Implementation of a conservative cut-cell method for the simulation of two-phase cavitating flows

*Yiannis-Panayiotis G. Vrionis[1], Konstantinos D. Samouchos[1], and Kyriakos C. Giannakoglou[1]

[1]Parallel CFD & Optimization Unit, National Technical University of Athens, Greece.
*Presenting and Corresponding author: vrionis.yiannis@gmail.com

Abstract

A cut-cell method utilizing 2D cartesian meshes with embedded boundaries is employed to simulate steady-state, turbulent and cavitating flows over isolated hydrofoils. The multi-phase Reynolds-Averaged Navier-Stokes equations for a homogeneous mixture are extended with an additional transport equation accounting for the liquid volume fraction and the Kunz cavitation model. The mean-flow equations are appropriately preconditioned to render the system hyperbolic in space and time and to alleviate numerical stiffness due to the low speed of sound of the mixture. The standard $k - \varepsilon$ turbulence model is implemented. A one-layer submesh is generated to solve Spalding's composite wall function on the embedded boundaries. Mesh generation employs a hierarchical quadtree-based data structure resulting in a fast and memory-efficient process. Cut-cells are constructed by intersecting the discretized geometry with the Cartesian mesh, creating thus faces where boundary conditions are imposed. The capabilities of the cut-cell solver are demonstrated over two hydrofoils featuring mid-chord and leading-edge cavitation. Results show satisfactory agreement with numerical and experimental data.

Keywords: Cut-cell method; Cavitation; CFD; Multi-phase flow.

Introduction

Immersed Boundary Methods (IBMs) enjoy a high level of mesh generation automation and flexibility when simulating complex flows with moving or stationary geometries. Initially developed by [1], IBMs focused on making flow solvers to accurately predict inviscid [2], viscous [3] and, more recently, turbulent [4][5] flows. The mesh generation process, employing a hierarchical tree-based data structure, demonstrates high efficiency in terms of CPU cost and storage requirements [6].

IBMs can be classified as continuous and discrete, [7]. In the continuous IBMs, source terms are introduced into the flow equations to simulate the effect of solid boundaries. Since the source terms are not imposed directly on the geometry shape, the effect of the source terms is smeared across multiple neighbouring cells. Difficulties arise in the conservation of mass, momentum and energy in cells intersecting the solid boundaries, [8]. In contrast, in discrete IBMs, solid boundaries are represented by sharp interfaces and boundary conditions are imposed, without altering the governing PDEs. This representation, however, introduces temporal discontinuity, for the cells without time history, and additional actions are required to simulate moving geometries. Among the discrete IBMs, the cut-cell method reshapes finite volumes using their intersections with the geometry. Very small finite volumes may appear near the interface during reshaping that can cause numerical instabilities [9], if not properly treated. However, the

discrete representation of the geometry, employed by the cut-cell method, guarantees the satisfaction of local and global conservation laws.

In fluid flows, the inception of cavitation occurs when the static pressure of the liquid drops below the vapor pressure. Sheet, bubble, vortex or supercavitation may appear. In some applications, such as high-speed underwater bodies, cavitation is beneficial for drag reduction, while in some others, such as propellers, undesirable since it can reduce their performance and durability. The ability to predict and control its inception is, thus, of importance. As a result, cavitation, and its forms, has been studied experimentally in cases, such as the Venturi channel [10], the flow over hydrofoils [11] and cylindrical head-forms [12], and extensive research has been dedicated to the development of numerical models [13][14][15][16] capable of simulating cavitating flows. These models are classified into two categories, namely, interface tracking and interface capturing methods. In interface tracking methods, the liquid/vapor interface is explicitly tracked and acts as an internal boundary, whereas, the interface capturing methods provide the liquid/vapor interface as part of the solution procedure. Differences of the interface capturing methods lie in the set of governing equations satisfied. In one-fluid models, the conservation laws for the mixture are solved and the phase change is controlled by a state law, such as the barotropic mixture law. In two-fluid models, the conservation laws are solved for each fluid with appropriate mass transfer rates at their interface, whereas hybrid models, such as that employed in the present work, are based on the conservation laws for the mixture and a phasic transport equation coupled with a cavitation model to control the phase change rates. In cases, such as ventilated cavitating flows, where more than two phases are considered, the models can be extended either by modifying their state law [17], including additional conservation laws or by including additional transport equations [13] to account for additional phases. Several formulations for the cavitation model have been proposed, mostly relying on empirical formulas, see [13] and [15], or bubble dynamics such as the Full Cavitation Model [16].

Previous work combining IBMs and cavitating flows is limited to the use of one-fluid models. [18] employed a cut-cell method coupled with a modified Tait law, to simulate weakly compressible cavitating flows through a closing fuel injector control valve. Recently, [17] and [19] presented computational studies on cavitating flows for an external gear pump and a projectile impacting a water jet, both using a continuous IBM.

The present work employs a hybrid model in conjunction with the cut-cell method to simulate cavitating flows. The effects of the rapidly varying finite volumes near the solid boundary, as well as their impact on the mass transfer model are investigated. A hybrid model has been selected as it may provide high flexibility and capture baroclinic vorticity generation [20]. The proposed implementation allows the prediction of cavitating flows while also maintaining the benefits of employing a cut-cell method. Studies of turbulent flows over isolated hydrofoils are presented.

Mathematical Model

The multi-phase Reynolds-Averaged Navier-Stokes (RANS) equations are adopted to simulate steady-state cavitating flows. A homogemeous mixture comprises liquid and vapor, both considered to be incompressible. The mixture is described by the momentum and phasic continuity equations. The continuity equation for the vapor phase is replaced by the one for the mixture, rather than using separate continuity equations for each phase. The retained mixture and liquid continuity equations are enriched with source terms to simulate mass transfer associated with the cavitation. From the numerical point of view, pseudo-time derivatives are introduced to

render the system hyperbolic in space and time and alleviate numerical stiffness created by the low speed of sound of the mixture.

The Jones and Launder $k - \varepsilon$ turbulence model [21], coupled with the Spalding's composite wall function [22] technique, is employed. The turbulence model takes into account the mixture densities and viscosities, with the turbulent variables referring to the mixture.

