CHALMERS

k-ε k-ω SST k-ω SST SAS S-A DDES S-A IDDES

Numerical Investigations of the Unsteady Flow in the Stuttgart Swirl Generator with OpenFOAM

Master's thesis in Fluid Dynamics

MARTIN GRAMLICH

MASTER'S THESIS IN FLUID DYNAMICS

Numerical Investigations of the Unsteady Flow in the Stuttgart Swirl
Generator with OpenFOAM

MARTIN GRAMLICH

Department of Applied Mechanics
Division of Fluid Dynamics
CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg, Sweden 2012

Numerical Investigations of the Unsteady Flow in the Stuttgart Swirl Generator with OpenFOAM
MARTIN GRAMLICH

Cover:
Isosurface of pressure (colored by axial velocity) and axial velocity equal to -0.001 $\frac{m}{s}$ (solid color). The axial velocity is positive in flow direction, i.e. from top to bottom. From left to right: k-ϵ, k-ω SST, k-ω SST SAS, $Spalart$-$Allmaras$ $DDES$ and $Spalart$-$Allmaras$ $IDDES$.

Numerical Investigations of the Unsteady Flow in the Stuttgart Swirl Generator with OpenFOAM
Master's thesis in Fluid Dynamics
MARTIN GRAMLICH
Department of Applied Mechanics
Division of Fluid Dynamics
Chalmers University of Technology

ABSTRACT

As a consequence of the current change in the electric energy supply structure and present available electric energy storage capabilities, hydro power plants are tending to be operated on a wide spread of off-design conditions. The operation of turbines not running at their best operating point could lead to a physical phenomenon called helical vortex or vortex rope. Although the helical vortex has been investigated for several decades already, it is still far from being completely understood. Moreover, it could have undesirable impacts on the hydraulic system of the power plant, namely, an increase of the risk of fatigue failure, the development of resonance vibration and a decrease of the efficiency of the power plant. In order to analyze the vortex rope and to assess its effects on the hydraulic system, it is therefore in the interest of both science and industry to simulate the physical phenomenon of the helical vortex accurately. In response to this, the Institute of Fluid Mechanics and Hydraulic Machinery at the University of Stuttgart, Germany, constructed a Swirl Generator to investigate a swirling flow, which is similar to the downstream of the turbine runners. With the help of eight non-rotating blades and a conical diffuser, this test rig is capable of generating a helical vortex.

In this work, cases for the simulation of the unsteady flow in the Stuttgart Swirl Generator with OpenFOAM are presented. To do so, this work firstly explains the background theory of the helical vortex phenomenon and the method of Computational Fluid Dynamics. Furthermore, the underlying test rig, geometry and mesh, discretisation schemes, algorithm for interequation coupling, solvers for the systems of linear algebraic equations and boundary and initial conditions are elucidated. As the flow in the Swirl Generator is characterized by a Reynolds number in a range of approximately $1.1*10^5$ to $1.8*10^5$ and unsteady flow features caused by the helical vortex, the treatment of turbulence plays a vital role for an appropriate setup of the cases. Here, common Reynolds-Averaged Navier-Stokes (RANS) turbulence models are assumed to be limited in their capability of dealing with unsteady flows. The method of Large Eddy Simulation (LES) on the other hand, would be capable of addressing these drawbacks, but are still unfavorable in terms of their demands on the computational effort. This work therefore, investigates hybrid strategies, which combine the advantages of both the RANS and LES approach. For this, cases based on two standard high-Reynolds number RANS models: k-ϵ and k-ω SST, and three hybrid RANS-LES models: k-ω SST SAS, $Spalart$-$Allmaras$ $DDES$ and $Spalart$-$Allmaras$ $IDDES$ are set up. In order to assess the results of the simulations, the present work shows a general evaluation of the results and compares experimental and simulated measurement data.

Keywords: Vortex rope, helical vortex, turbulence modeling, OpenFOAM, Stuttgart Swirl Generator, DES, SAS.

Kurzfassung

Der gegenwärtige Wandel in der Energieversorgung und die derzeit verfügbaren Speichertechnologien für elektrische Energie erfordern eine sichere und stabile Regulierung des elektrischen Netzes. Vor diesem Hintergrund werden Wasserkraftanlagen verstärkt abseits des optimalen Betriebspunktes eingesetzt. Diese Betriebspunkte sind mit einer drallbehafteten Strömung im Nachlauf der Turbine verbunden, welche zu einem Zusammenbruch der ursprünglich stabilen Strömung führen kann. Dieser Zusammenbruch erfolgt meist in Form eines helikalen Wirbels, auch Wirbelzopf genannt. Obwohl das physikalische Phänomen des Wirbelzopfs bereits seit Jahrzehnten analysiert wird, ist dessen Dynamik noch nicht komplett verstanden. Darüber hinaus, könnte der Wirbelzopf unerwünschte Auswirkungen auf das hydraulische System der Wasserkraftanlage haben. Während die auftretenden Druckschwankungen das Risiko eines Ermüdungsbruchs erhöhen, kann die Frequenz des Wirbelzopfs bei Erreichen der Eigenfrequenz des hydraulischen Systems zu Resonanzschwingungen führen. Des Weiteren wird der Wirkungsgrad des Kraftwerks durch die Präsenz von Rückströmungen vermindert. Um den Wirbelzopf zu analysieren und dessen Auswirkungen vorherzusagen, ist es deshalb im Interesse von Wissenschaft und Industrie, präzise Simulationen dieses physikalischen Phänomens zu erstellen. In Anbetracht dessen, hat das Institut für Strömungsmechanik und Hydraulische Strömungsmaschinen (IHS) an der Universität Stuttgart einen Drallgenerator konstruiert, um eine drallbehaftete Strömung vergleichbar mit der Strömung im Nachlauf von Turbinen, zu untersuchen. Mit der Hilfe von acht nicht-rotierenden Schaufeln und einem konischen Diffusor, ist dieser Versuchsstand in der Lage einen Wirbelzopf zu generieren.

Die vorliegende Arbeit präsentiert numerische Simulationen der Strömung im Stuttgarter Drallgenerator mit OpenFOAM. Hierfür werden zunächst die erforderlichen Grundlagen des Wirbelzopf-Phänomens und der Methode Computational Fluid Dynamics erklärt. Darüber hinaus wird der Versuchsstand, die verwendete Geometrie, das Gitter, Diskretisierungs-Methoden, Algorithmen für Druck-Geschwindigkeits-Kopplung, Algorithmen für Systeme linearer algebraischer Gleichungen sowie Rand- und Anfangsbedingungen erläutert. Da die Strömung in dem Drallgenerator durch eine Reynolds Zahl im Bereich von etwa $1,5*10^5$ bis $1,8*10^5$ und instationären Vorgängen, verursacht durch den Wirbelzopf, charakterisiert ist, spielt die Modellierung von Turbulenz eine große Rolle bei der Simulation der Strömung. Hierbei wird angenommen, dass Reynolds-Averaged Navier-Stokes (RANS) Turbulenzmodelle nur eingeschränkt fähig sind, instationäre Vorgänge der Strömung zu simulieren. Die Methode der Large Eddy Simulation (LES) hingegen, wäre in der Lage die instationären Strukturen der Strömung aufzulösen. Aufgrund des hohen Rechenaufwands sind LES-Simulationen derzeit jedoch nicht favorisierbar. Der Fokus dieser Arbeit ist deshalb die Analyse von hybriden Strategien, welche die Vorteile der RANS- und LES-Modelle kombinieren. Hierfür werden Simulationen basierend auf zwei high-Reynolds number RANS Turbulenzmodellen, und zwar: k-ϵ und k-ω SST sowie drei hybriden RANS-LES Turbulenzmodellen, und zwar: k-ω SST SAS, *Spalart-Allmaras DDES* und *Spalart-Allmaras IDDES* durchgeführt. Um die Ergebnisse der Simulationen zu beurteilen, wird abschließend eine Auswertung der Ergebnisse und ein Vergleich zwischen experimentellen und simulierten Daten präsentiert.

ACKNOWLEDGMENTS

NOMENCLATURE

Abbreviations

CD	Cross-Diffusion
CFD	Computational Fluid Dynamics
DDES	Delayed Detached Eddy Simulation
DES	Detached Eddy Simulation
DIC	Diagonal Incomplete-Cholesky
DILU	Diagonal Incomplete-LU
DNS	Direct Numerical Simulation
FOAM	Field Operation And Manipulation
FVM	Finite Volume Method
IDDES	Improved Delayed Detached Eddy Simulation
LDV	Laser Doppler Velocimetry
LES	Large Eddy Simulation
PBiCG	Preconditioned Bi-Conjugate Gradient
PCG	Preconditioned Conjugate Gradient
PIMPLE	Combination of PISO and SIMPLE
PISO	Pressure Implicit with Splitting of Operators
RANS	Reynolds Averaged Navier-Stokes
SAS	Scale Adaptive Simulation
SGS	Sub-Grid-Scale
SIMPLE	Semi-Implicit Method for Pressure-Linked Equations
SST	Shear Stress Transport
S-A	Spalart-Allmaras
WMLES	Wall-Modeled-LES

Greek

α	Turbulence model constant
β	Turbulence model constant
γ	Turbulence model constant
δ	Boundary layer thickness or Kronecker delta
Δ	Cell length or Size of the spatial filter in LES
ϵ	Dissipation rate of turbulence kinetic energy
ζ	Turbulence model constant
κ	Von Kármán constant
μ	Dynamic viscosity
ν	Kinematic viscosity
π	Pi
ρ	Density
σ	Turbulence model constant
τ	Stress
ϕ	Parameter representing an arbitrary property
χ	Turbulence model parameter
ψ	Turbulence model parameter

Mathematical Operators

cos	Cosine
div	Divergence
grad	Gradient
tanh	Hyperbolic tangent
∂	Partial derivative

Roman

A	Area or turbulence model constant
C	Turbulence model constant
F	Turbulence model function
D	Diameter
d	Day
h	Hour
I	Turbulence intensity
k	Turbulence kinetic energy
L	Length scale
l	Length
m	Meter
mm	Millimeter
P	Position or Production
p	Pressure
Q	Volume flow rate
R	Radius
Re	Reynolds number
S	Source or strain rate of the mean flow
s	Second
t	Time
u	Velocity vector
u, v, w	Velocity components in Cartesian coordinate system
x, y, z	Cartesian coordinates
y	Wall distance

Super- and Subscripts

i, j	Principal directions or tensor indices
t	Turbulent
x, y, z	Spatial directions or derivatives
w	Wall

CONTENTS

1 Introduction

The challenge of the 21st century to achieve a sustainable and climate-neutral energy supply leads to an increasing use of renewable energy sources like wind and solar energy. Since these sources are not predictable and volatile in nature, their availability does not necessarily meet the demand of energy at all times. This leads to fluctuations in the electric grid, which need to be compensated in order to maintain a stable net frequency. Due to the limitations of present storage capabilities for electric energy, it is common practice to regulate the electric grid with the help of hydro power plants. In response to this, hydro power plants are operated on a wide spread of off-design conditions. These operating points are however, closely associated with a vortex, caused by the swirling flow downstream of the turbine runner. Once the vortex reaches a critical swirl level, it breaks down - most often in form of a helical vortex. This helical vortex could have undesirable impacts on the power plant. While its frequency may lead to resonance vibration, resulting pressure fluctuations are assumed to be the reason for an increasing risk of fatigue failure due to dynamic pressure loads. Furthermore, the efficiency of the power plant decreases as a result of higher wall friction losses. Although the helical vortex has been investigated for several decades already, it is still far from being completely understood. It is therefore, in the interest of both science and industry to simulate the helical vortex precisely and to prevent destructive effects on the power plant.

In the light of that, the Institute of Fluid Mechanics and Hydraulic Machinery at the University of Stuttgart, Germany, constructed a test rig, which is able to generate a helical vortex, resembling the corresponding operating points of hydro turbines. To develop the helical vortex, eight non-rotating blades are placed symmetrically around a hub. The different operation conditions are then adjusted by changing the angle of these blades. Downstream of the Swirl Generator the flow enters a conical diffuser representing a draft tube and finally ends in a container. In order to analyze the helical vortex phenomenon, experiments as well as simulations based on the method of Computational Fluid Dynamics (CFD) have already been performed in the Stuttgart Swirl Generator. However, no simulations taking both the up and downstream into account simultaneously have been conducted with sufficient accuracy. The setup of simulations taking both the up and downstream into account simultaneously is therefore, the baseline of the present work. OpenFOAM[1], which refers to Open Field Operation and Manipulation, represents a CFD-code and is used for the numerical simulations conducted in this work. Since both the demand on reliable as well as accurate results of numerical simulations and their supply in a reasonable time are high, there is a need for an appropriate setup of the simulations. This setup is closely associated with the features of the underlying flow system and generally requires the generation of a computational domain as well as the specification of the numerical solution process. The flow in the Stuttgart Swirl Generator is characterized by a Reynolds number in a range of approximately $1.1 * 10^5$ to $1.8 * 10^5$ and is therefore turbulent. In addition, the occurrence of the helical vortex generally yields unsteady flow features. In order to allow for accurate simulations it is therefore, necessary to consider an appropriate treatment of turbulence in the simulation. To do so, the method of CFD offers different kinds of models, which generally differ in the extent to which they model turbulent flow structures and consequently in the required computational effort of the simulation. Due to their comparably low demands on the computational effort, so-called Reynolds-Averaged Navier-Stokes (RANS) simulations are widely used to simulate turbulent flow in industrial applications. These simulations are however, assumed to be limited in their capability of resolving unsteady flow structures. The application of so-called Large Eddy Simulations (LES) on the other hand, would be able to resolve these flow structures. Its demands on the computational effort are however, too high. Recent work is therefore, aiming at developing hybrid RANS-LES simulations, which combine the advantages of both approaches. Here, the Detached Eddy Simulation (DES) and its derivations as well as the Scale Adaptive Simulation (SAS), represent two of these hybrid strategies, which are used in this work to investigate the unsteady flow in the Stuttgart Swirl Generator.

1.1 Objectives and Limitations

To begin with, the goal of this work is to set up cases for the Stuttgart Swirl Generator and to provide a baseline for further investigations for both scientific and industrial realm. To do so, the geometry and the mesh, which have been provided by the Institute of Fluid Mechanics and Hydraulic Machinery, University of Stuttgart, are modified for the purpose of this work. Furthermore, the numerical setup including discretisation

---

[1]Throughout the text OpenFOAM refers to OpenFOAM®. OpenFOAM and OpenCFD are registered trademarks of SGI Corp.

schemes, algorithms for pressure-velocity coupling, solvers for systems of linear algebraic equations and the specification of boundary as well as initial conditions is considered.

As explained in the Introduction, the treatment of turbulence plays an important role in simulating the flow in the Stuttgart Swirl Generator. The focus of this work is therefore, the investigation of turbulence models, which are provided by the CFD-code used in this work. For this, two high-Reynolds number RANS models, namely, k-ϵ and k-ω *SST* and three hybrid models, namely, k-ω *SST SAS*, *Spalart-Allmaras DDES* and *Spalart-Allmaras IDDES* are applied. In order to assess the simulations in terms of their accuracy, the results are evaluated and compared with experimental measurements. These measurements have been carried out by the Institute of Fluid Mechanics and Hydraulic Machinery at the University of Stuttgart and include pressure measurements at the wall before as well as after the draft tube, and velocity measurements along three radial lines placed 47.5 mm downstream of the blade axes. Alongside to the evaluation of the results, the helical vortex phenomenon is analyzed.

The simulations are performed with OpenFOAM 1.6-ext. In accordance with the experimental measurements, the flow for a blade angle of 30° and a discharge of 25 $\frac{m^3}{h}$ as well as 40 $\frac{m^3}{h}$ is simulated.

1.2 Outline of the Thesis

Section 2 elucidates the theoretical background, which is required for the present work and is divided into four subsections. While the first subsection explains the theory regarding the underlying flow system and the helical vortex phenomenon, the following two subsections concern the theory of the method of Computational Fluid Dynamics as well as turbulence and its modeling including an explanation of the Reynolds-Averaged Navier-Stokes (RANS), Large Eddy Simulation (LES) as well as hybrid RANS-LES approach. The fourth subsection finally gives an overview of OpenFOAM, the CFD-code used in this work.

A presentation of the cases and results is given in **section 3**, which contains four subsections. The first two subsections give a description of the computational domain, i.e. the underlying geometry as well as the mesh including a description of the test rig, and the numerical setup, i.e. the general numerical setup and boundary as well as initial conditions. The last two subsections show the results of the simulations with one subsection for each value of discharge. These subsections include an evaluation of the development as well as performance of the simulations, an analysis of the flow as well as the turbulence models and a comparison of the results with the experimental data.

Section 4 finally provides the conclusion and prospects of this work. In order to maintain a better overview, the explanation of the turbulence models used in this work, the information about the experimental measurements and figures regarding the computational domain are given in **Appendix A**, **B** and **C** respectively.

2 Theoretical Background

This section elucidates the theoretical background and is divided into four subsections. The first subsection gives an overview of the flow system and the helical vortex phenomenon. The method of Computational Fluid Dynamics as well as turbulence and its modeling are furthermore, explained in subsection 2.2 and 2.3 respectively. In subsection 2.4 an introduction to OpenFOAM is given.

2.1 Flow System and Helical Vorticies

The hydraulic system of hydro power plants usually consists of pipes, which supply the head-water to the turbine and subsequently lead it away. In addition, it is common practice to install a so-called draft tube in the downstream of the turbine in order to improve the efficiency of the power plant. The draft tube converts the remaining kinetic energy into static pressure. This conversion is practically achieved by a streamwise increase of the cross-sectional area of the draft tube [6].

Due to the working principle of reaction turbines, the flow upstream of the turbine needs to be imposed with a swirling component. This is usually done by guide vanes and a spiral casing - if available. Depending on the operating point of the turbine, this swirl is then more or less extracted by the blades of the turbine runner. In general, the design of the turbine aims to have no or only a minor swirling component downstream of the runner at its best operating point. When running the turbine beside this operating point, it is due to the construction conditions not possible to avoid a swirling flow downstream of the turbine. In contrast to Kaplan turbines, where the angle between the guide vanes and the blades can be adjusted to a certain extent, the angle in Francis turbines is fixed and usually designed for the best operating point of the turbine. Therefore, especially Francis turbines tend to yield a strong swirling flow downstream of the runner at part load operation. Above a critical swirl level a series of events takes place, which lead to a breakdown of this vortex. Here, the formation of a stagnation point is assumed to be the onset of the vortex breakdown. Figure 2.1.1 shows the physical process, which leads to the stagnation point [6][13][11].
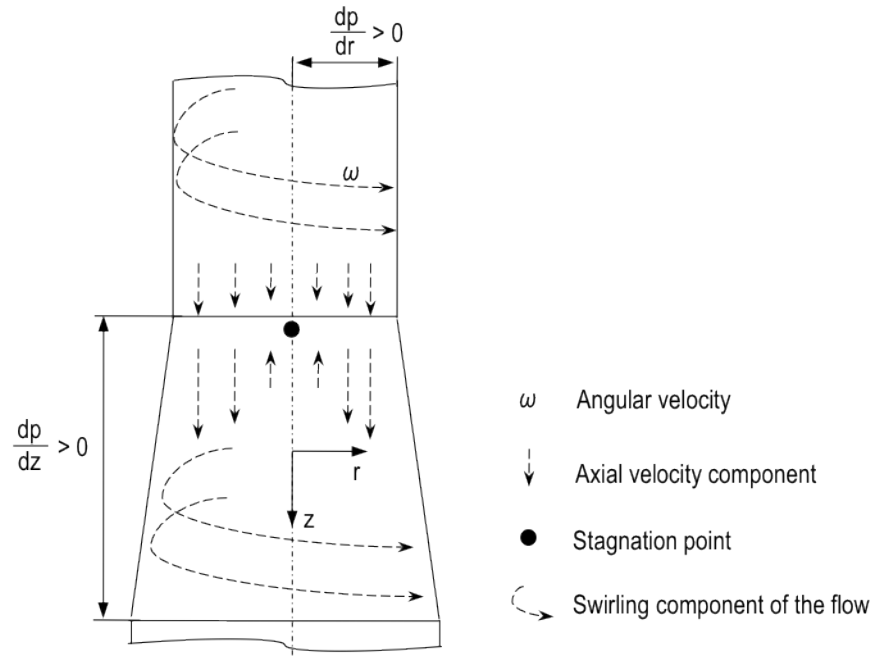
Figure 2.1.1: *Schematic figure of the basic flow features, which lead to the vortex breakdown.*

Figure 2.1.1 depicts a swirling flow through a flow system including a straight pipe followed by a conical diffuser. The figure presents a stagnation point, which is assumed to occur at the beginning of the draft tube. This exemplary assumption is likely, since the working principle of the draft tube enhances the formation of a stagnation point. As indicated by the direction of the axial velocity component, the point of stagnation

is formed by a reversed flow in the center-region of the flow system. This counterflowing region is mainly determined by the radial and axial pressure gradient shown in figure 2.1.1. The radial pressure gradient is a result of the swirling component of the flow, which introduces a centripetal force. This force, in turn, yields a positive pressure gradient in radial direction, i.e. the radial pressure decreases with a decreasing radius. As indicated in figure 2.1.1, the radial pressure gradient consequently influences the axial velocity component of the flow, which decreases towards the center-line of the flow system. The axial pressure gradient on the other hand, is due to the working principle of the draft tube, which converts kinetic energy into static pressure. This leads to a streamwise increase of static pressure as the flow flows through the draft tube. Consequently, the pressure downstream is higher than upstream of the draft tube. Once this pressure gradient is high enough, it causes the flow to reverse the actual flow direction. Due to the radial pressure gradient, this counterflowing region occurs most likely in the center-region of the flow system. The resulting stagnation point acts as a disturbance on the initially stable flow, which finally could lead to a vortex breakdown. In hydro power plants, this breakdown most often forms a *helical vortex*, also called *vortex rope*. Figure 2.1.2 presents a schematic picture of the helical vortex based on the flow system presented in figure 2.1.1.
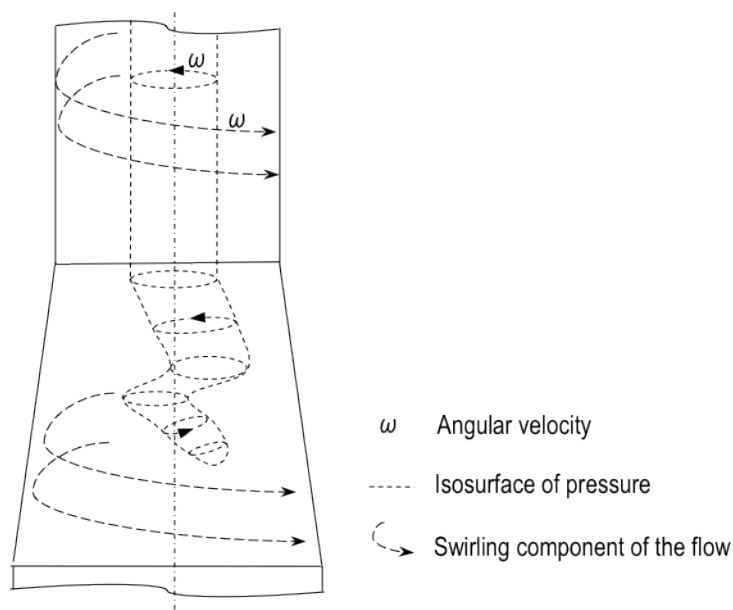

Figure 2.1.2: *Schematic figure of a helical vortex, visualized by an isosurface of pressure.*

As can be seen in figure 2.1.2, the helical vortex can be visualized by isosurfaces of pressure. Based on the flow system of figure 2.1.1 and provided that the stagnation point yields a vortex breakdown, the isosurface of pressure would contain two different sections: a quasi-cylindrical, axis-symmetric section in the region of the straight pipe and a helical section, which rotates asymmetrically around the center-line of the draft tube. A consequence of this is, that the helical vortex could have undesirable impacts on the entire power plant. Firstly, it causes a dynamic load on the hydraulic system. This in turn increases the risk of **fatigue failure**. Secondly, the frequency of the helical vortex might meet the eigenfrequency of the hydraulic system, which would lead to **resonance vibration**. Finally, the reversed flow leads to a **reduction of the draft tube efficiency**. This reduction is caused by the counterflowing region, which includes a negative velocity component in the axial direction. Due to the conservation of mass, the flow in streamwise direction has to increase. Consequently, the efficiency of the power plant decreases as a result of higher wall friction losses [6].

The previous paragraphs gave an overview of the basic physical process, which leads to the breakdown of the vortex rope. It should however be stressed, that the explanations are based on a schematic figure and do not claim to give a complete picture of the dynamic of the physical phenomenon. In fact, the dynamic of the helical vortex depends on a more complex system of parameters and is far from being completely understood [11].

2.2 Computational Fluid Dynamics

The idea of Computational Fluid Dynamics (CFD) is to analyze a system involving fluid flow, heat transfer or associated phenomena with the help of computer-based simulations. First applied by the aerospace industry in the 1960s, the method of CFD has become an important component in research and a wide range of industrial applications since the 1990s. One reason for the importance of CFD is, that computer-based simulations allow a better insight in a system, especially where experimental measurements can not be executed. This feature enables research as well as industry to have a better understanding of fluid flow systems. Furthermore, the total cost for numerical investigations is generally lower than for experiments [15].

The core of CFD are numerical algorithms, which are used to solve the governing equations of physical systems. It is common practice to divide the CFD-process into three main elements: **pre-processing**, **solving** and **post-processing**. Basically the first element, the **pre-processing**, involves the preparation of the required input for the numerical solution process. The most important steps at this point of the CFD-process are [15]:

- Mathematical expression of physics

- Definition of the geometry

- Generation of the mesh

- Specification of the fluid properties

- Determination of the boundary and initial conditions

Prior to the numerical solution of the flow system, it is necessary to develop mathematical expressions for the underlying physics. For isothermal fluid flow, these fundamental equations are derived from conservation laws of physics, namely, the conservation of mass and momentum. Equations 2.2.1 to 2.2.4 represent the governing equations for an incompressible fluid, e.g. water [15].

$$\text{div}\,\mathbf{u} = 0 \tag{2.2.1}$$

$$\frac{\partial u}{\partial t} + \text{div}(u\mathbf{u}) = \text{div}(\mu\,\text{grad}\,u) - \frac{\partial p}{\partial x} + S_{Mx} \tag{2.2.2}$$

$$\frac{\partial v}{\partial t} + \text{div}(v\mathbf{u}) = \text{div}(\mu\,\text{grad}\,v) - \frac{\partial p}{\partial y} + S_{My} \tag{2.2.3}$$

$$\frac{\partial w}{\partial t} + \text{div}(w\mathbf{u}) = \text{div}(\mu\,\text{grad}\,w) - \frac{\partial p}{\partial z} + S_{Mz} \tag{2.2.4}$$

The four equations, eqns. 2.2.1 to 2.2.4, describe the motion of three-dimensional, unsteady fluid flow of an incompressible, Newtonian fluid. The l.h.s. of the first equation, the so-called **continuity equation**, describes the divergence of the velocity in each point. The latter three equations on the other hand, the so-called **Navier-Stokes equations**, represent the conservation of momentum in each direction, x, y and z. The l.h.s. of the Navier-Stokes equations refers to the material derivative and describes the change of the velocity based on the *Eulerian* approach. The material derivative can be split into the temporal and spatial derivative of the velocity. Here, the spatial derivative refers to the transport of the velocity due to convection. While the first term on the r.h.s. of the momentum equations describes the rate of increase of the velocity due to diffusion, the second and third term represent source terms in the Navier-Stokes equations. The term S_M represents an arbitrary source, which has an influence on the motion of the fluid [15].

In order to discretize these partial differential equations, which is done in the next element of the CFD-process, it is necessary to define the geometry of the physical system. The geometry corresponds to the computational domain, which is taken into account by the CFD-code and should therefore consist of all geometry features, which have an influence on the flow of the fluid. However, it is by means of the following mesh generation not always possible to mesh every detail of the geometry. In that case, the definition of the geometry also includes the consideration of appropriate simplifications of the geometry. As a part of the discretisation process, the computational domain is furthermore discretized in space. This is done by the generation of the mesh, which consists of small cells throughout the computational domain. These cells practically represent finite fluid elements. During the numerical solution process, the pressure and velocity for each of these elements are calculated in the corresponding cell center. It seems logical to store these properties at the same point of the grid. This however, could lead to unphysical behavior in the simulations as a result of a highly non-uniform

pressure field [15]. The application of a so-called *staggered grid* or an appropriate interpolation method represent two methods to avoid this unphysical behavior. The staggered grid refers to the storage of values for pressure and velocity at different locations. For complex geometries, the staggered grid is however not favorable in terms of mesh generation and organization. The application of an appropriate interpolation method on the other hand, is capable of avoiding unphysical behavior on a *co-located grid* [8]. As can be seen in the explanation of the next step in the CFD-process, the discretisation of the partial differential equations leads to a system of linear algebraic equations. In order to be able to solve this system, it is furthermore necessary to specify the fluid properties and the boundary as well as initial conditions of the physical system [15]. In the case of an incompressible fluid, the specification of the fluid properties reduces to the specification of the molecular viscosity and the fluid density. The boundary conditions determine the values of the transported properties at all boundaries of the computational domain. From a numerical point of view, there are principally two kinds of boundary conditions, which can be built into the system of algebraic equations: *fixed value* and *fixed gradient* boundary conditions. While the fixed value condition, also called *Dirichlet* boundary condition, prescribes the value of the corresponding property on the boundary, the fixed gradient condition, also called *Neumann* boundary condition, prescribes the gradient of the property normal to the boundary. Based on these two kinds of numerical boundary conditions, the physical boundary conditions of the computational domain are specified. In general, the physical boundaries of the computational domain consist of an inlet, outlet and surrounding walls. The initial conditions on the other hand, represent a first guess of the corresponding property values. This guess is required to start the iterative algorithm and allows the user to improve the solution process by specifying values, which are as close to the correct value as possible [7].

The second stage, the stage of **solving**, concerns the primary goal of finding a solution for the equations. With the *Finite Difference*, *Finite Element* and *Finite Spectral Method* there are several techniques available, which can be applied in order to solve the equations. The following explanations refer to the *Finite Volume Method* (FVM). This method represents a special formulation of the Finite Difference Method and is used by OpenFOAM. The second stage of the CFD-process consists of the following steps [15]:

- Discretisation of the partial differential equations

- Application of an iterative algorithm for the pressure-velocity coupling

- Solution of the resulting system of algebraic equations

The term discretisation generally refers to the expression of the partial differential equations with the help of neighboring-values in space or previous values in time. This transformation process yields a system of algebraic equations and can be divided into two steps: the *discretisation of the solution domain*, which has been done by the generation of the mesh, and the *discretisation of the equations*. In the case of an unsteady flow, the underlying equations have to be discretized not only in space but also in time. In the FVM, the spatial discretisation of the equations is principally done by the integration of the underlying equations over a control volume. The volume integral of the convection and diffusion terms is furthermore replaced by a surface integral. This transformation is based on the *Gauss Theorem* and introduces face values into the system of algebraic equations. Analogous to the spatial discretisation, the temporal discretisation starts with a split of the time interval into a finite number of timesteps. The discretisation of the equations in time is then done by the integration of the governing equations in time [7][15].

