CHALMERS

Studies of the ERCOFTAC Centrifugal Pump with OpenFOAM

Master's Thesis in Fluid Dynamics

SHASHA XIE

Department of Applied Mechanics
Division of Fluid Dynamics
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2010
Master's Thesis 2010:13

MASTER'S THESIS 2010:13

Studies of the ERCOFTAC Centrifugal Pump with OpenFOAM

Master's Thesis in Fluid Dynamics

SHASHA XIE

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

Göteborg, Sweden 2010

Studies of the ERCOFTAC Centrifugal Pump with OpenFOAM
SHASHA XIE

Master's Thesis 2010:13
ISSN 1652-8557
Department of Applied Mechanics
Division of Fluid Dynamics
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: + 46 (0)31-772 1000

Cover:
Contours of the relative velocity magnitude of the ERCOFTAC Centrifugal Pump model, computed in 2D and unsteady-state, using the OpenFOAM opensource CFD tool.

Chalmers Reproservice
Göteborg, Sweden 2010

Studies of the ERCOFTAC Centrifugal Pump with OpenFOAM
Master's Thesis in Fluid Dynamics
SHASHA XIE
Department of Applied Mechanics
Division of Fluid Dynamics
Chalmers University of Technology

Abstract

Numerical solutions of the rotor-stator interaction using OpenFOAM-1.5-dev was investigated in the ERCOFTAC Centrifugal Pump, a testcase from the ERCOFTAC Turbomachinery Special Interest Group [1]. The case studied was presented by Combès at the *ERCOFTAC Seminar and Workshop on Turbomachinery Flow Prediction VII*, in Aussois, 1999 [2]. It has 7 impeller blades, 12 diffuser vanes and 6% vaneless radial gap, and operates at the nominal operating condition with a Reynolds number of $6.5 * 10^5$ at a constant rotational speed of 2000 rpm.

2D and 3D models were generated to investigate the interaction between the flow in the impeller and that in the vaned diffuser using the finite volume method. The incompressible Reynolds-Averaged Navier-Stokes equations were solved together with the standard k-ε turbulence model. Both steady-state and unsteady simulations are employed for the 2D and 3D models. A Generalized Grid Interface (GGI) is implemented both in the steady-state simulations, where the GGI is used to couple the meshes of the rotor and stator, and in unsteady simulations, where the GGI is applied between the impeller and the diffuser to facilitate a sliding approach [3].

Several numerical schemes are considered such as Euler, Backward and Crank-Nicholson (with several off-centering coefficients) time discretization, and upwind and linear upwind convection discretization. Furthermore, the choice of different maximum Courant number and different unsteady solvers have been studied, and the required computational time has been compared for all the cases. The ensemble-averaged velocity components and the distribution of the ensemble-averaged static pressure coefficient at the impeller front end are calculated and compared against the available experimental data provided by Ubaldi [4].

The computational results show good agreement with the experimental results, although the upwind convection discretization fails in capturing the unsteady impeller wakes in the vaned diffuser. The case with a maximum Courant number of 4 is regarded as having the most efficient set-up, predicting the unsteadiness of the flow with a large time-step.

Keywords: CFD, OpenFOAM, Turbomachinery, GGI, ERCOFTAC Centrifugal Pump

Preface

This work has been carried out from January 2010 to June 2010 at the Division of Fluid Dynamics, Department of Applied Mechanics at Chalmers University of Technology, Göteborg, Sweden. In this study, 2D steady-state simulation was first performed, and those results were used as initial conditions for the unsteady simulations. A similar procedure was used for the 3D simulations.

The appendix includes all the results from all the cases, showing comparisons between the numerical results and all the available experimental data.

Acknowledgements

I would like to express my sincere gratitude to my supervisors, Dr. Håkan Nilsson and PhD student Olivier Petit for all the help, support and guidance during this work.

I also would like to acknowledge the Swedish National Infrastructure for Computing (SNIC) and Chalmers Centre for Computational Science and Engineering (C^3SE) for providing computational resources.

Furthermore, for sharing the measurement data of the ERCOFTAC Centrifugal Pump I am grateful to Professor Ubaldi and her team. Also I am thankful to Maryse Page, Martin Beaudoin, and Anne Marie Giroux from Hydro-Québec, Canada, and Mikko Auvinen from Aalto University, Finland, for sharing their new implementations.

In addition, I would like to thank Oscar Bergman, Andreu Oliver Gonzalez and Xin Zhao, who were performing their diploma work in parallel to mine, for the discussions and help.

Finally, I would like to give my special thanks to my parents for their continuous love and support.

Göteborg, June 2010
Shasha Xie

Contents

Nomenclature

A	pipe cross-sectional area
b	impeller blade span
$\widetilde{C}_p$	ensemble-averaged static pressure coefficient
c_r	radial absolute velocity
c_u	tangential absolute velocity
D_1	impeller inlet blade diameter
D_2	impeller outlet diameter
D_3	diffuser inlet vane diameter
D_4	diffuser outlet vane diameter
G_i	impeller circumferential pitch
k	turbulent kinetic energy
L	a characteristic linear dimension, (traveled length of fluid)
n	rotational speed
p	static pressure
p_t	total pressure
Q	flow rate
r	radial coordinate
Re	Reynolds number, $Re = \frac{\rho V L}{\mu} = \frac{VL}{\nu} = \frac{QL}{\nu A}$
R_n	meridional curvature radius
T	temperature
T_i	impeller blade passing period
t	time
$\bar{t}$	circumferential-averaged time
U_0	inlet radial speed
U_2	peripheral velocity at the impeller outlet
V	mean fluid velocity
w_r	radial relative velocity
w_u	tangential relative velocity
y_i	circumferential coordinate in the relative frame

Z axial thickness

z_d number of diffuser vanes

z_i number of impeller blades

ε dissipation

μ dynamic viscosity of the fluid

ν kinematic viscosity, $\nu = \frac{\mu}{\rho}$

ω angular velocity

ψ total pressure rise coefficient, $\psi = 2\left(p_{t4} - p_{t0}\right)/\rho U_2^2$

ρ air density

θ angular coordinate

φ flow rate coefficient, $\varphi = 4Q/\left(U_2 \pi D_2^2\right)$

■ diffuser vane position

▼ impeller blade position

Subscripts

0 in the suction pipe

1 at the impeller leading edge

2 at the impeller outlet

3 at the diffuser inlet

4 at the diffuser outlet

d relative to the diffuser

i relative to the impeller

m relative to the measuring point

r in the radial direction

u in the tangential direction

z in the axial direction

1 Introduction

In this section, the history of pumps will be mentioned as well as the development of the centrifugal pumps. Then the ERCOFTAC Centrifugal Pump studied in this work will be described including the previous studies.

1.1 Background

Pumps are designed to increase the pressure of a fluid. This principle is used in hydraulic pumps, ventilating fans and blowers since the earliest ages [5]. According to Reti [6], the Brazilian soldier and historian of science, the first machine that could be regarded as a centrifugal pump was a mud lifting machine in 1475 in a treatise by the Italian Renaissance engineer Francesco di Giorigio Martini. Real centrifugal pumps did not appear until the late 1600's, when Denis Papin made one with straight vanes [6]. The curved vane was invented by the British inventor John Appold in 1851 [6]. Afterward, due to the increasingly larger number of engines required for vehicle and aircraft propulsion, the centrifugal pumps have developed greatly. Due to the centrifugal pumps could be designed smaller for the same efficiency than the other pumps, they were preferred as the main element of engines.

In principle, the centrifugal pumps use a rotating impeller with blades to give rotation to the fluid, which is sucked through an inlet pipe. To optimize the design of centrifugal pumps, a lot of measurements are carried out. However, since experiments are limited by the facilities and the costs, computational fluid dynamics (CFD) is also used to complement the experiments. The lead times and costs of new designs may then be substantial reduced.

As an open-source, license-free, and object oriented C++ CFD toolbox, OpenFOAM (Open Field Operation and Manipulation) is becoming more and more popular for numerical simulations. Released as open source in 2004, it is the most widespread general purpose open-source CFD package, providing the option to modify the source code to fit the user's unique requirements.

This study uses OpenFOAM with a recently implemented Generalized Grid Interface (GGI) method [7] to conduct the steady-state simulation and the transient flow analysis of the ERCOFTAC Centrifugal Pump [4], shown in Fig.1.1. The report covers the relevant numerical methodologies together with the associated numerical approach of computational models and describe the simulation set-up of 2D and 3D models. In the end, the obtained dynamic flow results of several different numerical solutions are presented together with the comparison against the experimental performance.

1.2 Testcase description

The original ERCOFTAC Centrifugal Pump case was presented by Combès at a Turbomachinery Flow Prediction ERCOFTAC Workshop [2]. It is a simplified model of a centrifugal turbomachine which consists of a rotor with an outlet diameter of 420 mm and 7 backward impeller blades, and a rotatable vaned diffuser with 12 vanes and a 6% vaneless radial gap. The geometry illustrated in Fig.1.2 is given in Ubaldi *et al.* [4]. The measuring techniques used were a constant-temperature hot-wire anemometer with single sensor probes and fast response pressure transducers. The viscous and potential flow effects in the small radial gap between rotor and vaned diffusers in the ERCOFTAC Centrifugal Pump have been investigated. Also LDV measurements were performed by Ubaldi *et al.* [8] in the impeller and in the diffuser of the ERCOFTAC Centrifugal Pump by means of a four-beam two-color laser Doppler velocimeter. Recently, two-component LDV measurements of the unsteady boundary layer of the vane were published by Canepa, Cattanei, Ubaldi and Zunino [9].

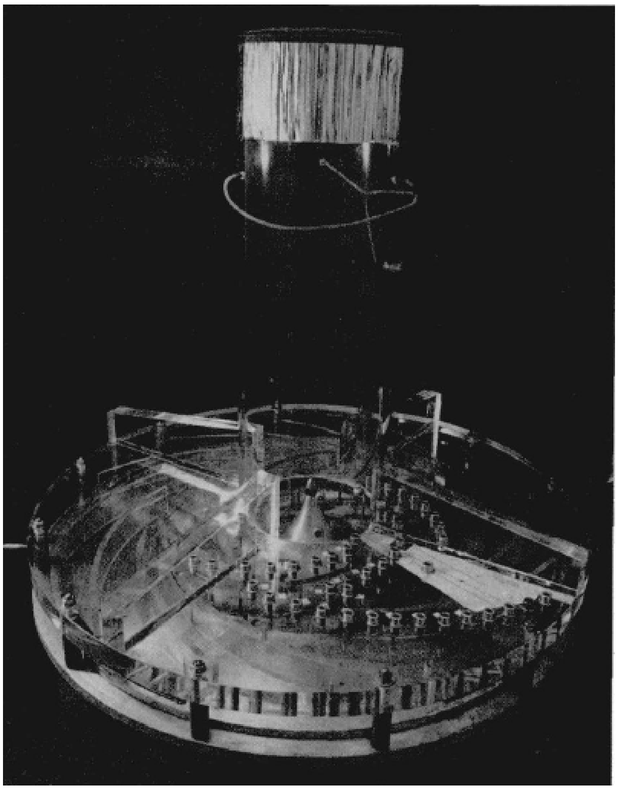


Figure 1.1: Centrifugal pump model [4].

Detailed flow measurements within the impeller and the vaneless diffuser were published by Ubaldi, Zunino and Ghiglione [10].

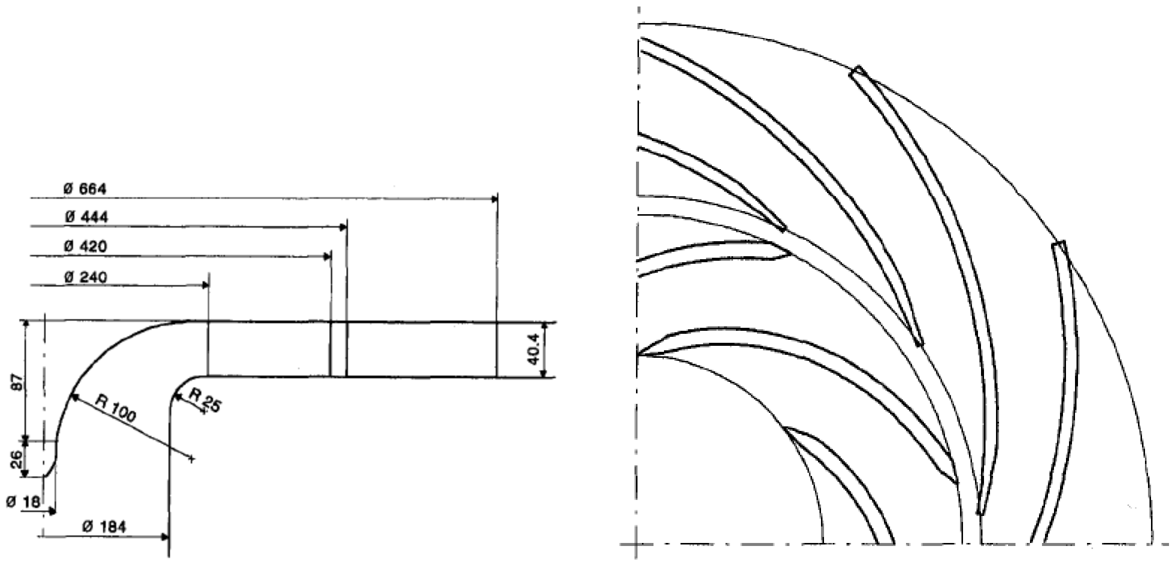


Figure 1.2: Impeller and vaned diffuser geometry [4].

The pump operates in an open circuit with air directly discharged into the atmosphere from the radial diffuser at the nominal operating condition with a constant rotational speed of 2000 rpm. The geometric data is shown in Tab.1.1, whereas the operating conditions are summarized in Tab.1.2.

Table 1.1: Geometric data [4].

Impeller		**Diffuser**	
inlet blade diameter	D_1=240 mm	inlet vane diameter	D_3=444 mm
outlet diameter	D_2=420 mm	outlet vane diameter	D_4=664 mm
blade span	b=40.4 mm	vane span	b=40.4 mm
number of blades	z_i=7	number of vanes	z_d=12

Table 1.2: Operating conditions [4].

Operating conditions	
rotational speed	n=2000 rpm
flow rate coefficient	φ=0.048
total pressure rise coefficient	ψ=0.65
Reynolds number	$Re=6.5 * 10^5$
temperature	T=298 K
air density	$\rho=1.2\ kg/m^3$

1.3 Related computational studies

There have been some numerical studies of the flow generated in the ERCOFTAC Centrifugal Pump and other similar devices. Based on a 2D model of the ERCOFTAC Centrifugal Pump, both steady and unsteady simulation were carried out using a finite element Navier-Stokes code by Bert, Combès and Kueny [11]. Good agreements were found compared with the experimental data. However, the main differences explained by the 3D secondary flows generated by the unshrouded impeller can be improved by 3D modeling simulations. A 2D model of the ERCOFTAC Centrifugal Pump corresponding to a meridional plane with a radial inlet, and a 3D model were initially analyzed by Combès, Bert and Kueny [12]. A multidomain method was implemented in a Navier-Stokes finite element code developed in the Research Division of Electricite de France. The results showed that the computational method developed was able to reproduce the unsteady flow effects and also complement the unsteady flow analysis performed by Ubaldi *et al.* [4]. Transient simulation of incompressible flow in the impeller and diffuser clearance in the ERCOFTAC Centrifugal Pump was performed by Torbergsen and White [13]. They also discussed how the velocity and pressure distribution can be related to the calculation of the dynamic forces. A 2D impeller and diffuser of the ERCOFTAC Centrifugal Pump model was simulated with the k-ε turbulence model. Those gave satisfactory agreement with published test results of radial velocities and pressure distributions in the impeller and diffuser clearance area [4], but was less good for the tangential velocity distribution. Sato and He [14][15][16] performed a 3D unsteady simulation of a single impeller and two diffuser blade passages in the ERCOFTAC Centrifugal Pump, and also of a complete 3D model. A 3D unsteady incompressible Navier-Stokes method based on the dual-time stepping and the pseudo-compressibility method was used. The predicted unsteady flow results showed reasonable agreement with the experimental data. They also gave a prediction of a mesh-independent trend where maximum efficiency is achieved when the radial gap is largest, and efficiency is decreased as the radial gap decreases. Unsteady rotor-stator simulations for the 2D complete (7 impeller blades/12 stator vanes) and simplified model (1 impeller blades/2 stator vanes) of the ERCOFTAC Centrifugal Pump using CFX-TASCflow code were performed by Page, Théroux and Trépanier [17]. The results from the detailed comparison with the experimental data showed that the rotor-stator interactions were captured. However, the computational results can be improved by extending the 2D model to 3D. Page and Beaudoin [18] have shown that OpenFOAM can produce similar results as other CFD codes for Frozen Rotor computations.

Although there are many CFD research activities on the flow in centrifugal pumps, most of them are based on 2D modeling but few of them succeed in simulating unsteady 3D flow on the whole rotor-stator mesh.

1.4 Approach in the present work

The block-structured mesh was generated by ICEM-HEXA with the Frozen Rotor approach for steady-state simulation and with the sliding grid approach for the unsteady simulation. A GGI method is used in the steady-state simulation to couple the meshes of rotor and stator, while in the unsteady simulation the GGI method is applied between the impeller and the diffuser to facilitate a sliding approach. In the unsteady simulation, the incompressible Reynolds-Averaged Navier-Stokes equations using a standard k-ε turbulence model are solved using the finite volume method. The choices of time discretization, convection discretization, maximum Courant number and solver are evaluated thoroughly, as well as the required computational time. For post-processing Paraview and Gnuplot are used. To verify the accuracy of the numerical solution with OpenFOAM there are many comparisons between numerical results and experimental data. As part of the activities in the OpenFOAM Turbomachinery working group [1], the aim of this work is to validate the simulation of the ERCOFTAC Centrifugal Pump as an application of turbomachines.

2 Data processing

In this section, the data processing of the available experimental data and simulated numerical results will be described together with some assumptions of unknown parameters. Using the assumptions and available measured data, the experimental performance will be replotted. To compare with the experimental data, the position used to investigate the features of the computed flow will be illustrated.

2.1 Data processing of the experimental results

The experimental data was provided by Ubaldi [4]. In order to reconstruct the distribution of the ensemble-averaged velocity (w_r and w_u), 17 measuring points were traversed in the axial direction at the impeller outlet ($D_m/D_2 = 1.02$) by hot-wire probes. To investigate the distribution of the ensemble-averaged static pressure coefficient ($\widetilde{C}_p$), 10 radial measuring locations were used from the impeller inlet to the outlet (R_m/R_2 from 0.53333 to 1.02), taken by means of miniature fast response transducers mounted at the stationary casing of the impeller. For each measuring point in both investigations, the probe was maintained at a fixed position with respect to the absolute frame of reference and the various relative probe-diffuser vane positions by rotation of the diffuser. Therefore, the distribution of the ensemble-averaged velocity (w_r and w_u) and the ensembled-averaged static pressure coefficient ($\widetilde{C}_p$) were investigated as a function of the relative frame circumferential coordinate y_i/G_i at the time instant t,

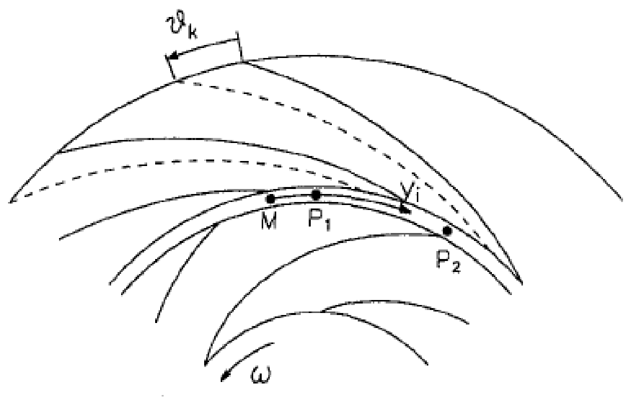


Figure 2.1: Sketch of the blades and reference coordinates [4].

Fig.2.1 shows a sketch of the reference coordinates. The circumferential coordinate for the probe fixed in point M in the mth diffuser passage with respect to the circumferential position θ_k of the diffuser is defined as following:

$$y_i(P_m) = \omega r\bar{t} + r\theta_k + (m-1)\frac{2\pi r}{z_d} \tag{2.1}$$

$$G_i = \frac{2\pi r}{z_i} \tag{2.2}$$

In order to reconstruct the distribution of the ensemble-averaged relative velocity (w_r and w_u) and the static pressure coefficient ($\widetilde{C}_p$), it is assumed that $\bar{t} = 0$ and $m = 1$. Then based on the Eqs.2.1 and 2.2, the instantaneous distributions of the ensemble-averaged radial (w_r) and tangential (w_u) relative velocity at the impeller outlet ($D_m/D_2 = 1.02$) at midspan position ($z/b = 0.5$) were replotted using the experimental data from Ubaldi *et al.* [4] as shown in Fig.2.2.

	#	Yi/Gi	Z/B	Cr/U2	Cu/U2
2		.00331	.50000	.16437	.42237
3		.00745	.50000	.16330	.41944
4		.01127	.50000	.16267	.41693
5		.01541	.50000	.16420	.41208
6		.01923	.50000	.16680	.40723
7		.02718	.50000	.18099	.39298
8		.04310	.50000	.20152	.37827
9		.05901	.50000	.23305	.36815
10		.07493	.50000	.24466	.36641

Figure 2.2: Original data file of radial (c_r) and tangential (c_u) absolute velocity [4].

According to the data shown in Fig.2.2, the radial relative velocity w_r, which is same as the radial absolute velocity c_r (in Fig.2.2), as a function of the relative frame circumferential coordinate y_i/G_i at the midspan position $z/b = 0.5$ could be plotted using the measured radial absolute velocity c_r as shown in the left-hand side of Fig.2.3. The tangential relative velocity w_u, which can be calculated by the measured tangential absolute velocity c_u (in Fig.2.2), could be plotted as a function of the relative frame circumferential coordinate y_i/G_i at the midspan position $z/b = 0.5$ as shown in the right-hand side of Fig.2.3.

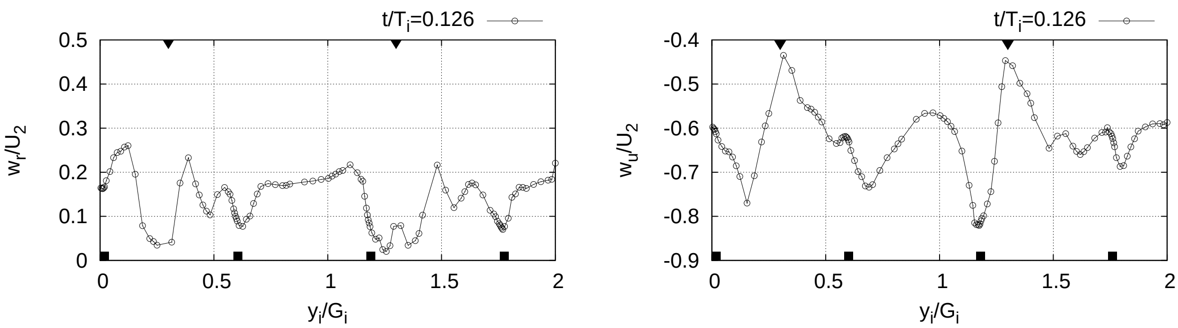


Figure 2.3: Ensemble-averaged distribution of the radial w_r (left) and tangential w_u (right) relative velocity at the impeller outlet $D_m/D_2 = 1.02$, at the midspan position $z/b = 0.5$.

Using the available experimental data, the instantaneous pictures of the ensemble-averaged radial (w_r) and tangential (w_u) relative velocity at the impeller outlet ($D_m/D_2 = 1.02$) with different span distance (z/b from 0 to 1) could be plotted shown in Fig.2.4, using the same method as plotting the instantaneous distribution of the ensemble-averaged radial (w_r) and tangential (w_u) relative velocity at the impeller outlet ($D_m/D_2 = 1.02$) at the midspan position ($z/b = 0.5$).

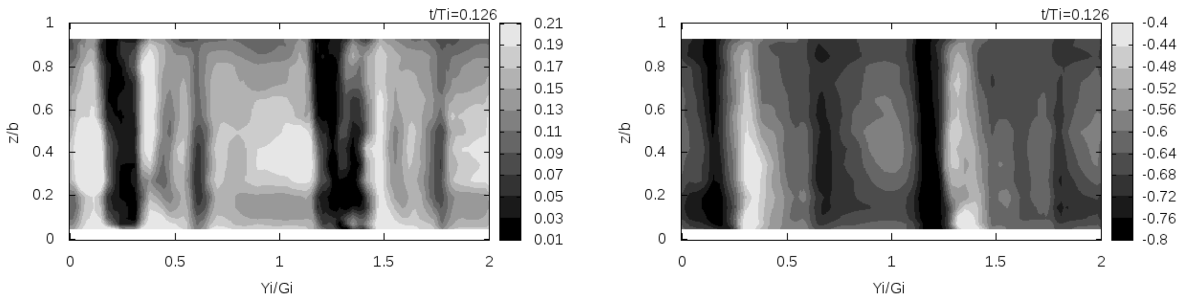


Figure 2.4: Instantaneous pictures of the ensemble-averaged radial (w_r) (left) and tangential (w_u) (right) relative velocity at the impeller outlet ($D_m/D_2 = 1.02$) with different span distance (z/b from 0 to 1).

Under the same definition of the relative frame circumferential coordinate system in Eqs.2.1 and 2.2, the instantaneous distribution of $\widetilde{C}_p$ at the impeller outlet using the experimental data from Ubaldi *et al.* [4] is shown in Fig.2.5.

1	R/R2	Teta(deg)	Cp
2	1.01905	4.274	.62056
3	1.01905	6.157	.60501
4	1.01905	8.039	.59435
5	1.01905	9.676	.57489
6	1.01905	10.495	.56715
7	1.01905	11.313	.56336
8	1.01905	12.132	.55543
9	1.01905	12.950	.54924
10	1.01905	14.587	.52289
11	1.01905	16.224	.48785
12	1.01905	17.043	.46532

Figure 2.5: Original data file of $\widetilde{C}_p$ [4].

With the same assumptions of $\bar{t} = 0$ and $m = 1$, for the first position shown in the first line of Fig.2.5 (with the value $Teta(deg) = 4.274$), assuming that the second column ($Teta(deg)$) is θ_k, Eqs.2.1 and 2.2 yield

$$y_i(P_m)|_{\widetilde{C}_p=0.62056} = r\theta_k = 1.01905R_2 \times 4.274 \times \frac{\pi}{180} \quad (2.3)$$

$$G_i = \frac{2\pi r}{z_i} = \frac{2\pi \times 1.01905R_2}{z_i} \quad (2.4)$$

where $R_2 = 0.21m$, $z_i = 7$. From Eqs.2.3 and 2.4, the x-axis value of point P can be calculated as:

$$(y_i/G_i)|_{\widetilde{C}_p=0.62056} = \frac{1.01905 \times 0.21 \times 4.274 \times \frac{\pi}{180}}{\frac{2\pi}{7} \times 1.01905 \times 0.21} = \frac{0.016}{0.192} = 0.083 \quad (2.5)$$

Then, according to the result from Eq.2.5, the distribution of $\widetilde{C}_p$ is shown in the left-hand side of Fig.2.6. The point described above is the left-most point in the left-hand side of Fig.2.6. For comparison, the plot of the distribution of $\widetilde{C}_p$ on the paper of Ubaldi *et al.* [4] is shown in the right-hand side of Fig.2.6.

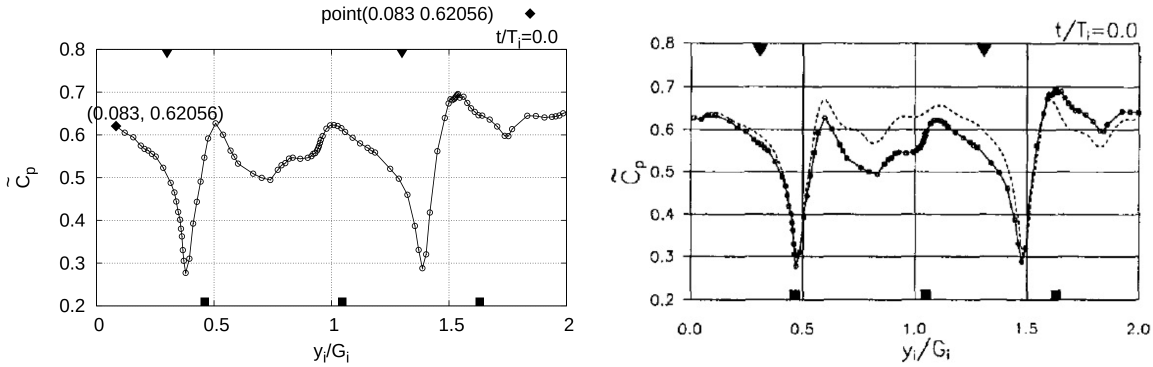


Figure 2.6: Instantaneous distribution of $\widetilde{C}_p$ at the impeller outlet $D_m/D_2 = 1.02$ for the replot using experimental data (left) and the original plot (right).

The plot in the left-hand side of Fig.2.6 shows some similarity with the one in the right-hand side of Fig.2.6 except for a shift in the x-axis of y_i/G_i, which is probably due to a wrong understanding from the part of the circumferential coordinate relative to the rotor in the left-hand side plot of Fig.2.6, of the measuring point M using the second column ($Teta(deg)$). By a trial-and-error method, the most similar plot was obtained by shifting the x-scale, yielding an addition of 4.6 degrees to the angles of Fig.2.5. This results in Fig.2.7 which is used to compare the numerical results of OpenFOAM with the experimental data.

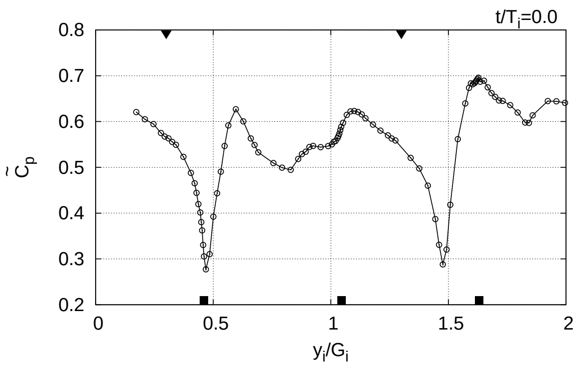


Figure 2.7: Modified plot of instantaneous distribution of $\widetilde{C}_p$ at the impeller outlet $D_m/D_2 = 1.02$.

