VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ENERGETICKÝ ÚSTAV

ENERGY INSTITUTE

MODELOVÁNÍ VTOKOVÝCH VÍRŮ

INTAKE VORTEX MODELING

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE AUTHOR	Bc. Jiří Galuška
VEDOUCÍ PRÁCE SUPERVISOR	doc. Ing. Pavel Rudolf, Ph.D.

BRNO 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA STROJNÍHO INŽENÝRSTVÍ

Zadání diplomové práce

Ústav:	Energetický ústav
Student:	**Bc. Jiří Galuška**
Studijní program:	Strojní inženýrství
Studijní obor:	Fluidní inženýrství
Vedoucí práce:	**doc. Ing. Pavel Rudolf, Ph.D.**
Akademický rok:	2016/17

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Modelování vtokových vírů

Stručná charakteristika problematiky úkolu:

Vtokové víry způsobují zhoršení provozních vlastností čerpadel a vodních turbín. V současné době je jejich studium založeno především na experimentálním modelovém výzkumu a jednofázových výpočtových simulacích. S nárůstem výpočtového výkonu a vývojem nových metod se otevírají možnosti pro dvoufázové simulace hladinových vírů.

Cíle diplomové práce:

– odladění modelu dvoufázového proudění založeného na metodě Volume of Fluid pro testovací úlohu hladinového víru
– experimentální výzkum vtokových vírů na laboratorním modelu sací jímky
– jedno a dvoufázový výpočet laboratorního modelu sací jímky
– vyhodnocení experimentu a výpočtu, závěr

Seznam doporučené literatury:

ŠKERLAVAJ, A., A. LIPEJ, J. RAVNIK a L. ŠKERGET. Turbulence model comparison for a surface vortex simulation. IOP Conference Series: Earth and Environmental Science. 2010, 12(1), 1-10.

CRISTOFANO, L., M. NOBILI a G. CARUSO. Numerical evaluation of gas core length in free surface vortices. Journal of Physics: Conference Series. 2014, 547(1), 1-11.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2016/17

V Brně, dne

L. S.

doc. Ing. Jiří Pospíšil, Ph.D. ředitel ústavu	doc. Ing. Jaroslav Katolický, Ph.D. děkan fakulty

ABSTRAKT

V diplomové práci jsou shrnuty základní pravidla návrhu čerpacích jímek. Dále byly rozebrány matematické modely vírů a metody k jejich určování a vizualizaci. Následně se autor zaměřil na numerické modelování pomocí CFD metod. Jsou zde rozebrány základní poznatky o jednofázovém a dvoufázovém modelování pomocí metody Volume of Fluid s aplikací na hladinové víry v programu OpenFOAM a ANSYS Fluent. Dále se autor v práci zaměřuje na rešerši vhodných turbulentních modelů a porovnání jejich výsledků na jednoduchém verifikačním příkladu. V práci byla úspěšně verifikována jednofázová metoda na určování délky vzduchového jádra hladinového víru. Na stejném příkladu byla verifikována i dvoufázová simulace v programu OpenFOAM a ANSYS Fluent. Výsledky obou řešičů a metod byly mezi sebou porovnány.

Dosažené poznatky byly použity k návrhu experimentální nádrže s geometrii podobnou průmyslovým čerpacím jímkám. Byla vytvořena mapa výskytů hladinových vírů pro různé provozní body. Jeden provozní bod s výskytem vírů byl přepočítán jednofázovou a dvoufázovou metodou. Bylo dosaženo pouze částečné shody experimentu s numerickým modelem.

ABSTRACT

This paper covers information research of basic design rules of industrial wet sumps. It describes mathematical models of vortices and method for their identification and visualization. Then the author focuses on CFD modeling of surface vortices with single phase and multiphase approach with Volume of Fluid method. Basic principles of multiphase CFD modelling in OpenFOAM and ANSYS Fluent are given. Description and benchmarking of suitable turbulence models is also present. The single phase and multiphase approach were successfully validated for a simple test case of bathtub surface vortex. Satisfactory agreement with experimental data was achieved. The accuracy and behavior of both solvers were compared between each other. This gives us useful tool for evaluation of inflow condition and danger of surface vortex occurrence in wet sumps.

The acquired knowledges were used to design an experimental test case with geometry similar to industrial wet sump. A map of surface vortex occurrence has been created for different operating points. One of the operating point has been used for numerical simulation (both single phase and multiphase). Partial agreement with experimental observation has been achieved.

KLÍČOVÁ SLOVA

Numerický model, OpenFOAM, ANSYS Fluent, určování délky víru, hladinový vír, analytická definice víru, návrh mokré čerpací jímky, vícefázové proudění, turbulence.

KEYWORDS

Numerical model, OpenFOAM, interFoam, ANSYS Fluent, gas core length determination, surface vortex, analytical definition of vortex, wet sump design, multiphase flow, Volume of Fluid, turbulence modelling, curvature correction.

BIBLIOGRAPHIC CITATION

GALUSKA, J. *Simulation of Intake Vortices*. Brno: Brno University of Technology, Faculty of Mechanical Engineering, 2017. 83 s., 1 attachment. Supervisor doc. Ing. Pavel Rudolf, Ph.D.

DECLARATION

I declare that I have prepared this master's thesis independently, under supervision of the master’s thesis supervisor. I have used no source other than those quoted and the literature I have used has been clearly cited in the text

26. May 2017

Author’s signature

…………………………………….

Jiri Galuska

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to doc. Ing. Pavel Rudolf, Ph.D. for his great support and guidance during this work.

CONTENT

INTRODUCTION

Recently, there has been a pressure on improving performance and efficiency of pumping systems. Wet sumps must be smaller to save construction costs and pumps must have higher efficiency to reduce operational cost. In case of reconstructions of existing facilities, engineers are required to fit more powerful pumps into small sumps, which are usually not suitable any more. To overcome these difficulties and design flaws, it is necessary to study flow conditions in wet sumps and to develop reliable methodology (experimental and numerical) for identification of undesirable flow conditions, such as strong swirling flow and vortex occurrence.

Bad design of a wet sump can induce swirling motion in pump intake and in extreme case lead to formation of surface vortices. Vortices can drag impurities floating on a water surface or suck detached bubbles from the gas core. Swirling flow and uneven flow distribution in a pump intake results in nonuniform loading of an impeller, increased loading of bearings, vibration and noise. Increased amount of vorticity can increase liability of impeller to cavitation. All this may lead to pumping system failure and consequently to dangerous situations. In extreme situations, full gas entrainment can occur, releasing great amount of air into the pumping system. Air in pumping systems is undesirable and may cause additional technological problems.

Traditionally the flow conditions in wet sumps were studied on a small laboratory models. However, model research is often expensive, require well equipped laboratory and it is difficult to achieve reproducibility. Therefore, it is natural to utilize Computational Fluid Dynamics (CFD) for this cause. As the computational performance increases and powerful scientific or commercial computational clusters (like Amazon cloud computing) are accessible to almost everybody, it is convenient to utilize additional information and increased accuracy of results obtained from multiphase numerical simulation. Because transient, multiphase simulation is very CPU intensive, validating open source CFD software would significantly reduce the final simulation costs. In this paper, numerical model has been verified on experimental data from simple test case of a bathtub vortex. Then the acquired knowledges were used for designing experiment, which geometry is similar to real industrial wet sump. Comparison of experimental study on a scaled model and results from numerical simulation are given in last chapters.

1 WET SUMP DESIGN

The design of pump intake is the most important aspect which determines the future pump performance, efficiency and life time. The pump inflow conditions are often overlooked and are not considered in the final choice of pumps, although they are probably the key aspect which determine if the pumping system will fail or will operate successfully. Excessive swirling in the intake may cause severe cavitation of impeller and reduce significantly the life time of the pump, bearings or shafts. Drawn air bubbles can cause abrupt changes in pump flow rate and with strong vibrations and pressure shocks can damage the piping and other technological equipment of pumping station.

The "P*umping Station Design*" (2005) with references to a directive ANSI/HI/i.8-1998 gives basic guidelines on designing sumps, wet wells and pump intakes. It summarizes the basic principles and rules and gives practical examples of sumps, which were experimentally tested and proven themselves to be reliable.

1.1 Guidelines on Designing Pump Intakes [1] [2]

The correct design of sump should allow the pumps to operate in optimal hydraulic condition in all operating points of the system. To achieve this, designer should mitigate following design flaws and operating conditions:

- Poor velocity distribution and excessive swirling in the pump intake. Sudden changes in flow direction shall be omitted, because it can cause vortices and uneven velocity profile.
- Conduit upstream form the bell mouth should be coaxial with a pump intake axis and it should be straight without elbows, fittings. It should also maintain constant diameter. Elbows and fittings may cause swirling and disturb the uniformity of the flow.
- Air entrainment to the pumping system either from surface vortices or air bubbles dispersed in sump. A free fall of a water from an inlet to wet sumps is undesirable, because air bubbles dispersed in liquid are easily transported to the inlet of a pump.
- Unstable inflow conditions in a sump.
- Inadequate NPSHA at the pump inlet.
- Avoid all sharp corners without fillets, because they cause subsurface vortices.
- Mean horizontal velocity in wet sump should be less than 0.3 m/s. The local velocities vary greatly from the mean value.
- Distance between multiple suction bells should not be closer than 2.5 D (D is the diameter of the pipe bell).
- Local and mean velocity in bell mouth should not vary more than 10% from each other.

There is no clear agreement on the maximal allowable velocity in pump intakes. The ANSI/HI 9.8-1998 recommends not to exceed 1.7 m/s, however authors in "Pumping Station Design" recommend limiting the maximal velocity even further to 1.3 m/s. When intake velocity is higher, adverse flow conditions such as uneven velocity distribution in bell mouth or underwater vortices may occur in pump intake. In cases, where is necessary to increase the velocity over 1.3 m/s, it is highly recommended to perform experimental study or CFD analysis of the sump.

Further on we will focus on rectangular sumps which are especially convenient for clear water and cases, where large volume of water is desirable. They are commonly used in cooling systems of big power plants and big industrial plants. In the Figure 1.1 is typical design with dimensions based on pipe intake diameter D.

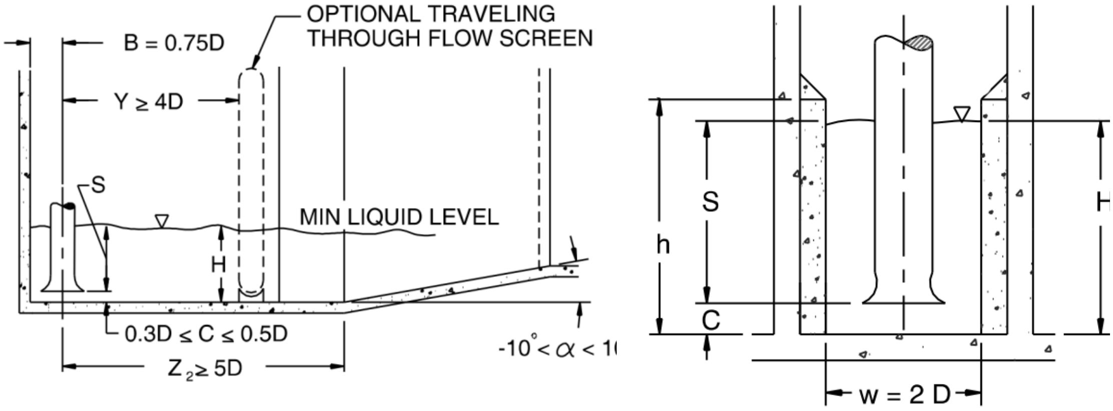

Figure 1.1 Rectangular wet pit sump. Figure taken from [ANSI/HI-9.8-1998]

The minimal submergence is based on Froude number defined as:

$$F_D = \frac{v}{\sqrt{g \cdot D}} \qquad [-] \qquad (2.7)$$

Where v is the mean velocity in pipe intake, g is the gravitational acceleration and D is the intake pipe diameter.

According to Hecker [3], the minimal submergence S, to prevent formation of surface vortices with gas core, is then calculated as:

$$\frac{S}{D} = 1 + 2.3 \cdot F_D \qquad [-] \qquad (2.7)$$

1.2 Reduction of Vorticity

In rectangular wet sump with multiple suction bells, unseparated form each other, recirculation zones are likely to occur. This may lead to increase of vorticity in the pump intakes. Vorticity can be reduced with flow splitters, also called M-type AVD (Anti Vortex Device), or with floor cones. Flow splitter is depicted in Figure 1.2. A practical application with numerical and experimental verification was done by Zhipeng Shi [4]. He has proven that AVD effectively mitigates subsurface vortices and vorticity is reduced in several orders of magnitude.

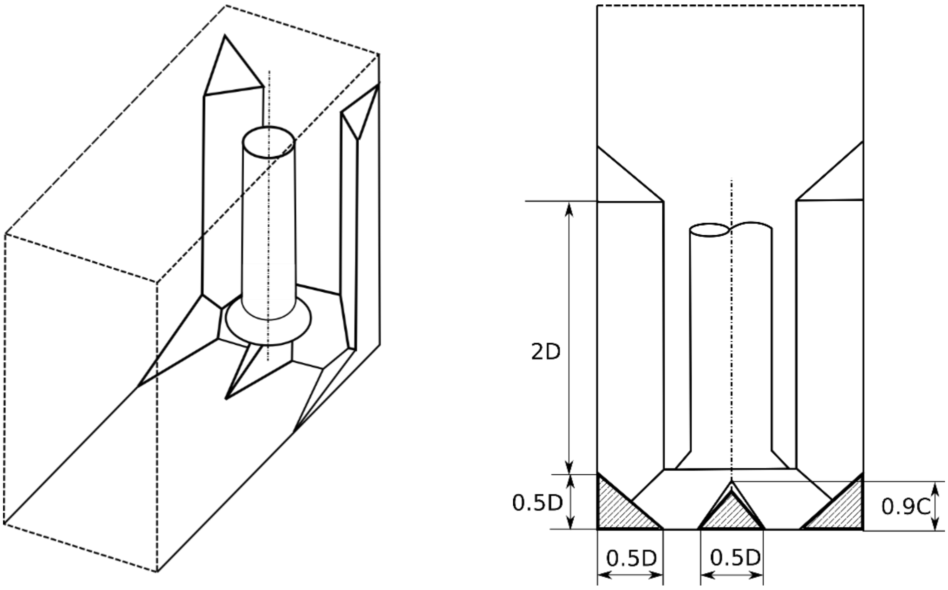

Figure 1.2 Anti vortex device. Figure taken from [1]

1.3 Measurement of Swirl [5]

During a model research, there is often a need to assess the intensity of swirling motion of fluid in bell mouth. Swirl is property of flow and it express the tangential component of flow velocity vector. As a consequence of conservation of angular momentum, in swirling flow circumferential velocity increase as the radius decrease. Close to the axis of rotation, viscous forces prevail and tangential velocity goes to zero.

The PIV (particle Image Velocimetry) is completely non-intrusive technique and is frequently used to measure the inflow conditions near bell mouth. However, it is still difficult to implement this technique to measure the flow patterns inside the bell.

In Pumping Station Design [1] a simple *swirl meter* was proposed (Figure 1.3), which measure swirl angle formed by non-uniform velocity field or flow unsteadiness. It should have 4 vanes with one painted red as a reference for counting number of revolution per time interval. It is also recommended to record the rotation on camera. Because of the unsteady nature of velocity field in bell mouth, swirl meter rotates discontinuously with periods of fast rotation and periods where it stops completely. Therefore, it must have small moment of inertia. Swirl angle is then defined as

$$\theta = \tan^{-1}\left(\frac{\pi \cdot d \cdot n}{v}\right) \qquad \text{[rad]} \qquad (2.7)$$

Where:

D [m] is the intake pipe diameter

n [1/s] are the revolutions per second of swirl meter

v [m/s] is mean axial velocity in a suction tube

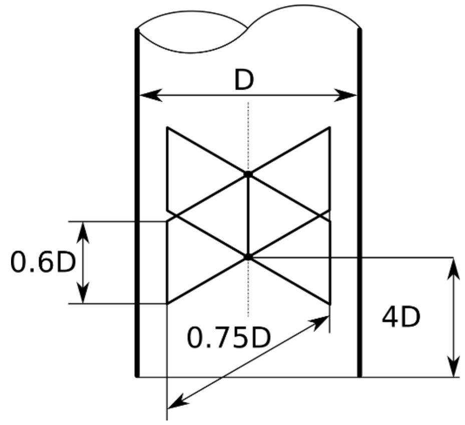

Figure 1.3 Design of a swirl meter proposed in [1]

1.4 Classification of Surface Vortices

Caruso et al. [6] performed extensive experimental investigation of free surface vortex. He used stage definition previously proposed by [7] and tried to characterize vortex evolutional stages by dimensionless numbers. Further he observed and described individual stages.

During *first stage*, no surface distortion is visible and the flow is characterized only by vorticity in the whole region above the outlet.

During *second stage* a small dimple, deeper than one millimeter, is visible on the water surface. Still, the vortex structure can be considered as symmetrical without bubbles detached from the tip. Gas core is wide and its position is stable over time. As the vortex grows, it passes to third stage.

During the *third stage* the gas core length is growing and getting thinner. Small bubbles start to detach from the tip and because the gas core is very sensitive to local flow conditions, it is not axial any more. This stage is unstable and the location and length of gas core strongly vary. It tends to pass quickly to the fourth stage.

During the *fourth stage*, the gas core reaches the pump intake. Air and impurities floating on the water surface are sucked to a pumping system which immediately loses performance. This stage is highly unstable and tends to pass quickly to lower stages.

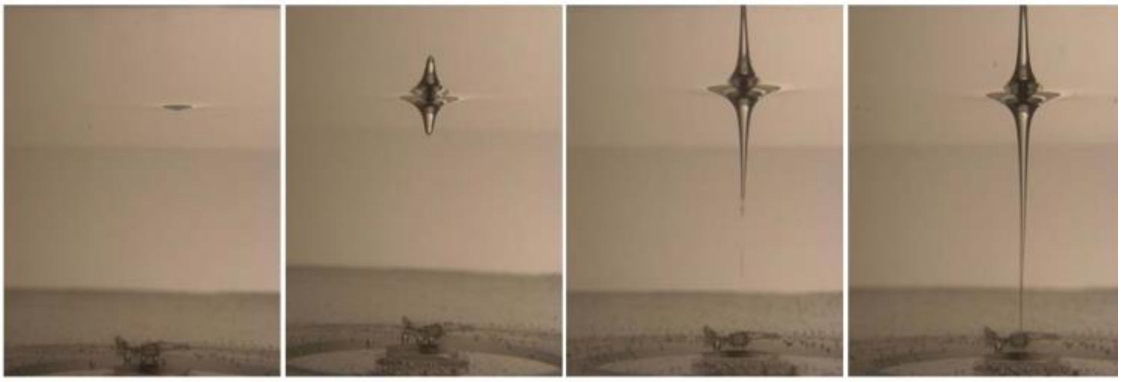

Figure 1.4 Stages of free surface vortex. From left to right: stage 1, stage 2, stage 3, stage 4. Figure taken from [6]

1.5 Acceptance criteria

The acceptance criteria of wet sump vary with the size, application and type of pumps. Nevertheless, following provisions are usually satisfactory:

- Surface vortices of type 3 or more and subsurface vortices of type 2 or more are unacceptable.
- Swirl angles should be less than 5 degrees, except that 6 degrees may be acceptable only if they occur less than 10% of the time or only during infrequent pump operating conditions
- The time-averaged velocities at specific points in the bell throat should be within 10% of the area average velocity. Time -varying fluctuations should have standard deviation of 10% or less.
- Air bubbles shall not be permitted to enter any pump intake during normal operation.

2 VORTEX MODELS

Vortices are often considered as regions with high vorticity, but this concept does not have to be always accurate and advantageous for post processing of numerical simulations. For example, flow near a wall has high vorticity, but it does not mean that a vortex exists. Hunt et al. [8] described vortical structures as a region containing second invariant of velocity gradient and low pressure. It was shown that the pressure criterion determines the vortex core, but it is often difficult to estimate appropriate pressure level and in many cases single pressure threshold is not enough to recognize all vortices.

2.1 Intuitive indication of vortices

The common indications of vortices are region with low pressure, spiraling streamlines and tube-like surface with constant vorticity.

Local pressure minimum: Vortices tends to have a local pressure minimum on the axis of a flow circulation. In this region, the centrifugal force is balanced by pressure force. However viscous effects in the vortex core might eliminate a pressure minimum in the vortex core. Another counter example might be unsteady strain rate in unsteady irrotational axisymmetric flow, which can cause local pressure minimum without swirling motion of the fluid.

Circular streamlines: This definition indicates vortices by closed or spiral pathlines. This approach requires lots of tedious work in postprocessing, when an engineer has to use streamlines and slices through a domain to determine, if any vortex is present. Furthermore, Hussain [9] proved that this criterion fails to indicate vortex if two or more vortices are moving with different speed in one reference frame, or vortices undergo tearing, pairing or breakdown. This criterion will certainly fail in turbulent flow region. Figure 2.1 shows streamlines of axisymmetric lamb vortex and Figure 2.2 shows early stage of vortex pairing in axisymmetric jet (arrows highlight centers of vortices).

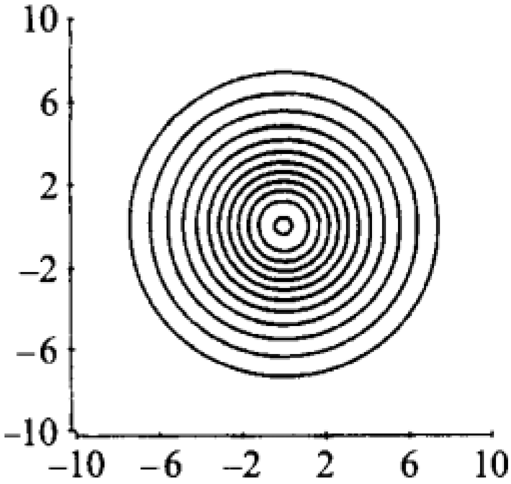

Figure 2.1 Streamlines of laminar Lamb vortex. [9]

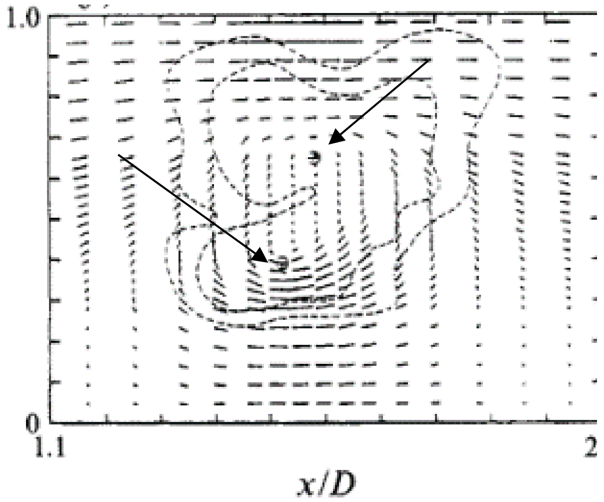

Figure 2.2 Velocity vectors distribution, dotted lines enclose vortical structure boundary region. [9]

Vorticity magnitude: As it was already mentioned, the vorticity criterion was widely used to visualize coherent structures. It was considered as a reliable criterion for free shear flows. However according to the investigation done by Hussain [9], this criterion even fails to identify coherent structures in free shear flows, where the background shear stress is comparable to vorticity magnitude within the vortex. Another counter example could be wall-bounded flow. It is characterized by shear and high values of vorticity in region near wall, but it does not contain swirling motion. Third counter example is vortex sheet, which is flow with velocity discontinuity between fluid layers. This flow exhibits theoretically infinite vorticity in discontinuity region, but again, it cannot be considered as a vortex.

All this counter examples shows the inconvenience of using mentioned intuitive characteristics of a vortex. To identify a vortex unambiguously, criterions based on magnitude of symmetry tensors S and Ω were proposed.

2.2 Q-criterion

Chong et al. [10] proposed to use complex eigenvalues of velocity gradient as an identification criterion of vortex core. The complex eigenvalues in a region results in closed or spiraling streamlines around any given point.