Governing equations

In vector form, the system of the preconditioned multi-phase mean-flow and turbulence equations for the mixture, is written as:

$$\Gamma \frac{\partial \mathbf{Q}}{\partial \tau} + \frac{\partial \mathbf{F_j}}{\partial x_j} - \frac{\partial \mathbf{F_j^v}}{\partial x_j} - \mathbf{S} = \mathbf{0}$$

$$\frac{\partial \mathbf{Q_T}}{\partial \tau} + \frac{\partial \mathbf{F_{T_j}}}{\partial x_j} - \frac{\partial \mathbf{F_{T_j}^v}}{\partial x_j} - \mathbf{S_T} = \mathbf{0} \tag{1}$$

with $\mathbf{Q} = [\, p \; \mathbf{u} \; a_l \,]^T$ the mean-flow state vector, $\mathbf{Q_T} = [\, \varrho_m k \;\; \varrho_m \varepsilon]^T$ the turbulent variables state vector, τ the pseudo-time and x_j the cartesian directions. The convective, $\mathbf{F_j}, \mathbf{F_{T_j}}$, viscous, $\mathbf{F_j^v}, \mathbf{F_{T_j}^v}$ fluxes and source vectors, $\mathbf{S}, \mathbf{S_T}$ are given by:

$$\mathbf{F_j} = \begin{bmatrix} u_j \\ \varrho_m u_j u_i + \delta_j^i \, p \\ a_l u_j \end{bmatrix}, \mathbf{F_j^v} = \begin{bmatrix} 0 \\ (\mu_m + \mu_{m,t})\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \\ 0 \end{bmatrix}, \mathbf{S} = \begin{bmatrix} (\dot{m}^+ + \dot{m}^-)\left(\frac{1}{\varrho_l} - \frac{1}{\varrho_v}\right) \\ \mathbf{0} \\ (\dot{m}^+ + \dot{m}^-)\frac{1}{\varrho_l} \end{bmatrix}$$
$$\tag{2}$$

$$\mathbf{F_{T_j}} = \begin{bmatrix} \varrho_m k u_j \\ \varrho_m \varepsilon u_j \end{bmatrix}, \; \mathbf{F_{T_j}^v} = \begin{bmatrix} \left(\mu_m + \frac{\mu_{m,t}}{\sigma_k}\right)\frac{\partial k}{\partial x_j} \\ \left(\mu_m + \frac{\mu_{m,t}}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j} \end{bmatrix}, \; \mathbf{S_T} = \begin{bmatrix} \mathbb{P} - \varrho_m \varepsilon \\ (C_1 \mathbb{P} - C_2 \varrho_m \varepsilon)\frac{\varepsilon}{k} \end{bmatrix} \tag{3}$$

where p, $\mathbf{u} \left(=[\, u, \; v]^T\right)$, a, ϱ, μ, k, ε refer to pressure, velocity vector, volume fraction, density, viscosity coefficient, turbulent kinetic energy and turbulence dissipation rate, respectively. Subscripts m, l, v refer to the mixture, liquid and vapor phase, respectively. δ is the Kronecker delta. Quantities $\dot{m}^\pm$ are associated with the mass transfer model rates characterizing both phase changes; for them, the Kunz cavitation model [13] is implemented. The evaporation term m^- is a function of the pressure and the liquid volume fraction whereas the condensation term m^+ is a function of the liquid volume fraction, a_l,

$$\dot{m}^- = \frac{C_{dest} \varrho_v a_l \, min\,[0, p - p_v]}{\left(\frac{1}{2}\varrho_l U_\infty^2\right) t_\infty}, \qquad \dot{m}^+ = \frac{C_{prod} \varrho_v a_l^2 \,(1 - a_l)}{t_\infty} \tag{4}$$

The empirical time rate constants C_{dest} and C_{prod} are case-dependent and their values are non-dimensionalized with respect to the mean-flow time scale, $t_\infty = \frac{L}{U_\infty}$. L is the characteristic length scale, in the case of hydrofoils their chord length, and p_v is the vapor pressure.

The preconditioning matrix Γ takes the form [13]

$$\Gamma = \begin{bmatrix} \frac{1}{\varrho_m \beta^2} & 0 & 0 & 0 \\ 0 & \varrho_m & 0 & u\Delta\varrho \\ 0 & 0 & \varrho_m & v\Delta\varrho \\ \frac{a_l}{\varrho_m \beta^2} & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

with β the pseudo-compressibility parameter and $\Delta\varrho = \varrho_l - \varrho_v$. The mixture density and molecular viscosity are based on a local volume-averaging:

$$\varrho_m = a_l \varrho_l + (1 - a_l)\varrho_v \tag{6}$$

$$\mu_m = a_l \mu_l + (1 - a_l)\mu_v \tag{7}$$

The turbulence production term is $\mathbb{P} = \mu_{m,t} \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right) \frac{\partial u_j}{\partial x_i}$ and the mixture turbulent viscosity coefficient is $\mu_{m,t} = C_\mu \varrho_m \frac{k^2}{\varepsilon}$. The turbulence model constants are: $C_\mu = 0.09$, $C_1 = 1.44$, $C_2 = 1.92$. Spalding's wall function formula, resolves both the viscous and the log layer through a single formula,

$$y^+ \left(u^+ \right) = u^+ + e^{-\kappa B} \left(e^{\kappa u^+} - 1 - \kappa u^+ - \frac{1}{2} \left(\kappa u^+ \right)^2 - \frac{1}{6} \left(\kappa u^+ \right)^3 \right) \tag{8}$$

with $u^+ = \frac{u_t}{u_\tau}$, $y^+ = \frac{u_\tau \Delta}{\nu}$ and u_τ, Δ, ν, κ represent the friction velocity, distance from the wall, kinematic viscosity and von Karman constant, respectively.

Discretization and Numerical Solution

To solve the homogeneous multi-phase flow equations, a cell-centered, finite-volume scheme on unstructured meshes is employed. Eqs. (1) are integrated and discretized with second-order accuracy in space. The resulting linearized system is solved using a block Gauss-Seidel method. The implementation is able to run in parallel on distributed memory systems by partitioning the mesh into subdomains, communicating using the MPI protocol.