Based on this general discretisation process, several discretisation schemes are available for the discretisation of the governing equations. These schemes mainly differ in their accuracy and boundedness of the solution. In terms of accuracy, it is necessary for the order of the discretisation to meet or exceed the order of accuracy of the underlying partial differential equation. Since the diffusion term of the Navier-Stokes equations contains the second derivative of the velocity in space, the Navier-Stokes equations are second-order equations. In general, it would be favorable, to obtain the same order of accuracy from the discretisation of the equation. The application of second-order accurate discretisation schemes is however, not always possible. In convection dominated flows for example, it is necessary to take the direction of the flow into account in order to avoid unphysical oscillations of the solution. This adjustment generally leads to a deviation from the order of accuracy of the discretisation. The discretisation process yields a system of algebraic equations, which can be written as follows [7][15]:

$$[A] * [x] = [b] \tag{2.2.5}$$

The l.h.s. of eqn. 2.2.5, consists of the product of the matrix A, which refers to the coefficients multiplying the discretized property values according to the chosen discretisation procedure, and the vector x, which represents

the property value in each cell of the mesh. The vector b on the r.h.s. contains the source terms of the equations. In theory, this system of equations would be solved with an appropriate solver in the next step. The nature of the underlying equations however, requires further considerations. As can be seen in eqns. 2.2.2 to 2.2.4, the momentum equations contain a **non-linear** term and the flow properties of interest, pressure and velocity, are intrinsically **coupled** over the three momentum equations. Moreover, the underlying system of equations does not provide an equation for the pressure. In order to overcome these problems, several iterative algorithms have been developed. These algorithms generally consist of so-called *momentum predictors* and *pressure correctors*. The momentum predictor refers to the calculation of a new velocity field with the momentum equations, based on the pressure of the previous timestep or the initial guess. The pressure corrector on the other hand, corresponds to a correction of the pressure field with the help of the continuity equation. The Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) as well as the Pressure Implicit with Splitting of Operators (PISO) and their derivations are the most common algorithms for the inter-equation coupling [7]. The SIMPLE algorithm consists principally of a momentum predictor and a following pressure corrector step. In order to improve the convergence of the calculation, the SIMPLE algorithm uses so-called under-relaxation factors for pressure and velocity. These factors are used to calculate the value for the next timestep by blending the actual value with the old value of the pressure and velocity respectively. The PISO method on the other hand, consists of a momentum predictor, followed by two corrector steps for the pressure. The resulting systems of algebraic equations finally have to be solved with the help of an appropriate solver. In general, this can be done with either *direct* or *iterative* methods. While the direct method gives the solution of the system in a finite number of arithmetic operations, the iterative method improves an initial guess of the solution, until convergence is achieved. The convergence of the iteration process however, depends on the diagonal dominance of the matrix A of equation 2.2.5 [7][15].

After solving the equations, the final step of the CFD-process is the **post-processing**. This process helps the user to visualize and assess the numerical results and ensures a careful consideration of the results.

2.3 Turbulence and Its Modeling

Above a critical Reynolds number, which is generally defined as the ratio of a characteristic velocity times a characteristic length to the kinematic molecular viscosity, a series of events takes place, which may lead to a so-called **turbulent** flow structure. Although the turbulent regime is not yet clearly defined, the achievements of analytical and experimental work allow the description of the origin of turbulence as well as its characteristics and influences to the flow field. The following paragraphs give an overview of the most important aspects of turbulence to provide a baseline for the understanding of its modeling.

Flows with a Reynolds number, which is below a so-called critical Reynolds number Re_{crit}[1], are observed to be stable and laminar, i.e. the flow layers pass each other in an orderly fashion. Above this critical Reynolds number however, initially stable flow becomes sensitive to disturbances, e.g. obstacles or wall roughness. During the transition process, these disturbances are amplified and most likely accumulate to turbulent spots, before they finally propagate through the flow field. The final state of the transition process is called turbulent flow. In contrast to laminar flow, turbulent flow is characterized by a random and chaotic behavior of the flow. Moreover, the flow contains rotational flow structures, so-called **turbulent eddies**, which effectively change the entire flow field. This can among other things be illustrated by means of the distribution of the mean velocity in fully developed laminar and turbulent channel flow, which is shown in figure 2.3.1 [15].
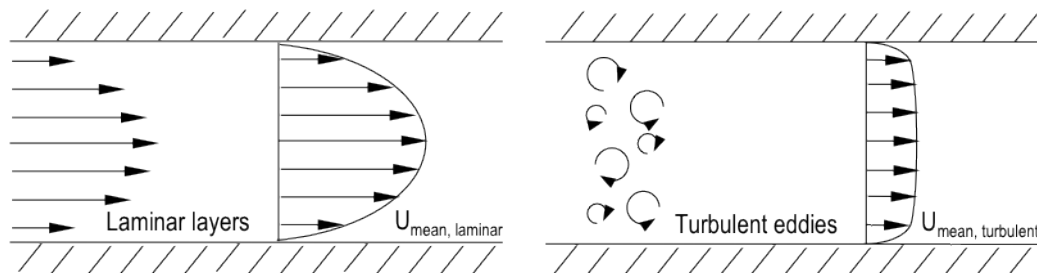

Figure 2.3.1: *Schematic figure of the flow structure and mean velocity distribution of a channel flow. Left: laminar flow; right: turbulent flow.*

---

[1]The critical Reynolds number for pipe-flow is 2300 [15].

Figure 2.3.1 gives a simplified picture of the eddies of a turbulent flow, which are superimposed on the flow layers. These eddies appear in a wide range of length and time scales. The length scale reaches from the largest scales, which are of the order of the geometry, to the smallest scales, the so-called *Kolmogorov* scales. Within that range, the kinetic energy from the large scales, which extract energy from the mean flow, is lost to slightly smaller scales. This process is called **energy cascade** and transfers most of the energy to the *Kolmogorov* scales, where the energy is finally dissipated [2]. Furthermore, as can be seen in figure 2.3.1, the mean velocity distribution of a turbulent flow contains, in contrast to laminar flow, high gradients as the flow reaches towards the wall. These gradients principally require an appropriate refinement of the mesh towards the wall. However, as can be seen in the explanation of the Reynolds-Averaged Navier-Stokes approach, the application of so-called *wall functions* can replace a resolution of the region close to the wall and consequently reduce the number of cells in the mesh. These wall functions are based on the theory of a fully developed boundary layer, which can be split into different layers. Figure 2.3.2 gives an overview of the layers in the inner region of a fully developed boundary layer.
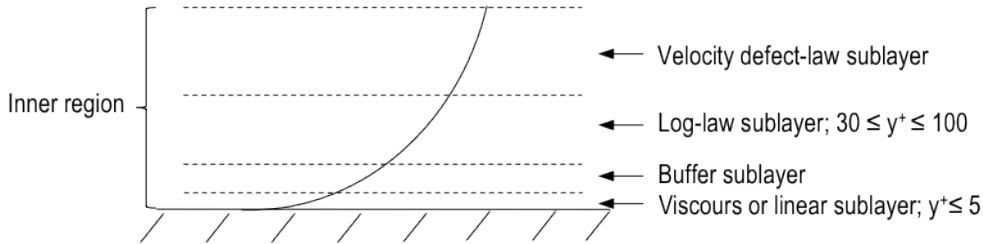

Figure 2.3.2: *Schematic structure of the inner region of a fully developed turbulent boundary layer.*

Figure 2.3.2 represents the inner region of a fully developed turbulent boundary layer, which is assumed to contain 10-20% of the entire boundary layer. As can be seen, the inner region can be divided into four sublayers: the *viscous* or *linear*, *buffer*, *log-law* and *velocity defect-law* sublayer. These sublayers can be determined in terms of the so-called y^+-value. This value refers to the dimensionless wall distance and is defined by the equation

$$y^+ = \frac{u_\tau y_P}{\nu},\tag{2.3.1}$$

whereas u_τ refers to the *friction velocity*, y_P to the *wall distance of point P* and ν to the local *kinematic viscosity*. The application of wall-functions, which are used by the RANS models in this work, requires the placement of the first cell point as measured from the wall in the log-law sublayer. This sublayer is assumed to reach from a y^+-value of 30 to 100 [14][15].

As is clear from the previous paragraphs, it is necessary to take turbulence into account in order to simulate turbulent flow. In general, the solution of the Navier-Stokes equations, eqns. 2.2.2 to 2.2.4, would be capable of doing so. This approach refers to the so-called Direct Numerical Simulation (DNS). In order to simulate all length and time scales of the turbulent flow however, the method of DNS would require a corresponding high mesh resolution as well as an appropriate small timestep of the simulation. These requirements lead to an enormous computational effort and consequently computational cost for the simulation of complex flow systems. It is therefore, necessary to apply so-called **turbulence models**, which model the influence of turbulence to the flow. The Reynolds-Averaged Navier-Stokes simulation (RANS) as well as the Large Eddy Simulation (LES), represent two common kinds of simulations, which use turbulence models. Since these simulations are however, limited either due to the capability of resolving unsteady flow structures or high demands on the computational effort, recent work is aiming to develop hybrid strategies to combine the advantages of both RANS and LES models. While the RANS, LES and hybrid approaches are explained within this subsection, the formulations of the turbulence models used in this work are shown in Appendix A.

2.3.1 Reynolds-Averaged Navier-Stokes Simulation

For the sake of analyzing the influence of turbulence on the mean flow, which is sufficient for many industrial applications, the flow can be separated into a mean and fluctuating part. This process is called *Reynolds-decomposition* and is the starting point of the Reynolds-Averaged Navier-Stokes (RANS) approach. The Reynolds-decomposition reads

$$\phi = \bar{\phi} + \phi',\tag{2.3.2}$$

whereas ϕ represents an arbitrary property, which is divided into its mean value $\bar{\phi}$ and its fluctuating part ϕ'. The application of the Reynolds-decomposition to eqns. 2.2.1 to 2.2.4, yields the so-called Reynolds-Averaged Navier-Stokes equations, which read

$$\text{div } \bar{u} = 0, \tag{2.3.3}$$

$$\frac{\partial \rho \bar{u}}{\partial t} + \text{div}(\rho \bar{u} \mathbf{\bar{u}}) = \mu \, \text{div}(\text{grad } \bar{u}) - \frac{\partial \bar{p}}{\partial x} + \left[\frac{\partial \left(-\rho \overline{u'u'} \right)}{\partial x} + \frac{\partial \left(-\rho \overline{u'v'} \right)}{\partial y} + \frac{\partial \left(-\rho \overline{u'w'} \right)}{\partial z} \right], \tag{2.3.4}$$

$$\frac{\partial \rho \bar{v}}{\partial t} + \text{div}(\rho \bar{v} \mathbf{\bar{u}}) = \mu \, \text{div}(\text{grad } \bar{v}) - \frac{\partial \bar{p}}{\partial y} + \left[\frac{\partial \left(-\rho \overline{u'v'} \right)}{\partial x} + \frac{\partial \left(-\rho \overline{v'v'} \right)}{\partial y} + \frac{\partial \left(-\rho \overline{v'w'} \right)}{\partial z} \right], \tag{2.3.5}$$

$$\frac{\partial \rho \bar{w}}{\partial t} + \text{div}(\rho \bar{w} \mathbf{\bar{u}}) = \mu \, \text{div}(\text{grad } \bar{w}) - \frac{\partial \bar{p}}{\partial z} + \left[\frac{\partial \left(-\rho \overline{u'w'} \right)}{\partial x} + \frac{\partial \left(-\rho \overline{v'w'} \right)}{\partial y} + \frac{\partial \left(-\rho \overline{w'w'} \right)}{\partial z} \right]. \tag{2.3.6}$$

As can be seen, the Reynolds-Averaged Navier-Stokes (RANS) equations have a similar form as the Navier-Stokes equations. However, the Reynolds-decomposition introduced specific source terms on the r.h.s. of eqns. 2.3.4 to 2.3.6. These terms represent the so-called *Reynolds-stresses*, which are unknown. In order to avoid a closure problem for the system of equations, it is necessary to determine these terms. This can be done with the help of the so-called *Boussinesq-assumption*. Analogous to the molecular viscosity, the Boussinesq-assumption defines a so-called *eddy viscosity* to link the Reynolds-stresses with the mean rate of deformation of the fluid. This assumption yields

$$\mu_t = \frac{\nu_t}{\rho} \tag{2.3.7}$$

with the turbulent eddy viscosity μ_t and the turbulent kinematic eddy viscosity ν_t. On dimensional grounds, the turbulent kinematic eddy viscosity can be expressed by a product of a turbulence *velocity* and *length* scale. Based on this dimensional analysis, the k-ϵ and k-ω *SST* model, used in this work, define a turbulence velocity scale with the help of the turbulence kinetic energy k. Furthermore, a turbulence length scale is determined with the help of a second variable, the turbulence dissipation rate ϵ and the turbulence frequency ω respectively. Depending on the number of additional transport equations, which have to be solved during the solution process, the k-ϵ as well as the k-ω SST model can be allocated to the so-called two-equation RANS models. Furthermore, RANS models can be split into *low-Reynolds number* and *high-Reynolds number* models respectively. These kinds of RANS models generally differ in their treatment of the near wall region. While low-Reynolds number models integrate to the wall, high-Reynolds number models apply so-called *wall functions* to calculate the transported properties with algebraic equations. These wall functions are based on the theory of the different regions in a fully developed turbulent boundary layer, which are shown in subsection 2.3. Figure 2.3.3 illustrates the difference between the application of a low- and high-Reynolds number RANS model [15].
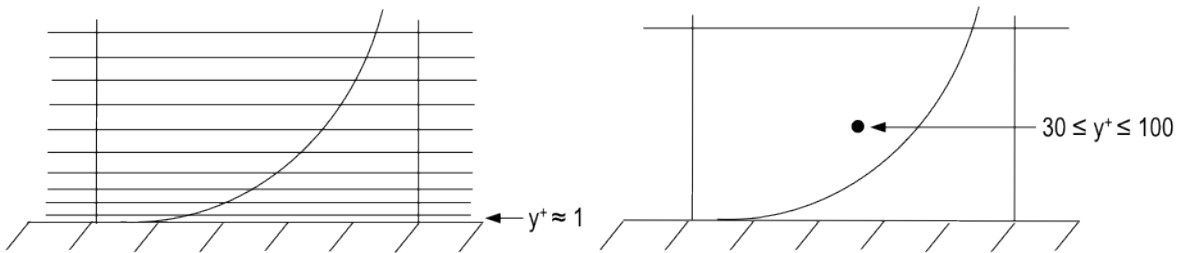

Figure 2.3.3: *Schematic figure of the mesh density for Reynolds-Averaged Navier-Stokes simulations in the boundary layer. Left: low-Reynolds number model; right: high-Reynolds number model.*

As shown in figure 2.3.3, the placement of the first grid point as measured from the wall in the log-law region significantly reduces the number of cells compared to the resolution of the entire boundary layer. The illustration of the cell centers in the left picture of figure 2.3.3 has been omitted to maintain a better overview.
For simulations where the attention is focused on the mean flow and the influence of the turbulence to its properties, eddy viscosity models are widely used due to their modest computational effort. Moreover, considering a fully developed turbulent flow, further savings can be achieved with the application of wall functions. The RANS models are however, assumed to be limited in their capability of resolving unsteady flow structures. This drawback can mainly be addressed to the determination of the turbulence length scale, which

is used to compute the Reynolds-stresses. This length scale is primarily determined by the strain-rate S, which is introduced in the source terms of the additional transport equations. As a consequence of that, only one length scale can be obtained from the standard two-equation RANS models. Furthermore, the length scale depends on the diffusion terms, which yields a correlation of the length scale and the shear-layer thickness. Standard two-equation models therefore, always return the shear-layer thickness as an appropriate length scale, independent of any resolved content. As a consequence of that, these models are tending to dampen the unsteady structures of the flow, which corresponds to too high an amount of turbulent eddy viscosity. The application of a different approach to model the turbulence would therefore be favorable for the simulation of the flow in the Stuttgart Swirl Generator [15][17].

2.3.2 Large Eddy Simulation

The idea of the Large Eddy Simulation (LES) approach is to resolve the larger eddies of the turbulent flow while modeling the influence of the smaller eddies with a turbulence model. To do so, a spatial filtering function is applied to the Navier-Stokes equations. This spatial filtering operation is defined as

$$\bar{\phi}(x,t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(x,x',\Delta)\phi(x',t)dx_1'dx_2'dx_3', \tag{2.3.8}$$

whereas $G(x,x',\Delta)$ represents the filter function and Δ the cutoff width of the filter. It should be stressed that the overbar in eqn. 2.3.8 indicates spatial filtering and not time averaging. The application of the filtering operation yields the LES continuity and momentum equations. The final form of these equations reads

$$\operatorname{div}\bar{u} = 0, \tag{2.3.9}$$

$$\frac{\partial \rho\bar{u}}{\partial t} + \operatorname{div}(\rho\bar{u}\mathbf{u}) = \mu\operatorname{div}(\operatorname{grad}\bar{u}) - \frac{\partial\bar{p}}{\partial x} - \left[\operatorname{div}(\rho\overline{u\mathbf{u}}) - \operatorname{div}(\rho\bar{u}\bar{\mathbf{u}})\right], \tag{2.3.10}$$

$$\frac{\partial \rho\bar{v}}{\partial t} + \operatorname{div}(\rho\bar{v}\mathbf{u}) = \mu\operatorname{div}(\operatorname{grad}\bar{v}) - \frac{\partial\bar{p}}{\partial y} - \left[\operatorname{div}(\rho\overline{v\mathbf{u}}) - \operatorname{div}(\rho\bar{v}\bar{\mathbf{u}})\right], \tag{2.3.11}$$

$$\frac{\partial \rho\bar{w}}{\partial t} + \operatorname{div}(\rho\bar{w}\mathbf{u}) = \mu\operatorname{div}(\operatorname{grad}\bar{w}) - \frac{\partial\bar{p}}{\partial z} - \left[\operatorname{div}(\rho\overline{w\mathbf{u}}) - \operatorname{div}(\rho\bar{w}\bar{\mathbf{u}})\right]. \tag{2.3.12}$$

Analogous to the time filtering in the RANS approach, the spatial filtering introduces a new term on the r.h.s. of the LES Navier-Stokes equations. These terms represent the so-called *Sub-Grid-Scale (SGS) stresses*, which result from the filtering of the Navier-Stokes equations in space. The *SGS stresses* have to be modeled based on the same argument as for the *Reynolds-stresses*. This is done by a so-called *Sub-Grid-Scale (SGS) model*. The LES approach represents a remedy for the drawbacks of the two-equation RANS models. However, its demands on the computational effort are quite high. This problem can mainly be addressed to the requirements on the cells of the mesh. In order to resolve turbulent structures, the resolution of the mesh should be similar in each direction, i.e. the cells should ideally be cubic. The resolution of the high velocity gradient close to the wall shown in figure 2.3.3 therefore, requires a similar refinement of the mesh in wall parallel as well as span-wise direction. This refinement, in turn, has an effect on the entire mesh and consequently increases the number of cells. Figure 2.3.4 illustrates the mesh resolution for LES in the boundary layer schematically.
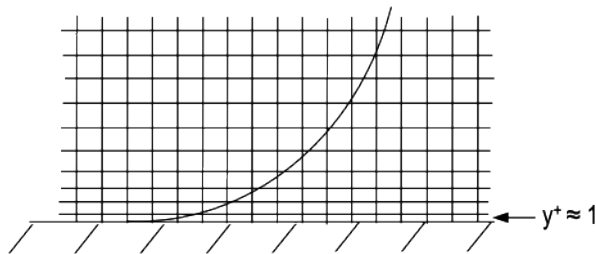

Figure 2.3.4: *Schematic figure of the mesh density for Large Eddy Simulations in the boundary layer.*

As can be seen in figure 2.3.4, the number of cells in a mesh for LES is significantly higher than in a mesh for RANS simulations, which can be seen in the explanation of the RANS approach. Furthermore, the timestep of the simulation has to be reduced in order to maintain a Courant number less than 1. This yields simulations with an unfavorable high computational effort.

2.3.3 Hybrid Simulation

As is clear from the explanations of the RANS and LES strategies, the LES approach would be favorable in terms of simulating the unsteady flow of the helical vortex. At high Reynolds numbers and complex geometries however, this approach leads to an undesirably high computational effort. The RANS approach on the other hand has, especially in the boundary layer, lower demands on the mesh resolution, but is due to its approach limited in its ability to resolve unsteady structures of the flow field. Recent work is therefore aiming to combine the advantages of both methods with hybrid strategies. The general idea of these hybrid methods is to provide turbulence models, which are capable of allowing LES-predictions to a certain extent with a lower computational effort, in comparison with full LES simulations. This is achieved by leaving the remainder of the flow to a RANS model. Figure 2.3.5 illustrates the application of hybrid methods in contrast to full LES and full RANS simulations based on an exemplary flat plate boundary layer flow.
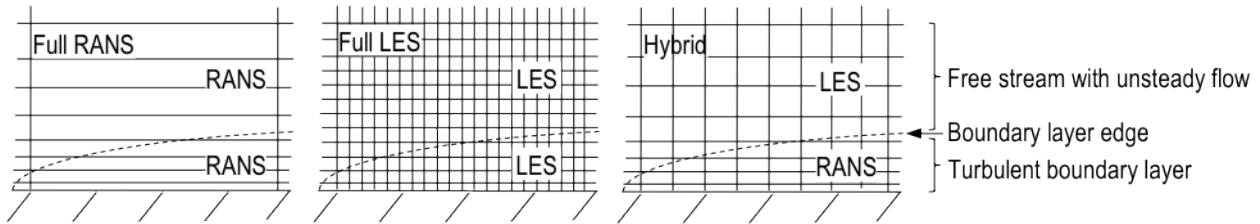

Figure 2.3.5: *Illustration of the differences between full LES, full RANS and hybrid methods based on an exemplary turbulent flat plate boundary layer flow with unsteady flow in the free stream. Left: full LES; middle: full RANS; right: hybrid method. All pictures show the same section.*

Figure 2.3.5 shows three different mesh densities of different kinds of simulations. It can be seen, that the use of a RANS model in the entire domain requires a significantly lower mesh refinement in wall parallel direction, in contrast to the application of a LES model. This applies as well for the span-wise direction. It is clear from theory of the RANS simulation, that the application of a RANS model in the entire domain is however, not favorable in order to resolve the unsteady flow in the free stream. As to be seen in the middle picture of figure 2.3.5, the application of a LES model on the other hand requires a relatively high mesh resolution, which can be addressed to the fact, that the mesh for LES simulations should ideally contain cubic cells. This leads to a similar mesh resolution in all dimensions of the mesh, i.e. in direction to the wall, wall parallel and span-wise direction. It can be seen in the middle picture of figure 2.3.5, that the relatively fine resolution of LES simulations in wall direction close to the wall, consequently determines the resolution of the entire domain and the number of cells of the mesh respectively. As to be seen in the right picture of figure 2.3.5, the application of hybrid RANS-LES simulations represents a remedy for both the limited capability of RANS simulations as well as the relatively high computational effort of LES simulations. While the "LES-mode" of the hybrid model allows for resolving the unsteady flow in the free stream, the relaxation of the requirements of the mesh resolution, lowers the computational effort, in contrast to full LES simulations. This relaxation falls back on the application of the "RANS-mode" in the boundary layer, which does not require cubic cells in the corresponding region. Based on this general idea of hybrid RANS-LES methods, different approaches are available. The method of **Detached Eddy Simulation** (DES) and **Scale Adaptive Simulation** (SAS) represent two of these hybrid strategies. The following paragraphs give an overview of both approaches and the corresponding turbulence models, which are used in this work.

The idea of **DES** is the simulation of massively separated flows based on a common RANS model, which switches from "RANS-mode" to "LES-mode" in regions away from the wall and vice versa. This switch of the simulation mode is achieved by a change of the turbulence length scale, which is used by the turbulence model. The initial formulation of DES, which was introduced by Spalart et al. in the year 1997, is based on the *Spalart-Allmaras* RANS turbulence model and changes the turbulence length scale depending on the local cell length or volume, and the distance to the wall. Although this model has been proved successful in simulating certain flows, it turned out that in case of an ambiguous grid, the criterion for the switch from RANS to LES might lead to a so-called *Grid Induced Separation*. Figure 2.3.6 shows a clear as well as an ambiguous mesh for the application of DES [17].
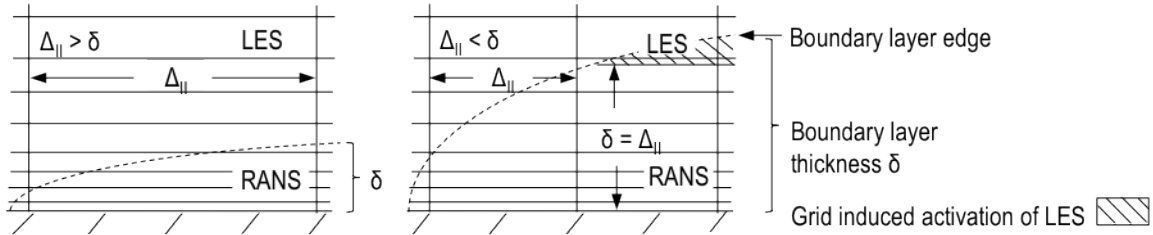
Figure 2.3.6: *Comparison of a clear and ambiguous DES mesh. Left: clear; right: ambiguous.*

The left picture of figure 2.3.6 shows a clear mesh for the use of the DES formulation. Here, the switch from RANS- to LES-mode would happen in the region outside of the boundary layer, where the wall distance reaches towards the cell distance in wall parallel direction. The right picture of figure 2.3.6 on the other hand, presents an ambiguous mesh for DES. As can be seen, the boundary layer thickness exceeds the cell distance in wall parallel direction at a certain point. This might be a result either of an inappropriate mesh or an unexpected growth of the boundary layer. As a consequence of that, the simulation would switch from RANS- to LES-mode inside the boundary layer. The LES-mode of the simulation however, might not be able to replace the RANS-predictions, which leads to a wrong prediction of the turbulence stresses. This yields the so-called *Modeled Stress Depletion.* In the light of that, the method of **Delayed Detached Eddy Simulation** (DDES) has been developed. This derivation of the original DES formulation presents a remedy for the Grid Induced Separation problem of the DES method. To do so, a second criterion, the kinematic turbulent eddy viscosity, is introduced to the definition of the turbulence length scale used by the turbulence model. Based on this modification of the turbulence length scale definition, the RANS-mode is self-perpetuating and avoids an activation of the LES-mode inside the boundary layer. The so-called **Improved Delayed Detached Eddy Simulation** (IDDES), in turn, represents a modification of the DDES method. The idea of the IDDES method is to provide one turbulence model, which is able to simulate not only massively separated but also wall bounded flows. The idea of the IDDES is based on a combination of the DDES and *Wall Modeled LES* (WMLES) approach. This combination is firstly based on a new definition of the sub-grid length-scale. Furthermore, the IDDES model is designed to switch between its DDES- and WMLES-branch. This switch is practically done by a new definition of the turbulence length-scale used by the turbulence model. This length-scale definition will either be equivalent to the length-scale definition of the original DDES formulation or to the length-scale definition of the WMLES-branch. The WMLES length-scale and consequently the WMLES-mode of the IDDES model will be activated, when turbulence is present and the provided mesh resolution is sufficient for LES resolved content [17].

The **SAS** approach principally represents a remedy for the limitations of RANS turbulence models in unsteady simulations and allows the model to adjust for unsteady flow features. This adjustment is done by the introduction of a second mechanical scale in the source/sink terms of the underlying RANS turbulence model. The starting point of the SAS method is the definition of an exact equation for the transport of kL formulated by *Rotta*, where L refers to an integral length-scale of turbulence. Due to the appearance of the third derivative of the velocity field however, the application of this formulation is unfavorable for general purpose CFD-calculations. The formulation by *Rotta* has therefore been reformulated by *Menter* and *Egorov*. This reformulation yielded the so-called **KSKL model**. Based on this model, the k-ω SST SAS model has been developed. The modification of the original k-ω SST model is achieved by an additional source term in the ω-equation. This additional term allows to adjust for SAS behavior in unsteady situations while sustaining the performance of the original model in the boundary layer [17].

2.4 OpenFOAM

OpenFOAM, which refers to *Open Field Operation and Manipulation* is a free, object-oriented open source CFD-code. It is based on a C++ library and provides the user with applications, i.e. precompiled utilities and solvers. These applications, in turn, have access to numerous dynamically linked libraries, which provide necessary implementations, e.g. turbulence models for the application. OpenFOAM is furthermore equipped with pre- as well as post-processing tools, which enable the user to perform complete simulations with a consistent software package. In comparison with other CFD-codes, OpenFOAM has interesting features from both a technical as well as economical point of view. From a technical point of view, OpenFOAM enables the user not only to select and modify provided applications and implementations but also to create ones own

implementations. This feature makes OpenFOAM an interesting code for the investigation of flow systems and numerical methods. From an economical point of view, OpenFOAM stands out because of its open source availability. This feature saves not only the acquisition cost, which incur with the use of commercial CFD-codes but also enables the user to a license-free performance of simulations. The latter advantage should not be underestimated when conducting investigations or simulations with a relatively high demand on computational performance [10].

The code uses the Finite Volume Method based on a co-located grid. In order to avoid unphysical behavior due to the storage of both pressure and velocity at the same point, the code applies an interpolation method in the spirit of *Rhie-Chow*. OpenFOAM does not provide a graphical user interface and follows a certain directory as well as file structure throughout the entire code. Figure 2.4.1 gives an overview of the basic structure of a general case definition of OpenFOAM [8][10].
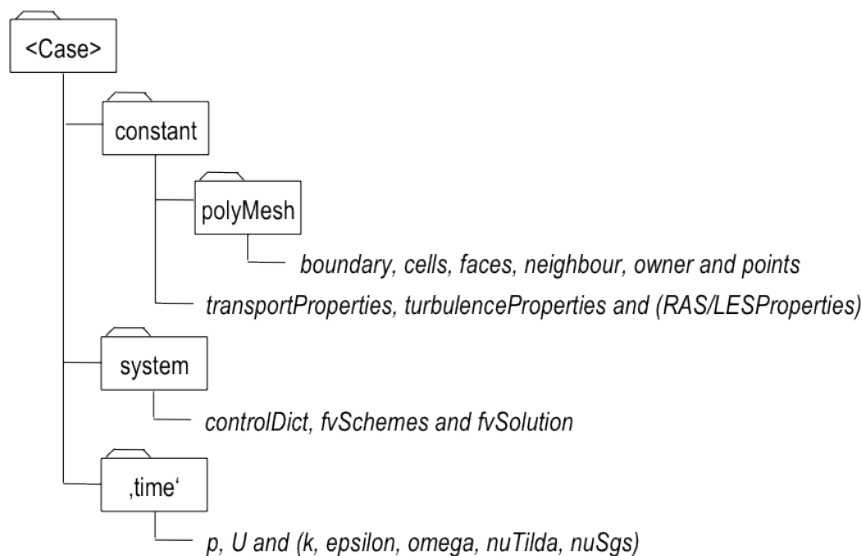

Figure 2.4.1: *Basic OpenFOAM directory and file structure of the cases performed in this work. Files in parenthesis are chosen according to the corresponding turbulence model.*

As can bee seen in figure 2.4.1, a case in OpenFOAM basically consists of three directories: *constant*, *system* and *"time"*. These folders, in turn, contain so-called *dictionaries*, which allow the user to specify the required data for the simulation. The dictionaries generally consist of a header and a following list of *keywords*, which enable the user to specify information by setting a value most likely in form of a number or a word. The *constant*-folder mainly includes dictionaries, which specify the physical properties as well as information regarding the turbulence model - provided that a turbulence model is used. Furthermore, the description of the case mesh can be found in the *polyMesh* sub-directory. The *system*-folder on the other hand, contains dictionaries, which refer to the solution process itself. These dictionaries primarily define the numerical schemes, solution methods and run-control parameters of the simulation. Finally, the *time*-folder defines the boundary and initial values of the properties, which are solved during the simulations. While the name of the folder refers to the timestep of the flow, the dictionary-files are named according to the corresponding property, e.g. *p* or *U* [10].

3 Cases and Results

This section presents the cases as well as the results of each case. To do so, the section is split into four subsections. While the first two subsections give an overview of the computational domain and the numerical setup of the simulations respectively, the last two subsections show the results of the cases. Here, subsection 3.3 presents the results of the simulations with a discharge of 25 $\frac{m^3}{h}$ and subsection 3.4 shows the results of the simulations with a discharge of 40 $\frac{m^3}{h}$.