Hunt et al. [8] defined vortex as a positive value of second invariant of velocity gradient ∇u. This criterion was called Q-criterion and it expresses that in the vortex core the vorticity magnitude dominates over the strain-rate magnitude and it is a region with low pressure. This criterion is advantageous for post-processing, because Q is equal 0 near wall, where vorticity equals the strain rate.

$$Q = \frac{1}{2}[|\Omega|^2 - |S|^2] \tag{2.1}$$

Where S and Ω are the symmetrical and antisymmetrical part of velocity gradient and are defined as

$$S = \frac{1}{2}\left[\frac{\delta v_i}{\delta v_j} + \left(\frac{\delta v_i}{\delta v_j}\right)^T\right] = symm(\nabla U) \tag{2.2}$$

$$\Omega = \frac{1}{2}\left[\frac{\delta v_i}{\delta x_j} - \left(\frac{\delta v_i}{\delta x_j}\right)^T\right] = skew(\nabla U) \tag{2.3}$$

Where S is the rate-of strain tensor and expresses the speed of deformation by shear forces. Strain is 0 in inviscid flows, where the fluid acts like solid body e.g. in rotating frame. Ω is the vorticity tensor and expresses the microscopic rotation of the fluid in any given point. [11]

The magnitude of a tensor is then defined as a square root of a double inner product.

$$|S| = \sqrt{2S{:}S} \tag{2.4}$$

The vortical motion of a fluid exists, when $Q > 0$. For practical use the $Q = 0.1$ prove to be appropriate value for reliable identification of vortex region.

To see the exact implementation in CFD solver, the definition of lambda 2 was compared with the theory mentioned above.

```
        Q = 0.5*(sqr(tr(gradU)) - tr(((gradU)&(gradU))))
```

2.3 λ₂ criterion [12]

λ_2 criterion uses local pressure minimum as a starting point for its definition of a vortex, but it tries to eliminate the unambiguous identification of the vortex for the following model situations:

- Unsteady straining and pressure minimum without swirling motion.
- Viscous effects in the center of a vortex which can eliminates the pressure minimum in the center of the vortex.

The λ_2 criterion is derived from acceleration gradient a_{ij} of Navier-Stokes (NS) equation. To remove the effect of unsteady straining and viscosity, we simply omit these terms from the NS equation and receive:

$$a_{ij} = -\frac{1}{\rho}p_{ij} + \nu u_{ijkk} \tag{2.5}$$

Acceleration tensor can be decomposed into the symmetric and antisymmetric part.

$$a_{ij} = \underbrace{\left[\frac{DS_{ij}}{Dt} + \Omega_{ik}\Omega_{kj} + S_{ik}S_{kj}\right]}_{symmetrical\ part} + \underbrace{\left[\frac{D\Omega_{ij}}{Dt} + \Omega_{ik}S_{kj} + S_{ik}\Omega_{kj}\right]}_{antisymmetrical\ part} \tag{2.6}$$

The antisymmetrical part represents the vorticity transport equation. The symmetrical part of the acceleration vector will be the scope of our interest, because it contains the necessary pressure minimum criterion.

$$\frac{DS_{ij}}{Dt} + \Omega_{ik}\Omega_{kj} + S_{ik}S_{kj} - \nu S_{ij,kk} = -\frac{1}{\rho}p_{ij} \tag{2.7}$$

We will omit the material derivative $\frac{DS_{ij}}{Dt}$ and the viscous effects $\nu S_{ij,kk}$, because both terms do not contribute to the pressure drop due to fluid rotation (material derivative represents unsteady straining). Therefore, only the magnitude of $\Omega_{ik}\Omega_{kj} + S_{ik}S_{kj}$ contribute to the local pressure minimum due to the vortical motion of a fluid. Tensors $\Omega^2 + S^2$ has 3 real eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3$. Vortex core is then defined as a continuous region with negative λ_2 eigenvalue.

In OpenFoam the λ_2 criterion is defined as a function object that returns the second largest eigenvalue of the sum of the square symmetrical and antisymmetrical part of velocity gradient ∇u tensor.

```
const volTensorField SSplusWW
        (
            (symm(gradU) & symm(gradU))
          + (skew(gradU) & skew(gradU))
        );
Lambda2 = -eigenValues(SSplusWW)().component(vector::Y);
```

2.4 Comparison of Q criterion and λ₂ criterion

Comparison of Q and λ_2 criterion was done on laminar single phase simulation (chapter 7). Figure 2.3 compares iso-surfaces of Q and λ_2 for two iso values. It is clearly visible that both criterions yield very similar results. The only visible difference is that λ_2 produces more jagged surface. The presence of a vortex is defined by condition $Q > 0$ or $\lambda_2 > 0$ (vortical magnitude dominates over strain rate magnitude in vortex centre). However, it is also obvious that value of $\lambda_2 = 0$ and $Q = 0$ are not suitable for practical identification of dominant vortex and tends to visualize large vortical structures in the domain. These large structures are weak and do not represent any danger for the pumping system. Iso-values of 0.1 or greater proved to be reliable in identification only of the main vortex with potential to grow or evolve to next stage.

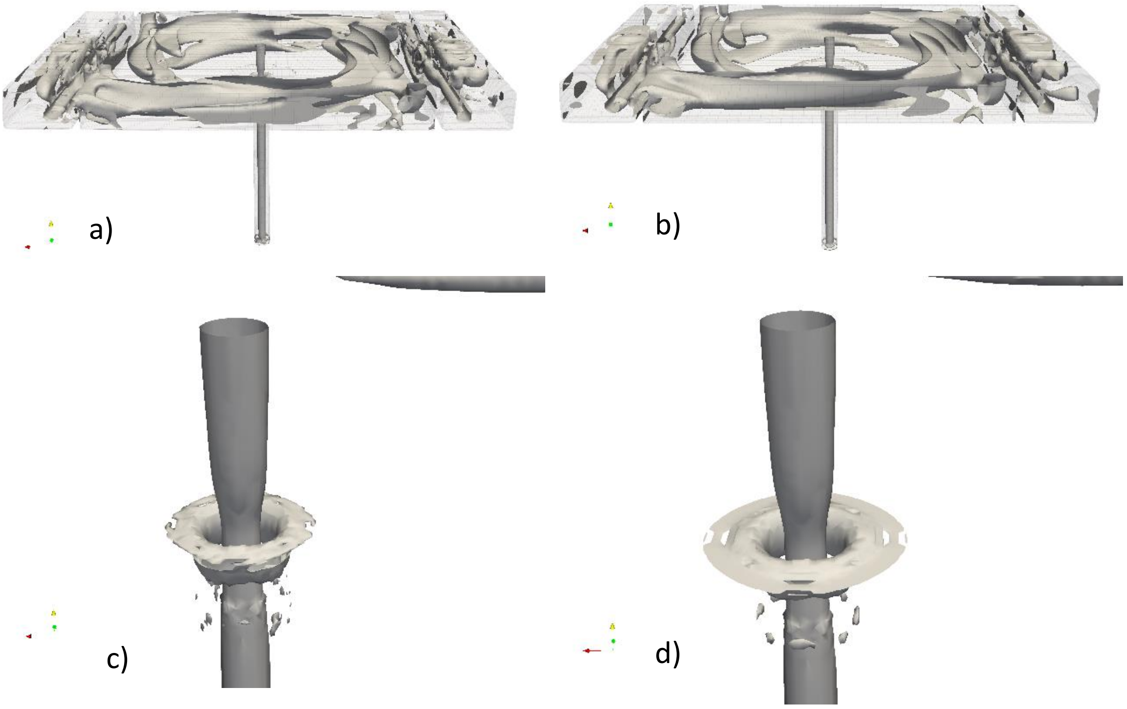

Figure 2.3 Comparison of lambda 2 and Q criterion: a) $\lambda_2 = 0$, b) $Q = 0$, c) $\lambda_2 = 0.1$, d) $Q = 0.1$

3 TWO PHASE SIMULATION IN ANSYS FLUENT

Because the flow conditions in wet sumps are usually site specific, traditionally they were studied on small-scale laboratory models. However, such experimental studies are always very expensive and surface vortices are unstable, sensitive to boundary conditions and disturbances on an intake. Therefore, several experimental studies were recently performed to explore the behaviour of the free surface vortex and validate numerical simulation, which would be able to reliable predict swirling flow in a pump intake and formation of free surface and subsurface vortex. [13] [14]

3.1 Determination of Appropriate Two Phase Model

To accurately capture the multiphase flow phenomena and receive physically correct results, it is vital to choose appropriate numerical model. ANSYS Theory Guide contains detailed guideline with many flow examples to help user to choose suitable multiphase model. ANSYS divided fluid flow into following regimes:

a) **Bubbly flows:** Discrete bubbles in a fluid**.**
b) **Droplet flow:** Discrete droplets in gas.
c) **Particle-laden-flows:** Discrete solid particles in a fluid.
d) **Sludge flow:** Large-scale bubbles in fluid.
e) **Annular flow:** Continuous liquid along core with gas core.
f) **Free surface flows:** Immiscible flows with sharp interface between them.

And available multiphase models with its application:

a) **Eulerian Model:** General-purpose model, simulation of bubbles and droplets in a continuous phase. Mixing and separation of phases. Eulerian treatment is used with momentum equation for every phase and one pressure equation is shared among phases. Compared to VOF model, Eulerian model is more computational intensive and uses considerably more memory.

b) **Discrete Phase Model:** Suitable for cyclone separators, internal combustion engines, spray dryers. It is state of the art model. It uses Eulerian-Lagrange treatment. The liquid is considered as a continuum, while the discrete phases are considered as particles and are tracked through the whole domain. Motion of particles is a result of forces of continuum phase acting on them. This model is not appropriate for flow, where the volume of discrete phase and particle-particle interaction can’t be neglected. For such cases, user ought to use Discrete Element Method which accounts particle-particle interaction.

c) **Mixture Model:** It is simplified Eulerian model capable of modelling any number of continuum and granular phases. Similarly to VOF model it uses single fluid approach and shares the momentum and pressure equation among all phases. Additionally, volume fraction equation is solved only for secondary phases. Compared to VOF model, the phases can be interpenetrating. This model is suitable for modelling non-Newtonian liquids, hydro-cyclones, cavitation in fuel injectors and centrifugal pumps.

It is good alternative for Eulerian model with much less computational effort for cases with many phases.

d) Volume of Fluid Model: Appropriate for immiscible fluids with sharp interface. This model can't be used, when the interface length is smaller than computational mesh.

This paper concerns with prediction of a gas core length in vortical structures, therefore main interest is put into tracking of water-gas interface. Water and air are immiscible fluids separated by sharp interface. According to the above stated recommendations Volume of Fluid (VOF) model was chosen for all the following simulations.

3.2 Volume of Fluid Model [15]

Volume of Fluid model is designed for two or more immiscible fluids, where main interest is on position of the phase interface. All material and fields properties are shared among phases. Thus, only one set of momentum equation is solved using mean values of density, kinematic viscosity and turbulence quantities.

Momentum equation can be written:

$$\rho\frac{\partial U}{\partial t}+\rho\frac{\partial u_i}{\partial x_j}u_j-\frac{\partial\left(\mu_i\partial u_j\right)}{\partial^2 x_i}-\rho g=-\frac{\partial p}{\partial x_i}-F_s \qquad (3.1)$$

Where

$\rho\frac{\partial U}{\partial t}$	is local acceleration
$\rho\frac{\partial u_i}{\partial x_i}u_j$	is convective acceleration
$\frac{\partial(\mu_i\partial u_j)}{\partial^2 x_i}$	is force from viscous stress
ρg	is force due to gravitation acceleration, or hydrostatic effect
$\frac{\partial p}{\partial x_i}$	is pressure gradient
F_s	is volume source term accounting for surface tension forces

Mean scalar value of density is calculated as:

$$\rho=\rho_{primary}+(1-\alpha)\cdot\rho_{secondary} \qquad (3.2)$$

Additionally, volume fraction α is tracked for every cell in a domain using volume fraction transport equation:

$$\frac{\delta\alpha}{\delta t}+\nabla\cdot(\alpha U)=0 \qquad (3.3)$$

Where $\alpha=1$ corresponds to a cell fully filled with primary phase and $\alpha=0$ corresponds to a cell fully filled with secondary phase. In VOF method it is not allowed to have void where no fluid is present. In case of more than two phases, for each phase a volume fraction variable is introduced. Sum over all volume fractions must add to unity.

Compared to the reality, where there is discontinuous jump of a density at interface, in VOF method we have region with mean density which vary between density of primary and secondary phase according to the equation (3.2). This means that sharpness of interface strongly depends on mesh resolution normal to the interface boundary and spatial discretization of volume fraction scalar value.

3.3 Implicit Body Forces

To improve convergence of a solution it is highly recommended to turn on implicit body forces option for flows

1. with big difference between phase densities and gravity acting on them
2. with large rotational acceleration. For example, turbomachinery, centrifugal pumps, hydrocyclones etc.

3.4 Implicit Time Discretization

Implicit interpolation scheme is used for obtaining face fluxes of cells and it uses current time step value of volume fraction α.

Implicit scheme can be used for both steady and transient calculations. The biggest advantage compared to the explicit scheme is that it can be run with significantly larger time steps (and Courant number) and it can be used with poor mesh quality. On the other hand, this scheme is not accurate for flows, where surface tension plays significant role which makes it inappropriate for modeling surface vortex with gas core. Implicit formulation also produces more numerical diffusion than explicit scheme which deteriorate the accuracy. [16]

3.5 Explicit Time Discretization

Explicit interpolation formulation is non-iterative scheme, which uses values of volume fraction from previous time step. This scheme can be used only with transient solvers. Compared to implicit scheme generally it has better accuracy and should be used, where surface tension plays important role. It captures well flows with big curvature and enables to use sharp interface capturing with Geo-reconstruct and CICSAM schemes. Drawbacks are poor convergence on skewed meshes, poor convergence on compressible cases and strict limitation of Courant number (time step) size.

During the evaluation phase, it was found out that to ensure numerical stability of OpenFoam simulation, the Courant number must be below 1, or preferably below 0.5.

3.6 Interpolation Near Interface

To track the phase interface, the transport equation for volume fraction α must be solved. On the interface, cells are partly filled with both phases. To accurately predict the phase interface and prevent numerical diffusion, following special treatments are applied to the volume fraction scalar field. The choice of suitable discretization scheme greatly influence sharpness and accuracy of the interface reconstruction and computational costs.

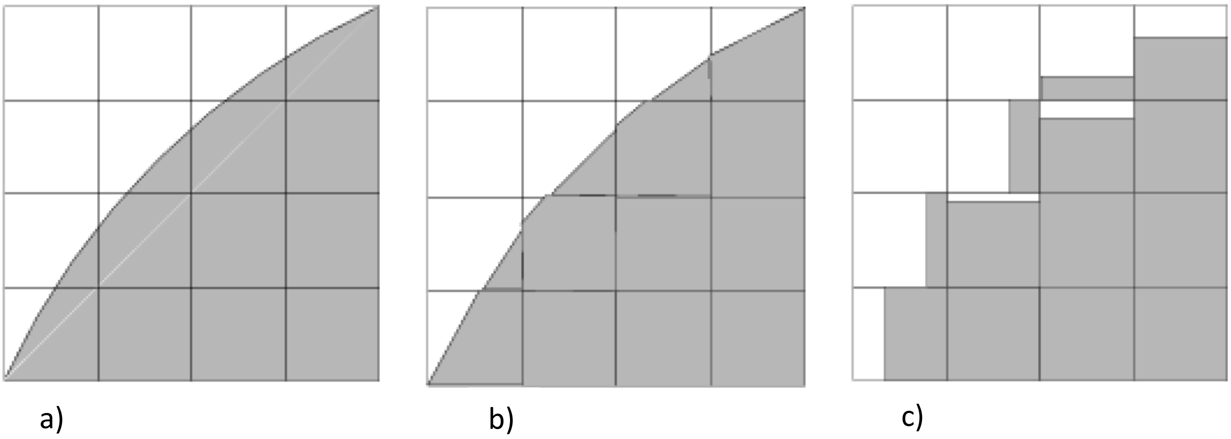

Figure 3.1 interpolation of volume fraction near interface, a) actual shape of interface, b) Geo-Reconstruct (piecewise-linear), c) Donor-Acceptor scheme. Figure taken from [16].

Geometric reconstruct: When a cell is close to the interface, the geo-reconstruction scheme is used. This scheme produces linear interface in every cell. It is the most accurate discretization scheme, but it is also the most computationally expensive one. Generally, it can be used for unstructured and poor quality meshes.

The solver first solves the linear phase interface related to every cell center from phase fraction. Then calculates the mass advection through cell boundaries from phase fraction and velocity field. Last step is to calculate the volume fraction from equilibrium of fluxes.

Compressible interface Capturing Scheme for Arbitrary Meshes (CICSAM): CICSAM is particularly suitable for cases, where the density ratio between primary and secondary phase is larger than 1000 and delivers as sharp interface as geometric reconstruction with less computational effort. All these properties make CICSAM scheme first choice in simulating water-air interaction with VOF. It is available only with time discretization.

Compressive: Compressible scheme is second order accuracy based on slope limiter function:

$$\phi_f = \phi_d + \beta \cdot \nabla\phi_d \tag{3.4}$$

Where

ϕ_f is the face VOF value

ϕ_d is the donor cell VOF value

β is the slope limiter function

$\nabla\phi_d$ is gradient of the donor cell VOF value

Compressive scheme can be used for both implicit and explicit interpolation scheme. The slope limiter function acquires value between 0 and 2, where

β=0 corresponds to first order upwind scheme β=1 corresponds to second order reconstruction

β=2 corresponds the compressible scheme

Values between $0 < \beta < 1$ and $0 < \beta < 1$ means blended function between upwind, second order scheme and compressible scheme.

Modified HRIC: scheme is based on the donor acceptor method. At the interface one cell is determined as donor and neighbouring cell is specified as acceptor. The amount of fluid transported through common face is limited by minimum volume of donor cell filled with primary phase ($\alpha = 1$) or by minimum volume of acceptor cell filled with secondary phase ($\alpha = 0$). The advection though interface is obtained from pure upwind scheme. The orientation of interface can be only either horizontal or vertical and it depends on direction of volume fraction gradient in a cell. This scheme can be used only for unsteady simulations with hexahedral meshed. Comparison of Geometric reconstruction scheme and Donnor-Acceptor schemes is demonstrated on simple transport of scalar value in uniform velocity field. It’s clearly visible that in case of the Modified HRIC (Donnor-Acceptor) scheme the interface shape is heavily distorted due to the interface reconstruction inaccuracies.

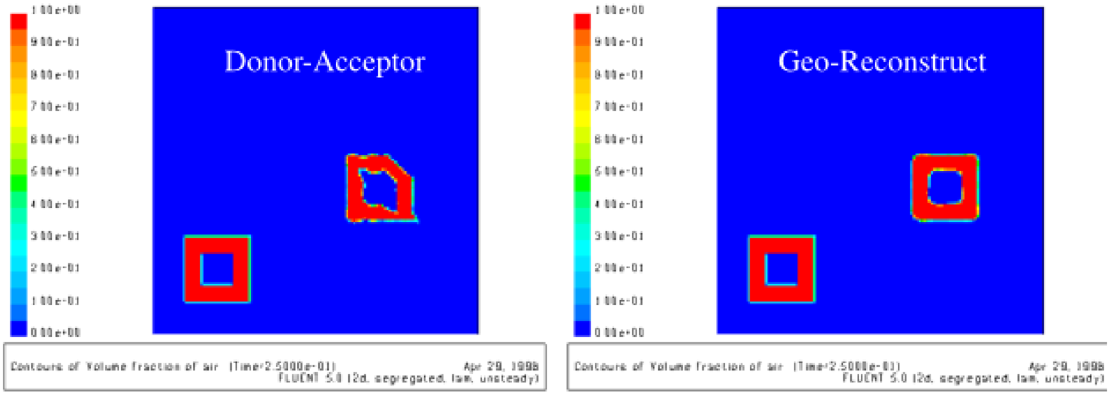

Figure 3.2 transport of scalar value with different interpolation schemes. Figure taken from [17].

3.7 Modeling of Surface Tension

Surface tension is result of attracting forces between molecules and it forms the shape of interface. Surface tension compresses the interface and it increases the pressure on concave side of the surface. When two phases are separated by sharp interface, surface tension tries to minimize the free energy of the interface. Fluent implements continuum surface force (CSF) model which adds additional volume source term into momentum equation. The source term can be expressed as the pressure jump across the surface and can be write as: [16]

$$F_{vol} = \frac{\sigma_{ij}(\rho\kappa_i\nabla\alpha)}{\frac{1}{2}(\rho_i + \rho_j)} \tag{3.5}$$

Where:

κ [-] is divergence of surface normal vector $\kappa = \nabla \cdot \vec{n}$

ρ [kg/m] is mean value of density calculated from equation (3.2)

σ [N/m] is surface tension

$\nabla\alpha$ [-] is gradient of phase fraction

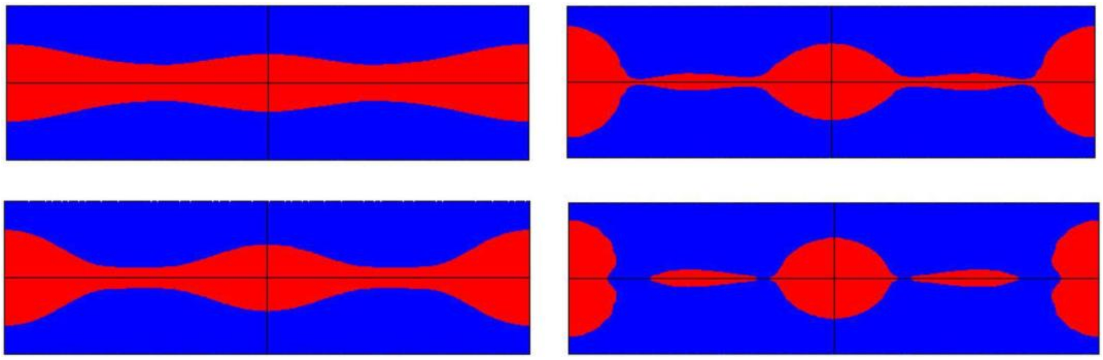

Figure 3.3 Effects of surface tension in a case with zero gravity. Figure taken from [16].

The effect of surface tension is best visible on test case with zero gravitation. Figure 3.3 demonstrates such an example. The top left image depicts the initial case, where distortion from rectangular shape accelerates the formation of droplets. Only one quarter of the image was simulated. [17]

Zhi-peng et al. [4] highlighted the importance of surface tension on formation of surface vortex. It was estimated that surface tension has no effect when Weber number is greater than 120. [4] Weber number can be calculated as:

$$We = \frac{\rho \cdot v^2 \cdot D}{\sigma} \tag{3.6}$$

Where:

v [m/s] pump intake velocity

D [m] is diameter of a bell mouth

σ [N/m] is surface tension

4 NUMERICAL SIMULATION IN OPENFOAM [18] [19]

Due to the extremely high computational costs of CFD analysis of free surface vortices, it is practically impossible to perform such a job on a personal computer or small number of CPU cores. In case of ANSYS Fluent, users must buy additional parallel licenses and allocate great deal of them for weeks or months. This makes the simulation extensively expensive or it slows down other research groups. Therefore, it was decided to validate free CFD software as an alternative to commercial solver. As this kind of simulation is aiming at big HPCs, OpenFOAM CFD package has been chosen, because it is available on most of scientific computational clusters.

OpenFOAM is free, open source software which offers environment to solve engineering problems in fluid flow. Like any other commercial CFD software, OpenFoam is based on finite volume method and It contains utilities for handling the whole CFD workflow i.e. pre-processing, solving and post-processing. The main idea is to separate complicated software into much smaller and relatively simple programs, which are executed separately from controlling script and are relatively easy to use. This is beneficial when running cases in batch or analyzing geometrically similar cases. For pre-processing Users are encouraged to use blockMesh and snappyHexMesh for meshing of the computational domain. OpenFoam contains broad list of CFD solvers with big difference between their complexity and capabilities. To choose the appropriate solver for given task, each solver contains its description in */opt/openfoam4/applications/*. For visualization of results OpenFoam uses implementation of Paraview called ParaFoam. The differences between Paraview and ParaFoam are minor. Paraview enable user to postprocess decomposed results and ParaFoam handles better work with *faceZones* and *cellZones* inside computational domain. The workflow presented here is modular and it is possible to replace every stage with commercial software. For example snappyHexMesh can be replaced with ANSA, cfMesh or Fluent Meshing and Paraview can be replaced by EnSight. Equally it is possible to export mesh from OpenFoam case and use commercial solver as ANSYS Fluent.

Compared to commercial codes, OpenFoam offers several advantages. It is free of charge and the source code is open. Therefore, every user can check an implementation of a theoretical model directly in the code and gives the user better understanding how a solver works. User may modify existing solvers and tailor its capabilities to his specific needs. There is also active community which provides support, publish useful pieces of code, new solvers and turbulence models and scripts for handling specific tasks, e.g. measuring height of water surface.

To solve multiphase flow problems, OpenFoam offers several solvers which incorporate principally similar models as ANSYS Fluent, or any other commercial CFD software. These models were already discussed in previous chapter thus only brief description of available solvers in OpenFoam are outlined below.

