave, and viscous drag. They also provided the foundation for the development of the adjoint solver carried out in the present work.

In the next sections we present the mathematical formulation of the shape optimization problem and we summarize the numerical method used for solving both the flow and the adjoint equations. Computed results of shape optimization of a parabolic wigley hull are also presented to illustrate the current state of the development, leading to a discussion of future research directions.

## Formulation of the Optimal Design Problem

A general cost function can be written as:

$$I = \int_{\mathcal{B}} \mathcal{M}\left(w, S\right) d\mathcal{B} + \int_{\mathcal{D}} \mathcal{P}\left(w, S\right) d\mathcal{D},$$

containing both boundary and field contributions where $d\mathcal{B}$ and $d\mathcal{D}$ are the surface and volume elements in the computational domain. In general, $\mathcal{M}$ and $\mathcal{P}$ will depend on both the flow variables $w$ and the metrics $S$ defining the computational space. In the case of a multi-point design the flow variables may be separately calculated for several different conditions of interest. For this work a cost function based on minimization of wave drag is used, therefore we will drop the $\int_{\mathcal{D}} \mathcal{P}\left(w, S\right) d\mathcal{D}$ term from here on.

The design problem is now treated as a control problem where the boundary shape represents the control function, which is chosen to minimize $I$ subject to the constraints defined by the flow equations. A shape change produces a variation in the flow solution $\delta w$ and the metrics $\delta S$ which in turn produce a variation in the cost function

$$\delta I = \int_{\mathcal{B}} \delta \mathcal{M}(w, S)\, d\mathcal{B} \tag{1}$$

with

$$\delta \mathcal{M} \quad = \quad \left[\mathcal{M}_w\right]_I \delta w + \delta \mathcal{M}_{II}, \tag{2}$$

where we use the subscripts $I$ and $II$ to distinguish between the contributions associated with the variation of the flow solution $\delta w$ and those associated with the metric variations $\delta S$. Thus $\left[\mathcal{M}_w\right]_I$ represents $\frac{\partial \mathcal{M}}{\partial w}$ with the metrics fixed, while $\delta \mathcal{M}_{II}$ represents the contribution of the metric variations $\delta S$ to $\delta \mathcal{M}$.

Assuming that the flow is governed by a system of conservation laws, in the steady state, the constraint equation specifies the variation of the state vector $\delta w$ by

$$\frac{\partial}{\partial x_i}\delta\left(f_i\right) = 0. \tag{3}$$

Here $\delta f_i$, the variation of the flux in the $i$-th coordinate direction, can also be split into contributions associated with $\delta w$ and $\delta S$ using the notation

$$\delta f_i \quad = \quad \left[f_{i_w}\right]_I \delta w + \delta f_{i_{II}} \tag{4}$$

Multiplying by a co-state vector $\psi$, which will play an analogous role to the Lagrange multiplier, and integrating over the domain produces

$$\int_{\mathcal{D}} \psi^T \frac{\partial}{\partial x_i}\delta\left(f_i\right) d\mathcal{D} = 0.$$

If $\psi$ is differentiable this may be integrated by parts to give

$$\int_{\mathcal{B}} n_i \psi^T \delta\left(f_i\right) d\mathcal{B} - \int_{\mathcal{D}} \frac{\partial \psi^T}{\partial x_i} \delta\left(f_i\right) d\mathcal{D} = 0.$$

Since the left hand expression equals zero, it may be subtracted from the variation in the cost function (1) to give

$$\begin{aligned} \delta I &= \int_{\mathcal{B}} \left[\delta\mathcal{M} - n_i \psi^T \delta\left(f_i\right)\right] d\mathcal{B} \\ &+ \int_{\mathcal{D}} \left[\frac{\partial \psi^T}{\partial x_i} \delta\left(f_i\right)\right] d\mathcal{D}. \end{aligned} \tag{5}$$

Now, since $\psi$ is an arbitrary differentiable function, it may be chosen in such a way that $\delta I$ no longer depends explicitly on the variation of the state vector $\delta w$.

