Shape Sensitivity of Free-Surface Interfaces Using a Level Set Methodology

Francisco Palacios*, Juan J. Alonso†, and Antony Jameson‡
Stanford University, Stanford, CA, 94305, U.S.A.

In this paper we develop the continuous adjoint methodology to compute shape sensitivities in free-surface hydrodynamic design problems using the incompressible Euler equations and the level set methodology. The identification of the free-surface requires the convection of the level set variable and, in this work, this equation is introduced in the entire shape design methodology. On the other hand, an alternative continuous adjoint formulation based in the jump condition across the interface, and an internal adjoint boundary condition is also presented. It is important to highlight that this new methodology will allow the specific design of the free-surface interface, which has a great potential in problems where the target is to reduce the wave energy (ship design), or increase the size of the wave (surfing wave pools). The complete continuous adjoint derivation, the description of the numerical methods (including a new high order numerical centered scheme), and numerical tests are detailed in this paper.

I. Introduction

In gradient-based optimization techniques, the goal is to minimize a suitable cost or objective function (drag, lift, etc.) with respect to a set of design variables (a hydrofoil profile, the surface of a ship hull, etc.). Minimization is achieved by means of an iterative process that requires the computation of the gradients or sensitivity derivatives of the cost function with respect to the design variables.

Hydrodynamic applications of optimal shape design in systems governed by partial differential equations are formulated on a fluid domain Ω, delimited by disconnected boundaries divided into an inlet, outlet, and one or more solid wall boundaries S. From now on we will restrict ourselves to the analysis of optimization problems involving functionals J defined on the solid wall S, and in the entire domain Ω, whose value depends on the flow variables U obtained from the solution of the fluid flow equations. In this context, the generic optimization problem can be succinctly stated as follows: find $S^{min} \in \mathcal{S}_{ad}$ such that

$$J(S^{min}) = \min_{S^{min} \in \mathcal{S}_{ad}} J(S)$$

where $\mathcal{S}_{ad}$ is the set of admissible boundary geometries and

$$J(S) = \int_S j_S(U, \vec{n})\, ds + \int_\Omega j_\Omega(U)\, d\Omega$$

is the objective function, where $j_S(U, \vec{n})$, and $j_\Omega(U)$ are smooth functions which depends on $\vec{n}$ (inward-pointing unit vector normal to S) and the flow variables.

Gradients of this objective function can be computed in a variety of ways, some of the most popular are the adjoint methods,[1,5,13,21] due, among other factors, to their ability of computing these derivatives at a cost comparable to solving the state PDEs (Partial Differential Equations). Adjoint methods are conventionally divided into continuous and discrete.[14,20] In the continuous approach,[4,15] the adjoint equations are derived from the governing PDE and then subsequently discretized, whereas in the discrete approach[20] the adjoint equations are directly derived from the discretized governing equations.

*Engineering Research Associate, Department of Aeronautics & Astronautics, AIAA Member.
†Associate Professor, Department of Aeronautics & Astronautics, AIAA Senior Member.
‡Thomas V. Jones Professor, Department of Aeronautics & Astronautics, AIAA Fellow.

In this work the continuous adjoint approach has been selected because it allows understanding the physical significance of the adjoint equations and boundary conditions. It also has the advantage that the adjoint system has a unique form independent of the scheme used to solve the flow-field system, and hence offers flexibility in choosing the discretization scheme for the adjoint system.

The simulation of free free-surface problem using level set techniques.[26,27] These kinds of techniques can handle nonlinear steep waves, near-breaking waves and complex geometries, however the solution of the density jump is challenging from the numerical point of view.[11] Furthermore, as the density varies with the location, the gravity forces should be added as source terms and specific techniques have been developed to perform an accurate integration of the convective terms to remove spurious velocities in the vertical direction.[22,28]

In this particular application, the artificial compressibility[6] method has been chosen to solve this problem. This method looks to apply standard hyperbolic numerical techniques to solve the incompressible Euler equations.[23] The key aspect of the technique is the introduction of a parameter that guarantee that the different wave speeds in the system are not going to be too different (which is the problem with simulating low speed problems using compressible solvers). A time accurate integration method is required to solve unsteady problems.

Optimal shape design using free-surface problems has been an important area of research,[3,9,19] and several successful methodologies have been developed to minimize pressure based functionals using high and low fidelity models. However, the introduction of the free-surface shape on the optimization process is not so common, and to the best of our knowledge this paper is one of the first applications of this methodology to the entire free-surface problem using level set technique. On the other hand, it is important to highlight that in this work a continuous adjoint formulation based on the exact jump internal conditions is shown.

With respect to the numerical method, second order numerical schemes (upwind or centered) are the most common choice for solving the incompressible equations using the artificial compressibility methodology. In this paper, a modification of the high order Jameson-Schmidt-Turkel (JST) centered scheme[16] for incompressible flows is also presented.

The organization of the paper is as follows: In Sec. II we describe the hydrodynamic model; In Sec. III we state the optimization problem; and the continuous adjoint method to compute surface is derived in Section IV; The practical implementation of the method is described in Sec. V; Some numerical experiments illustrating the relevance of the developments described in this work are presented in Sec. VI; The conclusions will be in Sec. VII

II. Description of the model

A. Free-surface model using level set and artificial compressibility equations

The incompressible Euler's equations express the conservation of mass and momentum of an incompressible fluid. In this paper we will assume that two fluids of different densities (typically water and air) are governed by these equations on a domain $\Omega \subset \mathbb{R}^3$, delimited by disconnected boundaries divided into an inlet Γ_{in}, outlet Γ_{out} and a solid wall denoted by S. In this particular two phase problem, the density is prescribed via an auxiliary variable $\phi = \phi(\vec{v}, \vec{x})$ described below. The non-dimensional artificial compressibility formulation (only valid for steady-state) can be written in the following form:

$$\begin{cases} R_U(U,\phi) \equiv \vec{\nabla} \cdot \vec{F} - S = -\partial_t U & \text{in } \Omega,\ t > 0, \\ \vec{v} \cdot \vec{n} = 0 & \text{on } S, \\ U(t_0) = U_0 & \text{in } \Omega, \\ (W)_{\vec{v}} = W_{in} & \text{on } \Gamma_{in}, \\ (W)_P = W_{out} & \text{on } \Gamma_{out}, \end{cases} \tag{1}$$

where $U = (P, \rho u, \rho v, \rho w)^\intercal$ stands for the vector of conservative variables, $\rho = \rho(\phi)$ is the fluid density, P is the pressure and $\vec{v} = (u, v, w) \in \mathbb{R}^3$ is the flow speed in a Cartesian system of reference, $\vec{n}$ is the inward-pointing unit vector normal to the surface S. The last two equations in (1) are the inlet/outlet boundary conditions (imposed velocity at the inlet and pressure at the outlet). Finally, the convective fluxes

$\vec{F} = (F_x, F_y, F_z)$, and the source term S are

$$F_x = \begin{pmatrix} \beta^2 u \\ \rho u^2 + P \\ \rho uv \\ \rho uw \end{pmatrix}, \quad F_y = \begin{pmatrix} \beta^2 v \\ \rho vu \\ \rho v^2 + P \\ \rho vw \end{pmatrix}, \quad F_z = \begin{pmatrix} \beta^2 w \\ \rho wu \\ \rho wv \\ \rho w^2 + P \end{pmatrix}, \quad S = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\frac{\rho}{Fr^2} \end{pmatrix}, \tag{2}$$

where Fr stands for the Froude number and β^2 is the artificial compressibility parameter. Note that, in free-surface problems, the density depends on $\vec{x}$ and should be inside the divergence operator.