Generation of a Geometry Adapted Computational Mesh

The generation of an adapted computational mesh with embedded boundaries is fully automated [23]. Initially, a uniform mesh is generated and cells intersected with the body surface are identified. Then, a recursive algorithm is employed to refine all intersected cells until the user-defined minimum cell volume has been reached, without allowing the refinement levels of two adjacent cells to differ by more than one. Once the maximum refinement level is reached at the intersected cells, the fluid part of the cells is retained, whereas the solid part is discarded, thus creating the cut-cells. Occasionally, cut-cells with volumes noticeably smaller than originally can be created that could stiffen the numerical solution of the flow equations [9]. To avoid convergence difficulties due to these cells, algorithms that merge cut-cells with volumes less than a user-defined threshold value with one of their immediate neighbours are employed [9]. Merged cells are differently shaped finite volumes, treated identically to all other cut-cells.

This procedure produces Cartesian meshes that are generally unsuitable for CFD simulations with large flow gradients present, e.g the viscous boundary layers and the cavity interface, since the cells in the immediate vicinity of solid walls abruptly change refinement levels, Fig. 1a.

Therefore, the refinement levels across the computational domain are smoothed to increase mesh resolution close to solid walls, by employing a progressive refinement algorithm. The algorithm, based on the distance of each cell from the geometry and a sigmoid function, computes the maximum allowable volume of each cell. Cells exceeding this value are refined. In the absence of smoothing, the refinement levels of neighbouring cells, result in increased numerical errors near the geometry. The mesh generation and adaptation process is based on a hierarchical quadtree data structure [9] and follows stages illustrated in Fig. 1.

(a)   **(b)**
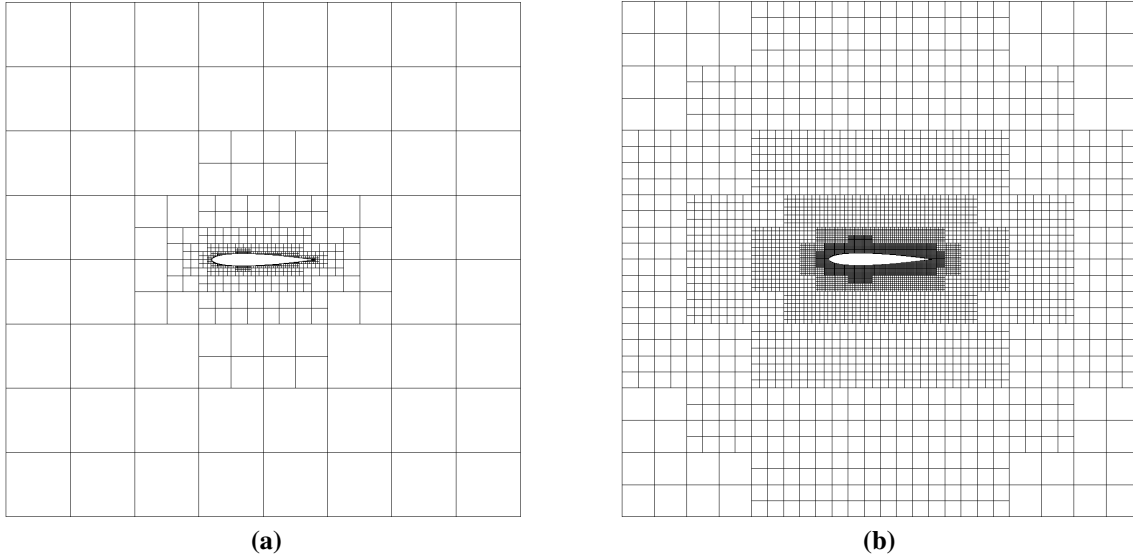
Figure (1): Stages of mesh generation. Computational mesh of the NACA0012 hydrofoil (a) before and (b) after implementing the progressive refinement algorithm.

Discretization of the Governing Equations

Equations (1) are integrated, over a finite volume Ω, yielding:

$$\Gamma \frac{V}{\Delta \tau} \Delta \mathbf{Q} + \int_{\partial\Omega} \mathbf{F_j} n_j \, dS - \int_{\partial\Omega} \mathbf{F_j^v} n_j \, dS - \int_{\Omega} \mathbf{S} dV = \mathbf{0} \tag{9}$$

where $\mathbf{n}$ the unit normal vector to the faces of each volume. The discretization of the turbulence model equations is similar, thus, omitted.

The discretization of the convective fluxes in Eq. (9) is based on the Roe scheme [24], using a second-order linear reconstruction. The state variable gradients at each cell centre are estimated by a linear least-squares method using all face neighbours. In the presence of strong flow gradients, the extrapolation procedure is limited to provide a monotone-preserving TVD scheme. The pseudo-time step $\Delta \tau$ is determined by stability criteria.

The inviscid flux, $\boldsymbol{\Phi}$, at each cell face reads

$$\boldsymbol{\Phi} = \frac{1}{2} \left(\mathbf{F^L} + \mathbf{F^R} - \Gamma_{Roe} \left| \hat{A_{Roe}} \right| \left(\mathbf{Q}^- - \mathbf{Q}^+ \right) \right)$$
$$\mathbf{Q}^{\pm} = \mathbf{Q^{L/R}} + \psi \, \nabla \mathbf{Q} \Delta \mathbf{x} \tag{10}$$

$$\left| \hat{A_{Roe}} \right| = \left| \Gamma_{Roe}^{-1} A_{Roe} \right|, \qquad |A_{Roe}| = M_{Roe} |\Lambda_{Roe}| M_{Roe}^{-1} \tag{11}$$

In the equations above, ψ is the Barth-Jespersen limiter [25] value, M, M^{-1} are the right and left eigenvectors of the preconditioned Jacobian matrix, $\Gamma^{-1}A$, and Λ is the diagonal eigenvalue matrix, $diag\left(\Lambda\right) = [\mathbf{u}, \ \mathbf{u} + c, \ \mathbf{u} - c, \ \mathbf{u}]^{T}$. The pseudo-sound, c, is computed as $c = \sqrt{(u_n + \beta^2)}$ and u_n is the normal to the face velocity, [13]. Superscripts $+, -$ refer to the reconstructed variables, while L/R to the adjacent to the face cells. Subscript (Roe) denotes Roe-averaged variables, defined as

$$\varphi_{Roe} = \frac{\sqrt{\varrho^+}\varphi^+ + \sqrt{\varrho^-}\varphi^-}{\sqrt{\varrho^+} + \sqrt{\varrho^-}}, \quad \varphi \in (\mathbf{u}, a_l, k, \varepsilon) \tag{12}$$

The velocity gradients at each cell face, necessary for the viscous fluxes, are computed using central differences. At the faces of cut-cells or those separating cells with different refinement levels, the barycenters of the corresponding cells are not aligned with the face midpoint and, thus, undergo a non-orthogonality correction.