Comparing equations (2) and (4), the variation $\delta w$ may be eliminated from (5) by equating all field terms with subscript "$I$" to produce a differential adjoint system governing $\psi$

$$\frac{\partial \psi^T}{\partial x_i} \delta\left[f_{iw}\right]_I = 0 \quad \text{in } \mathcal{D}. \tag{6}$$

The corresponding adjoint boundary condition is produced by equating the subscript "$I$" boundary terms in equation (5) to produce

$$n_i \psi^T \delta\left[f_{iw}\right]_I = \delta\mathcal{M}_w \quad \text{on } \mathcal{B}. \tag{7}$$

The remaining terms from equation (5) then yield a simplified expression for the variation of the cost function which defines the gradient $\mathcal{G}$. Using the relationship between the mesh deformation and the surface modification, the field integral is reduced to a surface integral by integrating along the coordinate lines emanating from the surface. Thus the expression for $\delta I$ is finally reduced to the form

$$\delta I = \int_{\mathcal{B}} \mathcal{G} \delta\mathcal{F} \, d\mathcal{B}$$

where $\mathcal{F}$ represents the design variables, and $\mathcal{G}$ is the gradient, which is a function defined over the boundary surface.

The boundary conditions satisfied by the flow equations restrict the form of the left hand side of the adjoint boundary condition (7). Consequently, the boundary contribution to the cost function $\mathcal{M}$ cannot be specified arbitrarily. Instead, it must be chosen from the class of functions which allow cancellation of all terms containing $\delta w$ in the boundary integral of equation (5).

## Flow Model and Solver

For a viscous incompressible fluid moving under the influence of gravity, the differential form of the continuity equation and the RANS equations in a Cartesian coordinate system can be cast, using tensor notation, in the form,

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0. \tag{8}$$

$$\begin{aligned} &\rho\left(\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j}\right) = \\ &-\frac{\partial \bar{p}}{\partial x_i} + \rho g_i + \frac{\partial}{\partial x_j}\left(\mu \frac{\partial \bar{u}_i}{\partial x_j} - \rho\overline{u_i u_j}\right) \end{aligned} \tag{9}$$

Here, $\bar{u}_i$ is the mean velocity components in the $x_i$ direction, $\bar{p}$, the mean pressure, $\rho g_i$ the body forces, and $\overline{u_i u_j}$ is the Reynolds stress tensor which requires an additional model for closure. For implementation in a computer code, it is more convenient to use a dimensionless form of the equation which is obtained by dividing all lengths by a characteristic length $L$ and all velocity by the free stream velocity $U_\infty$. Also, it is convenient subtract the hydrostatic component from the pressure and re-define $\bar{p} = \bar{p} - \frac{x_3}{Fr^2}$ where we have assumed that the $x_3$ is the vertical axis, and $Fr = U_\infty/\sqrt{gL}$ is the Froude number. Note that in the limit of high Reynolds numbers, the inviscid Euler equations are recovered. Thus, a hierarchy of mathematical models can be easily implemented on a single computer code, allowing study of the controlling mechanisms of the flow. Although, the theory presented holds for the RANS system, and our flow solver is fully validated for the simulation of turbulent flow, for this work we have only considered inviscid flow models models.

**Preconditioner.** Many methods of integrating the incompressible RANS equations have been developed. The difficulty in integrating the incompressible equations stems from the constraint provided by the continuity equation which requires the velocity field to be divergence free at all times. For this work, the method of artificial compressibility, developed by Chorin [3] is utilized. In this method, the continuity equation is augmented by a time derivative of the pressure.

$$\frac{\partial \bar{p}}{\partial t^*} + \Gamma^2 \left(\frac{\partial \bar{u}_i}{\partial x_i}\right) = 0. \tag{10}$$

This equation has no physical meaning until a steady-state solution has been reached in which case the divergence free velocity field is obtained. The value of $\Gamma^2$ affects stability of the system as well as the rate of convergence. Extensive analysis by Rizzi and Eriksson [17] determined the bounds of $\Gamma$ under the constraint that the system remain hyperbolic. Since the system is hyperbolic in the transient, efficient algorithms developed for solution of compressible flow are adapted for solving the present incompressible system. The preconditioned incompressible RANS equations cast in a form amenable to the control

problem become:

$$\frac{\partial w}{\partial t^*} + I_\Gamma \left( \frac{\partial f_i}{\partial x_i} - \frac{\partial f_{vi}}{\partial x_i} \right) = 0. \quad \text{in } \mathcal{D}, \tag{11}$$

where the state vector $w$, inviscid flux vector $f$ and viscous flux vector $f_v$ are described respectively by