To identify the free-surface, a level set method is used. The main idea is to use the level set function ϕ, to track the interface between the gas and the liquid.[25–27] In particular, the interface will be the zero level set of ϕ, and the level set function will be positive in the gas (density ρ_g) and negative in the liquid (density ρ_l). The value of the density is defined using an approximation of the Heaviside function $H = H(\phi, \epsilon)$:

$$H(\phi) = \begin{cases} 1 & \text{if } \phi < -\epsilon, \\ 1 - \frac{1}{2}\left[1 + \frac{\phi}{\epsilon} + \frac{1}{\pi}\sin(\pi\phi/\epsilon)\right] & \text{if } |\phi| \leq \epsilon, \\ 0 & \text{if } \phi > \epsilon, \end{cases} \tag{3}$$

where ϵ is a measure of the interface thickness. Ideally ϵ should be as small as possible, but numerical stability issues require an ϵ greater than zero (5 to 10 computational nodes in the interface is a reasonable number). Finally, the density is computed as:

$$\rho(\phi) = H(\phi) + \left(\frac{\rho_g}{\rho_l}\right)(1 - H(\phi)). \tag{4}$$

And the level set variable ϕ should satisfy the following transport equation:

$$\begin{cases} R_\phi(\vec{v}, \phi) \equiv \vec{\nabla} \cdot (\phi\, \vec{v}) = -\partial_t \phi & \text{in } \Omega,\ t > 0, \\ \phi(t) = \phi_0 & \text{on } S, \Gamma_{in}, \Gamma_{out}, \\ \phi(t_0) = \phi_0 & \text{in } \Omega, \end{cases} \tag{5}$$

which simply states that the interface moves with the fluid, and $\phi_0 = \phi_0(\vec{x})$ is the initial distance from the free-surface to the boundaries. Systems (1) and (5) constitute a complete system of equations and boundary conditions for the flow variables. From the modeling point of view, the viscosity effects and the surface tension can be added in a very straightforward way.[25] Finally, it is important to note that the level set equation is typically damped (using a function that depend on the distance to the boundaries and the wave elevation) to avoid reflections at the inlet/outlet.[24]

As a final remark, note that the water-air interface appears as a mixture zone, showing a transition from water to air, and if the transition is not smooth a jump condition should be imposed. In short, the Euler equations are satisfied throughout this mixture, and the same equations are valid everywhere in the domain.

B. Interface jump conditions

The Euler equations applied to an inviscid mixture of air and water admit discontinuities. When this occurs, jump conditions relate the flow variables on both sides of the discontinuity Σ. For a steady discontinuity located at Σ, these relations across the sharp gas/liquid interface are:

$$\left[\vec{F} \cdot \vec{n}_\Sigma\right]_\Sigma = 0. \tag{6}$$

Note that the gravity source term has been integrated in the flux function, and the jump function $[\cdot]_\Sigma$ across the interface Σ is defined by $[\phi]_\Sigma = \phi_g - \phi_l$. In this case along the discontinuity, the following holds:

$$[\rho]_\Sigma \neq 0, \qquad \left[\hat{P}\right]_\Sigma = 0, \qquad [\vec{v} \cdot \vec{n}_\Sigma]_\Sigma = 0, \qquad \left[\vec{v} \cdot \vec{t}_\Sigma\right]_\Sigma \neq 0, \tag{7}$$

where the surface tension is being neglected in this formulation, and $\hat{P}$ is computed as

$$\hat{P} = P + \frac{\rho z}{Fr^2}. \tag{8}$$

III. Optimal shape design problem using the continuous adjoint

A. Objective function definition

Optimal shape design of hydrodynamic problems is an active area of research, and some previous attempts[18,19] have been made to apply the adjoint methodology to this particular problem. In this paper we will introduce a new objective function which not only depends on pressure at the boundary S, but also on the level set function ϕ on the entire domain Ω. Thus, we will consider the following general choice of objective function:

$$J = \int_S \vec{d}\cdot(P\,\vec{n})\,ds + \int_\Omega \frac{1}{2}(\phi-\phi^d)^2\,d\Omega = \int_S j(P,\vec{n})\,ds + \int_\Omega \frac{1}{2}(\phi-\phi^d)^2\,d\Omega, \tag{9}$$

where $\vec{d}$ is a constant vector, $\vec{n}$ is the inward-pointing unit vector normal to the surface S, and ϕ^d is a target level set value.

B. Variation of the objective function

As usual in the adjoint approach, flow equations are incorporated into the cost functional as constraints by means of Lagrange multipliers $\Psi_U^\mathsf{T} = (\psi_1,\psi_2,\psi_3,\psi_4)$ for the Euler equations, and ψ_ϕ for the level set transport equation. The Lagrangian reads:

$$\mathcal{J}(S) = \int_S j(P,\vec{n})\,ds + \int_\Omega \frac{1}{2}(\phi-\phi^d)^2\,d\Omega - \int_\Omega \left(\Psi_U^\mathsf{T} R_U(U,\phi) + \psi_\phi R_\phi(\vec{v},\phi)\right)\,d\Omega, \tag{10}$$

where we have placed a special emphasis on the boundary surface dependence S. Let us consider an arbitrary (but small) perturbation of the boundary. The cost function varies due to the changes in the solution induced by the infinitesimal deformation δS of the control surface along the normal direction $\vec{n}$:

$$\begin{aligned}
\delta J(S) &= \int_{\delta S} j\,ds + \int_S \delta j\,ds + \int_S \frac{1}{2}(\phi-\phi^d)^2\delta S\,d\Omega + \int_\Omega (\phi-\phi^d)\,\delta\phi\,d\Omega \\
&= \int_S (\partial_n j - 2H_m j)\delta S\,ds + \int_S \vec{d}\cdot(\delta P\vec{n}_S + P\delta\vec{n}_S)\,ds + \int_\Omega (\phi-\phi^d)\,\delta\phi\,d\Omega \\
&= \int_S \vec{d}\cdot(\partial_n(P\vec{n}_S) - 2H_m(P\vec{n}_S))\delta S\,ds + \int_S \vec{d}\cdot\delta P\vec{n}_S\,ds - \int_S \vec{d}\cdot P\nabla_S(\delta S)\,ds + \int_\Omega (\phi-\phi^d)\,\delta\phi\,d\Omega \\
&= \int_S \vec{d}\cdot\delta P\vec{n}_S\,ds + \int_S \vec{d}\cdot(\partial_n(P\vec{n}_S) + \nabla_S P - 2H_m(P\vec{n}_S))\,\delta S\,ds + \int_\Omega (\phi-\phi^d)\,\delta\phi\,d\Omega \\
&= \int_S \vec{d}\cdot\delta P\vec{n}_S\,ds + \int_S (\vec{d}\cdot\vec{\nabla}P)\delta S\,ds + \int_\Omega (\phi-\phi^d)\,\delta\phi\,d\Omega
\end{aligned} \tag{11}$$

where we have supposed that the level set solution on the boundaries is constant ($\phi-\phi^d=0$, on S) and we have used $\delta\vec{n} = -\nabla_S(\delta S)$, which holds for small deformations, and H_m is the mean curvature of S computed as $(\kappa_1+\kappa_2)/2$, where (κ_1,κ_2) are curvatures in two orthogonal directions on the surface. Here ∇_S represents the tangential gradient operator on S. Note that we have performed an integration by parts and that we have used the gradient operator on local coordinates in S.

Assuming a regular flow solution U and a smooth boundary S, the variation of the augmented functional $\mathcal{J}(S)$ under the deformation of the geometry can be evaluated as

$$\delta\mathcal{J} = \delta J(S) - \int_\Omega \left(\Psi_U^\mathsf{T}\delta R_U(U,\phi) + \psi_\phi\delta R_\phi(\vec{v},\phi)\right)\,d\Omega, \tag{12}$$

where $\delta J(S)$ has been computed in Eq. (11), and δR_U and δR_ϕ represent the variations of R_U and R_ϕ,

respectively. Supposing a steady-state solution, the linearized system of equations are:

$$\begin{cases} \delta R_U(U,\phi) = \dfrac{\partial R_U}{\partial U}\delta U + \dfrac{\partial R_U}{\partial \phi}\delta\phi = 0 & \text{in } \Omega, \\ \delta\vec{v}\cdot\vec{n} = \vec{v}\,\nabla_S(\delta S) - (\partial_n\vec{v})\,\delta S\,\vec{n} & \text{on } S, \\ (\delta W_{in})_{\vec{v}} = 0 & \text{on } \Gamma_{in}, \\ (\delta W_{out})_P = 0 & \text{on } \Gamma_{out}. \end{cases} \tag{13}$$

$$\begin{cases} \delta R_\phi(\vec{v},\phi) = \dfrac{\partial R_\phi}{\partial U}\delta U + \dfrac{\partial R_\phi}{\partial \phi}\delta\phi = 0 & \text{in } \Omega, \\ \delta\phi = 0 & \text{on } S, \Gamma_{in}, \Gamma_{out}. \end{cases} \tag{14}$$

where $(\delta W)_{\vec{v}}$, and $(\delta W)_P$ represent the incoming characteristics on the boundaries. It is also useful to compute the variation of the density ρ with respect to the level set function:

$$\delta\rho = \begin{cases} 0 & \text{if } \phi < -\epsilon, \\ \left[\left(\frac{\rho_g}{\rho_l}\right) - 1\right] \frac{1+\cos(\pi\phi/\epsilon)}{2\epsilon}\ \delta\phi & \text{if } |\phi| \le \epsilon, \\ 0 & \text{if } \phi > \epsilon. \end{cases} \tag{15}$$

As usual the domain integrals in Eg. (12) are eliminated using integration by parts and introducing the associated adjoint operators. The integration by parts also provides some boundary terms. These boundary terms are combined with the boundary terms in Eq. (11) resulting the boundary conditions for the adjoint operators.