- **InterFoam family:** is incompressible, isothermal solver for 2 immiscible fluids. It utilizes volume of fluid approach and explicit time discretization.
    - **InterDyMFoam:** is extension of interFoam with capabilities of dynamic mesh. User can set adaptive mesh refinement according to any scalar value. Typical application could be automatic mesh refinement of free

  surface. This can be achieved by setting grid refinement condition to $\alpha = 0.5$.
  - **CompressibleInterFoam:** solves energy equation and consider compressible phases. Typical application could be high velocity flows, where compressibility of gas phase must be considered.
  - **CompressibleMultiphaseInterFoam:** introduces any number of compressible phases with energy equation.
  - **InterCondensingEvaporatingFoam:** introduces non-isothermal phase-change (evaporation-condensation) between fluids.
  - **InterPhaseChangeFoam** which involves isothermal phase-change (cavitation).
  - **MPPICInterFoam** which includes MRF (multiple frame of reference) and discrete particles modelling.
- **CavitatingFoam:** is transient cavitation code based on the homogenous equilibrium model. Gas fraction is compressible.
- **MultiphaseEulerFoam:** fully corresponds to Euler model implemented in Fluent [3.1].
- **PotentialFreeSurfaceFoam:** Incompressible, single phase solver with wave height field for approximation of free surface.
- **ReactingEulerFoam:** is solver for 2 or more compressible phases with shared pressure equation and separate momentum equation for secondary phases. The phase model represents multiple species and in-phase reactions with heat and mass transfer.
- **TwoLiquidMixingFoam:** is solver for mixing of 2 incompressible phases. It corresponds to Mixing model implemented in Fluent [3.1].

4.1 Using interDyMFoam

From this list, interDyMFoam has been chosen as the most suitable solver for given task. As it was already mentioned, it contains all the features of basic interFoam solver and extend its capabilities further to moving mesh, simulation of floating solid body e.g. 6DOF, or automatic mesh refinement and mesh regeneration. The automatic mesh refinement is particularly useful for maintaining relatively sharp interface in vortex gas core, which is often not stable and its position and length varied over time.

To better understand following chapters, reader should have at least basic knowledge how to use OpenFOAM (OF). Therefore, following lines will explain the file structure of OF and how to set a new case.

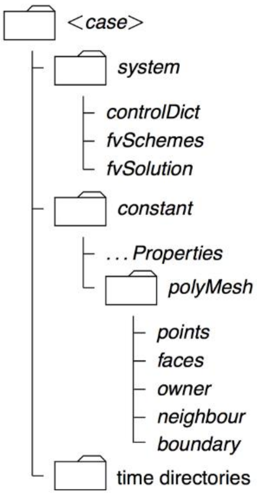

Figure 4.1 Case structure in OpenFoam

OpenFoam, unlike to Fluent, saves the case and calculated data to template set of folders with template dictionaries. To set a new case, user should copy existing case, which has similar physics, to working folder and modify input dictionaries to fulfill given task.

0/ In this folder boundary conditions and initial initialization is stored. Every quantity used during simulation is stored in separate text file. For example, typical multiphase calculation with k-ω turbulence model contains following files:

- **alpha.** denotes to phase fraction. *alpha.water* then denotes to primary fraction named water. This specie must be defined in "*constant/transportProperties*" file.
- **k** is kinetic turbulent energy
- **omega** is specific turbulent dissipation
- **nut** is eddy viscosity
- **p_rgh** is operating pressure minus hydrostatic pressure $p_{rgh} = p - \rho g h$
- **U** is velocity vector field

constant/ folder contains *polyMesh/* folder with mesh and cell sets. Further there are dictionaries to define material properties (*transportProperties)* and choice of turbulence model (turbulenceProperties).

system/ folder contains dictionaries, which controls solver e.g. *fvScheme* with definition of discretization schemes, *fvSolution* with definition of linear solvers and controlling parameters of SIMPLE/PIMPLE loops and *controlDict* with definition of time step length, number of iterations and run time functions.

All commands, which are necessary to be executed to launch the simulation are usually stored in *Allrun* script. User is encouraged to consult the script and find more information about all commands stored there. Typically, it is necessary to create background mesh with *blockMesh*, initialized the simulation with *setFields*, or decompose the case to run in parallel with *decomposePar*. After all preparatory steps, the simulation is launched by entering the name of a solver to command line e. g. *interDyMFoam*. To launch the simulation in parallel, it is necessary to decompose the case with *decomposePar* and then start the solver with *mpi* e. g. for simulation on 8 cores run *"mpirun -np 8 interDyMFoam -parallel"*.

4.2 Stability of VOF Solvers

The stability of VOF solvers is generally limited by global Courant number and Courant number of alpha field at the interface.

Global Co should be smaller or equal to 1 for explicit formulation and in range of 1-5 for implicit formulation. In case of higher global Co it is usually necessary to use more time step sub-iteration to maintain simulation stability and the benefit of larger time step does not always result in faster time advancement.

Courant number of alpha field at the interface (*AlphaCo)* basically means, how many times the interface travels through given cell in one time step. To maintain stability, *AlphaCo* must be below 0.25. Because the alpha transport equation is relatively cheap compared to pressure and turbulence equations, it is always better to use alpha sub-cycles, which solves the alpha equation multiple times with smaller time step. So, in order not to constrain the time step length by condition *AlphaCo*, user should set *maxCo* to *1*, *maxAlphaCo* to *1* in *controlDict* and *nAlphaSubCycles 4* in *fvScheme*. Now the alpha equation will be solved 4 times with maximum Courant number per cycle 0.25.

4.3 Dynamic mesh refinement [20]

OpenFoam offers a dynamic library which handles the automatic mesh refinement during calculation. A good example of an application of automatic refinement is the *damBreakWithObstacle* tutorial located in *$FOAM_TUTORIALS/multiphase/interDyMFoam/ras/* folder. The parameters and solvers for dynamic mesh are in *constant/dynamicMeshDict*. Solver for mesh refinement is *dynamicRefineFvMesh*.

For the dynamic mesh refinement, the following parameters should be set according to the Table 4.1 settings of dynamicMeshDictTable 4.1. The user should be very careful with the parameters refinementInterval and maxRefinement. Setting the first parameter too low and solver will spend most of its time in mesh refinement loop. Setting the latter parameter to high values will cause excessive number of cells which slow down the simulation even more. Bear in mind that small cells lead to higher Courant numbers and therefore the need of reducing the time step length.

During the refinement process, each cell is divided into 8 smaller cells. First, original face is split into 4 faces. Then internal faces are created and new owners and neighboring faces are matched. In the last step, a value from the original cell is mapped into the newly created cells. Therefore, no interpolation take place during mesh refinement.

During the unrefinement stage, the points created during the refinement stage are removed and their connected faces are recreated. Then the fields are mapped to the coarse mesh, where the values from 8 fine cells are averaged into the original coarse cell.

The new mesh is stored in new time step folder, or inside processors folder in case of parallel computing. Therefore, the original mesh is kept untouched in constant/polyMesh. If there is need to work with the refined mesh further, the easiest way is to reconstruct it using command *reconstructParMesh -constant -sourceTime <time>.* Keep in mind that the original mesh will be deleted and it cannot be recovered back.

Table 4.1 settings of dynamicMeshDict

Parameter	Value	Description
dynamicFvMesh	dynamicRefineFvMesh	Solver.
refineInterval	2e+4	Mesh refinement after number of outer iterations.
field	alpha.water	Refinement based on scalar field. In this case volume fraction of secondary phase called water.
lowerRefinementLevel	0.001	When the field value is between the lower and upper bound, the refinement is triggered. When the value is out of the bounds, the coarsening is triggered.
upperRefineLevel	0.999	
maxRefinement	1	The maximal refinement level achieved during the refinement stage. Works only with hexahedral cells, where one cell is divided into 4 smaller cells. The final size of the cell decrease with the two to the power of maxRefinement.
unrefineLevel	1	Mesh coarsening. It is applied only on refined cells in previous step, when the field value is out of the bounding values.
nBufferLayers	2	The number of cell layers between the individual refinement levels. It is used to smoothen the transition region of refinement.
maxCells	2e6	Controls the maximum number of cells in your mesh. If this value is exceeded, the refinements terminates.
dumpLevel	False	Debugging option to visualize the refinement distribution as a volume scalar field.

However, dynamicRefineFvMesh will refine the water-air interface in the whole domain. This is not favorable in case, where we require sharp interface only in small region and interface refinement in the entire domain results in unwanted and useless increase of cell counts. Typical example could be the analysis of the vortex gas core length, where we need sharp interface only in vortex vicinity to precisely capture its shape, position and length. Mesh refinement in any other parts of the flow domain is undesirable and leads only to higher computational cost without any added value.

So, to achieve the mesh modification only in defined region, a modification of class dynamicRefinementFvMesh is necessary.

To overcome the modification of source code, an alternative mesh refinement procedure with refineMesh tool was proposed in chapter 4.3.1. However, this procedure is no longer dynamic so it should be used only as a temporary solution, till proper modification of dynamicRefinementFvMesh library will be done.

Simultaneously, transient VOF simulation with small time step and long simulation time brings us to situation, where we must fine tune the mesh to have minimum number of cells, uniform Courant number distribution without peaks and still obtain accurate results.

4.3.1 Mesh refinement with refineMesh tool

To do this, we will use *topoSet* command to make two cell sets based on fraction volume and geometrical restriction e.g. bounding box or bounding cylinder. It is necessary to copy topoSetDict to system folder from any tutorial folder.

To define the cell set containing the air-water interface, use fieldToCell source and define fieldName entry according to your secondary specie name (in most cases alpha.water). Then define the maximal and minimal bounding values of volume fraction, when the cells are added to the cell set. For reliable identification of the interface, it is recommended to use 0.01 as a minimum and 0.99 as a maximum. To create cell set containing cells in the region close to an orifice, use cylinderToCell source and define two points on the cylinder axis (p1 and p2) and radius of the cylinder.

To obtain cell set containing cells only on the water interface and inside predefined cylinder, perform Boolean intersection of the cell sets. However, OpenFoam does not contain such a tool in the standard distribution, so a custom python script has been created. In OpenFoam, the cell set is defined by its header, number of selected cells in the list, brackets and list of cells. The script extracts data form the header, perform data intersection and recreate new cell set called vortexRefSet. The function matchSets.py accepts two parameters which are the names of sell sets, on which an intersection will be done. These parameters are preconfigured and can be changed in the header section shown below.

```
#************************USER-ADJUSTABLE-PRAMETERS*****************
#Global variables

firstSetName = "cylinderRegion"
secondSetName= "waterSurfaceRegion"
relPath = "constant/polyMesh/sets/"

#******************************************************************
```

The last step is to refine the mesh using refineMesh. The tool needs additional dictionary in the system folder called refineMeshDict. There is no need to change anything inside except the “set” entry, which should be set to “set vortexRefSet”.

After executing command refineMesh a new mesh is created, but the fields stored from the previous mesh are nor mapped, neither interpolated! Because of that, the whole workflow becomes more complicated and additional manipulation with case and tools is necessary.

The whole workflow could be outlined as follows:

1. Terminate solver
2. Duplicate the case with “*cp -r 01_baseCase 02_refinedCase*”
3. Go to case 02_refinedCase “*cd 02_refinedCase”*
4. Reconstruct the latest time step with “*reconstructPar -latestTime”*
5. Create the cell sets with “*topoSet*”
6. Make the sell sets intersection *“python matchSets.py”*
7. Refine the mesh with “*refineMesh -overwrite -constant”*

8. Map fields to the new mesh: “*mapFields ../01_baseCase -lastestTime -consistent -parallelSource”*
9. Decompose mesh for parallel running “*decomposePar”*
10. Change controlDict with *“changeDictionary”* or with “*sed*”
11. Rerun the solver

This process can be easily automated with bash script, which is called once per given simulation time interval.

Figure 4.2 a) shows mesh after static refinement. Unfortunately, the refinement produced several non-orthogonal cells and worsen overall mesh quality. Moreover, the mesh refinement is very thin and due to the instability of a vortex gas core, the interface gets quickly out of the refined domain. Which means that user should perform such a refinement procedure frequently.

Figure 4.2 b) shows volume fraction contours of a case that has been run for additional 100 s (simulation time). The interface in vortex gas core got out of the refined domain.

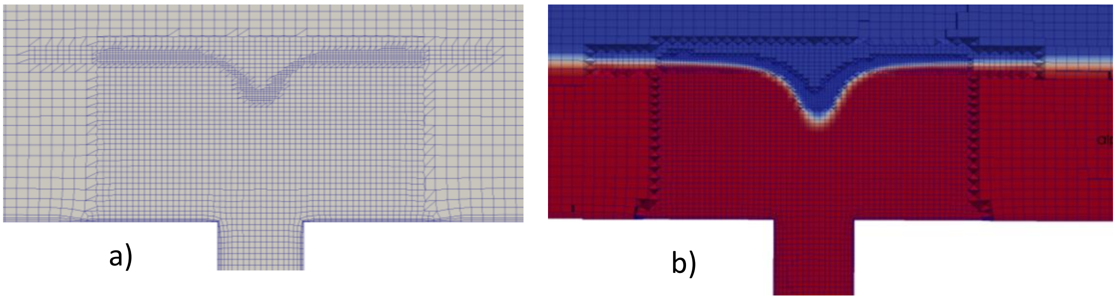

Figure 4.2 Static mesh refinement: b) with contour of volume fraction. Test case from chapter 7.

Note the difference in mapping fields between dynamic and static mesh refinement. As it was already mentioned, during the dynamic mesh refinement no actual field interpolation takes place. The same value is assigned to all 8 cells that emerge from splitting a cell. Furthermore, during the unrefinement stage, an average from 8 previously refined cells is assigned to one coarsen cell. In contrary, during the static refinement the “*mapFields”* utility is always used. It utilizes an inverse interpolation scheme, which is first order accuracy and mainly in region with strong gradient It can cause errors and simulation instabilities.

5 TURBULENCE MODELLING

Turbulence modelling is key aspect of all CFD simulations and appropriate choice of turbulence model determine final accuracy of the simulation. Ansar et al. [14] performed inviscid simulation of single- and dual-pump intake and achieve good agreement with measurement. He stated that inviscid simulation is appropriate approximation of real flow in the core and vicinity of the vortex. Škerlavaj et al. [21] performed extensive study to determine the best turbulence model and size of the time step for modelling swirling flow in wet sump. He used ANSYS CFX to simulate cylindrical vessel, experimentally tested by Monji et. al. [22], and he confirmed good agreement of laminar simulation with experimental data. Further he determined k-ω SST-CC (RANS) model and Scale Adaptive Simulation (SRM, chapter 5.2) model as the most appropriate choice in terms of computational demand, numerical stability and accuracy. He emphasizes the importance of using the curvature correction option to accurately capture the flow in vortex core. Furthermore, the Reynolds Stress Model (RSM, chapter 5.3) is often recommended for swirling flow with strong stream curvature and therefore should be investigated as well.

5.1 Curvature correction

According to Shur et al. [23], simple linear eddy-viscosity models have problems to predict strong flow curvature in free stream region. Because RANS models are usually the model of the first choice, naturally there was effort to improve the model accuracy and sensitize RANS models to strong flow curvature. In this paper, curvature correction proposed by Shur et al. [23] is presented.

Basically, curvature correction modifies the production term of eddy viscosity to limit the production of eddy-viscosity in region with strong flow curvature.

ANSYS Fluent offers the curvature correction option in a standard distribution and it can be used with following models:

- Spalar Allmaras
- k-ε family models (Standard, RNG, Realizable)
- k-ω family models (Standard, SST, BSL)
- Scale-Adaptive Simulation and Detached Eddy Simulation models

Even though the CC is available for all above listed models, the user should consult the use with ANSYS Theory Guide. For example, curvature correction is generally not recommended for Realizable and RNG k-ε models, because they already have some sensitization in their formulation.

There are many different approaches how the sensitize the model to streamline curvature. ANSYS Fluent uses the Spalart-Shur approach. It uses the symmetry part of velocity gradient to track the principal axis of strain rate tensor. From this assumption, an empirical equation was derived, which limits the production term of eddy viscosity.

$$f_{r1}(r^*,\tilde{r}) = (1+c_{r1})\frac{2r^*}{1+r^*}\left[1-c_{r3}\tan^{-1}(c_{r2}\tilde{r})\right]-c_{r1} \tag{5.1}$$

Where r^* and $\tilde{r}$ are nondimensional quantities defined as

$$r^* = \frac{S}{\Omega} \tag{5.2}$$

$$\tilde{r} = 2\omega_{ij}S_{jk}\left(\frac{DS_{ij}}{Dt} + \left(\epsilon_{imn}S_{jn} + \epsilon_{jmn}S_{in}\right)\Omega_m\right)\frac{1}{D^4} \tag{5.3}$$

Where the constants are set to $c_{r1} = 1.0$, $c_{r2} = 2.0$, $c_{r3} = 1.0$ and the value D is set to $D^2 = \max(S^2, 0{,}09\omega^2)$.

The eddy production term is multiplied by the value f_{r1}, which is bound to lower value of 0 and maximal value of 1.25. The lower value corresponds to strong convex streamline curvature, where the production term is fully suppressed. The maximal value corresponds to strong concave curvature, where the turbulent production is enhanced. The lower and upper bound is introduced because of the numerical stability. The upper bound value was fine-tuned on numerous cases such as U turn in hydro cyclone or wing tip vortex to obtain the best performance of curvature correction. [23]

Standard distribution of OpenFoam from OpenFOAM foundation does not include the curvature correction in any of the available turbulence models. However recently there was an effort to reformulate the standard kOmegaSST model proposed by Menter et. al [24]. This modified turbulence model was developed under supervision of TotalSim Ltd and made public under GitHub distribution. This modified model was developed for OpenFOAM 2.2.0 and received new name TkOmegaSSTCC. The behavior of this new model was tested by Martinez [25]. He tested the behavior of the new model on two test cases, NASA bump and on wingtip vortex flow. He compared the results to experiments and to simulations performed in commercial code. He observed 30% improvement of vortex decay in case of commercial code. In case of OpenFOAM, the improvement of vortex decay was around 20%.

Unfortunately, the model proved to be incompatible with the standard distribution from OpenFoam Foundation. Furthermore, from version 3.0 the whole concept of turbulence modelling was re-written and now is template based. It means that there is only one formulation of turbulence model for both incompressible and compressible flow. This concept brings clearly huge advantage in simplified maintenance of the code, as there is only one version of a mode.

5.1.1 Performance of Curvature Correction in ANSYS Fluent

To test the performance of realizable k-ε model, cylindrical vessel measured by Monji et al. [22] was duplicated using single phase approach. The realizable k-ε model is well adopted in Fluent and it is frequently used in industrial application because of its numerical stability, small computational demands and consistent performance over broad range of flow regimes. Curvature correction was used to check, if it will help to reduce numerical dissipation. The results were compared to experimental data and laminar simulation, which should presumably capture well the laminar vortex core.

In Figure 5.1 it is apparent that realizable k-ε model with curvature correction failed to predict the downward velocity. This would also lead to wrong prediction of vortex gas core length, which is in close correlation to water surface depression. Therefore, k-ε turbulence models with or without curvature correction cannot be recommended for surface vortex modeling. Nevertheless, it has been proven that laminar simulation well captures the real flow condition in the vortex core and should be used as the model of the first choice.

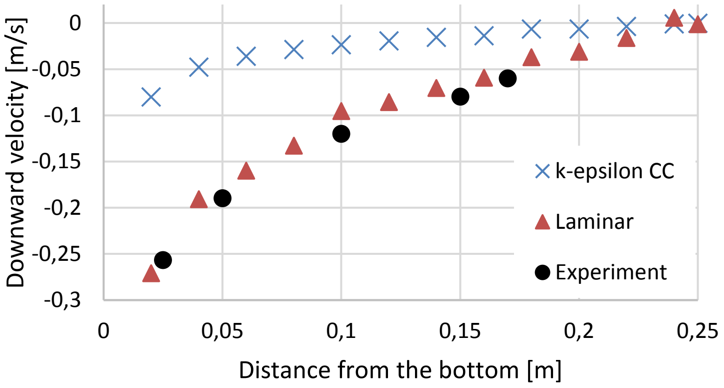

Figure 5.1 Downward component of velocity vector, flow rate 4.5 l/min, experimental data from **[22]**

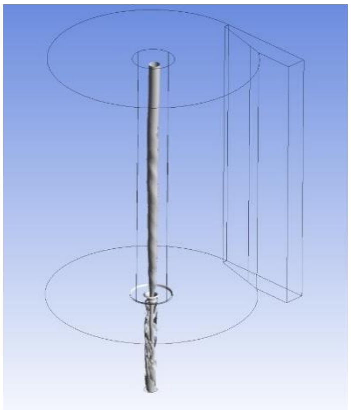

Figure 5.2 Visualization of vortex with Q-criterion, Q=0.1, Laminar

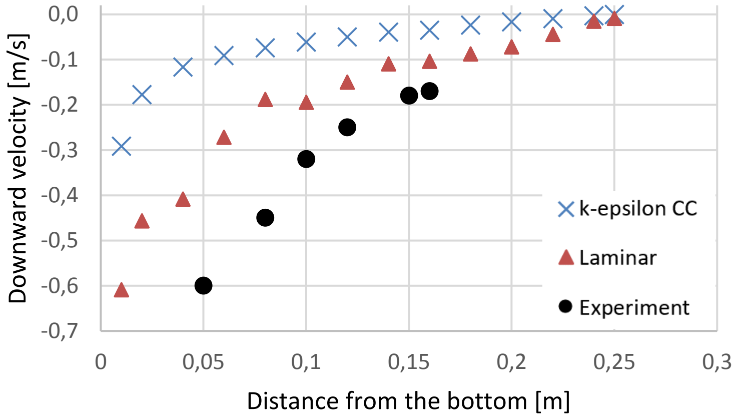

Figure 5.3 Downward component of velocity vector, flow rate 7.8 l/min, test case from [22]

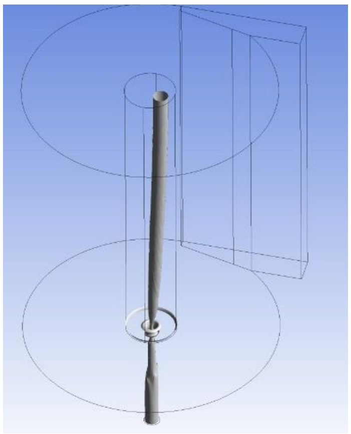

Figure 5.4 Visualization of vortex with Q-criterion, Q=0.1, k-epsilon-CC

5.2 Scale-Resolving Simulation methods [26]

Conventional turbulence models based on Reynolds-Averaged Navier-Stokes equation (RANS) has proven to be accurate and reliable in wall-bounded flows. On the other hand, their performance in free shear flow is already not that consistent. Scale-Resolving Simulation is a method that resolves large turbulence structures in LES like manner and provides additional information to the user about unsteady turbulence behavior. Resolving the largest turbulent structures is important mainly in cases such aeroacoustics, unsteady cooling, or swirl combustors, where the turbulence has big influence on generated noise, heat transfer between thermal boundary layer and flow core, or in cases, where the rate of chemical reaction is governed by turbulent mixing. The scale-resolving simulation (SRS) can also bring significant accuracy improvement in flows, where some flow phenomena are induced by weak unsteady turbulence. However, the SRS methods are challenging to use. It is a must to pay attention to proper numerical settings, meshing and model selection.

The most common SRS models and their brief description is outlined below:

- **Scale-Adaptive Simulation (SAS)** is two equation turbulence model. SAS stays in RANS mode in flows near wall and switches to LES like mode in large unsteady separation zones. In RANS models, the turbulence length scale is often modeled only with single variable k. In SAS model the turbulence length scale is based on exact transport equation. Additionally, it includes the von Karman length scale $L_{vK} = \kappa \left|\frac{U'}{U''}\right|$. In unstable zones, the L_{vK} is decreased which consequently leads to reduction of eddy viscosity and LES like solution. Generally, the SAS model requires time step and grid resolution and fine enough to resolve the largest turbulence structures in the flow. But in regions, where the grid is too coarse to resolve the turbulence or the time step is longer than frequency of the unsteadiness, SAS will safely fall through to RANS mode and accuracy of RANS simulation is maintained.

