RESEARCH REPORT 2012:02

Transient and steady-state air flow simulations in generators using OpenFOAM

by

Pirooz Moradnia, Håkan Nilsson, Maryse Page, Martin Beaudoin, Federico Torriano, Jean-Francois Morissette, Kristopher Toussiant

Department of Applied Mechanics
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden, 2012

Transient and steady-state air flow simulations in generators using OpenFOAM

Pirooz Moradnia, Håkan Nilsson, Maryse Page, Martin Beaudoin, Federico Torriano, Jean-Francois Morissette, Kristopher Toussiant

Research report 2012:12
ISSN 1652-8549

Department of Applied Mechanics
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone +46-(0)31-7721000

This document was typeset using LaTeX

Göteborg, Sweden, 2012

Transient and steady-state air flow simulations in generators using OpenFOAM

PIROOZ MORADNIA, HÅKAN NILSSON, MARYSE PAGE, MARTIN BEAUDOIN, FEDERICO TORRIANO, JEAN-FRANCOIS MORISSETTE, KRISTOPHER TOUSSIANT

pirooz.moradnia@chalmers.se
Department of Applied Mechanics
Chalmers University of Technology

Abstract

The flow of air in two generator geometries has been simulated with OpenFOAM-1.6-ext and the results compared against CFX. The effect of the different simulation strategies as well as numerical methods, turbulence models, matrix linear solvers, matrix solver preconditioners, mesh resolutions, rotor-stator interface locations and boundary conditions on the flow has been examined. The convergence, computational time, flow field, axial torque on the rotor and windage losses have been observed and conclusions have been drawn.

Keywords: CFD, Rotating machines, OpenFOAM-1.6-ext, $pimpleDyMFoam$, $transientSimpleDyMFoam$, CFX

Acknowledgments

The present work is a result of a three-month research exchange at $IREQ$ research institute of $Hydro-Quebec$ company located in Varennes, Quebec-Canada. I would like to take the opportunity to thank my advisors and colleagues whose invaluable support and advice were the keys to deliver this work, and who made my stay in Quebec so pleasant.

The research presented was carried out as a part of the "Swedish Hydropower Centre - SVC". SVC has been established by the Swedish Energy Agency, Elforsk and Svenska Kraftnät together with Luleå University of Technology, The Royal Institute of Technology, Chalmers University of Technology and Uppsala University, www.svc.nu.

Pirooz Moradnia
Gothenburg-Sweden
March 2012

Contents

Chapter 1

Introduction

LOSSES the in electric generators are mainly classified in three groups. Electric losses appear because of the electric current passing through the electric resistances, which lead to an increase in the machine temperature and a decrease in the machine efficiency, as the electric resistance in the generator cables increase with the temperature. Magnetic losses are another group of losses which appear by the alternating magnetic field inside the generators, exerting forces on the magnetic particles and causing friction. Cooling down the generators to within the design temperature interval is, thus, important to operate the machines near their best efficiency points. The third group of losses are known as mechanical losses, which mainly include ventilation losses. The ventilation losses are caused by the flow of the cooling air into the generators. To increase the efficiency of the generators in converting the mechanical energy into electricity, the proper cooling of the machine together with minimal ventilation losses should be taken into consideration. A detailed study of the convective heat transfer in generators requires a correct understanding of the comlex air flow whithin these machines. A number of different strategies in simulating the air flow in rotating machines, including generators, are presented in [1]. Experimental and numerical studies have previously been performed to understand the flow in generators ([2], [3],[4],[5] and [6]).

The focus in the present work is to compare the different simulation strategies and parameters in simulating the flow in generators. A number of steady and unsteady simulations are performed on a 2D generator geometry as well as its 3D extrusions. The simulations are performed using the OpenFOAM and CFX solvers and the results are compared to each other. A simplified 2D test case, presented in the appendix, is designed to reduce the geometric complexities and a number of further studies are performed on it.

Chapter 2

Cases and Results

THE generator geometry is adopted from an experimental generator test rig located at the Hydro-Quebec research institue "IREQ", located in Varennes, Quebec-Canada.