$$w = \begin{Bmatrix} \bar{p} \\ \bar{u}_1 \\ \bar{u}_2 \\ \bar{u}_3 \end{Bmatrix}, \; f_i = \begin{Bmatrix} \bar{u}_i \\ \bar{u}_i\bar{u}_1 + \bar{p}\delta_{i1} \\ \bar{u}_i\bar{u}_2 + \bar{p}\delta_{i2} \\ \bar{u}_i\bar{u}_3 + \bar{p}\delta_{i3} \end{Bmatrix}, \tag{12}$$

$$f_{vi} = \begin{Bmatrix} 0 \\ \sigma_{ij}\delta_{j1} \\ \sigma_{ij}\delta_{j2} \\ \sigma_{ij}\delta_{j3} \end{Bmatrix}.$$

$$I_\Gamma = \begin{bmatrix} \Gamma^2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13}$$

and $t^*$ is a *pseudotime* variable.

The stress tensor $\sigma_{ij}$ includes both laminar and Reynolds' stresses, which needs to be modelled by using a suitable closure turbulence model. For clarity we will be dropping the average symbol, $\bar{}$ , from here on.

**Boundary Conditions.** 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

$$p = constant$$

$$\frac{d\beta}{dt} = w = \beta_t + u\beta_{x_1} + v\beta_{x_2} \tag{14}$$

where $x_3 = \beta(x_1, x_2, t)$ is the free surface location. The remaining boundaries consist of the ship hull, the meridian, or symmetry plane, and the far field of the computational domain. In the viscous formulation, a no-slip condition is enforced on the ship hull. For the inviscid case, flow tangency is preserved. A radiation condition should be imposed on the outflow domain to allow the wave disturbance to leave the computational domain. Although fairly sophisticated formulations have be devised to represent the radiation condition, simple extrapolation proved to be sufficient in this work.

**Discretization.** Following the general procedures used in the finite volume formulation, the governing differential equations are integrated over an arbitrary volume. In practice, the discretization scheme 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 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 term is third order in truncation terms so as not to detract from the second order accuracy of the flux discretization.

The discretized set of equation are solved using a full approximation multigrid scheme [8] which uses a sequence of independently generated coarser meshes by eliminating alternate points in each coordinate direction. With properly optimized coefficients, multistage time stepping schemes can be very efficient drivers of the multigrid process. In this work we use a five stage scheme with three evaluation of dissipation to drive a $W$-cycle of the type illustrated in Figure 1.

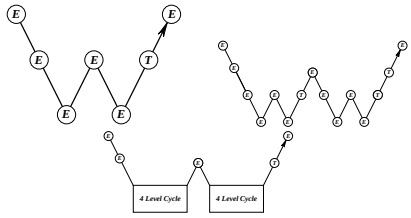



Figure 1: Multigrid $W$-cycle for managing the grid calculation. $E$, evaluate the change in the flow for one step; $T$, transfer the data without updating the solution.

In a three-dimensional case the number of cells is reduced by a factor of eight on each coarser grid and the work measured in units corresponding to a step on the fine grid is of the order of

$$1 + 2/8 + 4/64 + \cdots < 4/3.$$

In order to capture the non-linear dynamics of the surface evolution, the computational domain is made to conform to the computed location of the free surface.

**Accuracy.** The flow solver has been extensively validated against experimental data and other computed results for a wide range of problems [6, 4]: from

the standard parabolic Wigley hull to fully appended IACC configurations.

A typical prediction of total drag obtained on a bare IACC hull, in downwind sailing is presented in Figure 2,

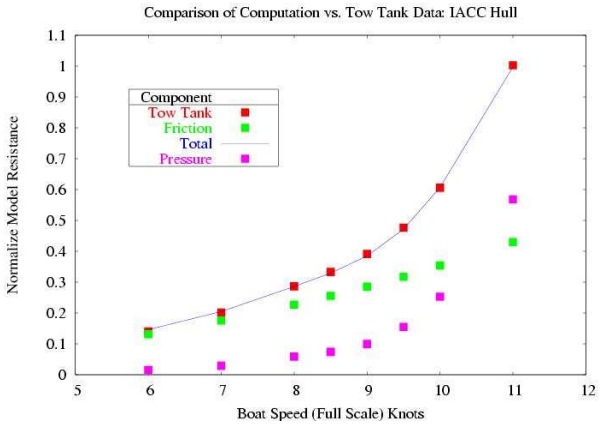



Figure 2: Computed and measured Drag on a IACC Hull as a function of speed

which shows the predicted total drag to be in excellent agreement with experimental data, for a range of speeds. Also, the computational results yield a break down of the total drag into a viscous and a wave component.