In general, the simulations are performed with OpenFOAM 1.6-ext, compiled with the *gcc* compiler release *4.1.2*. Furthermore, the calculations are decomposed for parallel simulations based on the *scotch*-method provided by OpenFOAM. Here, RANS-simulations are executed on 16 and hybrid simulations on 32 processors. All cases are done with a molecular kinematic viscosity of $1*10^{-6}\frac{m^2}{s}$ and a blade angle of 30° from the vertical direction. In order to save computational time, the transient phase of the simulation is carried out with a maximum Courant number greater than 1. This phase corresponds to the part of the simulation, where the flow, introduced at the inlet of the computational domain, propagates through the entire domain. Since an evaluation of the flow in this phase would most likely result in the wrong results, the requirement of a maximum Courant number below 1 can be relaxed in order to reduce the time needed to complete the transient phase. The relaxation however, is limited in terms of avoiding a diverging simulation. The maximum Courant number should ideally be reduced to a value below 1 after the transient phase, i.e. when the results of the simulations are evaluated. In order to determine the end of the transient phase, probes of the pressure at different positions in the Swirl Generator have been observed with the *probe*-function object provided by OpenFOAM. Furthermore, the average values for the velocity as well as the pressure are determined with the help of the *fieldAverage*-function object. These mean values are used for the comparison of the axial as well as tangential velocity and the pressure difference with the corresponding experimental data. Based on the probes of the pressure at the pressure measurement positions, the average has been evaluated over several periods of the fluctuating pressure.

3.1 Computational Domain

This subsection firstly gives an explanation of the test rig and the underlying geometry, before the resulting meshes are elucidated.

3.1.1 Test Rig and Geometry

The test rig, which has been constructed by the Institute of Fluid Mechanics and Hydraulic Machinery at the University of Stuttgart, is designed as an open hydraulic circle. The head-water, measured by a flow meter and regulated by a valve, is directed to the Swirl Generator by a pipe. After the swirl is imposed to the flow with the help of 8 non-rotating blades, which are distributed symmetrically around a hub, the water flows through a conical diffuser and enters a container, before it reaches the tail-water. A pump closes the hydraulic circle and provides the head-water tank with the required water. Figure 3.1.1 gives a schematic overview of the whole test rig [5].
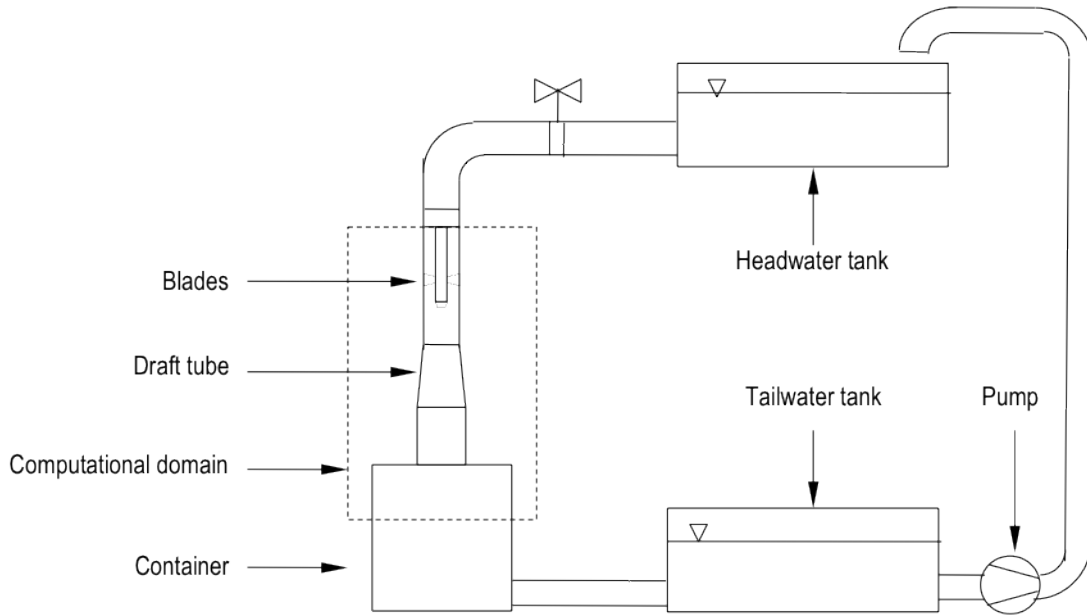
Figure 3.1.1: *Schematic figure of the test rig.*

As can be seen in figure 3.1.1, the computational domain for the simulation takes the swirl generating section, i.e. the blades, the following conical diffuser and the sudden expansion caused by the container into account. For this domain, the basic geometry as well as the mesh have been provided by the Institute of Fluid Mechanics and Hydraulic Machinery at the University of Stuttgart, and are modified for the purpose of this work. While the geometry has been created with *Pro/ENGINEER*[1], the mesh has been generated with *ANSYS ICEM CFD*[2]. Figure 3.1.2 shows further details of the computational domain.
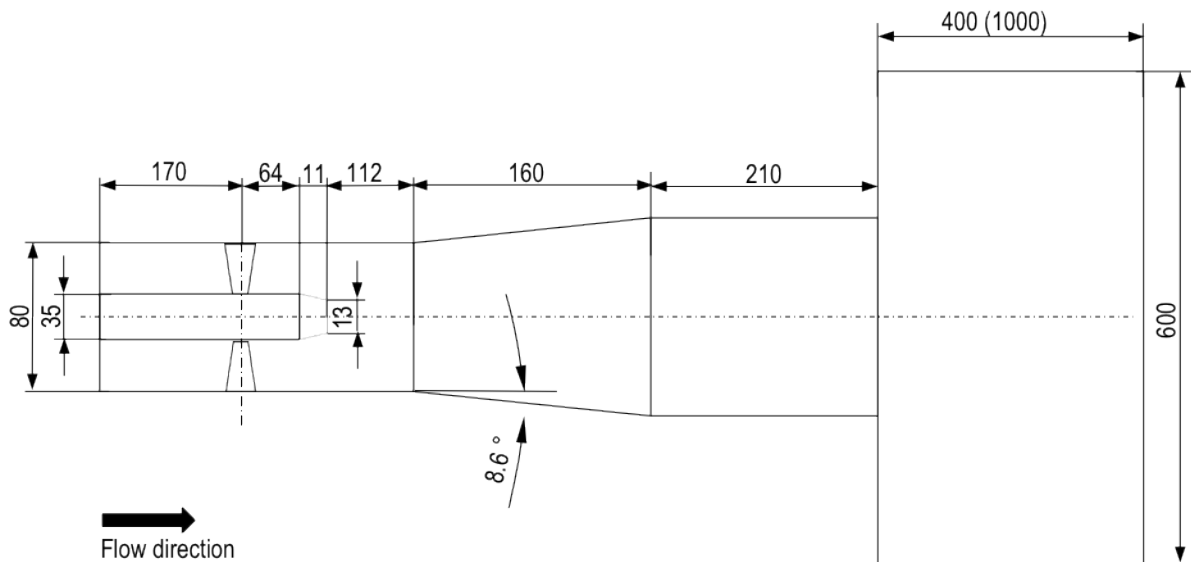

Figure 3.1.2: *Schematic figure of the geometry for the computational domain with dimensions in [mm]. Values in parenthesis refer to the original value of the test rig.*

Figure 3.1.2 shows a schematic sketch of the underlying geometry for the computational domain, rotated 90° counter clockwise. As can be seen, the hub includes a diameter of 35 *mm* and a total length of 245 *mm*.

---

[1]Throughout the text Pro/ENGINEER refers to Pro/ENGINEER®. Pro/ENGINEER is a trademark or registered trademark of Parametric Technology Corporation or its subsidiaries in the U.S. and in other countries.

[2]Throughout the text ANSYS ICEM CFD refers to ANSYS® ICEM™ CFD. ANSYS ICEM CFD is a trademark or registered trademark of ANSYS, Inc. or its subsidiaries in the United States or other countries.

The blades axes are placed 170 mm downstream of the inlet. Moreover, the pipe of the Swirl Generator has a diameter of 80 mm before the expansion of the conical diffuser. The diffuser expands with an angle of 8.6° over a length of 160 mm and is followed by a pipe with a length of 210 mm. This pipe finally ends in the container, which has a diameter of 600 mm and an original length of 1000 mm. In order to reduce the number of cells of the mesh, the length of the container has been set to 400 mm. Further simplifications have been done at the blades. Figure 3.1.3 presents a sketch of the original as well as simplified blade geometry.
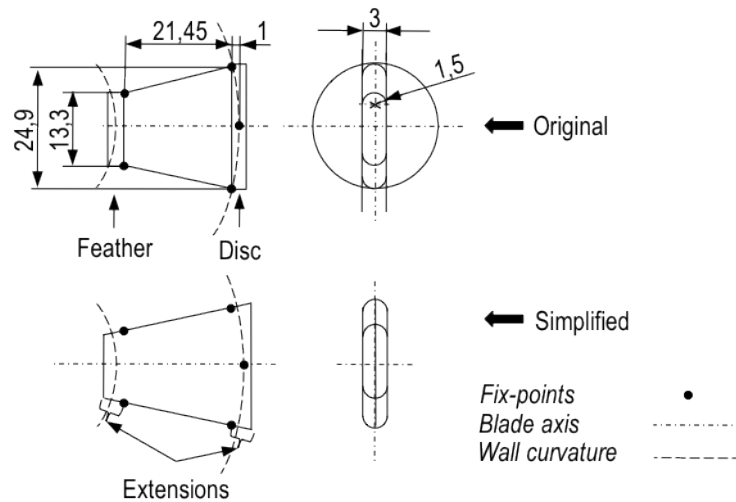

Figure 3.1.3: *Schematic figure of the original and simplified blade geometry. Left: side-view; right: front-view.*

As presented in figure 3.1.3, the blades consist of straight plates with a thickness of 3 mm. The edges of the blades include a rounding with a radius of 1.5 mm. It can be seen, that the original blades contain a feather, followed by a trapezoidal part and a disc. The feather as well as the disc are part of the mechanism to change the angle of the blades. These features however, had to be simplified due to the mesh generation. To do so, the trapezoidal part has been extended while retaining the original angle of the trapezoid. This angle is formed by a width of 21.45 mm and a height of 13.3 mm as well as 24.9 mm. Based on this geometry, the boundaries of the computational domain have been created. The boundaries of the entire computational domain can be seen in figure 3.1.4.
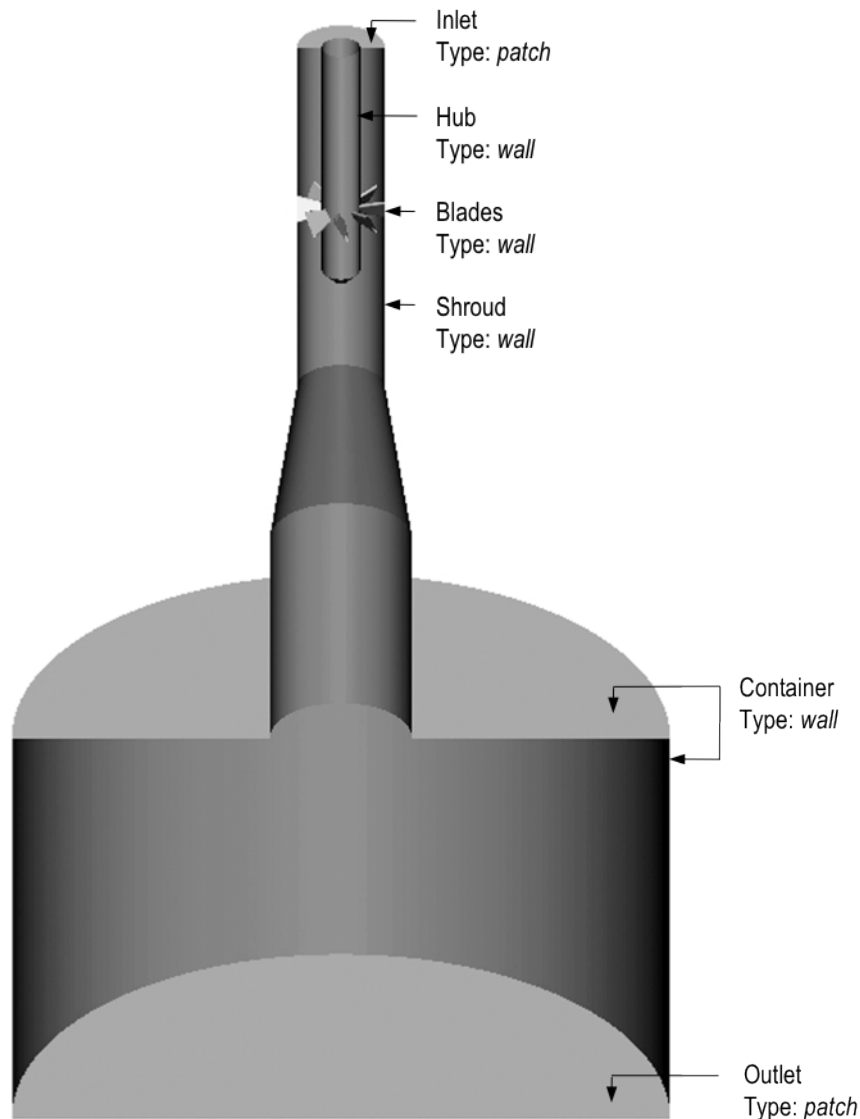
Figure 3.1.4: *Allocation of the boundaries of the geometry, presented with a clipped figure of the geometry surfaces. The term "Type" corresponds to the keyword in the boundary-dictionary of the case.*

As to be seen in figure 3.1.4, the computational domain basically consists of the hub, blades, container and a shroud, which includes the pipes and the conical diffuser. Moreover, it contains an inlet as well as an outlet. In order to specify the boundary conditions in OpenFOAM, different types are assigned to each boundary. While the parts hub, blades and shroud are of type *wall*, the inlet as well as the outlet are of type *patch*.

3.1.2 Mesh

For the discretisation of the computational domain, a structured mesh with hexahedral cells is used. This mesh is generated with the help of a mesh-blocking[3]. The division into blocks allows the application of different resolutions and cell distributions in different regions of the mesh. To simplify the generation process and to obtain a consistent mesh, the blocking used for the generation of the mesh for the Swirl Generator, only takes a section of 45° into account. This corresponds to one blade. Once the mesh for this slice of the geometry is created, it is copied 7 times to take the full geometry, i.e. 360° into account. Periodic faces and merging of the corresponding vertices assure that the resulting mesh consists of 8 seamless combined meshes which finally represent one computational domain.

The mesh topology basically consists of a so-called *O-grid* block, which enables to approximate the curved walls

---

[3]Figures of the blocking, which has been used in this work, are shown in Appendix C.

of the hub and the shroud without causing bad mesh quality properties, e.g. small cell-angles. In regions close to walls, the block is split parallel to the wall in order to be able to adjust the distance and distribution of the cells to the wall. Based on these features of the mesh topology, the design of the mesh has been influenced not only by the goal of a reasonable mesh size and quality but also the supply of an appropriate mesh resolution for the application of the turbulence models used in this work. These limitations lead to two different kinds of meshes, one for the application of high-Reynolds number RANS models and one for hybrid RANS-LES turbulence models. These grids mainly vary in their resolution of the regions close to walls and consequently in the number of cells they contain. While the mesh for the RANS models does not resolve the near-wall region, the mesh for the hybrid models integrates to the wall. These differences in the resolution of the near-wall region lead to a mesh for the RANS models with approximately 2.85 million and for the hybrid models with approximately 6.5 million cells. Furthermore, the different discharges have been taken into account with the variation of the distance of the first grid point as measured from the wall. This finally yields four different meshes, a mesh for RANS as well as hybrid models for each value of discharge.

In order to be able to assess the final results of the simulations, it is beneficial to know the features of these meshes. In terms of quality features, the mesh can be described by several criteria. The cell *angle*, *aspect ratio*, *determinant* and *volume change* represent four of these criteria, which have been verified for the meshes used in this work. The angle criterion refers to the internal angles of a cell, which should ideally be 90°. The aspect ratio on the other hand, is a measurement for the aspect ratio of a cell and is calculated differently for different element types. In case of hexahedral cells, it is calculated by the ratio of the minimum to the maximum cell edge. Based on this definition, an aspect ratio value of 1 corresponds to an ideal, cubic cell. The determinant criterion checks the deformation of a cell. While a value of 1 refers to a perfect hexahedral cube, values equal or below 0 refer to inverted cells with negative volumes. Finally, the volume change refers to the cell growth from one cell to the surrounding cell with the largest volume. A value of 1 or 1.2 corresponds to a cell growth of 0% and 20% respectively [1][4]. The mesh generation for complex geometries generally requires a deviation from the ideal values of these quality features. The influence of this deviation however, depends on the underlying CFD-code. In this work, the meshes are designed to mainly contain cells with an angle in a range from 45° to 90°, an aspect ratio in the range of 1-10, a cell determinant greater than 0.45 and a volume change in the range of 1 to 1.3. For the value of the aspect ratio, the reciprocal of the original value is used. Tables 3.1.1 and 3.1.2 give an overview of the approximate quality features of both kinds of meshes. Furthermore, figures of the meshes are shown in Appendix C.

Table 3.1.1: Approximate quality features of the mesh for RANS models according to ANSYS ICEM. The sum of the percentage of the mesh for the volume change criterion is not 100% since its diagnostic is undefined in some elements of the mesh.

Criterion	Range of values	Percentage of the mesh in [%]
Angle in [°]	25-45	0.2
	45-90	99.8
Aspect ratio	1-10	77.0
	10-20	18.0
	20-45	5.0
Determinant	≥ 0.45	100.0
Volume change	1.0-1.3	82.0
	1.3-2.0	8.0
	2.0-7.0	4.0

As shown in table 3.1.1, the meshes for the RANS models consists to 99.8% of cells with an angle between 45° and 90° and 0.2% of cells with an angle in the range of 25° and 45°. Furthermore, an aspect ratio between 1 and 10 is achieved in 77% of the mesh. 18% of the mesh consists of cells with an aspect ratio between 10 and 20, and 5% with an aspect ratio between 20 and 45. A determinant greater than 0.45 assures positive cell volumes in each cell of the mesh. The cell growth in the mesh is between 1.0 and 1.3 in 82% of the mesh. 8% correspond to a cell growth of 1.3 to 2.0 and 4% to a cell growth in a range from 2.0 to 7.0. Table 3.1.2 presents the quality features of the mesh for hybrid models.

Table 3.1.2: Approximate quality features of the mesh for hybrid models according to ANSYS ICEM. The sum of the percentage of the mesh for the volume change criterion is not 100% since its diagnostic is undefined in some elements of the mesh. Values in parenthesis refer to the mesh designed for a discharge of 40 $\frac{m^3}{h}$.

Criterion	Range of values	Percentage of the mesh in [%]
Angle in [°]	25-45	0.5
	45-90	99.5
Aspect ratio	1-10	50.0
	10-20	20.0
	20-900 (1400)	30.0
Determinant	≥ 0.65	100.0
Volume change	1.0-1.3	77.0
	1.3-2.0	17.0
	2.0-10 (16)	2.0

Table 3.1.2 shows, that all quality features except for the aspect ratio of the mesh for hybrid models, are quite similar to the features of the mesh for RANS models. With 99.5%, the majority of cells include an angle between 45° and 90°. The remainder of the mesh consists of cells between 25° and 45°. The aspect ratio is in a range from 1 to 10 in 50% of the mesh. 20% of the mesh contains an aspect ratio between 10 and 20 and 30% correspond to a range from 20 to 900 for a discharge of 25 $\frac{m^3}{h}$ and from 20 to 1400 for a discharge of 40 $\frac{m^3}{h}$. These aspect ratios result mainly from the refinement of the mesh to the walls of the geometry and can be found not only close to walls but also in the center line downstream of the hub. The determinant of the cells is greater than 0.65 in each cell of the mesh. Finally, the volume change is in a range of 1.0 and 1.3 in 77%, between 1.3 and 2.0 in 17% and from 2.0 to 10.0 and 16.0 respectively in 2% of the mesh. Based on these approximate quality features, the *checkMesh*-function of OpenFOAM gives no error message for all quality features except the aspect ratio greater than 1000 of the mesh for hybrid models and a discharge of 40 $\frac{m^3}{h}$. This relatively high aspect ratio falls back on the reduction of the cell length of the first cells measured from the wall in wall direction.

As elucidated in subsection 2.3, the placement of the first grid point as measured from the wall plays an important role for the corresponding turbulence model. Table 3.1.3 therefore, gives an overview of the distances of the cells adjacent to a wall and the resulting y^+-values.

Table 3.1.3: Approximate cell length of the first cell as measured from the wall in wall direction and corresponding y^+-values. y^+-values of the RANS-mesh are represented by the values of the simulations based on the *k-ω SST* model and y^+-values of the hybrid-mesh are represented by the simulations based on the *Spalart-Allmaras DDES* model.

Discharge Q in [$\frac{m^3}{h}$]	Mesh	Shroud y in [mm]	y^+	Hub y in [mm]	y^+	Blade y in [mm]	y^+	Container y in [mm]	y^+
25	RANS	1.0	22-68	0.9	7-107	1.1	6-86	5.0/9.0[4]	1-101
	Hybrid	0.01	≤ 2	0.01	≤ 2	0.01	≤ 2	0.60	≤ 9
40	RANS	1.0	30-113	0.9	13-142	1.1	15-136	5.0/9.0[4]	4-142
	Hybrid	0.006	≤ 2	0.006	≤ 2	0.006	≤ 2	0.600	≤ 12

As can be seen in table 3.1.3, the RANS mesh contains a cell length in normal direction to the wall of approximately 1 mm at the wall of the shroud, hub and blades. The distance at the wall of the container is 5 mm at the upper-wall and 9 mm at the side-wall. The mesh for the hybrid models on the other hand, includes a cell length in normal direction to the wall of approximately 0.01 mm and 0.006 mm at the wall of the shroud, hub and blades. Here, the first value refers to the mesh for a discharge of 25 $\frac{m^3}{h}$, the second value corresponds to the mesh for a discharge of 40 $\frac{m^3}{h}$. The distance at the wall of the container is approximately 0.6 mm for

---

[4]The value 5 refers to the upper wall and the value 9 to the side-wall of the container.

both values of discharge. Furthermore, it is clear from table 3.1.3, that the y^+-values of the RANS simulations reach from 6 to 142 at the boundaries of interest, i.e. shroud, hub and blades. The y^+-values of the hybrid simulations on the other hand, are less or equal than 2 at the same boundaries. In order to assess the mesh resolution in wall parallel and span-wise direction, the y^+-value is moreover used to calculate the approximate x^+ and z^+ values. Based on the mesh for hybrid models with a discharge of 25 $\frac{m^3}{h}$ and the *Spalart-Allmaras DDES* model, the following values are obtained for an exemplary point close to the wall inside of the draft tube[5]:

$$x^+ \approx 390, \tag{3.1.1}$$

$$z^+ \approx 680. \tag{3.1.2}$$

3.2 Numerical Setup

This subsection describes the numerical setup of the simulations performed in this work. For this, the subsection firstly presents the general numerical setup and secondly shows the boundary and initial conditions, which are used for the different cases.

3.2.1 General Setup

In general, the numerical settings are firstly chosen to meet the demands of the goals of this work, i.e. to provide a baseline for accurate simulations and the investigation of the different turbulence models. Within that framework, the setup should provide not only numerically stable calculations but also the supply of the results in a reasonable time. Table 3.2.1 gives an overview of the discretisation schemes, which are used in this work.

Table 3.2.1: Setup of the discretisation schemes in the *fvSchemes*-dictionary.

Sub-dictionary	Keyword	Entry
ddtSchemes	default	backward
gradSchemes	default	Gauss linear
divSchemes	default	Gauss linear
	div(phi,ϕ)[6]	Gauss limitedLinear 1[7]
laplacianSchemes	default	Gauss linear corrected
interPolationSchemes	default	linear
snGradSchemes	default	corrected

As can be seen in table 3.2.1, the **backward** differencing scheme is applied for the discretisation of the temporal terms. This discretisation scheme takes the previous two values into account and corresponds to a second-order accurate, implicit discretisation scheme [7][10]. Furthermore, it can be seen that the simulations are based on the standard *Gaussian finite volume integration* method, which is specified by the entry *Gauss* in the gradient-, divergence- and laplacian-schemes. Since the *Gaussian* discretisation method requires not only the cell center values but also values on the cell faces, an interpolation scheme has to be specified. Here, the entry *linear* corresponds to the *central differencing* interpolation. Based on the *Gaussian* integration method and central differencing as the interpolation method, the default entries for the gradient and divergence terms finally read **Gauss linear**. These entries represent a second-order gradient-term discretisation and a second-order, unbounded divergence-term discretisation. As explained in subsection 2.2, the discretisation of the convection terms in flows with a high convection-diffusion ratio require a scheme, which takes the direction of the convection into account. Therefore, a Total Variation Diminishing (TVD) interpolation method is used for the convection terms of the transport equations. The entries of these terms are specified additionally to the default entry for the divergence schemes and will automatically substitute the default entry for the corresponding term. The additional entries read **Gauss limitedLinear 1** and **Gauss limitedLinearV 1**. Here, *limitedLinearV* corresponds to the interpolation method for a vector and is used for the convection of the

---

[5]The coordinates of the corresponding point are (0 54.5245 -450).

[6]ϕ represents the corresponding properties, transported by convection. Here: U, k, ϵ, ω and $\tilde{\nu}$.

[7]The scheme for the convection of the velocity vector *div(phi,U)* reads *Gauss limitedLinearV 1*.

velocity vector. The addition 1 refers to the activation of the full TVD method [10]. The laplacian-term of the momentum equations moreover requires the specification of a further discretisation scheme in order to calculate the surface normal gradient. The default entry for the laplacian-term therefore, reads **Gauss linear corrected**, whereas *corrected* refers to the discretisation scheme for the surface normal gradient, which gives conservative, unbounded behavior with second-order accuracy. Finally, the general interpolation scheme is specified by the default entry **linear**, which refers to central differencing, and the general surface normal gradient scheme by the default entry **corrected**, which corresponds to an explicit, non-orthogonal correction [10].

As elucidated in subsection 2.2, the Navier-Stokes equations require an algorithm in order to solve the systems of linear algebraic equations, which are obtained from the discretisation of the partial differential equations. In this work the so-called PIMPLE algorithm, provided by OpenFOAM, is used to do so. This algorithm principally refers to a combination of the SIMPLE and PISO algorithm. Table 3.2.2 gives an overview of the settings of the PIMPLE algorithm [10].

Table 3.2.2: Setup of the PIMPLE algorithm for pressure-velocity coupling in the *fvSolution*-dictionary.

Sub-dictionary	Keyword	Entry
PIMPLE	nOuterCorrectors	5
	nCorrectors	2
	nNonOrthogonalCorrectors	1
relaxationFactors	p	0.3
	U, k, ϵ, ω and $\tilde{\nu}$	0.7^8

Table 3.2.2 shows, that the iterative PIMPLE algorithm consists of 5 outer-corrector loops, 2 corrector loops and 1 non-orthogonal correction loop in each timestep. Furthermore, it can be seen that an under-relaxation of the transported properties is applied. While the under-relaxation factor for the pressure is 0.3, the velocity as well as turbulence properties are under-relaxed with a factor of 0.7. Due to high values of k in the transient phase however, the factor for the property k of the k-ω *SST SAS* model has been set to 0.5. Table 3.2.3 gives an overview of the solvers for the linear algebraic equations, which are applied in the simulations.

Table 3.2.3: Setup of the solvers for the systems of linear algebraic equations in the *fvSolution*-dictionary.

Sub-dictionary	Sub-sub-dictionary	Keyword	Entry
solvers	p	solver	PCG
		preconditioner	DIC
		tolerance	$1e^{-6}$
		relTol	0.01
	pFinal	solver	PCG
		preconditioner	DIC
		tolerance	$1e^{-6}$
		relTol	0
	U	solver	PBiCG
		preconditioner	DILU
		tolerance	$1e^{-5}$
		relTol	0.1
	UFinal, k, ϵ, ω, $\tilde{\nu}$	solver	PBiCG
		preconditioner	DILU
		tolerance	$1e^{-5}$
		relTol	0

As is clear from table 3.2.3, the simulations are based on two different iterative solvers. These solvers distinguish between symmetric and asymmetric matrices, whereas the resulting matrix depends on the structure of the underlying equation. While the so-called Preconditioned Conjugated Gradient (PCG) solver is used for the

---

[8]The relaxation factor for k in the k-ω *SST SAS* model has been set to 0.5.

symmetric matrix of the pressure, the Preconditioned Bi-Conjugated Gradient (PBiCG) solver is applied for the asymmetric matrices of the velocity as well as turbulence properties. Both solvers use a so-called preconditioner in order to improve the solution process. Here, the preconditioner for the PCG solver reads Diagonal Incomplete-Cholesky (DIC) and for the PBiCG, Diagonal Incomplete-LU (DILU) [10]. The tolerance for the pressure calculation is set to $1 * 10^{-6}$ and the relative tolerance to 0.01. For the final iteration, the relative tolerance is changed to 0. The tolerance level for the velocity and the turbulence properties corresponds to $1 * 10^{-5}$. For the velocity, the relative tolerance is 0.1 with a value of 0 for the final iteration step. In contrast to that, the relative tolerance for the turbulence properties is 0 at all times.

3.2.2 Boundary and Initial Conditions

As can be seen in tables 3.2.4 and 3.2.5, boundary and initial conditions are specified for the pressure, velocity, and for each property, which introduces an additional transport equation in order to model the influence of turbulence. As a result of the object-oriented implementation of OpenFOAM however, the specification of boundary and initial conditions is also necessary for the Sub-Grid-Scale kinematic viscosity, which is actually not derived by a transport equation but calculated by the corresponding turbulence properties of the turbulence model. Table 3.2.4 presents the boundary and initial conditions of the properties pressure and velocity.

Table 3.2.4: Boundary and initial conditions of pressure and velocity. The patches correspond to the boundaries and types shown in figure 3.1.4. For a better overview, all patches of type *wall* are represented by a common patch "*Wall*".

Property	Patch	Type	Value		Formula
			Q=25 $\frac{m^3}{h}$	Q=40 $\frac{m^3}{h}$	
p in $[\frac{m^2}{s^2}]$	Inlet	zeroGradient	-		-
	Outlet	fixedMeanValue[9]	0		
	Internal field	uniform	0		
	Wall	zeroGradient	-		
U in $[\frac{m}{s}]$	Inlet	fixedValue	(0 0 1.708588)	(0 0 2.733740)	$v = \frac{Q}{A_{inlet}}$
	Outlet	zeroGradient	-		
	Internal field	uniform	(0 0 0)		
	Wall	fixedValue	(0 0 0)		

As to be seen in table 3.2.4, the pressure is for both values of discharge set to a zero gradient boundary condition at the inlet and a fixed mean value equal to 0 at the outlet. The selection of these boundary conditions is due to the fact, that no information about the prevailing pressure level is available for the corresponding positions in the hydraulic system. The choice of these values is however not important for the results of the simulations, since only the pressure gradient goes into the underlying system of equations, and not the pressure itself. The pressure level can subsequently be adjusted to the level of the experimental results, if necessary. While the pressure for the internal field is set to 0, the pressure at the walls corresponds to a zero gradient boundary condition. The boundary and initial conditions for the velocity are specified for each direction by a velocity vector. For the inlet, the axial velocity component is calculated by the ratio of the discharge to the area at the inlet. This yields a fixed value boundary condition with an axial velocity of 1.708588 $\frac{m}{s}$ for a discharge of 25 $\frac{m^3}{h}$ and 2.733740 $\frac{m}{s}$ for a discharge of 40 $\frac{m^3}{h}$. The velocity at the outlet is set to a zero gradient boundary condition and the internal field as well as the velocity at the walls is equal to 0 in each direction. While the boundary and initial conditions for the pressure and velocity are the same for all simulations, the conditions for the turbulence properties depend on the turbulence model, which is applied. Table 3.2.5 presents all properties, which are needed in order to use the turbulence models used in this work.