Next, the entire procedure is detailed. Starting with the linearized form of the incompressible Euler and level set equations, taking the inner product with the adjoint variables, supposing that the functions are smooth, and then integrating over the domain one gets:

$$\begin{aligned} 0 &= \int_\Omega \Psi_U^T \vec{\nabla}(\delta\vec{F})\,d\Omega - \int_\Omega \Psi_U^T \delta S\,d\Omega + \int_\Omega \psi_\phi \vec{\nabla}(\phi\delta\vec{v} + \vec{v}\delta\phi)\,d\Omega \\ &= \int_\Omega \Psi_U^T \vec{\nabla}(\vec{A}_U\delta U)\,d\Omega + \int_\Omega \Psi_U^T \vec{\nabla}(\vec{A}_\phi\delta\phi)\,d\Omega - \int_\Omega \Psi_U^T B_U \delta U\,d\Omega - \int_\Omega \Psi_U^T B_\phi \delta\phi\,d\Omega \\ &+ \int_\Omega \psi_\phi \vec{\nabla}(\phi\delta\vec{v})\,d\Omega + \int_\Omega \psi_\phi \vec{\nabla}(\vec{v}\delta\phi)\,d\Omega \end{aligned} \tag{16}$$

where $\vec{A}_U = \partial\vec{F}/\partial U$, $\vec{A}_\phi = \partial\vec{F}/\partial\phi$, $B_U = \partial S/\partial U = 0$, and $B_\phi = \partial S/\partial\phi$. Integrating by parts

$$\begin{aligned} 0 &= -\int_\Omega (\vec{\nabla}\Psi_U^T)\vec{A}_U\delta U\,d\Omega - \int_\Omega \vec{\nabla}\psi_\phi(\phi\delta\vec{v})\,d\Omega \\ &- \int_\Omega \vec{\nabla}\psi_\phi(\vec{v}\delta\phi)\,d\Omega - \int_\Omega (\vec{\nabla}\Psi_U^T)\vec{A}_\phi\delta\phi\,d\Omega - \int_\Omega \Psi_U^T B_\phi\delta\phi\,d\Omega \\ &+ \int_S \Psi_U^T\left(\vec{A}_U\cdot\vec{n}\right)\delta U\,ds + \int_S \psi_\phi\phi(\delta\vec{v}\cdot\vec{n})\,ds. \end{aligned} \tag{17}$$

Note that the boundary condition of the linearized level set problem has been used, and the integral over the inlet/outlet boundary can be forced to vanish with the appropriate choice of boundary conditions. The final step to obtain the adjoint equations is collecting the terms that depend on δU and $\delta\phi$. The variation of the velocity must be written in terms of the conservative variables:

$$\int_\Omega \vec{\nabla}\psi_\phi(\phi\delta\vec{v})\,d\Omega = \int_\Omega \frac{\phi}{\rho}\vec{\nabla}\psi_\phi\left(\delta(\rho\vec{v}) - \vec{v}\,\delta\rho\right)\,d\Omega = \int_\Omega \frac{\phi}{\rho}\vec{\nabla}\psi_\phi\delta(\rho\vec{v})\,d\Omega - \int_\Omega \frac{\phi}{\rho}\vec{\nabla}\psi_\phi\vec{v}\,\frac{\partial\rho}{\partial\phi}\delta\phi\,d\Omega \tag{18}$$

Combining the previous result with the variation of the objective function, the adjoint equations are:

$$\begin{cases} -\vec{A}_U^T\cdot\vec{\nabla}\Psi_U = S^* & \text{in } \Omega, \\ -\vec{v}\cdot\vec{\nabla}\psi_\phi = \left(\vec{\nabla}\Psi_U^T\frac{\partial\vec{F}}{\partial\rho} - \frac{\phi\vec{\nabla}\Psi_\phi\vec{v}}{\rho} - \frac{\Psi_4}{Fr^2}\right)\frac{\partial\rho}{\partial\phi} + (\phi - \phi^d) & \text{in } \Omega, \end{cases} \tag{19}$$

where

$$S^* = \frac{\phi}{\rho}\begin{pmatrix} 0 \\ \partial_x\psi_\phi \\ \partial_y\psi_\phi \\ \partial_z\psi_\phi \end{pmatrix}, \quad \vec{A}_U\cdot\vec{n} = \begin{pmatrix} 0 & \frac{\beta^2}{\rho}n_x & \frac{\beta^2}{\rho}n_y & \frac{\beta^2}{\rho}n_z \\ n_x & n_x u + Q & n_y u & n_z u \\ n_y & n_x v & n_y v + Q & n_z v \\ n_z & n_x w & n_y w & n_z w + Q \end{pmatrix}, \tag{20}$$

From the integral over the solid surface it is possible to compute the boundary conditions for the adjoint problem. The integral over the solid boundary S gives

$$\int_S \Psi_U^T\left(\vec{n}\cdot\vec{A}_U\right)\delta U + \psi_\phi\phi(\vec{n}\cdot\delta\vec{v})\,ds = \int_S (\vec{n}\cdot\delta\vec{v})\left(\frac{\beta^2}{\rho}\psi_1 + \vec{v}\cdot\vec{\varphi} + \psi_\phi\phi\right)ds + \int_S (\vec{n}\cdot\vec{\varphi})\,\delta P\,ds, \tag{21}$$

where we have used the boundary condition $\vec{v}\cdot\vec{n} = 0$ to evaluate the jacobian $\vec{A}_U$. To eliminate the dependence on $\vec{n}\cdot\delta\vec{v}$ we will use the linearized boundary condition on the surface S to obtain

$$\begin{aligned} \int_S \Psi_U^T\left(\vec{n}\cdot\vec{A}_U\right)\delta U + \psi_\phi(\vec{n}\cdot\delta\vec{v})\phi\,ds &= -\int_S ((\delta S\partial_n\vec{v})\cdot\vec{n} + \delta\vec{n}\cdot\vec{v})\left(\frac{\beta^2}{\rho}\psi_1 + \vec{v}\cdot\vec{\varphi} + \psi_\phi\phi\right)ds + \int_S (\vec{n}\cdot\vec{\varphi})\,\delta P\,ds \\ &= -\int_S ((\delta S\partial_n\vec{v})\cdot\vec{n} - \nabla_S(\delta S)\cdot\vec{v})\,\vartheta\,ds + \int_S (\vec{n}\cdot\vec{\varphi})\,\delta P\,ds \\ &= -\int_S ((\partial_n\vec{v}\cdot\vec{n})\,\vartheta + \nabla_S(\vec{v}\,\vartheta))\,\delta S\,ds + \int_S (\vec{n}\cdot\vec{\varphi})\,\delta P\,ds. \end{aligned} \tag{22}$$

where $\vartheta = \frac{\beta^2}{\rho}\psi_1 + \vec{v}\cdot\vec{\varphi} + \phi\,\psi_\phi$, and $\partial_n = \vec{n}\cdot\vec{\nabla}$ and $\partial_{tg} = \vec{t}\cdot\vec{\nabla}$ are the normal and tangential derivatives, respectively. Note that to obtain this last expression we have used the linearized boundary conditions, the value of $\delta\vec{n}$ at the surface, and we have also done an integration by parts.

To sum up, the following set of adjoint equations have to be solved to compute the variation of the functional:

$$\begin{cases} -\vec{A}_U^T\cdot\vec{\nabla}\Psi_U - S^* = 0 & \text{in } \Omega, \\ -\vec{v}\cdot\vec{\nabla}(\psi_\phi) - (\phi - \phi^d) = 0 & \text{in } \Omega, \\ \vec{d}\cdot\vec{n} = \vec{n}\cdot\vec{\varphi}, \quad \psi_\phi = 0 & \text{on } S, \\ (\delta W_{in})_{\vec{v}} = 0 & \text{on } \Gamma_{in}, \\ (\delta W_{out})_P = 0 & \text{on } \Gamma_{out}.. \end{cases} \tag{23}$$

Once the adjoint system is solved, the variation of the functional is computed as

$$\delta J = \int_S \left(\vec{d}\cdot\vec{\nabla}P + (\partial_n\vec{v}\cdot\vec{n}_S)\,\vartheta + \nabla_S(\vec{v}\,\vartheta)\right)\delta S\,ds, \tag{24}$$

where it is important to recall here that using this formulation it is possible to do inverse design of the interface between two fluids just by selecting the appropriate value for $\phi^d = \phi^d(\vec{x})$ (distance function from the target free-surface).