## Adjoint System

Equation ( 6) can be integrated to a steady state in pseudo time yielding,

$$\frac{\partial \psi}{\partial t^*} - I_\Gamma \left( [A_i]^T \frac{\partial \psi}{\partial x_i} - [A_{vi}]^T \frac{\partial \psi}{\partial x_i} \right) = 0. \quad \text{in } \mathcal{D},$$

where the co-state vector is given by

$$\psi = \left\{ \begin{array}{c} p^* \\ \phi_1 \\ \phi_2 \\ \phi_3 \end{array} \right\} \tag{15}$$

The adjoint equations are very similar in form to flow equations, thus the adjoint system can be solved using a preconditioner similar to that used in the method of artificial compressibility. In particular The divergence free conditions on the co-state "velocity" $\phi_i$ is augmented by a time derivative of the adjoint pressure $p^*$ [2].

$$\frac{\partial p^*}{\partial t^*} - \Gamma^2 \frac{\partial \phi_i}{\partial x_i} = 0. \tag{16}$$

The form of $\Gamma^2$ is identical to that used for the flow equations since the magnitude of the eigenvalues of the flux Jacobians for the two systems are identical. Together with equation ( 16), the adjoint system is discretized and solved in a manner that is consistent with that used for the flow equation. Since the laminar viscous operator is self-adjoint in incompressible flow, the form of the "viscous" component in the adjoint system is identical to its counterpart in the fluid equations, provided that the Reynolds stresses are modeled using an eddy viscosity, which is kept constant in the calculation of the adjoint system. Thus, the same discretization scheme used to calculate the viscous fluxes is utilized to compute the adjoint "viscous" fluxes. The boundary conditions for the adjoint system are derived from equation ( 7), once the form of the cost function has been specified. For the inviscid flow model considered in this study, when the cost function is defined as quadratic deviation from a target pressure distribution $p_d$ on the solid boundary, an expansion of this term gives:

$$n_i \phi_i = p - p_d. \tag{17}$$

Thus, the transpiration "velocity" of the adjoint variables is equal to the deficit between desired and target pressures. When a drag penalty term is added to the cost function, the boundary condition at the solid surface must be modified accordingly [11].

In the far field, the conditions on the adjoint variables can be determined from equation ( 7). Since the eigenvalues of the flux Jacobians of the two systems differ in direction but are equal in magnitude, the ingoing waves of the flow system correspond to outgoing waves of the adjoint system. For this work, it was found to be sufficient to set a simpler relation for the adjoint variables in the far field.

$$\psi_i = 0. \ \forall i \tag{18}$$

At the free-surface , the $\phi_i$ were fixed while a simple extrapolation of $p^*$ was used both at the free surface, and at all solid boundaries.

The cost of solving the adjoint equation is comparable to that of solving the flow equation. Hence, the cost of obtaining the gradient is comparable to the cost of two function evaluations, regardless of the dimension of the design space.

## Search Procedure

The remaining cost issue is related to finding a location in the design space where the gradient vanishes, and hence there is a local optimum. Normally, this search starts from a baseline design and the design space is traversed by a search method. The final state of the search may be subject to constraints imposed on the design space, yet there is no requirement that the trajectory adhere to these except at its end point. The efficiency of the search depends on the number of steps it takes to find a local minimum as well as the cost of each step.

In order to accelerate the search, one may resort to using the Newton method. Here, the search direction

is based on the equation represented by the vanishing of the gradient, $\mathcal{G}(\mathcal{F}) = 0$, and is solved by the standard Newton iteration for nonlinear equations.

Suppose the Hessian is denoted by

$$A = \frac{\partial \mathcal{G}}{\partial \mathcal{F}}$$

then the result of a step $\delta\mathcal{F}$ may be linearized as

$$\mathcal{G}(\mathcal{F} + \delta\mathcal{F}) = \mathcal{G}(\mathcal{F}) + A\delta\mathcal{F}$$

This is set to zero for a Newton step; therefore

$$\delta\mathcal{F} = -A^{-1}\mathcal{G}$$

The Newton method is generally very effective if the Hessian can be evaluated accurately and cheaply. Unfortunately, this is not the case with hydrodynamic shape optimization.