- **Detached Eddy Simulation (DES)** is hybrid formulation between LES and RANS. It switches between them based on mesh resolution. The switching function can be defined as follows:

$$C_{DES}\Delta_{max} > L_t \rightarrow RANS\ mode \tag{5.4}$$

$$C_{DES}\Delta_{max} < L_t \rightarrow LES\ mode \tag{5.5}$$

  Where:
  L_t is turbulent length scale
  Δ_{max} is the maximal cell edge length
  C_{DES} is constant

  DES model usually operates fully in RANS mode in wall bounded region and in LES mode in free shear region. The mesh resolution and length of the time step must be of LES quality. It is important to bear in mind that the transition from RANS to LES can happen already in boundary layer, where the mesh resolution activates the LES mode. In such cases, the eddy viscosity is suddenly decreased, which leads to Grid-Induced Separation. To mitigate such an unwanted behavior, a shilling function was introduced to protect the boundary layer from switching to LES mode. The new model was called **Delayed Detached Eddy Simulation (DDES)** and it should be always preferred over the standard DES.

- **Wall Modeled Large Eddy Simulation (WMLES)**

5.3 Reynolds Stress Model

Reynolds Stress Model (RSM) is often recommended as suitable model for flows with strong streamline curvature. It explicitly contains the streamline curvature production term. Therefore, it yields superior results compared to simple RANS models. However, this model is very computationally expensive as it contains 6 equations and is famous for the difficulties to maintain stability and convergence of a simulation.

6 SINGLE PHASE APPROACH FOR MODELLING SURFACE VORTICES

An alternative to computationally very expensive multiphase simulation is to use much cheaper and faster single phase approach and estimate the vortex gas core length with criterion proposed by Sakai et al. [27] and later revised by Škerlavaj et al. [28]. Škerlavaj successfully validated this approach on two experimentally measured test vessels published in [22] and [29]. The method does not consider the water surface tension. As it was already mentioned in chapter 3.7, the water surface tension can be neglected, if Weber number is greater than 120. The vortex gas core length can be calculated by using equation 6.1. Škerlavaj et al. defined the vortex center as a centroid of iso-surface $Q = 0$. However, such a low threshold will identify great number of small vortices, which are too weak to have gas core. Iso-value of $Q = 0.1$ proved to be better for identifying only the strong vortices. At the same time, slightly higher threshold does not affect much the vortex diameter.

$$L_{gc} = \frac{ln2 \cdot \alpha\Gamma^2}{4g\nu(2\pi)^2} \tag{6.1}$$

Where:

L_{gc} $[m]$ is the vortex gas core length

α $[m/s]$ is the downward velocity gradient

Γ $[1/\mathrm{s}]$ is the vortex velocity circulation

g $[m/s^2]$ is the gravitational acceleration

ν $[m^2/s]$ is the kinematic viscosity

By using the Stokes theorem, the velocity circulation Γ can be calculated as an:

1. integral of tangential velocity over curve *c*. Curve *c* is created as an intersection of the water surface and the iso-surface of Q=0.1 - Figure 6.1 a).
2. integral of vorticity over inner area *A* of the curve c.

$$\Gamma = \int_c u_i t_i ds = \int_A \left(\epsilon_{ijk} \frac{\partial u_k}{\partial x_j}\right) n_{A,I} dA \tag{6.2}$$

Where:

$u_i t_i$ $[m/s]$ is the tangential velocity to the vortex outer shell

$\epsilon_{ijk} \frac{\partial u_k}{\partial x_j}$ $[1/s]$ is vorticity

$n_{A,I}$ $[1]$ is unit vector orthogonal to the plane A.

It is better to use the second formulation of Γ, because the vortex gas core does not have to always align with the coordinate system and there would be need to calculate the $u_i t_i$ manually every time step (in case, vortex is unstable in time). In case of the second formulation, it is convenient to approximate the inner area *A* of the curve *c* by intersection of threshold value $Q > 0.1$ and water surface - Figure 6.1 b) and c). The surface A is then automatically recalculated every time step.

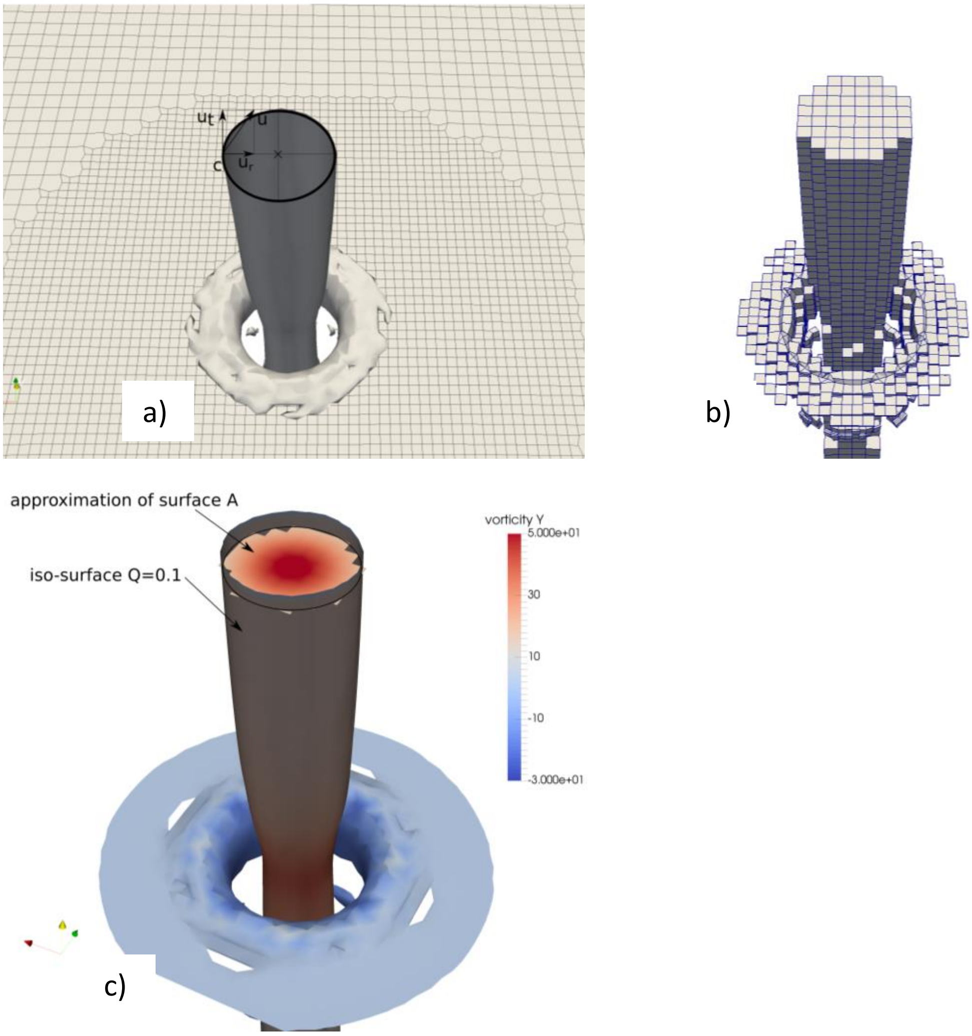

Figure 6.1 application of single phase approach for calculating the vortex gas core length on GETS test vessel (see chapter 7); a) iso-surface Q=0.1 with curve c and velocity vector components b) threshold Q>0.1, c) approximated surface A with contours of vorticity

Sakai et al. [27] proposed to calculate the downward velocity gradient α according to equation 6.3. This was later simplified by Škerlavaj et al. [28], who assumed curve *c* to be circle and replaced scalar product of $u_i n_{c,i}$ by radial velocity component. The equation 6.3 can be replaced by equation 6.4. Figure 6.2 then depicts the radial velocity pointing to the vortex center.

$$\alpha = \frac{1}{A} \oint_{c} u_i n_{c,i} ds \qquad (6.3)$$

Where:

$n_{c,i}$ [1] is unit normal vector pointing to the centroid of the curve *c*

$$\alpha = \frac{1}{\mathrm{A}} \oint_{c} u_{rad} ds \tag{6.4}$$

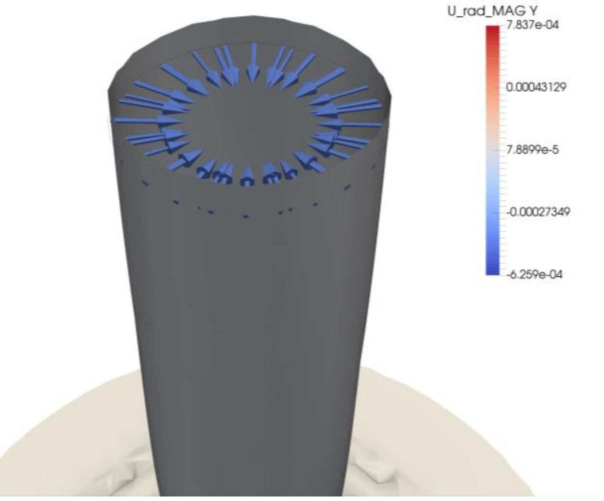

Figure 6.2 magnitude of radial velocity component

However, this still requires the user to manually identify the center of the vortex and calculate the radial velocity with python calculator or with UDF. Therefore, Škerlavaj et al. [29] suggested following equation

$$\alpha = \frac{\Delta \overline{v_z}}{\Delta z} = \frac{Q_{side}}{\Delta z \cdot A_{bottom}} \tag{6.5}$$

Where $\Delta \overline{v_z}$ $[m/s]$ is difference in mean downward velocity on surface A between two heights. First height should be the water surface and second height Δz below. Q_{side} represents the flow rate through the vortex walls. As we consider incompressible fluid, the Q_{side} term can be replaced by Q_{bottom} through surface A_{bottom}.

7 TEST CASES

The validation of numerical simulation was done on publicly available data from experimental investigation of surface vortices. Results from overall 3 experimental test cases presented by Monji et al. [22], Škerlavaj [29] and Carusco et al. [6] has been used to fine tune the numerical simulation. In this paper, only the results of numerical analysis of the test vessel design by Carusco are presented. Common characteristics of all 3 test cases are tangential water inflow to the vessel and orifice located on the bottom of the vessel. This arrangement induced vortex located above the orifice. The position of this vortex is relatively stable in place. Only the gas core grows and slightly fluctuates in time. Because by nature, vortices are very unstable, obtaining quasi stable state greatly simplifies the comparison of experimental data and numerical simulation. After the verification stage is completed, the numerical simulation (both single and two phase approach) will be utilized to investigate flow conditions and vortex occurrence in industrial wet sump designed according to recommendations in chapter 1.

Caruso et al. [29] studied formation and evolution of surface vortex for nuclear applications. He designed his Gas Entrainment Test Section (GETS), where he experimentally investigated the relations between dimensionless numbers (Reynolds number, Froude number and Weber number) and formation of vortex with gas entrainment. He designed his test vessel to have three independent parameters – diameter of an orifice, mass flow rate and height of the water in the vessel. The vessel features 2 independent inlets on both sides. To reduce possible disturbances from the inlets, water enters the vessel through two horizontal baffles, which calm the water surface, and creates tangential inflow to the orifice.

7.1 Computational Domain and Boundary Conditions

The geometry of the vessel is in Figure 7.1. The numerical analysis has been done in OpenFOAM v1606+. To compare commercial and open source solver, part of the simulations has been done also in ANSYS Fluent 16.2. Full comparison of both solvers was not possible, because of lack of computational resources.

For given geometry, only one operation condition was simulated. The water height was 50 mm and the mass flow through the outlet was $0.1\ kg/s$. Fluid with kinematic viscosity $\nu = 1 \cdot 10^{-6}\ m/s^2$, density $\rho = 1000\ kg/m^3$ was used. In case of two phase simulation, the surface tension $\sigma = 0.07\ N/m$ was also considered. Corresponding experimental results were published in [30] and [31].

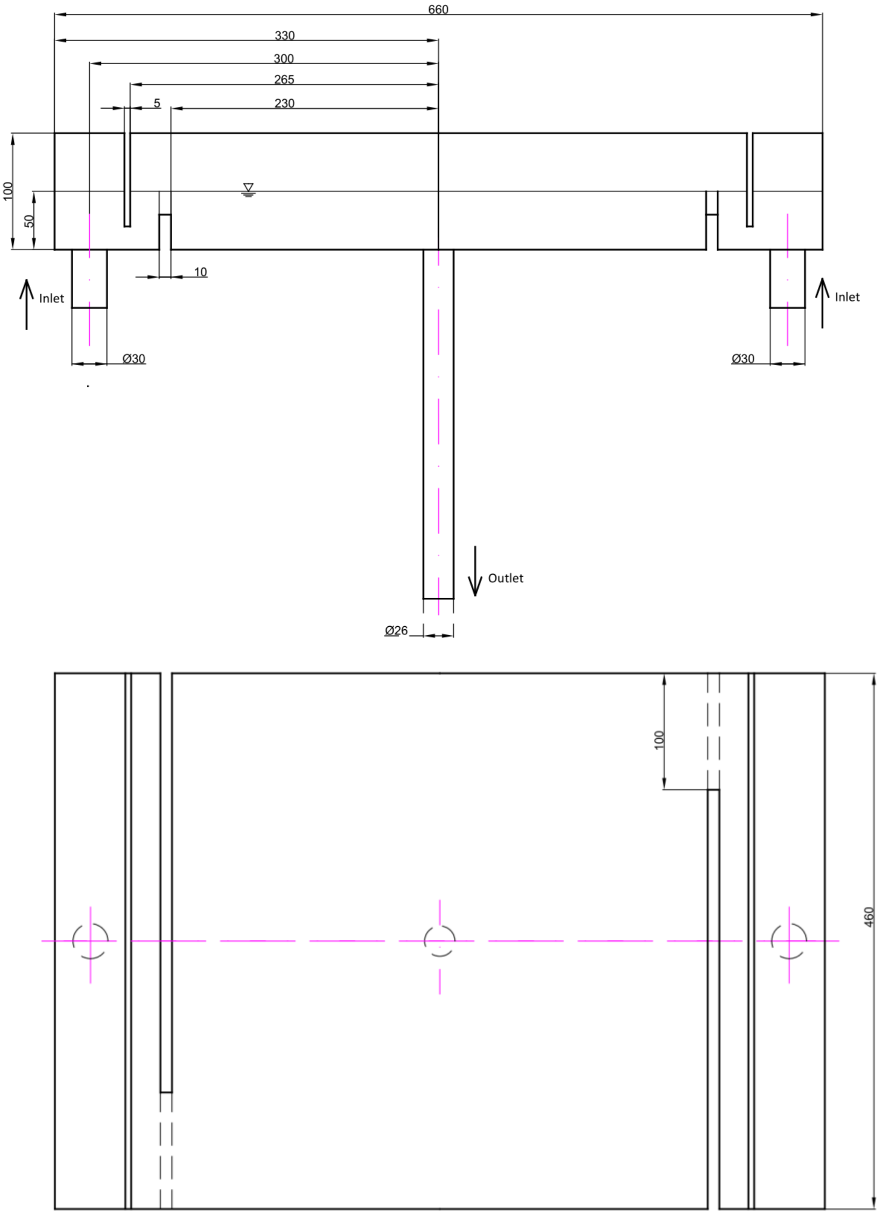

Figure 7.1 Geometry of GETS Vessel

7.2 Strategy for Meshing

The grid has been created in open-source library cfMesh, which can be easily implemented into standard distribution of OpenFOAM. cfMesh is capable of generating cartesian, tetrahedral or polyhedral meshes in both 2D and 3D. It utilizes inside-out method, which creates template (background) mesh to fit the geometry. Then it starts refining cells in user specified regions to meet required quality. Generating viscous layers is the last optional stage. The biggest advantages of cfMesh over the conventional grid generator are automated workflow, high quality cartesian mesh and small requirements on quality of input geometry – tolerates small gaps between surfaces. [32]

cfMesh requires the geometry to be defined by triangulated surface. To prevent future problems with "leaking" cells out of the domain, it's always better to put more effort and mesh the surface with quality 2D triangular grid as shown in Figure 7.2. The author considers SALOME to be the best utility to convert solid bodies from 3D CAD programs to high quality triangulated surfaces. 2D mesh is then exported to .stl file.

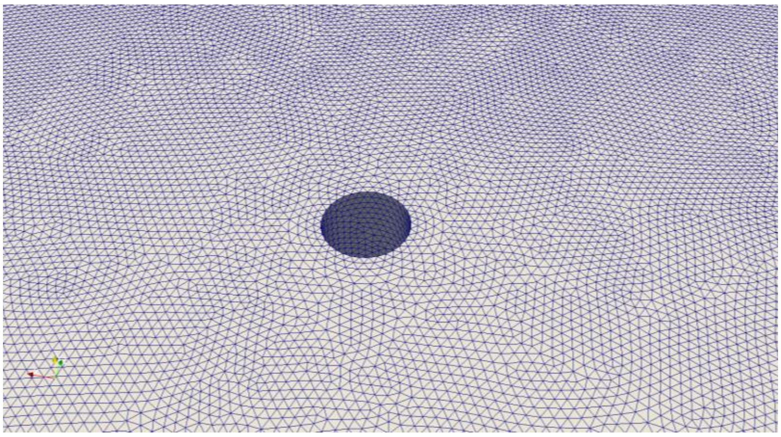

Figure 7.2 Detail of surface mesh near orifice

In Figure 7.3 is cut through the surface geometry, which is supplied to the mesher. Each colour in the figure corresponds to separate .stl file. This separation of boundary patches to individual files is very convenient, as we need to create several versions of computational grid that differ only in the height of the domain. By translating *atmosphere.stl* with function *surfaceTransformPoints -translate "(x y z)"* we obtain grid for single and multiphase analysis.

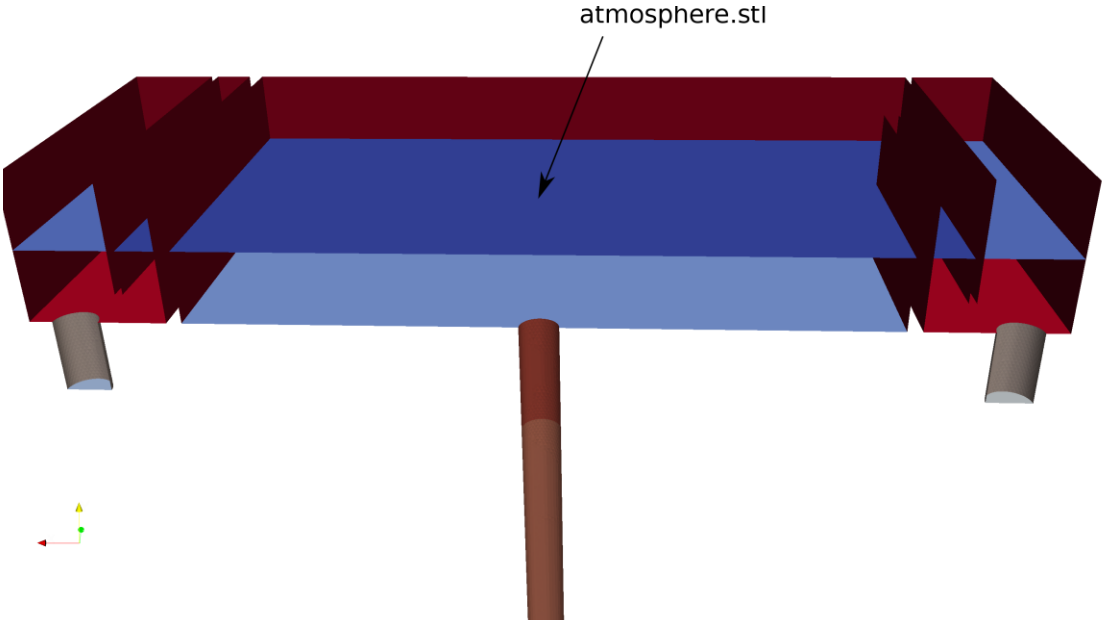

Figure 7.3 Cut through .stl files of the domain.

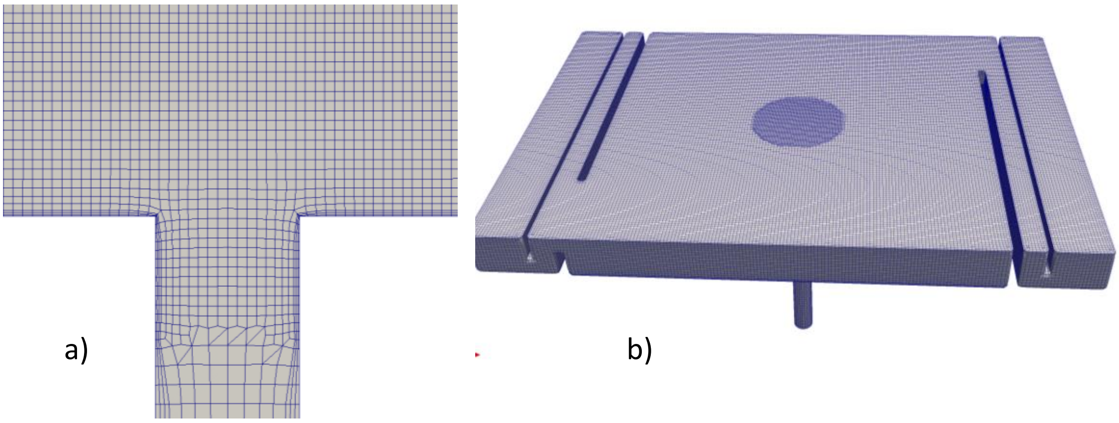

Figure 7.4 a) detail of viscous layer on walls; b) Computational domain for single phase analysis

In Figure 7.4 a) is demonstrated the ability to create viscous layer on the vessel walls and outlet tube. In Figure 7.4 b) is visible rounding of the sharp corners of the vessel. This usually happens on an edge connecting two perpendicular planes, which are defined in one boundary patch. Because *.stl* file does not contain information about sharp edges, mesher won't explicitly snap nodes to them and edge will get rounded or become jagged during smoothing stage of meshing. This can be mitigated by converting .stl file to formats which carry corresponding information about feature edges. cfMesh recommends to use format called .fms. This format has several advantages such as significant reduction of size (compared to stl does not duplicate each node), can carry multiple boundary patches in one file and contain explicit information about feature edges. Before converting of .stl to .fms, it is better to clean the 2D mesh by executing *"surfaceClean <input_surface> <mesh_resolution> <output_surface>"*, where parameter *mesh_resolution* should corresponds to the average cell size. Then the conversion can be done by utilizing *"surfaceFeatureEdges -angle 45 <input_surface.stl> <output_surface.fms>"*, where user can fiddle with angle parameter to get feature edges only where needed. The parameter "*-angle*" means that feature edge will be created between surfaces which hold angle greater than given number.

Further on, the numerical analysis splits into two parallel branches. First, the single phase approach presented in chapter 6 is used. Then similar test case is duplicated with two phase simulation. In the end, the results from both methods are compared.

7.3 Single Phase Approach

Numerical analysis has been done on two different grid resolutions to determine the influence of the mesh over results. Both grids have similar background mesh with 20 mm cell size and differ between each other only in refinement zone near orifice, where grid A (569 425 cells) utilizes additional refinement 1 over the background mesh and grid B (1 163 081 cells) utilizes additional refinement 2. Grid B has therefore 8 times higher resolution (see Figure 7.5) in region, where the occurrence of the vortex is expected. Both grids have similar quality results from *checkMesh*: maximal and average non-orthogonality 45 and 2.7, maximal aspect ratio 9.3 and maximal skewness 2.4.

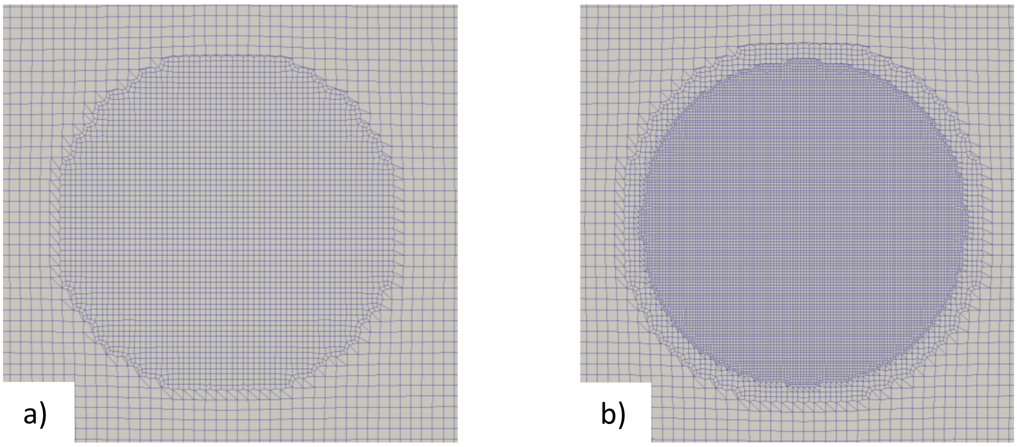

Figure 7.5 Computational domain for single phase simulation a) coarse mesh A b) fine mesh B

For single phase approach, only one solver, pimpleFoam from OpenFOAM library, was used for all simulations. PimpleFoam is incompressible, isothermal solver with generic turbulence modelling and utilization of pimple loop for large time steps. Generally, the stability of the solver is determined by time step length and consequently by Courant number. Courant number equal 5 has proven to be the best value in terms of stability and cpu time. It is possible to increase it up to 15, but more pimple loops (outer corrector loops) must be used and the stability of the solver becomes unpredictable. In case of Scale Adaptive Simulation turbulence model, the Courant number of 1 is recommended. However, to decrease the computational demands, Courant number equal to 5 was used for all single phase simulations. The best way how to ensure the constant Courant number is to use variable time step length by setting *adjustableTimeStep* to *"yes"* and *maxCo* to *5* in *controlDict*. This will also protect the simulation from blowing off in case of abrupt increase of local velocity.