C. Alternative formulation using jump conditions

When developing an adjoint method to address free-surface optimal design problems, the main mathematical difficulty is the presence of discontinuities (e.g. air/water interfaces). In this case, the formal linearization of the state equations, which can be rigorously justified for smooth solutions, fails to be true and the adjoint system changes its nature. Indeed, when this occurs, the state of the system needs to be understood instead as a multibody one in which both the state itself at both sides of the discontinuity and its geometric location are considered as part of the state.

Thus, the sensitivity of the model needs to take into account both perturbations of the solution and of the location of the free-surface. The linearized flow equations turn out to be the classical ones on both sides of the discontinuity, but with an additional linear transport equation along the free-surface, which stems from the linearization of the jump condition. This allows us to defining the adjoint solution in a unique way.

As before, the adjoint formulation is applied to an optimization problem, and the objective is to evaluate the variation of the functional (9) under shape changes of the surface S, where the flow governing equations are the steady Euler equations.

Assuming a flow discontinuity located along a smooth curve Σ, the variation of the functional δJ is given by Eq. (11). δU stands for the infinitesimal deformation of the state to both sides of the discontinuity line and solves the linearized Euler equations, while $\delta\Sigma$ describes the infinitesimal normal deformation of the discontinuity and solves a linearization of the free-surface jump condition:

$$\begin{cases} \vec{\nabla}\cdot\left(\vec{A}_U\,\delta U\right) = 0 & \text{in } \Omega\setminus\Sigma, \\ \delta\vec{v}\cdot\vec{n} = \vec{v}\,\nabla_S(\delta S) - (\partial_n\vec{v})\,\delta S\,\vec{n} & \text{on } S, \\ (\delta W)_+ = 0 & \text{on } \Gamma, \\ \left[\vec{A}_U(\delta\Sigma\,\partial_n U + \delta U)\right]_\Sigma\cdot\vec{n}_\Sigma + \left[\vec{F}\right]_\Sigma\cdot\delta\vec{n}_\Sigma = 0 & \text{on } \Sigma, \end{cases} \tag{25}$$

with $(\delta W)_+$ representing the incoming characteristics on the inlet/outlet boundaries which correspond to physical boundary conditions in the Euler problem. $\partial\vec{F}/\partial U = \vec{A}_U$ is the Jacobian matrix. Recall that $\delta\Sigma$ is not a design parameter, but rather a dependent variable whose value is determined by the simultaneous solution of the linearized Euler equations (25) and the linearized jump conditions.

The adjoint problem is introduced through the Lagrange multipliers $(\Psi^T; L^T) = (\psi_1, \psi_2, \psi_3; l_1, l_2, l_3)$. It is possible to demonstrate that $(\Psi^T; L^T)$ satisfies the following adjoint system in 2D:[2]

$$\begin{cases} -\vec{A}_U^T\cdot\vec{\nabla}\Psi = 0, & \text{in } \Omega\setminus\Sigma, \\ \Psi^T(\vec{A}_U\cdot\vec{n}_\Gamma)_- = 0, & \text{on } \Gamma, \\ \left[\Psi^T\right]_\Sigma = 0, & \text{on } \Sigma, \\ \partial_{tg}\Psi^T\left[\vec{F}\cdot\vec{t}_\Sigma\right]_\Sigma = 0, & \text{on } \Sigma, \\ L = \Psi|_\Sigma, & \text{on } \Sigma, \end{cases} \tag{26}$$

To sum up, solving the adjoint Euler equations plus a source term based on the adjoint level set equation is equivalent to solving the adjoint equations plus an internal adjoint boundary condition.

IV. Numerical implementation of the direct, and adjoint problem

A. Numerical implementation of the direct problem

The incompressible Euler equations (1) are solved using the artificial compressibility method[7,8,10,17] first proposed by Chorin.[6] The Euler equations have been discretized using an standard edge-based finite volume formulation on the dual grid, obtained by applying the integral formulation of the equations to a dual grid control volume surrounding any given node of the grid and performing an exact integration around the outer boundary of this control volume.

In this paper a Roe upwind scheme and a central scheme with Jameson-Schmidt-Turkel (JST)-type scalar artificial dissipation (with just high order dissipation) are used for the discretization of the convective flux. Additionally, an agglomeration multigrid method is used to speed up the computation.

It is important to note that in this particular application the gravity term in the z-momentum equation is added as a source term to the sum of the fluxes in each cell, and thus the cell-face states for the Riemann solver (or centered solver) must be adapted to avoid incorrect vertical velocity at the cell face (e.g. when using first order method, the problem is reduced by subtracting the pressure gradient due to gravity from the input states for the Riemann solver).

The time integration is implicit, and a dual time stepping artificial compressibility method has been also implemented to perform non-steady simulations.

Finally, the convection (see Eq. (5)) of the level set variable ϕ is discretized using an upwind scheme, second order accuracy is easily achieved via reconstruction of variables on the cell interfaces.

The entire implementation has been math with the Stanford University Unstructured (SU^2) flow solver. SU^2 is a suite of C++ analysis tools (including an iterative PDE solver, mesh perturbation and adaptation tools, among others) linked through python-based driver scripts, specifically architected to perform analysis and design of multiphysics problems on unstructured grid topologies.

B. Numerical implementation of the adjoint problem

As in the direct problem, a standard edge-based finite volume formulation has been used on the dual grid. To preserve consistency with the direct solver, the level set adjoint variable ψ_ϕ is also discretized using a second-order upwind scheme with face reconstruction. The solution is advanced in time using the same methods as in the direct problem.

A modified version of the JST scheme without low-order dissipation is used for the discretization of the adjoint convective term. The artificial dissipation between nodes i and j is computed as:

$$\mathcal{D}_{ij} = \kappa^{(4)}\varepsilon^{(4)}\left(\nabla^2\Psi_i - \nabla^2\Psi_j\right)\omega_{ij}\lambda_{ij} \tag{27}$$

$$\varepsilon^{(4)} = \left(3\frac{N_i + N_j}{N_i N_j}\right)^2 \tag{28}$$

$$\lambda_i = |\vec{v}_i \cdot \vec{S}| + c_i|\vec{S}|, \quad \lambda_{ij} = |\vec{v}_{ij} \cdot \vec{S}| + c_{ij}|\vec{S}| \tag{29}$$

$$\omega_i = \left(\frac{\lambda_i}{4\lambda_{ij}}\right)^{\frac{1}{2}}, \quad \omega_{ij} = \frac{\omega_i\omega_j}{\omega_i + \omega_j} \tag{30}$$

where $\nabla^2\Psi_i = \sum_{j\in\mathcal{N}_i}(\Psi_j - \Psi_i)$ denotes the undivided Laplacian operator, where $\mathcal{N}_i$ represents the set of neighboring points to node i with N_i its size, $\vec{v}_{ij} = \frac{1}{2}(\vec{v}_i + \vec{v}_j)$ and $c_{ij} = \frac{1}{2}(c_i + c_j)$ are the fluid and sound speeds at the cell face, and $\kappa^{(4)}$ is an adjustable parameter.

In addition to the central scheme, an upwind scheme based upon Roe's flux difference splitting scheme has been developed for the adjoint equations. In our case, the aim is to use an upwind-type formula to evaluate a flow of the form $A_U^T \cdot \Psi$. Taking into account that $\vec{A}_U^T = -(P^T)\Lambda P^T$, where $A_U^T = -(P^T)^{-1}\Lambda P^T$, in which $A_U^T = \vec{A}_U^T \cdot \vec{n}$ is the projected Jacobian matrix, Λ is the (diagonal) matrix of eigenvalues and P is the corresponding eigenvector matrix. The upwind flux is computed as:

$$f_{ij} = \frac{1}{2}(A_{Ui}^T)(\Psi_i + \Psi_j) - (P^T)^{-1}|\Lambda|P^T(\Psi_j - \Psi_i)$$

Note that, as in the previous case, the numerical method is not conservative.

C. Modification of the Jameson-Schmidt-Turkel (JST) centered scheme

As we commented above, a central scheme with Jameson-Schmidt-Turkel (JST)-type scalar artificial dissipation is used for the discretization of the convective flux. This numerical method introduces a dissipation that is proportional to the Laplacian of the conservative variables. Focusing on the 2D mass conservation equation:

$$\frac{\partial P}{\partial t} + \beta^2\,\nabla\vec{v} = 0, \qquad \text{in } \Omega,\ t > 0, \tag{31}$$

where $\vec{v} = (u, v)$ is the velocity, P the pressure, and β^2 the artificial compressibility coefficient.