Quasi-Newton methods estimate $A$ or $A^{-1}$ from the changes of $\mathcal{G}$ recorded during successive steps. For a discrete problem with $N$ design variables, it requires $N$ steps to obtain a complete estimate of the Hessian, and these methods have the property that they can find the minimum of a quadratic form in exactly $N$ steps. Thus in general, the cost of a quasi-Newton search scales with the dimension of the design space.

Efficient hydrodynamic shapes are predominantly smooth. This suggests a natural alternative approach to the search method. In order to make sure that each new shape in the optimization sequence remains smooth, one may smooth the gradient and replace $\mathcal{G}$ by its smoothed value $\bar{\mathcal{G}}$ in the descent process. This also acts as a preconditioner which allows the use of much larger steps. This procedure can be formalized as follows.

Define a modified Sobolev inner product

$$\langle u, v\rangle = \int_{\mathcal{D}} (uv + \epsilon \nabla u \cdot \nabla v) d\mathcal{D} \ ,$$

then

$$\langle u, v\rangle = (u, v) + (\epsilon \nabla u, \nabla v)$$

where the $(u, v)$ is the standard inner product in $\mathrm{L}_2$. Integration by parts yields

$$\langle u, v\rangle = (u - \nabla\left(\epsilon \nabla u\right), v) + \int_{\mathcal{B}} \epsilon v \frac{\partial u}{\partial n} d\mathcal{B}.$$

Using the inner product notation the variation of the cost function $I$ can be expressed as

$$\delta I = (\mathcal{G}, \delta\mathcal{F}) = \langle \bar{\mathcal{G}}, \delta\mathcal{F}\rangle = \left(\bar{\mathcal{G}} - \nabla\left(\epsilon \nabla \bar{\mathcal{G}}\right), \delta\mathcal{F}\right) \ .$$

Therefore we can solve implicitly for $\bar{\mathcal{G}}$

$$\bar{\mathcal{G}} - \nabla\left(\epsilon \nabla \bar{\mathcal{G}}\right) = \mathcal{G}.$$

Then, if one sets $\delta\mathcal{F} = -\lambda\bar{\mathcal{G}}$,

$$\delta I = -\lambda \langle \bar{\mathcal{G}}, \bar{\mathcal{G}}\rangle = -\lambda \left(\mathcal{G}, \bar{\mathcal{G}}\right) \ ,$$

and an improvement is assured if $\lambda$ is sufficiently small and positive, unless the process has already reached a stationary point at which $\bar{\mathcal{G}} = 0$ (and therefore $\mathcal{G} = 0$). Also, the original smoothness of the boundary is preserved.

It turns out that this approach is extremely tolerant to the use of approximate values of the gradient, so that neither the flow solution nor the adjoint solution need be fully converged before making a shape change. This results in very large savings in the computational cost of the complete optimization process.

**Computational Costs .** In order to address the issues of the search costs, Jameson and Vassberg investigated a variety of techniques in Reference [13] using a trajectory optimization problem (the brachistochrone) as a representative model. The study verified that the search cost (i.e., number of steps) of a simple steepest descent method applied to this problem scales as $N^2$, where $N$ is the number of design variables, while the cost of quasi-Newton methods scaled linearly with $N$ as expected. On the other hand, with an appropriate amount of smoothing, the smoothed descent method converged in a fixed number of steps, independent of $N$. Considering that the evaluation of the gradient by a finite difference method requires $N + 1$ flow calculations, while the cost of its evaluation by the adjoint method is roughly that of two flow calculations, one arrives at the estimates of total computational cost given in Tables 1-2.

Table 1: Cost of Search Algorithm.

| | |
|---|---|
| Steepest Descent | $\mathcal{O}(N^2)$ steps |
| Quasi-Newton | $\mathcal{O}(N)$ steps |
| Smoothed Gradient | $\mathcal{O}(K)$ steps |
| (Note: $K$ is independent of $N$) | |

To summarize, the steps required for one design cycle are:

(1) Solve the flow equations for $u_1$, $u_2$, $u_3$, $p$.
(2) Solve the adjoint equations for $\psi$ subject to appropriate boundary conditions.
(3) Evaluate $\mathcal{G}$ .
(4) Project $\mathcal{G}$ into an allowable subspace that satisfies any geometric constraints.
(5) Update the shape based on the direction of steepest descent.
(6) Return to 1 until convergence is reached.

Thus, practical implementation of the proposed shape optimization method relies heavily upon the speed and accuracy of our solvers for both the state ($w$) and co-state ($\psi$) systems.