The boundary conditions are summarized in Table 7.1. The top boundary was treated as symmetry patch – velocity component normal to the boundary is 0. Inlet is treated as volumetric inflow and fixed values of turbulence kinetic energy k and specific dissipation rate ω are assigned here. Their values are estimated from equation 7.1 and 7.3. Outlet is treated as pressure boundary condition. Walls are treated as no slip boundary condition and wall functions for omega and k have been used.

$$\mathrm{k} = 1.5(Iv^2) \qquad (7.1)$$

Where:

I is turbulence intensity on the inlet

v is the mean velocity on the inlet

$$\epsilon = \mathrm{C}_{\mathrm{mu}}^{0.75}\frac{k^{1.5}}{l} \tag{7.2}$$

Where:

C_{mu} is constant and is equal to 0.09 for family of $k-\epsilon$ turbulence models

l is the turbulence length scale in the inlet and is usually estimated from characteristic dimension of the inlet

$$\omega = \frac{\epsilon}{\mathrm{k}} \tag{7.3}$$

Table 7.1 Boundary conditions for single phase simulation

Pressure	Inlet	Outlet	Wall	Atmosphere
p	zeroGradient	fixedValue	zeroGradient	symmetry
U	flowRateInletVelocity	zeroGradient	noSlip	symmetry
k	fixedValue	inletOutlet	kqRWallFunction	symmetry
omega	fixedValue	inletOutlet	omegaWallFunction	symmetry
nut	calculated	calculated	calculated	symmetry

7.3.1 Numerical Schemes

For the temporal discretization, Crank-Nicolson scheme has been used. It is second order implicit, bounded scheme. Additionally, it requires a definition of off-centering coefficient ψ, which blends pure Crank-Nicolson and Euler scheme (first order accuracy, implicit, bounded). To maintain better stability of the simulation, off-centering coefficient $\psi = 0.9$ was used.

Only second order accuracy interpolation schemes were used for divergence terms. Velocity divergence was discretized by *linearUpwind* scheme, without engaging limiters for velocity gradient. Turbulence terms were discretized by *limitedLinear* ψ scheme, which limits the term in regions of rapid changing gradient. When $\psi = 1$ the scheme behaves as upwind scheme in regions of strong gradients and the best stability of the simulation is ensured (on the expense of accuracy). When $\psi = 0$ the scheme tends to linear scheme.

7.3.2 Turbulence modelling

Two simulations have been carried out without modelling turbulence (laminar model) and for the other two $k-\omega\ SST$ Scale Adaptive Simulation ($k-\omega\ SST\ SAS$) model has been used.

7.3.3 Results and discussions

The simulations were performed on THOR cluster located in Brno University of Technology. Each simulation utilized one blade server with two eight-core Intel Xeon E5-2690 2.9 GHz processor. The simulation times, number of cores and mean time step are given in Table 7.2.

The length of the time step for **grid A** and laminar run was 0.005 s in the beginning of the simulation and as the velocity field in the vessel was developing, the time step length was slowly decreasing during the whole simulation to 0.003 s in the end of the run. This brings us to conclusion, that the velocity field did not stabilize till the 200 s of the simulation time. In case of turbulent run with $k - \omega$ Scale Adaptive Simulation model, the time step was 0.0048 s and it has been almost constant during the whole run.

The length of the time step for **grid B** and laminar run was 0.0024 s in the beginning of the simulation and it was decreasing till 150 s. Then it started oscillating between 0.0014 and 0.0017. Similarly, to turbulent run on grid A, the time step was 0.0023 and it stayed almost constant for the whole run.

From the simple analysis of the solver log file it is apparent, that the turbulence model brought additional numerical damping and reduced the velocity fluctuation in cells with high Courants number. Such cells are localized on an edge of the orifice (see Figure 7.6). In near wall regions, the SAS model switch to RANS mode and the eddy viscosity is increased. This leads to higher energy dissipation and higher numerical stability. However, there is also suspicious that the SAS model did not switch to scale resolving mode in free shear region and stayed in RANS mode. It is possible that the time discretization resolution or special discretization resolution is not enough to resolve the turbulence structures in LES mode. To test the sensitivity of SAS model on temporal discretization, two additional simulations with Courant number equal to 1 have been done. However, lowering the Courant number to 1 drastically increased the computational demands.

Table 7.2 Comparison of computational demands

	Courant number [-]	Mean time step length [s]	Number of cores	CPU time [hours]
Grid A, laminar	5	$5 \cdot 10^{-3}$	14	6.5
Grid A, $k - \omega$ *SST SAS*	5	$3 \cdot 10^{-3}$	14	11.1
Grid A, $k - \omega$ *SST SAS*	1	$9 \cdot 10^{-4}$	16	39.2
Grid B, laminar	5	$1.5 \cdot 10^{-3}$	16	39.4
Grid B, $k - \omega$ *SST SAS*	5	$2.2 \cdot 10^{-3}$	16	41.0
Grid B $k - \omega$ *SST SAS*	1	$4 \cdot 10^{-4}$	16	108

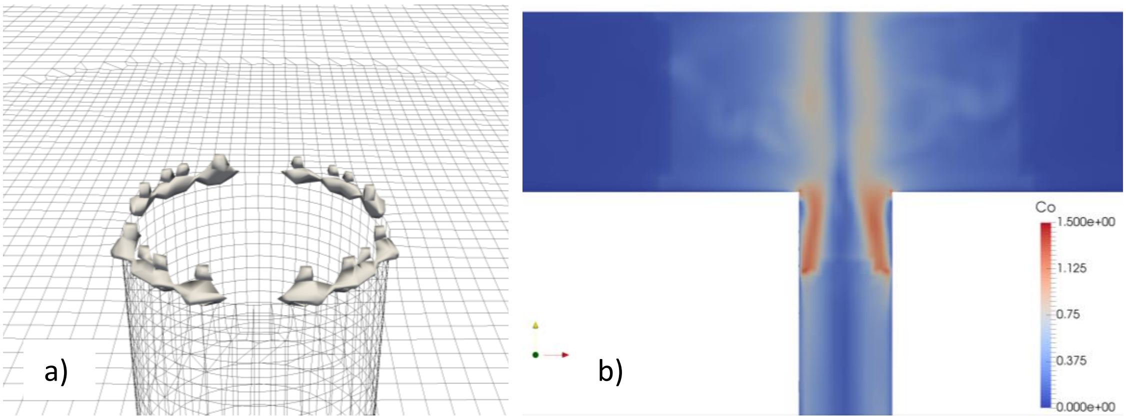

Figure 7.6 Post-processing of laminar case, figures from the end of the simulation T= 200 s; a) iso-surface of Courant number equal to 1.5 b) contours of Courant number.

Figure 7.9 shows the iso-surfaces of $Q = 0$ with contours of vorticity in downward direction. It is apparent that vortex core from laminar simulation is much thinner compared to simulation with turbulence model. There is also obvious difference between laminar run with Grid A and Grid B, where simulation with finer grid resulted in thinner vortex core with higher vorticity. This is expected behaviour as the strong curvature of the stream can be resolved more accurately on finer grid with smaller dissipation of energy through numerical viscosity.

According to Figure 7.7, the gas core length calculated from equation 6.1 and modified by Škerlavaj was heavily overestimated compared to the experimental results. Simulation with fine grid (grid B) features large fluctuation in estimated core length. Therefore, the single phase approach without modelling turbulence is not reliable and should not be used for cases with relatively low water level height.

In Figure 7.8 are plotted results with modelling turbulence. At the first sight, the turbulence model brings additional energy diffusion to the system and damps strong oscillation of velocity field and decreases vorticity. Simultaneously, small vortex ropes from laminar case circulating around main vortex in the middle of the domain (apparent in Figure 7.9 a) and b)) are completely suppressed by added eddy viscosity. It turned out that $\mathrm{k} - \omega$ SST SAS turbulent model is very sensitive to the length of the time step, as it yields different results for different Courant numbers. In case of $Co = 5$, the simulation with fine grid heavily overestimated the gas core length. When the Courant number was decreased to $Co = 1$, the production of eddy viscosity was increased (Figure 7.10 b) vs d)) and the estimated gas core moved closer to experimentally measured values. From the Figure 7.8 it is also apparent, that simulations with fine grid tends to overestimate the gas core in the beginning of the simulation and then match closely with experiment after 150 s of simulation. In case of coarse grid (grid A), the simulated values match almost exactly experiment till 140 s. The importance of the second order accuracy dicretization schemes for turbulence should be also emphasised. One additional simulation with upwind discretization scheme for kinetic turbulence energy k and specific turbulent dissipation ω has been done. The result was shocking, as no gas core was detected at all. The upwind schemes seem to be too dissipative for flows with strong stream curvature and disperse the energy from vortex core to surroundings.

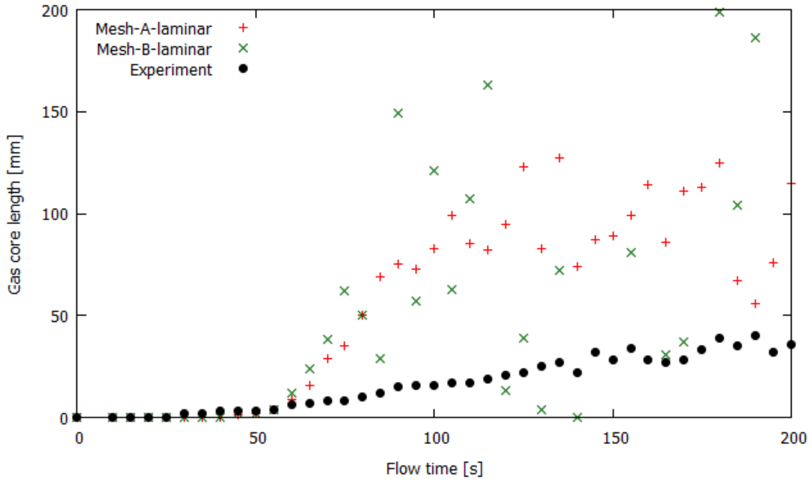

Figure 7.7 estimation of vortex gas core length from single phase, laminar simulation

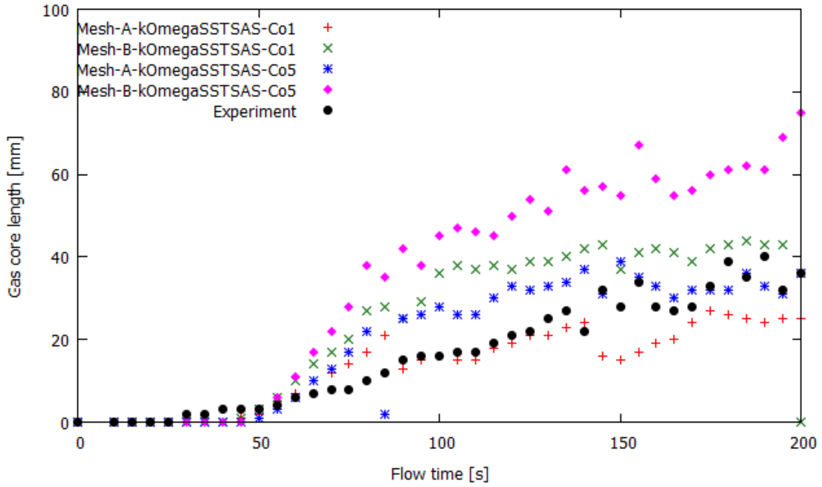

Figure 7.8 estimation of vortex gas core length from single phase, turbulent simulation

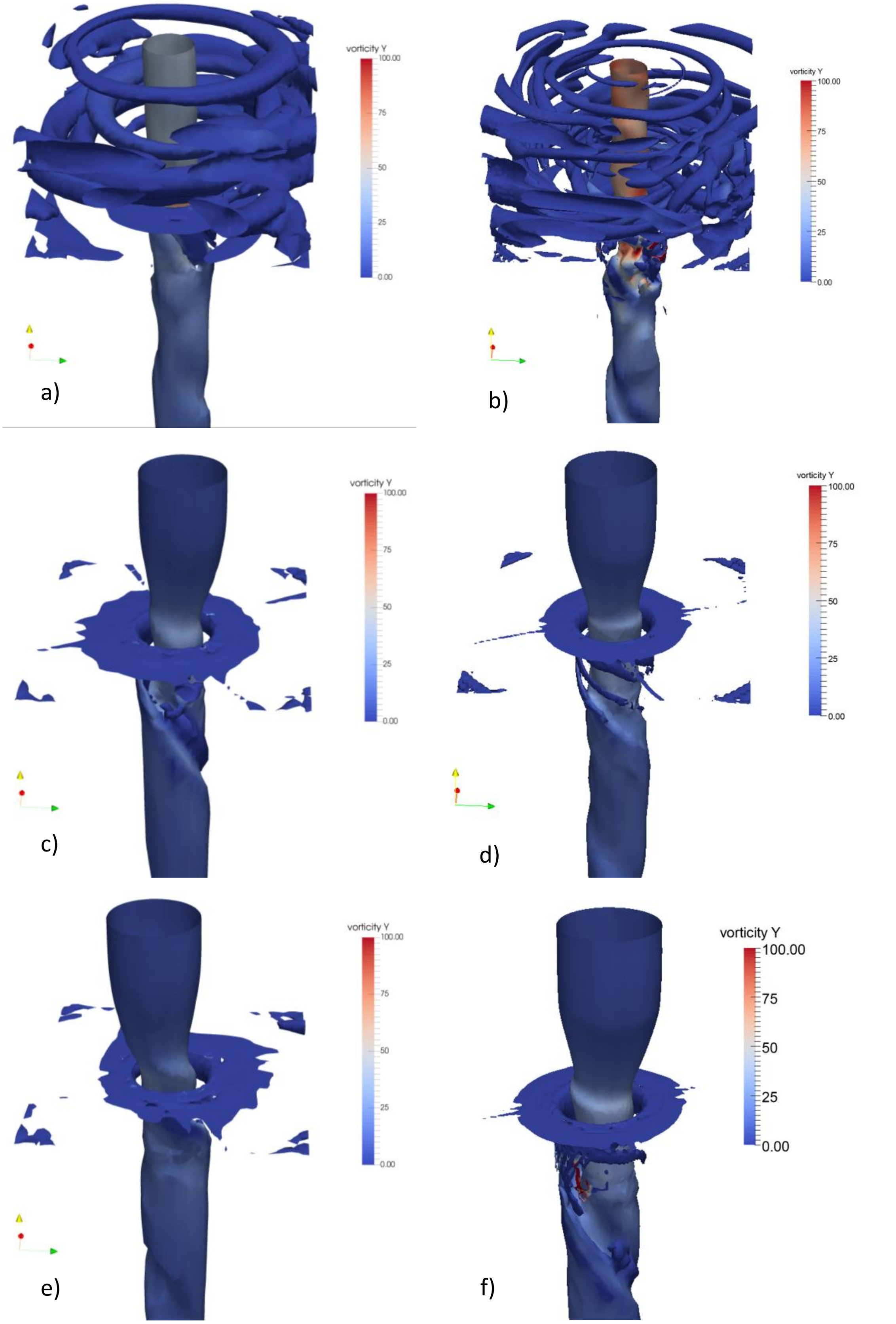

Figure 7.9 Iso-surface of Q=0.1 a) Grid A , laminar b) Grid B, laminar c) Grid A, $\mathrm{k}-\omega$ SAS model d) Grid B, $\mathrm{k}-\omega$ SAS e) Grid A, $\mathrm{k}-\omega$ SAS, $Co = 1$ f) Grid B, $\mathrm{k}-\omega$ SAS $Co = 1$

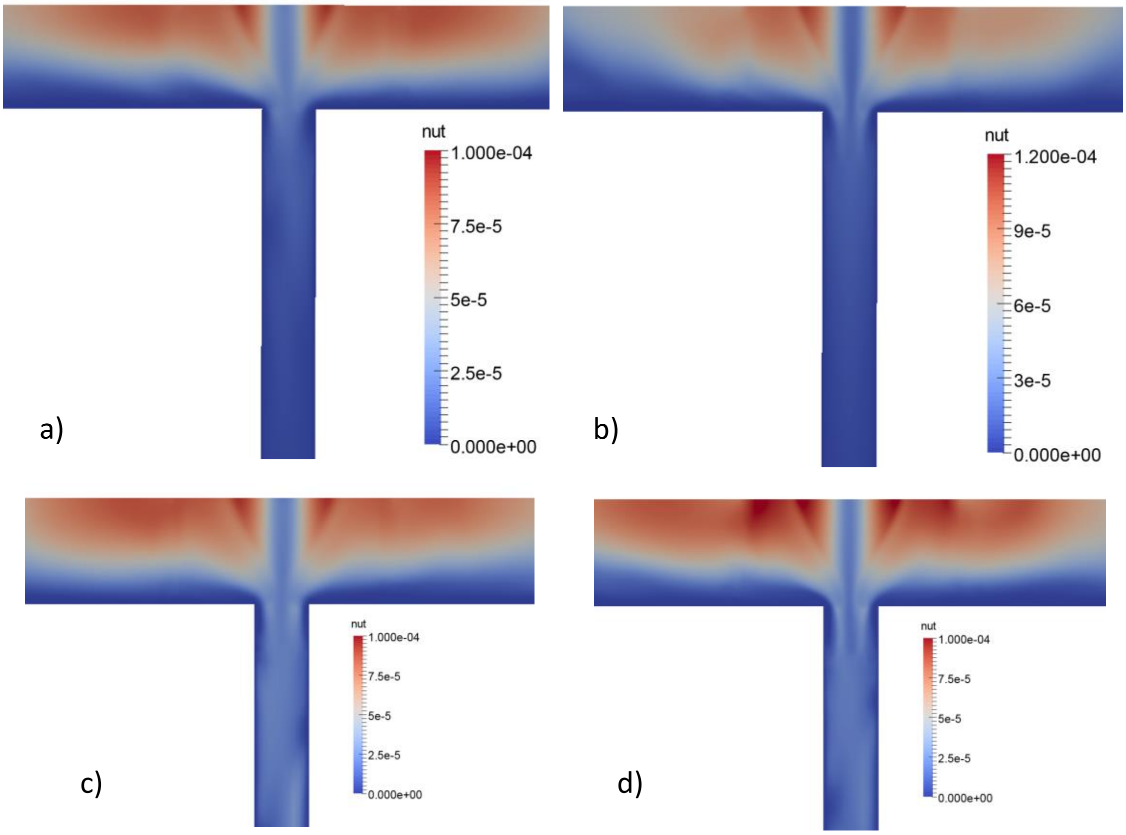

Figure 7.10 Contours of eddy viscosity a) Grid A, $\mathrm{k} - \omega$ SAS model b) Grid B, $\mathrm{k} - \omega$ SAS c) Grid A, $\mathrm{k} - \omega$ SAS, $Co = 1$ d) Grid B, $\mathrm{k} - \omega$ SAS $Co = 1$

To summarize the results, the single phase approach is suitable method for determination of vortex gas core length. It is necessary to use turbulence models, because laminar simulations heavily overestimate the gas core length. $k - \omega\ SST$ Scale Adaptive Simulation is suitable model for this purpose, but it should be used on fine meshes with small time step length, which fulfil condition $Co \leq 1$.

7.4 Two Phase Approach

For two-phase simulation two different solvers have been utilized and compared. First the simulation was done using interDyMFoam from OpenFoam package (see chapter 4.1) and then similar case was duplicated with commercial solver ANSYS Fluent 16.2. The operating conditions, boundary conditions, mesh and numeric have been kept consistent between the solvers.

The grid resolution for two phase simulation is identical to single phase version, but contains additional domain on top. This domain is 50 mm high and is filled mainly with air fraction. This domain does not require any additional refinement, as the flow of the air above the water level is not in a scope of our interest and has very little impact on the shape of the water interface – velocities of air close to 0 are expected. Furthermore, the spurious currents on the interface produce non-physical velocity field in the air domain. Figure 7.11 depicts the grid used for all two phase simulations both in

OpenFoam and Fluent. Resolution of the mesh corresponds to grid A (see Figure 7.5 a)) used during the single phase simulation. The grid has 1 092 360 cells and the quality result from check mesh is: maximal and average non-orthogonality 48 and 2.2, maximal aspect ratio 10.6 and maximal skewness 2.4.

The grid was exported from OF to Fluent via command *foamMeshToFluent*.

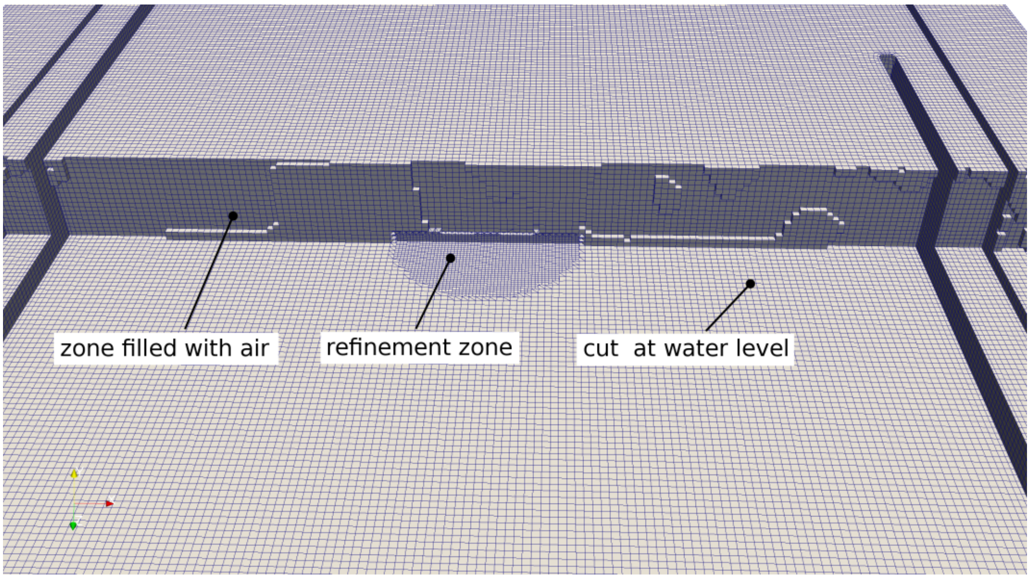

Figure 7.11 Grid for two phase simulation with cut through the domain

The boundary conditions for interDyMFoam solver are summarized in Table 7.3. To ensure that the water level will stay constant through the whole simulation, both inlet and outlet were treated as *flowRateIntelVelocity* boundary condition, where inlet had positive volumetric flow rate and outlet had negative volumetric flow rate. On patches, where backflow could occur, *inletOutlet* boundary condition has been defined. It switches from *zeroGradient* for flow out of the domain and *fixedValue* in case of flow into the domain. The top, atmospheric boundary patch was treated as total pressure boundary condition. It is more convenient to use total pressure as an atmospheric patch than simple pressure condition, because it is self-stabilizing boundary condition and can reliably reduce high peaks in local velocities. Similar to single phase simulation, walls were treated as no slip condition with wall functions for turbulence variables.

Table 7.3 Boundary conditions for interDyMFoam solver

Pressure	Inlet	Outlet	Wall	Atmosphere
p_rhg	zeroGradient	zeroGradient	fixedFluxPressure	totalPressure
U	flowRateInletVelocity	flowRateInletVelocity	noSlip	zeroGradient
alpha.water	fixedValue	inletOutlet	zeroGradient	inletOutlet
k	fixedValue	inletOutlet	kqRWallFunction	inletOutlet
omega	fixedValue	inletOutlet	omegaWallFunction	inletOutlet
nut	calculated	inletOutlet	calculated	calculated

The boundary conditions in Fluent were kept consistent with settings in InterFoam. However, there are some differences. The atmospheric patch was treated as *pressure inlet*, which should correspond to the *total pressure* BC in case of InterDyMFoam. Vessel walls were treated as no-slip condition with considering standard wall functions (in case of turbulent run). However, *mass flow inlet* was specified only to both inlet boundaries, because Fluent does not allow to use this boundary conditions for outlet. Outflow BC is also not applicable, as it only changes the static pressure on the boundary in order to redistribute the mass flow to two and more *outflow* boundaries. Therefore, the only possibility is to use *velocity inlet* BC. However, this BC will enforce piston like velocity profile to the outlet and it will suppress rotation of the liquid in the orifice. Therefore, it is necessary to use long outlet tube, not to suppress the rotation of the vortex in orifice.