If a finite volume method is used, the fully discrete numerical scheme at node i can be written as:

$$\Omega_i\,\frac{P_i^{n+1} - P_i^n}{\Delta t} + \sum_{j\in\mathcal{N}_i}\bar{f}_{ij}^n = 0, \tag{32}$$

where an Euler's explicit time integration has been used, Ω_i is the area of the dual control-volume i, $\mathcal{N}_i$ represents the set of neighboring nodes to node i, and the numerical flux $\bar{f}_{ij}$ between nodes i and j is defined by

$$\bar{f}_{ij} = f\left(\frac{\vec{v}_i + \vec{v}_j}{2}\right)\cdot\vec{n}_{ij} - \mathcal{D}_{ij}, \tag{33}$$

where $f(\vec{v}_i) = \beta^2\vec{v}_i$ is the analytical flux at point i, $\vec{n}_{ij}$ is the normal vector to the control-volume face between i and j, and $\mathcal{D}_{ij}$ is the artificial dissipation. In this particular application, the artificial dissipation is based on the JST model, and can be expressed as:

$$\mathcal{D}_{ij} = \kappa^{(4)}\varepsilon^{(4)}\left(\nabla^2 P_i - \nabla^2 P_j\right)\omega_{ij}\lambda_{ij}, \tag{34}$$

where we are using the same notation as in the previous subsection. The key issue of this formulation is the evaluation (in unstructured grids) of $\nabla^2 P_i$ where ∇^2 denotes the undivided Laplacian operator. Several numerical methods are available for the numerical estimation of this operator:

1. Direct approach: the gradient at the control-volume face ij is taken as the average of the two computed gradients (e.g. using Green-Gauss) within each neighboring control volume i and j. This method leads to an extended stencil of neighbors of neighbors, and does not suppress odd-even decoupling modes.

$$\nabla^2 P_i \approx \sum_{j \in \mathcal{N}_i} \frac{1}{2}(\nabla P_i + \nabla P_j) \cdot \vec{n}_{ij} \tag{35}$$

where ∇P_i, and ∇P_j are numerical approximation of the true gradients at point i and j.

2. Normal derivative approach: an approximate undivided Laplacian of the pressure is constructed by integrating the normal derivative around the control volume. This approach only constitutes a consistent discretization of the undivided Laplacian for cases where the control-volume face normal is aligned with the segment joining the two nodes.

$$\nabla^2 P_i \approx \sum_{j \in \mathcal{N}_i} \frac{(P_j - P_i)}{|\vec{x}_j - \vec{x}_i|} |\vec{n}_{ij}| \tag{36}$$

where $|\vec{n}_{ij}|$ is the area of the control-volume face, and $\vec{x}_i$ are the coordinates of the node i.

3. Mixed method: a combination of both methodologies using the following expression:

$$\nabla^2 P_i \approx \sum_{j \in \mathcal{N}_i} \left(\frac{(P_j - P_i)}{|\vec{x}_j - \vec{x}_i|} \alpha_{ij} + \frac{1}{2}(\nabla P_i + \nabla P_j) \cdot (\vec{n}_{ij} - \alpha_{ij} \vec{s}_{ij}) \right) \tag{37}$$

where $\vec{s}_{ij} = (\vec{x}_j - \vec{x}_i)/|\vec{x}_j - \vec{x}_i|$ is the normalized vector connecting nodes i and j, and $\alpha_{ij} = \vec{s}_{ij} \cdot \vec{n}_{ij}$. The gradients at the nodes are computed using the Green-Gauss theorem.

4. Simplified method: Most numerical unstructured CFD codes use the following simplified formula (based on the normal derivative approach):

$$\nabla^2 P_i \approx \sum_{j \in \mathcal{N}_i} (P_j - P_i) \tag{38}$$

where it is assumed that the distance between nodes is of the order of the size of the control-volume face (reasonable approximation for isotropic grids).

All this methods are approximations of the undivided Laplacian using the value of the pressure in the neighbor nodes. However, it is important to note that when dealing with incompressible flows, the undivided Laplacian of the pressure can be evaluated in 2D as:

$$\nabla^2 P_i \approx -\left(\frac{\partial u_i}{\partial x}\frac{\partial u_i}{\partial x} + \frac{\partial v_i}{\partial y}\frac{\partial v_i}{\partial y} + \frac{\partial u_i}{\partial x}\frac{\partial v_i}{\partial y} + \frac{\partial u_i}{\partial y}\frac{\partial v_i}{\partial x} \right) \Omega_i, \tag{39}$$

where only first order velocity derivatives are needed. The gradient at the control-volume face ij is taken as the average of the two computed gradients within each neighboring control volume i and j:

$$\begin{aligned} \partial_x u_i \approx D_{x,i}(u) = \frac{1}{\Omega_i} \sum_{j \in \mathcal{N}_i} \frac{1}{2}(u_i + u_j)\, n_{x,ij}, \quad \partial_y u_i \approx D_{y,i}(u) = \frac{1}{\Omega_i} \sum_{j \in \mathcal{N}_i} \frac{1}{2}(u_i + u_j)\, n_{y,ij}, \\ \partial_x v_i \approx D_{x,i}(v) = \frac{1}{\Omega_i} \sum_{j \in \mathcal{N}_i} \frac{1}{2}(v_i + v_j)\, n_{x,ij}, \quad \partial_y v_i \approx D_{y,i}(v) = \frac{1}{\Omega_i} \sum_{j \in \mathcal{N}_i} \frac{1}{2}(v_i + v_j)\, n_{y,ij}, \end{aligned} \tag{40}$$

and finally, the undivided Laplacian (in 2D) is estimated as

$$\nabla^2 P \approx (D_{x,i}(u)D_{x,i}(u) + D_{y,i}(v)D_{y,i}(v) + D_{x,i}(u)D_{y,i}(v) + D_{x,i}(v)D_{y,i}(u))\, \Omega_i. \tag{41}$$

where this last expression will be used to compute the undivided Laplacian instead of performing a numerical approximation using the value of the pressure. Results of this new technique will be presented in the numerical result section.

V. Numerical results

A. Validation of the artificial compressibility solver (without free-surface)

The objective of this section is to validate of the artificial compressibility method implemented in this work (including the modified JST method). As test case a symmetric NACA 0012 hydrofoil has been selected (angle of attack = 0.0°, density = $998\,Kg/m^3$, free-stream pressure = $101325\,N/m^2$, free-stream velocity $1.77\,m/s$). The steady-state simulations have been performed using first and second order upwind schemes, and a multigrid strategy.

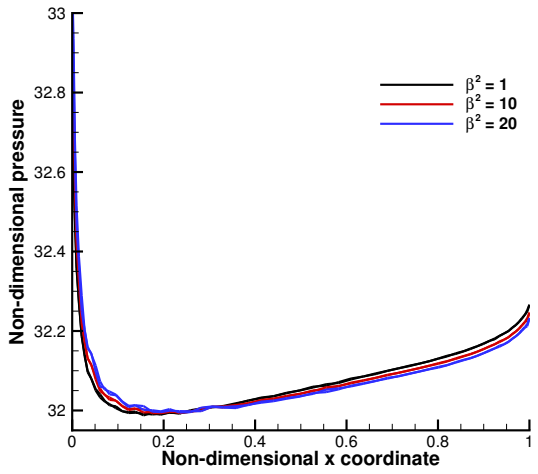

Figure 1. Pressure distribution on a NACA 0012 hydrofoil (symmetric configuration), first order upwind scheme. Note the important dependence on the artificial compressibility factor.

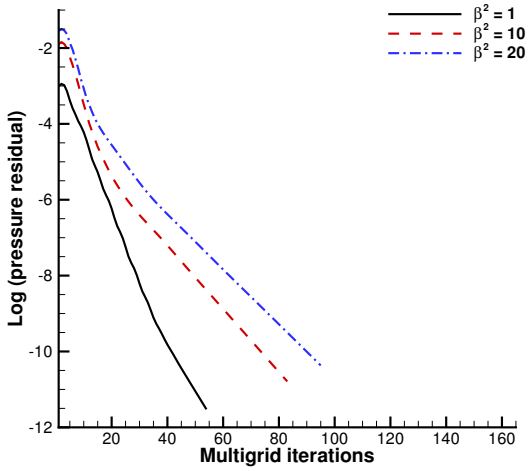

Figure 2. Convergence of the pressure residual using a first order upwind scheme. Machine precision is obtained in ≈ 50 multigrid 2V iterations.

The convergence history is presented in Fig. 2 (first order in space) and Fig. 4 (second order in space). Between 50 and 100 multigrid iterations are needed to obtain machine precision, and note that an increasing value of the artificial compresibility factor degrades the convergence. In Fig. 1 the pressure distribution on a NACA 0012 hydrofoil is shown using a first order upwind scheme, note the important dependence on value of the artificial compressibility factor. On the other hand, if a second order space integration is used (see Fig. 3), the dependence on the compressibility factor disappears.