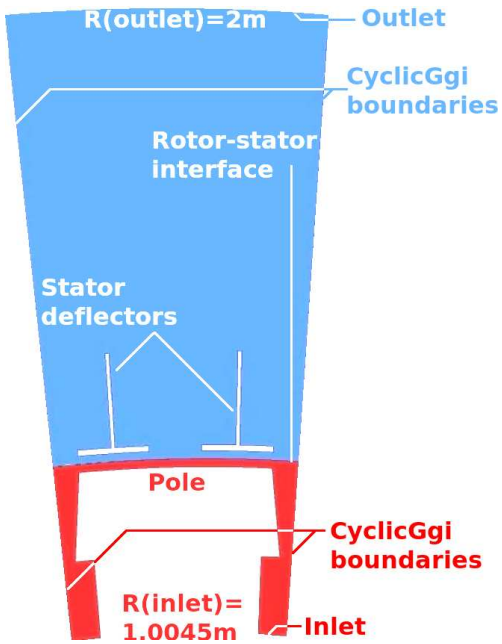

Figure 2.1: The generator geometry in the computational domain.

The generator is comprised of 36 rotor poles and 72 stator deflectors. The rotational speed of the rotor is $400rpm$. The computational domain is based on 10° sectors including a rotor pole and two stator deflectors. Figure 2.1 shows the computational domain and its different parts and the inlet and outlet radii. The inlet is radial, supplying a radial flow at a uniform speed of $5.22(m/s)$. The inlet boundary conditions result in a ratio $\frac{\mu_t}{\mu} = 10$, where the turbulent viscosity is found through $\mu_t = C_\mu \frac{k^2}{\varepsilon}$.

The velocity boundary condition at the outlet is set to $inletOutlet$, with zero $inletValue$ parameter at the outlet, which means that the velocity boundary condition at the outlet works as a $zeroGradient$ one as long as the flow is outwards. As soon as the outlet flow tends to be inwards, the $inletValue$ for the outlet velocity boundary condition will hinder the flow to be directed inwards to the boundary. The periodic sides of the computational domain are assigned $cyclicGgi$ boundary conditions, while maintaining a conformal mesh on both sides.The $cyclicGgi$ boundary conditions are $cyclic$ boundary conditions adapted for the general grid interfaces, meaning that the two periodic boundaries are coupled through averaging the flow properties over the cells in the same radial and axial position.

Two radial positions for the rotor-stator interface are considered: closer to the stator with a pole distance of $2/3$ of the air gap, $I : 2/3$, or halfway between the pole and the stator in the gap, $I : 1/2$. The cases are simulated in 2D and 3D computational domains, where the 3D domains are the extrusions of one or more sectors. The details of the simulated cases follow in the sections below.

2.1 2D geometry

The geometry used in the 2D cases is a 10° sector with a rotor pole and two stator deflectors in the computational domain. The interface is placed both in $I : 1/2$ and $I : 2/3$ positions in order to investigate the effect of the radial positioning of the interface on the results. The 2D cases are simulated in frozen rotor and sliding grid modes.

2.1.1 Frozen rotor

The focus in the frozen rotor simulations is to investigate the effect of the radial position of the interface, interface boundary condition, cell distributions between the two sides of the interface, as well as the mesh resolution on the results. Furthermore, the dependency of the flow on the relative position of the rotor and stator are analyzed.

The cases are simulated in aligned and offset modes. Figure 2.2 shows the schematic relative positions of the rotor poles and the stator deflectors in the aligned and the offset cases. The interfaces in the aligned cases are assigned either ggi or $overlapGgi$ boundary conditions, while the offset cases use $overlapGgi$ at the interface. The ggi boundary condition couples the two sides of the interface by integrating and averaging the quantities over the faces which are in front of