7.4.1 Numerical schemes

Similar to single phase simulation, Crank-Nicholson 0.9 scheme has been used for temporal discretization. However, Crank-Nicholson scheme has one big disadvantage compared to Euler scheme in VOF solvers. It does not support sub-cycle stepping of alpha field (presented in 4.2). This means that the stability of the solver is limited by AlphaCo equal to 0.25. In the worst scenario, where the interface is moving with relatively high velocity (for example dam break tutorial), this stability condition will reduce the length of the time step by factor of four. This would increase four times the computational demands for the simulation, which is unacceptable for industrial cases. Fortunately, GETS vessel is designed to have relatively stable vortex in the middle of the domain so no rapid motion of the interface is expected and we can use Crank-Nicholson scheme without limiting the time step. To secure the stability, the keywords *maxAlphaCo* and *maxCo* have been set in *controlDict* dictionary to 0.25 and 1 for the latter.

Only second order accuracy discretization schemes have been used. Additionally, they were coupled with limiters to achieve better stability in regions of strong gradients. To clarify more the use of limiters, let's have a look at a Table 7.4. For example, The keyword *div(rhoPhi,..)* identifies mass flow on a cell face and *div(rhoPhi,U)* then denotes to advective of velocity. Keyword *Gauss linearUpwind* denotes to interpolation scheme, *Gauss linear 1* denotes to discretization scheme of velocity gradient with a limiter, where

1 means full limiting operation and 0 means no limiting operations. Keyword *cellMDLimited* specifies the type of the limited gradient scheme. [18], [33]

Table 7.4 Divergence schemes for two phase simulation with interDyMFoam

div(rhoPhi,U) Gauss linearUpwind cellMDLimited Gauss linear 1
div(phi,alpha) Gauss MUSCL
div(phirb,alpha) Gauss interfaceCompression
div(phi,k) Gauss linearUpwind cellLimited Gauss linear 1
div(phi,omega) Gauss linearUpwind cellLimited Gauss linear 1

Variable time step has been used and maximal courant number has been set to 0.9. To improve the convergence and robustness in the beginning of the run, following measures have been implemented

- 2 time iterations per one time steps – *nOuterCorrectors 1* in *PIMPLE* sub-dictionary
- 2 correction loops for pressure equation – *nCorrectors 2* in *PIMPLE* sub-dictionary
- 3 MULES correctors to correct alpha field – *nLimiterIter 3* in *alpha.water* sub-dictionary

In case of Fluent, explicit formulation of Volume of Fluid model with Implicit Body Forces and sharp interface Modelling has been used. Despite higher computational costs, the Geo-Reconstruct scheme has been used for discretization of Volume Fraction. This decision was done, because the CICSAM scheme, which is computationally cheaper, did not yield satisfactory results on a cartesian mesh and did not produced smooth interface. The momentum equation and turbulence quantities were discretized with Second Order Upwind scheme. Pressure equation was then discretized with PRESTO scheme. Similar to interDyMFoam, variable time stepping method has been used with condition of maximal Courant number equal to 0.9. Maximum 3 sub-iterations per time step were done with convergence criterions set to 10^{-5}.

7.5 Spurious Currents on the Interface

During testing phase of VOF solver, spurious currents have been detected. Spurious currents are region of high velocities, which are induced on the phase interface. Such currents are not physical and are caused only by numerical errors in curvature calculation. At a water-air interface, there is ratio of density between primary and secondary phase 1000. Steep density (volume fraction) gradients have large discretization errors and leads to large inaccuracies in curvature calculation. There are several ways, how to reduce the strength of the spurious currents. Either we can artificially increase the density of the air and therefore decrease the density gradient, or smooth the volume fraction field by using first order accuracy divergence terms for alpha field (see Table 7.5, line 1).

In Figure 7.12 are contours of velocity magnitude and volume fraction from case, where second order divergence terms have been used. There is clear onset of the spurious currents where the mesh refinement starts (highlighted by a circle). The peak of velocity is in vortex gas core, where the water circulates with relatively high velocity and thin interface.

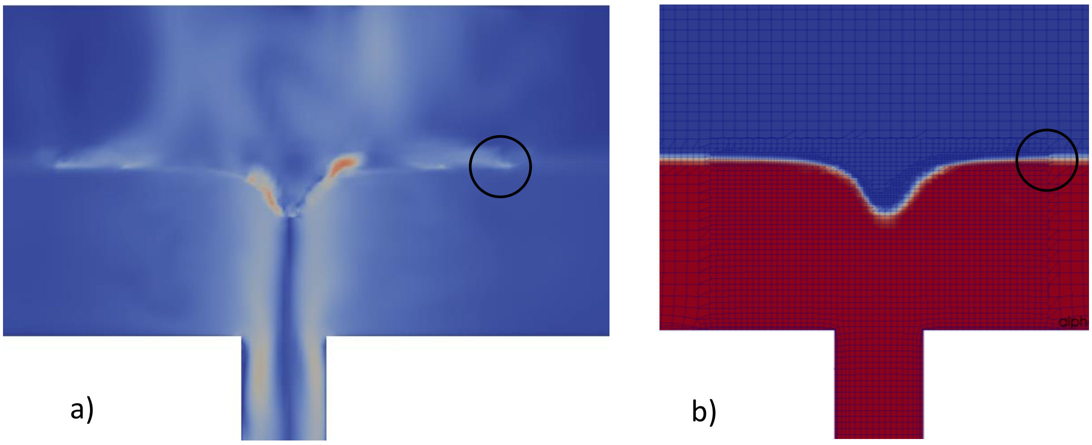

Figure 7.12 Spurious currents on interface with **second order discretization** a) contours of velocity magnitude b) contours of volume fraction

In Figure 7.13 is the same case, but first order divergence terms for alpha field have been used. It is apparent that the strength of the spurious currents has been significantly reduced, but not completely mitigated. A drawback is slightly more diffused and not that sharp interface. Summary of divergence discretization schemes is Table 7.5.

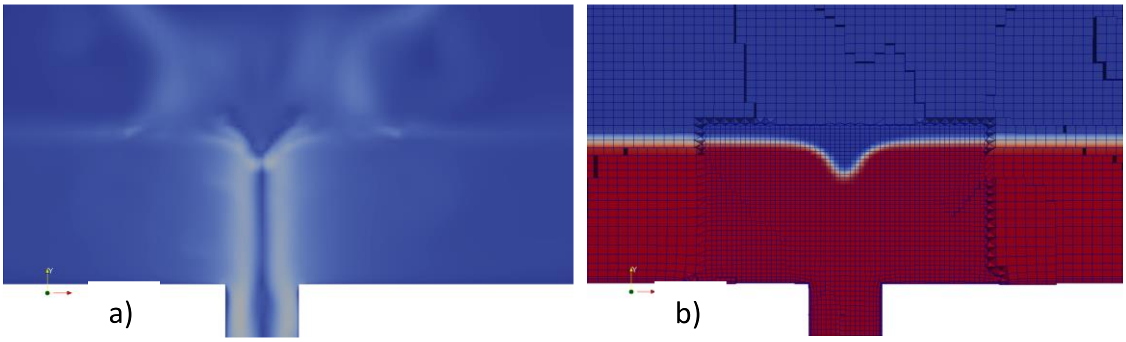

Figure 7.13 Spurious currents on interface with **first order discretisation** a) contours of velocity magnitude b) contours of volume fraction

Last method how to control the thickness of the interface is *cAlpha* coefficient in *fvScheme* dictionary. If *cAlpha* is equal to 0, no interface compression is used. In this case, the interface is very diffused and usually stretches through more than 15 cells. *cAlpha* equal to 1 is recommended value and it corresponds to conservative interface compression. *cAlpha* larger than one corresponds to enhanced compression and it is generally not recommended, as it causes solver instability.

Table 7.5 Divergence discretization for alpha field

	div(phi,alpha)	div(phirb,alpha)
First order accuracy	Gauss vanLeer	Gauss linear
Second order accuracy	Gauss MUSCL	Gauss interfaceCompression

7.5.1 Turbulence models

According to Christofano and Nobili [30], the laminar simulation is the most reliable method to estimate vortex gas core length with VOF solver. Therefore, first two simulations have been carried out without modelling turbulence. Additionally, several turbulence models were also used and compared to laminar simulation. Reynolds stress model is the most suitable turbulence model to resolve swirling flow, because it contains exact Reynolds stresses transport equation and calculates the Reynolds stresses directly. However, this makes the model very complex and requires large computational resources. To test, if the RSM would be feasible solution, one simulation in ANSYS Fluent has been carried out on HP Z800 workstation with two quad-core Intel Xeon E5530 2.4 GHz processor. After two days of computations, the simulation was stopped and the computational time needed to simulate 200 s has been estimated. On a grid with 1 million cells (see Figure 7.11), at least 8 000 hours (334 days) are needed, which makes this turbulence model not feasible for unsteady analysis, even on small clusters. Beside the RSM model, $k - \omega\ SST$ and $k - \omega\ SST\ SAS$ models have been examined.

7.5.2 Run time sampling

There are two possibilities, how to track the evolution of the interface through the time. It is possible to save every given time step, but this would save all data from the whole domain and it would require lots of free storage. Second possibility is to use run functions, which enable us to export cutting planes, point probes or is-surface. Code below shows, how to set sampling library to export water surface and cutting plane every 1000 time steps:

```
elevation
    {
        type            surfaces;
        functionObjectLibs
        (
            "libsampling.so"
        );
        outputControl   timeStep;
        outputInterval  1000;

       setFormat raw;
       surfaceFormat   vtk;

       interpolationScheme cellPointFace;

        fields
        (
            alpha.water
        );
        surfaces
        (
            topFreeSurface
            }
                type        isoSurface;
                isoField    alpha.water;
                isoValue    0.5;
                interpolate true;
            }
        );
}
cuttingPlane
{
    type            surfaces;
    functionObjectLibs ("libsampling.so");
    outputControl   timeStep;
    outputInterval  1000;

    setFormat raw;
    surfaceFormat   vtk;


    interpolationScheme cellPointFace;

    fields          (U alpha.water);

    surfaces
    (
        zNormal-plane
        {
            type            cuttingPlane;
            planeType pointAndNormal;
            pointAndNormalDict
            {
                basePoint       (0  0  0);
                normalVector    (0  0  1);
            }
            interpolate     true;
        }
        xNormal-plane
        {
            type       cuttingPlane;
            planeType pointAndNormal;
            pointAndNormalDict
            {
                basePoint       (0  0  0);
                normalVector    (1  0  0);
            }
            interpolate     true;
        }

    );
}
```

7.5.3 Results and discussion

First, we will focus on laminar simulation done in interDyMFoam solver. From Figure 7.14 is apparent that the first and second stage of surface vortex (chapter 1.4) is captured accurately. However, from 75 s of flow time, interDyMFoam underestimated the gas core, which grows slower and does not pass to its third stage. These results comply with the results from single phase simulation with turbulence modelling, where the simulation on coarse grid captured well the initial evolution phase of the surface vortex, but underestimated the core length in later stages. Therefore, it is expected that by refining the mesh to resolution similar to Grid B from single phase approach, the numerical dissipation would be reduced, and the results would be improved. However, we can estimate that by refining the mesh by one order of magnitude, we would increase the computational time from 222 hours to 560 hours (23 days on 16 cores).

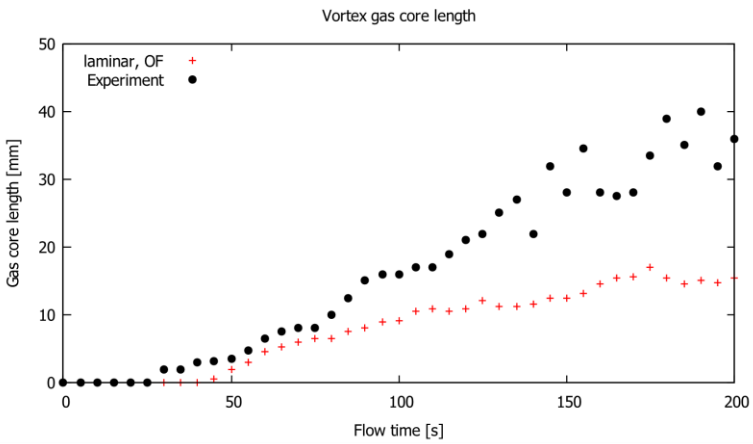

Figure 7.14 Gas core length predicted by interDyMFoam solver, without modelling turbulence

Figure 7.15 shows comparison of volume fraction contours after 200 s of simulation. The final length of the gas core was in case of interDyMFoam 15.4 mm and in case of Fluent 9 mm. Similar trend persisted throughout the whole simulation, where the Fluent underestimated the core length even more than OF. One possible reason for this issue could be that OpenFOAM is natively coupled with *snappyHexMesh/cfMesh* and it is tuned to work well on cartesian (non-conformal) grids. On the other hand, Fluent is designed to work well on tetrahedral, or structured, conformal meshes from TGrid, Gambit, or ICEM. Therefore, further research should examine influence of a grid type on accuracy of results, while maintaining similar resolution.

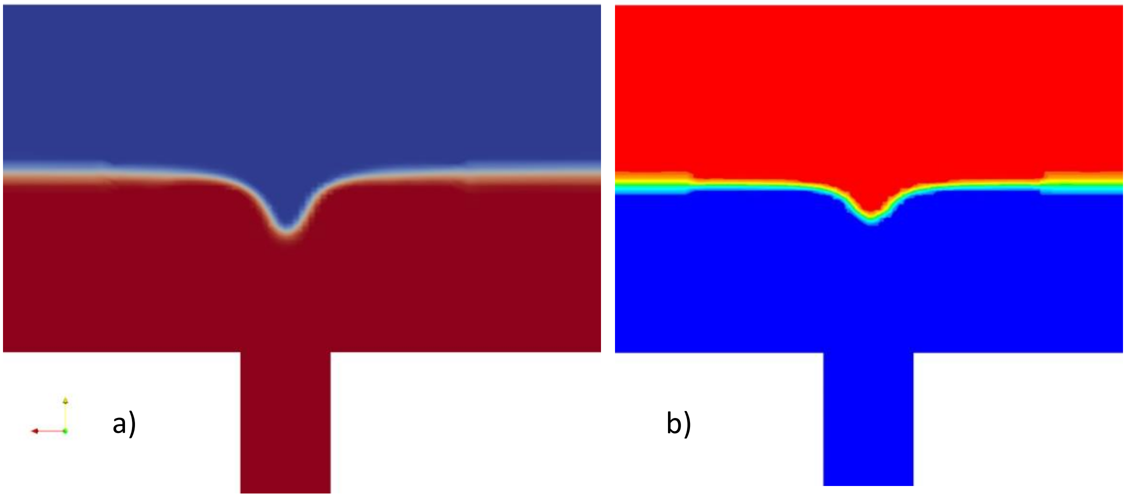

Figure 7.15 Contours of volume fraction, laminar simulation, flow time 200 s a) interDyMFoam b) Fluent 16.2

Figure 7.16 depicts comparison of spurious currents on the interface between both solvers. It is apparent that both solvers have difficulties to resolve the velocity field above the water surface. interDyMFoam produces strong spurious currents in the vortex gas core. In this case, the velocity magnitude went over 0.8 m/s, which is almost twice as much as maximum velocity of water in orifice. It is therefore recommended, to use only first order accuracy divergence schemes to discretize the alpha field and stick with more diffuse interface. Results from fluent does not suffer from such strong spurious currents. Only minor increase in velocity is apparent on the interface in region, where the grid refinement starts (non-conformal cells).

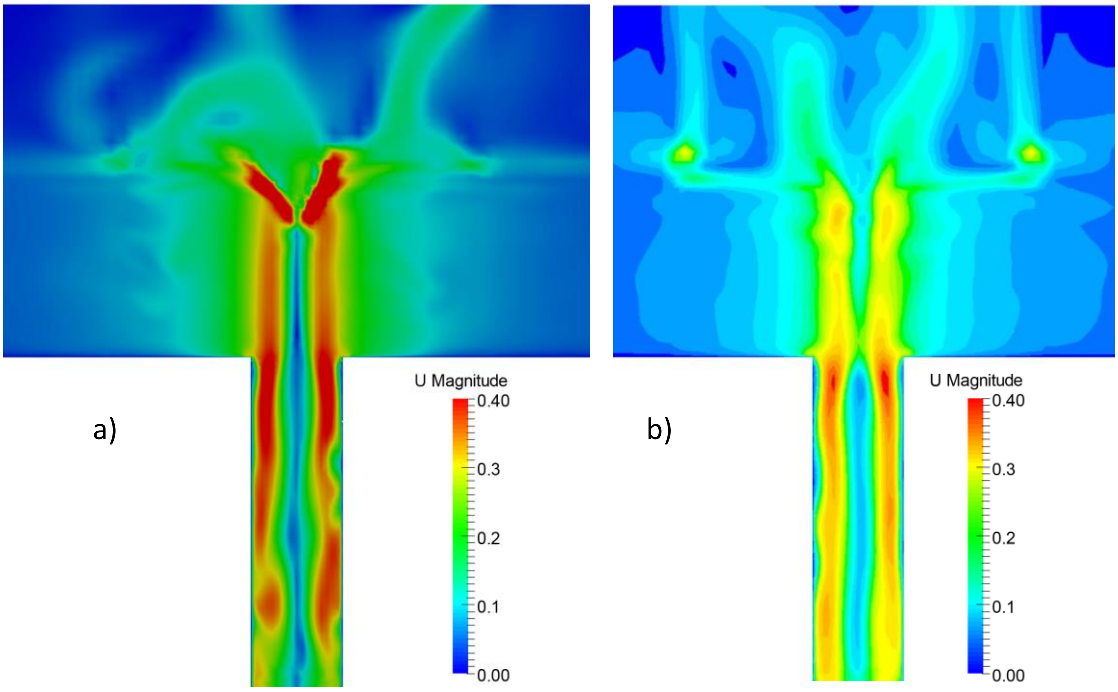

Figure 7.16 Contours of velocity magnitude, laminar simulation, flow time 200 s a) interDyMFoam b) Fluent 16.2

Despite the unsatisfactory results from the laminar simulation, where the gas core was underestimated, the performance of turbulence model coupled with VOF solver has been tested. The turbulence model will bring to the simulation additional energy diffusion through eddy viscosity, so the expectancy was that the estimated vortex core will be even shorter. Nevertheless, the behavior of the solver can be observed and by refining the mesh it is possible to obtain better results with modeling turbulence than from simple laminar model.

Figure 7.17 a) depicts strange behavior of water-air interface. By further investigation of the problem, strange propagation of eddy viscosity through the *atmosphere* patch was discovered (Figure 7.17 b)). Even reducing inlet values of kinetic turbulent energy did not solve the problem and the eddy viscosity spread easily through the interface (see Figure 7.20 for comparison with Fluent).

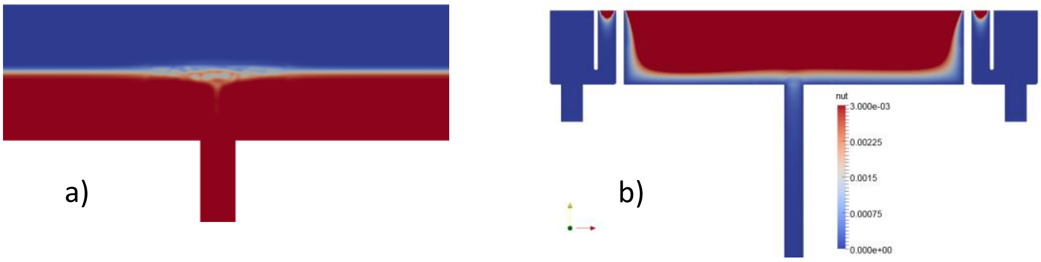

Figure 7.17 Turbulent run with interDyMFoam, $k - \omega\ SST\ SAS$ turbulence model, simulation time 40 s, contours of: a) volume fraction b) eddy viscosity

The only solution was to change boundary condition of patch *atmosphere* to wall. After this modification, the simulation has been stabilized and reasonable values of eddy viscosity and volume fraction have been obtained. In Figure 7.18 are depicted results from the run. The eddy viscosity field closely matches single phase run and from the figure b) is visible, that the spurious currents on the interface have been also suppressed. However, even after 140 s of simulation vortex gas core has not been created and only small dimple on the water surface appeared. To save computational resources, the simulation was stopped at this point and declared as too dissipative.

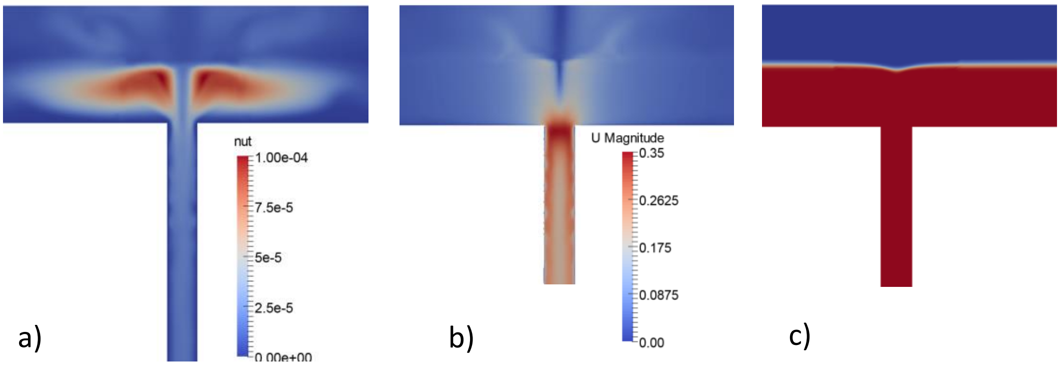

Figure 7.18 interDyMFoam, $k - \omega\ SST\ SAS$, simulation time 143 s a) contours of eddy viscosity b) contours of velocity magnitude c) contours of volume fraction

In Figure 7.19 are depicted results from ANSYS Fluent with $k - \omega\ SST\ SAS$ turbulence model with curvature correction option enabled. The curvature correction (see chapter 5.1) should sensitize the simulation to strong stream curvature. Figure 7.19 b) shows the contours of curvature correction coefficient. It is apparent that the curvature coefficient is zero in the vortex core, which stops the production of eddy viscosity. This should reduce the dissipation of energy from the vortex centre and boost the formation of vortex gas core. However, even after 270 s of simulation the vortex gas core was not formed (Figure 7.19 a)) and only small dimple in water surface has been detected. The investigation of eddy viscosity (see Figure 7.20) reveal that still large amount of eddy viscosity is produced in vicinity of the vortex core. Comparison with single phase simulation (see Figure 7.10 b)), multiphase analysis in ANSYS Fluent has produced 300 times more of eddy viscosity.

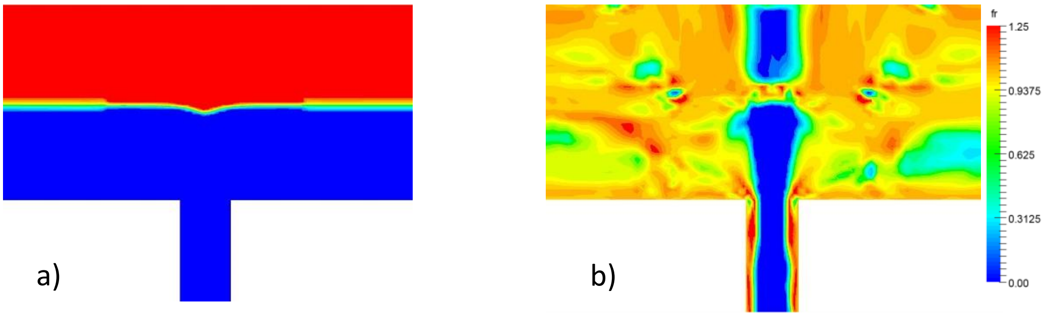

Figure 7.19 Fluent 16.2, $k-\omega\ SST\ SAS$ with curvature correction, flow time 270 s, contours of: a) volume fraction b) curvature correction coefficient

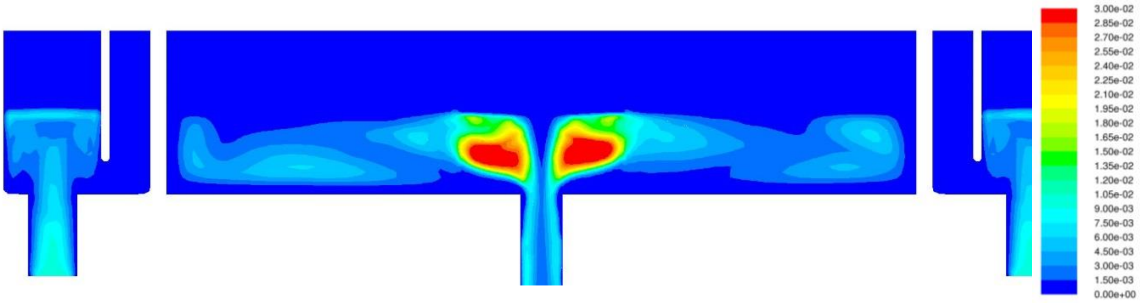

Figure 7.20 Fluent 16.2, contours of turbulent viscosity, flow time 270 s

To summarize the results, the multiphase approach has proven to be prospective method for determination of occurrence of vortices with gas core. In comparison with single phase approach it simplifies the post processing and determination of vortices, which are potentially dangerous. On the other hand, it is much more computationally expensive and only laminar model can be recommended.