Based on the above assumptions, for each radial measuring location (R_m/R_2 from 0.53333 to 1.02), the instantaneous pictures of the ensemble-averaged static pressure coefficient $\widetilde{C}_p$ could be replotted as well using the available experimental data shown in Fig.2.8

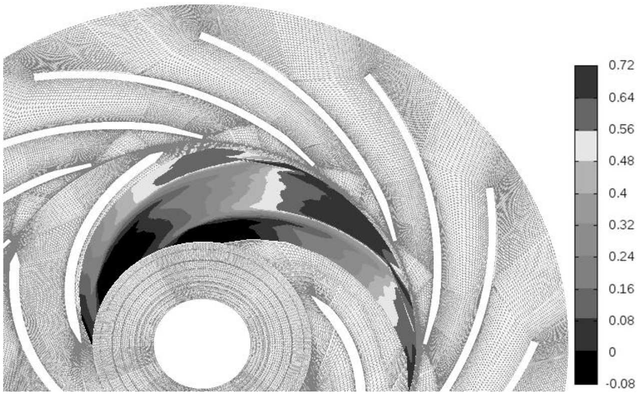


Figure 2.8: Instantaneous pictures of the ensemble-averaged static pressure coefficient $\widetilde{C}_p$ for each radial measuring location (R_m/R_2 from 0.53333 to 1.02).

2.2 Data processing of the numerical results

To compare with the experimental data, the numerical results are plotted along two impeller blades and three diffuser vanes at the small gap ($D_m/D_2 = 1.02$) between the impeller and the diffuser. For time $t/T_i = 0.126$, the relative position of the runner and stator is shown in Fig.2.9. Also the three positions (Probe 1, 2 and 3) used to put probes to monitor the pressure value during the simulations are shown in Fig.2.9, which has radials of 0.121 m, 0.2142 m and 0.32 m, respectively.

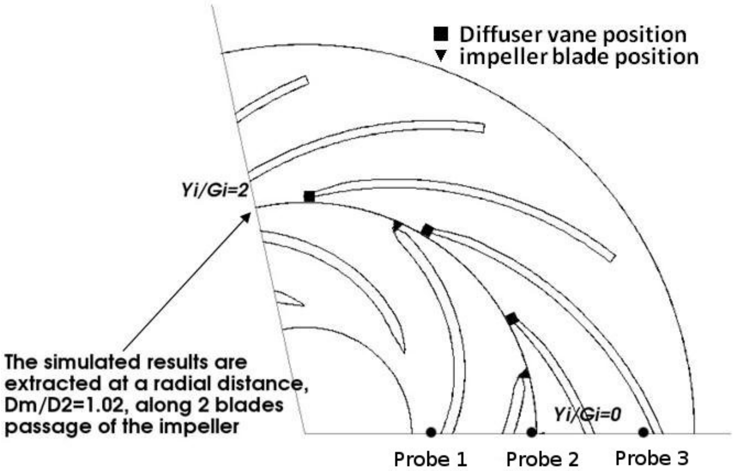


Figure 2.9: Position of the sampling of the simulated data for $t/T_i = 0.126$.

Furthermore, the simulated distribution of $\widetilde{C}_p$ uses the same normalization as that used by Ubaldi *et al.* [4]:

$$C_p = 2(p - p_0)/\rho U_2^2 \tag{2.6}$$

The parameter p_0 is the static pressure in the suction pipe. An assumption of p_0 in the numerical results is made by trying to obtain a similar level of $\widetilde{C}_p$ as the one presented by Ubaldi *et al.* [4], yielding $p_0 = 700Pa$. The same assumption has been used for plotting the instantaneous pictures of the ensemble-averaged static pressure coefficient $\widetilde{C}_p$ for each radial measuring location (R_m/R_2 from 0.53333 to 1.02) using numerical results. The positions for plotting such instantaneous pictures of $\widetilde{C}_p$ for each radial measuring location

(R_m/R_2 from 0.53333 to 1.02) are shown in Fig.2.8, where is between two impeller blades with respect to the position of the related diffuser vanes.

3 Theory

In this section, the governing equations and turbulence model used in the numerical simulations are described. Several time discretization methods and convection discretization schemes available in OpenFOAM are discussed together with the choice of maximum Courant number. Based on this analysis, the set-up for the simulation cases are presented.

3.1 The general transport equation

The general form of the transport equation for the flux ϕ is given by [19]

$$\underbrace{\frac{\partial(\rho\phi)}{\partial t}}_{temporal\,derivative} + \underbrace{div(\rho U\phi)}_{convection\,term} - \underbrace{div(\Gamma_\phi(div\phi))}_{diffusion\,term} = \underbrace{S_\phi}_{source\,term} \tag{3.1}$$

For an incompressible fluid the density ρ is constant, therefore the above equation becomes [19]

$$\rho\frac{\partial\phi}{\partial t} + \rho(div(U\phi)) - div(\Gamma_\phi(div\phi)) = S_\phi \tag{3.2}$$

3.2 The k-ε turbulence model

There are many different turbulence models, of which the k-ε model is used in this study. The k-ε model is the most common type of turbulence model. In the k-ε model the transport equations for the turbulent kinetic energy, k, and the dissipation, ε, are solved. For incompressible flow the equations read [19]

$$\frac{\partial k}{\partial t} + \frac{\partial(\overline{U_i}k)}{\partial x_i} = \frac{\partial}{\partial x_i}[(\nu + \frac{\nu_t}{\sigma_k})\frac{\partial k}{\partial x_i}] + P_k - \varepsilon \tag{3.3}$$

$$\frac{\partial \varepsilon}{\partial t} + \frac{\partial(\overline{U_i}\varepsilon)}{\partial x_i} = \frac{\partial}{\partial x_i}[(\nu + \frac{\nu_t}{\sigma_\varepsilon})\frac{\partial \varepsilon}{\partial x_i}] + \frac{\varepsilon}{k}(c_{\varepsilon1}P_k - c_{\varepsilon2}\varepsilon) \tag{3.4}$$

Where P_k is the production term and ν_t is the turbulent viscosity, which are expressed as [19]

$$P_k = \nu_t(\frac{\partial\overline{U_i}}{\partial x_i})^2 \tag{3.5}$$

$$\nu_t = c_\mu\frac{k^2}{\varepsilon} \tag{3.6}$$

Coefficients c_μ, $c_{\varepsilon1}$, $c_{\varepsilon2}$, σ_k and σ_ε in Eqs.3.3 - 3.6 are empirical constants, and the default values in OpenFOAM are shown in Tab.3.1.

Table 3.1: Values of the constants for the standard k-ε model.

Constants	Values
c_μ	0.09
$c_{\varepsilon1}$	1.44
$c_{\varepsilon2}$	1.92
σ_k	1
σ_ε	1.3

Eqs.3.3 and 3.4 are discretized and solved by a number of iterations until the solution is converged. A criteria is used to judge if the solution is converged by means of residuals.

3.3 Time discretization

In this section three different time discretization schemes are described, namely the first-order Euler Implicit discretization and two second-order schemes: Backward Differencing, and the Crank-Nicholson method.

The general form of the transport equation for the incompressible fluid is described above as Eq.3.2, which can be rewritten as [20]

$$\int_t^{t+\triangle t}[\rho\frac{\partial}{\partial t}\int_{V_P}\phi dV+\rho\int_{V_P}div(U\phi)dV-\int_{V_P}div(\Gamma_\phi(div\phi))dV]dt=\int_t^{t+\triangle t}(\int_{V_P}S_\phi(\phi)dV)dt \tag{3.7}$$

Where V_P is the control volume. Assuming that the control volume does not change in time, and the density and diffusivity in the control volume do not change in time as well, then Eq.3.7 becomes

$$\frac{\rho\phi_P(t+\triangle t)-\rho\phi_P(t)}{\triangle t}V_P+A[\phi_f(t+\triangle t)+\phi_f(t)]-B[(div\phi)_f(t+\triangle t)+(div\phi)_f(t)]=S \tag{3.8}$$

Where A and B are coefficients.

3.3.1 Euler

The Euler Implicit discretization only uses the value of the present time $(t+\triangle t)$ in all terms except the time term in Eq.3.8, yielding [20]

$$\frac{\rho\phi_P(t+\triangle t)-\rho\phi_P(t)}{\triangle t}V_P+A\phi_f(t+\triangle t)-B(div\phi)_f(t+\triangle t)=S \tag{3.9}$$

It can be seen from Eq.3.9 that the flux of node P is only related to the face flux at $t+\triangle t$. Therefore, the Euler Implicit scheme is only first-order accurate in time. The main error introduced by the first-order Euler scheme is given by the difference between the flux at $t+\triangle t$ and the flux at t, which is not include in the higher-order schemes. For this reason, the Euler method is less accurate than other higher-order methods, such as Crank-Nicholson and backward. On the other hand it can be more stable.

OpenFOAM provides the Euler time scheme for solving not only the first time derivative $\frac{\partial}{\partial t}$ terms but also any second time derivative $\frac{\partial^2}{\partial t^2}$ terms and it is the only scheme available for solving the second time derivative terms.

3.3.2 Crank-Nicholson

The form of Eq.3.8 is called the Crank-Nicholson method, which shows that the flux of node P is related not only to the face flux at $t+\triangle t$ but also to the face flux at t. It means the Crank-Nicholson method is second-order accurate in time. The Crank-Nicholson method provided by OpenFOAM has a off-centering coefficient, which refers to pure Crank-Nicholson when the off-centering coefficient is equal to 1, and refers to pure Euler time discretization when the off-centering coefficient equals to 0. In the range of 0 to 1, the off-centering coefficient is used to combine the second-order Crank-Nicholson and first-order Euler discretization schemes in time. The Crank-Nicholson method is less stable than the fully implicit scheme. However, with sufficiently small time steps, in principle, it is possible to achieve considerably accuracy with the Crank-Nicholson method in time. Compared with the Euler time discretization scheme, the Crank-Nicholson has an additional adjustment term the face flux at t, which makes the Crank-Nicholson time discretization scheme more suitable to the time-dependent problem than the Euler time discretization method.

3.3.3 Backward differencing

The second-order Backward Differencing in time is an implicit method and still neglects the variation of the flux value at the face of the cell. The discretized form is obtained by using the Taylor series expansion of the flux values $\phi(t)$ and $\phi(t - \triangle t)$ [20]

$$\phi(t) = \phi(t + \triangle t) - \frac{\partial \phi}{\partial t}\triangle t + \frac{1}{2}\frac{\partial^2 \phi}{\partial t^2}\triangle t^2 + O(\triangle t^3) \tag{3.10}$$

$$\phi(t - \triangle t) = \phi(t + \triangle t) - \frac{\partial \phi}{\partial t}(2\triangle t) + \frac{1}{2}\frac{\partial^2 \phi}{\partial t^2}(2\triangle t)^2 + O((2\triangle t)^3) \tag{3.11}$$

Eq.3.11 can be rewritten as

$$\phi(t - \triangle t) = \phi(t + \triangle t) - 2\frac{\partial \phi}{\partial t}\triangle t + 2\frac{\partial^2 \phi}{\partial t^2}\triangle t^2 + O(\triangle t^3) \tag{3.12}$$

Then from Eqs.3.10 and 3.12 it can be derived as [20]

$$\frac{\partial \phi}{\partial t} = \frac{\frac{2}{3}\phi(t + \triangle t) - 2\phi(t) + \frac{1}{2}\phi(t - \triangle t)}{\triangle t} \tag{3.13}$$

Thus the final discretized equation with Backward Differencing in time is given by [20]

$$\frac{\frac{2}{3}\rho\phi(t + \triangle t) - 2\rho\phi(t) + \frac{1}{2}\rho\phi(t - \triangle t)}{\triangle t}V_P + A[\phi_f(t + \triangle t)] - B[(div\phi)_f(t + \triangle t)] = S \tag{3.14}$$

3.3.4 Maximum Courant number

The limitation for the Courant number is that the system might become unstable if the Courant number is too large, Therefore, the maximum Courant number defined per control volume face is used to guarantee stability of the system, i.e.

$$Co = \frac{\delta t U_f}{\delta x} \tag{3.15}$$

where δx is the distance between the adjacent cell centers. It can be seen from Eq.3.15 that, for the same mesh, the Courant number is directly proportional to the time step. OpenFOAM provides the possibility to specify either a constant time step, or a constant Courant number. The higher the Courant number the larger the time-step will be adjusted, of course, which can be adjusted automatically in OpenFOAM.

3.4 Convection discretization

The convection term of Eq.3.2 can be discretized in many different ways. The two alternations that have been used in the present work are described in the following sections.

3.4.1 The upwind

The upwind convection scheme is a numerical discretization method for solving differential equations by using differencing biased in the direction of the flux. Based on a one-dimensional control volume, consider the flux at the east face of the control volume with the node P in the center of the control volume and the west neighbor node of W and the east neighbor node of E as shown in Fig.3.1

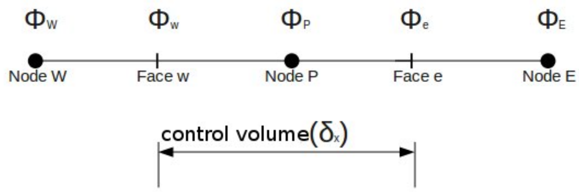


Figure 3.1: Upwind convection discretization.

The flux value at the east face is determined according to the direction of the flow, according to

$$\phi_e = \begin{cases} \phi_P \; if\, the\, direction\, of\, the\, flux\, at\, the\, east\, face\, is\, out\, of\, the\, control\, volume \\ \phi_E \; if\, the\, direction\, of\, the\, flux\, at\, the\, east\, face\, is\, into\, the\, control\, volume \end{cases} \tag{3.16}$$

This scheme is always bounded but only first-order. Usually the upwind convection scheme is used in the initial phase of a simulation for unsteady flows, and a higher-order scheme is then used to get accurate results.

3.4.2 The linear-upwind

The spatial accuracy of the first-order upwind scheme can be improved by choosing an additional correction. In addition to the first-order upwind estimation ϕ_P the linear-upwind convection scheme assume the linear variation of flux between P and N as shown in Fig.3.2

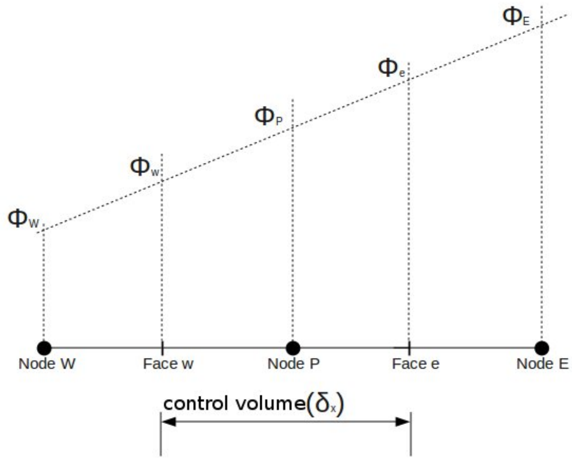


Figure 3.2: The linear-upwind convection discretization.

The flux value at the east face for the case where the flux is out of the control volume is calculated according to [19]

$$\phi_e = \phi_P + \frac{(\phi_P - \phi_W)}{\delta x}\frac{\delta x}{2} = \phi_P + \frac{1}{2}(\phi_P - \phi_W) \tag{3.17}$$

The linear-upwind scheme has a second-order accuracy.

4 Numerical approach

In this section, the computational mesh will be described, as well as the algorithms and the solvers. Boundary conditions and case settings used for the 2D and 3D simulations will be discussed as well.

4.1 Computational mesh

The block-structured mesh, see Fig.4.1 was generated by ICEM-HEXA, and the rotor and the stator were meshed separately. The mesh consists of about 94 000 cells for the 2D cases and 2 000 000 cells for the 3D cases. To couple the two parts of the mesh (rotor and stator) the Generalized Grid Interface (GGI) is used. Developed by M. Beaudoin and H. Jasak [7], the purpose of the GGI is to couple multiple non-conformal meshes. The GGI interface is widely used in turbomachinery, where complicated geometries can be coupled together.

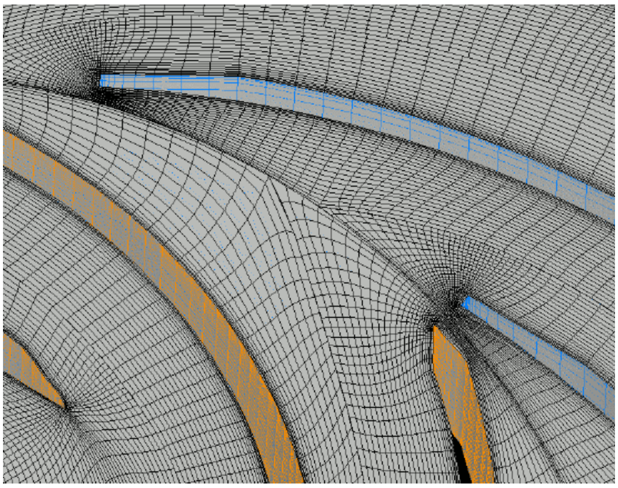


Figure 4.1: Grid mesh with GGI [7].

In the steady-state simulation, the GGI is used to couple the meshes statically. In the unsteady simulation, the GGI is applied between the rotor and the stator yielding a sliding approach [3].

4.2 Pressure-velocity coupling

The SIMPLE algorithm and the PISO algorithm are used for coupling the pressure-velocity system. The SIMPLE pressure-velocity coupling procedure by Patankar [21] is used in the *simpleTurboMFRFoam* solver (see section 4.3.2) and the *transientSimpleDyMFoam* solver (see section 4.3.3). The PISO procedure proposed by Issa [22] is used in the *turbDyMFoam* solver (see section 4.3.1) [20].

4.2.1 The SIMPLE algorithm

The SIMPLE algorithm has been used for the pressure-velocity coupling in some of the simulations of the present work. Much larger time-steps are allowed with the SIMPLE algorithm compared with the PISO algorithm. The main procedure of the SIMPLE algorithm is described in [19]. To get more proximity pressure, more times for solving the pressure equation is defined by the parameter named *nNonOrthogonalCorrectors* in the SIMPLE function located in the file named *fvSolution* in the folder of *system*.

4.2.2 The PISO algorithm

For transient flow calculations of the pressure-velocity coupling is solved by the PISO algorithm. The main procedure of the PISO algorithm is described in [19]. In OpenFOAM there are three parameters defined in the PISO algorithm, namely *nCorrectors*, *nOuterCorrectors* and *nNonOrthogonalCorrectors*. The parameter *nNonOrthogonalCorrectors* defines how many times the pressure equation is solved in case the mesh is not good enough. How many times the pressure equation is iterated is defined by the parameter of *nCorrectors*. While *nOuterCorrectors* is used to control the number of iterations of the Reynolds-Averaged Navier-Stokes equations, which includes the pressure and the velocity components.

4.3 Solvers

In this section three different solvers are described, which are the *turbDyMFoam* solver used in the unsteady simulation of the 2D model, the *simpleTurboMFRFoam* solver used in the steady-state simulation for both the 2D and 3D models, and the *transientSimpleDyMFoam* solver used in the unsteady simulation for both the 2D and 3D models. The steady Reynolds-Averaged Navier-Stokes equation is first solved in the steady-state simulation with the help of the *simpleTurboMFRFoam* solver, and then the time-dependency unsteady Reynolds-Averaged Navier-Stokes equation is resolved by the *turbDyMFoam* solver or the *transientSimpleDyMFoam* solver.

4.3.1 The *turbDyMFoam* solver

The *turbDyMFoam* solver is used as a transient solver for incompressible turbulent flow of Newtonian fluids with moving mesh. The *turbDyMFoam* solver uses the PISO algorithm for pressure-velocity coupling and uses libraries for mesh motion and deformation of polyhedral meshes [3]. It solves the Reynolds-Averaged Navier-Stokes equations at each time step, then the rotating part rotates and the procedure is repeated for the next time step. The coupling between the rotating and stationary parts is done through a sliding GGI interface. The time-step limitation of the PISO algorithm makes the *turbDyMFoam* solver less robust for the 3D unsteady simulation in this work.

4.3.2 The *simpleTurboMFRFoam* solver

The solver used in this work for both 2D and 3D steady-state simulations, namely *simpleTurboMFRFoam*, is a finite volume steady-state solver for incompressible, turbulent flow of non-Newtonian fluids, using the SIMPLE algorithm for pressure-velocity coupling. The solver uses the Multiple Reference Frame (MRF) approach, which requires no relative mesh motion of the rotating and the stationary parts (also referred to as the Frozen Rotor approach). The momentum equations are solved by a mix of inertial and relative velocities in the relative frame together with the additional Coriolis term for the rotating part, i.e.

$$\nabla \cdot (\overrightarrow{u_R} \otimes \overrightarrow{u_I}) + \underbrace{\overrightarrow{\Omega} \times \overrightarrow{u_I}}_{Coriolis\,term} = -\nabla(p/\rho) + \nu \nabla \cdot \nabla(\overrightarrow{u_I}) \tag{4.1}$$

$$\nabla \cdot \overrightarrow{u_I} = 0 \tag{4.2}$$

The *simpleTurboMFRFoam* solver is quick and provides a good starting guess for unsteady simulation, but does not predict the transient behavior of the flow.

4.3.3 The *transientSimpleDyMFoam* solver

The *transientSimpleDyMFoam* solver is also a transient solver for incompressible turbulent flow of Newtonian fluids with dynamic mesh. However, a SIMPLE-based algorithm in time-stepping mode is implemented in the *transientSimpleDyMFoam* solver. The turbulence model solution is moved inside the SIMPLE loop, which makes it more robust than the *turbDyMFoam* solver. The *transientSimpleDyMFoam* solver allows big time-step to be taken but still always with a proper number of iterations within each time-step. Therefore, in this study the 3D unsteady simulations are carried out with the *transientSimpleDyMFoam* solver.

4.4 Boundary conditions

The boundary conditions used in all the simulations are shown in Tab.4.1. It should be noted that the 2D cases have a radial inlet, while the 3D cases have an axial inlet. The Z thickness of the 2D cases is due to the fact that OpenFOAM needs a one-cell-thickness. The inlet velocity is however eveluated using the physical Z thickness.

Table 4.1: Boundary conditions for all the cases.

Calculated data for the 2D cases		**Boundary conditions for the 2D cases**	
Inlet Diameter	D_0=200 mm	At the inlet	V_{radial}=U_0
Z thickness	Z=1 mm		$\frac{\mu_T}{\mu}$=10 (viscosity ratio)
Flow rate	Q=$\frac{\varphi U_2 \pi D_2^2}{4}$=0.292 m^3/s		k=$\frac{3}{2}U_0^2 I^2$=0.48735 m^2/s^2 (I=5%)
Inlet radial speed	U_0=$\frac{Q}{A_0}$=$\frac{Q}{2\pi r_0 * 0.04}$=11.4 m/s		ε=$\frac{C_\mu \rho k^2}{\mu_T}$=$\frac{C_\mu \rho k^2}{\mu(\mu_T/\mu)}$=$\frac{C_\mu k^2}{\nu(\mu_T/\mu)}$
Rotating speed	$\omega = 2000 rpm$	At the outlet	Average static pressure 0
Calculated data for the 3D cases		**Boundary conditions for the 3D cases**	
Inlet Diameter	D_0=184 mm	At the inlet	V_{axial}=U_0
Z thickness	Z=40 mm		$\frac{\mu_T}{\mu}$=10 (viscosity ratio)
Flow rate	Q=$\frac{\varphi U_2 \pi D_2^2}{4}$=0.292 m^3/s		k=$\frac{3}{2}U_0^2 I^2$=0.4521 m^2/s^2 (I=5%)
Inlet axial speed	U_0=$\frac{Q}{A_0}$=$\frac{Q}{2\pi r_0 * 0.04}$=10.98 m/s		ε=$\frac{C_\mu \rho k^2}{\mu_T}$=$\frac{C_\mu \rho k^2}{\mu(\mu_T/\mu)}$=$\frac{C_\mu k^2}{\nu(\mu_T/\mu)}$
Rotating speed	$\omega = 2000 rpm$	At the outlet	Average static pressure 0

4.5 Case set-up

The computational cases included in this work were constructed to evaluate numerous aspects concerning analysis of turbomachinery with OpenFOAM. The main elements of consideration were convection schemes, time discretization methods for unsteady simulations, choice of different maximum Courant Number as well as required computational time for unsteady numerical solutions. The list of all the simulation cases is shown in Tab.4.2. The case names are designed to reflect the differences between the cases. Learning to understand the case names help understatnding the presentation of the results. Furthermore, four listings of the essential computational settings characterizing the numerical simulations are shown in Tab.4.3-Tab.4.6.

Table 4.2: Description of the different cases and **BOLD** text build up the case names.

Name of case	**Simulation type**	**Time scheme**	**convection scheme**	**maxCo**	**solver**
2DSteady	**2D steady-**state simulation	-	linearUpwind	-	simpleTurbo-MFRFoam
2DEulerU0.5T	**2D** unsteady simulation	**Euler**	**u**pwind	**0.5**	**t**urbDyMFoam
2DEulerL0.5T	**2D** unsteady simulation	**Euler**	linearUpwind	**0.5**	**t**urbDyMFoam
2DBackL0.5T	**2D** unsteady simulation	**Back**ward	linearUpwind	**0.5**	**t**urbDyMFoam
2DCN0.2L0.5T	**2D** unsteady simulation	**C**rank-**N**icholson **0.2**	linearUpwind	**0.5**	**t**urbDyMFoam
2DCN0.5L0.5T	**2D** unsteady simulation	**C**rank-**N**icholson **0.5**	linearUpwind	**0.5**	**t**urbDyMFoam
2DCN0.8L0.5T	**2D** unsteady simulation	**C**rank-**N**icholson **0.8**	linearUpwind	**0.5**	**t**urbDyMFoam
2DCN1.0L0.5T	**2D** unsteady simulation	**C**rank-**N**icholson **1.0**	linearUpwind	**0.5**	**t**urbDyMFoam
2DCN0.5L1.0T	**2D** unsteady simulation	**C**rank-**N**icholson **0.5**	linearUpwind	**1.0**	**t**urbDyMFoam
2DCN0.5L2.0T	**2D** unsteady simulation	**C**rank-**N**icholson **0.5**	linearUpwind	**2.0**	**t**urbDyMFoam
2DCN0.5L4.0T	**2D** unsteady simulation	**C**rank-**N**icholson **0.5**	linearUpwind	**4.0**	**t**urbDyMFoam
2DCN0.5L0.5S	**2D** unsteady simulation	**C**rank-**N**icholson **0.5**	linearUpwind	**0.5**	transient**S**imple-DyMFoam
3DSteady	**3D steady-**state simulation	-	linearUpwind	-	simpleTurbo-MFRFoam
3DBackL0.5S	**3D** unsteady simulation	**Back**ward	linearUpwind	**0.5**	transient**S**imple-DyMFoam

4.5.1 2D steady-state simulation

In the 2D steady-state simulation the Frozen Rotor approach is used, where the rotor and stator have fixed relative position with respect to each other. It greatly simplifies the problem, shortening the computational time. This allows many parameters to be compared to the experimental data, but the results are not as accurate as unsteady results [3]. The settings of the 2D steady-state case is listed in Tab.4.3.

Table 4.3: Settings for the 2D steady-state simulation.

Schemes	Convection schemes of	U	linearUpwind
		k,ε	upwind
Solvers	p	GAMG	
		smoother	GaussSeidel
		tolerance	1.0e-08
		relTol	0.05
	U,k,ε	smoothSolver	
		smoother	GaussSeidel
		tolerance	1.0e-07
		relTol	0.1

4.5.2 2D unsteady simulation

Using the converged results from the 2D steady-state simulation as initial guess, 2D unsteady simulations have been performed. Because of the time-dependency, 2D unsteady simulations are more complex than steady-state simulations. In the unsteady simulation a sliding grid approach is applied, where the rotor mesh rotates with respect to the stator mesh. The interaction between the rotor and stator is thus fully resolved [3]. The settings of the 2D unsteady cases are listed in Tab.4.4.

Table 4.4: Settings for the different 2D unsteady simulations.

Schemes	Time discretization schemes	backward/Euler/Crank-Nicholson(0.2/0.5/0.8/1.0)	
	Convection schemes of	U	upwind/linearUpwind
		k,ε	upwind
Control	Time step	maxCo	0.5/1/2/4
	Correctors for solver *turbDyMFoam*	nCorrectors	2
		nOuterCorrectors	1
		nNonOrthogonalCorrectors	1
	Correctors for solver *transientSimpleDyMFoam*	nCorrectors	0
		nOuterCorrectors	1
		nNonOrthogonalCorrectors	0
Solvers	p,U,k,ε	BiCGStab	
		preconditioner	DILU
		tolerance	1.0e-07
		relTol	0
	pcorr	BiCGStab	
		preconditioner	DILU
		tolerance	1.0e-02
		relTol	0
	pFinal	BiCGStab	
		preconditioner	DILU
		tolerance	1.0e-09
		relTol	0

To compare the results of the 2D unsteady cases with the experimental data, the solutions need to get fully developed. The pressure fluctuations of case 2DCN0.5L0.5T observed at three different points (Probe 1, 2 and 3 in Fig.2.9) are shown in Fig.4.2. The results for all the 2D unsteady cases are considered developed after 0.3s, which is used to stop the simulations.

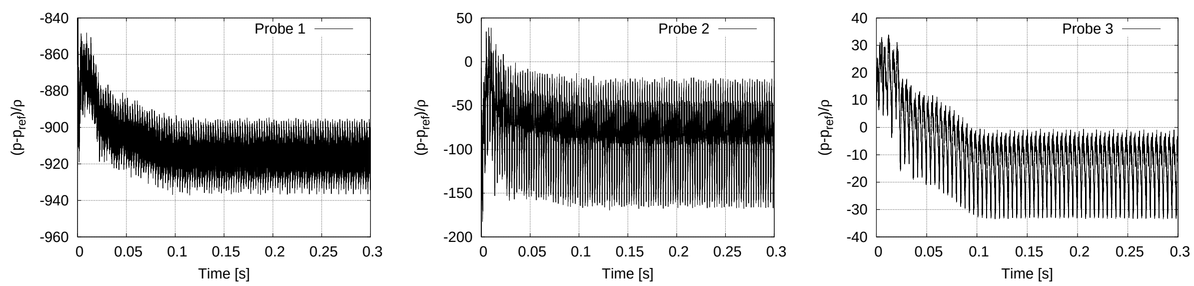


Figure 4.2: Pressure fluctuations of case 2DCN0.5L0.5T at Probe 1, 2 and 3.

4.5.3 3D steady-state simulation

From the 2D analysis, the best parameters are underlined, and used to perform the 3D analysis. A steady-state simulation is first performed to quickly obtain a general flow behavior. The settings for the 3D steady-state simulation are listed in Tab.4.5.