---

[9]Changed to type *fixedValue* for the *k-ω SST* model.

Table 3.2.5: Boundary and initial conditions of the turbulence properties. The patches correspond to the boundaries and types shown in figure 3.1.4. For a better overview, all patches of type *wall* are represented by a common patch "*Wall*". The calculated values correspond to a turbulence intensity of approximately 4%.

Property	Patch	Type	Value		Formula
			Q=25 $\frac{m^3}{h}$	Q=40 $\frac{m^3}{h}$	
k in $[\frac{m^2}{s^2}]$	Inlet	fixedValue	0.008006	0.018223	$k = \frac{3}{2}(U * I)^2$
	Outlet	zeroGradient	-		
	Internal field	uniform	0.008006	0.018223	$I = 0.16 R_e^{-\frac{1}{8}}$
	Wall (RANS)	kqRWallFunction	0.008006	0.018223	$U = \frac{Q}{A_{inlet}}$
	Wall (hybrid)	fixedValue	0		$R_e = \frac{U * l_t}{\nu}$
ϵ in $[\frac{m^2}{s^3}]$	Inlet	fixedValue	0.318360	1.093286	$\epsilon = \frac{k^{\frac{3}{2}}}{l_t}$
	Outlet	zeroGradient	-		
	Internal field	uniform	0.318360	1.093286	$l_t = 0.1 * R$
	Wall	epsilonWallFunction	0.318360	1.093286	$R = R_a - R_i$
ω in $[s^{-1}]$	Inlet	fixedValue	441.850533	666.623114	$\omega = \frac{k^{\frac{1}{2}}}{C_\mu * l_t}$
	Outlet	zeroGradient	-		
	Internal field	uniform	441.850533	666.623114	$C_\mu = 0.09$
	Wall (RANS)	omegaWallFunction	441.850533	666.623114	
	Wall (hybrid)	fixedValue	0^{10}		
$\tilde{\nu}$ in $[\frac{m^2}{s}]$	Inlet	fixedValue	0.000018	0.000027	
	Outlet	zeroGradient	-		$\frac{\tilde{\nu}}{\nu} = \frac{\nu_t}{\nu}$
	Internal field	uniform	0.000018	0.000027	
	Wall	fixedValue	0		
ν_{SGS}[11] in $[\frac{m^2}{s}]$	Inlet	zeroGradient	-		
	Outlet	zeroGradient	-		-
	Internal field	uniform	0		
	Wall	fixedValue	0		
$\frac{\nu_t}{\nu}$	-	-	18	27	-

Table 3.2.5 shows the boundary and initial conditions for the relevant properties of the turbulence models and for the property ν_{SGS}. Furthermore, the resulting ratio of the turbulent kinematic viscosity ν_t to the molecular kinematic viscosity ν is shown in order to prove the calculated values to be reasonable. In general, the values of the properties are firstly calculated for the inlet. Here, the value of the turbulence kinetic energy k is 1.5 times the squared product of the velocity U and the turbulence intensity I [14]. While the turbulence dissipation rate ϵ is the ratio of k to the power of 1.5 to the turbulence length scale l_t [3], the turbulence frequency ω is the ratio of the square root of k to the product of the constant $C\mu$ and the turbulence length scale l_t [3]. The modified turbulent kinematic viscosity $\tilde{\nu}$ is derived by the ratio of the turbulent kinematic viscosity to the molecular kinematic viscosity, whereas the turbulent kinematic viscosity is based on the calculated values of k and ϵ or ω. Although, the specification of ν_{SGS} is not necessary from a numerical point of view, the implementation in OpenFOAM requires the corresponding dictionary in the timestep folder of the case. The boundary and initial conditions for ν_{SGS} are arbitrary set to a zero gradient boundary condition at the inlet as well as outlet and equal to 0 in the internal field as well as the walls. Based on a turbulence intensity of approximately 4 % for both values of discharge and a turbulence length scale of 10 % of the distance between the inner and outer radius of the inlet, the following values are obtained: the turbulence kinetic energy k is set to 0.008006 $\frac{m^2}{s^2}$ for a discharge of 25 $\frac{m^3}{h}$ and 0.018223 $\frac{m^2}{s^2}$ for a discharge of 40 $\frac{m^3}{h}$. The turbulence dissipation rate ϵ is 0.318360 $\frac{m^2}{s^3}$ for a discharge of 25 $\frac{m^3}{h}$ and 1.093286 $\frac{m^2}{s^3}$ for a discharge of 40 $\frac{m^3}{h}$. For the turbulence frequency ω, 441.850533 s^{-1} is calculated for a discharge of 25 $\frac{m^3}{h}$ and 666.623114 s^{-1} for a discharge of 40 $\frac{m^3}{h}$. The values of the turbulence properties k, ϵ and ω yield a ratio of the turbulent kinematic viscosity ν_t to the molecular viscosity ν of 18 for a discharge of 25 $\frac{m^3}{h}$ and 27 for a discharge of 40 $\frac{m^3}{h}$. These ratios are

---

[10]Changed to $1 * 10^{-12}$ for the k-ω SST SAS model to avoid 0 in the numerator.

[11]The ν_{SGS} file has to be present in the timestep folder, although ν_{SGS} does not represent a transported property.

considered to represent reasonable values based on experiences from other simulations. Furthermore, the ratios are used to derive the value of the modified turbulent kinematic viscosity $\tilde{\nu}$. Assuming that the modified turbulent kinematic viscosity $\tilde{\nu}$ is equal to the turbulent kinematic eddy viscosity ν_t, the value of the modified turbulent kinematic viscosity is set to 0.000018 $\frac{m^2}{s}$ for a discharge of 25 $\frac{m^3}{h}$ and 0.000027 $\frac{m^2}{s}$ for a discharge of 40 $\frac{m^3}{h}$. While the inlet values are moreover used as an initial guess for the internal field, the outlet is set to a zero gradient boundary condition for all properties. At the walls, the properties are set to the inlet value for turbulence models based on wall functions[12] and equal to 0 for turbulence models, which integrate to the wall. For the turbulence frequency ω in the k-ω SST SAS model however, the value at the walls is set to $1 * 10^{-12}$ in order to avoid 0 in the numerator of the ω-transport equation.

3.3 Results for a Discharge of 25 $\frac{m^3}{h}$

This subsection presents the results of the simulations with a discharge of 25 $\frac{m^3}{h}$. To do so, an overview of the development and performance of the simulations is firstly given. Furthermore, the flow as well as the turbulence models are analyzed, before the evaluation finally shows a comparison of the results with the experimental data.

3.3.1 Development and Performance

As explained in section 3, simulations of unsteady flow include a transient phase in the beginning of the simulations. The evaluation of the results therefore, depends on the development of the simulation, i.e. whether the transient phase of the simulation is completed or not. In order to assess the development of the simulations, figures 3.3.1 and 3.3.2 present the development of the pressure signal at position $P1$.
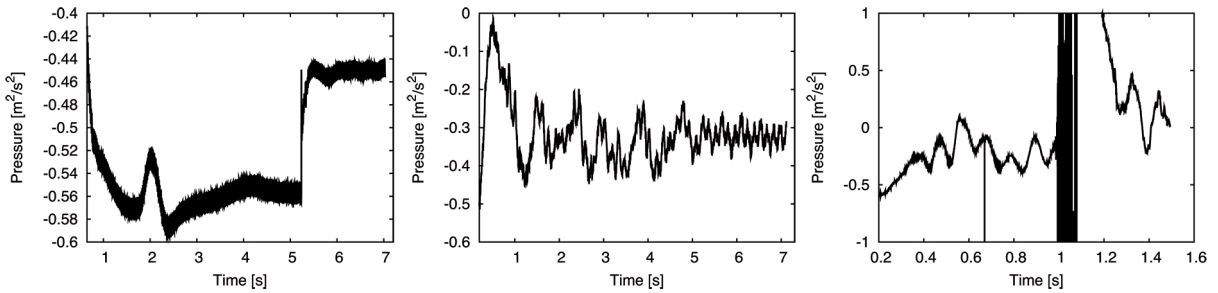

Figure 3.3.1: *Development of the pressure signal at position P1. From left to right: k-ε, k-ω SST and k-ω SST SAS.*

As can be seen in the left and middle picture of figure 3.3.1, the simulations based on the k-ϵ and k-ω SST turbulence model are with approximately 7 seconds of simulation relatively well developed. The sudden deflection of the pressure signal of the simulation based on the k-ϵ model refers to a change of the timestep of the simulation. Moreover it is clear from the right picture of figure 3.3.1, that the progress of the k-ω SST SAS simulation is with approximately 1.5 seconds relatively short and includes severe deflections at a time of approximately 1 second. The development of the simulations based on the *Spalart-Allmaras DDES* and *Spalart-Allmaras IDDES* model is depicted in figure 3.3.2.

---

[12]The "Type"-entries for the wall functions read: *kqRWallFunction*, *epsilonWallFunction* and *omegaWallFunction* respectively.
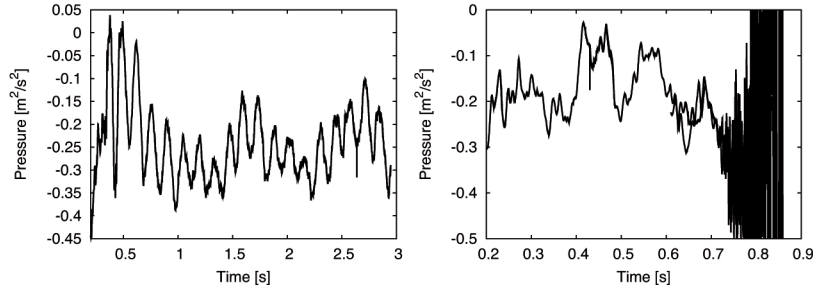
Figure 3.3.2: *Development of the pressure signal at position P1. Left: Spalart-Allmaras DDES; right: Spalart-Allmaras IDDES.*

The left picture of figure 3.3.2 shows with approximately 3 seconds of simulation a relatively well developed pressure signal of the simulation based on the *Spalart-Allmaras DDES* model. Furthermore, it can be seen in the right picture that the transient phase of the simulation based on the *Spalart-Allmaras IDDES* model is with approximately 0.85 seconds of simulation probably not completed and contains severe deflections starting at a time of approximately 0.72 seconds of the simulation. These deflections as well as the intermediate deflections of the *k-ω SST SAS* model might be a result of a too large timestep in the transient phase. It is clear from section 1, that the required computational effort represents a substantial criteria in the assessment of numerical simulations. The performance of the simulations based on their latest timesteps is therefore, presented in table 3.3.1.

Table 3.3.1: Performance of the simulations for a discharge of 25 $\frac{m^3}{h}$. The values are normalized for simulations decomposed on 32 processors and a maximum Courant number of 1. Influences of different Courant numbers or degrees of decomposition on the performance of the simulations are neglected. The calculations are based on the timestep, required time to calculate one timestep and maximum Courant number of the latest executed timestep of the simulation.

	k-ϵ	k-ω SST	k-ω SST SAS	S-A DDES	S-A IDDES
Timestep in [s]	$9*10^{-5}$	$1*10^{-4}$	$1*10^{-5}$	$2*10^{-5}$	$9*10^{-7}$
Time to calculate 1 timestep in [s]	74	73	227	231	226
Time divided by the average time of the RANS models	1	1	3	3	3
Time to simulate 1 s of the flow in [d]	9	9	198	142	2977
Time divided by the average time of the RANS models	1	1	22	16	330

Table 3.3.1 shows the timestep for a maximum Courant number of 1, the required time to calculate 1 timestep and to simulate 1 second of the flow. Furthermore, table 3.3.1 presents the corresponding times divided by the average time of the RANS models in order to express the relatively high computational effort of the hybrid models in contrast to the RANS models. As is clear from table 3.3.1, the simulation of 1 second of the flow with the RANS models takes 9 days on 32 processors. In contrast to that, the simulation of 1 second of the flow takes 198 days based on the *k-ω SST SAS*, 142 days based on the *Spalart-Allmaras DDES* and 2977 days based on the *Spalart-Allmaras IDDES* model. The differences of the performances between the RANS and hybrid simulations can firstly be addressed to the differences of the timestep. The required timestep of the simulations with a maximum Courant number of 1 is: $9 * 10^{-5}$ seconds based on the *k-ε*, $1 * 10^{-4}$ seconds based on *k-ω SST*, $1 * 10^{-5}$ seconds based on *k-ω SST SAS*, $2 * 10^{-5}$ seconds based on *Spalart-Allmaras DDES* and $9 * 10^{-7}$ seconds based on *Spalart-Allmaras IDDES* model. The relatively low timestep of the simulation based on the *Spalart-Allmaras IDDES* model is assumed to be a consequence of the deflections shown in figure 3.3.2. A comparison of the time required for the calculation of 1 timestep subsequently shows a difference between the performance of RANS and hybrid models. The time required by the simulations corresponds to: 74 seconds

25

based on the k-ϵ, 73 seconds based on the k-ω *SST*, 227 seconds based on the k-ω *SST SAS*, 231 seconds based on the *Spalart-Allmaras DDES* and 226 seconds based on the *Spalart-Allmaras IDDES* model. The significant difference between the simulations based on RANS and hybrid models can firstly be addressed to the higher number of cells of the meshes for the hybrid models. A comparison of the number of iterations for the calculation of the pressure furthermore reveals, that the hybrid models require considerably more iterations to calculate the pressure of each timestep. Table 3.3.2 presents the average number of iterations of non-orthogonal corrector steps of the simulations. Moreover, the difference between the corresponding hybrid model to the average number of iterations of the RANS models is shown.

Table 3.3.2: Comparison of the average number of iterations for the calculation of the pressure based on the latest timestep of the simulations and the non-orthogonal corrector step.

	k-ϵ	k-ω SST	k-ω SST SAS	S-A DDES	S-A IDDES
Average number of iterations per timestep	437	431	646	680	510
Difference to the average number of iterations of the RANS models	-	-	212	246	76

As table 3.3.2 shows, the hybrid models tend to conduct significantly more iterations to calculate the pressure field. While the average number of iterations of the simulations based on the k-ϵ and k-ω *SST* model is 437 and 431 respectively, the simulation based on the k-ω *SST SAS* model iterates 646, based on the *Spalart-Allmaras DDES* model 680 and based on the *Spalart-Allmaras IDDES* model 510 times.

3.3.2 Evaluation of the Results

As explained in subsection 2.1, the flow in the Stuttgart Swirl Generator is characterized by two significant flow features. It firstly contains quasi-cylindrical as well as helical isosurfaces of pressure. Secondly, it includes a region of reversed flow. An evaluation of the presence of these features allows a first assessment of the simulations, i.e. whether the turbulence model was capable of simulating the striking flow features or not. To do so, figure 3.3.3 presents an isosurface of pressure and axial velocity.
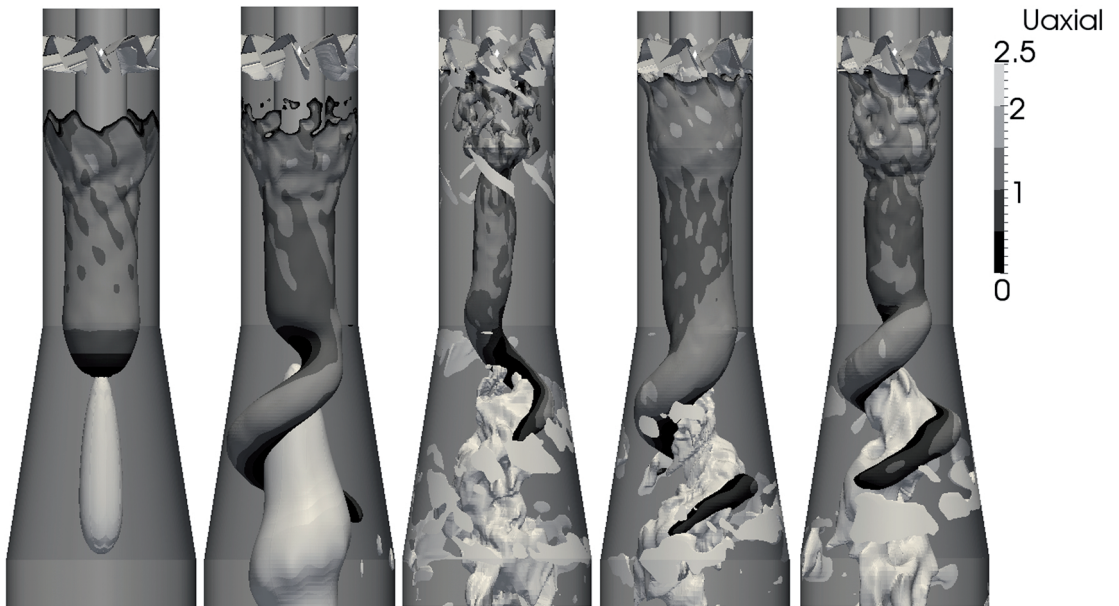

Figure 3.3.3: *Isosurface of pressure (colored by axial velocity in $[\frac{m}{s}]$) and axial velocity equal to -0.001 $\frac{m}{s}$ (solid color). The axial velocity is positive in flow direction, i.e. from top to bottom. From left to right: k-ϵ, k-ω SST, k-ω SST SAS, Spalart-Allmaras DDES and Spalart-Allmaras IDDES.*

All pictures in figure 3.3.3 show an isosurface of pressure, which represents the helical vortex, and axial velocity equal to -0.001 $\frac{m}{s}$, which refers to the region of reversed flow. As can be seen, all turbulence models except the k-ϵ model were capable of simulating the helical vortex. The failure of the k-ϵ model is assumed to be the result of a severe dampening of the unsteady flow in terms of too high an amount of modeled turbulence. The comparably small dimension of the helical vortex of the k-ω SST SAS model is assumed to be the result of a not fully developed simulation. As expected, the helical vortex rotates asymmetrically around a region of reversed flow in the center-line. It is clear from the pictures of the simulations based on the k-ω SST and $Spalart$-$Allmaras$ $DDES$ model, that the helical section of the vortex rope starts at the beginning of the conical diffuser. The isosurface of pressure, colored by axial velocity, shows the asymmetric distribution of pressure, caused by the helical vortex. The distribution of the axial velocity furthermore shows the connection between the vortex rope and the counterflowing region and indicates, that the helical vortex rotates around its own axis. Here, black refers to an axial velocity between 0 $\frac{m}{s}$ and 0.5 $\frac{m}{s}$. A comparison of the region of reversed flow reveals, that the hybrid models present a significantly more detailed counterflowing region compared to the relatively smooth pictures of the RANS models. In order to give a more distinct picture of the counterflowing regime, the pictures in figure 3.3.4 depict a slice in the x or y plane of the Swirl Generator with an isocurve of axial velocity.
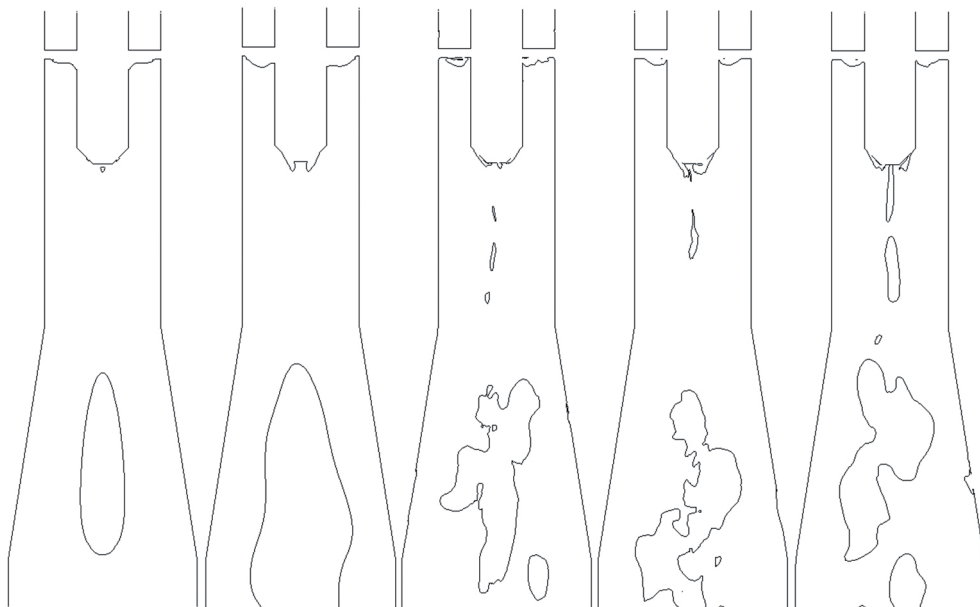

Figure 3.3.4: *Slice in the x or y plane of the Swirl Generator with an isocurve of axial velocity equal to 0.001 $\frac{m}{s}$. The axial velocity is positive in flow direction, i.e. from top to bottom. From left to right: k-ϵ, k-ω SST, k-ω SST SAS, Spalart-Allmaras DDES and Spalart-Allmaras IDDES.*

The comparison of the reversed flow regions in figure 3.3.4 further emphasizes the higher amount of detailed flow structures of the hybrid models, compared to the RANS models. As elucidated in subsection 2.3, the difference in the amount of modeled turbulence can be addressed to the turbulence length scale applied by the turbulence model. Here, the turbulent kinematic viscosity represents a criterion to measure the amount of modeled turbulence and the turbulence length scale respectively. Figure 3.3.5 therefore, presents the turbulent kinematic viscosity ν_t normalized by the molecular kinematic viscosity ν. All pictures in figure 3.3.5 depict a slice in the x or y plane of the Swirl Generator.
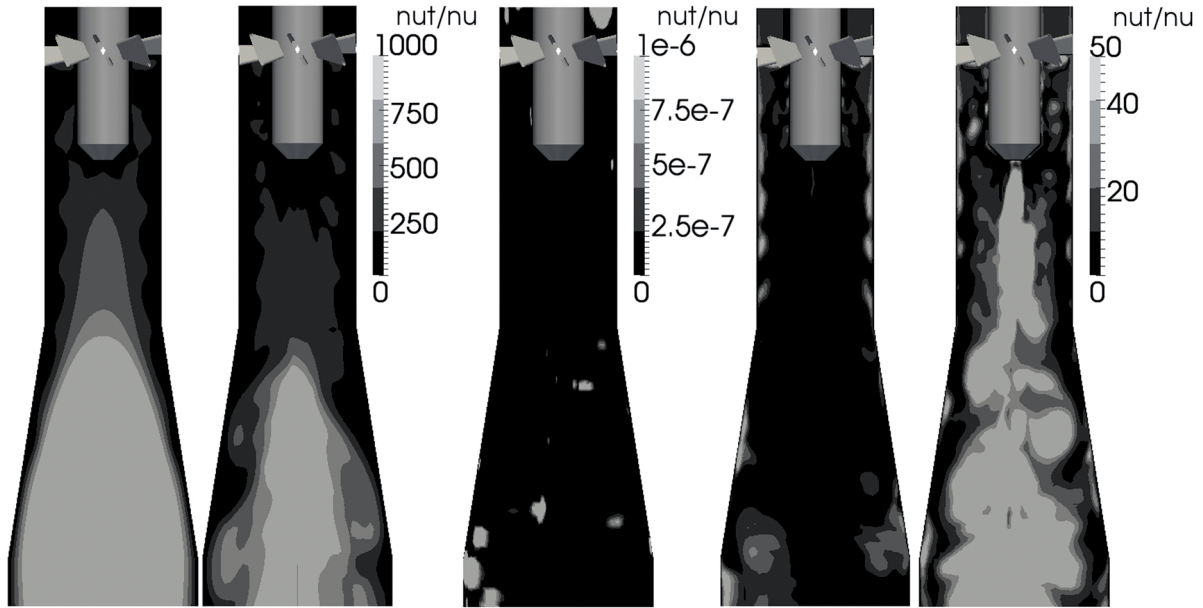
Figure 3.3.5: *Slice in the x or y plane of the Swirl Generator colored by the ratio of the turbulent kinematic viscosity to the molecular kinematic viscosity. From left to right: k-ϵ, k-ω SST, k-ω SST SAS, Spalart-Allmaras DDES and Spalart-Allmaras IDDES.*

As figure 3.3.5 shows, the amount of modeling significantly differs between the RANS and hybrid models. While the ratio of $\frac{\nu_t}{\nu}$ of the RANS models in the downstream of the Swirl Generator reaches from a value below 250 to approximately 1000, the hybrid models contain mainly a ratio of $\frac{\nu_t}{\nu}$, which reaches from 0 to approximately 50 in the same region. This significant difference underlines the limited capability of RANS models to simulate unsteady flow structures. In terms of flow physics, the RANS simulations contain an effective viscosity which is 1000 times higher than the molecular viscosity. This high diffusivity, in turn, leads to a dampening of the rotational flow structures or even hindering of the breakup of the helical vortex as can be seen in figure 3.3.3. A comparison between the simulations based on the hybrid models shows, that the *k-ω SST SAS* model gives a relatively low ratio of $\frac{\nu_t}{\nu}$, which reaches from 0 to approximately $1 * 10^{-6}$. Here, the calculated turbulent kinematic viscosity is significantly smaller than the molecular kinematic viscosity. The simulations based on the *Spalart-Allmaras DDES* and *Spalart-Allmaras IDDES* model on the other hand, include a ratio of $\frac{\nu_t}{\nu}$ below 10 in the majority of the regions of interest, whereas the simulation based on the *Spalart-Allmaras IDDES* model exhibits a higher amount of regions with a value up to 50. As is clear from the explanation of the hybrid RANS-LES strategies, the amount of modeling and the turbulence length scale used by the turbulence model respectively has an influence on the simulation of unsteady flow structures. A comparison of the rotational flow structures in the simulations therefore, helps to further illustrate the difference between RANS and hybrid models. Figure 3.3.6 presents an isosurface of the second invariant of the velocity gradient tensor, which represents the rotational flow structures.

Figure 3.3.6: *Isosurface of the second invariant of the velocity gradient tensor (colored by axial velocity in $[\frac{m}{s}]$). The axial velocity is positive in flow direction, i.e. from top to bottom. From left to right: k-ϵ, k-ω SST, k-ω SST SAS, Spalart-Allmaras DDES and Spalart-Allmaras IDDES.*

The pictures of figure 3.3.6 show considerable differences in the amount of rotational flow structures and reflect the conclusions drawn from figures 3.3.3 and 3.3.5. Here, the differences are most noticeable from a comparison of the region in the draft tube. As can be seen, the simulations based on the hybrid models are containing significantly more rotational flow structures than the simulations based on the RANS models. While the simulation based on the k-ω *SST* model still includes a relatively low amount of rotational flow structures, the k-ϵ model dampens all unsteady flow structures in the draft tube. The isosurface of the second invariant of the velocity gradient tensor, colored by axial velocity, furthermore shows the occurrence of a reversed flow in the center-region of the draft tube. Here, black refers to an axial velocity between -0.5 $\frac{m}{s}$ to 0 $\frac{m}{s}$. As can be seen in the comparison of the simulations with the experimental data, the results of the simulations are not in sufficient agreement with the experimental measurements. A failure in the prediction of the separation in general and around the blades in particular, is assumed to be a potential reason for this mismatch. Figure 3.3.7 shows the separation around the blades as well as streamlines of fluid particles, which pass the velocity measurement lines.
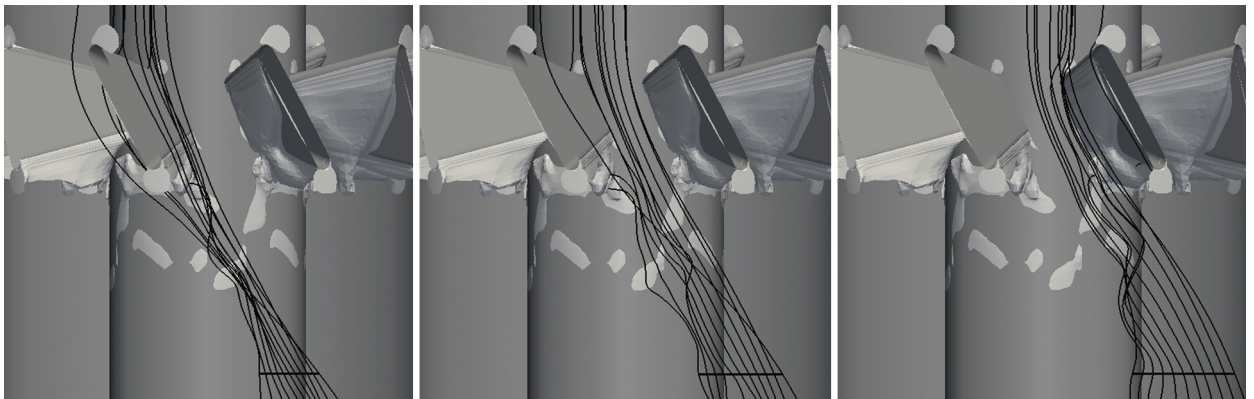

Figure 3.3.7: *Separation around the blades , visualized by an isosurface of axial velocity equal to -0.001$\frac{m}{s}$, and streamlines through the Laser Doppler Velocimetry measurement lines, which are indicated by black horizontal lines. From left to right: L1, L2, L3. The pictures are based on the simulation with the Spalart-Allmaras DDES model and a discharge of 25 $\frac{m^3}{h}$.*

As is clear from figure 3.3.7, the separation below the blades has probably an effect on the velocity along the

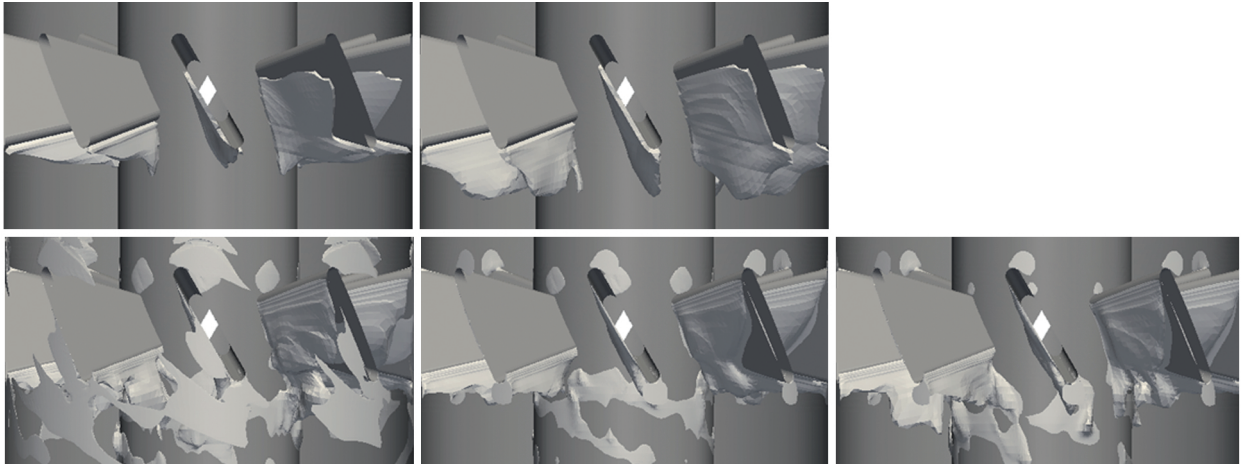measurement lines. Figure 3.3.8 therefore, presents the separation zones of the different simulations.