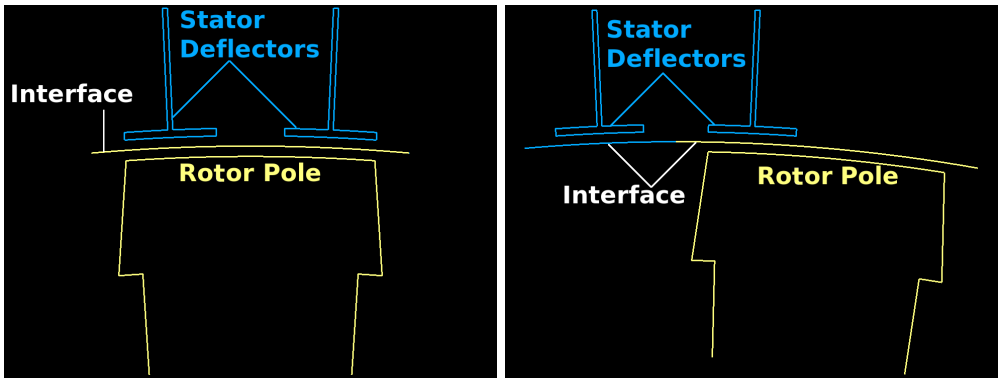

Figure 2.2: The relative rotor-stator position. Left: aligned. Right: offset.

each other on each side of the interface. The $overlapGgi$ boundary condition does the same but the two interfaces can have an offset. The uncovered faces from either side of the interface will then be coupled together. Therefore, the ggi boundary conditions can only be used for the aligned layout in steady state simulations, while the $overlapGgi$ boundary condition can be used for a general case.

Case name	Interface	Offset	Mesh
I:2/3,Aligned,ggi	I:2/3,ggi,general	$0°$	Coarse
I:2/3,Fine,Aligned,ggi	I:2/3,ggi,general	$0°$	Fine
I:2/3,Aligned,overlapGgi	I:2/3,overlapGgi,general	$0°$	Coarse
I:2/3,Offset,overlapGgi	I:2/3,overlapGgi,general	$2.5°$	Coarse
I:2/3,Aligned,1to1,ggi	I:2/3,ggi,1to1	$0°$	Coarse
I:2/3,Aligned,1to1,ovrlpGgi	I:2/3,overlapGgi,1to1	$0°$	Coarse
I:1/2,Aligned,ggi	I:1/2,ggi,general	$0°$	Coarse
I:1/2,Fine,Aligned,ggi	I:1/2,ggi,general	$0°$	Fine
I:1/2,Aligned,overlapGgi	I:1/2,overlapGgi,general	$0°$	Coarse
I:1/2,Offset,overlapGgi	I:1/2,overlapGgi,general	$2.5°$	Coarse

Table 2.1: Case properties: 2D, Frozen rotor

Table 2.1 summarizes the properties of the 2D frozen rotor cases. The computational meshes in the $I : 1/2$ and $I : 2/3$ cases differ in order to have the best cell distributions in each respective case. The purpose of simulating the cases with conformal interfaces, $I : 2/3\ Aligned\ 1to1\ ggi$ and $I : 2/3\ Aligned\ 1to1\ overlapGgi$, besides the cases with general interfaces is to investigate the sensitivity of the ggi and the $overlapGgi$ boundary conditions to the face distributions at the two sides of the interface. The two aforementioned cases share the same computational domain. Except for the cases with fine meshes, the cell distributions in the computational domains are adjusted to obtain wall y^+ values in

the range between 30 and 100 everywhere (except for the recirculation zones). The cases with fine meshes are based on the coarse meshes but are refined mainly in and close to the air gap. The refinements lead to wall y^+ values of less than 30 on the stator deflectors and the pole surface, which is aimed to investigate the sensitivity of the results to the mesh resolution. The computational meshes are the same for the cases $I : 2/3\, Aligned\, ggi$ and $I : 2/3\, Aligned\, overlapGgi$, as well as for the cases $I : 1/2\, Aligned\, ggi$ and $I : 1/2\, Aligned\, overlapGgi$. Depending on the cases, there are between $8k$ to $12k$ cells in the computational domain for each case. All cases are simulated in serial mode. The residuals for the 2D frozen rotor cases stay high, which is common for the frozen rotor cases. This is specially true for the pressure equation. The residuals for all cases are roughly of the same magnitudes.