Table 4.5: Settings for the 3D steady-state simulation.

<table>
<tr><td rowspan="2">Schemes</td><td rowspan="2">convection schemes of</td><td>U</td><td>linearUpwind</td></tr>
<tr><td>k,ε</td><td>upwind</td></tr>
<tr><td rowspan="4">Solvers</td><td rowspan="4">p,U,k,ε</td><td>GAMG</td><td></td></tr>
<tr><td>smoother</td><td>GaussSeidel</td></tr>
<tr><td>tolerance</td><td>1.0e-08</td></tr>
<tr><td>relTol</td><td>0.05</td></tr>
</table>

4.5.4 3D unsteady simulation

3D unsteady simulations are performed using the *transientTurbDyMFoam* solver and the converged value of the 3D steady-state case result as initial guess. The settings for the 3D unsteady case are listed in Tab.4.6.

Table 4.6: Settings for the 3D unsteady simulation.

<table>
<tr><td rowspan="3">Schemes</td><td colspan="3">Time discretization schemes</td><td colspan="3">backward</td></tr>
<tr><td colspan="3" rowspan="2">convection schemes of</td><td>U</td><td colspan="2">linearUpwind</td></tr>
<tr><td>k,ε</td><td colspan="2">upwind</td></tr>
<tr><td rowspan="4">Control</td><td colspan="3">time step</td><td colspan="2">maxCo</td><td>0.5</td></tr>
<tr><td colspan="3" rowspan="3">Correctors</td><td colspan="2">nCorrectors</td><td>0</td></tr>
<tr><td colspan="2">nOuterCorrectors</td><td>1</td></tr>
<tr><td colspan="2">nNonOrthogonalCorrectors</td><td>0</td></tr>
<tr><td rowspan="12">Solvers</td><td rowspan="7">U</td><td>PBiCG</td><td></td><td rowspan="7">k,ε</td><td>PBiCG</td><td></td></tr>
<tr><td>preconditioner</td><td>DILU</td><td>preconditioner</td><td>DILU</td></tr>
<tr><td>smoother</td><td>DILU</td><td>smoother</td><td>DILU</td></tr>
<tr><td>minIter</td><td>1</td><td>minIter</td><td>1</td></tr>
<tr><td>maxIter</td><td>4</td><td>maxIter</td><td>3</td></tr>
<tr><td>tolerance</td><td>1.0e-07</td><td>tolerance</td><td>1.0e-07</td></tr>
<tr><td>relTol</td><td>0</td><td>relTol</td><td>0</td></tr>
<tr><td rowspan="5">p</td><td>PCG</td><td></td><td rowspan="5">pcorr,pFinal</td><td>PCG</td><td></td></tr>
<tr><td>preconditioner</td><td>DIC</td><td>preconditioner</td><td>DIC</td></tr>
<tr><td>tolerance</td><td>1.0e-05</td><td>tolerance</td><td>1.0e-05</td></tr>
<tr><td>relTol</td><td>0.002</td><td>relTol</td><td>0.001</td></tr>
<tr><td>minIter</td><td>2</td><td>minIter</td><td>2</td></tr>
<tr><td>maxIter</td><td>140</td><td>maxIter</td><td>280</td></tr>
</table>

The 3D unsteady simulation need to be fully developed to be compared with the experimental data. The pressure fluctuations of case 3DBackL0.5S observed at three different points (Probe 1, 2 and 3 in Fig.2.9) are shown in Fig.4.3. Although the pressure value at Probe 1 still has very small development at the time very close to 0.3s, the results of the 3D unsteady case 3DBackL0.5S is considered developed at 0.3s, which is used to stop the simulation.

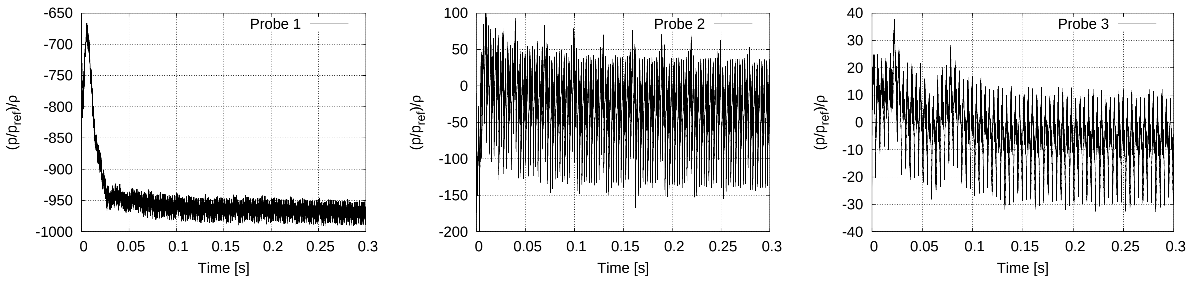


Figure 4.3: Pressure fluctuations of case 3DBackL0.5S at Probe 1, 2 and 3.

5 Results and discussions

In this section, the numerical results are compared with all the available experimental data. The differences between different numerical solutions are discussed with respect to the influence on the case set-ups and to the prediction of unsteady flow features. The 2D steady-state simulation is first discussed followed by the 2D unsteady simulations. Then the 3D steady-state simulation is followed by the 3D unsteady simulation.

5.1 2D steady-state simulation

A 2D representation of the geometry is used together with the Frozen Rotor approach and the *simpleTurboMFRFoam* solver. The 2D steady-state simulation was stopped after 5000 iterations, since all the residuals are below 10^{-5}, as shown in Fig.5.1.

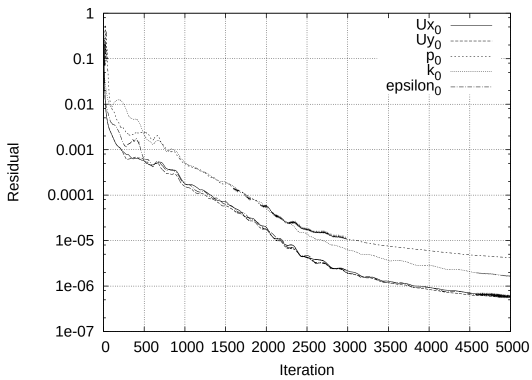


Figure 5.1: Residuals of velocity components, pressure, k and ε for case 2DSteady.

Since the Frozen Rotor approach resembles a snapshot of the real flow in the pump, the position of the impeller and the diffuser are fixed to each other. Therefore, the wakes in the diffuser region are not physical [3], as shown in Fig.5.2.

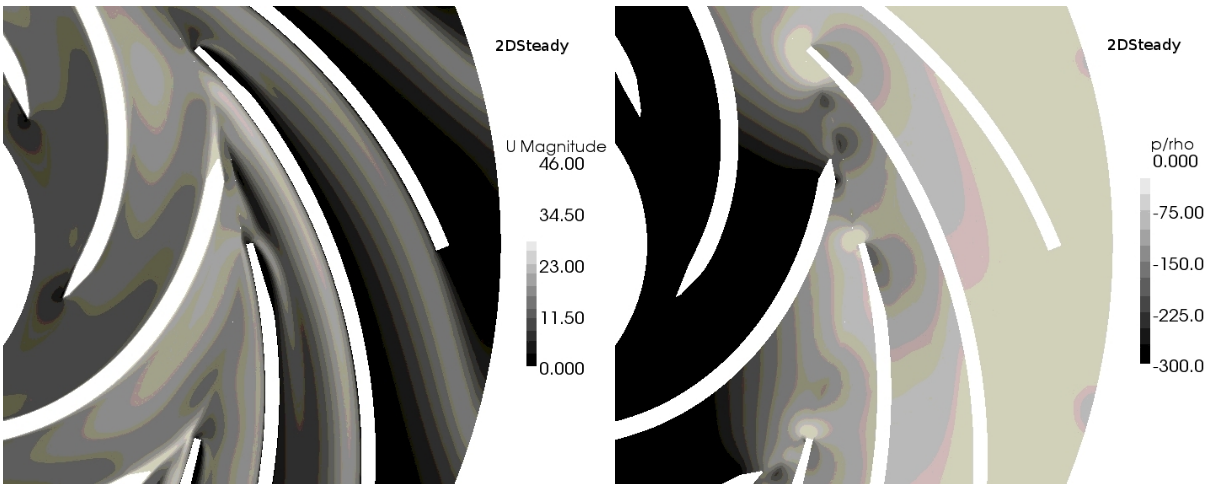


Figure 5.2: Relative velocity magnitude (left) and static pressure (right) for case 2DSteady.

The computed velocities and static pressure coefficient at the impeller outlet ($D_m/D_2 = 1.02$) are shown in Fig.5.3. Compared to the experimental data, they have some similarities but still do not perfectly agree, which is probably due to the Frozen Rotor approach rather than the OpenFOAM implementation [3].

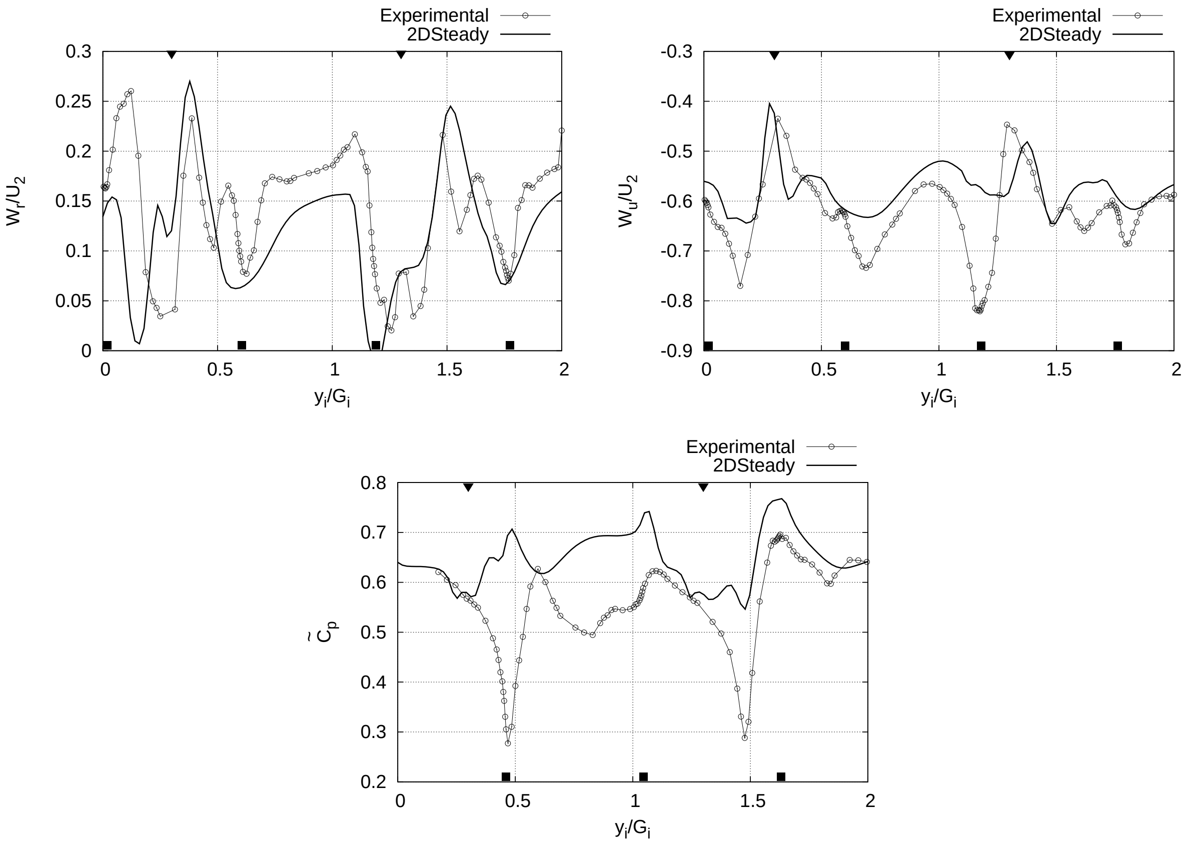


Figure 5.3: Radial (top left) and tangential (top right) velocity, and the static pressure coefficient (bottom) for case 2DSteady.

5.2 2D unsteady simulation

The 2D unsteady simulations are performed using the converged results of the 2D steady-state simulation as the initial guess. Since the flow in the centrifugal pump have an unsteady behavior, the unsteady solution is expected to have a better agreement with the experimental data.

5.2.1 Comparison of convection discretization schemes

The main mechanisms of the unsteady flow in the ERCOFTAC centrifugal pump are the wake and potential flow effects around the blades. The following discussions are mainly focused on the difference between the results from the linear upwind and upwind convection schemes, which have second-order and first-order accuracy, respectively. The second-order linear upwind convection scheme predicts flow unsteadiness better than the first-order upwind convection scheme. The results of the two numerical solutions are shown in Fig.5.4. The wakes of the rotor blades can be observed in the diffuser blade passages, in the 2DEulerL0.5T case, but not in the 2DEulerU0.5T case, as shown in Fig.5.4. That means that the upwind discretization scheme with first-order behavior fails to capture the wakes of the unsteady flow.

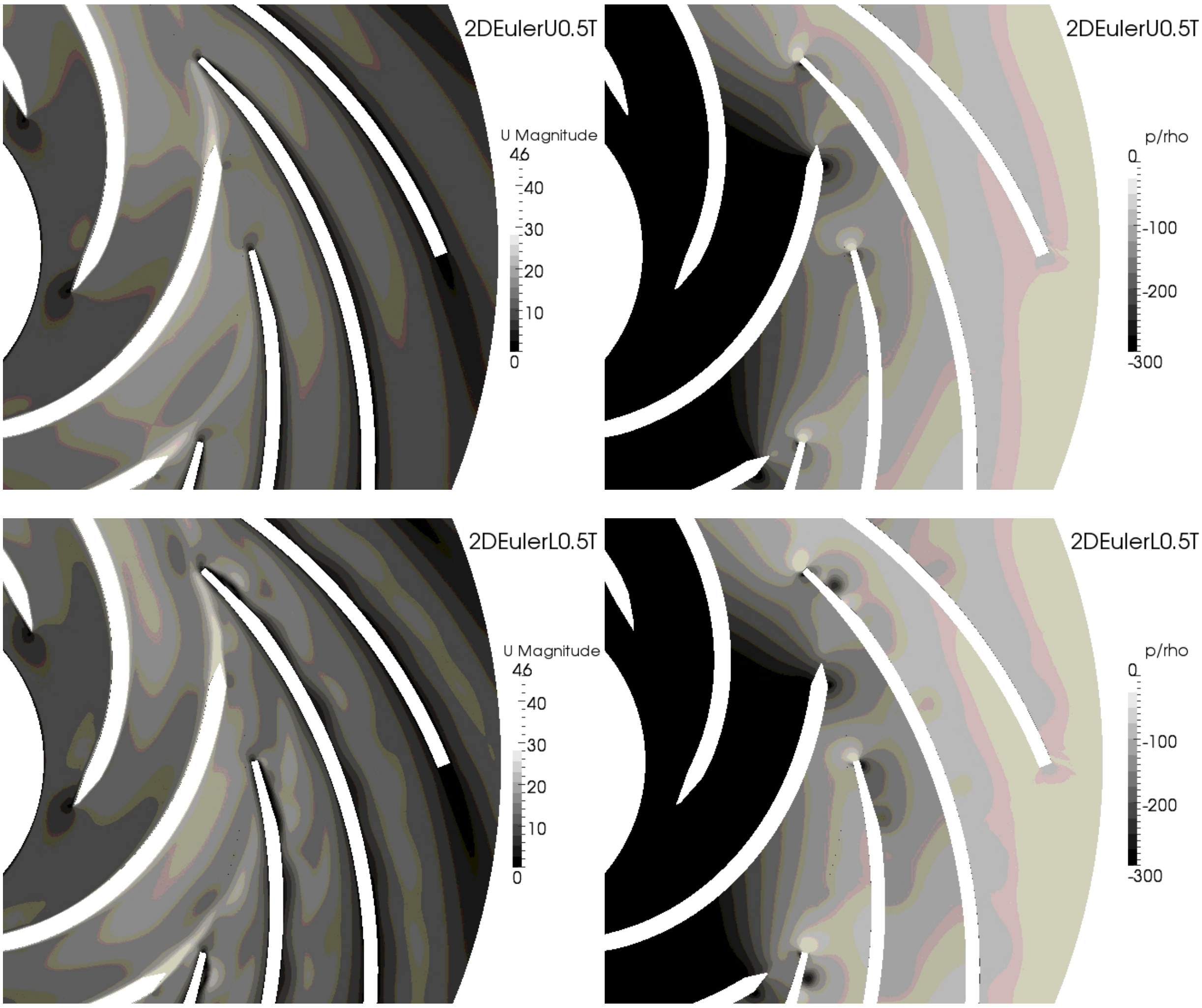


Figure 5.4: Relative velocity magnitude (left) and static pressure (right) for cases 2DEulerU0.5T (top) and 2DEulerL0.5T (bottom).

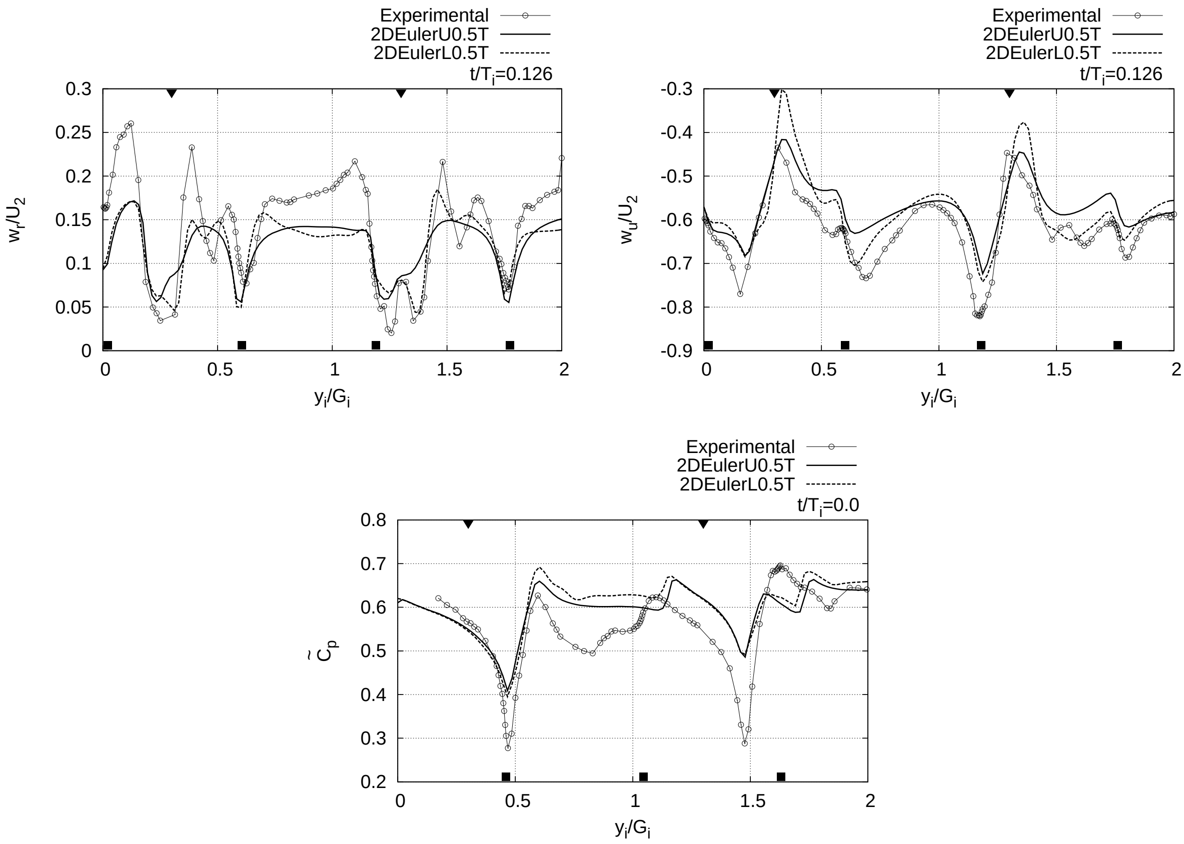


Figure 5.5: Radial (top left) and tangential (top right) velocities, and static pressure coefficient (bottom) for cases 2DEulerU0.5T and 2DEulerL0.5T.

Furthermore, the distributions of the radial and tangential relative velocities are compared with the experimental data in the gap between the rotor and stator, as shown in Fig.5.5. It is apparent that case 2DEulerL0.5T with the second-order linear upwind convection scheme has more accurately computed velocities than case 2DEulerU0.5T with the first-order upwind convection scheme. Both of the two cases show some similarity with the measured data as seen in Fig.5.5. However, the first-order upwind convection scheme does not predict the peaks of the velocities as the second-order linear upwind convection scheme does at the same position as in experimental data due to the principle of the upwind-biased estimation. It is clearly seen that case 2DEulerU0.5T has a more smooth curve than case 2DEulerL0.5T with the second-order linear upwind scheme, which is due to the fact that the first-order upwind convection scheme smears out the wakes behind the rotor blades.

Furthermore, the oscillations of the static pressure are influenced by the convection scheme as shown in Fig.5.6. It can be seen that case 2DEulerU0.5T cannot reach the same static pressure level as case 2DEulerL0.5T with the second-order linear upwind convection scheme observed in the three different points (Probe 1, 2 and 3 in Fig.2.9)

The instantaneous pictures of the ensemble-averaged static pressure coefficient ($\widetilde{C}_p$) for different radius (R_m/R_2 from 0.53333 to 1.02) are also investigated with respect to the experimental data as shown in Fig.5.7, which is looking in the other direction of the spanwise than in the other figures. The gradients of the static pressure coefficient in cases 2DEulerU0.5T and 2DEulerL0.5T are quite similar to each other but neither give perfect correspondence with the experimental result. One possible explanation to this is that the outlet boundary is too close in the numerical simulations.

Based on the above analysis of the second-order linear upwind and the first-order up-

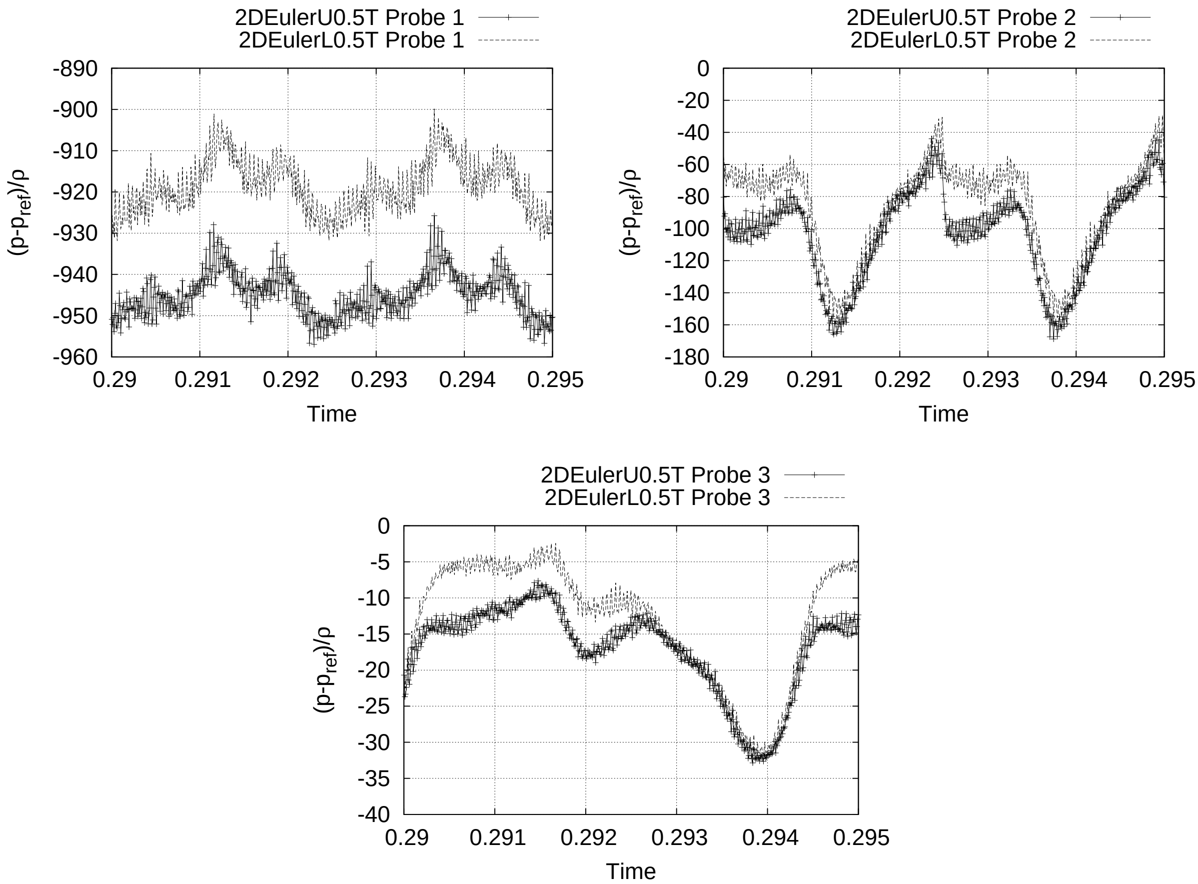
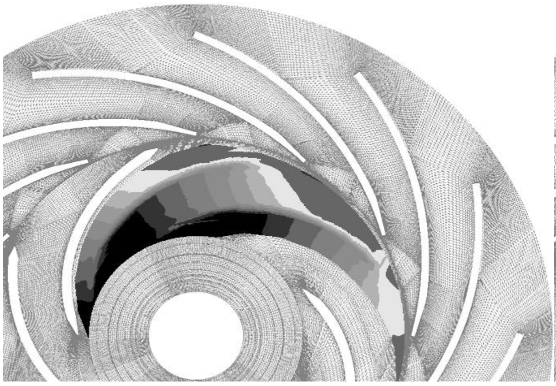
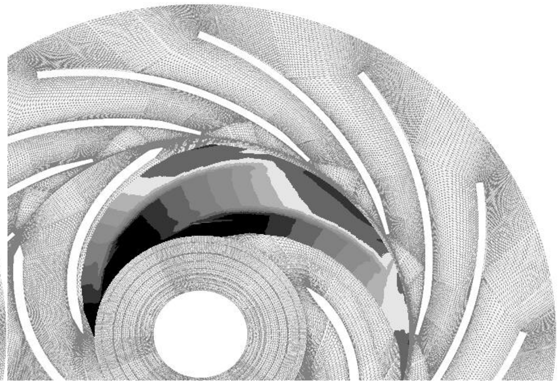


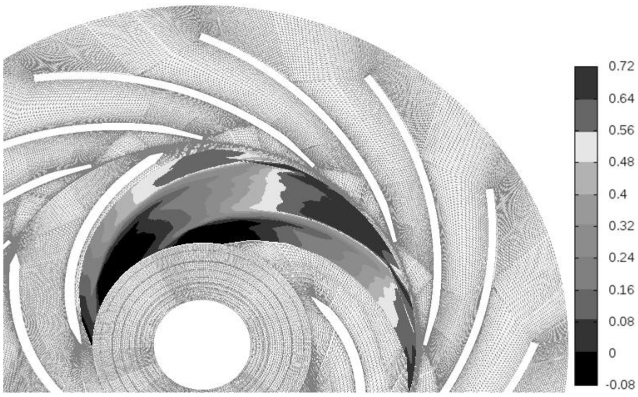
Figure 5.6: Oscillations of the static pressure at Probe 1 (top left), 2 (top right) and 3 (bottom) for cases 2DEulerU0.5T and 2DEulerL0.5T.



(a) 2DEulerU0.5T



(b) 2DEulerL0.5T



(c) Experimental

Figure 5.7: Static pressure coefficient ($\widetilde{C}_p$) distribution for cases 2DEulerU0.5T (top left), 2DEulerL0.5T (top right), and experimental data (bottom).

wind convection schemes, the second-order linear upwind convection scheme is considered to predict more accurately the flow unsteadiness in the ERCOFTAC centrifugal pump.

5.2.2 Comparison of time discretization schemes

Three temporal discretization schemes are used to predict the unsteady flow features in the ERCOFTAC centrifugal pump, i.e. Euler, backward and Crank-Nicholson. In the comparison between those schemes, a Crank-Nicholson coefficient of 0.5 has been used. The three results are very similar. The wakes in the diffuser region at time 0.3 s for case 2DBackL0.5T are shown in Fig.5.8, which represents the results of all three time discretization schemes.

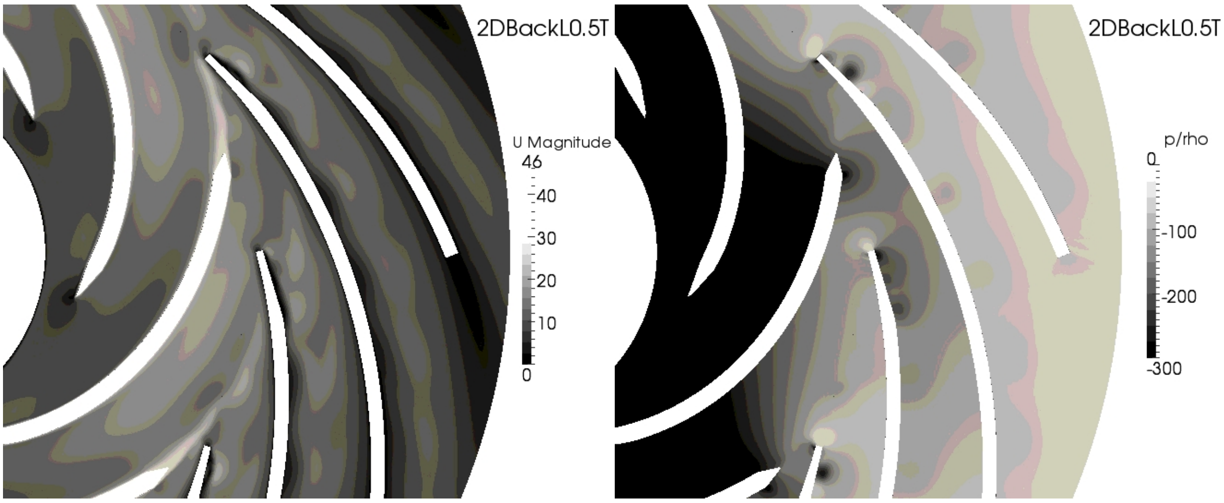


Figure 5.8: Relative velocity magnitude (left) and static pressure (right) for case 2DBackL0.5T.