8 EXPERIMENTAL STUDY

Next step was to design experimental test case, which will have geometry similar to real wet sump e.g. horizontal inlet and vertical suction tube. The design has been done according to rules for unconfined wet sumps mentioned in chapter 1.

To induce swirling motion in the sump and enforce formation of vortices with gas core, two rules of the best practice wet sump design were violated. The mean velocity in suction tube was exceeded from maximal allowable 1.7 m/s to 2.5 m/s. Then, by varying the submergence of the suction pipe, surface vortex occurrence map has been constructed. In sake of simplicity, only one flow rate of 20 l/s and one diameter of the suction tube has been considered. For this operational point, the minimal allowable submergence is 686 mm. It was observed that it is necessary to reduce the minimal submergence to less than half to detect any surface vortex formation.

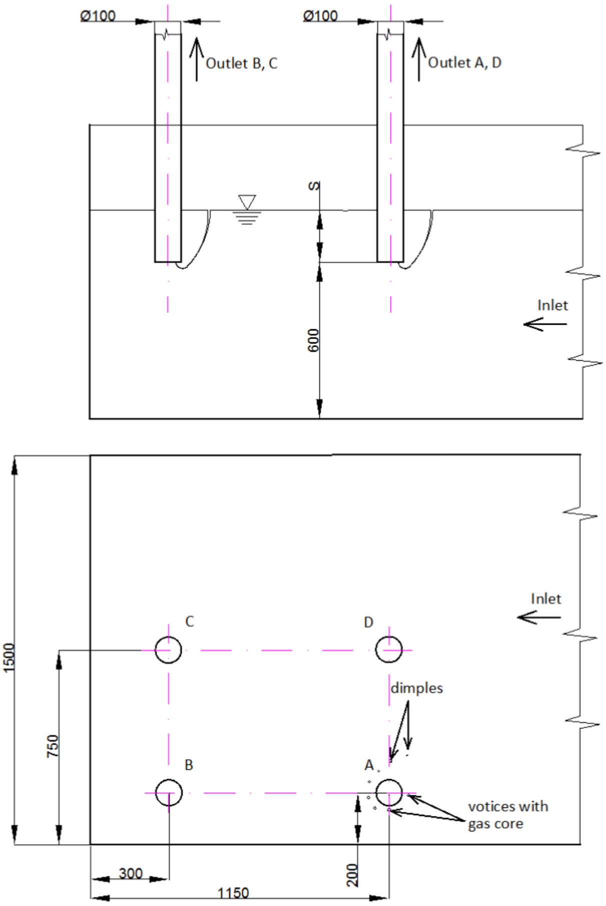

Figure 8.1 geometry of an experimental vessel with 4 configurations of the suction tube

The suction tube has been placed to 4 different positions in the sump to determine the influence of walls on formation of gas cores. In Figure 8.1 is the proposed geometry of the vessel with 4 different positions of the suction tube. Unlike experimental studies carried out by Monji et al. [22], Skerlavaj et al. [34] and Caruso et al. [6], the experimental vessel proposed in this paper does not feature tangential inlets to the vessel. On the one hand, this will impose realistic inflow condition similar to industrial wet sumps. On the other hand, it makes the occurrence of the surface vortices highly unstable and unpredictable.

Every suction tube configuration, was measured with following submergence S: 160 mm, 200 mm, 240 mm, 280 mm and 320 mm. Every operating point has been run for 6 minutes and highest observed stage of a surface vortex has been taken to the occurrence map. After 6 minutes of a run, the water height was increased and the experiment proceeded to the next operating point.

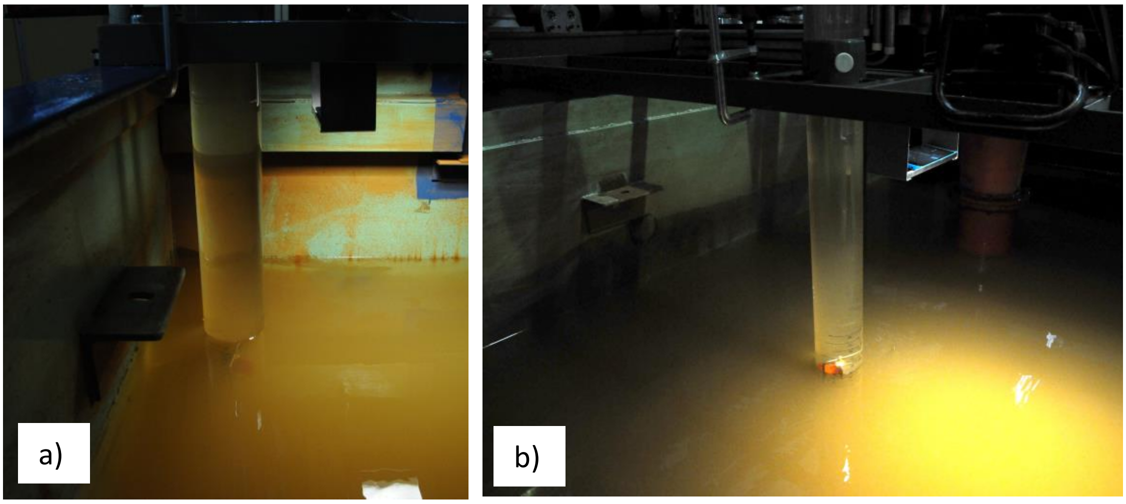

Figure 8.2 a) pipe configuration B b) pipe configuration D

In Figure 8.1, pipe configuration A, is a sketch of a vortex formation. First, small dimples are formed on the side opposite to the tank wall (sight glass). Dimples grow and pass to stage 3 and 4 as they travel around the suction tube. Region between suction tube and near vessel wall proved to be especially prone to formation of stage 4 vortices. In this region vortices are more stable, last longer and appear even with higher submergence. Figure 8.3 b) shows a surface vortex stage 4 from experimental run with pipe configuration A and submergence 160 mm. For this operating point, the surface vortices with full gas entrainment appear almost immediately. They are highly unstable and last only few seconds. Furthermore, they travel around the suction tube, maintaining long gas core with gas entrainment even 3D from the suction tube (see Figure 8.3 a)). This has significant consequence for building computational grid for the numerical analysis, as it is necessary to have sufficient mesh resolution to resolve vortex core even 3D from the suction tube.

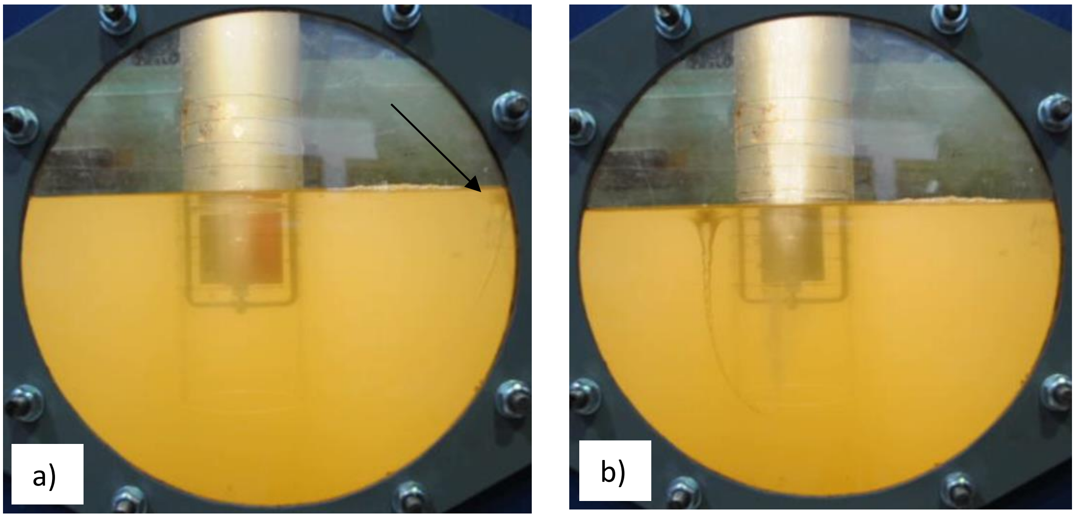

Figure 8.3: a) position of a vortex meter b) configuration A, submergence 160 mm

Figure 8.3 a) shows pipe configuration B. Pipe placed in tank corner resulted in asymmetrical inflow and in the lest favourable behaviour in terms of flow stability, swirl angle and vortex occurence. During experimental runs with submergency 200 mm and 240 mm, no full gas entrainment was observed. However, during experimental run with 280 mm, one stage 4 vortex has been observed in the tank corner. The vortex was stable, letting great amount of air to be sucked to the pumping system. Even with the highest tested submergence, a stage 3 vortex with detaching bubbles from its gas core has been observed.

Figure 8.3 b) shows pipe configuration D. This configuration evinces the best performance and the smallest danger of vortex occurrence. For detailed vortex occurrence map see Table 8.1.

Table 8.1 surface vortex occurrence map in pipe configuration A, B, C, D and in configuration with grating

S [mm]	Froude number [-]	Surface vortex stage				
		A	B	C	D	gratings
160	**0.0631**	4	4	4	4	2
200	**0.0564**	4	3	4	2	1
240	**0.0515**	3	2	4	1	1
280	**0.0477**	2	4	2	1	1
320	**0.0446**	1	3	1	1	1

To evaluate the swirling motion in the suction tube, a swirl meter has been placed to the suction tube. The swirl meter was designed according to standards mentioned in chapter 1.3, i.e. diameter and length of the vane is 75 mm and 60 mm respectively. In Figure 8.3 a) is position of the swirl meter inside the suction tube. The bottom edge of the swirl vane should be 4D from the pipe mouth. However, in this position the swirl meter stayed mostly stationary. To improve the sensitivity, the swirl meter has been moved to a distance 1D from the pipe edge (see Figure 8.4). Frequency of swirl meter rotation was averaged over 2 minutes. Swirl angle was then calculated from equation 1.7 and results were put into a Table 8.2. It was expected that with increasing submergence, the swirl angle in suction tube will decrease. This has not been observed, but correlation between swirl angle and pipe configuration has been found. The configuration D (suction pipe in a corner) shows the lowest values of swirl angle. Furthermore, it was observed that the swirl meter did not have preferred direction of rotation. This is an evidence of good flow conditions in suction tube. The water approach the bell mouth symmetrically and rotation of the swirl meter is induced only by local perturbations, which are weak and does not influence pump performance. The configuration B (suction tube in a corner) shows the highest values of swirl angle. The

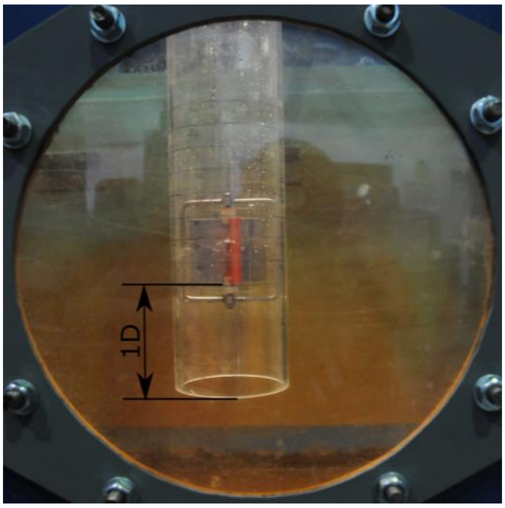

Figure 8.4 position of a surface meter in a suction tube

swirl meter has been rotating with constant direction and frequency. This indicate unfavorable inflow conditions and lead to increased danger of surface vortex occurrence.

Table 8.2 swirl angle in pipe configuration A, B, C, D and configuration with grating

S [mm]	Froude number [-]	Swirl angle [rad]				
		A	B	C	D	grating
160	0.0631	3.95	3.95	3.83	0.60	5.27
200	0.0564	2.64	3.36	3.60	2.04	3.60
240	0.0515	3.95	5.86	3.36	0.72	4.07
280	0.0477	2.16	6.33	3.95	0.84	3.95
320	0.0446	3.95	4.19	3.48	0.72	3.60

8.1 Mitigation of Excessive Swirling and Surface Vortex Occurrence

To reduce the excessive swirling inside the pump inlet and reduce the risk of a surface vortex occurrence, it is generally recommended to use floor cones, AVD devices, fins inside the suction tube or grating on the water level (described in detail in chapter 1.2). The effect of grating was examined in this paper.

Figure 8.5 a) and b) show a grating and its position on the water surface. The grating has square profile with 30 mm spacing and 20 mm height. It was hanged from above for easy manipulation. The effect of the grating was studied on suction pipe configuration A and results from this experimental run are in Table 8.1, section “gratings”. As it is visible from the table, grating prevented any surface vortex formation in all operating points. Vortex stage 2 in experimental run with submergence 160 mm has been probably formed, because the grating does not completely surrounds the outlet tube. Results from Table 8.2 indicates, that application of gratings cannot improve flow conditions in the suction tube. If excessive swirl is a problem,

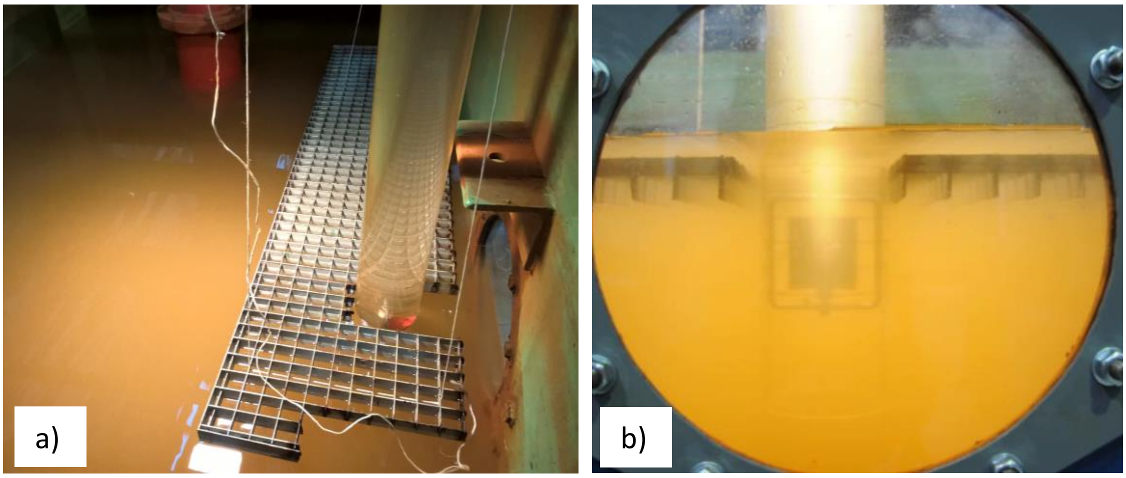

Figure 8.5 grating on the water surface

Figure 8.6 depicts experimental run that was measured with remnant pulley in the tank. It has been observed that in case the pulley was immersed in the water, close to the water surface, no vortex has been observed in the whole run. When the pulley was more than 50 mm under water, the vortices appeared again. However, engineers should be always careful about obstacles immersed under water surface, because their effect is unpredictable. On the one hand, they can mitigate vortex occurrence, on the other hand they can act as a initiator of local instabilities, which can lead to formation of strong, stage 4 vortex.

From the experimental results it seems, that if the surface vortex occurrence is likely to be issue, gratings are one of the best solution for this problem. They should be placed to height, which corresponds to minimal water height in the system. Note that the grating must be fully immersed in the water for optimal performance. The spacing of the bars should be smaller than 0.2D and reach at least 3D around the suction tube. Results from Table 8.2., section “grating” indicates, that application of gratings on the water surface cannot improve flow conditions in the suction tube. If excessive swirl is a problem, AVD device or fins should be used instead.

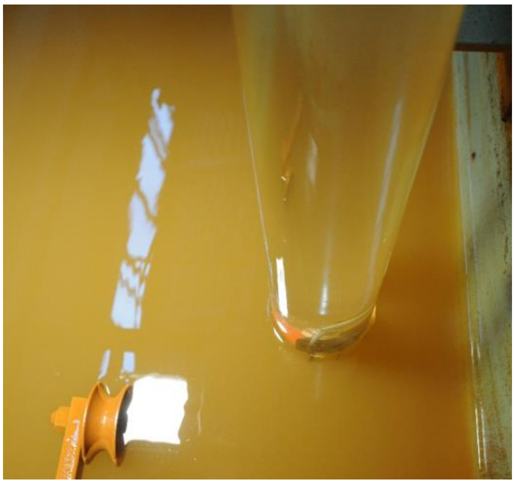

Figure 8.6 pipe configuration B, submergence 160 mm, remnant pulley from previous experiments

9 NUMERICAL ANALYSIS OF THE EXPERIMENTAL STUDY

In chapter 7, single and two phase numerical simulation has been verified on simple test case with bathtub like orifice and satisfactory agreement with experimental data has been observed. However, in case of real wet sump, flow patterns and formation of surface vortices are different. Therefore, it was necessary to assess the validity of the numerical model on a geometry, which is similar to industrial wet sump. The accuracy and feasibility of the numerical model should be also evaluated.

For numerical analysis, only one operating point and position of the suction tube in the sump has been simulated. The geometry used in numerical analysis closely corresponds to experimental setup, case a), pipe submergence 150 mm, flow rate 20 l/s (for geometry dimensions see Figure 8.1). This operating point has been chosen, because the large surface vortices with gas entrainment appears already after few seconds from the start of the experiment. Therefore, it should be sufficient to simulate only 10 to 20 seconds to evaluate the performance of the model.

Figure 9.1 depicts the computational grid used in single phase approach. The computational grid was generated using cfMesh tool, which produce hexahedral, cartesian mesh. The background cell size is 25 mm (edge length). In vicinity of the outlet tube, where the occurrence of vortices is expected, the mesh is refined by factor of 3, which corresponds to cell size of 3.125 mm. In addition, 3 viscous layers on the suction tube has been created. The first cell of the viscous layer is 1.5 mm high. The grid has 1 839 686 hexahedral cells, max skewness 2.45, maximal and average non-orthogonality 64.4 and 2.76 respectively.

The computational grid for two phase simulation is similar to the one used in single phase approach (see the difference between Figure 9.1 a) and b)). The only differences are that the domain is 100 mm higher and is refined (level 1) on the water-air interface. This results in significant increase of cell count to 2 695 289 hexahedral cells.

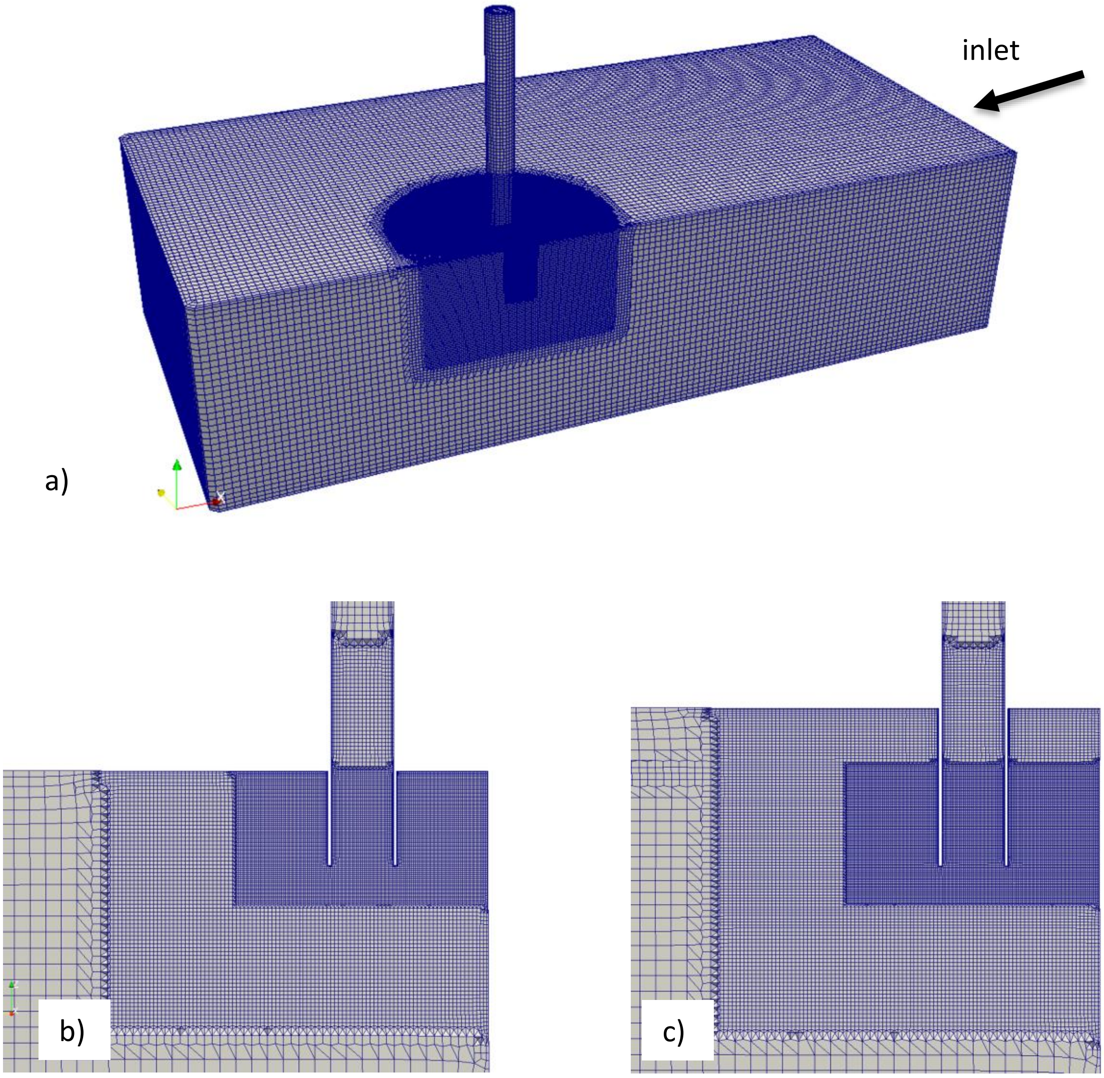

Figure 9.1 a) computational domain of VUT vessel b) slice of grid for single phase simulation c) slice of a grid for multiphase simulation

PimpleFoam and interDyMFoam from standard distribution of OpenFOAM v1606+ have been used for single phase and multiphase simulation respectively. Except for minor changes discussed in following chapters, the settings of the solver and boundary conditions are the same as mentioned in chapter 7.3 and 7.4. To improve the robustness of the simulation, variable time step length has been used. Two simulations have been performed. One without modelling turbulence and Courant number equal to 5 and second simulation with k-ω SST SAS turbulence model and Courant number set to 1.

The simulations have been done on one node of a blade server equipped with two octa-core intel Xeon E5-2690 2.9 GHz processors. To outline the computational demands, 55 s of a single phase simulation with modelling turbulence took approximately 456 hours and 9 s of a multiphase simulation without modelling turbulence took 235 s.