For the solution procedure, the Jacobians $A = \frac{\partial \mathbf{F}}{\partial \mathbf{Q}}$ are computed analytically. Source terms are linearized in order to promote diagonal dominance.

Implementation of Wall Functions

A one-layer submesh, Fig. 2, with constant normal distance from the solid boundary is generated, as in [26]. Cells dimension ($\Delta\mathbf{x}$) can be computed based on their refinement level, by exploiting the quadtree data structure [9]. Then, the normal distance is computed $\Delta = 1.6min(\Delta\mathbf{x})$, and is the same for all cut-cells, since all cut-cells reach the same level. The normal distance ensures that the submesh nodes reside outside the (possibly merged) cut-cell. Eq. (8) is solved to extract the required wall shear stress and turbulent variables boundary conditions, by employing the submesh. Finally, the velocity gradients of the cut-cells are replaced by the ones provided by the Spalding formula [26]. The submesh node corresponding to the solid-
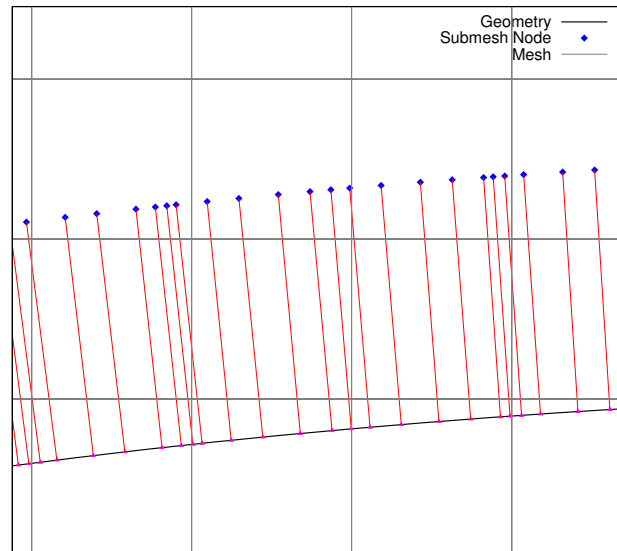

Figure 2: One-layer submesh used to implement the wall function technique. Each submesh node is connected with a solid face of the cut-cell.

node line closest to each cut-cell barycenter is located and used to compute the friction velocity u_τ. Each cut-cell is associated with a submesh node on which Eq. (8) is solved. The tangent to

the wall velocity u_t at that node is linearly reconstructed from the nearest cell to compute u_τ, iteratively. The shear stress is computed separately at each solid face to account for cases where the solid face direction changes inside a cut-cell, providing a more accurate computation of the corresponding solid face viscous flux.

Numerical Treatment of Cut-Cells - Application of Boundary Conditions

A blow-up view of the area close to the interface with a solid body is presented in Fig. 3a. Two cut-cells, a merged (blue) and non-merged (brown), are included for illustative purposes. A merged cell is created when a slave cell (the smaller one) attaches to a master cell to create a new bigger cell. A master cell accomodates one slave cell at most to prevent the formation of non-convex finite volumes. The sizes of the faces of a slave cell are compared to determine which neighbour should become its master cell; the one with the largest common face is selected as master cell.

Fig. 3b shows a cut-cell finite volume along with two fluxes (*4*, *5*) due to the presence of the solid boundaries. For brevity, the following refer to inviscid flows where the no-penetration $(\mathbf{u} \cdot \mathbf{n} = 0)$ condition is applied. For all internal faces, (*1*, *2*, *3*, *6*), fluxes are cast in Cartesian coordinates, e.g. flux through face *2* becomes $\mathbf{F} \cdot \hat{\mathbf{n}} = [u, \; \varrho_m u^2 + p, \; \varrho_m uv, \; 0]^T$. On solid faces, boundary conditions are weakly imposed. Flow variables are extrapolated at the midpoint of each solid face and the appropriate flux, $\mathbf{F}^w = [0, p_{face} \, \mathbf{n}, \; 0]^T$ is imposed.
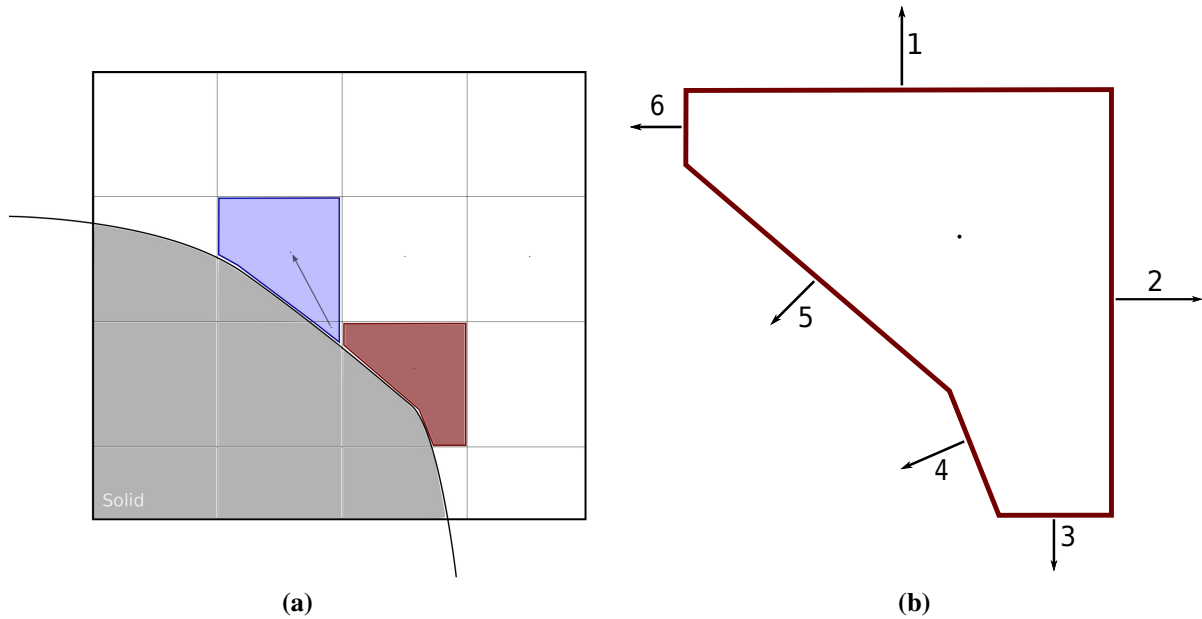

Figure (3): (a) Areas near the solid boundary of a mesh with embedded boundaries where a merged (blue) and a non-merged (brown) cut-cell can be seen. (b) Enlarged view of a cut-cell; normal vectors represent fluxes. Solid wall conditions are imposed along faces *4* and *5*.