The distributions of the velocity components and static pressure coefficient at the impeller outlet for the three cases compared with the experimental data are shown in Fig.5.9. The results for the three temporal discretization schemes are very close to each other, and the accuracy of them are reasonable but do not perfectly predict the unsteadiness of the flow. The predictions of radial and tangential velocities are different, the tangential velocities are over-predicted, but the radial velocities do under-predict the wakes of the rotor blades shown in Fig.5.9.

The distribution of the static pressure coefficient at the impeller outlet at midspan position is displayed in Fig.5.9. It shows good agreement with the experimental data except for some differences in the level, which is likely due to a wrong assumption of the static pressure in the suction pipe.

Pictures of the ensemble-averaged static pressure coefficient for two impeller blade passages are shown in Fig.5.10 for case 2DCN0.5L0.5T and the experimental results. Those show similar pressure coefficient distributions.

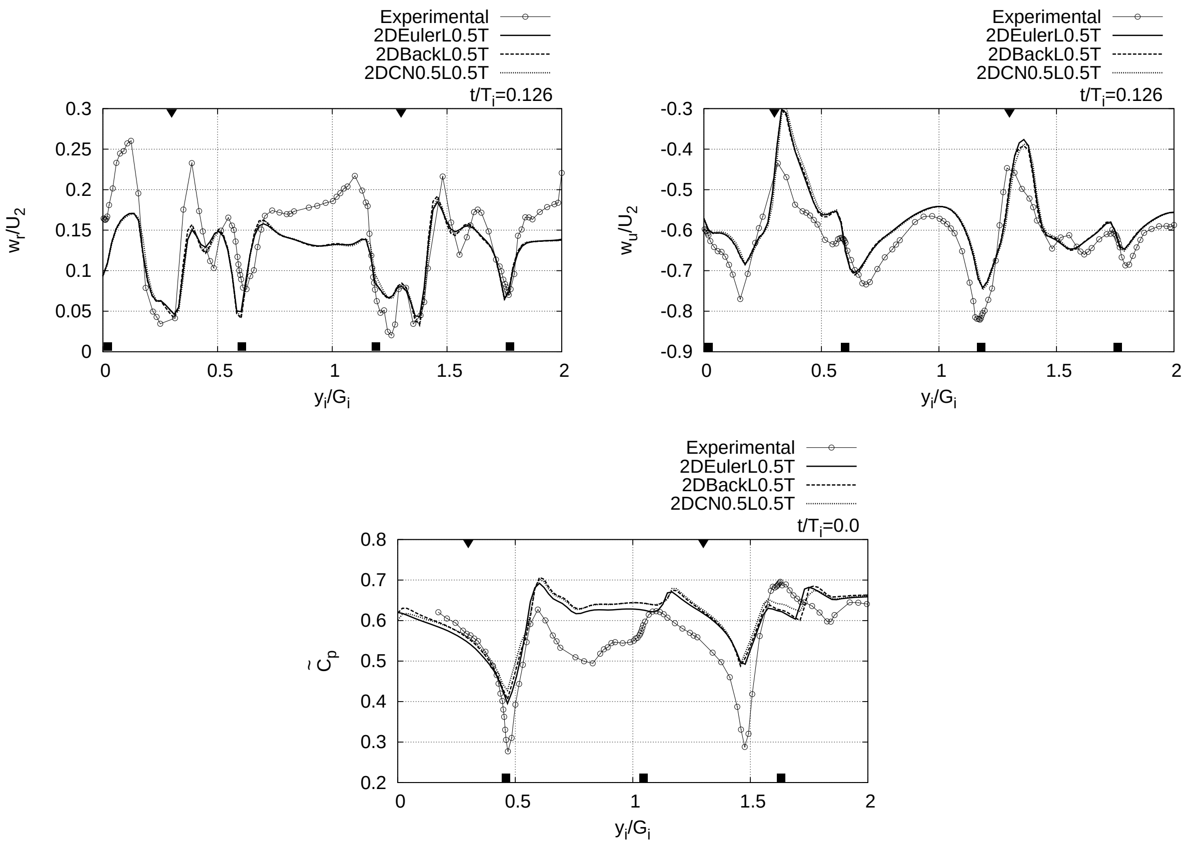
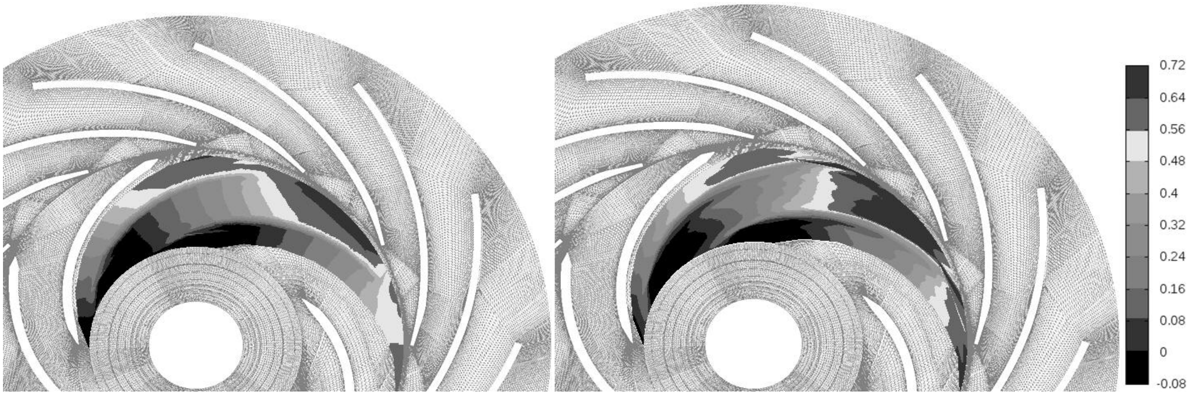


Figure 5.9: Radial (top left) and tangential (top right) velocity profile, and the static pressure coefficient (bottom) for cases 2DEulerL0.5T, 2DBackL0.5T and 2DCN0.5L0.5T.



(a) 2DBackL0.5T (b) Experimental

Figure 5.10: Static pressure coefficient for two impeller blade passages for case 2DBackL0.5T (left) and experimental (right).

Since the three temporal discretization schemes predict the unsteadiness of the flow similarly, the focus shifted towards the efficiency of the different discretization schemes, in term of computational time. The results shown in Tab.5.1. The three schemes are similar also with respect to the computational time.

Table 5.1: Computing time for cases 2DEulerL0.5T, 2DBackL0.5T and 2DCN0.5L0.5T.

Case	2DEulerL0.5T	2DBackL0.5T	2DCN0.5L0.5T
t=0 to t=0.3s	22.7 hours	22 hours	23.9 hours

5.2.3 Comparison of Crank-Nicholson time discretization scheme with different off-centering coefficients

To investigate the influence of the blending coefficient of the Crank-Nicholson scheme, four cases have been compared. They have all the same parameters except for the blending coefficient of Crank-Nicholson scheme, which are 0.2, 0.5, 0.8 and 1, respectively. It is found that case 2DCN1.0L0.5L with the pure Crank-Nicholson method (coefficient 1.0) crashed after running almost three laps, which reflects that the pure Crank-Nicholson scheme is unstable.

The distributions of the radial and tangential velocities, and the static pressure coefficient at the impeller outlet with respect to the measured data is shown in Fig.5.11. The under-prediction of the radial velocity and over-prediction of the tangential velocity still exist in Fig.5.11 no matter what the off-centering coefficient of Crank-Nicholson method is. Similarly, predicting the distribution of static pressure coefficient, Fig.5.11 shows that these three cases give correspondence with each other but not good enough to agree with the measured data.

However, the oscillation levels of the static pressure are quite different, which can be seen in Fig.5.12. Case 2DCN0.8L0.5T shows the highest level of oscillation, which shows that the results become more unstable as a pure Crank-Nicholson scheme is approached.

Finally, the computing time is listed in Tab.5.2 to investigate the efficiency of these three numerical solutions. There are no major differences considering that changing the blending coefficient does not effect the prediction of the flow features, and the needed computation time for each cases, the case 2DCN0.2L0.5T can be considered as the best case for this comparison.

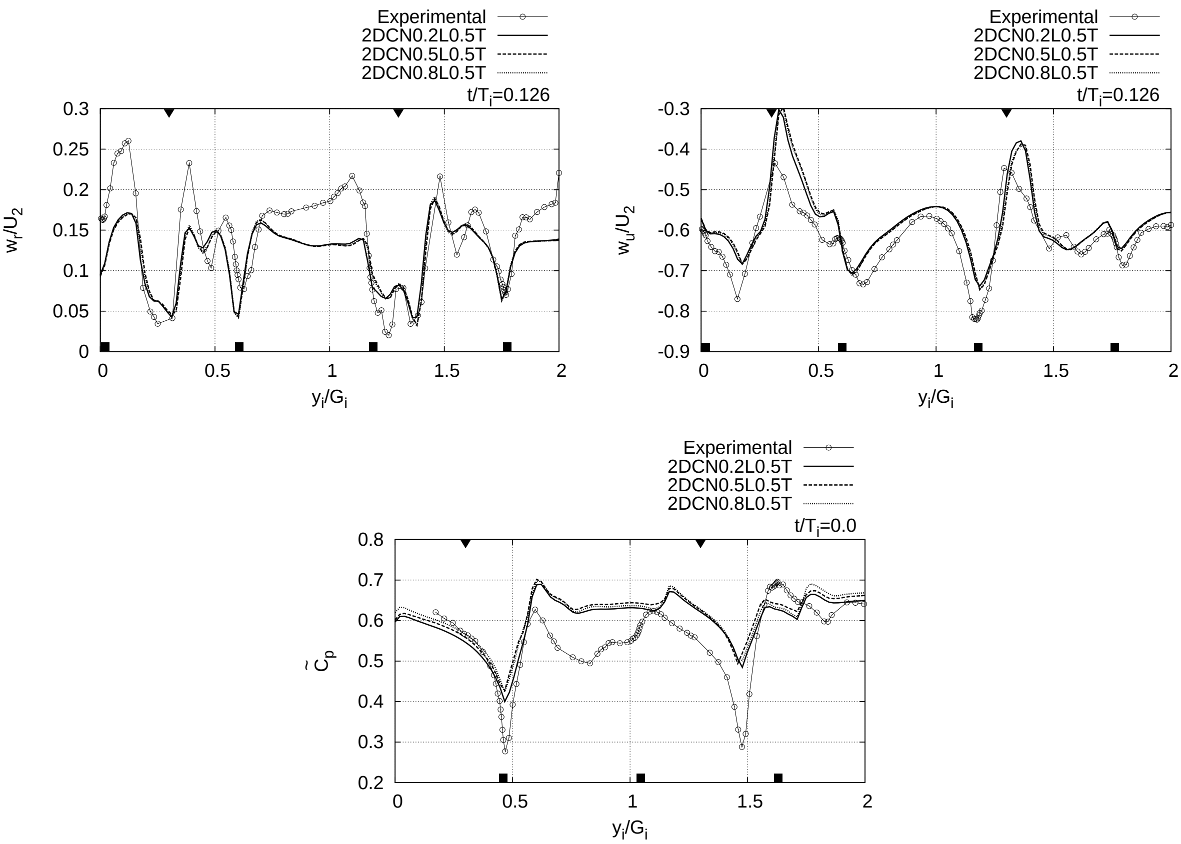


Figure 5.11: Radial (top left) and tangential (top right) velocities, and static pressure coefficient (bottom) for cases 2DCN0.2L0.5T, 2DCN0.5L0.5T and 2DCN0.8L0.5T.

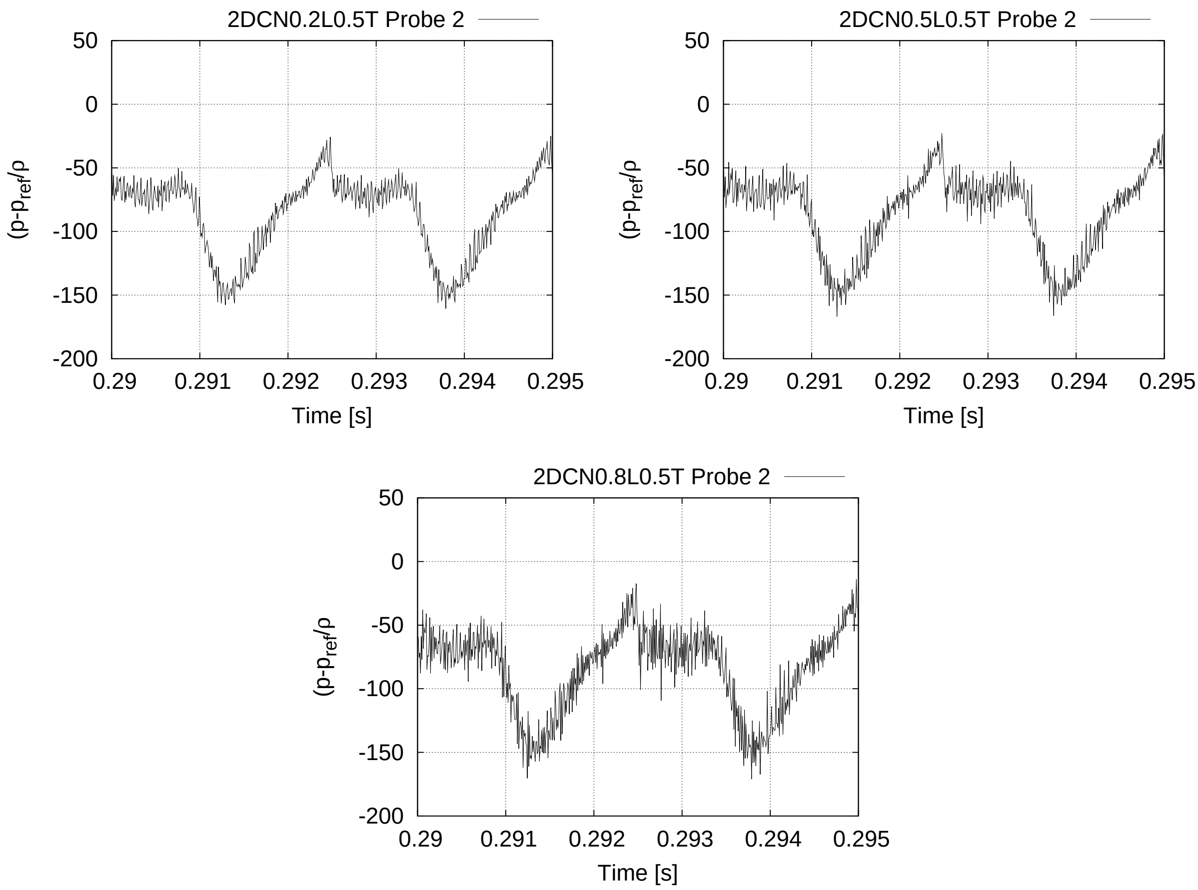


Figure 5.12: Oscillations of the static pressure at Probe 2 for cases 2DCN0.2L0.5T (top left), 2DCN0.5L0.5T (top right) and 2DCN0.8L0.5T (bottom).

Table 5.2: Computing time for cases 2DCN0.2L0.5T, 2DCN0.5L0.5T and 2DCN0.8L0.5T.

Case	2DCN0.2L0.5T	2DCN0.5L0.5T	2DCN0.8L0.5T
t=0 to t=0.3s	22.4 hours	23.9 hours	23.8 hours

5.2.4 Comparison of maximum Courant number

In OpenFOAM the time stepping can be chosen such that a maximum Courant number is preserved. In order to investigate the accuracy dependency on the size of the time-step, the flow was calculated with the Crank-Nicholson 0.5 temporal discretization and a maximum Courant number of 0.5, 1, 2 and 4, respectively.

It is found that the simulation crashes for Courant number larger than 4.

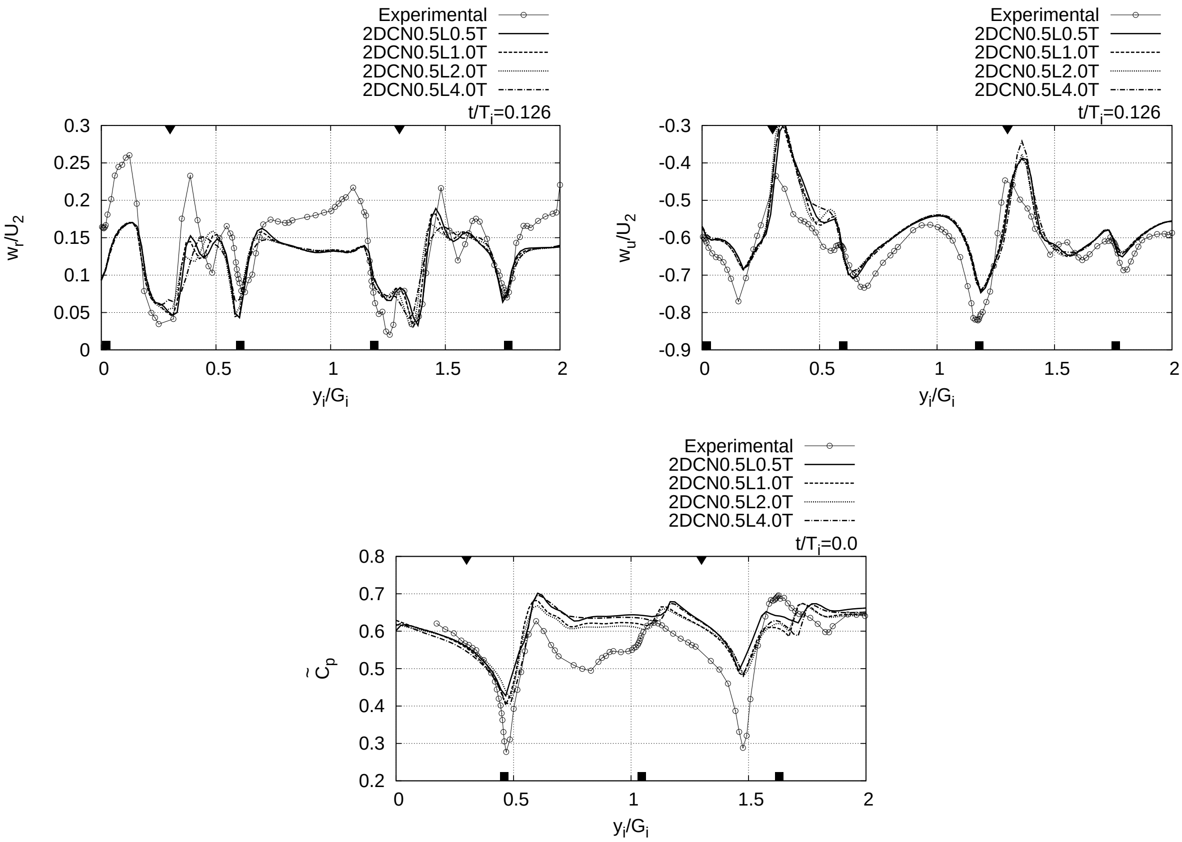


Figure 5.13: Radial (top left) and tangential (top right) velocities, and static pressure coefficient (bottom) for cases 2DCN0.5L0.5T, 2DCN0.5L1.0T, 2DCN0.5L2.0T and 2DCN0.5L4.0T.

Fig.5.13 shows that the distributions of the velocities and pressure coefficient at the impeller outlet are quite similar for maximum Courant number 0.5, 1, 2 and 4. It can be seen that the results smear out as the Courant number increases. Therefore, for good temporal accuracy, it is essential to keep the Courant number at a acceptable level.

The computational time for these four different cases are listed in Tab.5.3.

Table 5.3: Computing time for cases 2DCN0.5L0.5T, 2DCN0.5L1.0T, 2DCN0.5L2.0T and 2DCN0.5L4.0T.

Case	2DCN0.5L0.5T	2DCN0.5L1.0T	2DCN0.5L2.0T	2DCN0.5L4.0T
t=0 to t=0.3s	23.9 hours	11.7 hours	6.4 hours	3.5 hours

Considering the very different computational time for different maximum Courant number and the similar results these four numerical solutions gave, the 2DCN0.5L4.0T case is considered to be the most efficient.

5.2.5 Comparison of solvers

A new solver shared by Auvinen [23], named *transientSimpleDyMFoam*, is examined for the 2D unsteady simulation. The performance of this new solver is compared to the previous solver, *turbDyMFoam*. The wakes in the diffuser region are shown in Fig.5.14. The wakes predicted by the *transientSimpleDyMFoam* solver are more smeared out, while the complete wakes predicted by the *turbDyMFoam* solver reach the outlet. This can be seen by looking at the diffuser blade suction side boundary layer iso-line, as well as the pressure contours.

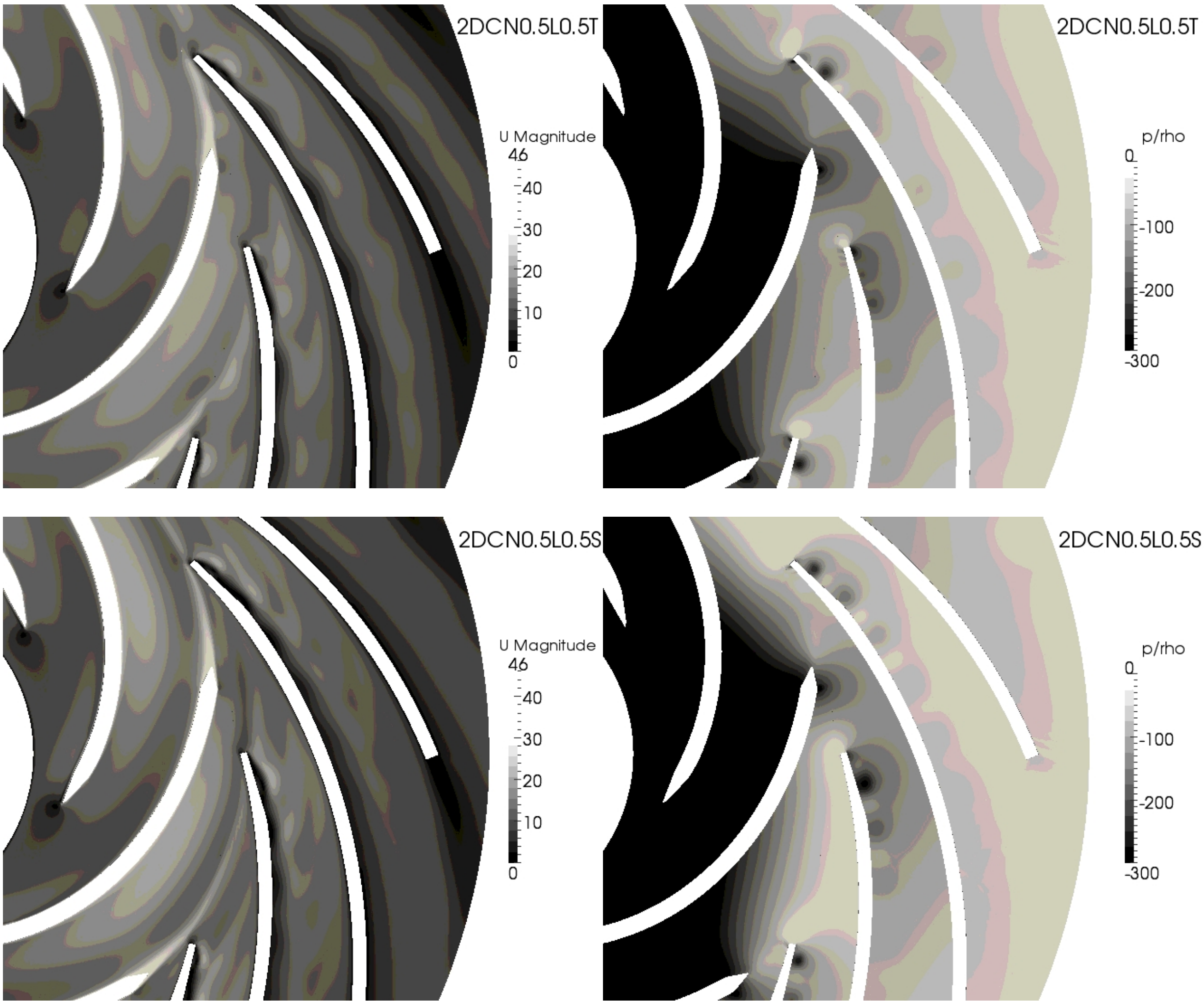


Figure 5.14: Relative velocity magnitude (left) and static pressure (right) for cases 2DCN0.5L0.5T (top) and 2DCN0.5L0.5S (bottom).

The distributions of the velocity components and the static pressure coefficient at the small gap between the rotor and stator are plotted in Fig.5.15. It shows that the *transientSimpleDyMFoam* solver does not predict as well as the *turbDyMFoam* solver, which is probably due to the turbulence model used for the unsteady simulation is not fitting the flow unsteadiness in the present work rather than the *transientSimpleDyMFoam* solver.

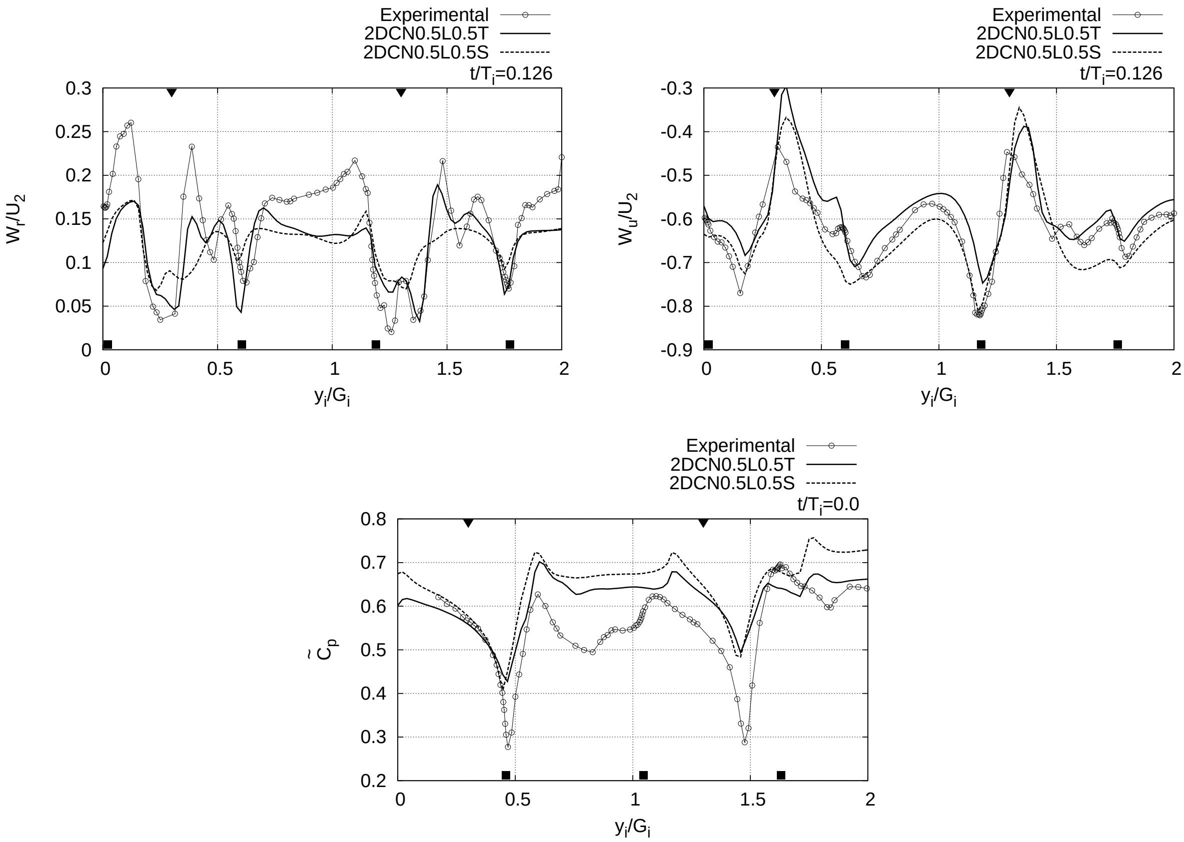


Figure 5.15: Radial (top left) and tangential (top right) velocities, and static pressure coefficient (bottom) for cases 2DCN0.5L0.5T and 2DCN0.5L0.5S.

5.3 3D steady-state simulation

In this section, a 3D representation of the geometry and the *simpleTurboMFRFoam* solver are used. The 3DSteady case was stopped after 7000 iterations, since all the residuals are below 10^{-5}. Using the same Frozen Rotor approach as in case 2DSteady, the position of the impeller and the diffuser are fixed to each other. The wakes in the diffuser region at the midspan position are shown in Fig.5.16, which is just a snapshot of the real flow in the pump. The computed velocities and static pressure coefficient at the impeller outlet at the midspan position can be compared with the experimental results, as shown in Fig.5.17. It is found that the 3DSteady case has the similar peak level for the radial and tangential velocities as the experimental results have, which is better than the under-prediction of the radial velocity and over-prediction of the tangential velocity in the previous 2D simulations.

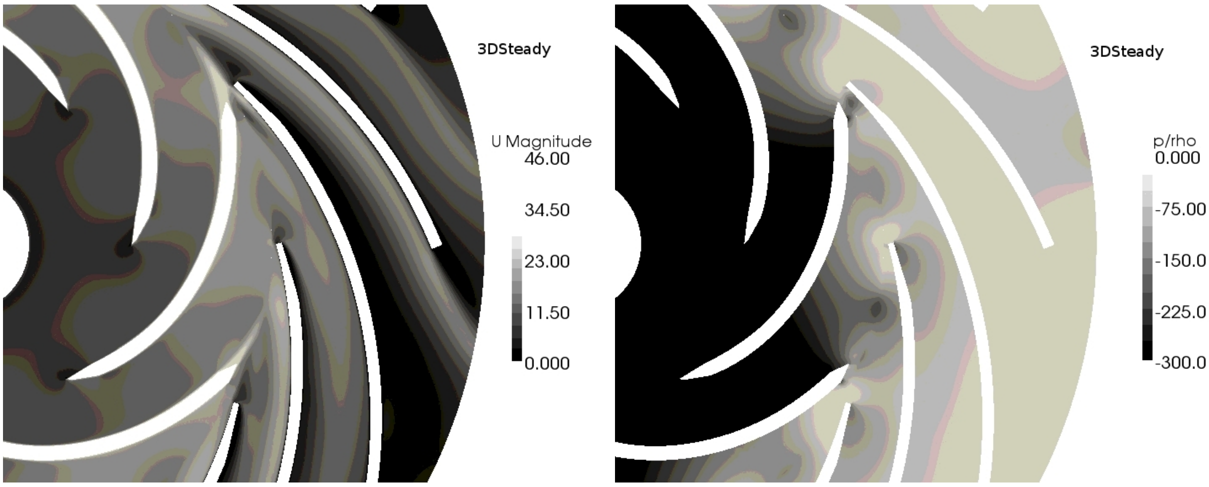


Figure 5.16: Relative velocity magnitude (left) and static pressure (right) at the midspan position for case 3DSteady.