Case name	$Time$ (s/iter)	$\overline{T}$ (Nm)	$\overline{WL}$ (W)
I:2/3,Aligned,ggi	0.23	0.012	1.60
I:1/2,Aligned,ggi	0.29	0.018	1.82
I:2/3,Fine,Aligned,ggi	0.30	0.012	1.58
I:1/2,Fine,Aligned,ggi	0.46	0.018	1.82
I:2/3,Aligned,overlapGgi	0.51	0.012	1.60
I:1/2,Aligned,overlapGgi	0.62	0.018	1.83
I:2/3,Offset,overlapGgi	0.57	0.008	1.95
I:1/2,Offset,overlapGgi	0.62	0.010	2.13
I:2/3,Aligned,1to1,ggi	0.24	0.013	1.62
I:2/3,Aligned,1to1,overlapGgi	0.46	0.013	1.62

Table 2.2: The required computational times per iteration (sec/iteration), the mean rotor torque and the mean windage losses for the 2D frozen rotor cases in OpenFOAM.

Table 2.2 shows the required computational times per iteration, the mean rotor torques and the mean windage losses for the 2D frozen rotor cases in OpenFOAM. The aligned cases with the $overlapGgi$ interface need about twice as long a computational time as the corresponding cases with ggi boundary conditions. Since the only difference in these cases is the interface boundary condition, there is an improvement potential in the $overlapGgi$ boundary condition to increase the computational performance.

The computed rotor torque values are not constant during the simulations and show periodic oscillations with iterations. The oscillations are stronger in the aligned cases. The mean torques are obtained by averaging the torque curves over an integer amount of periods and do not show considerable dependency on the mesh resolution or the inter-

face type. The interface radial position as well as the relative position of the rotor and stator strongly affect the torques.

The windage losses are computed as

$$WL = \frac{d}{dt}\left[\dot{E}_{in} - \dot{E}_{out}\right] + T \times \omega \tag{2.1}$$

where $\dot{E}_{in}$ and $\dot{E}_{out}$ are the mechanical energies at the inlet and at the outlet respectively, T is the axial torque on the rotor, and ω is the rotational speed of the rotor. The mean windage losses are not affected by the mesh resolution and the interface boundary condition, but by the interface radial position and the rotor-stator relative position. This is because both $\dot{E}_{out}$ T are dependent on the interface radial position and the rotor-stator relative position. The computed windage losses of the $I : 1/2$ cases by OpenFOAM and CFX have similar behaviors, with CFX mean-torque predictions of $1.74(W)$ and $2.02(W)$ respectively for the aligned and the offset cases in CFX, meaning 10% larger mean windage losses in CFX.

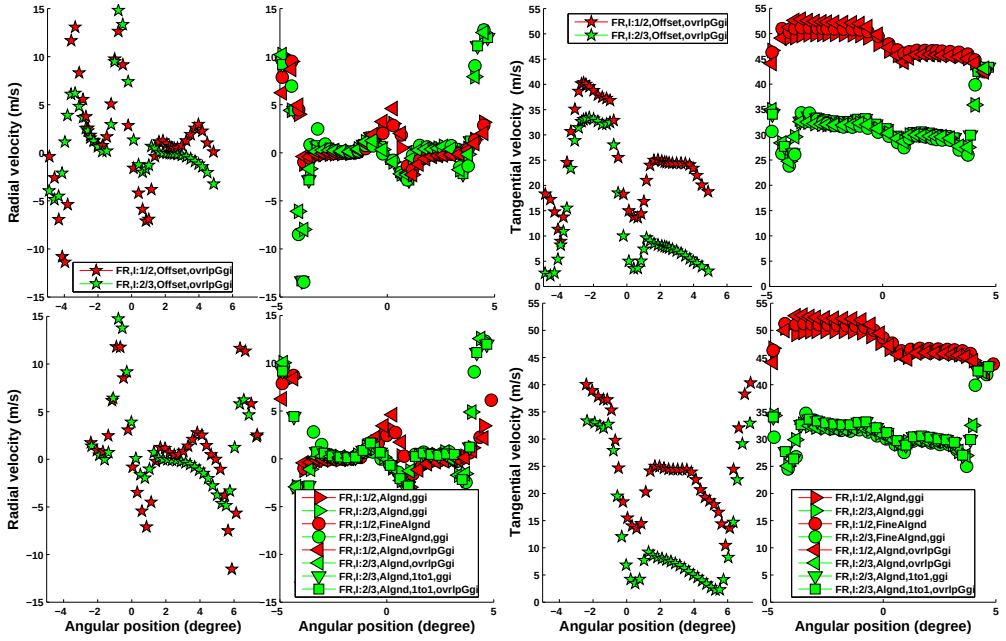

Figure 2.3: The velocity profiles at the two sides of the interface. Left: radial velocities in the offset cases. Middle left: radial velocities in the aligned cases. Middle right: tangential velocities in the offset cases. Right: tangential velocities in the aligned cases. Top: the stationary region. Bottom: the rotating region.