Results and Discussion

The capabilities of the cut-cell solver in the simulation of sheet cavitation are demonstrated in numerical studies of two isolated hydrofoils, for different cavitation numbers, and results are

compared with published data, [11][14][27]. Sheet cavitation has been experimentally observed to exhibit a quasi-steady behaviour and the experimental data [11] obtained regarding sheet cavitation are time-averaged. Therefore, it is a common practice, e.g [14][27][28], to perform simulations using a steady-state solver. The cavitation number is defined as $\sigma = \frac{p_\infty - p_v}{\frac{1}{2}\varrho_l U_\infty^2}$ and the pseudo-compressibility parameter as $\beta^2 \approx 10 U_\infty^2$.

Cavitating Flow around the NACA0012 hydrofoil

The first case considers mid-chord sheet cavitation on a NACA0012 hydrofoil, based on the numerical study presented in [14]. [14] employed the same cavitation model in a pressure-based (SIMPLE) algorithm, with a density ratio equal to $\frac{\varrho_l}{\varrho_v} = 1000$, which is also valid here. The Reynolds number is $Re_c = 2 \cdot 10^6$ based on the chord and the infinite flow angle is $1°$. With a cavitation number equal to $\sigma = 0.42$, a thin layer of vapor is created at the mid-chord of the hydrofoil that exhibits no re-entrance jet/recirculation. A mesh with $42K$ cells is generated with sufficient progressive refinement near the solid body and an average $\overline{y}^+ \approx 90$ of the first cell barycenters off the wall (Fig. 5a) to capture the thin vapor layer.

In the literature [14][28], it is reported that the empirical time rate constants (C_{prod}, C_{dest}) of the mass transfer model are case dependent and different values are proposed and used by different codes for the same test cases. In the current implementation, the proposed time rate constants also produced unrealistic results. A parametric study was conducted to calibrate the dimensionless mass transfer time rates constants, $C^+ = \frac{C_{prod}}{t_\infty}$, $C^- = \frac{C_{dest}}{t_\infty}$.

Fig. 4a shows pressure distributions obtained during this study with several pairs of constants and their impact on the converged solution. In the results presented hereby, the pressure inside the cavity is equal to p_v, but this is not the case if C^- becomes lower. The pressure gradients aft the cavity are influenced by the correct balance between the two source terms. Too small values of C^- shorten the cavity length, while too big values of C^+ steepen the pressure gradients at the cavity closure region.

Fig. 4b compares the results of the surface pressure profile, with the selected time rate constants, of the current implementation with two numerical results obtained on body-fitted mesh solvers under the same conditions and mass transfer model, though with different time rates. Differences are limited to the cavity closure region.

In Fig. 5b, the vapor volume fraction field and, therefore, the cavity is presented. A thin layer of vapor is generated along the suction side of the hydrofoil. The maximum vapor volume fraction inside the cavity $a_{v_{max}}$ approaches 0.9. The liquid-vapor interface is spread across multiple cells. This might be due to the smooth pressure gradients observed at the cavity closure, which implies that the condensation source term is weak enough to avoid an instantaneous phase change and make it appear gradually. Fig. 5a shows a close-up view of the area near the cavity. By plotting the liquid volume fraction iso-line $a_l = 0.99$ over the mesh, it can be seen that the cavity resides inside the finest refinement levels of the mesh.
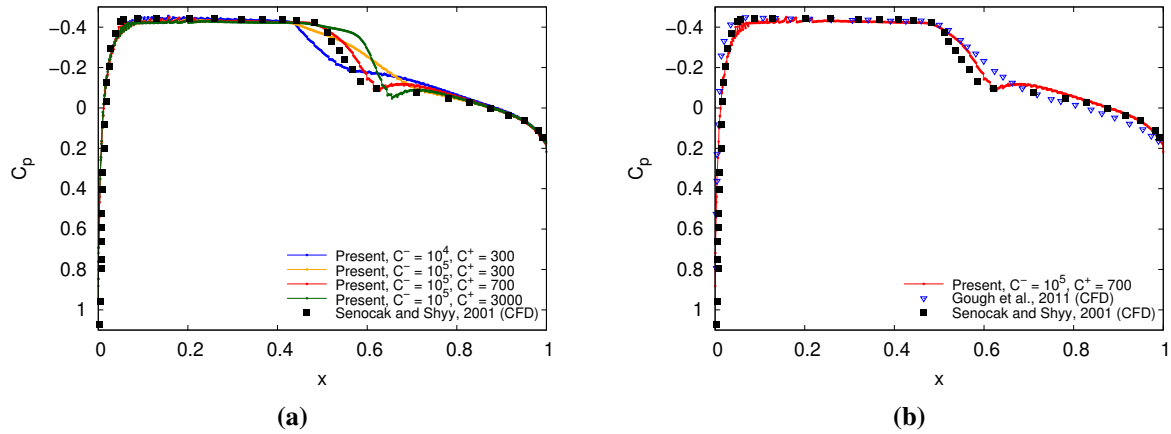
Figure 4: Surface pressure suction side profile over a NACA0012 **hydrofoil,** α=1°, Re_c=2·10⁶ **and** σ=0.42. **(a) Parametric study to find the best value set of** C^+ **and** C^-. **(b) Comparison with the numerical studies of [14][28] and the best value-set of time constants found.**
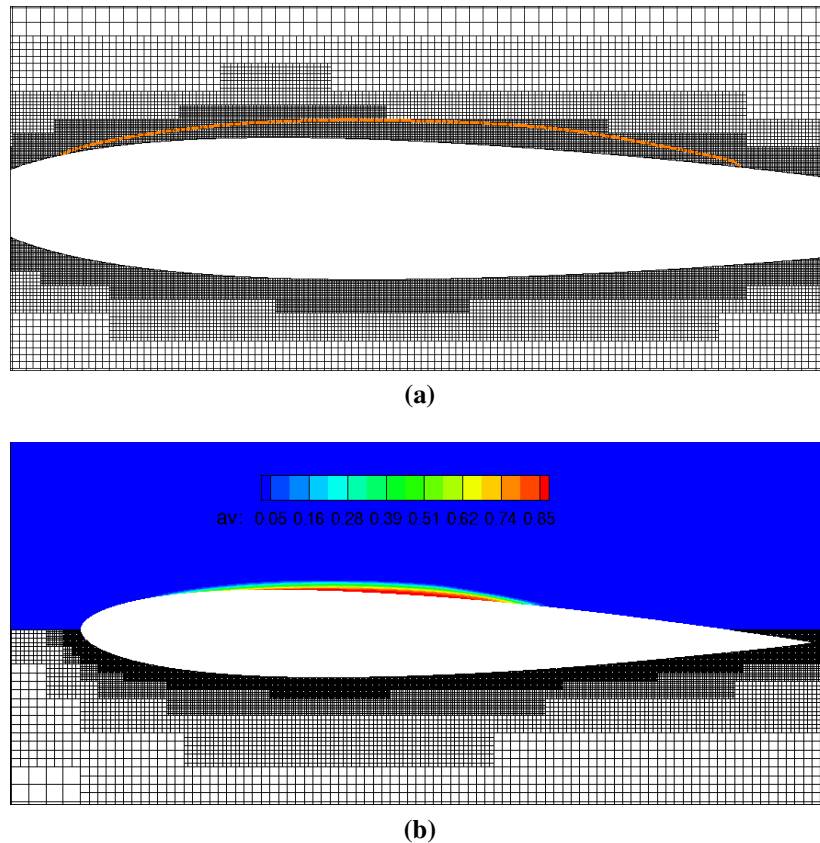