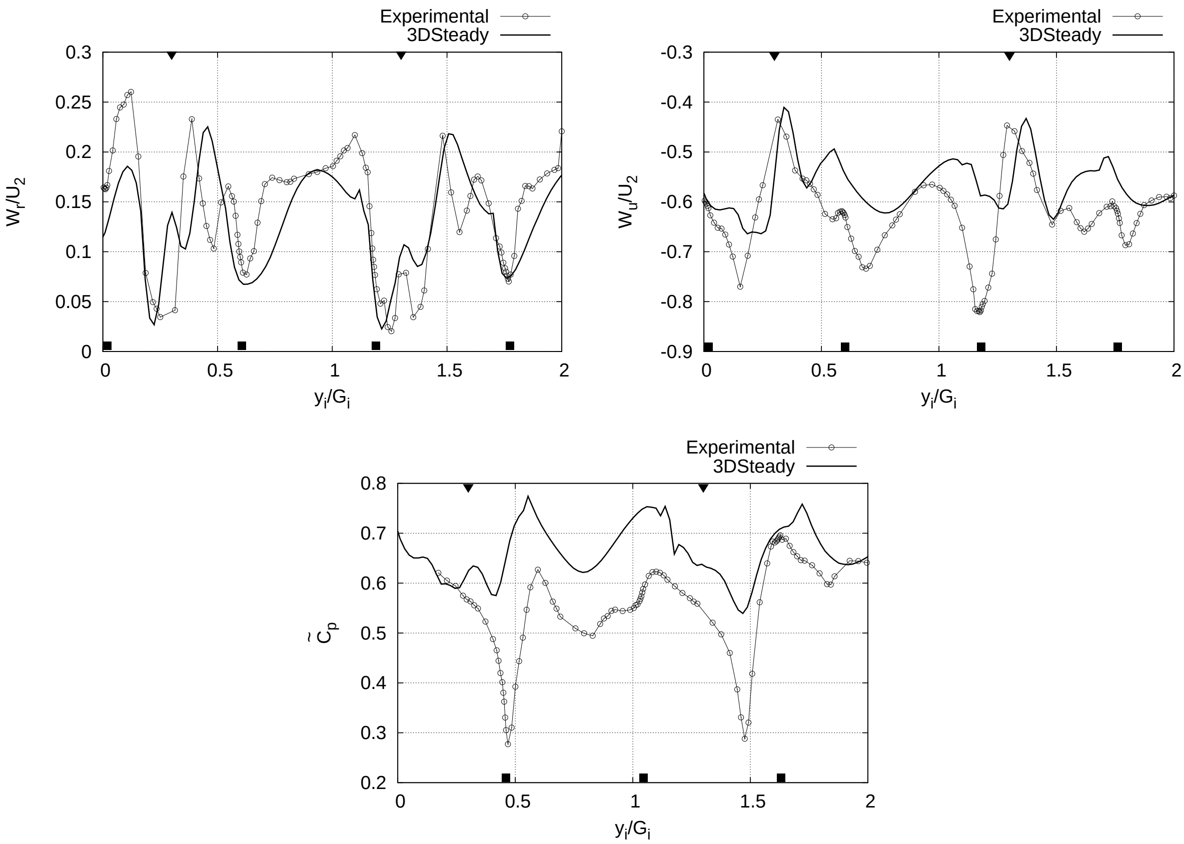
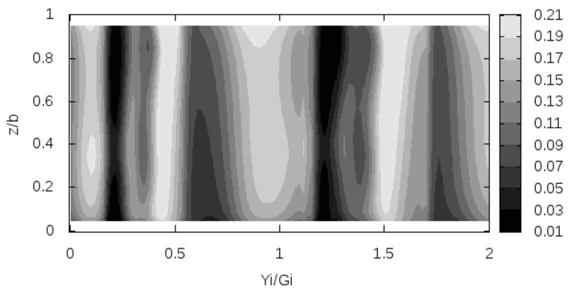
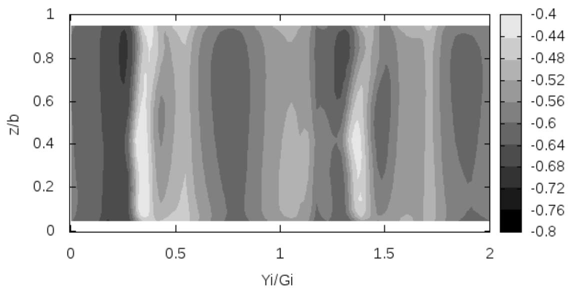


Figure 5.17: Radial (top left) and tangential (top right) velocities, and static pressure coefficient (bottom) for case 3DSteady and experimental results.

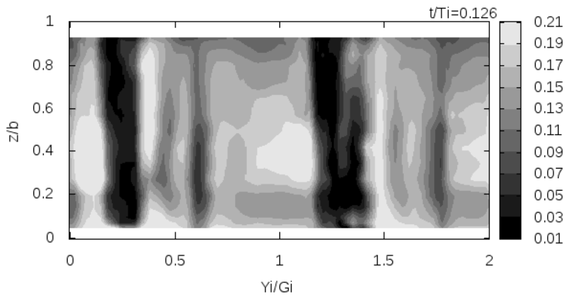
Furthermore, due to the 3D model has the real span thickness of the pump, the distributions of the velocities with respect to different span position at the impeller outlet can also be compared with the experimental data, as shown in Fig.5.18. Compared to the experimental results, the case 3DSteady has the similar value regions. But the iso-lines of the value are more smoother than the experimental one. Some similarity could be seen between the case 3DSteady and the experimental results but still not good enough, therefore, the following 3D unsteady simulation is expected to get better results.



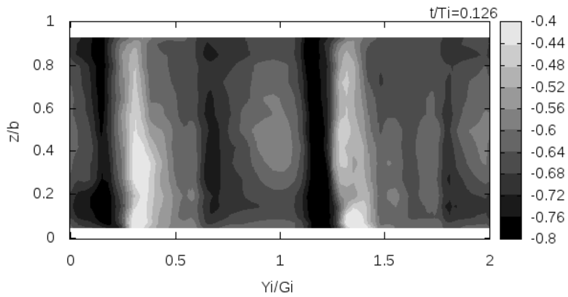
(a) Radial velocities for case 3DSteady.



(b) Tangential velocities for case 3DSteady.



(c) Radial velocities for experimental.



(d) Tangential velocities for experimental.

Figure 5.18: Radial (left) and tangential (right) velocities for case 3DSteady (top) and experimental (bottom).

5.4 3D unsteady simulation

The best parameters found for the 2D unsteady cases were used to predict the flow in the 3D model. Actually, the *turbDyMFoam* solver was used for the 3D unsteady simulation first of all. But it was found crashed many times and running extremely slow. Then the *transientSimpleDyMFoam* solver based on the SIMPLE algorithm is applied in case 3DBackL0.5S instead of the *turbDyMFoam* solver due to numerical stability problems, and the PCG for solving the velocity and pressure equations is used instead of the GAMG solver.

The relative velocity magnitude and static pressure at the midspan position of case 3DBackL0.5S are shown in Fig.5.19. The wakes can be seen clearly at the diffuser blade suction side boundary layer iso-lines, as well as at the pressure contours. It shows the possibility for the 3D unsteady simulation by the *transientSimpleDyMFoam* solver, and wake-prediction can be seen in both velocity magnitude and pressure coefficient contours, but not good enough. It probably means that the present k-ε turbulence model is not suitable for the *transientSimpleDyMFoam* solver. Other turbulence models therefore worth to try.

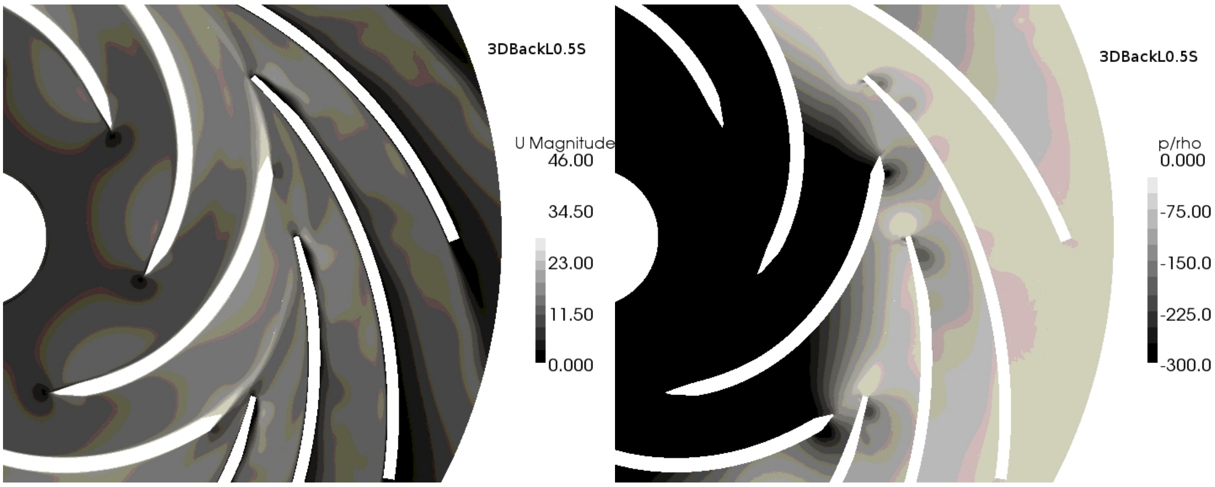


Figure 5.19: Relative velocity magnitude (left) and static pressure (right) at the midspan position for case 3DBackL0.5S.

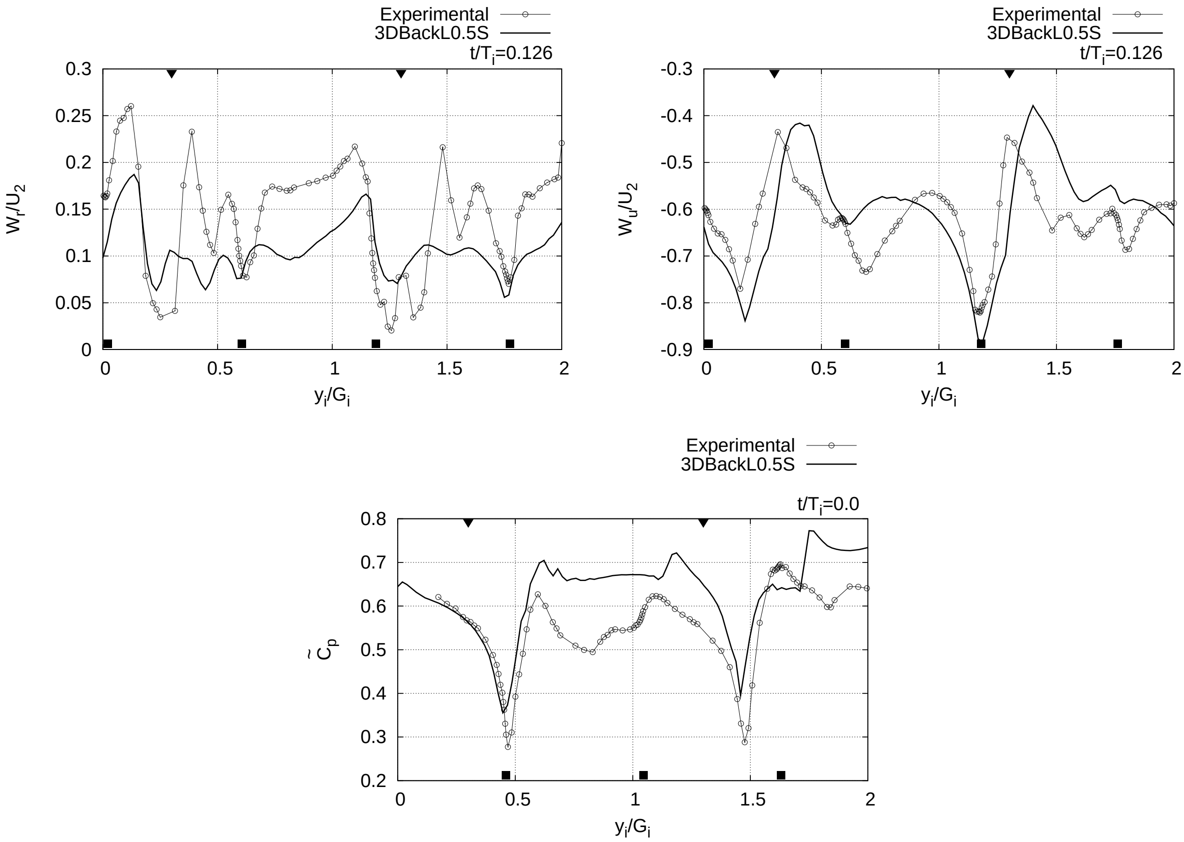
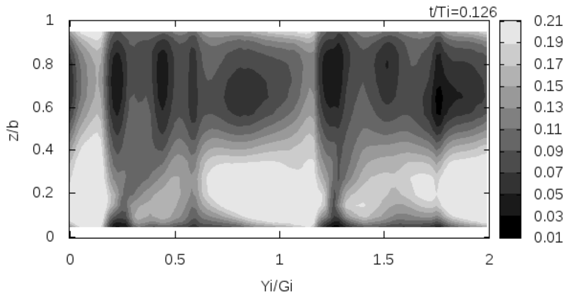
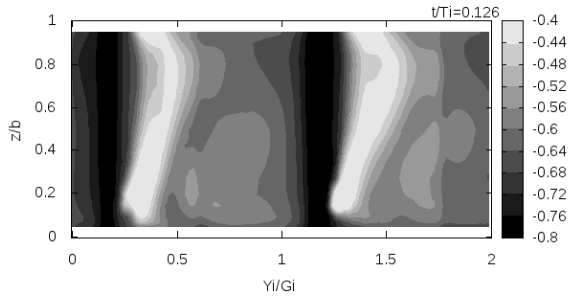


Figure 5.20: Radial (top left) and tangential (top right) velocities, and static pressure coefficient (bottom) at the midspan position for the case 3DBackL0.5S.

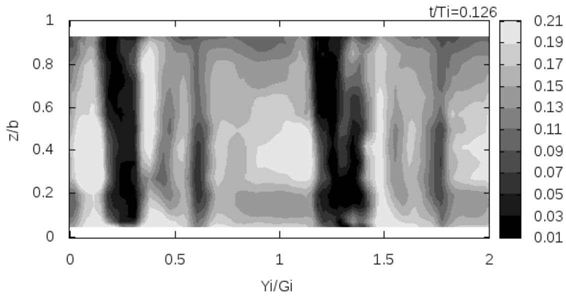
The distributions of the velocity components and static pressure coefficient for case 3DBackL0.5S are compared to the experimental results as shown in Fig.5.20, which are plotted at the small gap between the impeller blades and the diffusers at the midspan position. The radial velocity is still under-predicted, while the tangential velocity is predicted better than the over-prediction in the previous 2D unsteady cases. It is noticed that the wakes predicted by case 3DBackL0.5S is more smeared out both for the radial and tangential velocities, which means the *transientSimpleDyMFoam* solver needs more validations and testings in the future.



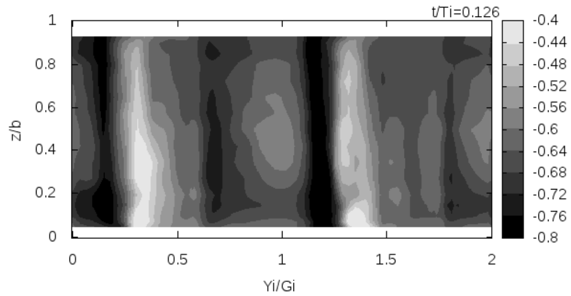
(a) Radial velocities for the case 3DBackL0.5S.



(b) Tangential velocities for the case 3DBackL0.5S.



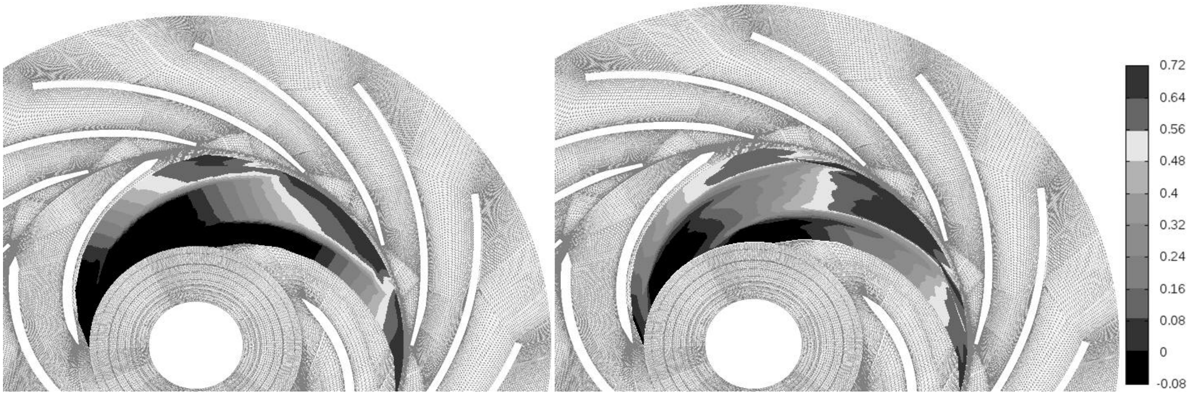
(c) Radial velocities for experimental.



(d) Tangential relative velocities for experimental.

Figure 5.21: Radial (left) and tangential (right) velocities for case 3DBackL0.5S (top) at the impeller outlet for the different spanwise positions, compared to the experimental (bottom).

The distributions of the radial and tangential vleocities with respect to the spanwise position are plotted and compared to the experimental data in Fig.5.21. Case 3DBackL0.5S has the similar contours and the similar value regions, compared to the experimental results. Since the wake-prediction are not good enough as discussed before, the 3D unsteady simulation needs to be tried by other numerical solutions. The distributions of the static pressure coefficient with respect to the radius are plotted and compared to the experimental data in Fig.5.22. The results of case 3DBackL0.5S are similar to the experimental reuslts. The gradient of the value and the position of the value regions are quite similar to each other, but case 3DBackL0.5S more smoother contours than the experimental one, which means the wake-prediction is more smeared out. The results in Figs.5.21 and Fig.5.22 in accordance with the investigation results of the wake-prediction in Fig.5.19.



(a) 3DBackL0.5S (b) Experimental

Figure 5.22: Static pressure coefficient at the midspan position for case 3DBackL0.5S (left) and experimental (right).

6 Conclusion

Numerical solutions of rotor-stator interaction using OpenFOAM-1.5-dev have been investigated in the ERCOFTAC Centrifugal Pump and compared with experimental results. Both steady-state simulations and unsteady simulations for 2D and 3D grid meshes have been performed. Good agreement has been shown with respect to the experimental data, although the upwind differencing scheme failed in preserving the wakes of the impeller blades in the diffuser vane passages. Furthermore, the unsteady simulations show better behavior of the wakes than the steady-state simulations. A series of comparison for different parameters have been performed, and the most efficient parameter were selected and underlined. Three different time discretization schemes, which are Euler, Backward and Crank-Nicholson, were found no much differences to each other, and were found some similarities compared with the experimental results. The stability problems were also investigated for the Crank-Nicholson time discretization scheme with different off-centering coefficients in the 2D unsteady cases. The SIMPLE-based *transientSimpleDyMFoam* solver was applied for the 2D unsteady simulation, which was found smeared the flows for the current k-ε turbulence model. On the other hand, the *turbDyMFoam* solver has difficulties to predict the flow in 3D, while the *transientSimpleDyMFoam* solver shows the possibility for the 3D unsteady simulation. Although a strong tendency of damping the flow, the *transientSimpleDyMFoam* solver proved to be a very promising stable solver, predicting accurately the unsteadiness of the flow in 3D cases. The wake was predicted much similar compared with the experimental results, but not perfect prediction, more validations and more testings therefore needs to be evaluated in the future.

7 Future work

In this work the Reynolds-Averaged Navier-Stokes (RANS) equations supplemented with the k-ε model were used to model the time-dependent turbulent flow in the ERCOFTAC Centrifugal Pump, and quite good results were achieved. However, there are other techniques for solving numerically the turbulent Navier-Stokes equations, such as Large-Eddy Simulation (LES). LES is computationally more expensive than RANS models, but could produce better results than RANS since the larger turbulent scales are explicitly resolved. Therefore, the LES approach with OpenFOAM should be evaluated in the future. An intermediate step would be to apply some Detached Eddy Simulation (DES) models, which is a mix of RANS and LES.

The k-ε SST (Shear Stress Transport) model is an eddy-viscosity model that is also worth to be evaluated. It is a combination of the k-ε model (in the outer region of and outside of the boundary layer) and the k-ω model (in the inner boundary layer). The k-ε model has weakness on its over-prediction of the shear stress in adverse pressure gradient flows, while the k-ω model is better at adverse pressure gradient flow. However, the standard k-ω model has the disadvantage of dependent on the free-stream value of ω. It shows some improvements, to some extent, to combine the two models [24]. Further, Gyllenram [25] developed a filtering technique for the k-ω SST method, moving it into the DES framework. That model has been implemented in OpenFOAM [26] and should be evaluated also for the ERCOFTAC Centrifugal Pump.

References

[1] H. Nilsson, M. Page, M. Beaudoin, B. Gschaider, and H. Jasak, "The OpenFOAM Turbomachinery Working Group, and Conclusions from the Turbomachinery Session of the Third OpenFOAM Workshop," in *24th IAHR Symposium on Hydraulic Machinery and Systems*, (Foz Do Iguassu, Brazil), October 27-31 2008.

[2] J.-F. Combès, "Test Case U3: Centrifugal Pump with a Vaned Diffuser," in *ERCOFTAC Seminar and Workshop on Turbomachinery Flow Prediction VII*, (Aussois), Jan 4-7 1999.

[3] O. Petit, M. Page, M. Beaudoin, and H. Nilsson, "The ERCOFTAC centrifugal pump OpenFOAM case-study," in *3rd IAHR International Meeting of the Workgroup on Cavitation and Dynamic Problems in Hydraulic Machinery and Systems*, (Brno, Czech Republic), October 14-16 2009.

[4] M. Ubaldi, P. Zunino, G. Barigozzi, and A. Cattanei, "An Experimental Investigation of Stator Induced Unsteadiness on Centrifugal Impeller Outflow," *Journal of Turbomachinery*, vol. 118, pp. 41–54, 1996.

[5] S. L. Dixon and B. Eng., *Fluid mechanics and thermodynamics of turbomachinery.* Elsevier, 5th ed., 2008.

[6] L. Reti, "Francesco di Giorgio (Armani) Martini's Treatise on Engineering and Its Plagiarists," *Technology and Culture*, vol. 4, pp. 287–298, summer 1963.

[7] M. Beaudoin and H. Jasak, "Development of a Generalized Grid Interface for Turbomachinery simulations with OpenFOAM," in *Open Source CFD International Conference*, 2008.

[8] M. Ubaldi, P. Zunino, G. Barigozzi, and A. Ghiglione, "LDV Investigation of the Rotor-Stator Aerodynamic Interaction in a Centrifugal Turbomachine," in *8th International Symposium on Applications of Laser Techniques to Fluid Mechanics*, (Lisbon), 1996.

[9] E. Canepa, A. cattanei, M. Ubaldi, and P. Zunino, "Wake-boundary layer interaction on the vaned diffuser of a centrifugal stage," *Proc. IMechE*, vol. 219 Part A: J. Power and Energy, pp. 401–411.

[10] M. Ubaldi, P. Zunino, and A. Ghiglione, "Detailed flow measurements within the impeller and the vaneless diffuser of a centrifugal turbomachine," *Experimental Thermal and Fluid Science*, vol. 17, pp. 147–155, 1998.

[11] P.-F. Bert, J.-F. Combès, and J.-L. Kueny, "Unsteady Flow Calculation in a Centrifugal Pump Using a Finite Element Method," in *Proceedings of the XVIII IAHR Symposium on Hydraulic Machinery and Cavitation*, pp. 371–380, 1996.

[12] J.-F. Combès, P.-F. Bert, and J.-L. Kueny, "Numerical Investigation of the Rotor-Stator Interaction in a Centrifugal Pump Using a Finite Element Method," in *Proceedings of the 1997 ASME Fluids Engineering Division Summer Meeting*, FEDSM97-3454, 1997.

[13] E. Torbergsen and M. F. White, "Transient Simulation of Impeller/Diffuser Interactions," in *Proceedings of the 1997 ASME Fluids Engineering Division Summer Meeting*, FEDSM97-3453, 1997.

[14] K. Sato, *Blade Row Interaction in Radial Turbomachines.* PhD thesis, Durham University, 1999.

[15] K. Sato and L. He, "Numerical Solution of Incompressible Unsteady Flows in Turbomachinery," in *Proceedings of the 3rd ASME/JSME Joint Fluids Engineering Conference*, FEDSM99-6871, (San Francisco), 1999.

[16] K. Sato and L. He, "Numerical investigation into the effects of a radial gap on hydraulic turbine performance," *Proceedings of the Institution of Mechanical Engineers*, vol. 215 Part A, pp. 99–107, 2001.

[17] M. Page, É. Théroux, and J.-Y. Trépanier, "Unsteady rotor-stator analysis of a Francis turbine," in *22nd IAHR Symposium on Hydraulic Machinery and Systems*, (Stockholm, Sweden), June 29 - July 2 2004.

[18] M. Page and M. Beaudoin, "Adapting OpenFOAM for Turbomachinery Applications," in *Second OpenFOAM Workshop*, (Zagreb), June 7-9 2007.

[19] H. Versteeg and W. Malalasekra, *An introduction to computational fluid dynamics.* Pearson Prentice Hall, 2nd edition ed., 2007.

[20] H. Jasak, *Error analysis and estimation for the finite volume method with applications to fluid flow.* PhD thesis, Imperical college, University of London, 1996.

[21] S. V. Patankar, *Numerical heat transfer and fluid flow.* Hemisphere Publishing Corporation, 1981.

[22] R. Issa, "Solution of the implicitly discretized fluid flow equations by operator-splitting," *Journal of Computational Physics*, vol. 62, pp. 40–65, January 1986.

[23] M. Auvinen, J. Ala-Juusela, N. Pedersen, and T. Siikonen, "Time-Accurate Turbomachinery Simulations with Open-Source CFD; Flow Analysis of A Single-Channel Pump with OpenFOAM." to be published on V European Conference on Computational Fluid Dynamics ECCOMAS CFD 2010, June 14-17 June 2010.

[24] L. Davidson, *The SST Model.* http://www.tfd.chalmers.se/doct/comp_turb_model.

[25] W. Gyllenram, *Analytical and Numerical Studies of Internal Swirling Flows.* PhD thesis, Division of Fluid Dynamics, Chalmers University of Technology, 2008.

[26] H. Nilsson, "The Dellenback Abrupt Expansion OpenFOAM Case-Study, and the kOmegaSSTF turbulence model.," in *Fourth OpenFOAM Workshop*, (Montreal, Canada), June 1-4 2009.

8 Appendix

Tab.8.1 shows all the simulated cases.

Table 8.1: Description of the different cases and **BOLD** text build up the case names.

Name of case	Simulation type	Time scheme	convection scheme	maxCo	solver
2DSteady	**2D steady-**state simulation	-	linearUpwind	-	simpleTurbo-MFRFoam
2DEulerU0.5T	**2D** unsteady simulation	**Euler**	**u**pwind	**0.5**	**t**urbDyMFoam
2DEulerL0.5T	**2D** unsteady simulation	**Euler**	**l**inearUpwind	**0.5**	**t**urbDyMFoam
2DBackL0.5T	**2D** unsteady simulation	**Back**ward	**l**inearUpwind	**0.5**	**t**urbDyMFoam
2DCN0.2L0.5T	**2D** unsteady simulation	**C**rank-**N**icholson **0.2**	**l**inearUpwind	**0.5**	**t**urbDyMFoam
2DCN0.5L0.5T	**2D** unsteady simulation	**C**rank-**N**icholson **0.5**	**l**inearUpwind	**0.5**	**t**urbDyMFoam
2DCN0.8L0.5T	**2D** unsteady simulation	**C**rank-**N**icholson **0.8**	**l**inearUpwind	**0.5**	**t**urbDyMFoam
2DCN1.0L0.5T	**2D** unsteady simulation	**C**rank-**N**icholson **1.0**	**l**inearUpwind	**0.5**	**t**urbDyMFoam
2DCN0.5L1.0T	**2D** unsteady simulation	**C**rank-**N**icholson **0.5**	**l**inearUpwind	**1.0**	**t**urbDyMFoam
2DCN0.5L2.0T	**2D** unsteady simulation	**C**rank-**N**icholson **0.5**	**l**inearUpwind	**2.0**	**t**urbDyMFoam
2DCN0.5L4.0T	**2D** unsteady simulation	**C**rank-**N**icholson **0.5**	**l**inearUpwind	**4.0**	**t**urbDyMFoam
2DCN0.5L0.5S	**2D** unsteady simulation	**C**rank-**N**icholson **0.5**	**l**inearUpwind	**0.5**	transient**S**im-pleDyMFoam
3DSteady	**3D steady-**state simulation	-	linearUpwind	-	simpleTurbo-MFRFoam
3DBackL0.5S	**3D** unsteady simulation	**Back**ward	**l**inearUpwind	**0.5**	transient**S**im-pleDyMFoam

8.1 2D steady-state simulation results

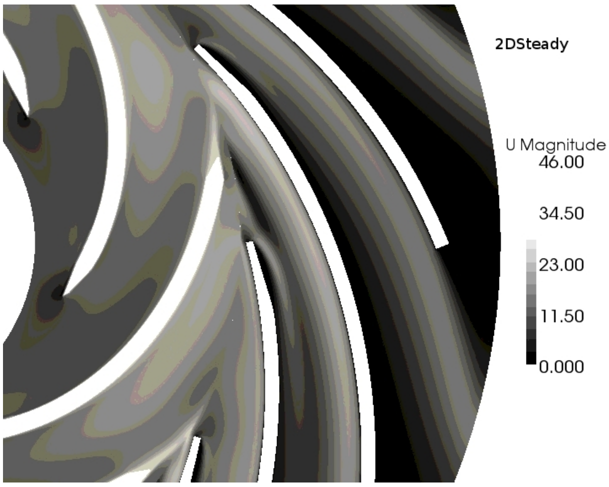


Figure 8.1: Relative velocity magnitude (left) and static pressure (right) for case 2DSteady.