In this paper a new centered scheme has also been developed. This new scheme is based on computing the undivided Laplacian of the pressure using the momentum equations instead of a direct discretization of the undivided Laplacian. As a test bed, the incompressible flow around a cylinder have been selected. In Fig. 5, the mach number distribution using a second order Roe scheme is compared with the traditional JST method. If the new method used (see Fig. 6) the solution symmetry is improved, but the convergence of the simulation is degraded, further numerical tests should be done to determine the advantages of the new procedure.

B. Validation of the Free-Surface simulations

A 2D super-critical and sub-critical steady-state flow over a submerged bump without wave breaking is presented as a baseline configuration. The selected 2D bump has the following shape.

$$z = \frac{2.7}{4} x(x-1)^2, \quad 0 \leq x \leq 1$$

The bump is placed on the bottom of a channel. For the cases studied here a final steady-state solution is achieved. Two free-surface cases with different Froude number (based on the bump length, $L = 1.0$) and undisturbed depth water H were selected:

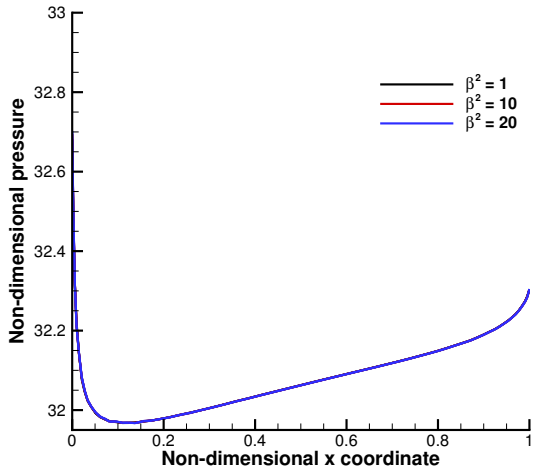

Figure 3. Pressure distribution on a NACA 0012 hydrofoil (symmetric configuration), second order upwind scheme. In this case the steady solution does not depend on the artificial compressibility factor.

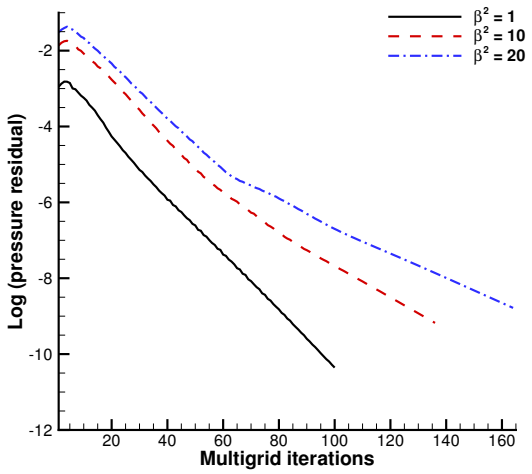

Figure 4. Convergence of the pressure residual using a second order upwind scheme. Machine precision is obtained in ≈ 100 multigrid 2V iterations.

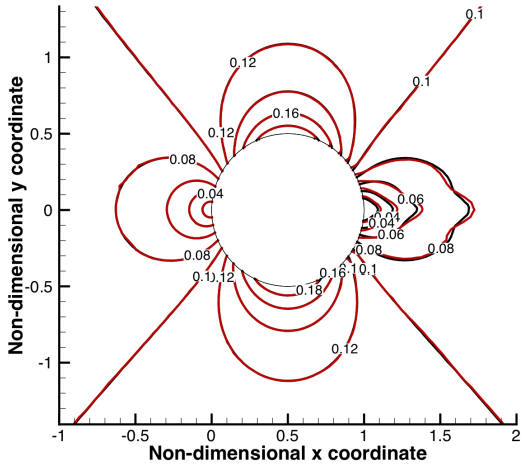

Figure 5. Mach number comparison between a second order upwind scheme (black), and the JST method (red).

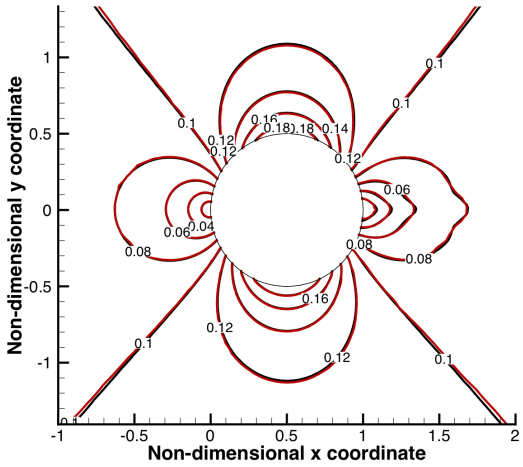

Figure 6. Mach number comparison between a second order upwind scheme (black), and a new method using the gradient of the velocity to estimate the the undivided laplacian of the pressure (red).

- Super-critical case, $Fr_L = 1.0$ and $H = 0.228$.
- Sub-critical case, $Fr_L = 0.304$ and $H = 0.500$.

The computation domain extends from $-24 < x < 25$, $-H < z < 1$, which includes extended regions at the inlet and outlet (used to reduce reflections). Appropriate absorbing boundary conditions for free-surface waves has been also implemented.[24] The air/water density ratio is $1.2\,10^{-3}$, and the air/water transition thickness was set to 0.1 (sub-critical case), and 0.2 (super-critical case).

Both super-critical and sub-critical simulations admit steady-state solutions and have been investigated by Cahouet.[12] In Fig. 7, a comparison between the simulation and the numerical super-critial experiment is presented. In Fig. 8, the sub-critical case is compared with the experiments. In both cases the wave profile shows good agreement with measurement, and we note that the under-predicted second wave crest has been reported by other numerical experiments.

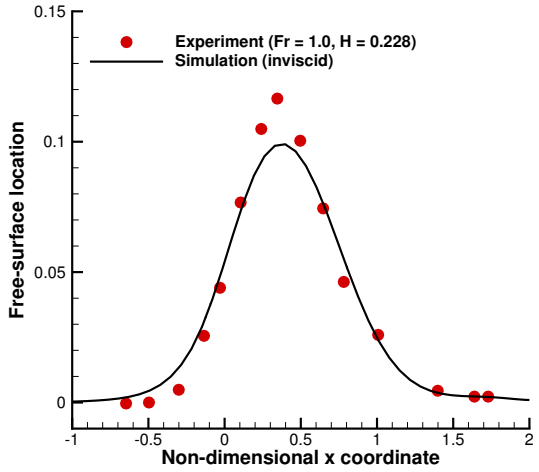

Figure 7. Comparison between experiments and inviscid simulations of this work (super-critical case, Cahouet experiment).

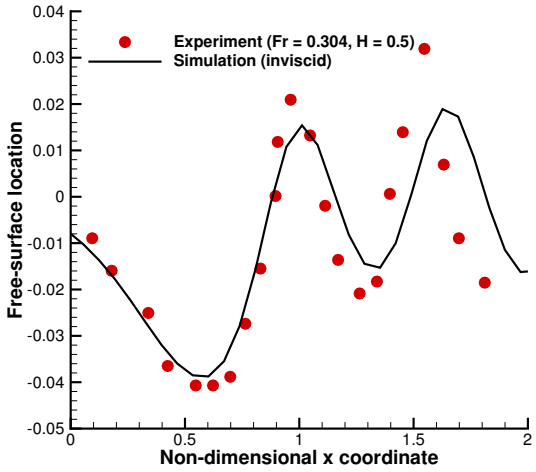

Figure 8. Comparison between experiments and inviscid simulations of this work (sub-critical case, Cahouet experiment).

In Fig. 9, and Fig. 10 it is possible to see the velocity profile for the sub-critical case (x and y components). In Fig. 11, and Fig. 12 the pressure field distribution and density field are plotted.

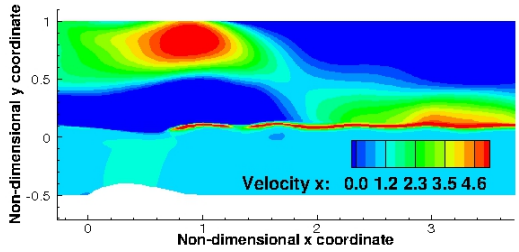

Figure 9. Velocity profile (x component), sub-critical simulation.

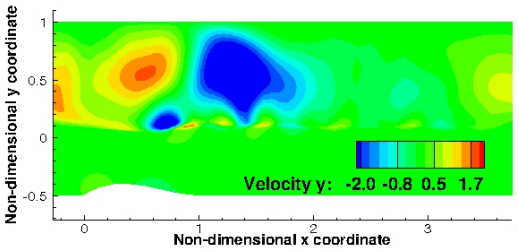

Figure 10. Velocity profile (y component), sub-critical simulation.