(a)

(b)

Figure 5: NACA0012 **hydrofoil: (a) Liquid volume fraction iso-line (orange)** a_l=0.99 **which determines the size of the cavity along with the smoothed mesh refinement near the geometry. (b) Iso-areas of the vapor volume fraction** a_v.

Cavitating Flow around the NACA66(MOD) hydrofoil

In this section, the cavitating flow over a NACA 6-series hydrofoil is presented. The hydrofoil has been experimentally investigated in [11] for leading edge and mid-chord cavitation

and time-averaged pressure distributions are available in the literature [11]. The cambered NACA66(MOD) hydrofoil has a camber ratio of 0.02, a mean line of 0.8, a thickness ratio of 0.09 and a chord length of $0.1524m$. The presented study focuses on steady-state simulations of flows with an infinite flow angle of $\alpha = 4°$, a Reynolds number $Re_c = 2 \cdot 10^6$, based on the chord length, a density ratio $\frac{\varrho_l}{\varrho_v} = 1000$ and three cavitation numbers, $\sigma = \infty$, $\sigma = 0.91$ and $\sigma = 0.84$. A cartesian mesh with $80K$ cells is generated with sufficient mesh resolution near the geometry and $\overline{y}^+ \approx 50$. The best value-set of the two time rate constants was found to be $C^- = 10^6$, $C^+ = 9000$ after a parametric study. Furthermore, for the cavitation numbers $\sigma = 0.91$, 0.84 the surface pressure distributions are also compared with the numerical results of [27]. The latter employed a commercial CFD solver and calibrated the mass transfer rates for the same conditions over a NACA66(MOD) hydrofoil.

The first case, $\sigma = \infty$, is the non-cavitating case and is included to validate the cut-cell solver with the experimental data of [11]. In the other cases, with $\sigma = 0.91$, $\sigma = 0.84$, vapor is generated since the pressure decreases below the pressure vapor and cavities are observed. Fig. 6a shows the surface pressure distributions of the non-cavitating case $(\sigma = \infty)$ and the experimental data from [11]. The surface pressure distribution compares favourably with the time-averaged experimental data. In Fig. 6b, the surface pressure distribution in the case with $\sigma = 0.91$ is presented and compared with measurements [11] and other numerical results [27]. The results obtained with the cut-cell method show that the pressure inside the cavity remains constant and agrees with both numerical and experimental data. Furthermore, comparing with the numerical results, an underprediction of the cavity length is also observed. Fig. 7a, shows the vapor vol-
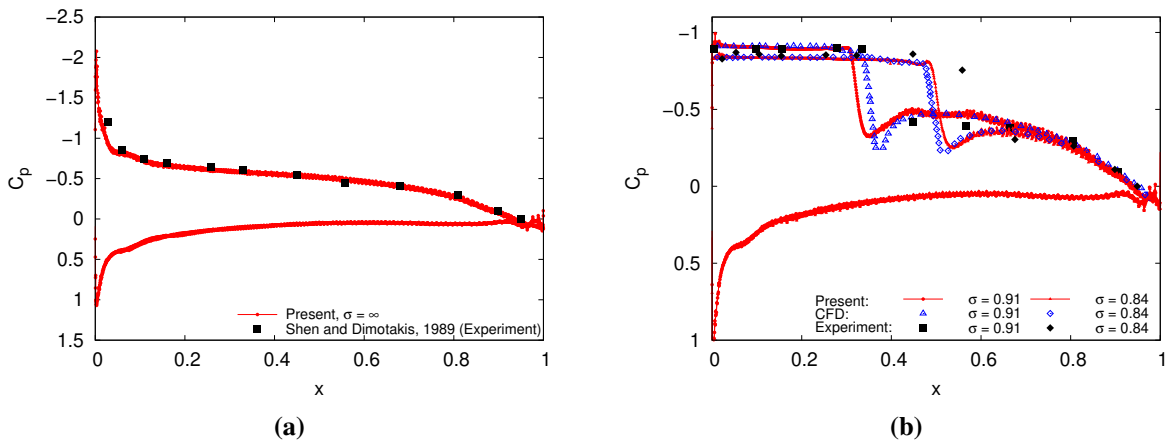

Figure 6: Surface pressure distribution over the NACA66(MOD), (a) The non-cavitating $(\sigma = \infty)$ case (b) Cavitating cases of $\sigma = 0.91$ and $\sigma = 0.84$.

ume fraction with the computational mesh employed. In Fig. 7c the pressure field is presented. The constant pressure inside the cavity can be seen and its value is approximately equal to the negative of the cavitation number $\sigma = 0.91$, as expected.