Table 2: Total Computational Cost of Design.

| | |
|---|---|
| Finite Difference Gradients + Steepest Descent | $\mathcal{O}(N^3)$ |
| Finite Difference Gradients + Quasi-Newton Search | $\mathcal{O}(N^2)$ |
| Adjoint Gradients + Quasi-Newton Search | $\mathcal{O}(N)$ |
| Adjoint Gradients + Sobolev Gradient Search (Note: $K$ is independent of $N$) | $\mathcal{O}(K)$ |

### Results for a Parabolic Wigley hull

A $289 \times 49 \times 49$ grid was used to discretize the domain, The semi-hull was discretized by using $193 \times 49$ grid points, thus a total of 9457 points defining the initial shape, as well as any of the re-designed forms. A partial top view of the initial mesh is shown in Figure 3

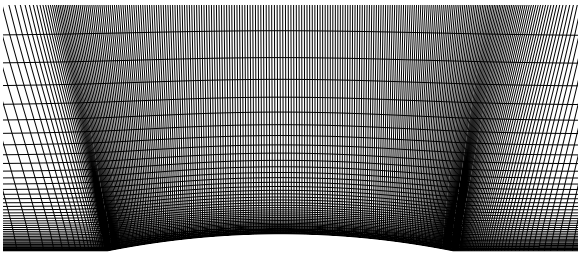


Figure 3: Mesh - Top View

In principle, every mesh point on the hull surface is a free and independent design parameter which is allowed to move in the direction of the computed local Sobolev gradient. Once the surface points are conformed to the new design, the mesh is adapted. However, in order to maintain the original hull length, the geometry modification scheme is set to preserve the original geometry near the bow and the stern.

The inviscid Euler equations were used to compute the flowfield couples with a fully-nonlinear evolution of the free surface in which the mesh is conforming, at every iteration step, with the new position of the free surface so that the dynamic condition is applied at the correct location. The incoming speed was set to correspond to a Froude number based on the hull length of .316.

For this case, hull shape optimization produced a reduction of the wave drag of approximately 30%, from an initial $C_d = .185$ to a $C_d = .126$ in 10 design cycles. A closeup comparison of the waterline in the bow region , shows the substantial decrease of the wave height produced by the redesigned hull form.

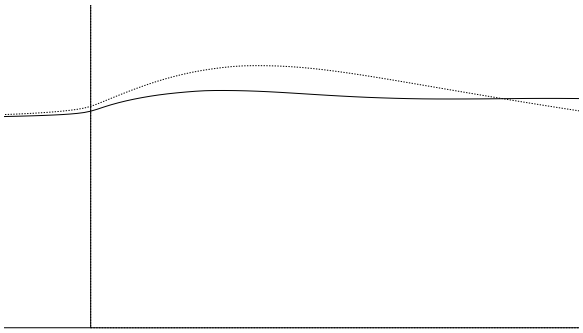
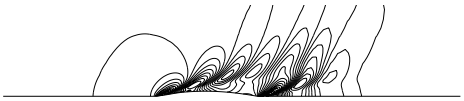


Figure 4: Waterline comparison between initial Wigley hull (dashed) , and optimized hull form (solid), at the bow

Figure  shows the top view of the computed wave elevation for the initial hull, and for several intermediate design iterations.

**(a)** ***Wave Pattern : Initial.***



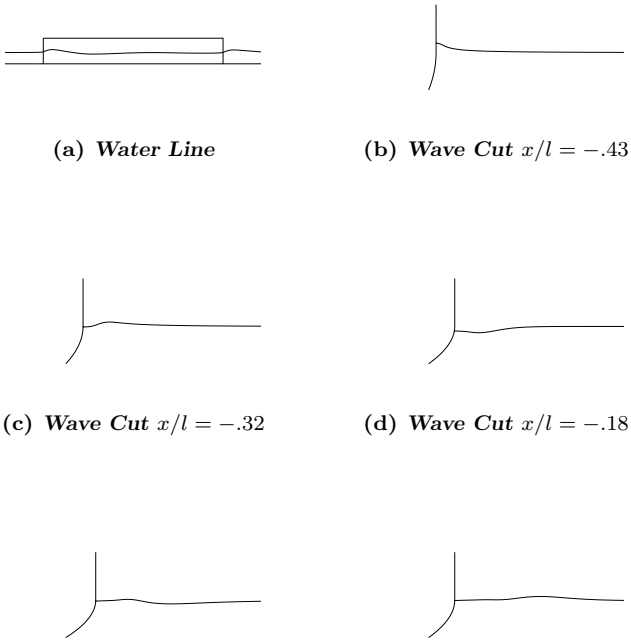
**(b)** ***Wave Pattern : Des 4***

**(c)** ***Wave Pattern : Des 10***

Figure 5: Wave Pattern on a Wigley Hull -Top View

Figures 6- 8 shows the sideview and several cross sections of the wave field, and a comparison of the

optimized hull form with the initial shape. The geometric modifications computed, tend to shift the volumes toward the stern, and form a small bulb near the bow, which is consistent with expectations.