Once the code has been validated in 2D configurations, the simulation machinery has been tested in 3D problems. In Fig. 13, and Fig. 14 the the geometry of a 3D channel and the pressure field distribution (sub-critical case, $Fr_L = 0.304$ and $H = 0.500$) are shown.

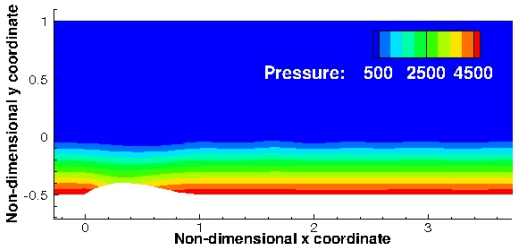

Figure 11. Pressure distribution, sub-critical simulation.

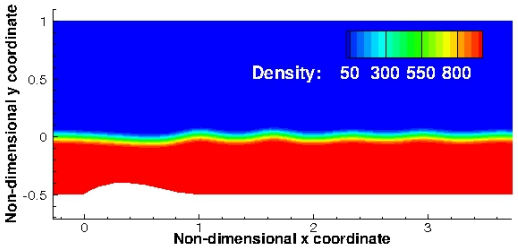

Figure 12. Density distribution, sub-critical simulation.

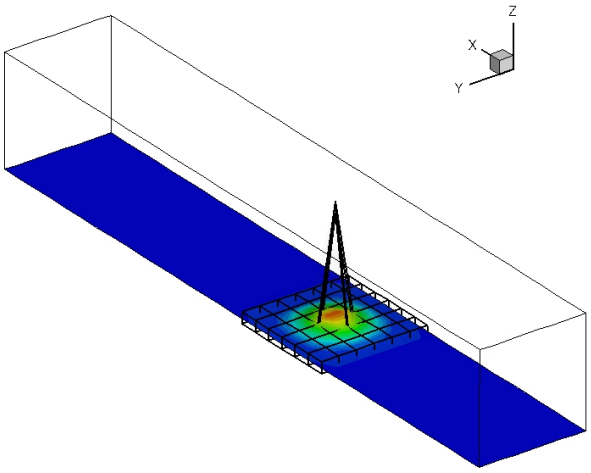

Figure 13. Channel geometry and bump defined with a Free Form Deformation box.

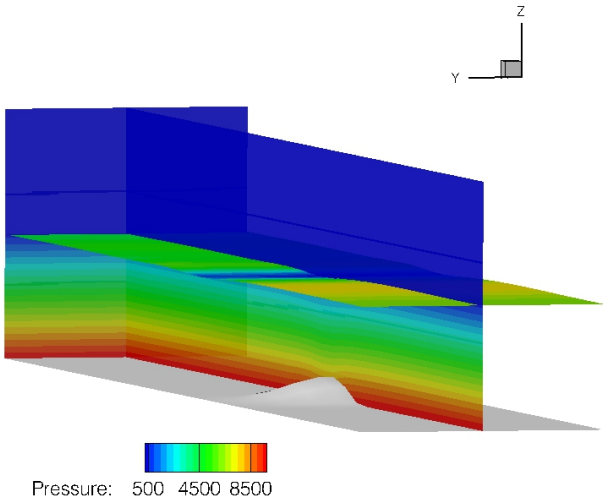

Figure 14. Pressure distribution and level set location, 3D sub-critical simulation.

C. Gradient validation

The first step in the gradient validation process is the comparison between finite differences and the continuous adjoint. As a test bed, the sub-critical case have been chosen. In the present work, the shape functions introduced by Hicks-Henne have been used as design variables. The Hicks-Henne function with maximum at point x_n is given by

$$f_n(x) = \sin^3\left(\pi x^{e_n}\right), \quad e_n = \frac{\log(0.5)}{\log(x_n)}, \quad x \in [0,1] \tag{42}$$

so the final deformation of the surface can be computed as $\Delta y = \sum_{n=1}^{N} \delta_n f_n(x)$, N being the number of bump functions and δ_n the design variable step. These functions are directly applied to the bump. Once the boundary displacements have been computed, a torsional spring method is used to reallocate the rest of vertices of the unstructured mesh.

In this particular problem, the objective function is the lift computed on the bump. The validation has been performed with and without free-surface in sub-critical problems. In Fig. 15 the continuous adjoint gradient is compared with the finite different gradient using different grid resolutions ($Fr_L = 0.304$, without free-surface), the pressure adjoint variable is shown in Fig. 16. Note that the greatest difference is located in the front of the bump, and a very good agreement is obtained at the rear of the bump. The same results are computed for the sub-critical case with an depth undisturbed water of $H = 0.500$. In Fig. 17 the sensitivity is presented, and in Fig. 18, the pressure variables are also shown. In all this cases the objective function is the pressure over the bump.

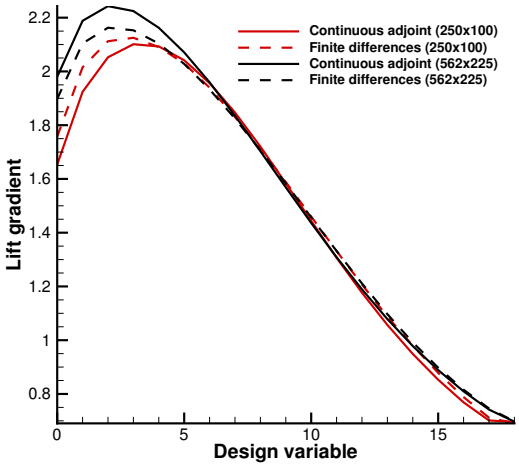

Figure 15. Comparison between continuous adjoint and finite differences, using different grid sizes (no free-surface).

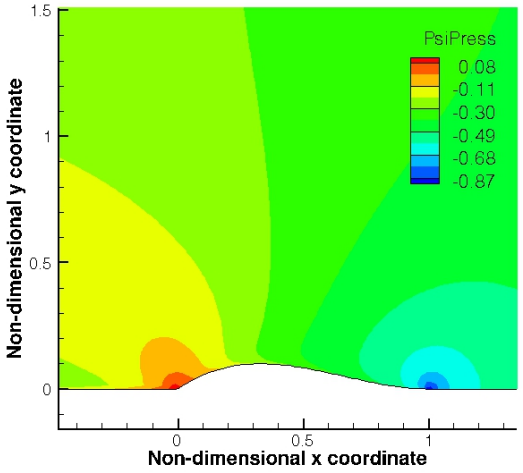

Figure 16. Adjoint variable of the pressure (no free-surface).

Apart from the classical lift functional, in this paper we have introduced the capability to do inverse design using the shape of the free-surface. In Fig. 19 the initial and the target free-surfaces are shown, in Fig. 20 the sensitivity of the functional with respect to infinitesimal variations of the surface is presented (note that the domain has been extended to design downstream of the original bump). Finally. in Fig. 21 and Fig. 22 the adjoint variables (pressure, and level set) are shown.

VI. Conclusions

In this work the continuous adjoint approach has been applied to shape design of free-surfaces. An alternative formulation based on jump conditions has been presented, and a new centered numerical method for incompressible flows using the artificial compressibility strategy has been developed.

Once the adjoint variables have been computed, this methodology only requires an integration on the

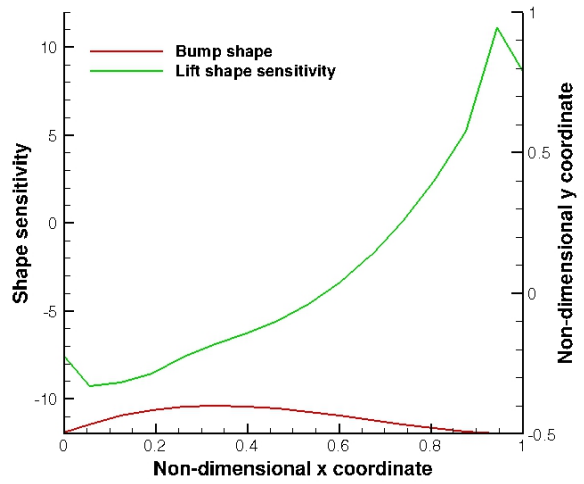

Figure 17. Lift shape sensitivity over the bump (sub-critical case, $Fr_L = 0.304$, and $H = 0.500$).

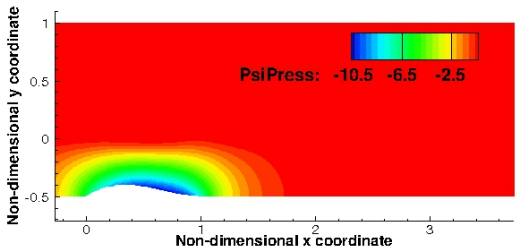

Figure 18. Adjoint of the pressure (sub-critical case, $Fr_L = 0.304$, and $H = 0.500$), lift objective function.