By further reducing the cavitation number, a larger cavity is created. In Fig. 6b, the surface pressure distribution for $\sigma = 0.84$ is compared with experimental time-averaged data [11] and numerical results [27]. Both numerical results produce the appropriate pressure inside the cavity and capture the same pressure recovery trend. Fig. 7b shows the vapor volume fraction on the suction side of the hydrofoil. The pressure flow field is shown in Fig. 7d. Finally, comparing Fig. 7(a)-(b), it can be seen that decreasing the cavitation number increases the size of the vapor cavity created.
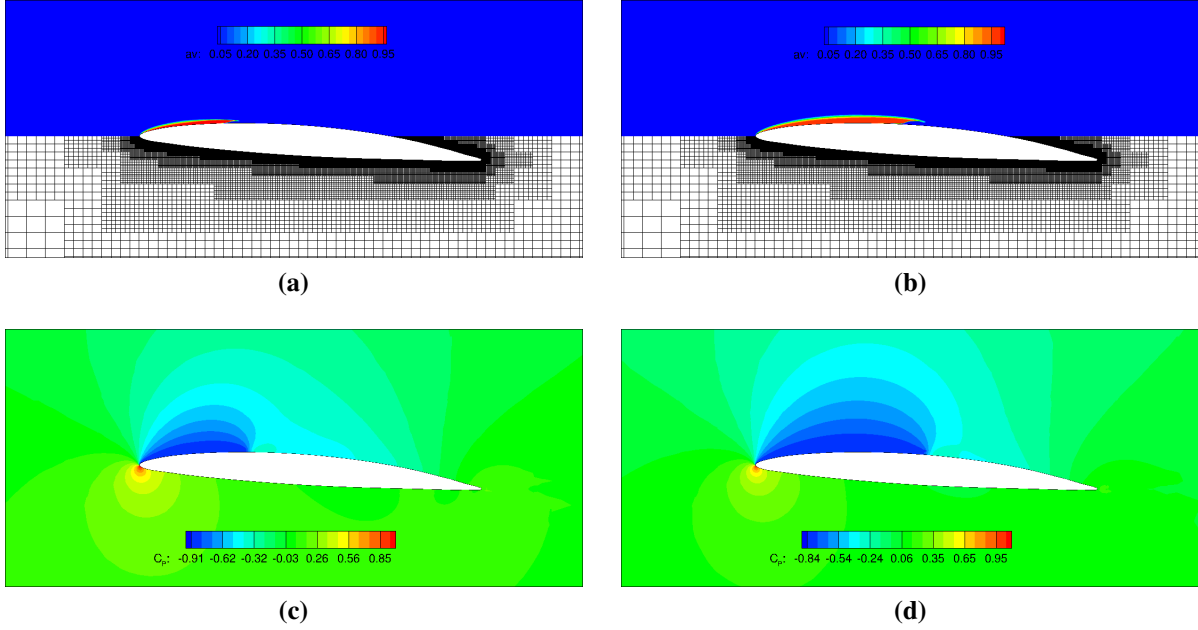
Figure 7: Cavitating NACA66**(MOD) hydrofoil.** σ=0.91 **(left) and** σ=0.84 **(right). (a) - (b) Vapor volume fraction along with the computational mesh used and (c) - (d) Pressure coefficient iso-areas.**

Compared to the experimental data, both numerical solvers underpredict the cavity length at $\sigma = 0.84$. The differences noted, could be attributed to the additional multi-phase turbulent effects which dominate in these areas. Addressing the additional turbulent effect in cavitating flows is still an ongoing research area and new findings could improve numerical predictions. However, this is beyond the scope of the present paper.

The use of flow-based adaptation is common in transonic and supersonic flows [23], where shock waves are present and require increased mesh resolution locally. IBMs, employing hierarchical data structures, make flow-based adaptation techniques easy to implement and is considered one of their main advantages over their body-fitted counterparts. Hence, a natural next step is the addition of flow-based refinement in the multi-phase solver as steep flow gradients also characterize cavitating flows. The choice and definition of the adaptation sensor, that identifies the areas of interest and implements the refinement algorithm, require special attention. An obvious choice may be the normalized $\|\nabla a_l\|$, where large gradients are present only near the cavity interface, see in Fig. 8. The pressure gradients could also be considered as an adaptation sensor since the mass transfer models are closely related to the field pressure.
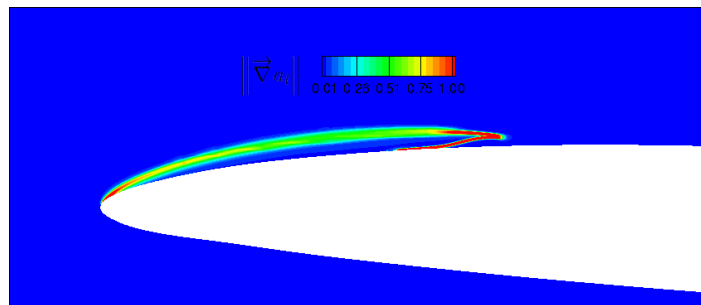

Figure 8: Iso-areas of the normalized $\|\nabla a_l\|$ **over the cavitating NACA**66 **hydrofoil.**

Conclusions

A cut-cell solver for a homogeneous mixture model has been implemented to simulate steady, turbulent, cavitating flows, based on the Kunz model. The cut-cells and the one-layer submesh that help to employ the Spalding's composite wall function model for the $k - \varepsilon$ turbulence model, together with the merging of the small cut-cells, are shown to handle the large density gradients present in cavitating flows.

To assess the programmed method, two hydrofoil cases featuring mid-chord and leading edge sheet cavitation have been selected and comparisons with experimental and numerical data are presented. The cavitation model implemented proved to be sensitive to the two time rate constants which should, therefore, be carefully selected to avoid nonphysical solutions or the divergence of the numerical scheme. The automatic mesh generation and refinement, even in complex geometries, offered by the cut-cell method, renders it an appealing alternative to CFD solvers employing body-fitted meshes. This is particularly important in case of moving bodies, which is not the case in this paper; to summarize, the programmed software simulates both 2D and 3D flows, with stationary or moving bodies (see [29], presented in the same conference).

Acknowledgement

References

[1] C. S. Peskin, "Flow Patterns Around Heart Valves: A Numerical Method," *Journal of Computational Physics*, vol. 10, pp. 252–271, 1972.