Figure 2.3 shows the radial and the tangential velocity profiles at the interfaces in the offset and the aligned cases. The results are shown after the same amount of iterations and sufficiently long for all cases

to converge. The plots with green and red markers show the profiles for the $I : 2/3$ and $I : 1/2$ cases respectively. The velocity profiles at the two sides of the interface are perfectly matching in all cases. The mesh resolution and the interface boundary condition do not affect the radial velocities at the interfaces. The shape of the radial velocity profiles are also dependent on the rotor-stator relative position. The mesh resolution and interface boundary condition do not considerably affect the tangential velocity profiles at the interfaces. The tangential velocities are more affected by the radial positioning of the interface than the radial velocities.

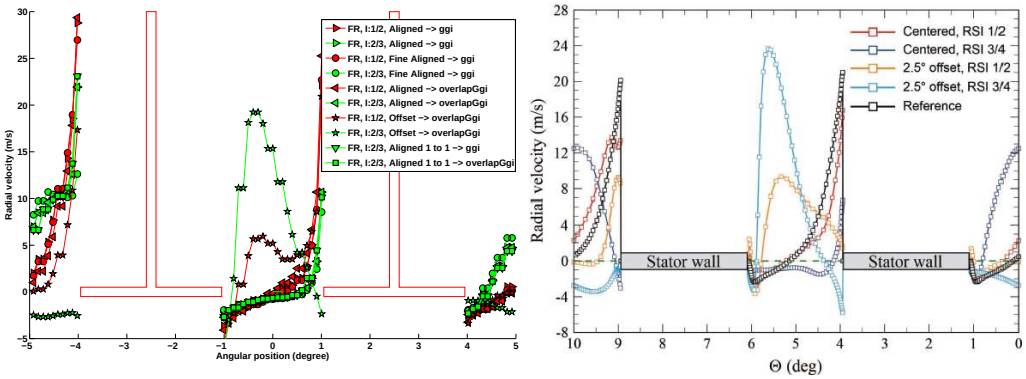

Figure 2.4: The radial velocities between the stator deflectors. Left: OpenFOAM. Right: CFX

Figure 2.4 shows the radial velocities between the stator deflectors in OpenFOAM and CFX. The velocity profiles for the corresponding cases in OpenFOAM and CFX have the same behaviors and about the same magnitudes. The radial velocities between the stator deflectors are not affected by the mesh resolution and interface boundary condition, but by the radial position of the interface, as well as the rotor-stator relative position.

2.1.2 sliding grid

The focus in the sliding grid simulations is investigating the effect of the mesh resolution and the interface radial position on the results. The in-house developed OpenFOAM solver $transientSimpleDyMFoam$ is utilized in simulations, in which the velocity and pressure fields are corrected through ”Simple” algorithm. The boundary conditions and the numerical setups as well as the turbulence models are similar to the 2D frozen rotor base case.

Table 2.3 summarizes the properties of the 2D sliding grid cases. Four cases are simulated, two with the interface position at $I : 1/2$ and

Case name	Interface	Interface BC	Mesh	p-U corr.
I:1/2,Coarse	I:1/2	overlapGgi	Coarse	50
I:2/3,Coarse	I:2/3	overlapGgi	Coarse	50
I:1/2,Fine	I:1/2	overlapGgi	Fine	20
I:2/3,Fine	I:2/3	overlapGgi	Fine	20

Table 2.3: Case properties: 2D sliding grid cases

two at $I : 2/3$. Each interface position is simulated with two mesh resolutions. The coarse cases have wall y^+ values in the allowed interval for the standard $k - \varepsilon$ turbulence model with wall functions, while the fine cases give too low y^+ values. The computational meshes differ in all cases in order to have the best cell distribution in each case. Depending on the case, the computational domains consist of between $7k$ to $12k$ cells. The time steps are set to $10^{-5}s$, guaranteeing a maximum Courant number of less than 1 everywhere in the computational domain. The simulations are performed in serial mode. The number of $p - U$ corrections in each time step determines the final residuals. The cases with more $p - U$ correction outer loops have lower residuals at the final iteration in each time step.