Figure 3.3.8: *Separation around the blades visualized by an isosurface of axial velocity equal to -0.001 $\frac{m}{s}$. Upper row: RANS models from left to right: k-ϵ, k-ω SST; lower row: hybrid models from left to right: k-ω SST SAS, Spalart-Allmaras DDES, Spalart-Allmaras IDDES.*

As can be seen in figure 3.3.8, the separation zone considerably differs between the simulations. This applies to the beginning, dimension and shape of the separation. A comparison of the beginning of the separation zones however, shows similarities among RANS and hybrid models respectively. While the separation zones of the simulations based on the hybrid models begin to develop right after the rounding of the blades, the separation zones of the simulations based on the RANS models begin noticeably later and start with a higher thickness of the separation zone. These similarities between RANS and hybrid models respectively might be addressed to the differences in the near-wall treatment of the turbulence models, i.e. that the hybrid models are integrating to the wall, in contrast to the high-Reynolds number RANS models.

3.3.3 Comparison with Experimental Data

In order to assess the simulations in terms of their accuracy, the results are compared with the experimental data given by the Institute of Fluid Mechanics and Hydraulic Machinery, University of Stuttgart. Due to the poor development of the simulations based on the k-ω SST SAS and *Spalart-Allmaras IDDES* model, the comparison is limited to the k-ϵ, k-ω SST and *Spalart-Allmaras DDES* model. Figure 3.3.9 gives a schematic overview of the compared data and the corresponding measurement positions in the Swirl Generator. A more detailed description of the experimental measurements is depicted in Appendix B.
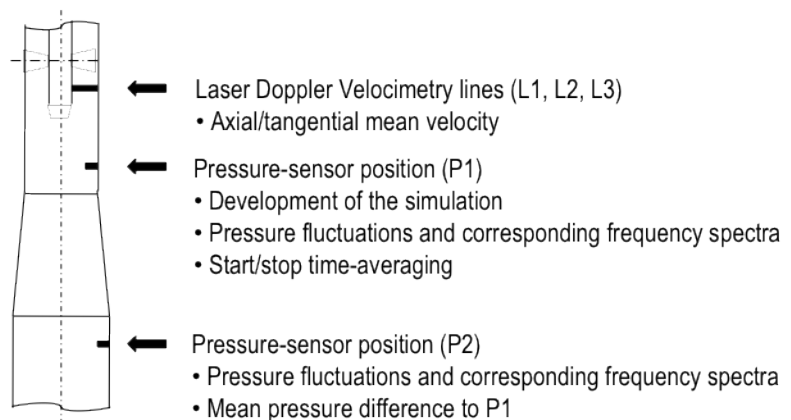

Figure 3.3.9: *Schematic figure of the data output and corresponding measurement positions.*

As figure 3.3.9 shows, the comparison includes the axial and tangential mean velocity at three radial lines

downstream of the blades, the pressure fluctuations as well as the corresponding frequency spectra at positions *P1* and *P2*, and the mean pressure difference between *P1* and *P2*. In order to prepare the simulated data for the comparison, the following methodologies are applied: the pressure fluctuations are obtained by plotting the pressure signal of positions *P1* and *P2*. The resulting signal is normalized to *0* by subtracting the latest mean value of the pressure at the corresponding position. This signal is furthermore used to determine the frequency spectra. For this, the *Fast Fourier Transform*-function of *MATLAB*[13] is applied to show the frequency spectrum of each signal. In accordance with the requirements of this function, the Fourier transformation is based on a period of the pressure signal with a constant timestep. For the axial and tangential mean velocity and pressure difference between positions *P1* and *P2* on the other hand, the required properties are time averaged with the help of the *fieldAverage*-function object. In order to compare the data, the corresponding values are then extracted with the *sample*-function object. For the comparison of the mean pressure difference, the values of a cell, which is closest to the vertical position of the pressure-sensor of the experiment, is extracted. For the comparison of the axial and tangential velocity along the three radial lines on the other hand, 22 values along a line between two points are sampled. Here, the two points represent the beginning and end of the experimental measurement lines. The extracted values of the velocity are finally allocated to the radial position of the experimental values. It is assumed, that possible deviations of the extracted positions from the experimental positions are negligible due to the short distances between the experimental measurement points. The pressure signal at position *P1* is furthermore used to asses the development of the simulation and to coordinate the time averaging of the relevant properties. This averaging should ideally be executed over a period or a multiple of a period. In order to assess the generation of the averaged properties of the simulations, figure 3.3.10 presents the start and stop of the time averaging based on the pressure signal at position *P1*.

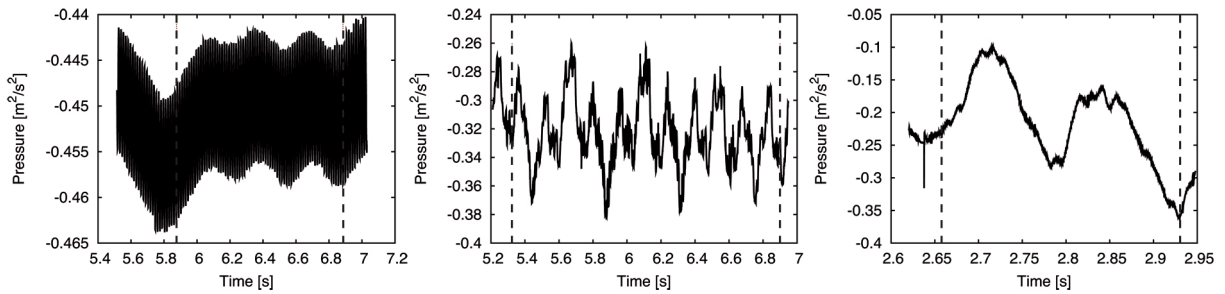

Figure 3.3.10: *Period of field averaging for the generation of time averaged properties determined by the pressure signal at position P1. Dashed lines indicate the start and stop of the field averaging. From left to right: k-ε, k-ω SST and Spalart-Allmaras DDES.*

It is clear from the pictures in figure 3.3.10, that the averaging for the simulations contains approximately: 3 periods based on the *k-ε*, 11 periods based on the *k-ω SST* and 2 periods based on the *Spalart-Allmaras DDES* model. As is explained in section 2, the helical vortex introduces a dynamic load, caused by the pressure fluctuations, on the hydraulic system. Figure 3.3.11 therefore, presents the pressure fluctuations of the experiment and simulations. While the experiment contains a signal length of 2 *s*, the signal length of the simulations corresponds approximately to the period of time averaging. In order to allow for comparability, all pressure signals are normalized to *0*. For this, the latest mean value of the pressure at the corresponding position has been subtracted from the actual pressure signal of the simulations.

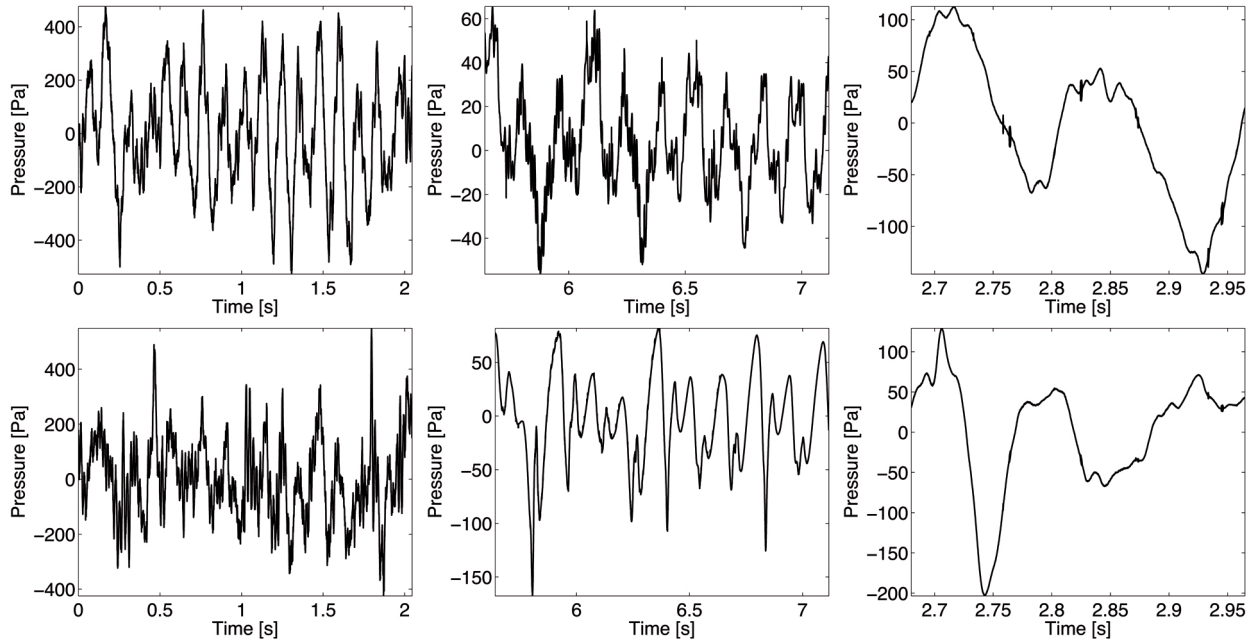[13]Throughout the text MATLAB refers to MATLAB®. MATLAB is a trademark or registered trademark of The MathWorks, Inc.

Figure 3.3.11: *Pressure fluctuations at positions P1 and P2. Upper row: position P1; lower row: position P2; From left to right: Experiment, k-ω SST and Spalart-Allmaras DDES.*

The pictures in figure 3.3.11 show, that the different scales of the plots, which result from the corresponding signals lengths, do not allow for an exact comparison of the pressure fluctuations. While the signal length of the simulation based on the *Spalart-Allmaras DDES* model is too short to draw any conclusion regarding the agreement of the development of the fluctuations with the experiment, the simulation based on the *k-ω SST* model indicates a relatively similar development of the fluctuations at position *P1*. The fluctuations of the simulation based on the *k-ω SST* model at position *P2* on the other hand, contain a relatively low amount of noise, in contrast to the experiment. A comparison of the amplitudes of the fluctuations furthermore, shows relatively big differences between experimental and simulated data. Here, the simulations tend to give significantly smaller amplitudes of the pressure fluctuations. In order to compare the frequency spectra at positions *P1* and *P2*, figure 3.3.12 presents the Fourier transformed pressure signals, based on the pressure fluctuations shown in figure 3.3.11.
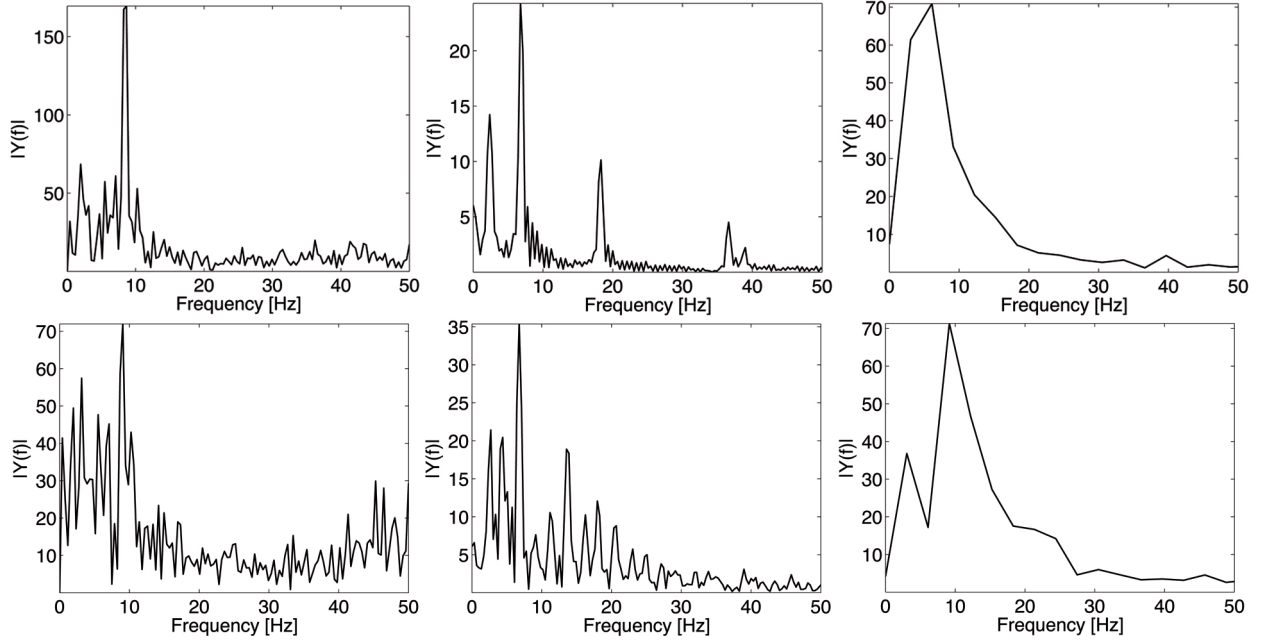
Figure 3.3.12: *Frequency spectrum of the pressure signal at positions P1 and P2. Upper row: position P1; lower row: position P2; From left to right: Experiment, k-ω SST and Spalart-Allmaras DDES.*

As to be seen in all pictures of figure 3.3.12, the frequency spectra of the experiment and the simulations show a relatively similar dominant frequency. These frequencies are assumed to represent the frequency of the helical vortex. It should be stressed, that the pressure signal of the simulation based on the *Spalart-Allmaras DDES* model does not provide a sufficient amount of data in order to give detailed frequency spectra. The corresponding frequency spectra are however, shown for the sake of completeness. Based on the Fourier plot with a frequency range close to the dominant frequency, the frequencies of the helical vortex in table 3.3.3 are obtained.

Table 3.3.3: Dominant frequency of the Fourier transformed pressure signal at positions *P1* and *P2*.

	Experiment	k-ω SST	S-A DDES
Frequency at *P1* in [*Hz*]	8.7	6.8	6.1
Deviation from the experimental frequency in [%]	0	22	30
Frequency at *P2* in [*Hz*]	9.1	6.8	9.2
Deviation from the experimental frequency in [%]	0	25	1

As table 3.3.3 shows, the dominant frequency of the experiment at position *P1* is 8.7 *Hz*. Here, the simulations based on the *k-ω SST* and *Spalart-Allmaras DDES* model give a lower frequency of 6.8 *Hz* and 6.1 *Hz* respectively. These differences correspond to a deviation from the experimental frequency of 22% and 30%. At position *P2*, the experimental data gives a dominant frequency of 9.1 *Hz*. The simulations based on the *k-ω SST* and *Spalart-Allmaras DDES* model on the other hand, contain a dominant frequency of 6.8 *Hz* and 9.2 *Hz* respectively. These differences refer to a deviation from the experimental frequency of 25% and 1%. Table 3.3.4 presents a comparison of the mean pressure difference between position *P1* and *P2*.

Table 3.3.4: Mean pressure difference between position *P1* and *P2*.

	Experiment	k-ϵ	k-ω SST	S-A DDES
Pressure difference in [Pa]	237	437	169	263
Deviation from the experimental pressure difference in [%]	0	85	29	11

As is clear from table 3.3.4, the mean pressure difference in the experiment is 237 *Pa*. In contrast to the experimental data, the mean pressure difference of the simulations is: 437 *Pa* for the simulation based on the k-ϵ, 169 *Pa* based on the k-ω *SST* and 263 *Pa* based on the *Spalart-Allmaras DDES* model. These differences correspond to a deviation from the experimental data of 85% (k-ϵ), 29% (k-ω *SST*) and 11% (*Spalart-Allmaras DDES*). Figure 3.3.13 presents the axial and tangential mean velocity profile of the experimental data in contrast to the velocity profiles of the simulations along the Laser Doppler Velocimetry line *L1*.
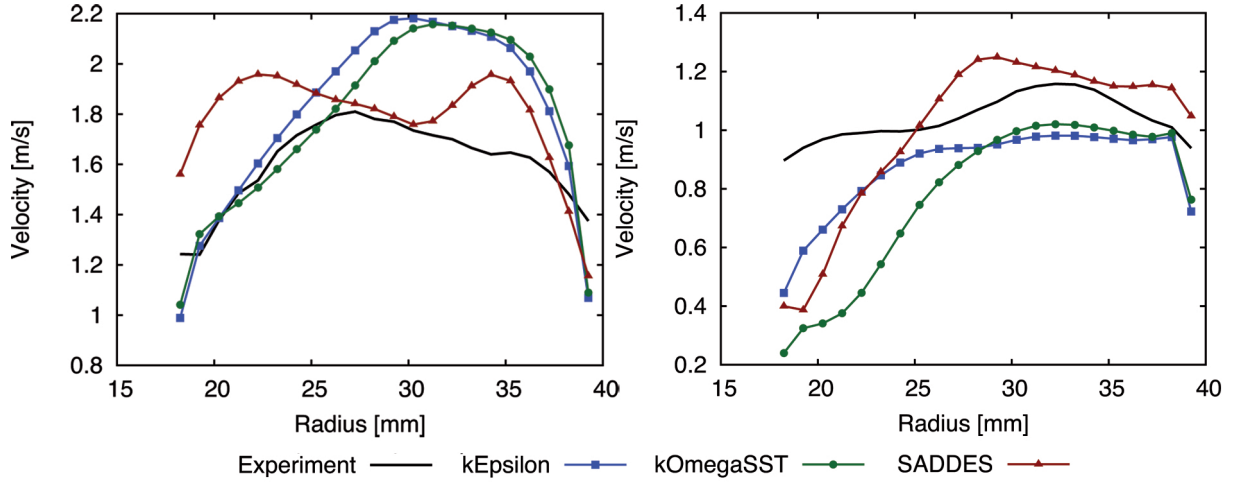
Figure 3.3.13: *Axial and tangential velocity along the measurement line "L1". Left: axial velocity; right: tangential velocity.*

Both pictures in figure 3.3.13 show relatively big differences of the axial and tangential mean velocity profiles between experimental and simulated data. Furthermore, it can be seen that the velocity profile of the simulation based on the *Spalart-Allmaras DDES* model differs from the relatively similar profiles of the simulations based on the RANS models. A comparison of the mean velocity profiles along the Laser Doppler Velocimetry line *L2* can be seen in figure 3.3.14.
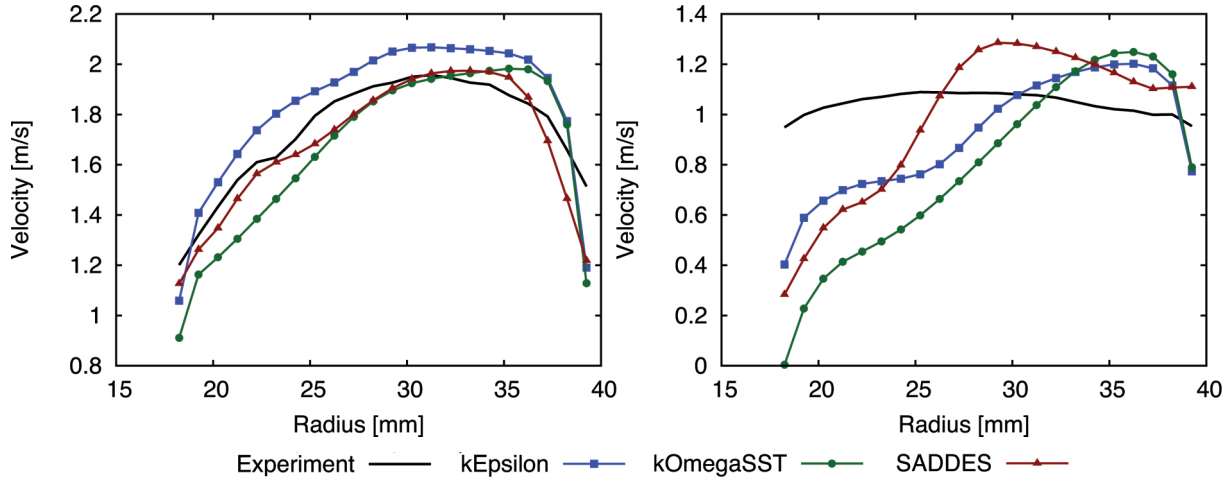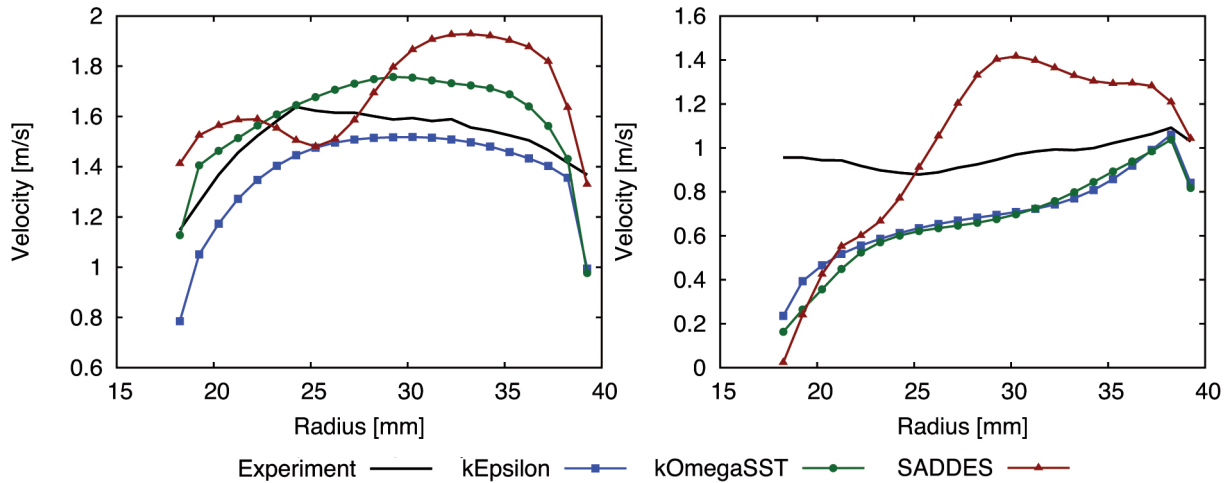
Figure 3.3.14: *Axial and tangential velocity along the measurement line "L2". Left: axial velocity; right: tangential velocity.*

The left picture of figure 3.3.14 shows that the axial mean velocity curves of the experimental and simulated data have a relatively similar shape, even though the magnitudes of the velocity do not match exactly. The tangential mean velocity along the measurement line *L2* on the other hand shows again a relatively large mismatch of experimental and simulated data. Figure 3.3.15 finally represents a comparison of the mean velocity profiles along the Laser Doppler Velocimetry line *L3*.

Figure 3.3.15: *Axial and tangential velocity along the measurement line "L3". Left: axial velocity; right: tangential velocity.*

As in figure 3.3.13, both pictures in figure 3.3.15 show inequalities between experimental an simulated data. Once again, similarities in the shape of the curves of the mean velocity profiles of the RANS models can be identified.

3.4 Results for a Discharge of 40 $\frac{m^3}{h}$

This subsection presents the results of the simulations with a discharge of 40 $\frac{m^3}{h}$. For this, the subsection firstly gives an overview of the development and performance of the simulations. Secondly, it shows an analysis of the flow as well as the turbulence models, before a comparison of the results with the experimental data is finally presented.

3.4.1 Development and Performance

As is clear from section 3, the development of unsteady simulations plays an important role in the evaluation of their results. Figures 3.4.1 and 3.4.2 therefore, depict the development of the simulations with a discharge of $40 \frac{m^3}{h}$ based on the pressure signal at position *P1*.
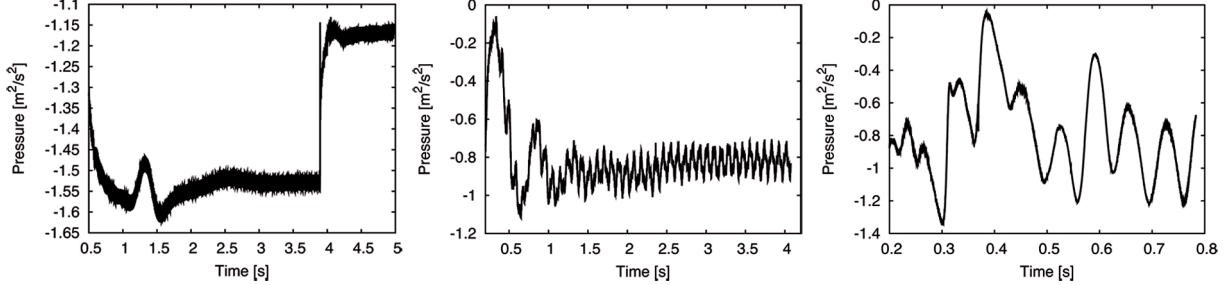

Figure 3.4.1: *Development of the pressure signal at position P1. From left to right: k-ε, k-ω SST and k-ω SST SAS.*

As all pictures from figures 3.4.1 and 3.4.2 show, the simulations with a discharge of $40 \frac{m^3}{h}$ simulate less time of the flow in comparison with the simulations with a discharge of $25 \frac{m^3}{h}$. This difference can be addressed to the corresponding smaller timestep, which is required due to the higher velocity of the flow. As can be seen in the left and middle picture of figure 3.4.1, the simulations based on the *k-ε* and *k-ω SST* model show with approximately 5 seconds and 4 seconds of the flow a relatively well developed pressure signal. It is clear from the right picture of figure 3.4.1, that the progress of the simulation based on the *k-ω SST SAS* model is with approximately 0.8 seconds of the flow relatively short. However, it does not include intermediate deflections as they occurred in the development of the simulation with a discharge of $25 \frac{m^3}{h}$. Figure 3.4.2 presents the development of the simulations based on the *Spalart-Allmaras DDES* and *Spalart-Allmaras IDDES* model.
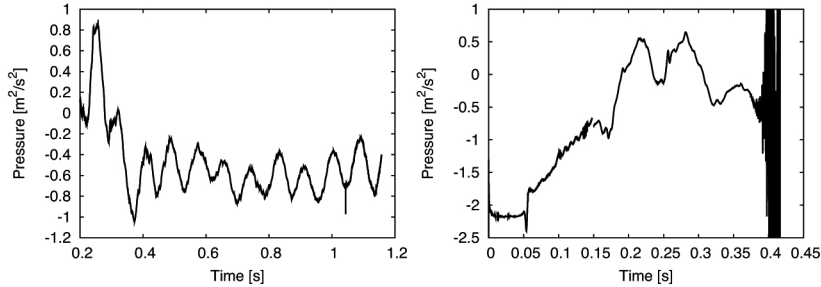

Figure 3.4.2: *Development of the pressure signal at position P1. Left: Spalart-Allmaras DDES; right: Spalart-Allmaras IDDES.*

Both pictures in figure 3.4.2 show a relatively similar development of the simulations in comparison with the simulations with a discharge of $25 \frac{m^3}{h}$. The left picture of figure 3.4.2 presents with approximately 1.2 seconds of simulation a relatively well developed pressure signal of the simulation based on the *Spalart-Allmaras DDES* model. The progress of the simulation based on the *Spalart-Allmaras IDDES* model on the other hand, is with approximately 0.4 seconds of simulation relatively short. Similar to the simulation with a discharge of $25 \frac{m^3}{h}$, the simulation based on the *Spalart-Allmaras IDDES* model contains severe deflections, which might be a result of a too large timestep in the transient phase of the simulation. As is clear from section 1, the computational effort represents a substantial criteria in the assessment of numerical simulations. Table 3.4.1 therefore, presents the performance of the simulations based on their latest timesteps.

Table 3.4.1: Performance of the simulations for a discharge of 40 $\frac{m^3}{h}$. The values are normalized for simulations decomposed on 32 processors and a maximum Courant number of 1. Influences of different Courant numbers or degrees of decomposition on the performance of the simulations are neglected. The calculations are based on the timestep, required time to calculate one timestep and maximum Courant number of the latest executed timestep of the simulation.

	k-ϵ	k-ω SST	k-ω SST SAS	S-A DDES	S-A IDDES
Timestep in [s]	$6*10^{-5}$	$6*10^{-5}$	$8*10^{-6}$	$7*10^{-6}$	$2*10^{-5}$
Time to calculate 1 timestep in [s]	75	51	171	255	158
Time divided by the average time of the RANS models	1	1	3	4	3
Time to simulate 1 s of the flow in [d]	15	10	247	434	118
Time divided by the average time of the RANS models	1	1	20	35	10

Table 3.4.1 presents the timestep for a maximum Courant number of 1, the required time to calculate 1 timestep and to simulate 1 second of the flow. Furthermore, the corresponding times divided by the average time of the RANS models are shown in order to express the relatively high computational effort of the hybrid models in contrast to the RANS models. As can be seen in table 3.4.1, the simulation of 1 second of the flow on 32 processors requires approximately 15 days based on the k-ϵ, 10 days based on the k-ω SST, 247 days based on the k-ω SST SAS, 434 days based on the *Spalart-Allmaras DDES* and 118 days based on the *Spalart-Allmaras IDDES* model. These differences of the computational effort of the simulations can firstly be addressed to the differences in the required timestep. As is clear from table 3.4.1, the required timestep of the RANS models for a maximum Courant number of 1 is $6*10^{-5}$ seconds. The timestep of the other simulations on the other hand corresponds to: $8*10^{-6}$ seconds for simulations based on the k-ω SST SAS, $7*10^{-6}$ seconds for simulations based on the *Spalart-Allmaras DDES* and $2*10^{-5}$ seconds for simulations based on the *Spalart-Allmaras IDDES* model. A comparison of the required time to calculate 1 timestep furthermore shows a contribution to the differences in the computational effort. The simulations require: 75 seconds based on the k-ϵ, 51 seconds based on the k-ω SST, 171 seconds based on the k-ω SST SAS, 255 seconds based on the *Spalart-Allmaras DDES* and 158 seconds based on the *Spalart-Allmaras IDDES* model. The relatively great difference between the simulations based on RANS and hybrid models can be addressed to the higher number of cells of the meshes for the hybrid models and the higher number of iterations, which are executed by the hybrid models to calculate the pressure of each timestep. The average number of iterations of non-orthogonal corrector steps of the simulations is presented in table 3.4.2. Moreover, the difference between the corresponding hybrid model to the average number of iterations of the RANS models is shown.

Table 3.4.2: Comparison of the average number of iterations for the calculation of the pressure based on the latest timestep of the simulations and the non-orthogonal corrector step.

	k-ϵ	k-ω SST	k-ω SST SAS	S-A DDES	S-A IDDES
Average number of iterations per timestep	373	277	415	663	348
Difference to the average number of iterations of the RANS models	-	-	90	338	23

It is clear from table 3.4.2, that the hybrid models are generally tending to require significantly more iterations to calculate the pressure per timestep. The latest executed timestep of the simulations contains an average of: 373 iterations based on the k-ϵ, 227 iterations based on the k-ω SST, 415 iterations based on the k-ω SST SAS, 663 iterations based on the *Spalart-Allmaras DDES* and 348 iterations based on the *Spalart-Allmaras IDDES* model.