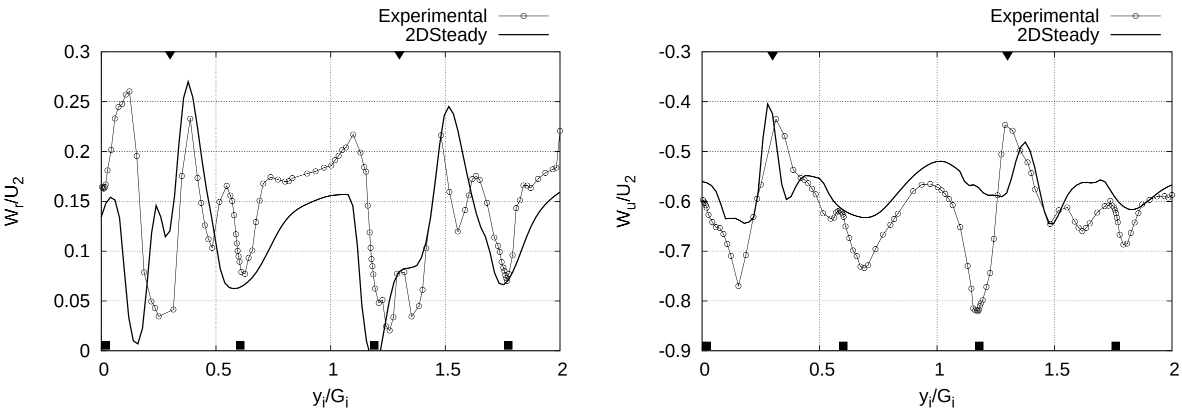


Figure 8.2: Radial (left) and tangential (right) velocities at the impeller outlet for case 2DSteady.

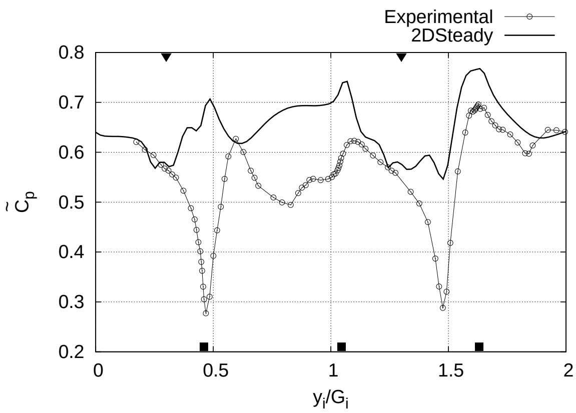


Figure 8.3: Static pressure coefficient at the impeller outlet for case 2DSteady.

8.2 2D unsteady simulation results

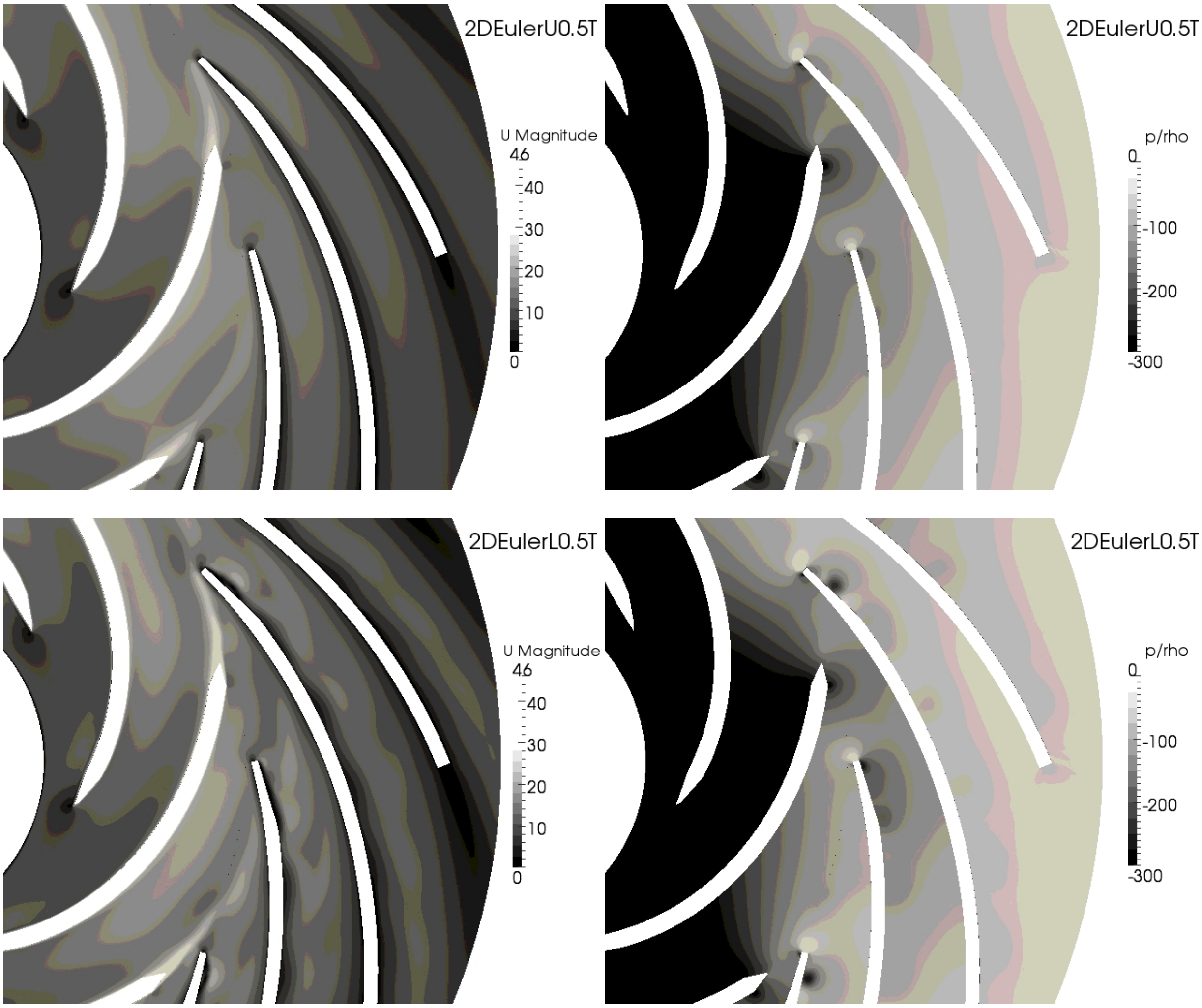


Figure 8.4: Relative velocity magnitude (left) and static pressure (right) for cases 2DEulerU0.5T (top) and 2DEulerL0.5T (bottom).

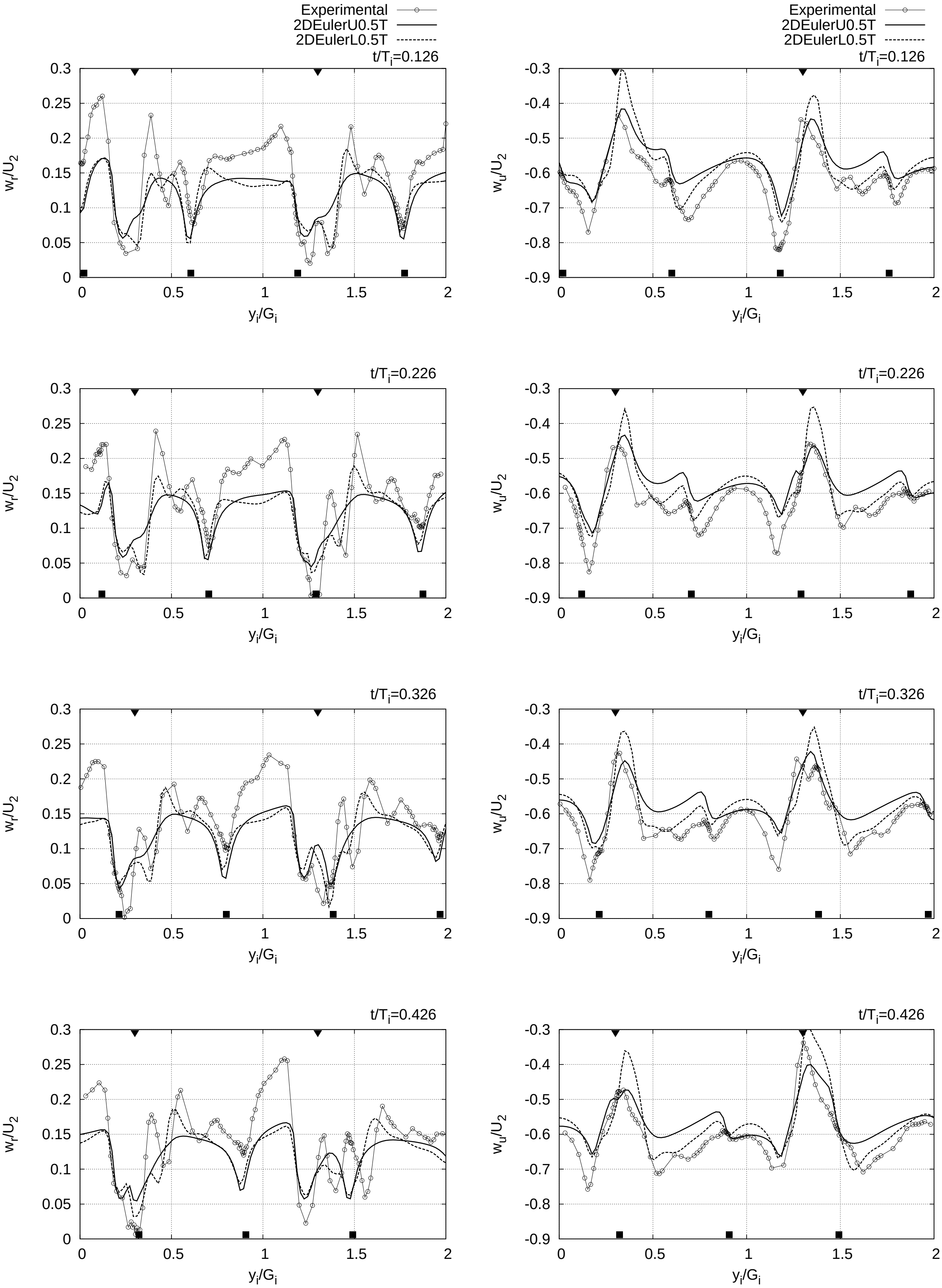


Figure 8.5: Radial (left) and tangential (right) velocities for cases 2DEulerU0.5T and 2DEulerL0.5T.

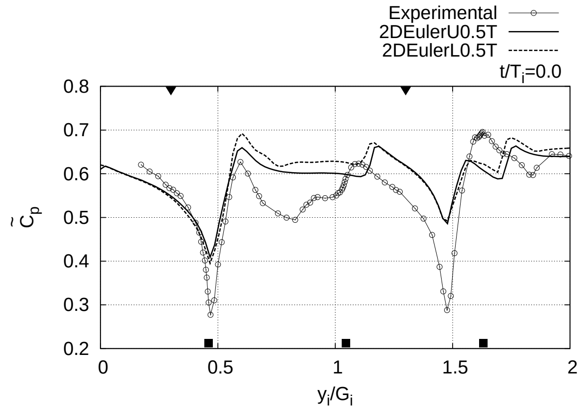


Figure 8.6: Static pressure coefficient at the impeller outlet for cases 2DEulerU0.5T and 2DEulerL0.5T.

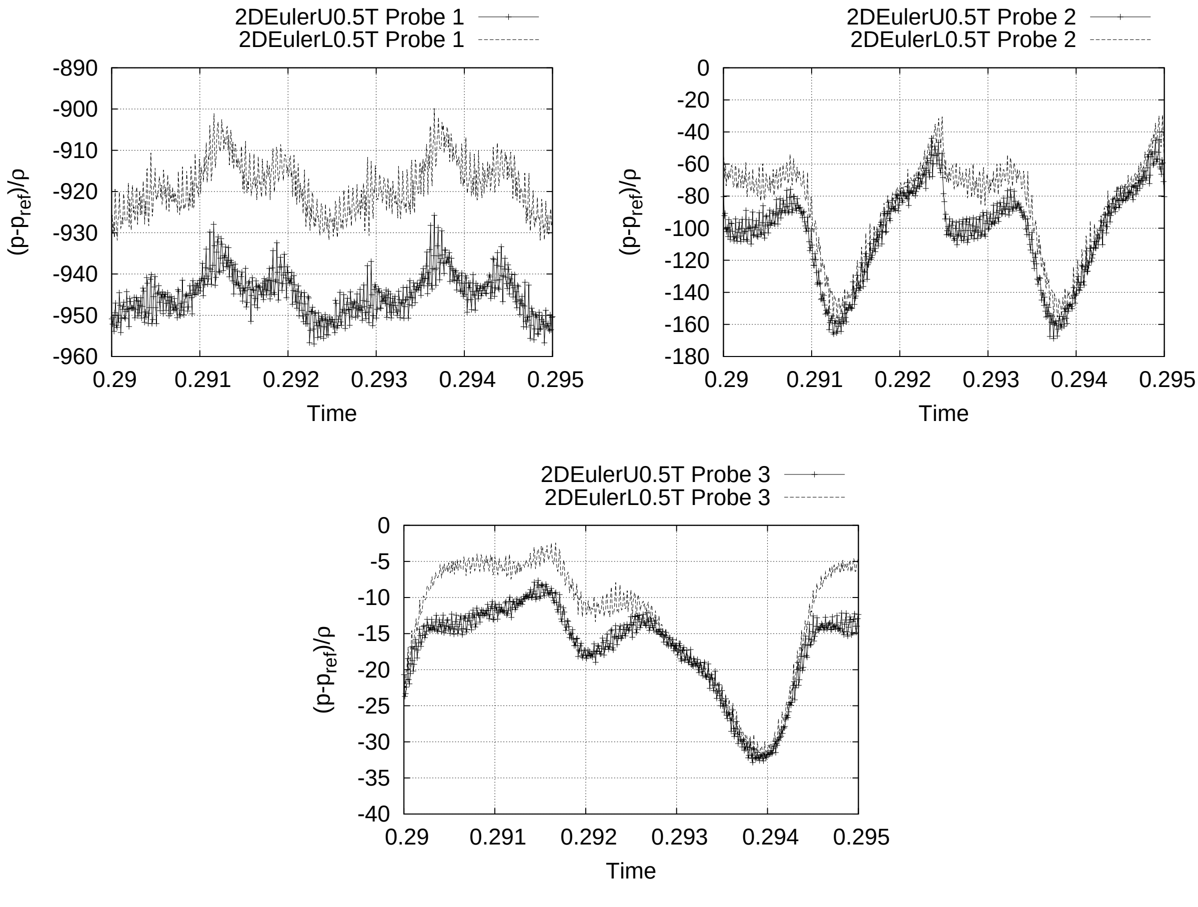
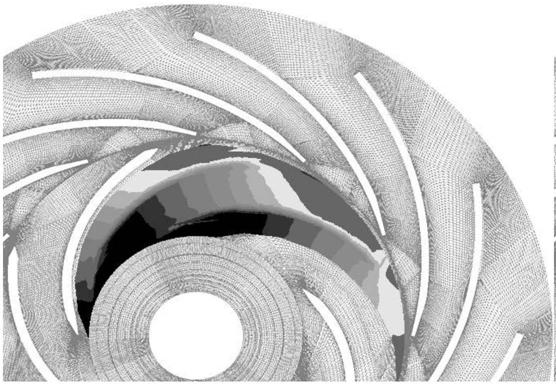
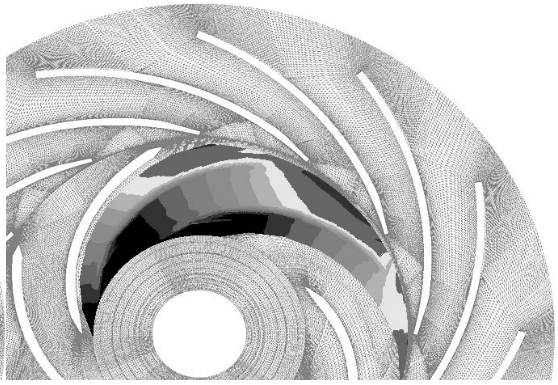


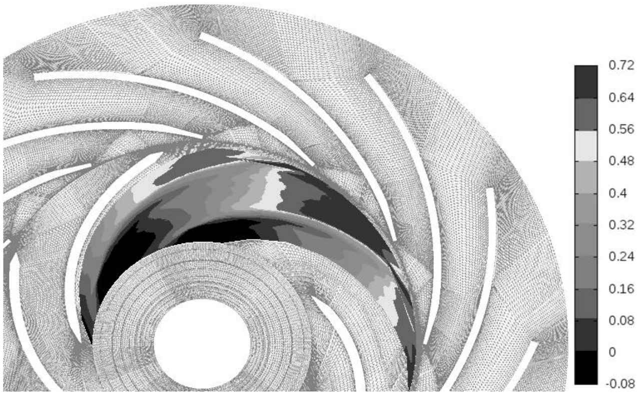
Figure 8.7: Oscillations of the static pressure at Probe 1 (top left), 2 (top right) and 3 (bottom) for cases 2DEulerU0.5T and 2DEulerL0.5T.



(a) 2DEulerU0.5T



(b) 2DEulerL0.5T



(c) Experimental

Figure 8.8: Instantaneous pictures of the static pressure coefficient at the front end of the impeller for cases 2DEulerU0.5T (top left), 2DEulerL0.5T (top right) and experimental (bottom).

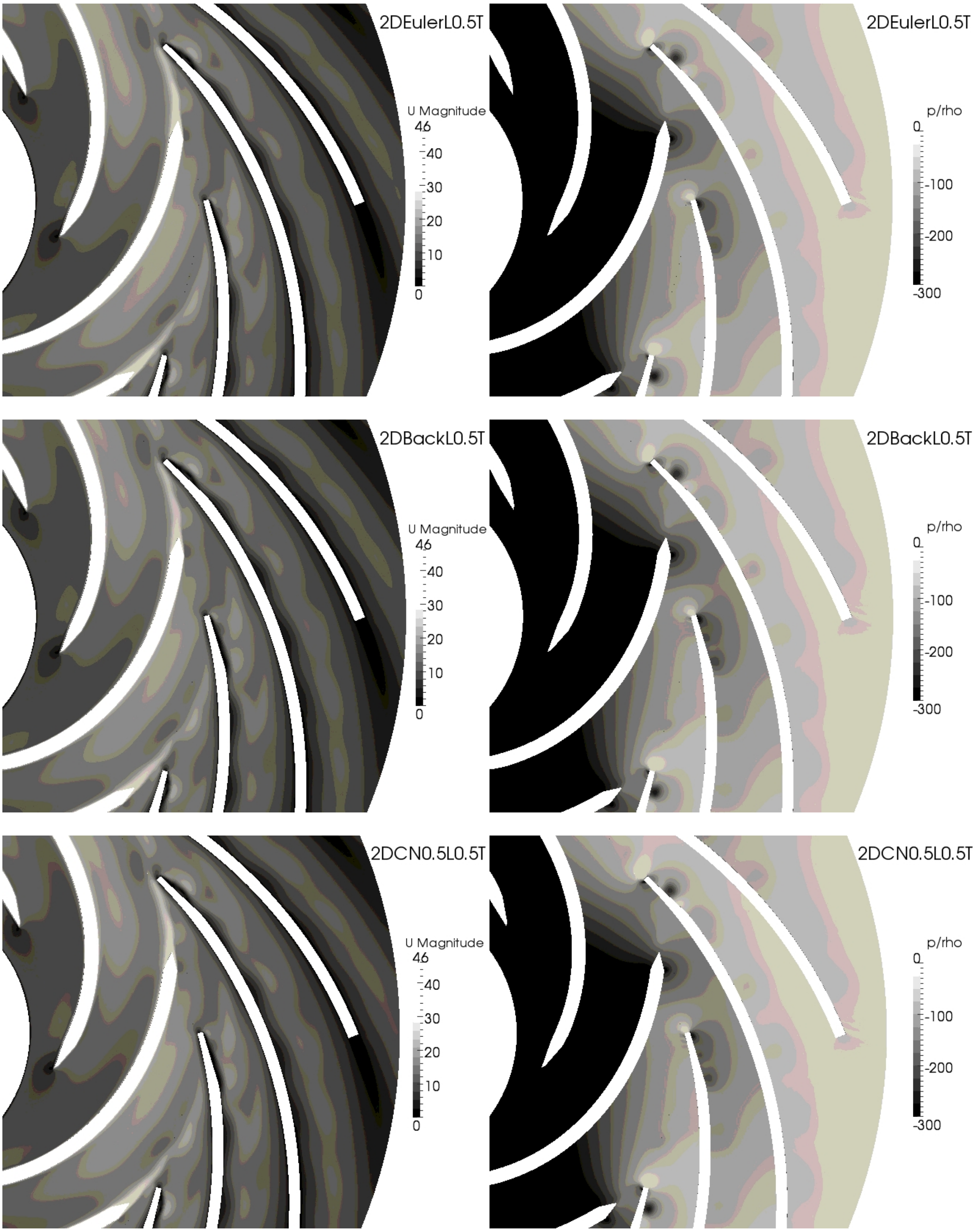


Figure 8.9: Relative velocity magnitude (left) and static pressure (right) for cases 2DEulerL0.5T (top), 2DBackL0.5T (middle) and 2DCN0.5L0.5T (bottom).

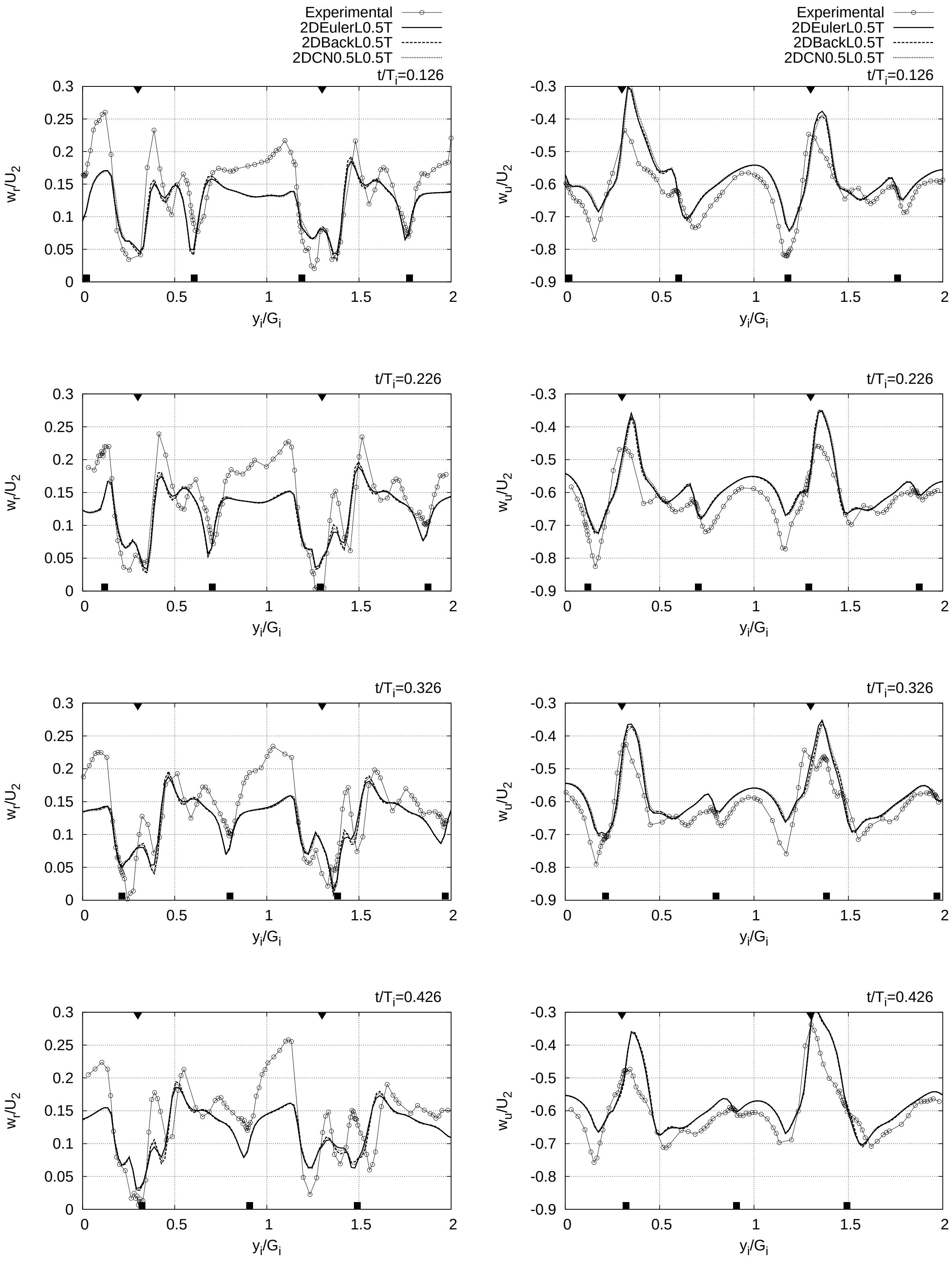


Figure 8.10: Radial (left) and tangential (right) velocities for cases 2DEulerL0.5T, 2DBackL0.5T and 2DCN0.5L0.5T.

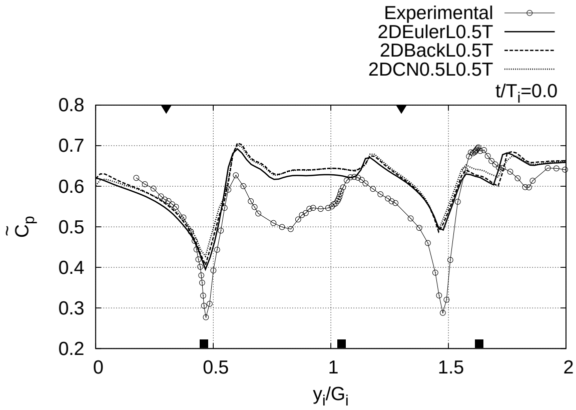
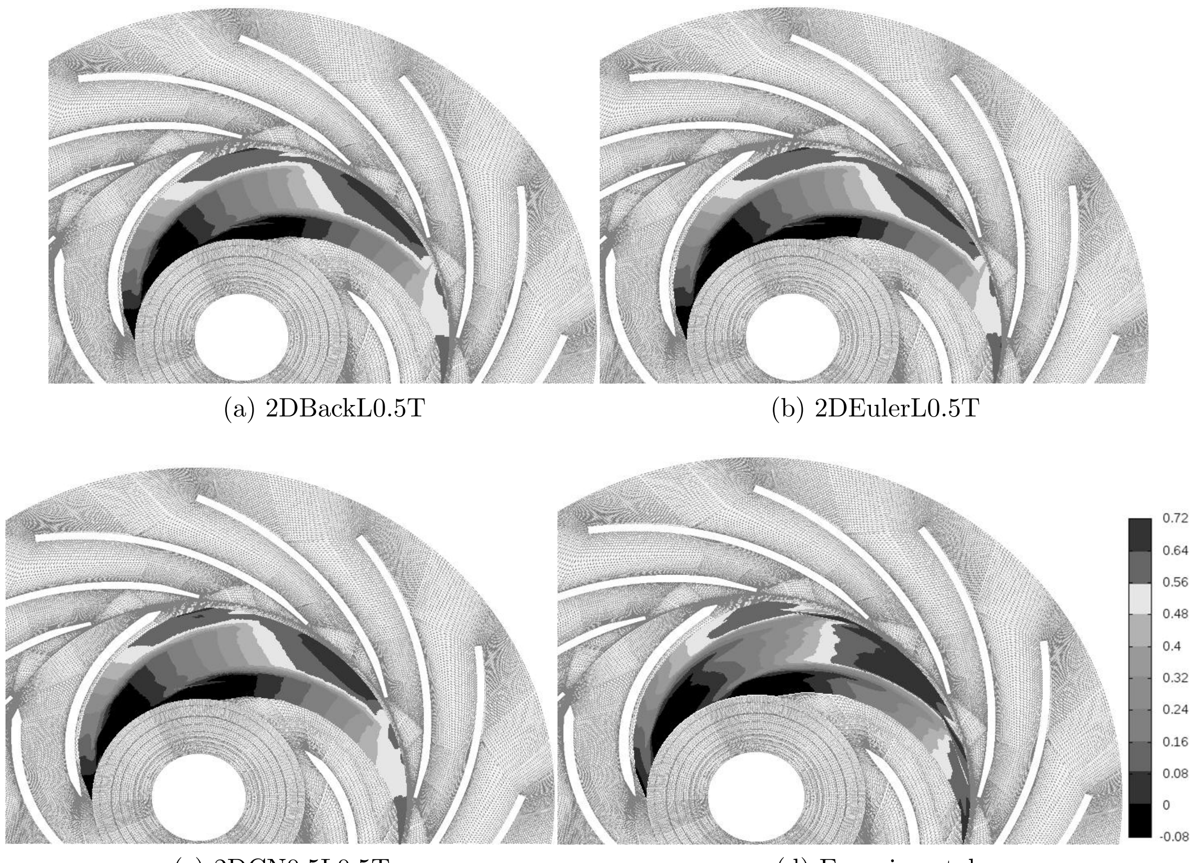


Figure 8.11: Static pressure coefficient at the impeller outlet for cases 2DEulerL0.5T, 2DBackL0.5T and 2DCN0.5L0.5T.



(a) 2DBackL0.5T

(b) 2DEulerL0.5T

(c) 2DCN0.5L0.5T

(d) Experimental

Figure 8.12: Instantaneous pictures of the static pressure coefficient at the front end of the impeller for cases 2DBackL0.5T (top left), 2DEulerL0.5T (top right), 2DCN0.5L0.5T (bottom left) and Experimental (bottom right).