[2] D. Clarke, M. Salas, and H. Hassan, "Euler Calculations for Multielement Airfoils Using Cartesian Grids," *AIAA Journal*, vol. 24, no. 3, pp. 353–358, 1986.

[3] T. Ye, R. Mittal, H. S. Udaykumar, and W. Shyy, "An Accurate Cartesian Grid Method for Viscous Incompressible Flows with Complex Immersed Boundaries," *Journal of Computational Physics*, vol. 156, pp. 209–240, 1999.

[4] F. Capizzano, "Turbulent Wall Model for Immersed Boundary Methods," *AIAA Journal*, vol. 49, pp. 2367–2381, 2011.

[5] M. J. Berger and M. J. Aftosmis, "An ODE-Based Wall Model for Turbulent Flow Simulations," *AIAA Journal*, vol. 56, pp. 700–714, 2018.

[6] M. J. Aftosmis, M. J. Berger, and J. E. Melton, "Robust and Efficient Cartesian Mesh Generation for Component-Based Geometry," *AIAA Journal*, vol. 36, pp. 952–960, 1998.

[7] R. Mittal and G. Iaccarino, "Immersed Boundary Methods," *Annual Review of Fluid Mechanics*, vol. 37, no. 1, pp. 239–261, 2005.

[8] J. Kim, D. Kim, and H. Choi, "An Immersed-Boundary Finite-Volume Method for Simulations of Flow in Complex Geometries," *Journal of Computational Physics*, vol. 171, pp. 132–150, 2001.

[9] H. Ji, F. S. Lien, and E. Yee, "A new Adaptive Mesh Refinement Data Structure with an Application to Detonation," *Journal of Computational Physics*, vol. 229, pp. 8981–8993, 2010.

[10] B. Stutz and J.-L. Reboud, "Two-phase Flow Structure of Sheet Cavitation," *Physics of Fluids*, vol. 9, pp. 3678–3686, 1997.

[11] Y. Shen and P. E. Dimotakis, "The Influence of Surface Cavitation on Hydrodynamic Forces," in *Proc. 22nd ATTC.*, (St. John's, Newfoundland, Canada), pp. 44–53, 1989.

[12] J. S. Rouse, Hunter and McNown, "Cavitation and Pressure Distribution: Head Forms at Zero Angle of Yaw," in *Studies in Engineering Bulletin*, (Iowa City), p. 32, State University of Iowa, 1948.

[13] R. F. Kunz, D. A. Boger, D. R. Stinebring, T. S. Chyczewski, J. W. Lindau, H. J. Gibeling, S. Venkateswaran, and T. R. Govindan, "A Preconditioned Navier-Stokes Method for Two-phase Flows with Application to Cavitation Prediction," *Computers and Fluids*, vol. 29, pp. 849–875, 2000.

[14] I. Senocak and W. Shyy, "Numerical Simulation of Turbulent Flows with Sheet Cavitation," in *CAV2001: Fourth International Symposium of Cavitation*, (California Institute of Technology, Pasadena, CA, U.S.A.), pp. 20–23, 2001.

[15] C. L. Merkle, J. Feng, and P. E. O. Buelow, "Computational Modeling of the Dynamics of Sheet Cavitation," in *Proceedings of the 3rd International Symposium on Cavitation*, (Grenoble, France), 1998.

[16] A. K. Singhal, M. M. Athavale, H. Li, and Y. Jiang, "Mathematical Basis and Validation of the Full Cavitation Model," *Journal of Fluids Engineering*, vol. 124, no. 3, p. 617, 2002.

[17] M.-G. Mithun, P. Koukouvinis, I. K. Karathanassis, and M. Gavaises, "Numerical Simulation of Three-phase Flow in an External Gear Pump Using Immersed Boundary Approach," *Applied Mathematical Modelling*, vol. 72, pp. 682–699, 2019.

[18] F. Örley, V. Pasquariello, S. Hickel, and N. A. Adams, "Cut-element based Immersed Boundary Method for Moving Geometries in Compressible Liquid Flows with Cavitation," *Journal of Computational Physics*, vol. 283, pp. 1–22, 2015.

[19] E. Stavropoulos Vasilakis, N. Kyriazis, P. Koukouvinis, M. Farhat, and M. Gavaises, "Cavitation Induction by Projectile Impacting on a Water Jet," *International Journal of Multiphase Flow*, vol. 114, pp. 128–139, 2019.

[20] I. Senocak, "Computational Methodology for the Simulation of Turbulent Cavitating Flows. PhD thesis, University of Florida, 2002.

[21] W. Jones and B. Launder, "The Prediction of Laminarization with a Two-equation Model of Turbulence," *International Journal of Heat and Mass Transfer*, vol. 15, pp. 301–314, 1972.

[22] D. B. Spalding, "A Single Formula for the Law of the Wall," *Journal of Applied Mechanics*, vol. 28, no. 3, p. 455, 1961.

[23] K. Samouchos, S. Katsanoulis, and K. C. Giannakoglou, "Unsteady Adjoint to the Cut-cell Method Using Mesh Adaptation on GPUs," in *ECCOMAS Congress*, (Crete Island, Greece), 2016.

[24] P. Roe, "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *Journal of Computational Physics*, vol. 43, pp. 357–372, 1981.

[25] T. Barth and D. Jespersen, "The Design and Application of Upwind Schemes on Unstructured Meshes," in *27th Aerospace Sciences Meeting*, (Reston, Virigina), AIAA, 1989.

[26] M. J. Berger and M. J. Aftosmis, "Progress Towards a Cartesian Cut-Cell Method for Viscous Compressible Flow," in *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2012.

[27] M. Morgut and E. Nobile, "Numerical Predictions of Cavitating Flow Around Model Scale Propellers by CFD and Advanced Model Calibration," *International Journal of Rotating Machinery*, vol. 2012, 2012.

[28] H. Gough, A. L. Gaitonde, and D. P. Jones, "A Dual-time Central-difference Interface-capturing Finite Volume Scheme Applied to Cavitation Modelling," *International Journal for Numerical Methods in Fluids*, vol. 66, pp. 452–485, 2011.

[29] K. Samouchos, D. Kapsoulis, X. Trompoukis, K. Giannakoglou, "Shape Optimization of 3D Diaphragm Pumps using the Continuous Adjoint Approach to the Cut-Cell Method," in *ICCM2019*, (Singapore), 2019.