3.4.2 Evaluation of the Results

As elucidated in subsection 2.1, the flow of the helical vortex phenomenon firstly contains quasi-cylindrical as well as helical isosurfaces of pressure. Moreover, it includes a region of reversed flow. An evaluation of the presence of these features allows a first assessment of the simulations, i.e. whether the turbulence model was capable of simulating the striking flow features of the helical vortex or not. For this, figure 3.4.3 presents an isosurface of pressure and axial velocity.
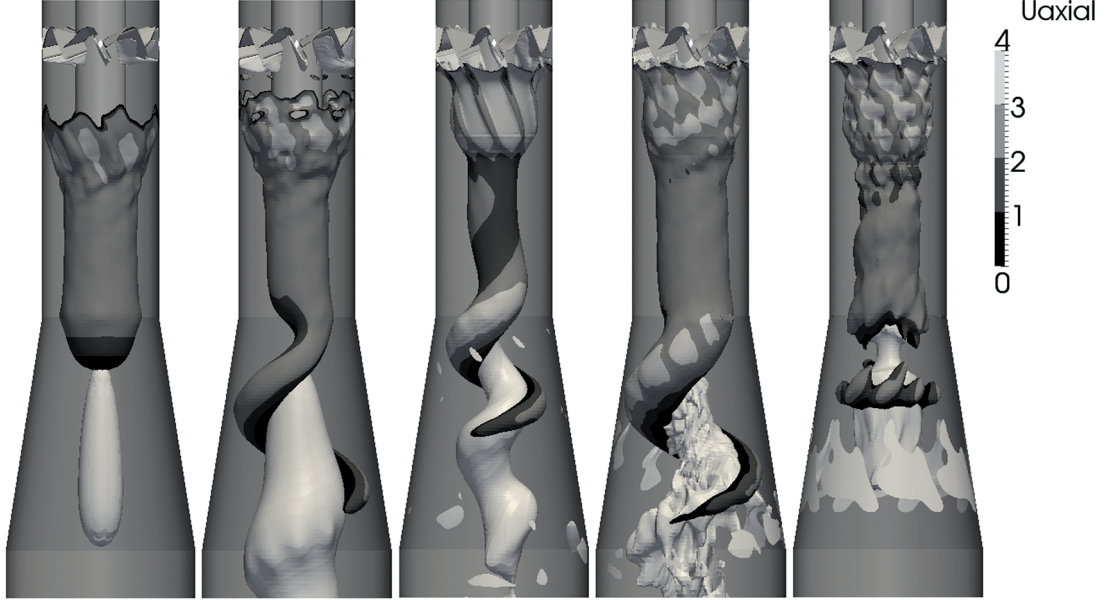
Figure 3.4.3: *Isosurface of pressure (colored by axial velocity in $[\frac{m}{s}]$) and axial velocity equal to -0.001 $\frac{m}{s}$ (solid color). The axial velocity is positive in flow direction, i.e. from top to bottom. From left to right: k-ϵ, k-ω SST, k-ω SST SAS, Spalart-Allmaras DDES and Spalart-Allmaras IDDES.*

The pictures in figure 3.4.3 depict an isosurface of pressure, which represents the helical vortex, and axial velocity equal to -0.001 $\frac{m}{s}$, which corresponds to the region of reversed flow. As can be seen, all turbulence models except the k-ϵ and *Spalart-Allmaras IDDES* model were capable of simulating the helical vortex. While the failure of the k-ϵ model is assumed to be the result of a severe dampening of the unsteady flow in terms of too high an amount of modeled turbulence, the failure of the *Spalart-Allmaras IDDES* model might be the result of a poor development of the simulation. The comparably small dimension of the helical vortex of the k-ω SST SAS model is assumed to be the result of a not fully developed simulation. As expected, the helical vortex rotates asymmetrically around a region of reversed flow in the center-line. The pictures of the simulations based on the k-ω SST and *Spalart-Allmaras DDES* model show, that the helical section of the vortex rope starts at the beginning of the draft tube. Provided that the corresponding simulations are fully developed, it can furthermore be seen, that the helical vorticies have relatively similar dimensions, in comparison with those of the simulations with a discharge of 25 $\frac{m^3}{h}$. The isosurface of pressure, colored by axial velocity, therefore firstly shows the asymmetric distribution of pressure, caused by the helical vortex. The distribution of the axial velocity furthermore shows the connection between the vortex rope and the counterflowing region and indicates, that the helical vortex rotates around its own axis. Here, black refers to an axial velocity between 0 $\frac{m}{s}$ and 1 $\frac{m}{s}$. In contrast to the simulations with a discharge of 25 $\frac{m^3}{h}$, the pictures in figure 3.4.3 do not show a clear tendency of a more detailed flow of the simulations based on the hybrid models compared with the simulations based on the RANS models. Only the region of reversed flow of the simulation based on the *Spalart-Allmaras DDES* model presents significantly more details. As to be seen in figure 3.4.5, the relatively low amount of details of the simulation based on the k-ω SST SAS model falls probably back on the relatively high amount of turbulence modeling. The failure of the *Spalart-Allmaras IDDES* model on the other hand is assumed to be the result of a relatively poor developed simulation. In order to give a more distinct picture of the counterflowing regime, the pictures in figure 3.4.4 depict a slice in the x or y plane of the Swirl Generator with an isocurve of axial velocity.
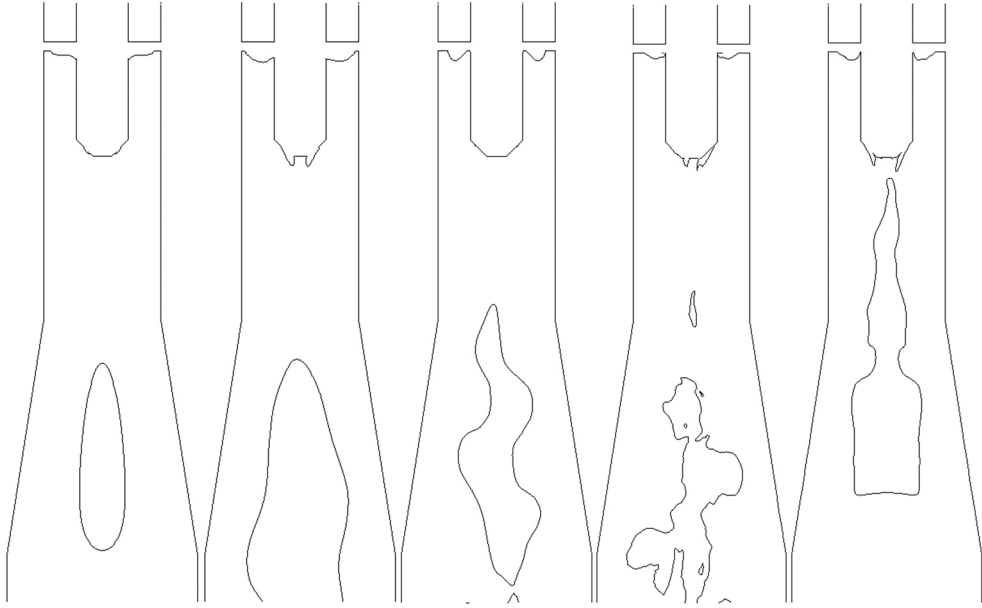
Figure 3.4.4: *Slice in the x or y plane of the Swirl Generator with an isocurve of axial velocity equal to 0.001 $\frac{m}{s}$. The axial velocity is positive in flow direction, i.e. from top to bottom. From left to right: k-ε, k-ω SST, k-ω SST SAS, Spalart-Allmaras DDES and Spalart-Allmaras IDDES.*

The comparison of the reversed flow regions in figure 3.4.4 further emphasizes the conclusions drawn from figure 3.4.3. As explained in subsection 2.3, the difference in the amount of modeled turbulence can be addressed to the turbulence length scale applied by the turbulence model. Here, the turbulent kinematic viscosity represents a criterion to measure the amount of modeled turbulence and the turbulence length scale respectively. The turbulent kinematic viscosity ν_t normalized by the molecular kinematic viscosity ν is therefore, presented in figure 3.4.5. All pictures in figure 3.4.5 depict a slice in the x or y plane of the Swirl Generator.
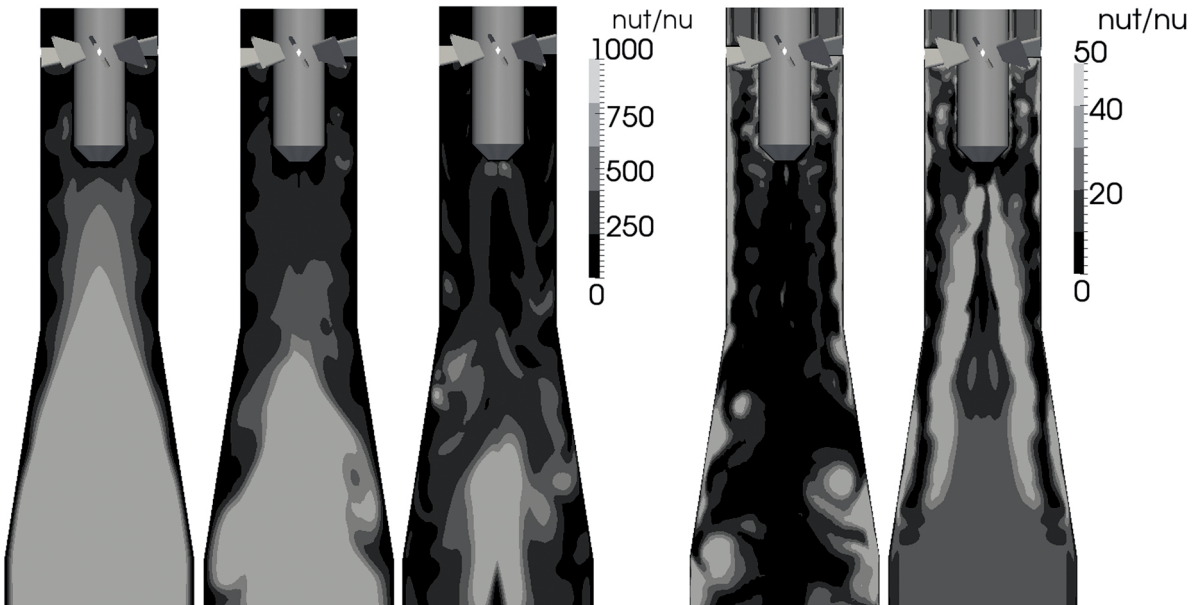

Figure 3.4.5: *Slice in the x or y plane of the Swirl Generator colored by the ratio of the turbulent kinematic viscosity to the molecular kinematic viscosity. From left to right: k-ε, k-ω SST, k-ω SST SAS, Spalart-Allmaras DDES and Spalart-Allmaras IDDES.*

As to be seen in figure 3.4.5, the amount of modeling of the turbulence models differs from the results of the simulations with a discharge of 25 $\frac{m^3}{h}$. While the simulations based on the RANS, *Spalart-Allmaras DDES*

39

and *Spalart-Allmaras IDDES* models show a relatively similar behavior compared to the simulations with a discharge of 25 $\frac{m^3}{h}$, the simulation based on the *k-ω SST SAS* model contains a considerably higher amount of modeled turbulence. The ratio of $\frac{\nu_t}{\nu}$ of the simulations based on the RANS models and the *k-ω SST SAS* model reaches from a value below 250 to approximately 1000 in the downstream of the Swirl Generator. The simulations based on the *Spalart-Allmaras DDES* and *Spalart-Allmaras IDDES* on the other hand, contain mainly a ratio of $\frac{\nu_t}{\nu}$, which reaches from 0 to approximately 50 in the same region. Once again, this significant difference underlines the limited capability of RANS models to simulate unsteady flow structures. Moreover, it shows a great variation of the adjustment of the *k-ω SST SAS* model to unsteady flow features. In terms of flow physics, the simulations with a ratio of $\frac{\nu_t}{\nu}$ up to 1000 contain an effective viscosity which is 1000 times higher than the molecular viscosity. This highly diffusive flow, in turn, leads to a dampening of the rotational flow structures or even hindering of the breakup of the helical vortex as to be seen in figure 3.4.3. As is clear from the explanation of the hybrid RANS-LES strategies, the amount of modeling and the turbulence length scale used by the turbulence model respectively has an influence on the simulation of unsteady flow structures. A comparison of the rotational flow structures in the simulations therefore, helps to further illustrate the difference between RANS and hybrid models. Figure 3.4.6 presents an isosurface of the second invariant of the velocity gradient tensor, which represents the rotational flow structures.
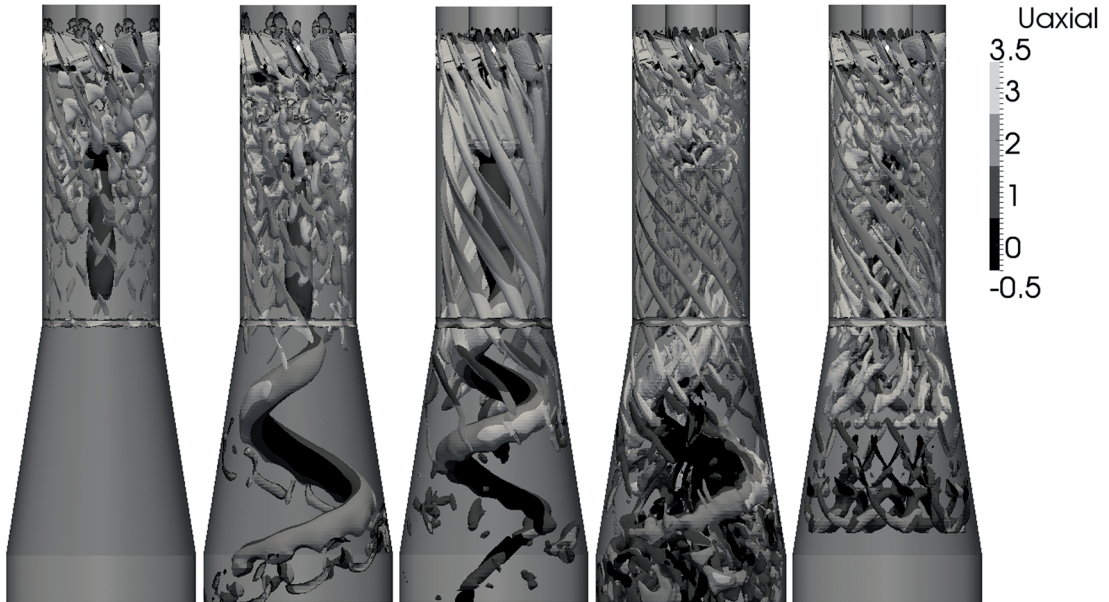

Figure 3.4.6: *Isosurface of the second invariant of the velocity gradient tensor (colored by axial velocity in $[\frac{m}{s}]$). The axial velocity is positive in flow direction, i.e. from top to bottom. From left to right: k-ϵ, k-ω SST, k-ω SST SAS, Spalart-Allmaras DDES and Spalart-Allmaras IDDES.*

All pictures of figure 3.4.6 show considerable differences in the amount of rotational flow structures and reflect the conclusions drawn from figures 3.4.3 and 3.4.5. The differences are most noticeable from a comparison of the region in the draft tube. It can be seen, that the simulation based on the *Spalart-Allmaras DDES* model includes significantly more rotational flow structures than the simulations based on the RANS models. While the simulation based on the *k-ω SST* model still includes a relatively low amount of rotational flow structures, the *k-ϵ* model dampens all unsteady flow structures in the draft tube. Moreover, it is clear from figure 3.4.6, that the simulation based on the *k-ω SST SAS* model contains a lower amount of rotational flow structures. In the simulation based on the *Spalart-Allmaras IDDES* model, no helical vortex can be identified, which further emphasizes the conclusion drawn from figure 3.4.3. The isosurface of the second invariant of the velocity gradient tensor, colored by axial velocity, furthermore shows the occurrence of a reversed flow in the center-region of the draft tube. Here, black refers to an axial velocity between -0.5 $\frac{m}{s}$ to 0.5 $\frac{m}{s}$. As can be seen in the comparison of the simulations with the experimental data, the results of the simulations are not in sufficient agreement with the experimental measurements. A failure in the prediction of the separation in general and around the blades in particular, is assumed to be a potential reason for this mismatch. It is clear from figure 3.3.7, which is depicted in the evaluation of the results for a discharge of 25 $\frac{m^3}{h}$, that the separation

below the blades has probably an effect on the velocity along the measurement lines. Figure 3.4.7 therefore, presents the separation zones of the different simulations.
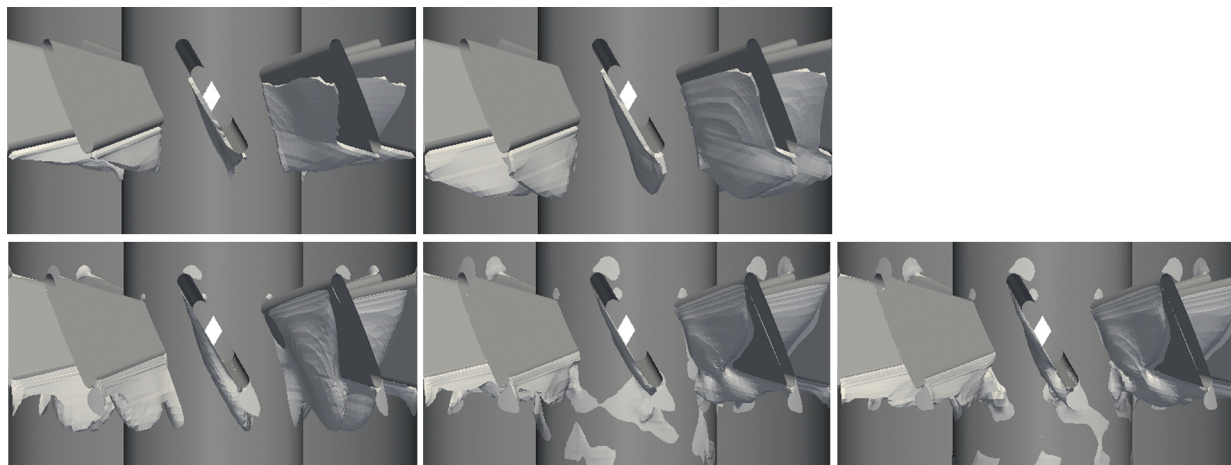

Figure 3.4.7: *Separation around the blades visualized by an isosurface of axial velocity equal to -0.001 $\frac{m}{s}$. Upper row: RANS models from left to right: k-ε, k-ω SST; lower row: hybrid models from left to right: k-ω SST SAS, Spalart-Allmaras DDES, Spalart-Allmaras IDDES.*

As to be seen in figure 3.4.7, the separation zone considerably differs between the simulations. This applies to the beginning, dimension and shape of the separation. A comparison of the beginning of the separation zones however, shows similarities among RANS and hybrid models respectively. While the separation zones of the simulations based on the hybrid models begin to develop right after the rounding of the blades, the separation zones of the simulations based on the RANS models begin noticeably later and start with a higher thickness of the separation zone. These similarities between RANS and hybrid models respectively might be addressed to the differences in the near-wall treatment of the RANS and hybrid models, i.e. that the hybrid models are integrating to the wall, in contrast to the high-Reynolds number RANS models.

3.4.3 Comparison with Experimental Data

In order to assess the simulations regarding their accuracy, the results are compared with the experimental data given by the Institute of Fluid Mechanics and Hydraulic Machinery at the University of Stuttgart. Due to the poor development of the simulations based on the *k-ω SST SAS* and *Spalart-Allmaras IDDES* model, the comparison is limited to simulations based on the *k-ε*, *k-ω SST* and *Spalart-Allmaras DDES* model. An overview of the compared data, corresponding measurement positions and methodologies is given in the comparison of the experimental data with the results for a discharge of 25 $\frac{m^3}{h}$. As is clear from that figure, the comparison with the experimental data requires time averaged pressure and velocity data. This averaging should ideally be executed over a period or a multiple of a period. In order to assess the generation of the averaged properties of the simulations, figure 3.4.8 presents the start and stop of the time averaging based on the pressure signal at Position *P1*.
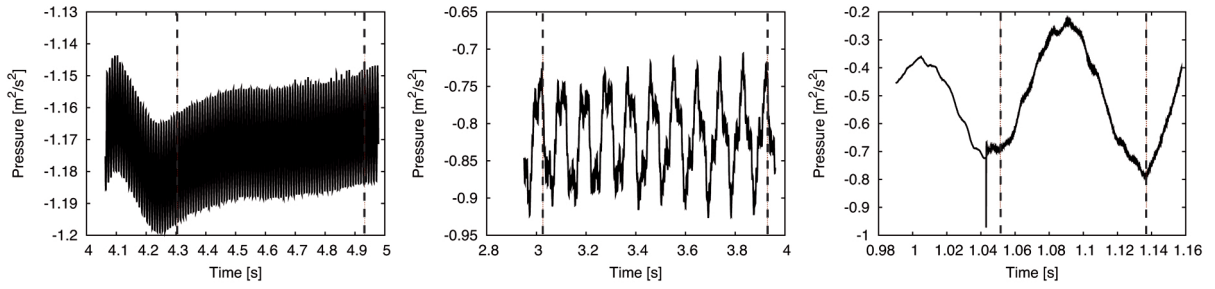
Figure 3.4.8: *Period of field averaging for the generation of time averaged properties determined by the pressure signal at position P1. Dashed lines indicate the start and stop of the field averaging. From left to right: k-ε, k-ω SST and Spalart-Allmaras DDES.*

The pictures in figure 3.4.8 show, that the averaging for the simulations contains approximately: no period based on the *k-ε*, 10 periods based on the *k-ω SST* and 1 period based on the *Spalart-Allmaras DDES* model. The missing periods in the simulation based on the *k-ε* model might be the result of a simulation, which is still developing after the adjustment of the timestep for a maximum Courant number of 1. As is elucidated in section 2, the helical vortex introduces a dynamic load, caused by the pressure fluctuations, on the hydraulic system. Figure 3.4.9 therefore, presents the pressure fluctuations of the experiment and simulations. While the experiment contains a signal length of 2 *s*, the signal length of the simulations corresponds approximately to the period of time averaging. In order to allow for comparability, all pressure signals are normalized to *0*. For this, the latest mean value of the pressure at the corresponding position has been subtracted from the actual pressure signal of the simulations.
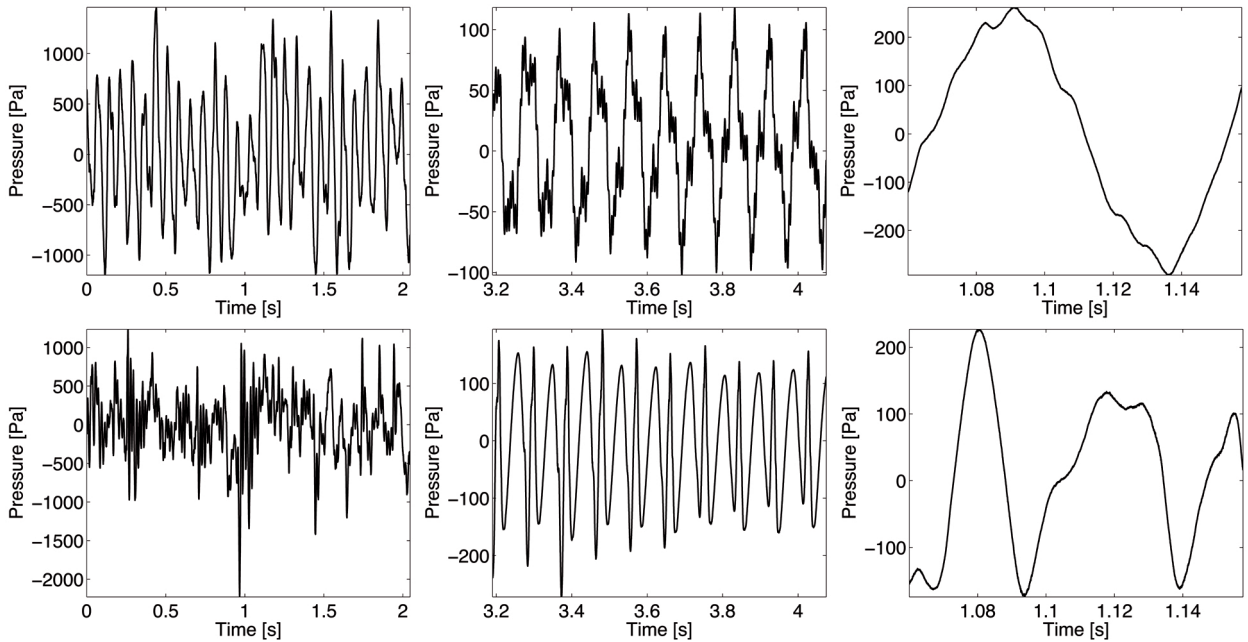

Figure 3.4.9: *Pressure fluctuations at positions P1 and P2. Upper row: position P1; lower row: position P2; From left to right: Experiment, k-ω SST and Spalart-Allmaras DDES.*

As is clear from the pictures in figure 3.4.9, the different scales of the plots, which result from the corresponding signals lengths, do not allow for an exact comparison of the pressure fluctuations. While the signal length of the simulation based on the *Spalart-Allmaras DDES* model is too short to draw any conclusion regarding the agreement of the development of the fluctuations with the experiment, the simulation based on the *k-ω SST* model indicates differences in the development of the fluctuations at both positions. Here, the simulation based on the *k-ω SST* model includes a higher amount of noise at position *P1* and a lower amount at position *P2*, in contrast to the experiment. A comparison of the amplitudes of the fluctuations furthermore, reveals relatively big differences between experiment and simulations. As in the results for a discharge of 25 $\frac{m^3}{h}$, the simulations

tend to give significantly smaller amplitudes of the pressure fluctuations. In order to compare the frequency spectra at positions *P1* and *P2*, figure 3.4.10 presents the Fourier transformed pressure signals, based on the pressure fluctuations shown in figure 3.4.9.
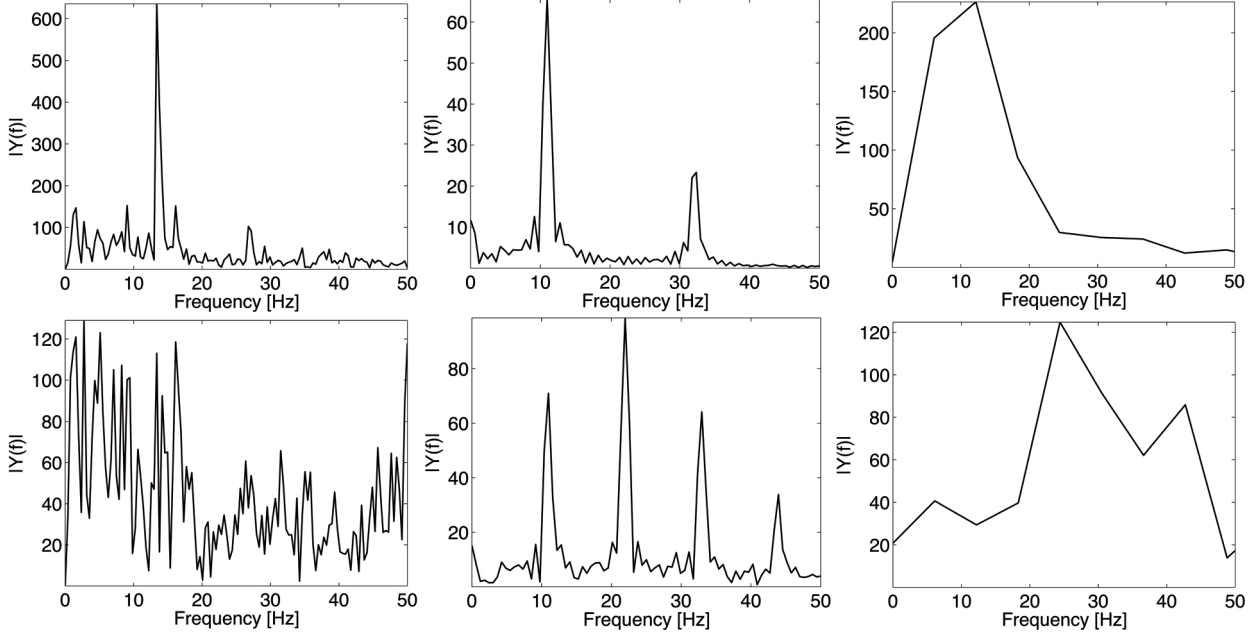

Figure 3.4.10: *Frequency spectrum of the pressure signal at positions P1 and P2. Upper row: position P1; lower row: position P2; From left to right: Experiment, k-ω SST and Spalart-Allmaras DDES.*

As to be seen in figure 3.4.10, the frequency spectra of the pressure signal at position *P1* contain a relatively similar dominant frequency. This frequency is assumed to represent the frequency of the helical vortex. In contrast to the frequency spectrum of the experiment at position *P1*, the experimental data does not show a clear dominant frequency at position *P2*. However, the simulations include dominant frequencies of the pressure signal at this position. It should be stressed, that the pressure signal of the simulation based on the *Spalart-Allmaras DDES* model does not provide a sufficient amount of data in order to give detailed frequency spectra. The corresponding frequency spectra are however, presented for the sake of completeness. Table 3.4.3 presents the dominant frequencies, which are obtained by the Fourier plot with a frequency range close to the dominant frequency.

Table 3.4.3: Dominant frequency of the Fourier transformed pressure signal at positions *P1* and *P2*.

	Experiment	k-ω SST	S-A DDES
Frequency at *P1* in [*Hz*]	13.4	11.0	12.2
Deviation from the experimental frequency in [%]	0	18	9
Frequency at *P2* in [*Hz*]	-	22.0	24.4

As table 3.4.3 shows, the dominant frequency of the experiment at position *P1* is 13.4 *Hz*. Here, the *k-ω SST* and *Spalart-Allmaras DDES* simulation predict a lower frequency of 11.0 *Hz* and 12.2 *Hz* respectively. These differences correspond to a deviation from the experimental frequency of 18% and 9%. At position *P2*, no dominant frequency can be determined for the experimental data. The simulations based on the *k-ω SST* and *Spalart-Allmaras DDES* model on the other hand, contain a dominant frequency of 22.0 *Hz* and 24.4 *Hz* respectively. A comparison of the mean pressure differences between position *P1* and *P2* is presented in table 3.4.4.

Table 3.4.4: Mean pressure difference between position *P1* and *P2*.