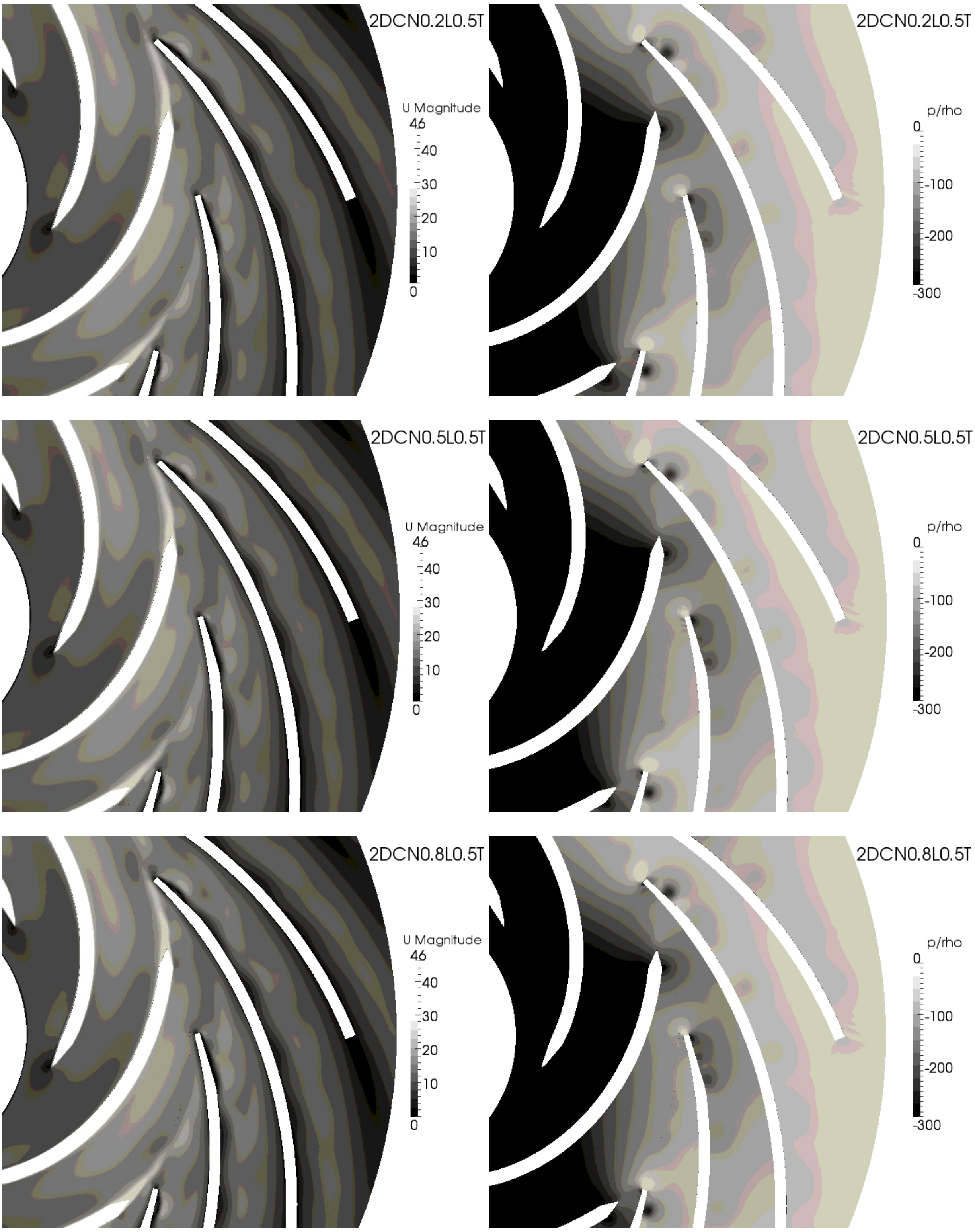


Figure 8.13: Relative velocity magnitude (left) and static pressure (right) for cases 2DCN0.2L0.5T (top), 2DCN0.5L0.5T (middle) and 2DCN0.8L0.5T (bottom).

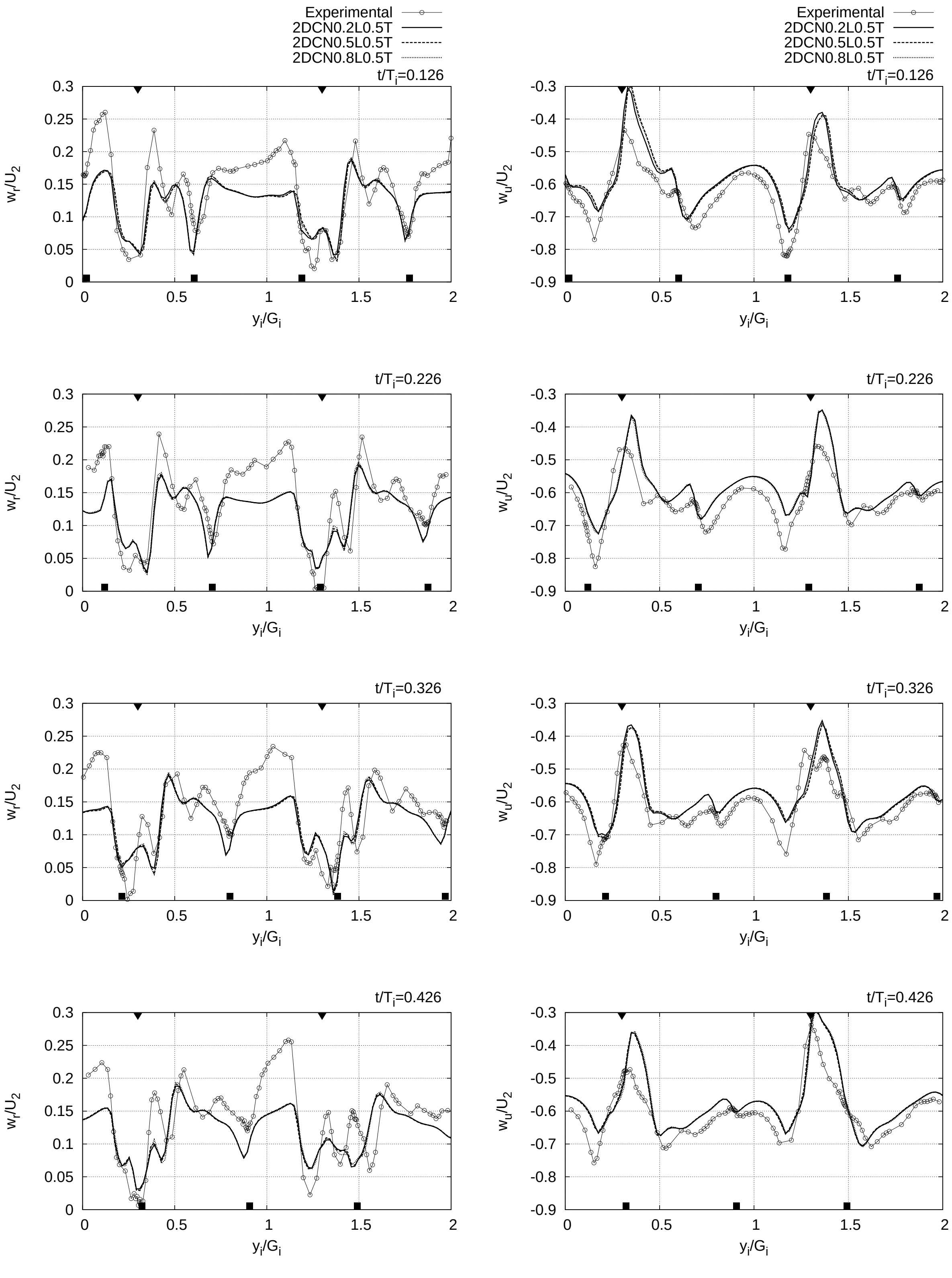


Figure 8.14: Radial (left) and tangential (right) velocities for cases 2DCN0.2L0.5T, 2DCN0.5L0.5T and 2DCN0.8L0.5T.

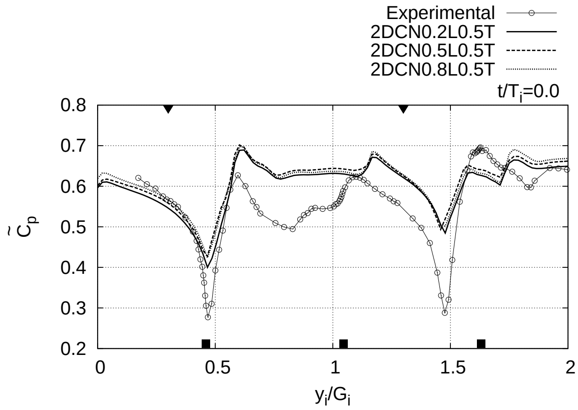


Figure 8.15: Static pressure coefficient at the impeller outlet for cases 2DCN0.2L0.5T, 2DCN0.5L0.5T and 2DCN0.8L0.5T.

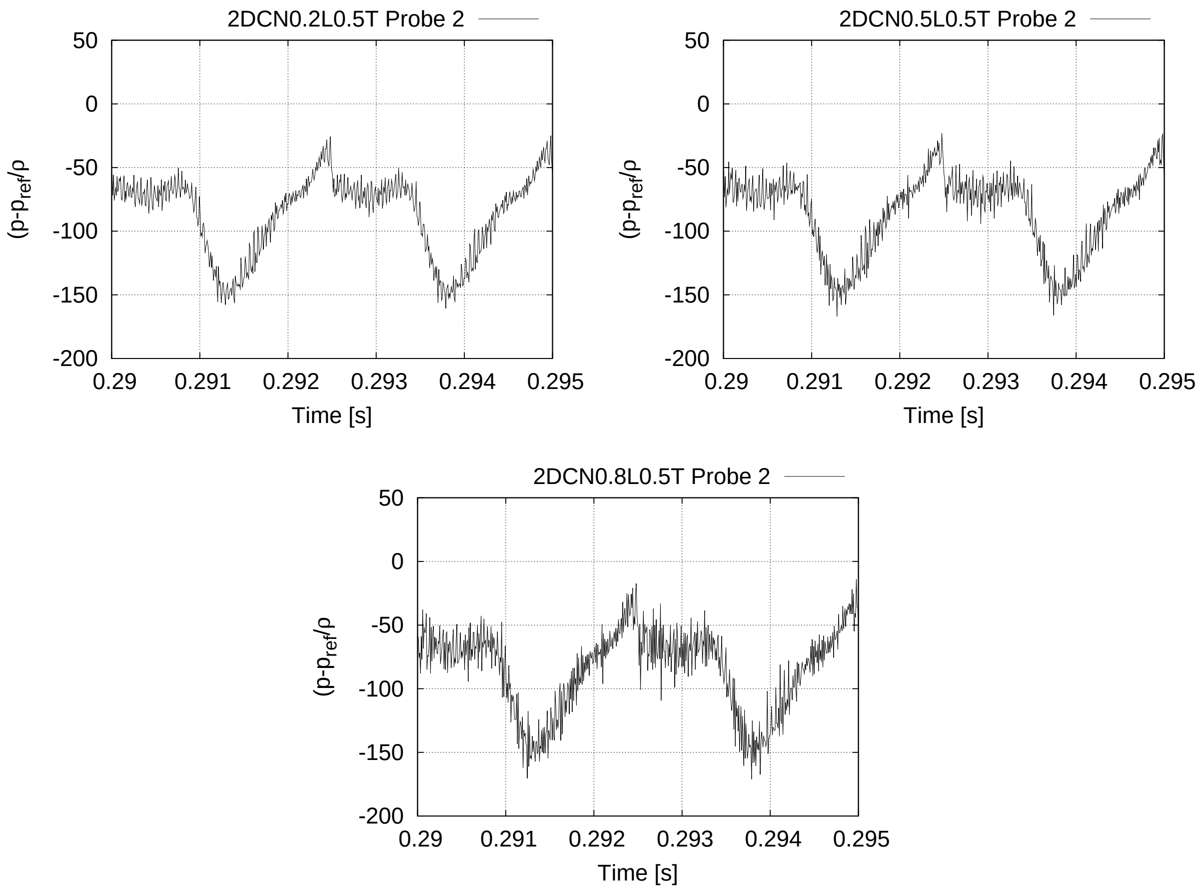
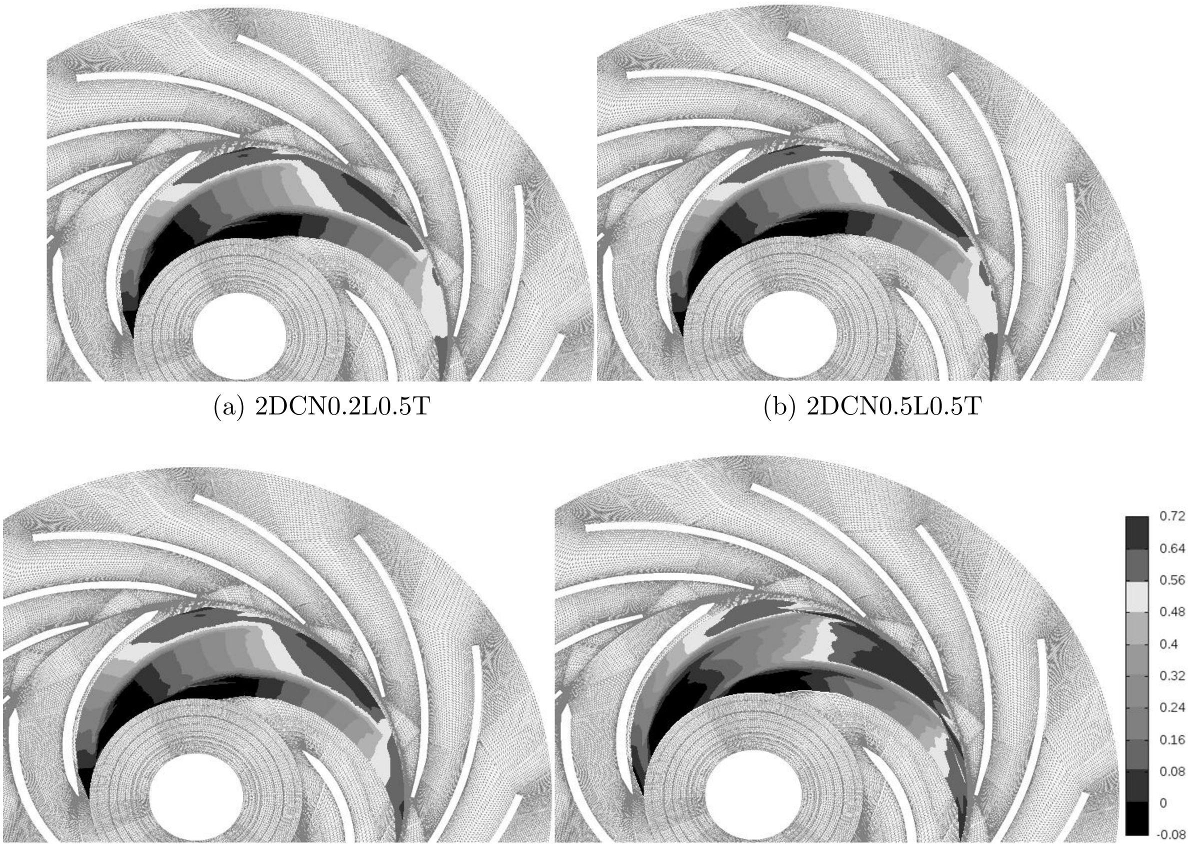


Figure 8.16: Pressure oscillations of Probe 2 in cases 2DCN0.2L0.5T (top left), 2DCN0.5L0.5T (top right) and 2DCN0.8L0.5T (bottom).



(a) 2DCN0.2L0.5T

(b) 2DCN0.5L0.5T

(c) 2DCN0.8L0.5T

(d) Experimental

Figure 8.17: Instantaneous pictures of the static pressure coefficient at the front end of the impeller for cases 2DCN0.2L0.5T (top left), 2DCN0.5L0.5T (top right), 2DCN0.8L0.5T (bottom left) and Experimental (bottom right).

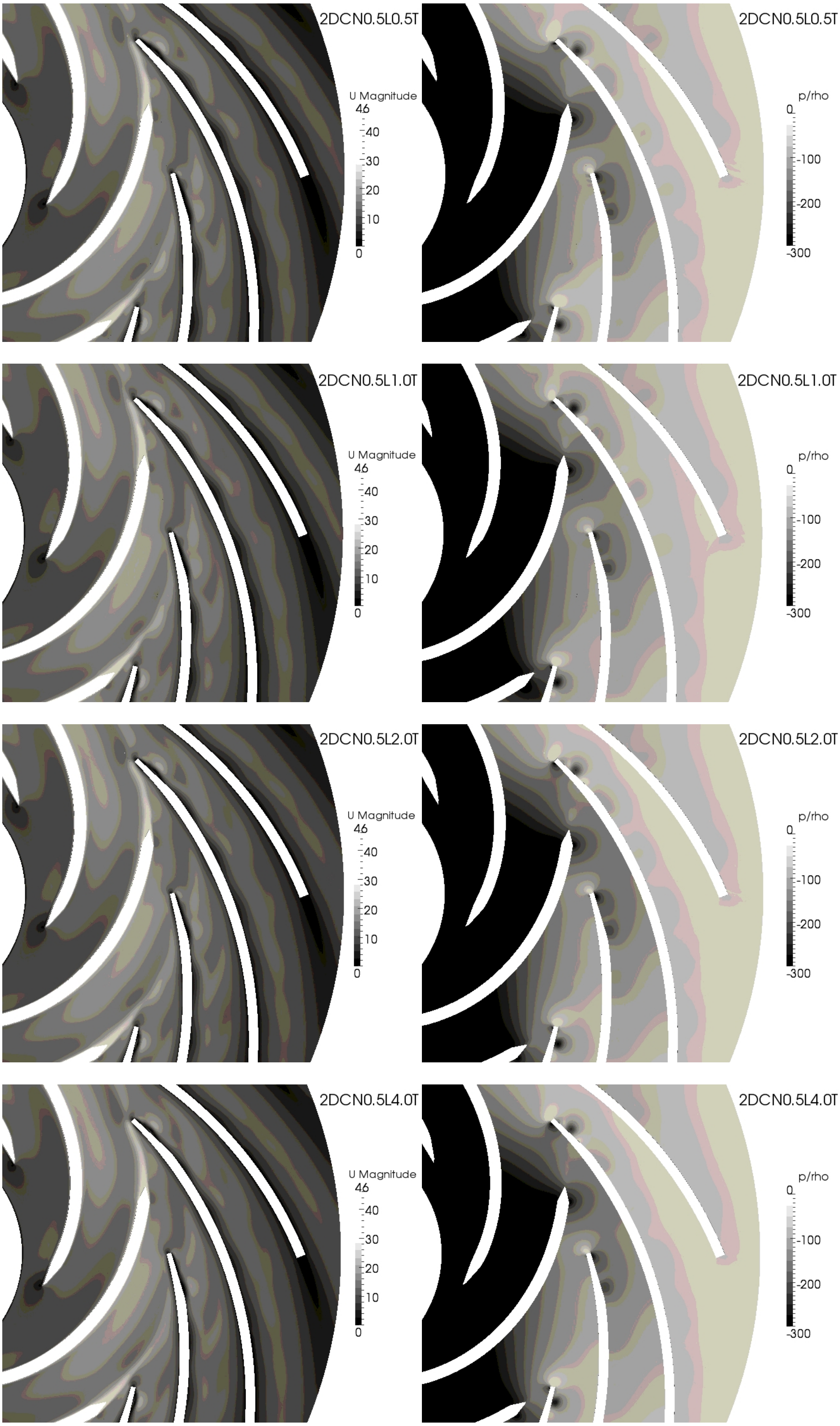


Figure 8.18: Relative velocity magnitude (left) and static pressure (right) for cases 2DCN0.5L0.5T, 2DCN0.5L1.0T, 2DCN0.5L2.0T and 2DCN0.5L4.0T.

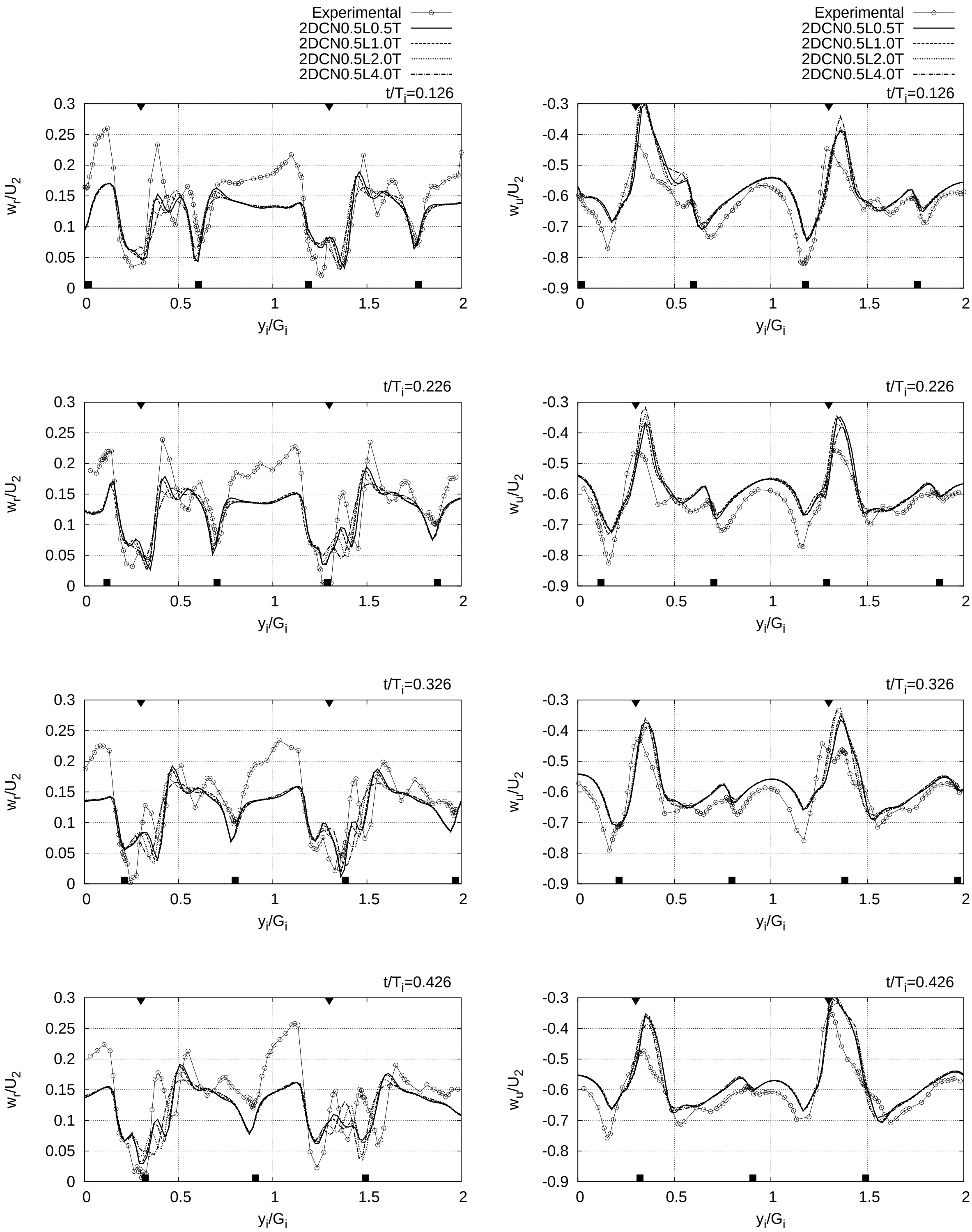


Figure 8.19: Radial (left) and tangential (right) velocities for cases 2DCN0.5L0.5T, 2DCN0.5L1.0T, 2DCN0.5L2.0T and 2DCN0.5L4.0T.

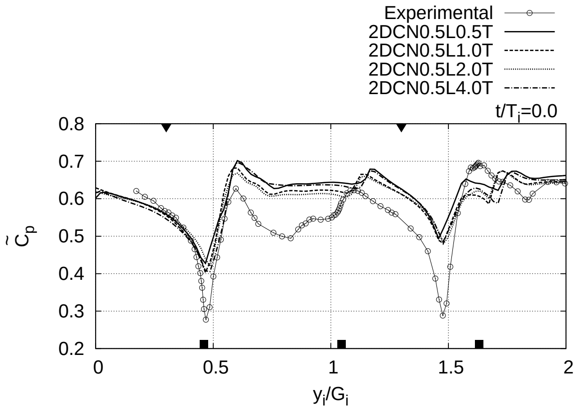
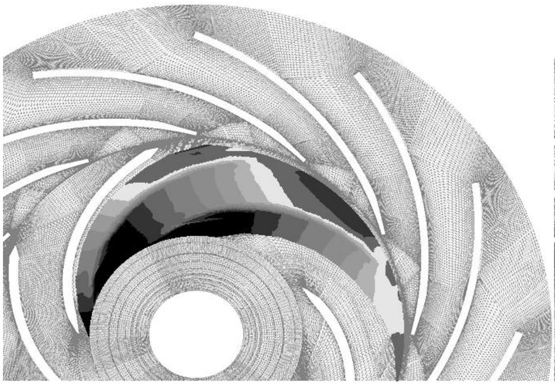
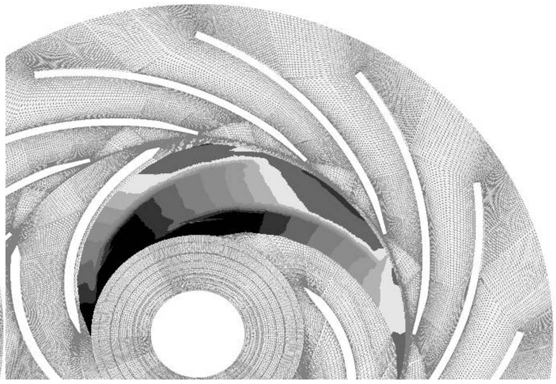


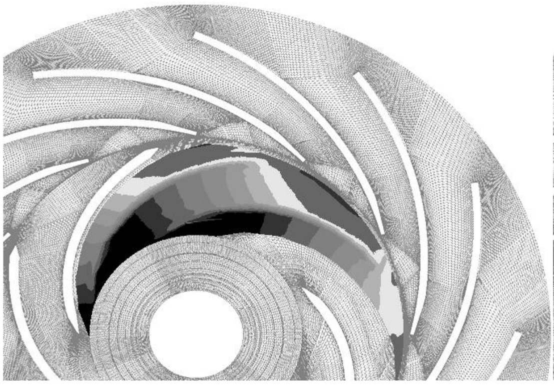
Figure 8.20: Static pressure coefficient for cases 2DCN0.5L0.5T, 2DCN0.5L1.0T, 2DCN0.5L2.0T and 2DCN0.5L4.0T.



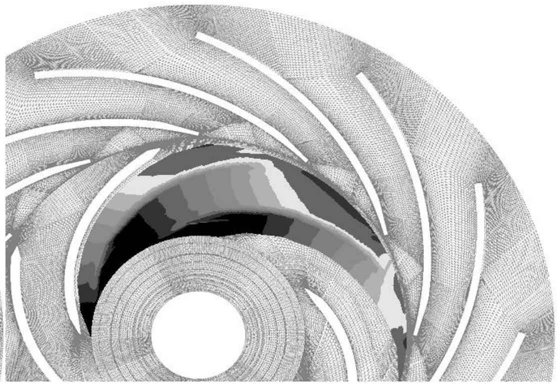
(a) 2DCN0.5L0.5T



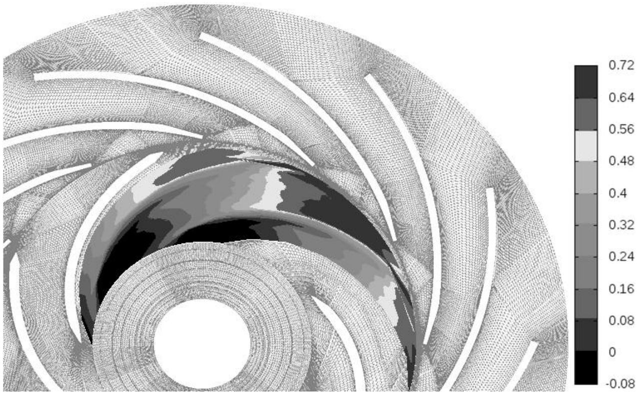
(b) 2DCN0.5L1.0T



(c) 2DCN0.5L2.0T



(d) 2DCN0.5L4.0T



(e) Experimental

Figure 8.21: Instantaneous pictures of the static pressure coefficient at the front end of the impeller for cases 2DCN0.5L0.5T (top left), 2DCN0.5L1.0T (top right), 2DCN0.5L2.0T (middle left), 2DCN0.5L4.0T (middle right) and Experimental (bottom).

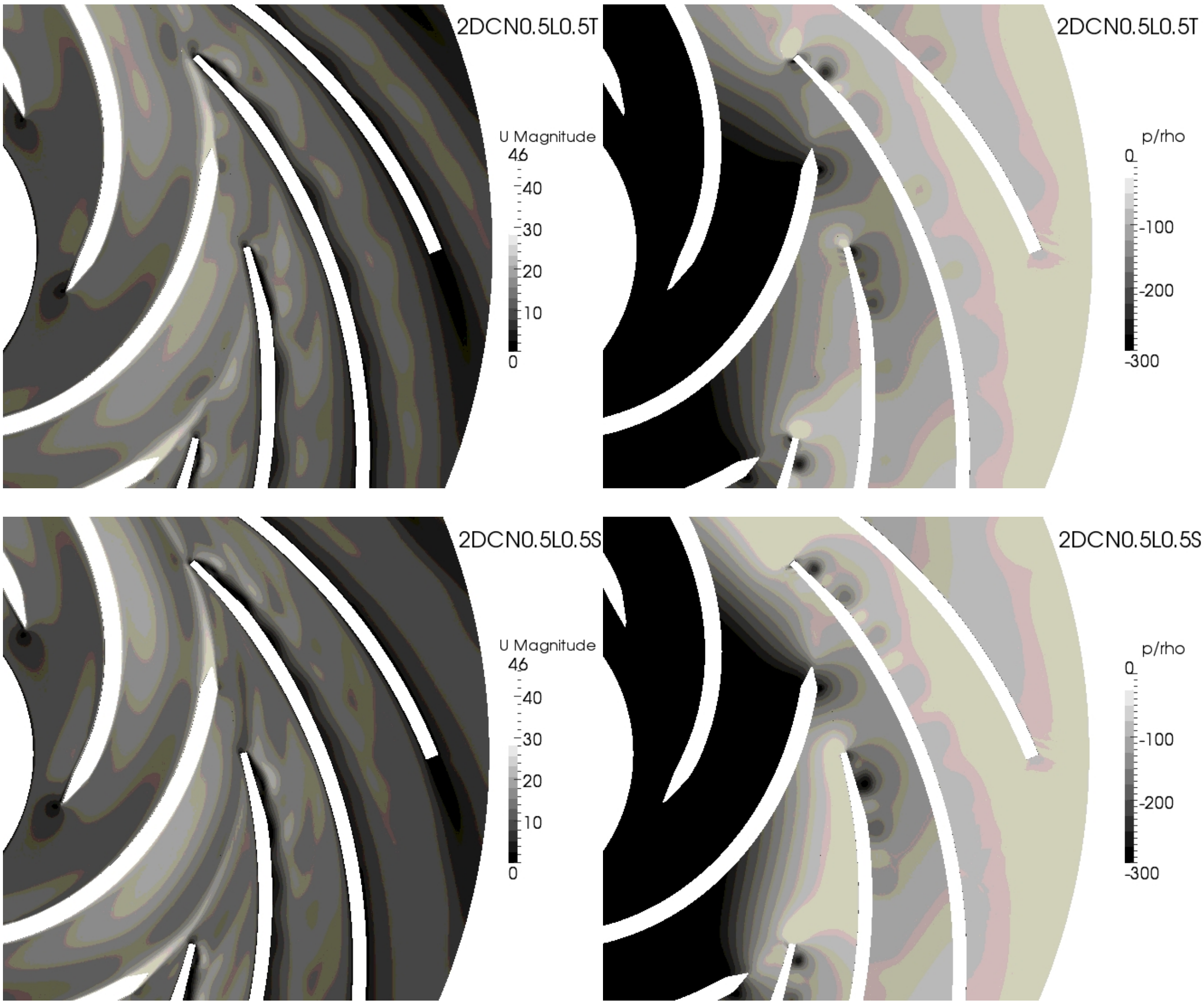


Figure 8.22: Relative velocity magnitude (left) and static pressure (right) for cases 2DCN0.5L0.5T (top) and 2DCN0.5L0.5S (bottom).