Case name	$Time$ (Hr/s)	$\overline{T}$ (Nm)	$\overline{WL}$ (W)
I:2/3,Coarse	394	0.0170	1.91
I:2/3,Fine	358	0.0175	1.94
I:1/2,Coarse	465	0.0168	1.90
I:1/2,Fine	381	0.0172	1.93

Table 2.4: The required computational per runtime (Hr/s), the mean rotor torque and the mean windage losses for the 2D sliding grid cases in OpenFOAM.

Table 2.4 shows the computational per runtime (Hr/s), the mean rotor torque and the mean windage losses for the 2D sliding grid cases. The computational times to get a well converged solution are much longer than in the frozen rotor cases. In the coarse cases the pressure and velocity fields are corrected 50 times in each time step, while in the fine cases 20 corrections are performed in each time step, which explains the shorter computational times for the fine cases compared to the coarse cases.

The time-averaged torques are calculated over enough number of periods for all cases. The time-averaged torques are independent of the interface radial position and the mesh resolution in sliding grid mode.

The time-averaged windage losses in OpenFOAM are almost independent of the interface radial position and the mesh resolution. The computed mean windage loss in CFX is about $1.9(W)$ which is very close to the OpenFOAM results. The mean windage losses by OpenFOAM and CFX are much closer in the sliding grid mode than in the frozen rotor computations.

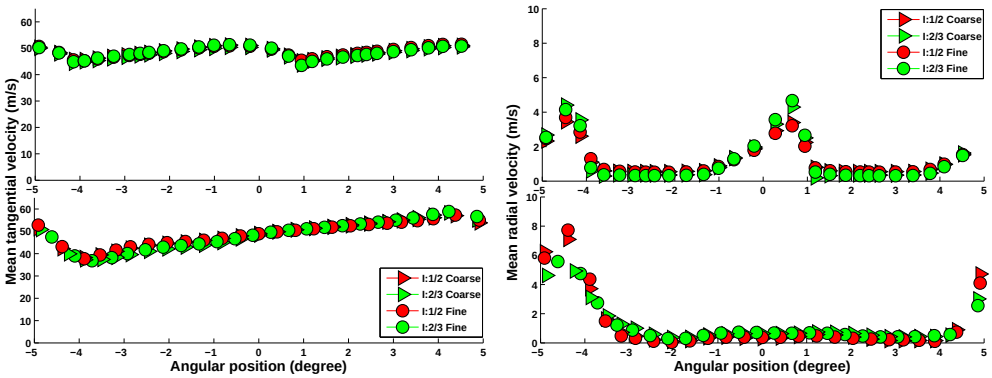

Figure 2.5: The time-averaged velocities at the two sides of the interface. Left: tangential. Right: radial. Top: the stationary region. Bottom: the rotating region.

Figure 2.5 shows the time-averaged tangential and radial velocities at the two sides of the interface. The profiles are averaged at the same time and over a number of periods. The tangential velocity profiles are unaffected by the mesh resolution and the radial position of the rotor-stator interface. The radial velocity profiles are unaffected by the mesh resolution, while are slightly affected by the interface radial position. The slight difference in the radial velocity profiles at different interface radial positions is a result of the continuity.

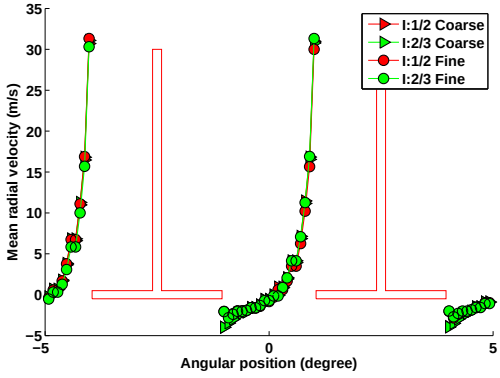

Figure 2.6: The time-averaged radial velocities between the stator deflectors in sliding grid cases

Figure 2.6 shows the time-averaged radial velocities between the

stator deflectors in the sliding grid cases. The time-averaged radial velocity profiles for all sliding grid cases fall over each other, meaning that the velocity field is independent of the interface radial position and the mesh resolution.