	Experiment	k-ϵ	k-ω SST	S-A DDES
Pressure difference in [Pa]	931	1129	360	484
Deviation from the experimental pressure difference in [%]	0	21	61	48

It is clear from table 3.4.4, that the experimental data corresponds to a mean pressure difference of 931 *Pa* between position *P1* and *P2*. Here, the simulations give a mean pressure difference of: 1129 *Pa* based on the k-ϵ, 360 *Pa* based on the *k-ω SST* and 484 *Pa* based on the *Spalart-Allmaras DDES* model. These values correspond to a deviation from the experimental mean pressure difference of: 21%, 61% and 48% respectively. A comparison of the velocity profiles along the radial measurement line *L1* is presented in figures 3.4.11.
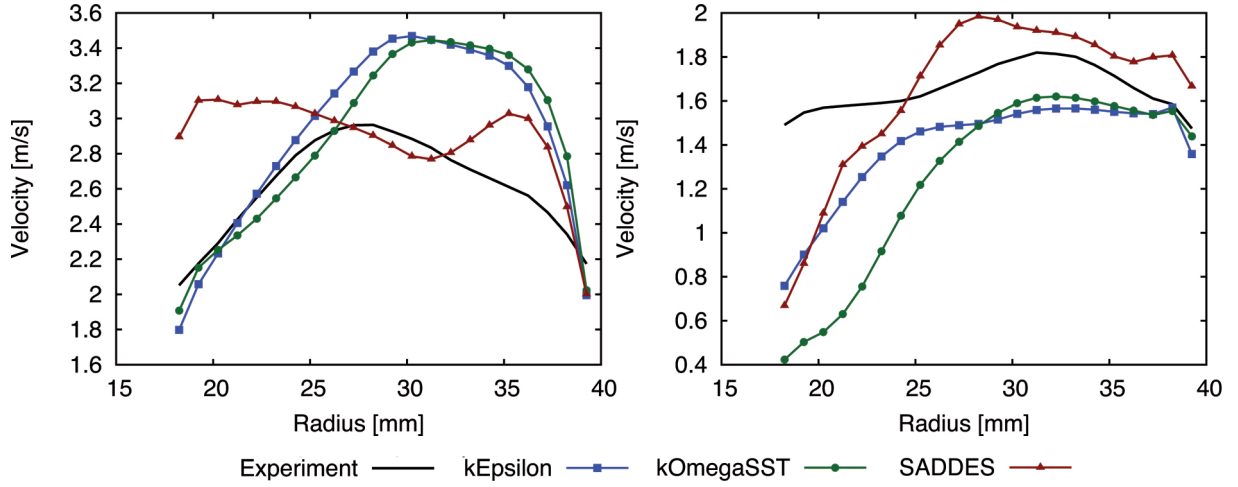

Figure 3.4.11: *Axial and tangential velocity along the measurement line "L1". Left: axial velocity; right: tangential velocity.*

As can be seen in figure 3.4.11, the axial and tangential mean velocity profiles along the measurement line *L1* include a relatively large mismatch between experimental and simulated data. Furthermore it can be seen, that the curve of the simulation based on the *Spalart-Allmaras DDES* model differs from the relatively similar curves of the RANS models. The velocity curves of the measurement line *L2* are depicted in figure 3.4.12.
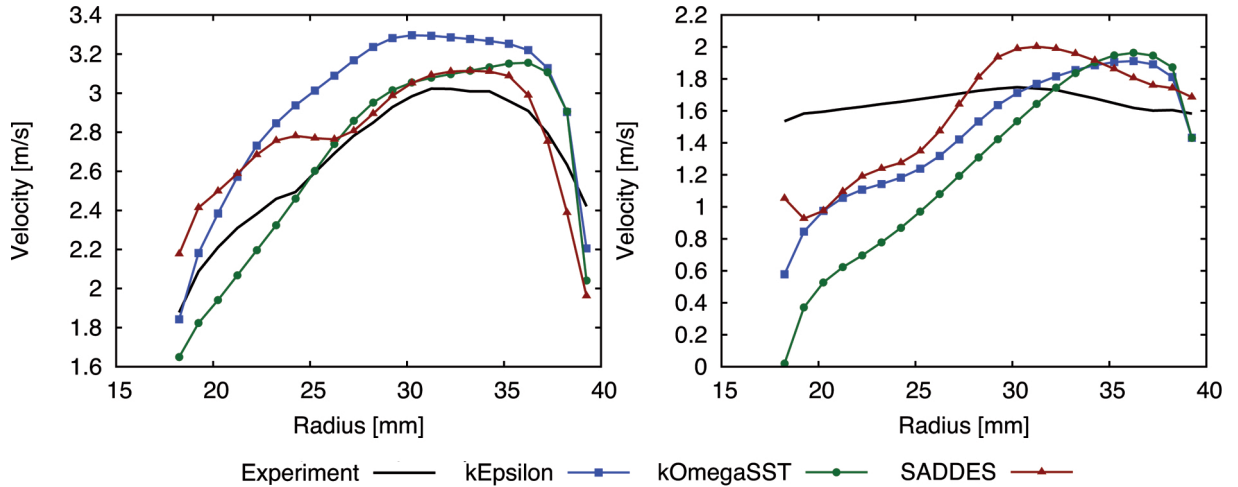

Figure 3.4.12: *Axial and tangential velocity along the measurement line "L2". Left: axial velocity; right: tangential velocity.*

44

Figure 3.4.12 presents the curves of the axial and tangential mean velocity along the measurement line *L2*. In the left picture of figure 3.4.12, it can be seen that the curves of the axial mean velocity show a relatively similar shape, even though the magnitudes of the velocities do not match exactly. A comparison of the tangential velocity on the other hand, includes relatively large differences between experimental and simulated data. Finally, a comparison of the velocity profiles along measurement line *L3* is shown in figure 3.4.13.
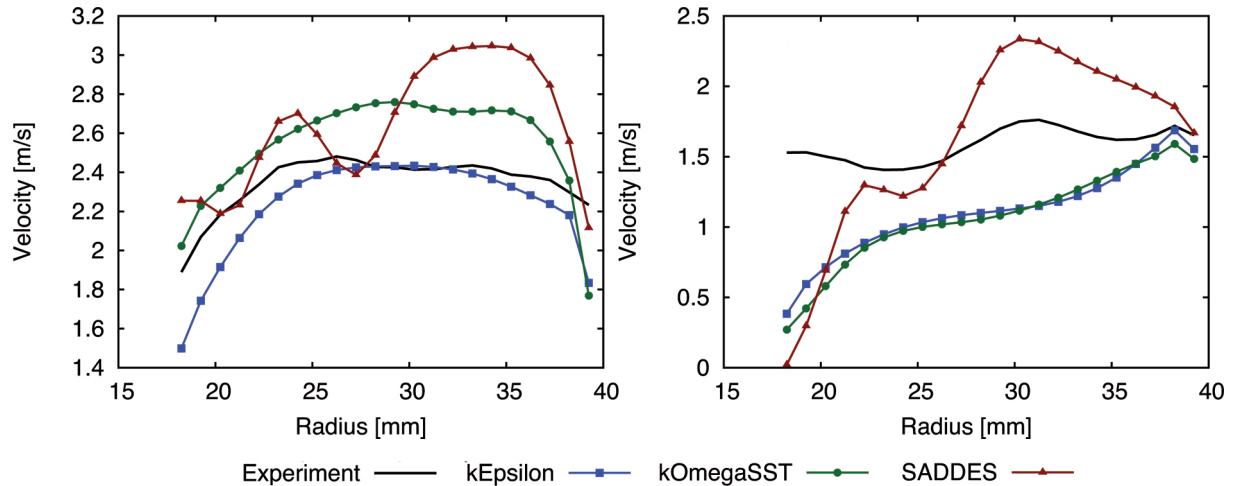

Figure 3.4.13: *Axial and tangential velocity along the measurement line "L3". Left: axial velocity; right: tangential velocity.*

As figure 3.4.13 shows, the mismatch of the velocity profiles between experiment and simulations applies as well for measurement line *L3*. Independent of the mismatch with the experimental data, similarities between the shape of the curves of the RANS models can be identified.

4 Conclusion and Prospects

In this work, cases for numerical investigations of the flow in the Stuttgart Swirl Generator have been set up with OpenFOAM. To do so, the geometry and mesh, provided by the Institute of Fluid Mechanics and Hydraulic Machinery at the University of Stuttgart, have been modified for the purpose of this work. Based on a simplified blade geometry and container length, two kinds of meshes have been generated: one for the application of high-Reynolds number RANS models, which do not integrate to the wall, and one for the application of hybrid RANS-LES models, which integrate to the wall. Here, the difference in the resolution of the near-wall region led to a mesh for the RANS models with approximately 2.85 million cells and a mesh for the hybrid models with approximately 6.5 million cells. The numerical setup of the cases consists mainly of second-order accurate discretisation schemes. For the discretisation of the convection-terms however, a Total Variation Diminishing scheme has been chosen in order to avoid unphysical oscillations in the solution. Furthermore, the PIMPLE algorithm has been applied for pressure-velocity coupling. This algorithm allowed performing simulations with relatively large timesteps in the transient phase of the simulations. The numerical setup including the solvers for the systems of linear algebraic equations and the specified boundary as well as initial conditions, generally enabled numerically stable simulations.

The focus of this work was the investigation of hybrid turbulence models in connection with the helical vortex phenomenon. For this, the flow with a discharge of 25 $\frac{m^3}{h}$ and 40 $\frac{m^3}{h}$ has been simulated with five different turbulence models, namely, k-ϵ, k-ω SST, k-ω SST SAS, $Spalart$-$Allmaras$ $DDES$ and $Spalart$-$Allmaras$ $IDDES$. In comparison with the RANS models and based on a maximum Courant number of 1, the hybrid models required approximately 16 to 35 times more time to simulate the flow in the Swirl Generator. This significant difference can firstly be addressed to the required timesteps, which differ due to the differences in the resolution of the near-wall region. A comparison of the computational effort, independent of the timestep further showed, that the hybrid models required approximately 3 to 4 times more time to calculate one timestep. Besides the higher number of cells, it was found that the simulations based on the hybrid models tend to require considerably more iterations to calculate the pressure for the corresponding timestep. An assessment of the results in terms of the different turbulence models showed, that all models except the k-ϵ model were able to simulate the striking flow features, i.e. the vortex rope and a region of reversed flow of the helical vortex phenomenon. The failure of the k-ϵ model is assumed to fall back on too high an amount of modeled turbulence, which does not allow a breakup of the helical vortex. It could be observed, that the helical section of the simulated vortex rope is determined by the conical diffuser of the flow system. Moreover, a comparison between the helical vorticies of both values of discharge showed relatively similar dimensions of the vortex ropes, which in turn would emphasize a geometry dependency of the helical vortex phenomenon. In terms of turbulence modeling, it was found that the hybrid models principally tend to give a more detailed flow with a higher amount of rotational flow structures. This tendency was observed for all hybrid models except the k-ω SST SAS model. While the simulation with a discharge of 25 $\frac{m^3}{h}$ showed a similar amount of resolved structures as the other hybrid models, the simulation with a discharge of 40 $\frac{m^3}{h}$ included a relatively high amount of modeled turbulence. Moreover, it has been shown that these differences between both turbulence model strategies in terms of modeled and resolved turbulence structures are in good agreement with a comparison of the turbulent kinematic viscosity. It was found, that the RANS models generally include a significantly higher amount of modeled turbulence. Here, the simulations based on the RANS models gave a highly diffusive flow containing regions with a turbulent viscosity greater than 1000 times the molecular viscosity. The turbulent viscosity of the hybrid models on the other hand was found to be less than 50 times the molecular viscosity in the majority of the regions of interest. Once again, this comparison revealed the ambiguous behavior of the k-ω SST SAS model in terms of the great variation of the amount of turbulence modeling. Although the simulations based on the hybrid models tended to give a more likely picture of the real flow, both turbulence modeling strategies failed to give results, which are in sufficient agreement with the experimental measurements. The comparison with the experimental data has due to the poor development of the other simulations been limited to the simulations based on the k-ϵ, k-ω SST and $Spalart$-$Allmaras$ $DDES$ model. A comparison of the pressure fluctuations showed that the simulations based on the k-ω SST model generally failed to give a similar development of the pressure signal. From the development of the pressure fluctuations of the simulations based on the $Spalart$-$Allmaras$ $DDES$ model on the other hand, no conclusions could be drawn due to a too short signal length. It was, however, clear from the comparison of the amplitudes of the fluctuations, that the simulations are tending to give pressure fluctuations with a significantly lower amplitude. The comparison of the frequency spectra at both pressure-sensor positions showed, that the k-ω SST and $Spalart$-$Allmaras$

DDES model generally tended to predict a lower dominant frequency in comparison with the experiments. While the deviation of the simulation based on the *k*-ω *SST* model from the experimental data was ranging from 18% to 25%, the *Spalart-Allmaras DDES* model deviated from the experimental frequency ranging from 1% to 30%. Furthermore, the simulations contained a dominant frequency at position *P2* with a discharge of 40 $\frac{m^3}{h}$, whereas no dominant frequency could be identified in the experimental frequency spectrum. The comparison of the mean pressure difference between both pressure-sensor positions on the other hand, did not allow the identification of any tendency of the turbulence models. The simulations deviated from the experiments ranging from: 21% to 85% based on the *k*-ϵ, 29% to 61% based on the *k*-ω *SST* and 11% to 48% based on the *Spalart-Allmaras DDES* model. While the simulations based on the *k*-ϵ model tended to predict a higher difference, the simulations based on the *k*-ω *SST* model tended to predict a lower mean pressure difference. The simulations based on the *Spalart-Allmaras DDES* model on the other hand, showed a higher difference for a discharge of 25 $\frac{m^3}{h}$ and a lower difference for a discharge of 40 $\frac{m^3}{h}$. In comparison, the velocity profiles showed a relatively similar behavior of the simulations for both values of discharge. However, only the comparison of the axial velocity along the radial measurement line *L2* was in relatively good agreement with the experimental data. All other velocity profiles showed relatively large differences between simulated and experimental data. Independent of the mismatch with the experimental data, similarities between the velocity profiles of the RANS models have been identified.

The results of this work firstly reveal a considerably higher computational effort of the hybrid strategies in contrast to the high-Reynolds number RANS models. Opposing this drawback of the hybrid models is the fact that simulations based on hybrid models generally predict a flow field with a higher amount of resolved flow structures. In terms of flow physics, this prediction of the flow is assumed to provide a more likely picture of the real flow and consequently to include a higher quality of the results. A comparison of the results of the simulations and the experimental data did not allow the validation of this conclusion regarding the quality of the results of the hybrid models. Both turbulence modeling strategies failed to deliver results with sufficient agreement of simulated and experimental data. Here, differences in the prediction of the separation below the blades have been identified to be a possible reason for the failure. The analysis of this region represents therefore, a first possibility for future investigations. Assuming that the measured velocity profiles are mainly influenced by the upstream, i.e. the blades of the Swirl Generator, simulations including only the blade region would enable the investigation of the origin of the mismatches between the experimental and simulated velocity profiles. These investigations would allow performing studies with less computational effort than for the simulations conducted in this work. Furthermore, the execution of experiments and simulations, which do not contain the corresponding separation zones, represents another possibility for future investigations. Based on the results of these experiments and simulations, it would be possible to prove the separation below the blades to be the origin of the mismatch. Should this be the case, these experiments and simulations would allow for a better assessment of the simulations with the help of experimental data. However, the execution of the corresponding experiments and simulations would probably require an adjustment of the design of the blades in the test rig. Finally, further studies regarding the cases performed in this work should be done in order to investigate the origin of occasional instabilities occurring in simulations based on the *k*-ω *SST SAS* and *Spalart-Allmaras IDDES* model and to exclude possible effects of the underlying geometry simplifications, meshes and numerical settings on the results of the simulations.

A Turbulence Models

This appendix depicts the original formulations of the turbulence models used in this work as well as differences of their implementation in OpenFOAM. In order to allow for comparability, the original notation of OpenFOAM has been replaced by its mathematical equivalent.

A.1 k-ϵ

The turbulent kinematic viscosity in the original formulation of the high-Re k-ϵ model [8][15] reads

$$\nu_t = C_\mu \frac{k^2}{\epsilon}, \tag{A.1.1}$$

whereas the turbulence kinetic energy k and its dissipation ϵ are calculated with

$$\frac{\partial k}{\partial t} + \text{div}(k\mathbf{u}) = \underbrace{\text{div}\left[\left(\nu + \frac{\nu_t}{\sigma_k}\right)\text{grad}\,k\right]}_{\text{Diffusion}} + \underbrace{P_k}_{\text{Production}} - \underbrace{\epsilon}_{\text{Destruction}}, \tag{A.1.2}$$

$$\frac{\partial \epsilon}{\partial t} + \text{div}(\epsilon\mathbf{u}) = \underbrace{\text{div}\left[\left(\nu + \frac{\nu_t}{\sigma_\epsilon}\right)\text{grad}\,\epsilon\right]}_{\text{Diffusion}} + \underbrace{C_{\epsilon 1}\frac{\epsilon}{k}P_k}_{\text{Production}} - \underbrace{C_{\epsilon 2}\frac{\epsilon^2}{k}}_{\text{Destruction}} \tag{A.1.3}$$

with

$$P_k = 2\nu_t S_{ij}^2 \tag{A.1.4}$$

and the constants shown in table A.1.1.

Table A.1.1: Parameters and constants of the high Re k-ϵ model.

C_μ	$C_{\epsilon 1}$	$C_{\epsilon 2}$	σ_k	σ_ϵ
0.09	1.44	1.92	1	1.3

The OpenFOAM implementation reads

$$\frac{\partial k}{\partial t} + \text{div}(k\mathbf{u}) - \text{div}(\mathbf{u})*k = \underbrace{\text{div}\left[\left(\nu + \frac{\nu_t}{\sigma_k}\right)\text{grad}\,k\right]}_{\text{Diffusion}} + \underbrace{G}_{\text{Production}} - \underbrace{\epsilon}_{\text{Destruction}}, \tag{A.1.5}$$

$$\frac{\partial \epsilon}{\partial t} + \text{div}(\epsilon\mathbf{u}) - \text{div}(\mathbf{u})*\epsilon = \underbrace{\text{div}\left[\left(\nu + \frac{\nu_t}{\sigma_\epsilon}\right)\text{grad}\,\epsilon\right]}_{\text{Diffusion}} + \underbrace{C_1\frac{\epsilon}{k}G}_{\text{Production}} - \underbrace{C_2\frac{\epsilon^2}{k}}_{\text{Destruction}}, \tag{A.1.6}$$

with

$$G = 2\nu_t|S_{ij}|^2, \tag{A.1.7}$$
$$C_1 = C_{\epsilon 1}, \tag{A.1.8}$$
$$C_2 = C_{\epsilon 2}. \tag{A.1.9}$$

The third term on the l.h.s. of equations A.1.5 and A.1.6 is a part of the material derivative of the corresponding property and is used to improve the convergence rate of the solution process. In the case of an incompressible formulation it becomes equal to zero when convergence is achieved [9].

48

A.2   k-ω SST

In the original formulation [9][15] of the high-Re k-ω SST model, the turbulent kinematic viscosity reads

$$\nu_t = \frac{A_1 k}{max(A_1 \omega, F_2 S)}, \tag{A.2.1}$$

whereas the turbulence kinetic energy k and frequency ϵ are calculated with

$$\frac{\partial k}{\partial t} + \text{div}(k\mathbf{u}) = \underbrace{\text{div}\left[\left(\nu + \frac{\nu_t}{\sigma_k}\right) \text{grad } k\right]}_{\text{Diffusion}} + \underbrace{min(P_k, C_{\epsilon 1}\epsilon)}_{\text{Production}} - \underbrace{\beta^* k\omega}_{\text{Destruction}}, \tag{A.2.2}$$

$$\frac{\partial \omega}{\partial t} + \text{div}(\omega\mathbf{u}) = \underbrace{\text{div}\left[\left(\nu + \frac{\nu_t}{\sigma_\omega}\right) \text{grad } \omega\right]}_{\text{Diffusion}} + \underbrace{\frac{\gamma}{\nu_t} P_k}_{\text{Production}} - \underbrace{\beta\omega^2}_{\text{Destruction}} + \underbrace{2(1 - F_1)\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}}_{\text{Cross-diffusion}} \tag{A.2.3}$$

with

$$\epsilon = \beta^* k\omega, \tag{A.2.4}$$

$$P_k = \tau_{ij}\frac{\partial U_i}{\partial x_j}, \tag{A.2.5}$$

$$F_1 = tanh\left\{\left\{min\left[max\left(\frac{\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\right), \frac{4\rho\sigma_{\omega 2}k}{CD_{k\omega}y^2}\right]\right\}^4\right\}, \tag{A.2.6}$$

$$F_2 = tanh\left\{\left[max\left(\frac{2\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\right)\right]^2\right\}, \tag{A.2.7}$$

$$CD_{k\omega} = max\left(2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, 10^{-10}\right) \tag{A.2.8}$$

and the constants are presented in table A.2.1.

Table A.2.1: Parameters and constants of the k-ω SST turbulence model [9].

A_1	$C_{\epsilon 1}$	β_1	β_2	β^*	γ_1	γ_2	σ_{k1}	σ_{k2}	$\sigma_{\omega 1}$	$\sigma_{\omega 2}$	κ
10	10	0.075	0.0828	0.09	0.5532	0.4403	1.176	1	2	1.168	0.41

The model coefficients of the k-ϵ and k-ω mode of the k-ω SST model are calculated with

$$\phi = F\phi_1 + (1 - F)\phi_2, \tag{A.2.9}$$

whereas ϕ refers to the corresponding model constant. The OpenFOAM implementation reads

$$\nu_t = \frac{A_1 k}{max(A_1 \omega, F_2\sqrt{2S_2})} \tag{A.2.10}$$

with

$$\frac{\partial k}{\partial t} + \text{div}(k\mathbf{u}) - \text{div}(\mathbf{u}) * k = \underbrace{\text{div}\left[(\nu + \alpha_k \nu_t) \text{grad } k\right]}_{\text{Diffusion}} + \underbrace{min(G, C_1\beta^* k\omega)}_{\text{Production}} - \underbrace{\beta^* k\omega}_{\text{Destruction}}, \tag{A.2.11}$$

$$\frac{\partial \omega}{\partial t} + \text{div}(\omega\mathbf{u}) - \text{div}(\mathbf{u}) * \omega = \underbrace{\text{div}\left[(\nu + \alpha_\omega \nu_t) \text{grad } \omega\right]}_{\text{Diffusion}} + \underbrace{2S_2\gamma}_{\text{Production}} - \underbrace{\beta\omega^2}_{\text{Destruction}} + \underbrace{2(F_1 - 1)\alpha_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}}_{\text{Cross-diffusion}}, \tag{A.2.12}$$

and

$$G = 2S_2\nu_t, \tag{A.2.13}$$

$$S_2 = |S_{ij}|^2, \tag{A.2.14}$$

$$F_1 = tanh\left\{\left\{min\left\{min\left[max\left(\frac{\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\right), \frac{4\alpha_{\omega 2}k}{max(CD_{k\omega}, 10^{-10})y^2}\right]\right\}, 10\right\}^4\right\}, \tag{A.2.15}$$

$$F_2 = tanh\left\{\left\{min\left[max\left(\frac{2\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\right)\right], 100\right\}^2\right\}, \tag{A.2.16}$$

$$CD_{k\omega} = 2\alpha_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial\omega}{\partial x_j}. \tag{A.2.17}$$

The third term on the l.h.s. of equations A.2.11 and A.2.12 is a part of the material derivative of the corresponding property and is used to improve the convergence rate of the solution process. In the case of an incompressible formulation it becomes equal to zero when convergence is achieved [9].

Table A.2.2 shows the constants in OpenFOAM, whereas α represents the reciprocal of σ in the original formulation.

Table A.2.2: Parameters and constants of the k-ω SST turbulence model.

A_1	α_{k1}	α_{k2}	$\alpha_{\omega 1}$	$\alpha_{\omega 2}$	C_1	κ
0.31	0.85034	1	0.5	0.85616	10	0.41

The model coefficients of the k-ϵ and k-ω mode of the k-ω SST model are calculated with

$$\phi = F_1\phi_1 + (1 - F_1)\phi_2, \tag{A.2.18}$$

whereas ϕ refers to the corresponding model constant.

A.3 k-ω SST SAS

The k-ω SST SAS model is a hybrid RANS-LES version of the k-ω SST model. While the k-equation is analogous to the equation in the k-ω SST model in appendix A.2. The ω-equation in the SAS model [17] is modified and reads

$$\frac{\partial\omega}{\partial t} + \text{div}(\omega\mathbf{u}) = \underbrace{\text{div}\left(\frac{\nu_t}{\sigma_\phi}\text{grad}\,\omega\right)}_{\text{Diffusion}} + \underbrace{\frac{\omega}{\rho k}P_k\left[1 - \zeta_1 + \zeta_2\left(\frac{L}{L_{vK}}\right)^2\right]}_{\text{Production}} - \underbrace{\omega^2\left(C_\mu - C_\mu^{\frac{1}{4}}\zeta_3\right)}_{\text{Destruction}}$$

$$+ \underbrace{\frac{2}{\sigma_\phi}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial\omega}{\partial x_j} - \frac{k}{\omega^2}\frac{\partial\omega}{\partial x_j}\frac{\partial\omega}{\partial x_j}}_{\text{Cross-diffusion}} + Q_{SAS}, \tag{A.3.1}$$

with the additional term

$$Q_{SAS} = max\left[\zeta_2\kappa S^2\left(\frac{L}{L_{vK}}\right)^2 - C_c\frac{2}{\sigma_\phi}k * max\left(\frac{1}{\omega^2}\frac{\partial\omega}{\partial x_j}\frac{\partial\omega}{\partial x_j}, \frac{1}{k^2}\frac{\partial k}{\partial x_j}\frac{\partial\omega}{\partial x_j}\right), 0\right]. \tag{A.3.2}$$

Here,

$$L = x, \tag{A.3.3}$$

$$L_{vK} = \kappa\left|\frac{\sqrt{2 * S_{ij}^2}}{\sqrt{\frac{\partial^2 U_i}{\partial x_k^2}\frac{\partial^2 U_i}{\partial x_j^2}}}\right|. \tag{A.3.4}$$

Table A.3.1: Parameters and constants of the k-ω SST SAS turbulence model.

C_c	C_μ	σ_k	σ_ϕ	ζ_1	ζ_2	ζ_3	κ
2	0.09	$\frac{2}{3}$	$\frac{2}{3}$	0.8	1.47	0.0288	0.41

The constants are shown in table A.3.1.
In OpenFOAM, the Sub-Grid-Scale viscosity of the k-ω SST SAS reads

$$\nu_{SGS} = \frac{A_1 k}{max(A_1\omega, F_2\sqrt{S_2})}, \tag{A.3.5}$$

whereas the modified ω-equation is implemented as

$$\frac{\partial \omega}{\partial t} + \text{div}(\omega\mathbf{u}) - \text{div}(\mathbf{u}) * \omega = \underbrace{\frac{\partial}{\partial x_j}\left[(\nu + \alpha_\omega\nu_t)\frac{\partial\omega}{\partial x_j}\right]}_{\text{Diffusion}} + \underbrace{2S_2\gamma}_{\text{Production}} - \underbrace{\beta\omega^2}_{\text{Destruction}}$$
$$+ \underbrace{2(F_1-1)\alpha_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial\omega}{\partial x_j}}_{\text{Cross-diffusion}} + Q_{SAS} \tag{A.3.6}$$

with the additional term

$$Q_{SAS} = F_{SAS} * max\left[S_2 * \tilde{\zeta}_2 * \kappa\left(\frac{L}{L_{vk2}}\right) - \frac{2}{\alpha_\phi}k * max\left(\frac{\left|\frac{\partial k}{\partial x_j}\right|^2}{(k+k_0)^2}, \frac{\left|\frac{\partial\omega}{\partial x_j}\right|^2}{(\omega+\omega_{small})^2}\right), 0\right]. \tag{A.3.7}$$

Here,

$$S_2 = |S_{ij}|^2, \tag{A.3.8}$$

$$L = \frac{\sqrt{k}}{\left[C_\mu^{\frac{1}{4}} * (\omega + \omega_{small})\right]}, \tag{A.3.9}$$

$$L_{vk2} = max\left(\frac{\kappa\sqrt{S_2}}{|\text{div}(\text{grad}\,\mathbf{u})| + 1 * 10^{-150}}, C_S * \Delta\right), \tag{A.3.10}$$

$$CD_{k\omega} = 2\alpha_{\omega 2}\frac{1}{\omega + \omega_{small}}\frac{\partial k}{\partial x_j}\frac{\partial\omega}{\partial x_j}. \tag{A.3.11}$$

The constants in OpenFOAM are presented in table A.3.2.

Table A.3.2: Parameters and constants of the k-ω SST SAS turbulence model in OpenFOAM.

α_ϕ	C_S	F_{SAS}	$\tilde{\zeta}_2$
0.666667	0.262	1.25	1.755

A.4   Spalart-Allmaras DES Formulation and Its Derivations

In the original Spalart-Allmaras RANS model [16], the turbulence length scale is defined as

$$\tilde{d} = min(y_w, C_{DES}\Delta). \tag{A.4.1}$$

The turbulent kinematic eddy viscosity is calculated with

$$\nu_t = f_{v1}\tilde{\nu}, \tag{A.4.2}$$

$$f_{v1} = \frac{\chi^3}{\chi^3 + C_{v1}^3}, \tag{A.4.3}$$

$$\chi = \frac{\tilde{\nu}}{\nu}. \tag{A.4.4}$$

The transport equation for the modified kinematic eddy viscosity $\tilde{\nu}$ reads

$$\frac{\partial \tilde{\nu}}{\partial t} + \operatorname{div}(\tilde{\nu}\mathbf{u}) = \underbrace{\frac{1}{C_\sigma}\left\{ \operatorname{div}\left[(\nu + \tilde{\nu})\operatorname{grad}\tilde{\nu}\right] + C_{b2}\frac{\partial \tilde{\nu}}{\partial x_j}\frac{\partial \tilde{\nu}}{\partial x_j}\right\}}_{\text{Diffusion}} + \underbrace{C_{b1}\tilde{S}\tilde{\nu}}_{\text{Production}} \underbrace{- C_{w1}f_w\left(\frac{\tilde{\nu}}{\tilde{d}}\right)^2}_{\text{Destruction}} \tag{A.4.5}$$

with

$$\tilde{S} = \omega + f_{v2}\frac{\tilde{\nu}}{(\kappa\tilde{d})^2}, \tag{A.4.6}$$

$$f_{v2} = 1 - \frac{\chi}{1 + f_{v1}\chi}, \tag{A.4.7}$$

$$f_w = g * \left(\frac{1 + C_{w3}^6}{g^6 + C_{w3}^6}\right)^{1/6}, \tag{A.4.8}$$

$$g = r + C_{w2}(r^6 - r), \tag{A.4.9}$$

$$r = \frac{\tilde{\nu}}{\tilde{S}(\kappa\tilde{d})^2}. \tag{A.4.10}$$

and the constants are shown in table A.4.1.

Table A.4.1: Constants of the original Spalart-Allmaras DES turbulence model [16].

C_σ	C_{b1}	C_{b2}	C_{v1}	C_{w1}	C_{w2}	C_{w3}	κ
$\frac{2}{3}$	0.1355	0.622	7.1	$\frac{C_{b1}}{\kappa^2} + \frac{1+C_{b2}}{C_\sigma}$	0.3	2	0.41

In OpenFOAM, the implementation of the Spalart-Allmaras DES model reads

$$\frac{\partial \tilde{\nu}}{\partial t} + \operatorname{div}(\tilde{\nu}\mathbf{u}) = \underbrace{\frac{1}{\sigma_{\nu_t}}\left\{ \operatorname{div}\left[(\nu + \tilde{\nu})\operatorname{grad}\tilde{\nu}\right] + C_{b2}\left|\frac{\partial \tilde{\nu}}{\partial x_j}\right|^2\right\}}_{\text{Diffusion}} + \underbrace{C_{b1}\tilde{S}\tilde{\nu}}_{\text{Production}} \underbrace{- C_{w1}f_w\left(\frac{\tilde{\nu}}{\tilde{d}}\right)^2}_{\text{Destruction}} \tag{A.4.11}$$

$$\tilde{S} = f_{v3}S + f_{v2}\frac{\tilde{\nu}}{(\kappa\tilde{d})^2}, \tag{A.4.12}$$

$$S = \sqrt{2} * |\Omega_{ij}|, \tag{A.4.13}$$

$$f_{v3} = \frac{(1 + f_{v1}\chi) * \left(\frac{1}{C_{v2}}\right) * \left[3 * \left(1 + \frac{1}{C_{v2}}\chi\right) + \left(\frac{1}{C_{v2}}\chi\right)^2\right]}{\left(1 + \frac{1}{C_{v2}}\chi\right)^3}, \tag{A.4.14}$$

$$f_{v2} = \frac{1}{\left(1 + \frac{\tilde{\nu}}{C_{v2}\nu}\right)^3}, \tag{A.4.15}$$

$$r = min\left(\frac{\tilde{\nu}}{max(S, 1 * 10^{-300})(\kappa\tilde{d})^2 + 1 * 10^{-150}}, 10\right) \tag{A.4.16}$$

with σ_{ν_t} equal to 0.66666 and the additional constant C_{v2} equal to 5.