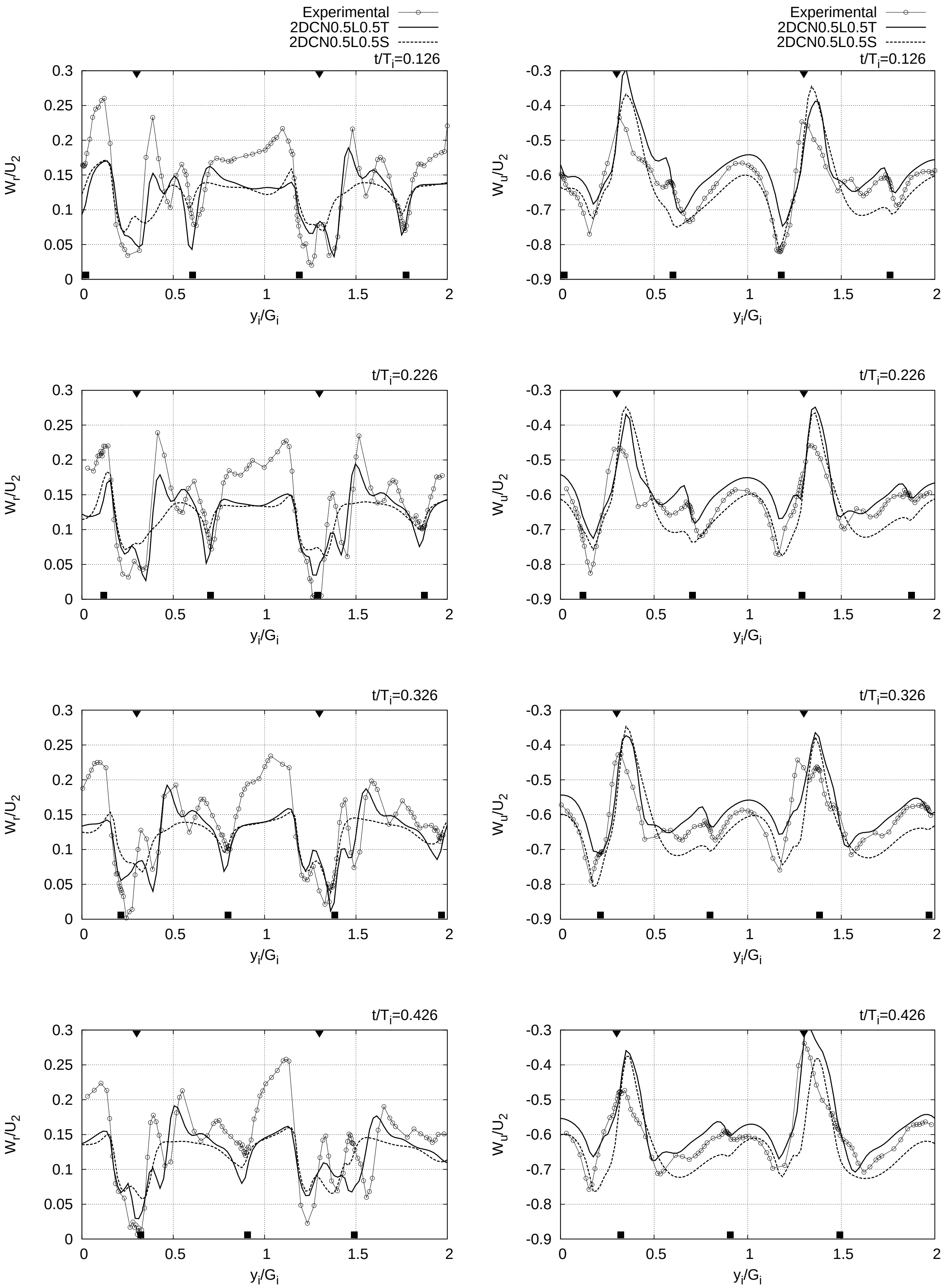


Figure 8.23: Radial (left) and tangential (right) velocities for cases 2DCN0.5L0.5T and 2DCN0.5L0.5S.

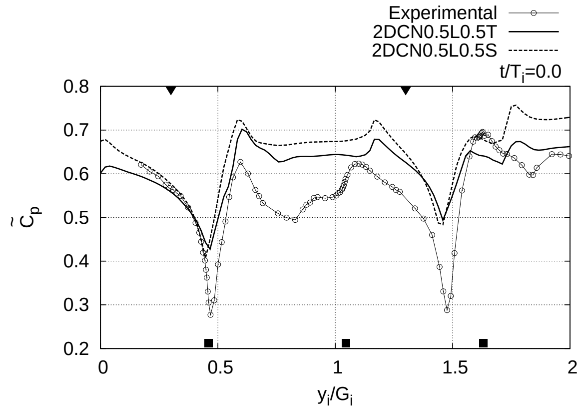
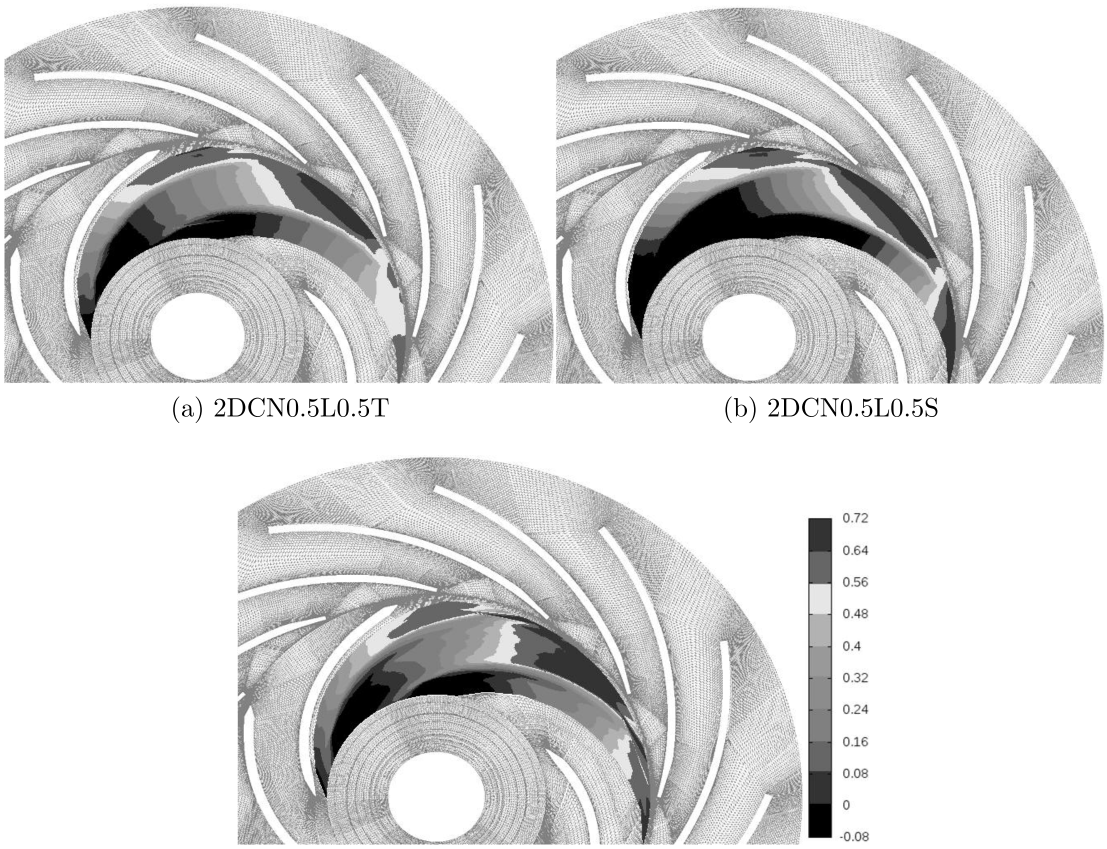


Figure 8.24: Static pressure coefficient at the impeller outlet for cases 2DCN0.5L0.5T and 2DCN0.5L0.5S.



(a) 2DCN0.5L0.5T

(b) 2DCN0.5L0.5S

(c) Experimental

Figure 8.25: Instantaneous pictures of the static pressure coefficient at the front end of the impeller for cases 2DCN0.5L0.5T (top left), 2DCN0.5L0.5S (top right) and experimental (bottom).

8.3 3D steady-state simulation results

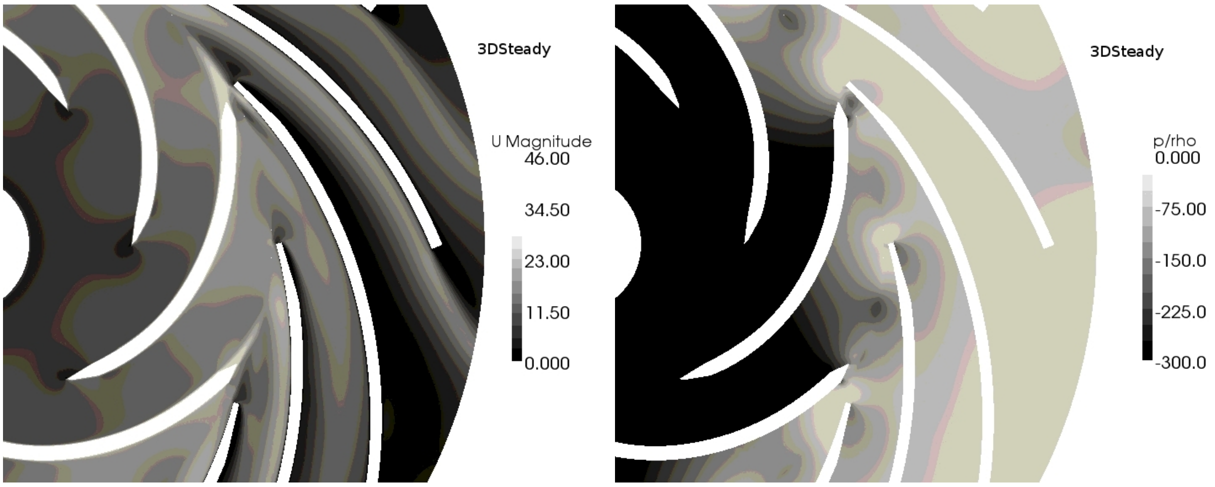


Figure 8.26: Relative velocity magnitude (left) and static pressure (right) at the midspan position for case 3DSteady.

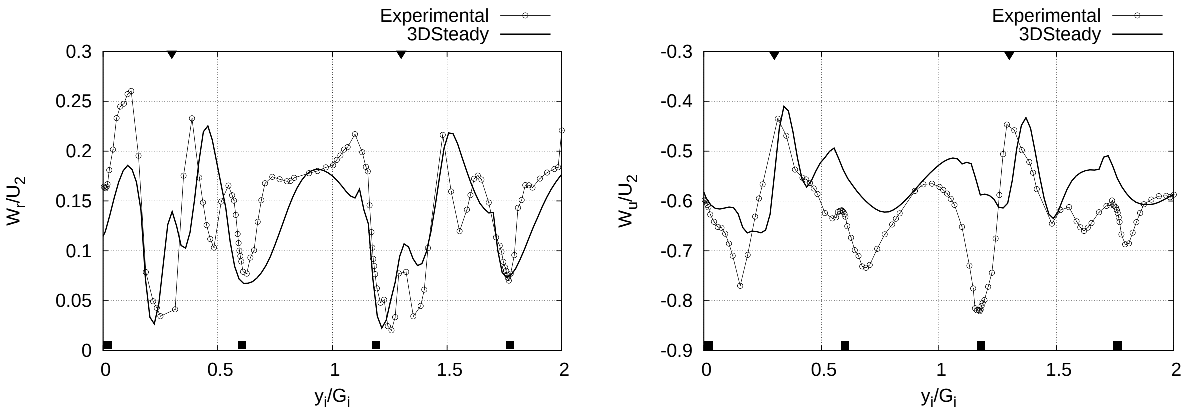


Figure 8.27: Radial (left) and tangential (right) velocities at the impeller outlet at the midspan position for case 3DSteady.

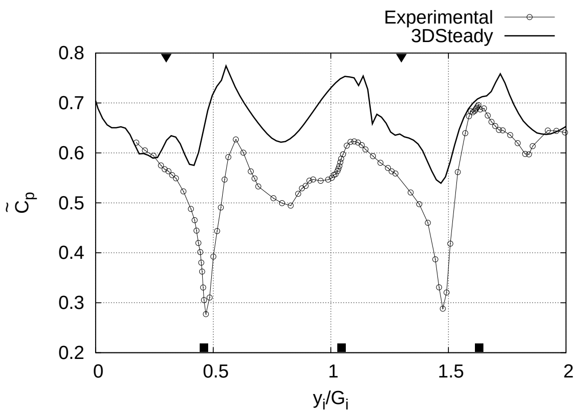
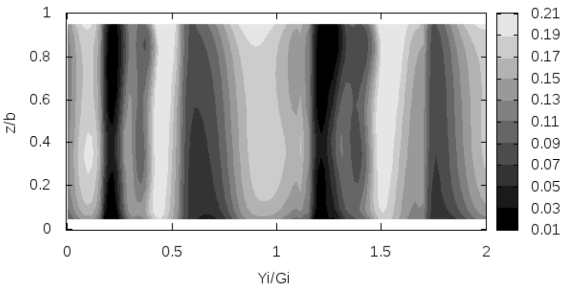
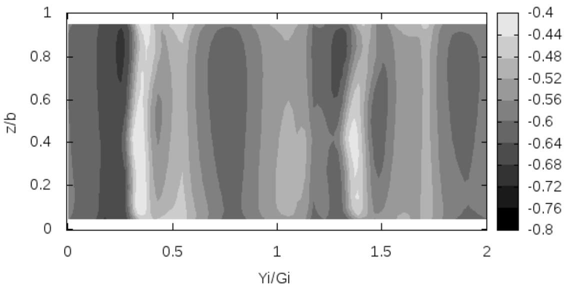


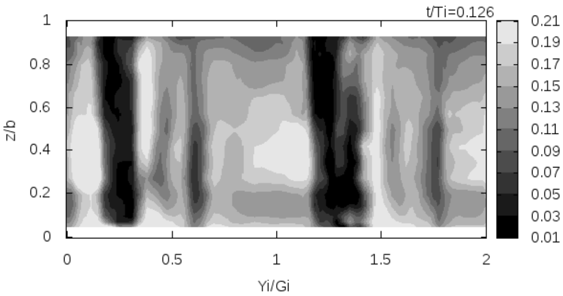
Figure 8.28: Static pressure coefficient at the impeller outlet at the midspan position for case 3DSteady.



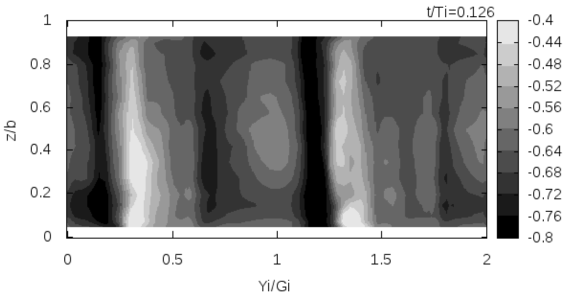
(a) Radial velocities for the case 3DSteady.



(b) Tangential velocities for the case 3DSteady.



(c) Radial velocities for experimental.



(d) Tangential velocities for experimental.

Figure 8.29: Radial (left) and tangential (right) velocities at the impeller outlet for the different span positions for case 3DSteady (top) and experimental (bottom).

8.4 3D unsteady simulation results

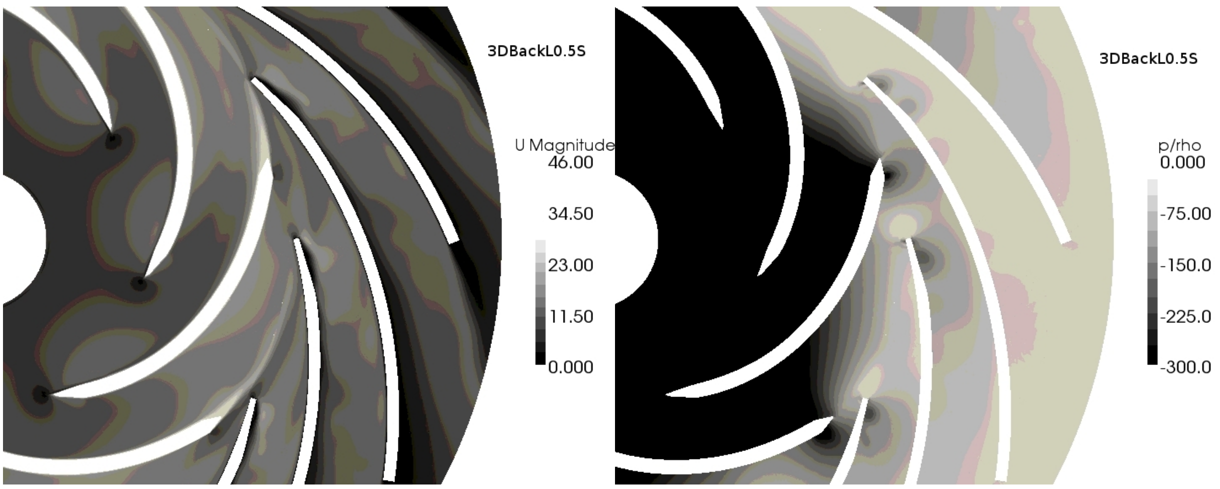


Figure 8.30: Relative velocity magnitude (left) and static pressure (right) for case 3DBackL0.5S.

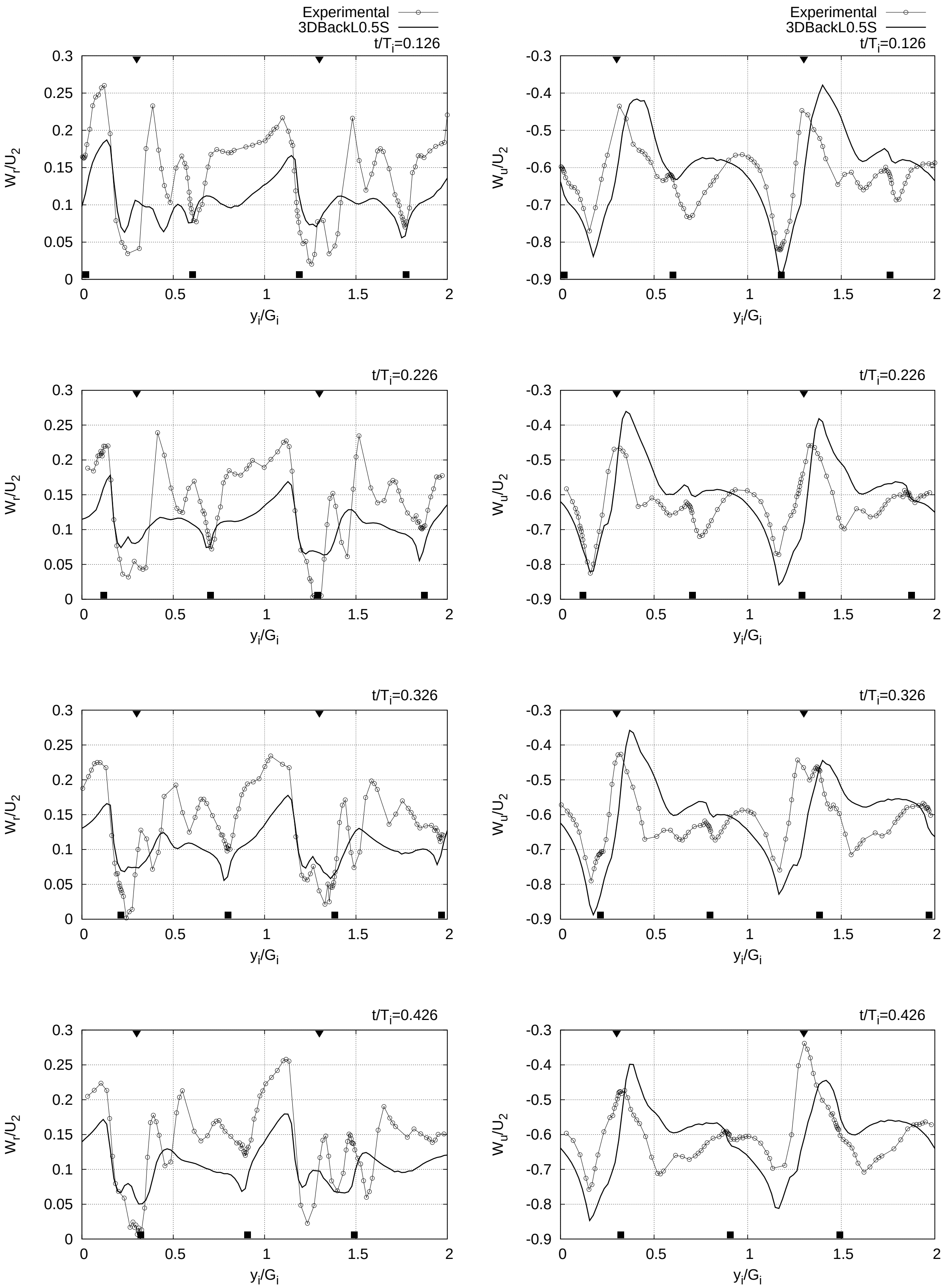


Figure 8.31: Radial (left) and tangential (right) velocities at the impeller outlet at the midspan position for case 3DBackL0.5S.

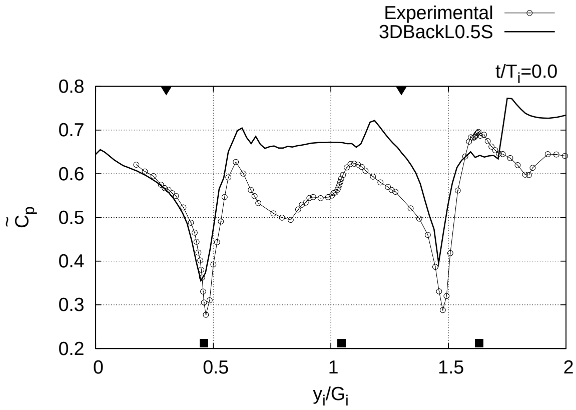
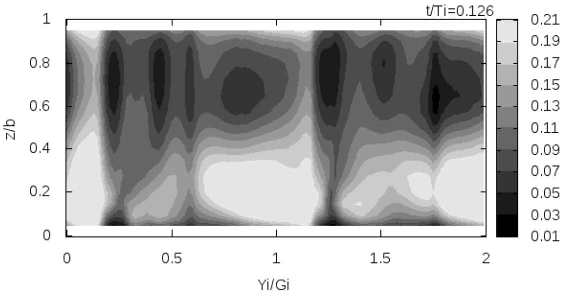
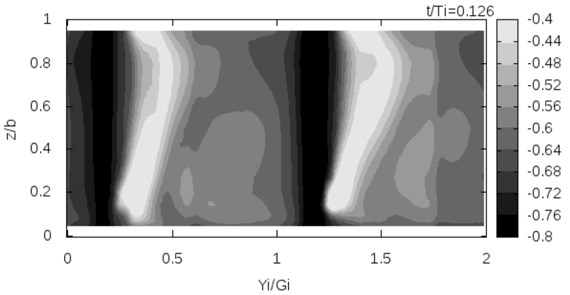


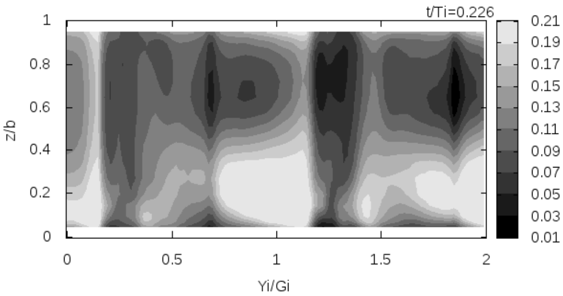
Figure 8.32: Static pressure coefficient at the impeller outlet at the midspan position for case 3DBackL0.5S.



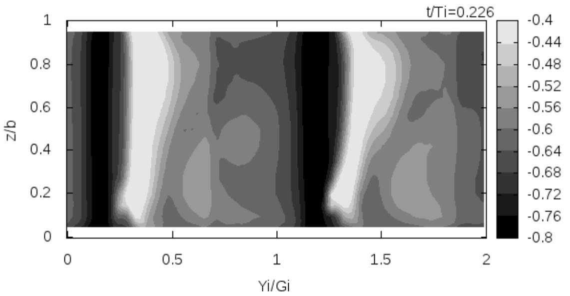
(a) Radial velocities for the case 3DBackL0.5S.



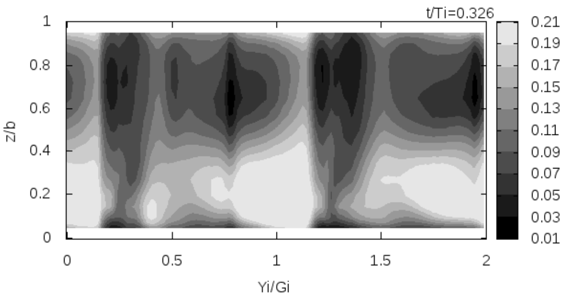
(b) Tangential velocities for the case 3DBackL0.5S.



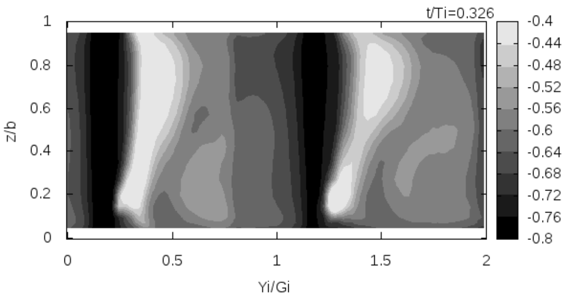
(c) Radial velocities for the case 3DBackL0.5S.



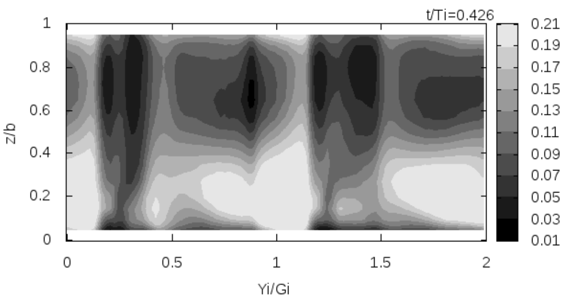
(d) Tangential velocities for the case 3DBackL0.5S.



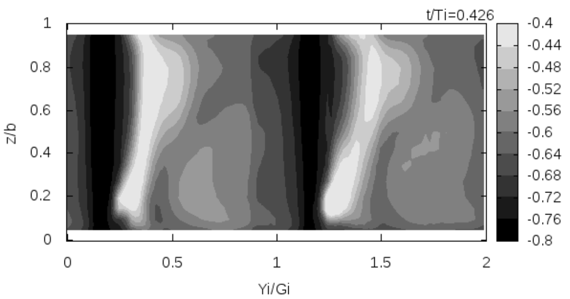
(e) Radial velocities for the case 3DBackL0.5S.



(f) Tangential velocities for the case 3DBackL0.5S.

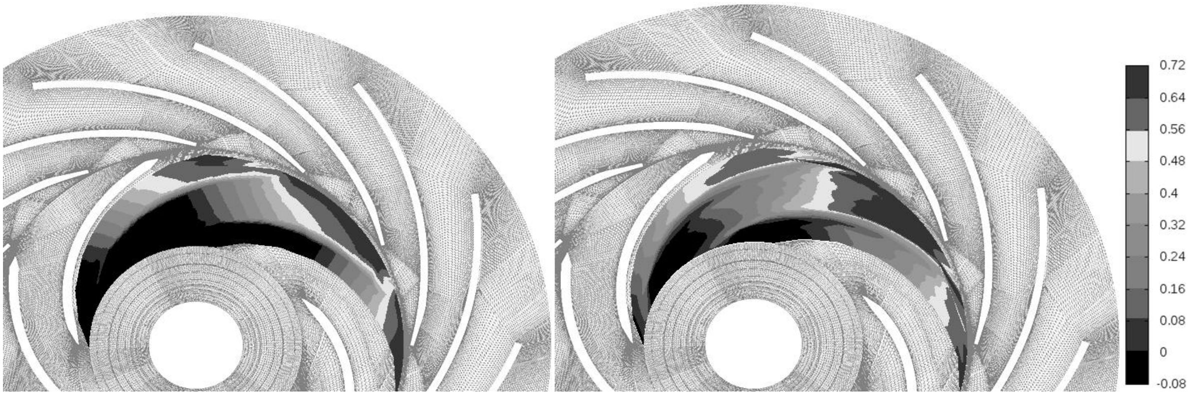


(g) Radial velocities for the case 3DBackL0.5S.



(h) Tangential velocities for the case 3DBackL0.5S.

Figure 8.33: Radial (left) and tangential (right) velocities at the impeller outlet for the different span positions for case 3DBackL0.5S.



(a) 3DBackL0.5S

(b) Experimental

Figure 8.34: Instantaneous pictures of the static pressure coefficient at the front end of the impeller for case 3DBackL0.5S (left) and experimental (right).