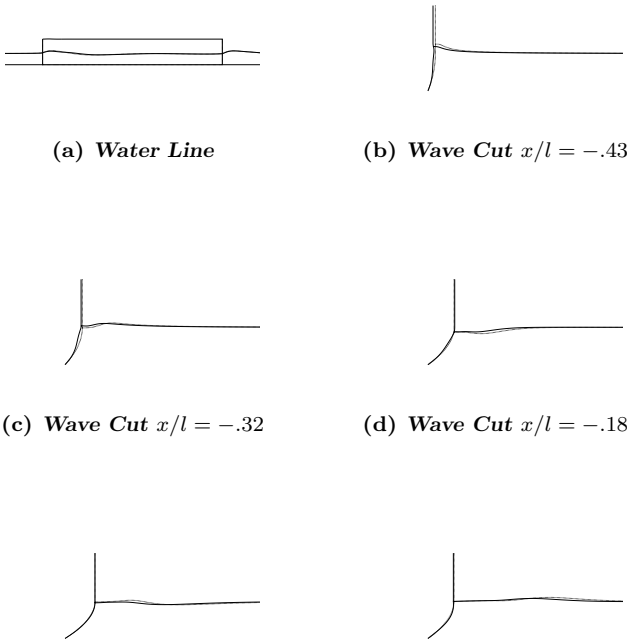


**(a) *Water Line*** **(b) *Wave Cut*** $x/l = -.43$

**(c) *Wave Cut*** $x/l = -.32$ **(d) *Wave Cut*** $x/l = -.18$

**(e) *Wave Cut*** $x/l = .00$ **(f) *Wave Cut*** $x/l = .18$

Figure 6: Initial Evolution of Wave Pattern on a Wigley Hull



**(a) *Water Line*** **(b) *Wave Cut*** $x/l = -.43$

**(c) *Wave Cut*** $x/l = -.32$ **(d) *Wave Cut*** $x/l = -.18$

**(e) *Wave Cut*** $x/l = .00$ **(f) *Wave Cut*** $x/l = .18$

Figure 7: Evolution of Wave Pattern on a Wigley Hull After 4 Design Cycles

## Conclusion and future development

We have extended our adjoint based shape optimization method to free-surface problems. During the course of this study we have tested the method by using the incompressible Euler equations with a nonlinear model for the evolution of the free surface, and performed shape optimization using our control theory based approach. Wave drag reduction has been demonstrated on realistic hull forms set upright, in conditions corresponding to downwind sailing.

Two major efforts need to be completed to make our approach useful in practical design: 1) a more flexible and capable geometry modification scheme needs to be implemented which will enable accurate constraints on the displaced volume to be imposed under dynamics sink and trim conditions. 2) implementations of the methods in our multiblock solver, which will provide boat designers with the capability of optimizing full configurations. Both efforts are currently underway, and results will be reported in future studies.

## References


[1] C. Bischof, A. Carle, G. Corliss, A. Griewank, and P. Hovland. Generating derivative codes from Fortran programs. *Internal report MCS-P263-0991*, Computer Science Division, Argonne National Lab. and Center of Research on Parallel Computation, Rice Univ., 1991.

[2] H. Cabuk and V. Modi. Optimum plane diffusers in laminar flow. *Journal of Fluid Mechanics*, 237:pp. 373–393, 1992.