A.4.1 DDES

The modifications of the DDES formulation are introduced to the Spalart-Allmaras DES formulation presented in appendix A.4 with the definition of a new length scale [17], which reads

$$l_{DES} = d_w - f_d * max(d_w - C_{DES}\Delta, 0), \tag{A.4.17}$$

$$f_d = 1 - tanh\left[(8 * r_d)^3\right], \tag{A.4.18}$$

$$r_d = \frac{\nu_t + \nu}{\sqrt{U_{i,j}U_{i,j}}(\kappa d_w)^2}. \tag{A.4.19}$$

The OpenFOAM implementation of the new length scale reads

$$\tilde{d} = max\left[y - f_d * max(y - C_{DES}\Delta, 0), 1 * 10^{-300}\right], \tag{A.4.20}$$

$$r_d = min\left(\frac{\nu_{SGS} + \nu}{max(S, 1 * 10^{-300}) * (\kappa y)^2 + 1 * 10^{-150}}, 10\right) \tag{A.4.21}$$

with

$$\tilde{d} = l_{DES}, \tag{A.4.22}$$

$$y = d_w, \tag{A.4.23}$$

$$\nu_{SGS} = \nu_t. \tag{A.4.24}$$

A.4.2 IDDES

Analogous to the DDES formulation, the modifications of the IDDES formulation are introduced to the original Spalart-Allmaras DES formulation with the definition of a new length scale [17][12], which reads

$$l_{hyb} = \tilde{f}_d * (1 + f_e) * l_{RANS} + (1 - \tilde{f}_d) * l_{LES} \tag{A.4.25}$$

with

$$l_{RANS} = d_w, \tag{A.4.26}$$

$$l_{LES} = C_{DES} * \psi * \Delta, \tag{A.4.27}$$

$$\Delta = min\{max[C_w d_w, C_w \Delta_{max}, \Delta_{wn}], \Delta_{max}\}, \tag{A.4.28}$$

$$\tilde{f}_d = max[(1-f_{dt}), f_B], \tag{A.4.29}$$

$$f_{dt} = 1 - tanh\left[(8r_{dt})^3\right], \tag{A.4.30}$$

$$f_B = min\left[2exp(-9\alpha^2), 1\right], \tag{A.4.31}$$

$$\alpha = 0.25 - \frac{d_w}{h_{max}}, \tag{A.4.32}$$

$$f_e = max[(f_{e1}-1), 0] * \psi * f_{e2}, \tag{A.4.33}$$

$$f_{e1} = \begin{cases} 2exp(-11.09\alpha^2) & \text{if } \alpha \geq 0 \\ 2exp(-9\alpha^2) & \text{if } \alpha < 0 \end{cases}, \tag{A.4.34}$$

$$f_{e2} = 1 - max(f_t, f_l), \tag{A.4.35}$$

$$f_t = tanh\left[\left(C_t^2 r_{dt}\right)^3\right], \tag{A.4.36}$$

$$f_l = tanh\left[\left(C_l^2 r_{dl}\right)^{10}\right], \tag{A.4.37}$$

$$r_{dt} = \frac{\nu_t}{(\kappa d_w)^2 * max\left\{\left[\sum_{ij}\left(\frac{\partial u_i}{\partial x_j}\right)^2\right]^{\frac{1}{2}}, 10^{-10}\right\}}, \tag{A.4.38}$$

$$r_{dl} = \frac{\nu}{(\kappa d_w)^2 * max\left\{\left[\sum_{ij}\left(\frac{\partial u_i}{\partial x_j}\right)^2\right]^{\frac{1}{2}}, 10^{-10}\right\}}, \tag{A.4.39}$$

$$\psi^2 = min\left\{10^2, \frac{1 - \frac{C_{b1}}{C_{w1}\kappa^2 f_w^*}[f_{t2} + (1-f_{t2})f_{v2}]}{f_{v1}max(10^{-10}, 1-f_{t2})}\right\}. \tag{A.4.40}$$

whereas d_w represents the distance to the wall. As can be seen, the IDDES formulation includes a new definition of the Sub-Grid length scale, whereas Δ_{wn} is the grid step in the wall normal direction. The constants of the original Spalart-Allmaras IDDES model are shown in table A.4.2.

Table A.4.2: Constants of the original IDDES turbulence model [17][12].

C_l	C_t	C_w	f_w^*
3.55	1.63	0.15	0.424

The OpenFOAM implementation reads

$$\underbrace{\tilde{d}}_{l_{hyb}} = max\left[\underbrace{f_{Hyb}}_{\tilde{f}_d}*(1+\underbrace{f_{Restore}*\psi}_{f_e})*\underbrace{y}_{l_{RANS}}+(1-f_{Hyb})*\underbrace{C_{DES}*\psi*\Delta}_{l_{LES}}, 1^*10^{-300}\right] \tag{A.4.41}$$

with

$$\Delta = \Delta_{Coeff} * min\left\{max\left[max\left(C_w y, C_w \Delta_{max}\right), h_{wn}\right], \Delta_{max}\right\}, \tag{A.4.42}$$

$$f_{Hyb} = max[(1 - \underbrace{f_d}_{f_{dt}}), \underbrace{f_{Step}}_{f_B}], \tag{A.4.43}$$

$$f_d = 1 - tanh\left[(8r_d)^3\right], \tag{A.4.44}$$

$$\alpha = max(0.25 - \frac{y}{h_{max}}, -5), \tag{A.4.45}$$

$$r_{dt} = r_d(\nu_{SGS}), \tag{A.4.46}$$

$$r_{dl} = r_d(\nu), \tag{A.4.47}$$

$$\psi^2 = min\left\{\frac{1 - \frac{C_{b1}}{C_{w1}\kappa^2 f_w^*}f_{v2}}{max(f_{v1}, 1 * 10^{-300})}, 10^2\right\}, \tag{A.4.48}$$

$$f_{Hill} = f_{e1}, \tag{A.4.49}$$

$$f_{Amp} = f_{e2}. \tag{A.4.50}$$

Here, Δ_{Coeff} and C_w have to be defined in the *LESproperties*-dictionary and h_{wn} is initialized as the wall distance. The parameter h_{max} in the definition of α corresponds to the maximum component of Δ. To obtain the Δ of the original formulation, the value of Δ_{Coeff} has been set to 1 in this work.

The definition of r_d is taken from the implementation of DDES in OpenFOAM, whereas $(\nu_{SGS} + \nu)$ in the numerator of r_d are substituted by ν_{SGS} and ν respectively.

B   Experimental Measurements

The experimental measurements used in this work, have been performed by the Institute of Fluid Mechanics and Hydraulic Machinery, University of Stuttgart. They consist of velocity as well as pressure measurements at certain positions in the Stuttgart Swirl Generator. The measurement of the velocity has been done with the method of Laser Doppler Velocimetry and includes the axial and tangential mean velocity along three lines downstream of the blades. For the pressure, two piezo-resistive transducers have been used to measure the pressure fluctuations in two points at the wall of the Swirl Generator. Figure B.0.1 gives an overview of the positions of the experimental measurements.
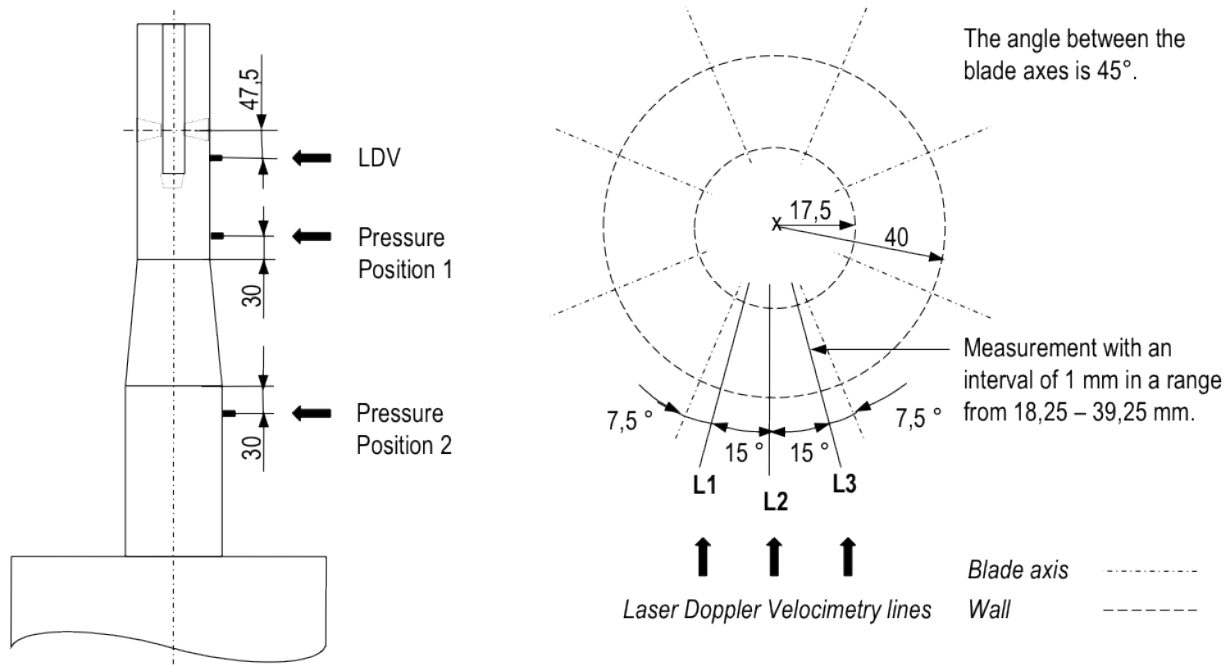
Figure B.0.1: *Schematic figures of the positions of the experimental measurements in [mm]. Left: vertical positions; right: horizontal positions of the Laser Doppler Velocimetry.*

As can be seen in the left sketch of figure B.0.1, the Laser Doppler Velocimetry (LDV) has been carried out in a vertical position of 47.5 mm downstream of the rotation axes of the blades. The right sketch of figure B.0.1 shows the horizontal position of the LDV. It can be seen, that the velocity has been measured at different radial positions along three radial lines, *L1*, *L2* and *L3* respectively. These lines are placed with an angle of 7.5° from the blade axis and an angle of 15° from each other. For each line, the velocity has been measured with an interval of 1 mm in a range from 18.25 - 39.25 mm. Furthermore, the left sketch of figure B.0.1 shows the vertical positions of the pressure measurements at the wall of the pipe. While 'Position 1' is located 30 mm before the draft tube, 'Position 2' is placed 30 mm after the draft tube. The reproducibility of the measured results is in a range of ±3.71% [5].

C   Figures of the Mesh

This appendix gives the reader an overview of the blocking, which has been used for the mesh generation and the cell distribution of the meshes for RANS and hybrid models in certain regions of the computational domain. The figures of the cell distribution include the horizontal as well as vertical cross-section, and the distribution around the blades. Figure C.0.1 shows the front- and side-view of the blocking.
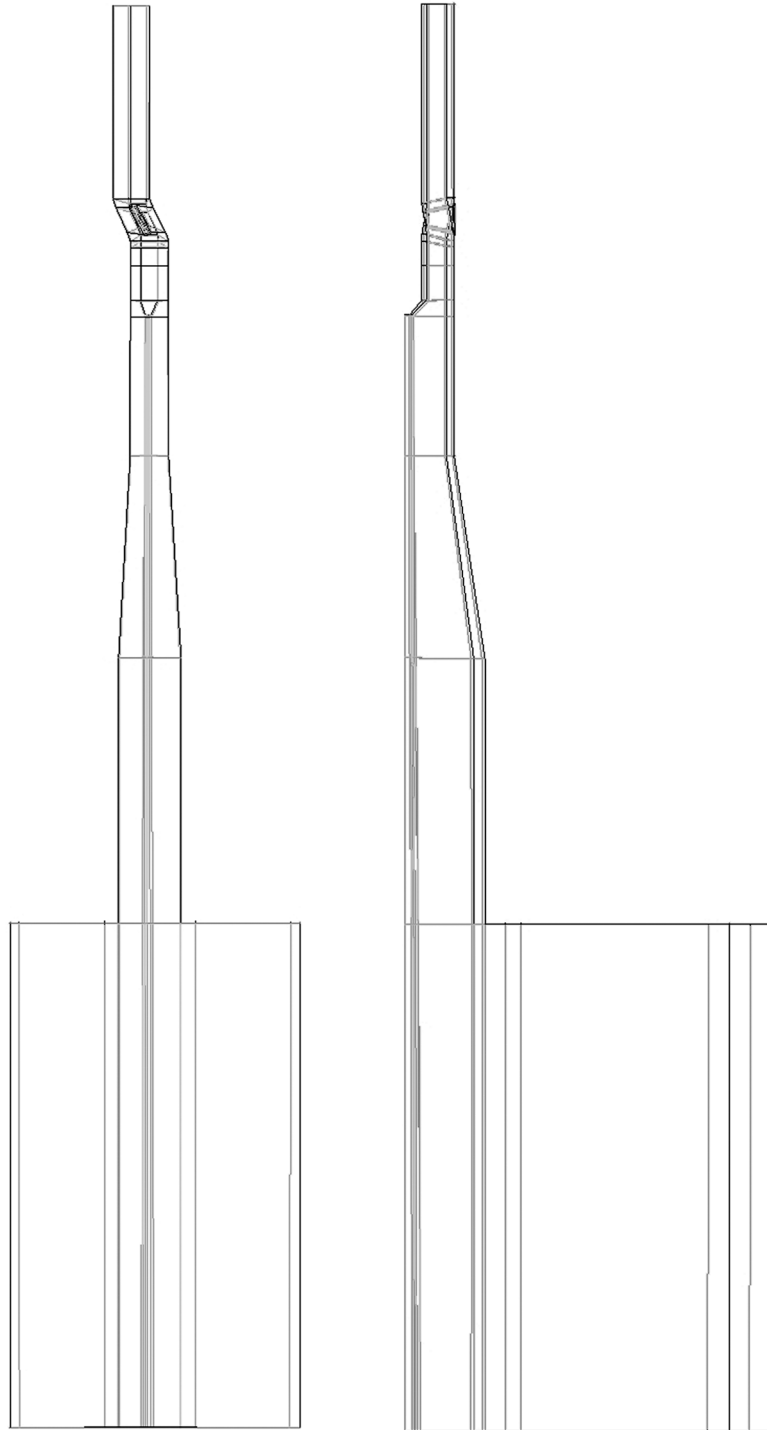

Figure C.0.1: *Blocking for the mesh generation. Left: front-view; right: side-view.*

In the left picture of figure C.0.1 it can be seen, that the blocking lines follow not only the shape of the pipes

of the geometry but also the angle of the blade. Moreover, the right picture of figure C.0.1 shows that some blocks of the entire domain have been spared out in order to reproduce the hub and the container. Figure C.0.2 presents the top-view of the blocking.
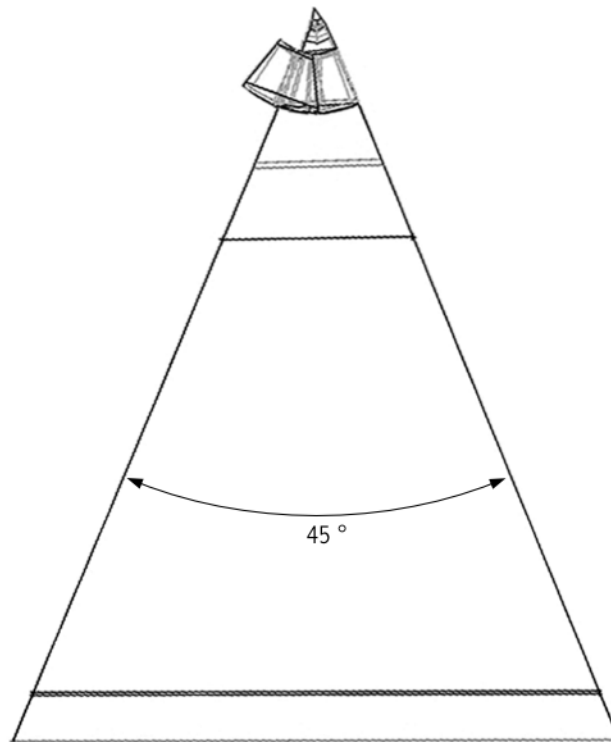

Figure C.0.2: *Blocking for the mesh generation, top-view.*

Figure C.0.2 shows that the blocking corresponds to a section with an angle of 45°. It should be mentioned, that the blocking lines, which approximate the rounding of the walls do not necessarily have to follow the shape of the wall. In ANSYS ICEM, the association of the blocking line with the corresponding geometry line automatically approximates the mesh to the shape of the geometry. Furthermore, it can be seen, that the blocking reaches from the outer boundary to the center line of the geometry. Blocking lines inside of the surrounding lines, represent splits, which have been created to individually choose the cell distribution of certain regions of the computational domain. Based on this blocking, a section of the mesh with an angle of 45° has been generated. The resulting mesh finally has been copied 7 times around the center line. Figure C.0.3 gives an overview of the cell distribution of the final mesh, visualized by a vertical cross-section.
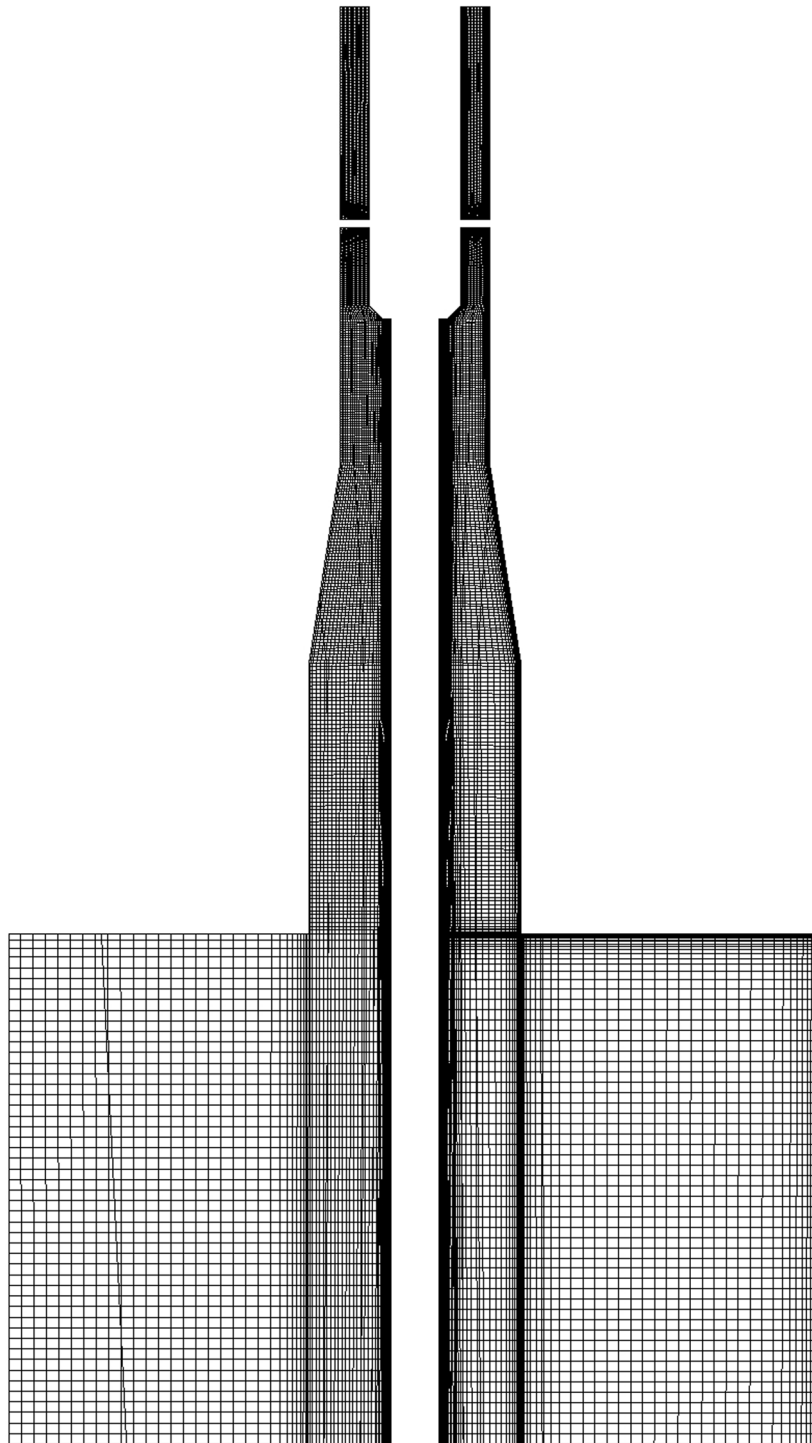
Figure C.0.3: *Cell distribution of the meshes for RANS and hybrid models, visualized by the x-plane cross-section. Left: mesh for RANS models; right: mesh for hybrid models.*

Figure C.0.3 shows the cell distribution and consequently the mesh density of both meshes. Here, dark regions correspond to a high mesh density. As can be seen, the mesh density of both meshes is quite high in the regions of the blade, draft tube as well as the following pipe and decreases noticeably in the container. Furthermore, it can be seen that the mesh for hybrid models contains a higher density close to the walls, in comparison with the RANS mesh. Figure C.0.4 presents a zoom of the x-plane cross-section, including the blades and the draft
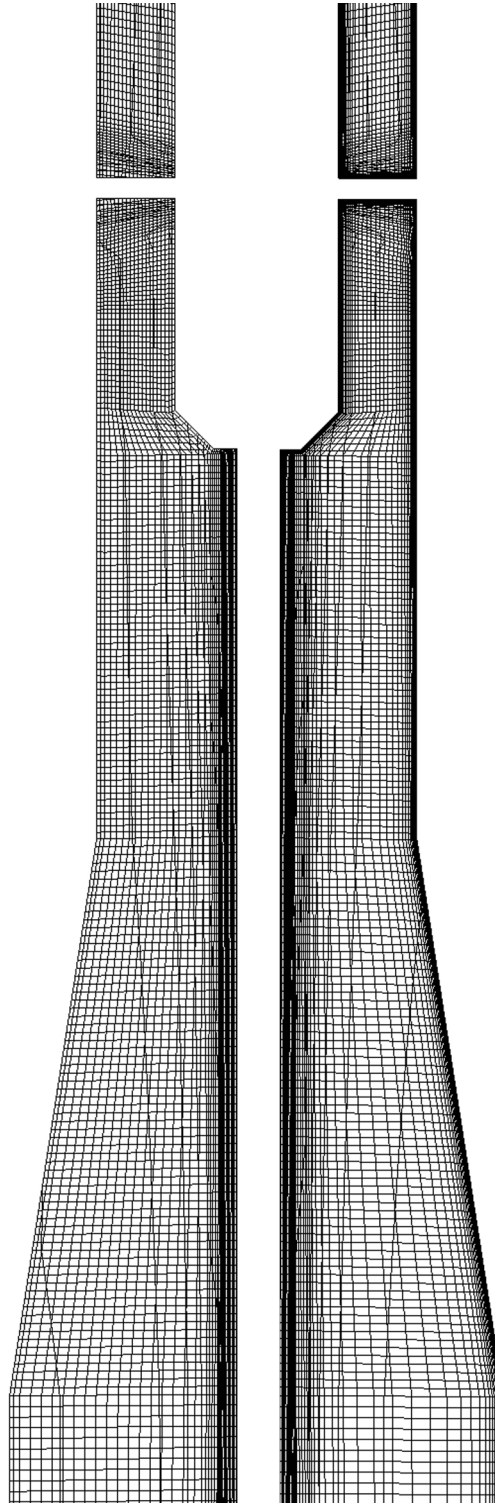
tube.

Figure C.0.4: *Cell distribution of the meshes for RANS and hybrid models, visualized by a zoom of the x-plane cross-section. Left: mesh for RANS models; right: mesh for hybrid models.*

A look at figure C.0.4 further clarifies the differences of the mesh for RANS and hybrid models in terms of the cell refinement towards the walls of the geometry. Figure C.0.5 gives an overview of the cell distribution in the horizontal cross-section.
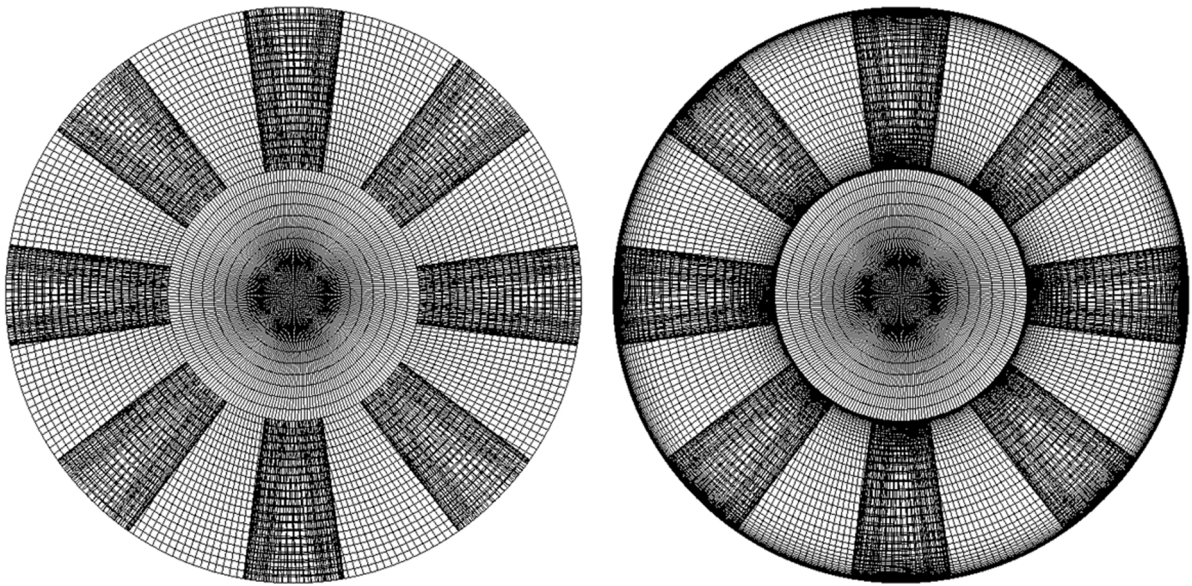
Figure C.0.5: *Cell distribution of the inlet plane of the meshes for RANS and hybrid models. Left: mesh for RANS models; right: mesh for hybrid models. View: in flow direction; with visible blades.*

In figure C.0.5, the cell distribution of the inlet plane can be seen. Furthermore, the cells of the blades and the bottom of the hub are visible. As can be seen, the cells approximate the rounding of the wall without causing small cell angles. As in figures C.0.3 and figure C.0.4, both meshes differ in their mesh resolution towards the wall. Figure C.0.6 reveals a closer look at the horizontal cell distribution in the center line, which is similar in both meshes.
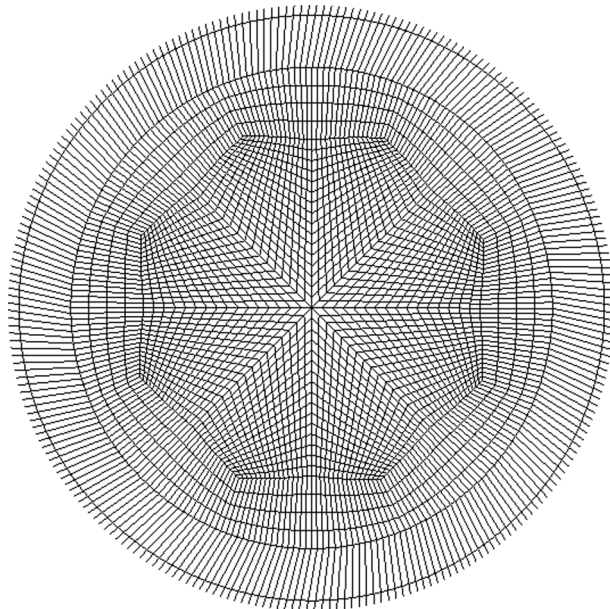

Figure C.0.6: *Cell distribution of the center line. View: in flow direction; extraction of the center line with the diameter of the hub.*

As can be seen in figure C.0.6, the center line contains comparably small cells in the horizontal directions. Moreover, it can be seen, that the center line includes one region with a high cell growth in radial direction. Figure C.0.7 shows the cell distribution around the blades.
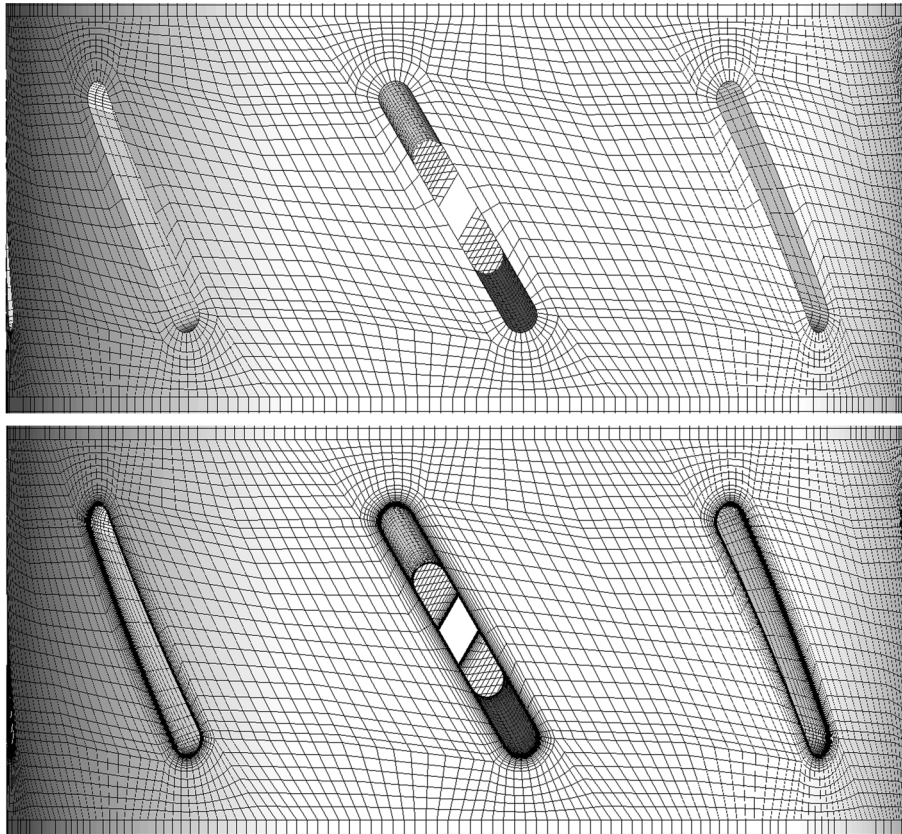
Figure C.0.7: *Cell distribution of the meshes for RANS and hybrid models around the blades. Upper picture: mesh for RANS models; Lower picture: mesh for hybrid models.*

Figure C.0.7 shows the cell distribution around the blades. Besides the difference in the mesh refinement towards the blades, it can be seen that the cell distribution is the same for all blades. As a result of the mesh generation process, this cell distribution continues around 360°.

References

[1] *ANSYS ICEM CFD 12.0 - User Manual.* SAS IP, Inc., 2009.

[2] L. Davidson. *Fluid mechanics, turbulent flow and turbulence modeling.* Course material. 2011.

[3] P. Dellenback, D. Metzger, and G. Neitzel. *Measurements in Turbulent Swirling Flow Through an Abrupt Expansion.* Slides. 1987.

[4] *Documentation for ANSYS ICEM CFD 12.1.* Help Navigator.

[5] "Experimental setup". Information given by the Institute of Fluid Mechanics and Hydraulic Machinery, University of Stuttgart. 2011.

[6] W. Gyllenram. "Analytical and Numerical Studies of Internal Swirling Flows". PhD thesis. Chalmers University of Technology, 2008.

[7] H. Jasak. "Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows". PhD thesis. Imperial College of Science, Technology and Medicine.

[8] L. Mangani. "Development and Validation of an Object Oriented CFD Solver for Heat Transfer and Combustion Modeling in Turbomachinery Applications". PhD thesis. Università degli Studi di Firenze.

[9] P. Moradnia. "CFD of Air Flow in Hydro Power Generators". Chalmers University of Technology, 2010.

[10] *OpenFOAM - User Guide. The Open Source CFD Toolbox.* Version 2.0.0. OpenCFD Limited, 2011.

[11] J. Paik and F. Sotiropoulos. "Numerical simulation of strongly swirling turbulent flows through an abrupt expansion". International Journal of Heat and Fluid Flow, 2010.

[12] M. L. Shur et al. "A hybrid RANS-LES approach with delayed-DES and wall-modelled LES capabilities". International Journal of Heat and Fluid Flow, 2008.

[13] R. Susan-Resiga et al. *Jet Control of the Draft Tube Vortex Rope in Francis Turbines at Partial Discharge.* $23^r d$ IAHR Symposium - Yokohama. 2006.

[14] *User's Guide.* Version 6.3. Fluent Inc., 2006.

[15] H. K. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics - The Finite Volume Method.* Pearson Education Limited, 2007.

[16] E. de Villiers. "The Potential of Large Eddy Simulation for the Modeling of Wall Bounded Flows". PhD thesis. Imperial college of Science, Technology and Medicine, 2006.

[17] M. B. Werner Haase and A. Revell. *DESider - A European Effort on Hybrid RANS-LES Modelling.* Springer, 2009.