9.1 Results for Single Phase Simulation

In Figure 9.2 are iso-surfaces of criterion $Q = 0.1$. The laminar simulation determined large amount of vortical structures between suction tube and tank wall. These vortices are initiated from the water surface and tank wall. Because the Q criterion indicated large number of vortical structures, which are not always strong enough to form gas core, it is not possible to simply use equation 6.1 to calculate actual gas core length of individual vortices. Increasing the threshold value of Q criterion to find only the strong, possibly dangerous vortices does not work either, as all of them have high values of Q. When we use the equation 6.1 for the whole region, defined by vortical structures, we will receive the length of a gas core equal to 3.4 mm. If this result could give us any usefull information, whether gas entrainment could occur, need further investigation

In case of turbulent simulation, the surface vortices are completely dissipated and only one sub-surface vortex is formed. This vortex is established after 20 s of simulation and persist till the end of the run. From the contours of eddy viscosity (see Figure 9.3) is apparent a region of high eddy viscosity between suction tube and vessel wall. In this region, a strong vortex was observed during experimental study. Therefore, it was expected that the turbulence model will switch to LES mode and it will decrease the eddy viscosity to values close to 0. This did not happen and the model probably stayed in RANS mode. There are several possibilities, why this has happened. Symmetry boundary condition applied on the top patch might be not suitable, or the frequency of the vortex unsteadiness was too low to be recognized by the turbulence mode. On the other hand, the unsteadiness of the sub surface was identified and eddy viscosity was significantly reduced. This prevent the energy dissipation through viscosity and a sub-surface vortex has been formed.

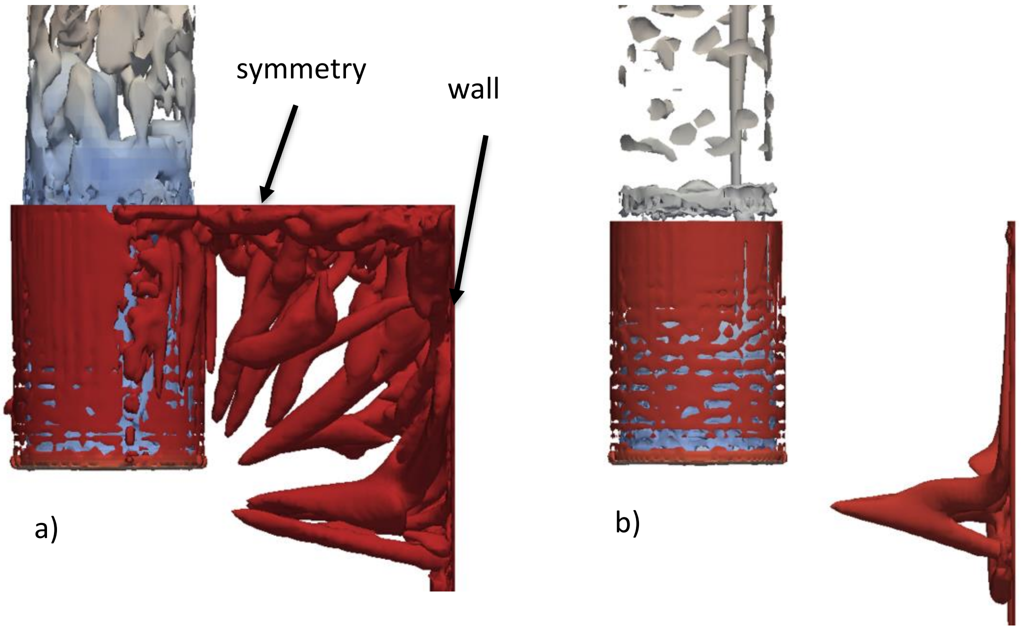

Figure 9.2 iso-surface Q=0.1 colorized by p_rgh a) laminar b) $k - \omega\ SST\ SAS$

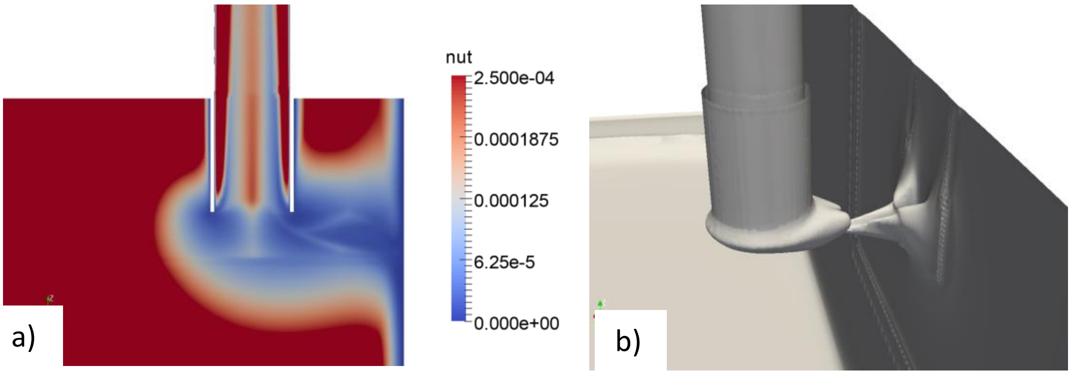

Figure 9.3 a) contours of eddy viscosity b) iso-surface of eddy viscosity $2 \cdot 10^{-6}\ m^2/s$

9.2 Results for Multiphase Simulation

In case of multiphase simulation, we must modify inlet boundary condition, to prescribe mass flow of primary and secondary phase through inlet patch and ensure correct free stream height. There are two ways how to do that. The first possibility is to split the inlet patch to two patches. Each phase will then enter the domain through separate patch. The disadvantage is that in case of waves the boundary condition behaves unphysically and produce bubbles or drops which significantly deteriorate stability. The other possibility is to use so called "channel flow" boundary condition. There is no need of separating the inlet patch to two patches. User have to prescribe only mass flow for primary and secondary phase and define height of water surface. This treatment is superior to the first one, as it improves solver stability and provide the possibility to change the free stream height without remeshing the grid.

In ANSYS Fluent, the first choice is preferred. The user should enable the VOF model and then activate the *Open Channel Flow* boundary condition. Then it is only required to set the free surface height and mass flow of primary and secondary phase.

OpenFOAM does not contain any native *open channel* boundary condition, but as a substitution it is possible to use boundary conditions from Table 9.1 or Table 9.2.

variableHeightFlowRate boundary condition use

- *zeroGradient BC* if $upperBound \leq alpha \leq lowerBound$
- set $alpha = lowerBound$ if $alpha < lowerBound$
- set $alpha = upperBound$ if $alpha > upperBound$

variableHeightFlowRateInletVelocity is a velocity boundary condition specifically designed for multiphase flows. As the BC name suggests, it prescribes volumetric flow rate to a patch. The volumetric flow rate is then weighted according to alpha field (see Figure 9.4).

This combination of boundary conditions ensures boundedness of alpha field between 0 and 1 and improve stability of the simulation

Table 9.1 Set of derived boundary conditions on inlet for channel flow, option 1

Field	Settings
alpha.water	variableHeightFlowRate; lowerBound 0; upperBound 0.9;
U	variableHeightFlowRateInletVelocity; flowRate $inletFlowRate;
p_rgh	zeroGradient;

In ANSYS Fluent, user should use hybrid initialization, which will initialize velocity and pressure field in a domain by considering constant water height from *Open Channel Flow* BC.

Unfortunately, OpenFOAM does not offer hybrid initialization for multiphase simulation. Hybrid initialization is done by potential flow solver, available only for incompressible solvers utilizing standard *p* pressure formulation (InterFoam uses *p_rhg* pressure formulation). The simulation in OpenFOAM has been initialized with zero velocity and constant volume fraction height. Figure 9.4 b) depicts difficulties in the beginning of the simulation. When we impose full flow rate to a stationary velocity field, a wave at the inlet patch is created. This wave increase volume fraction Courant number which can significantly reduce time step length. It is therefore recommended to use Euler time discretization scheme (ddt scheme) and at 5 *nAlphaSubCycles* in the begging of the simulation. After 1 second of the run, it is possible to reduce the *nAlphaSubCycles* to 1 and use Crank-Nicholson 0.9 ddt scheme.

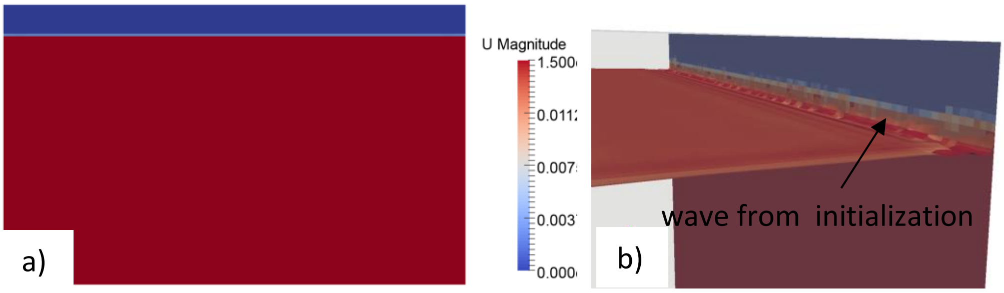

Figure 9.4 a) inlet velocity profile of *variableHeightFlowRateInletVelocity* b) contours of iso-surface $alpha.water = 0.5$ and inlet patch colored by $alpha.water$ after 1 s of simulation

Second possibility is to use boundary conditions from Table 9.2. However, this combination is not self-stabilizing and option 1 BC should be preferred.

Table 9.2 set of basic boundary conditions on inlet for channel flow, option 2

Field	Settings
alpha.water	zeroGradient;
U	fixedValue;
p_rgh	zeroGradient;

Figure 9.5 a) and b) shows results from the multiphase simulation. Contours of velocity magnitude show recirculating stream above the water surface. These streams are not problematic, as long as they do not influence the shape of water surface or do not restrict the time step length. Despite the usage of first order discretization schemes for alpha field, the solver has been able to maintain very sharp interface, which is only one cell thick. Furthermore, no strong spurious currents were detected.

Although, no surface vortex has been formed during 7 s of simulation, the contours of volume fraction field show signs of gas entrainment (highlighted by arrows in Figure 9.5 b)). To confirm, if the multiphase simulation can predict occurrence of surface vortices, longer simulation time is necessary. To simulate 50 s on 2 nodes of THOR cluster (32 physical cores), approximately additional 25 days of computation would be needed.

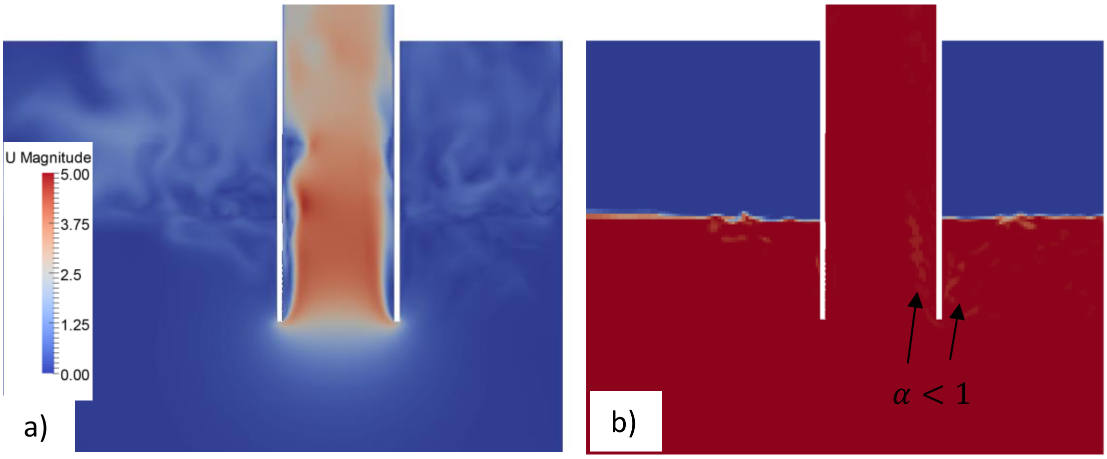

Figure 9.5 results from interFoam solver after 7 s of simulation a) contours of velocity magnitude b) contours of volume fraction, node values

CONCLUSION

In the theoretical part of the present Master's thesis, the author has summarized the key rules for designing industrial wet sumps and has described mathematical models for identification and visualization of vortices in fluid flow. Comparison of the two most common mathematical models that are used today has been done on a simple test case. It was proven that the difference between Q criterion and λ_2 criterion is negligible and both models are interchangeable.

As a next step the basics of multiphase modelling have been outlined. The Volume of Fluid model has been determined as the most suitable model for modelling surface vortices. Governing equations and recommended settings of the solver were discussed.

It turned out, that modelling of surface vortices requires long simulation time and it is very CPU intensive. To reduce the computational time to feasible degree, fine tuning of a mesh is necessary. The length of a time step is determined by Courant number. To accelerate the simulation, peaks in Courant number must be mitigated. Still large number of iterations is required, thus the computational domain should contain the least number of cells possible. In regions were a sharp water interface is needed, it is possible to perform mesh refinement. A simple python tool for refining the grid in vicinity of a vortex has been developed and its capability was demonstrated on a test case.

Then the author moved to information research about available open source CFD packages. CFD package OpenFOAM has been chosen as the most appropriate solver, because it is available on most computational clusters. OpenFOAM has steep learning curve, so basic structure and workflow is outlined here to help beginners to orient in a new system.

Turbulence modeling is integral part of every industrial simulation. It has been proven that traditional turbulence models like $k - \epsilon$ cannot be used for vortex modelling. They tend to overestimate eddy viscosity in a vortex core, where the streamlines have strong curvature. In literature *Curvature Correction* has been suggested as solution for this problem. However, only minor improvements were detected. Laminar simulation proved to be one of the best approach for accurate predictions of flow in vortex center. This method is fast and reliable for most applications. However, in situations, where employment of turbulence model is necessary, $k - \omega\, SST$ Scale Adaptive Simulation turbulence model yield satisfactory results.

Both single and multiphase simulation has been validated on a simple test case of bathtub surface vortex with relatively low water level. The single-phase approach matches well experimental observation only in coupling with $k - \omega\, SST$ Scale Adaptive Simulation turbulence model and Courant number smaller or equal to 1. The laminar simulation vastly overpredicted the length of the gas core. The multiphase simulation is more diffusive and tends to predict shorter gas cores than observed in experiment. In this case, only laminar approach is usable and yield satisfactory results. Turbulence model introduce to the system additional dumping which dissipate the energy and no vortex is formed.

To evaluate the performance of the numerical model on a case of real industrial wet sump, a scaled laboratory model was designed. The scope of the experimental study was to determine operating points, where surface vortices occur. The measurements were done for 4 different positions of suction tube and for 5 different submergences. As a result, vortex occurrence map was composed. It was observed that if no tangential inflow is imposed on the suction pipe, the surface vortices are highly unstable and travels randomly around the suction pipe.

The experimental study focused also on investigation of vortex occurrence reduction. An effect of gratings immersed under water was measured. The gratings proved to be very effective counter-measure to completely reduce the occurrence of the surface vortices in all operating points. Gratings should be completely immersed under water, close to the water surface in operating point with the lowest allowed submergence.

The experimental data were used to validate the numerical model on geometry of real wet sump. Application of single-phase model proved to be very difficult, because the numerical analysis displayed great number of vortices. It is not easy to distinguish, which vortex could be strong enough to have gas core and thus it is not possible to calculate the length of the vortex gas core. Single phase simulation with turbulence model failed to predict surface vortices at all and displayed only one strong underwater vortex. Multiphase simulation did not yield satisfactory results either. During the simulation, no proper surface vortex has been observed. However, there were some sights of gas entrainment, but longer simulation time would be needed to confirm if vortex will be formed.

Further work is necessary to perform additional multiphase simulation to confirm, if the presented numerical model is reliable tool for surface vortex identification. Furthermore, it is necessary to evaluate the accuracy for operating points with higher submergence. There are also possibilities to extend the range of applicability of the numerical model to different branches. For example, the model could be used for identification of minimal melted alloy height in a tundish during continuous steel casting. There is a danger of bathtube like vortex formation, which drags impurities and bubbles to a mold.

BIBLIOGRAPHY

[1] **JONES G M.** *Pumping Station Design.* San Diego, USA : s.n., 2008. ISBN: 978-1-85617-513-5.

[2] **Americal National Standards Institute, Inc.** *Americal National Standard for Pump Intake Design, ANSI/HI 9.8-1998.* 9 Sylvan Way, Parsippany, United States of America : Hydraulic Institute, 1998. ISBN 1-880952-26-2.

[3] **HECKER G E.** *Fundaments of vortex intake flow. Swirling flor problems at intakes.* Rotterdam, The Netherands : IHAR Hydraulics structures design manual 1, J. Knauss Ed. Balkema, 1987. pp. 13-38.

[4] **SHI Zhi-peng, ZHANG G, LIU J, HE T.** *Numerical and Experimental Study on Flow Characteristics and Sediment Deposition of Pump Sump.* Yangling, China : Collage of Water Resources and architectural Engineering, Northwest A&F University, 2016.

[5] **HEISENBERGER W.** *Swirling Flows.* Dartmouth, Canada : Thayer Schoool of Engineering at Dartmouth.

[6] **CARUSO G, CHRISTOFANO L, NOBILI M, VITALE DI MAIO D.** *Experimental Investigation of Free Surface Vortices and Definition of Gas Entrainment Occurrence Maps.* s.l. : Journal of Physics: Conference Series 501 (2014), 2013. 012019.

[7] **HECKER G E.** *Model-Prototype Comparison of Free-Surface Vortices.* s.l. : J. Hydr. Div 107, 1981. 1243-59.

[8] **HUNT J.C.R., WRAY A.A., MOIN P.** *Eddies, Streams, and Convergence Zones in Turbulent Flows.* s.l. : Center for Turbulence Research, 1900. CTR-S88.

[9] **HUSSAIN, F.** *On the identification of a vortex.* s.l. : Journal of Fluid Mechanics 285, 1995. 69-94.

[10] **CHONG M. S., PERRY A. E. CANTWELL B.** *A general classification of three-dimensional flow fields.* Melbourne, Australia : Physics of Fluids A2 (5), 765-777, 1990.

[11] **JIN-YI J.** *Lecture 4, Circulation and Vorticity.* Washington : Department of Earth System Science , 2012.

[12] **JIANG M, MACHIRAJU R, THOMSON D.** *Detection and Visualization of Vortices. Visualization Handbook.* Ohio, United State of America : Department of Computer and Information Science, 2005. ISBN 0-12-387582-X.

[13] **MATTEO N, CRISTOFANO L.** Influence of Boundary Condition in Numerical Simulation of Free Surface Vortices. *70th Conference of the Italian Thermal Machines Engineering Association, ATI2015, Volume: Energy Procedia Volume 82, Pages 1-1024.* 2015.

[14] **ANSAR M, NAKATO T, CONSTANTINESCU G.** *Numerical Simulation of Inviscid Three Dimensional Flows at Single- and Dual-Pump Intakes.* s.l. : Journal of Hydraulic Research 40(4):461-470, 2002.

[15] **SREEDEVI K.** *A Solution for Every Multiphase Challenge.* accesible from: https://support.ansys.com/staticassets/ANSYS/Conference/Confidence/Phoenix/Downloads/solution-every-multiphase-challenge.pdf] Chicago, USA : ANSYS, ANSYS, Inc., 2012 2012.

[16] **ANSYS, Inc.** *ANSYS Fluent Theory Guide, Release 15.0.* Canonsburg : ANSYS, Inc., 2015.

[17] **STŘASAK P.** *Volume of Fluid: Internal Training Materials.* Praha, Czech Republic : Enginn s.r.o., 2017.

[18] **GREENSHIELDS C J.** *OpenFOAM User Guide version 4.0.* s.l. : The OpenFOAM Foundation, 2016.

[19] **Holzinger G.** *OpenFOAM A Little User-Manual.* Linz, Austria : Johannes Kepler University, 2015.

[20] **BANIABEDALRUHMAN A.** Dynamic Meshing Around Fluid-Fluid interfaces with Applications to Droplet Tracking in Contraction Geometries. Dissertation : Michigan Technological University, 2015. Vol. http://digitalcommons.mtu.edu/etds/2005.

[21] **ŠKERLAVAJ A, LIPEJ P, RAVNIK J, ŠKERGET L.** *Turbulence Model Comparison for a Surface Vortex Simulation.* Timisoara, Romania : 25th IAHR Symposium On Hydraulic Machinery And Systems, 2010.

[22] **MONJI H, SHINOZAKI T, KAMIDE H, SAKAI T.** *Effect of Experimental Conditions on Gas COre Length and Downward velocity of Free Surface Vortex In Cylindrical Vessel.* Orlando, Florida, USA : ASME International Conference on Nuclear Engineering, 2008. ICONE16-48670.

[23] **SHUR L M, SPALLART P, STRELETS M, TRAVIN A.** *Turbulence Modeling in Rotating and Curved Channels: Assesing the SpalarShur Correction.* s.l. : AIAA Journal, 2000. vol 38, No. 5.

[24] **Menter F R, Smirnov P E.** *Sensitization of the SST Turbulence Model to Rotation and Curvature by Applying the Spalart-Shur Correction Term.*

[25] **CARLA VIDAL MARTINEZ I.** *Impact of Curvature Correction on Wingtip Vortex Flow.* s.l. : Master's thesis, 2015.

[26] **MENTER F R.** *Best Practice: Sclae-Resolving Simulation in ANSYS CFD.* s.l. : ANSYS Germany GmbH, 2012.

[27] **SAKAI T, EGUCHI Y, MONJI H, ITO K, OHSHIMA H.** *Proposal of design criteria for gas entrainment from vortex dimples based on a computational fluid dynamics method.* s.l. : Transfer Engineering, 2008. 29(8):731-739.

[28] **ŠKERLAVAJ A, ŠKERGET L, RAVNIK J, LIPEJ A.** *Predicting Free-Surface Vortices with Single Phase Simulation.* Ljublana, Slovenia : Engineering Applications of Computational Fluid Mechanics, 2014. Vol. 8.

[29] **ŠKERLAVAJ A, ERMENC D.** *Experimental Observations and Numerical Simulations of a Free-Surface vortex in a Test Vessel.* Ljubljana, Slovenia : 6th IAHR

International Meeting of the Workgroup on Cavitation and Dynamic Problems in Hydraulic Machinery and systems, 2015.

[30] **CRISTOFANO L, NOBILI M.** *Influence of Boundary Conditions in Numerical Simulation of Free Surface Vortices.* Rome, Italy : ATI 2015 -70th Conference of the ATI Engineering Association, 2015. 82 (2015) 893-899.

[31] **CRISTOFANO L, NOBILI M, CARUSO G.** *Numerical evaluation of gas core length in free surface vortices.* Pisa, Italy : 32nd UIT Heat Transfer Conference, 2014. doi 10.1088/1742-6596/547/1/012030.

[32] **NAGAWKAR J, SVENUNGSSON J, NILSSON H.** *Evaluate the Use of cfMesh for the Francis-99 Turbine.* Gothenburg, Sweden : Chalmers University of Technology, 2016.

[33] **MOHANAMURALY P.** Instructional workshop on OpenFOAM programming, LECTURE #7. [Online] 2014. [Cited: 15 April 2017.] http://pavanakumar.github.io/compressibleFoam/_downloads/week_3_day_1.pdf.

[34] **ŠKERLAVAJ A, VEHAR F, PAVLIN R, LIPEJ A.** *A Hydraulic Study of Cooling Water Intake Structure.* Brno, Czech Republic : 3rd IAHR International Meeting of the Workgroup on Cavitation and Dynamic Problems in Hydraulic Machinery and Systems, 2009.

[35] **RUDOLF P.** *Von Karman Vortices.* Olomouc : VUT, 1991. 23134.

[36] **Škerlavaj A, Pavlin R.** *Effect of Vortical Structures on Cavitation on Impleller Blades in Pumps with Suction Chambers.* Montreal, Canada : 27th IAHR Symposium on Hydraulic Machinery and Systems, 2014. doi: 10.1088/1755-1315/22/5/052002.

[37] **Rudolf P.** *A Contribution to Study of the Swirling Flow.* Brno, Czech Republic : Brno University of Technology, 2010. ISSN 1213-418X.

LIST OF ABBREVIATIONS AND SYMBOLS

Abbreviation	Explanation
AVD	Anti Vortex Device
BC	Boundary condition
CO	Courant number
CS	Coherent structure. It is flow phenomena that is present in the flow for relatively long time. An example could be a vortex. In contrary eddy is considered as a non-coherent structure, because it is transient phenomena a does not last long (only few periods of dynamical motion).
DES	Detached Eddy Simulation
DDES	Delayed Detached Eddy Simulation
GETS	Gas Entrainment Test Section
HPC	High performance computing
LES	Large Eddy Simulation
OF	OpenFOAM
PIV	Particle Image Velocimetry
RANS	Reynolds-averaged Navier-Stokes equations
RSM	Reynolds Stress Model
SAS	Scale Adaptive Simulation
URANS	Unsteady Reynolds-averaged Navier-Stokes equations
ddt	Time discretization

LIST OF ATTACHMENTS