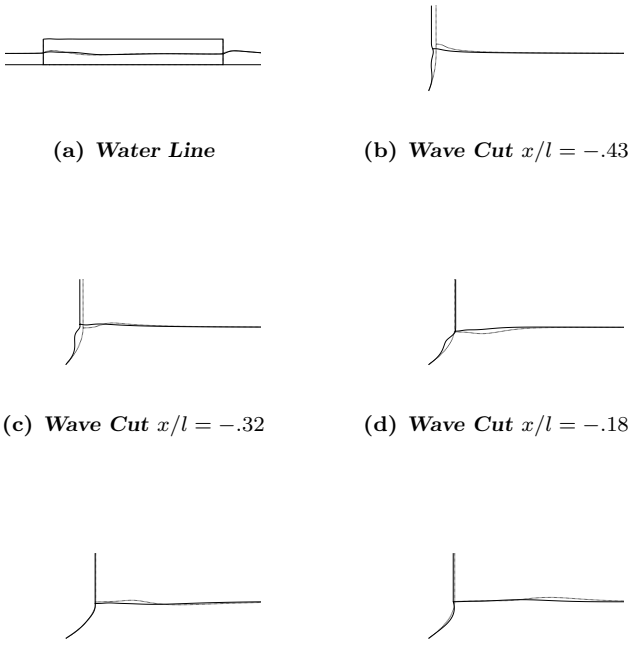
[3] A. J. Chorin. A numerical method for solving incompressible viscous flow problems. *J. Comp. Phys.*, 2:pp. 12–26, 1967.

[4] G.W. Cowles. *A Parallel Viscous Multiblock Flow Solver For Free Surface Flows Past Complex Geometries*. PhD thesis, Princeton University, Department of Mechanical and Aerospace Engineering, June 2001.

[5] J. Dreyer and L. Martinelli. Hydrodynamic shape optimization of propulsor configuration using a continuous adjoint approach. *AIAA paper 01-2580*, June 2001.

[6] J.R. Farmer. *A Finite Volume Multigrid Solution to the Three Dimensional Nonlinear Ship Wave Problem*. PhD thesis, Princeton University, Department of Mechanical and Aerospace Engineering, January 1993.

[7] J.R. Farmer, L. Martinelli, A. Jameson, and G. Cowles. Fully-nonlinear CFD techniques for ship performance analysis and design. *AIAA paper 95-1690*, AIAA 12th Fluid Dynamics Conference, San Diego, CA, June 1995.



**(a) *Water Line*** **(b) *Wave Cut*** $x/l = -.43$

**(c) *Wave Cut*** $x/l = -.32$ **(d) *Wave Cut*** $x/l = -.18$

**(e) *Wave Cut*** $x/l = .00$ **(f) *Wave Cut*** $x/l = .18$

Figure 8: Evolution of Wave Pattern on a Wigley Hull After 10 Design Cycles


[8] A. Jameson. Multigrid algorithms for compressible flow calculations. In W. Hackbusch and U. Trottenberg, editors, *Lecture Notes in Mathematics, Vol. 1228*, pages 166–201. Proceedings of the 2nd European Conference on Multigrid Methods, Cologne, 1985, Springer-Verlag, 1986.

[9] A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3:233–260, 1988.

[10] A. Jameson. Optimum aerodynamic design using cfd and control theory. 12th Computational Fluid Dynamics Conference, 1995.

[11] A. Jameson and L. Martinelli. Aerodynamic shape optimization techniques based on control theory. In V. Capasso, H. Engl, and J. Periaux, editors, *Lecture Notes in Mathematics, Vol. 1739, CIME Summer Course on Computational Mathematics Driven by Industrial Problems, Martina Franca, Italy, June 1999*. Springer, 2000.

[12] A. Jameson, L. Martinelli, J. J. Alonso, J. C. Vassberg, and J. Reuther. Simulation based aerodynamic design. *IEEE Aerospace Conference*, Big Sky, MO, March 2000.

[13] A. Jameson and J. C. Vassberg. Studies of alternative numerical optimization methods applied to the brachistochrone problem. In *Proceedings of OptiCON'99*, Newport Beach, CA, October 1999.

[14] L. Martinelli and G.W. Cowles. A control-theory based method for shape design in incompressible viscous flow using rans. *AIAA paper 00-2544*, June 2000.

[15] O. Pironneau. *Optimal Shape Design for Elliptic Systems*. Springer-Verlag, New York, 1984.

[16] J. Reuther, A. Jameson, J. Farmer, L. Martinelli, and D. Saunders. Aerodynamic shape optimization of complex aircraft configurations via an adjoint formulation. *AIAA paper 96-0094*, 34th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1996.

[17] A. Rizzi and Lars-Erik Eriksson. Computation of inviscid incompressible flow with rotation. *J. Fluid Mech.*, 153(pp. 275-312), 1985.