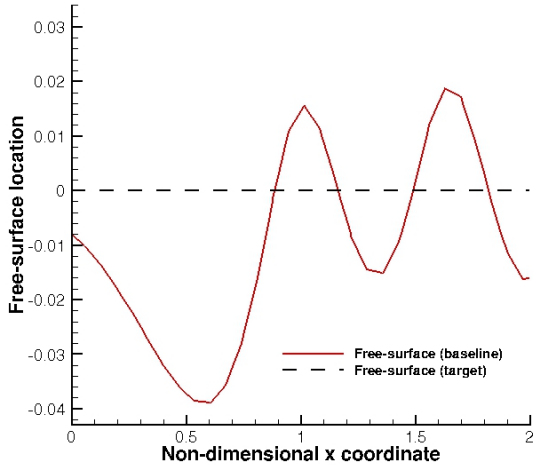

Figure 19. Initial and target free-surface location.

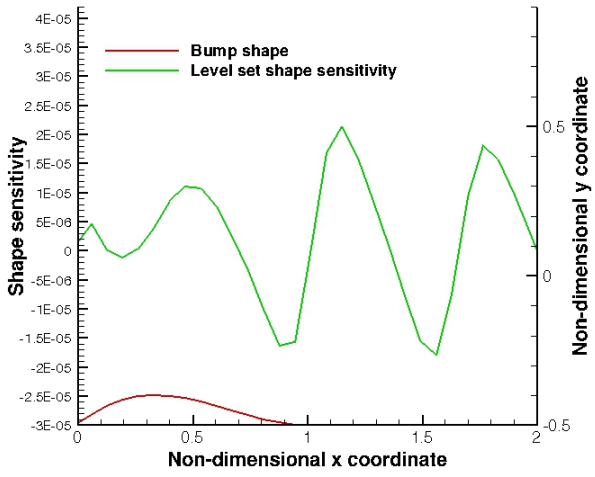

Figure 20. Inverse level set design shape sensitivity over the channel (sub-critical case, $Fr_L = 0.304$, and $H = 0.500$).

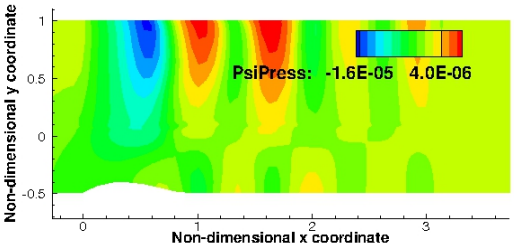

Figure 21. Adjoint of the pressure (sub-critical case, $Fr_L = 0.304$, and $H = 0.500$), level set objective function.

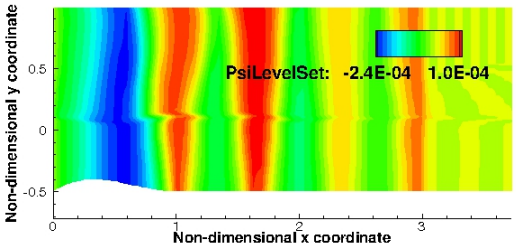

Figure 22. Adjoint of the level set variable (sub-critical case, $Fr_L = 0.304$, and $H = 0.500$), level set objective function.

surface to evaluate the gradient of the functional of interest under deformations of the body surface. This new methodology will allow the specific design of the free-surface interface, which has a great potential in problems where the target is to reduce the wave energy (ship design), or increase the size of the wave (surfing wave pools).

The code has been validated using the bump in channel problem (with sub- and super-critical examples). And the accuracy of the sensitivity derivatives that result from the application of the new method has been assessed by comparison with finite-difference computations.

The results presented here are very promising, but further numerical tests are still necessary before the complete continuous adjoint methodology can be applied to more complex problems. In particular, a full comparison with finite differences, optimization of complex configurations and convergence improvements are topics that have to be addressed in the future.

References

[1]W.K. Anderson and V. Venkatakrishnan. Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation. *AIAA Paper*, 97-0643, 1997.

[2]A. Baeza, C. Castro, F. Palacios, and E. Zuazua. 2-d euler shape design on nonregular flows using adjoint rankine-hugoniot relations. *AIAA Journal*, 47(3), 2009.

[3]E. H. Brummelen and A. Segal. Adjoint shape optimization for steady free-surface flows. *Int. J. Numer. Meth. Fluids*, 40:605614, 2002.

[4]A. Bueno-Orovio, C. Castro, F. Palacios, and E. Zuazua. Continuous adjoint approach for the spalart–allmaras model in aerodynamic optimization. *AIAA Journal*, 50(3), 2012.

[5]C. Castro, C. Lozano, F. Palacios, and E. Zuazua. A systematic continuous adjoint approach to viscous aerodynamic design on unstructured grids. *AIAA Journal*, 45(9):2125–2139, 2007.

[6]A. J. Chorin. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics*, 137:118–125, 1997.

[7]J. Farmer, L. Martinelli, and A. Jameson. Multigrid solution of the euler and navier-stokes equations for a serie 60 cb = 0.6 ship hull for froude numbers 0.160, 0.220 and 0.316. *CFD Workshop Tokyo*, 1994.

[8]J. Farmer, L. Martinelli, and A. Jameson. Yacht97: A fully viscous nonlinear free-surface analysis tool for iacc yacht design. *12th Chesapeake Sailing Yacht Symposium*, 1995.

[9]I. Y. Gejadze and G. J. M. Copeland. Adjoint sensitivity analysis for fluid flow with free surface. *Int. J. Numer. Meth. Fluids*, 47:1027–1034, 2005.

[10]T. Hino, L. Martinelli, and A. Jameson. A finite-volume method with unstructured grid for free surface flow simulations. *Sixth International conference on numerical ship hydrodynamics*, 1993.

[11]J. Huang, P. M. Carrica, and F. Stern. Coupled ghost fluid/two-phase level set method for curvilinear body-fitted grids. *Int. J. Numer. Meth. Fluids*, 55(2):867897, 2007.

[12]Cahouet J. Etude numerique et experimentale du probleme bidimensionnel de la resistance de vagues non-lineaire. Technical report, Technical report 185, Ecole Nationale Superieure de Techniques Avancees, 1984.

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

[14]A. Jameson. Aerodynamic shape optimization using the adjoint method. In *Lecture Notes Presented at the VKI Lecture Series*, Rhode Saint Genese Begium, 2 2003. Von karman Institute for fluid dynamics.

[15]A. Jameson and S. Kim. Reduction of the adjoint gradient formula for aerodynamic shape optimization problems. *AIAA Journal*, 41(11):2114–2129, 2003.

[16]A. Jameson, W. Schmidt, and E. Turkel. Numerical solution of the euler equations by finite volume methods using runge-kutta time stepping schemes. *AIAA Paper*, 81-1259, 1981.

[17]K. Kozel, P. Louda, and J. Prihoda. Numerical solution of an unsteady flow using artificial compressibility method. *Proceesings of the Czech-Japanese Seminar in Applied Mathematics*, 2006.

[18]M. Lombardi, N. Parolini, A. Quarteroni, and G. Rozza. Numerical simulation of sailing boats: dynamics, fsi, and shape optimization. Technical report, MOXReport No. 13/2011, 2011.

[19]L. Martinelli and A. Jameson. An adjoint method for design optimization of ship hulls. *9th International conference on numerical ship hydrodynamics*, 2007.

[20]S.K. Nadarajah and A. Jameson. A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization. *AIAA Paper*, 2000-0667, 2000.

[21]O. Pironneau. On optimum design in fluid mechanics. *J. Fluid Mech.*, 64:97–110, 1974.

[22]L. Qian, D.M. Causon, C.G Mingham, and D.M. Ingram. A free-surface capturing method for two fluid flows with moving bodies. *Proc. R. Soc. A*, 462:21–42, 2006.

[23]A. Rizzi and L-E. Eriksson. Computation of inviscid incompressible flow with rotation. *J . Fluid Mech.*, 153:275–312, 1985.

[24]J. E. Romate. Absorbing boundary conditions for free surface waves. *Journal of Computational Physics*, 99:135145, 1992.

[25]M. Sussman, A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, and M.L. Welcome. An adaptive level set approach for incompressible two-phase flows. *Journal of Computational Physics*, 148(1):81–124, 1999.

[26]M. Sussman, E. Fatemi, P. Smereka, and S. Osher. An improved level set method for incompressible two-phase flows. *Computers and Fluids*, 27:663–680, 1997.

[27]M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114(1):146–159, 1997.

[28]J. Wackersa and B. Korena. A surface capturing method for the efficient computation of steady water waves. *Journal of Computational and Applied Mathematics*, 215:618–625, 2008.