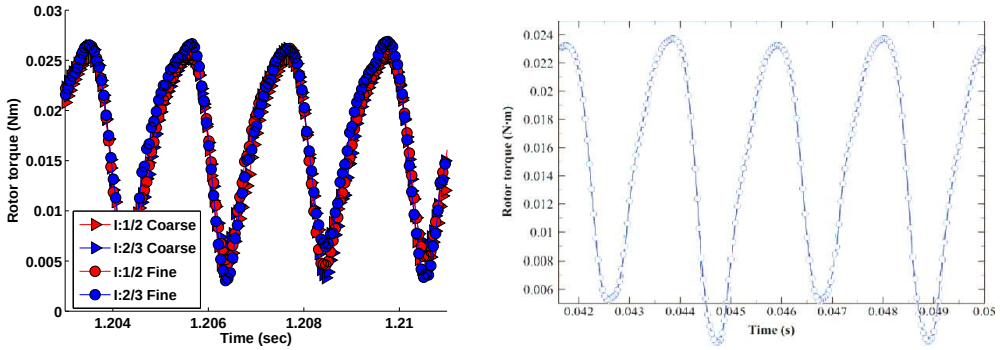

Figure 2.7: The computed torques for the 2D sliding grid cases. Left: OpenFOAM. Right: CFX.

Figure 2.7 shows the torque curves in OpenFOAM and in CFX for the 2D sliding grid cases. The curves are not completely periodic in either OpenFOAM or CFX. The results from both OpenFOAM and CFX have the same behaviors, while the maximum torque values from OpenFOAM are about 15% higher than the maximum torques from CFX. This results in about 10% higher mean torques in OpenFOAM than in CFX.

2.2 3D geometry

The geometries used in the 3D cases are based on a one meter-long axial extrusion of the 2D cases.

Case name	Periodic BC	CPUs	Sector	Cells
$cycGgi\,10°\,01\,CPU$	cyclicGgi	1	$10°$	390k
$cycGgi\,10°\,16\,CPU$	cyclicGgi	16	$10°$	390k
$cyclic\,10°\,16\,CPU$	cyclic	16	$10°$	390k
$cycGgi\,90°\,16\,CPU$	cyclicGgi	16	$90°$	3.5M
$cycGgi\,90°\,64\,CPU$	cyclicGgi	64	$90°$	3.5M

Table 2.5: Case properties: 3D frozen rotor cases

Table 2.5 shows the properties of the 3D frozen rotor cases. Five cases are simulated, out of which three include one $(10°)$ sector, and two include nine $(10°)$ sectors. All 3D cases are simulated in frozen

rotor mode with ggi boundary conditions on the rotor-stator interfaces. The interface radial position is set to $I : 2/3$ in all cases. The parameters changed for different simulations include the number of CPUs and the periodic boundary conditions, which were set to either $cyclic$ or $cyclicGgi$. The focus in this part of the study is to investigate the amount of computational time needed to compute the flow field.

Case name	$Time$ (s/iter)
$cycGgi\,10°\,01\,CPU$	30.6
$cycGgi\,10°\,16\,CPU$	7.3
$cyclic\,10°\,16\,CPU$	5.9
$cycGgi\,90°\,16\,CPU$	78.1
$cycGgi\,90°\,64\,CPU$	116.5

Table 2.6: The computational times per iteration (s/iter) for the 3D frozen rotor cases.

Table 2.6 shows the computational times for the 3D frozen rotor cases. The $cyclic$ boundary condition requires much less computational effort than the $cyclicGgi$ boundary conditions for the same geometry. This is because of the value averaging over the faces, included in the $cyclicGgi$ boundary condition which is not the case in the $cyclic$ boundary conditions. Therefore, much computational effort is saved if the boundaries utilizing ggi are reduced. The distribution of the computational cells between the CPUs should be performed carefully. Increasing the number of CPUs does not necessarily guarantee faster computations, as the communication between the CPUs may increase the required computational time. A correct domain decomposition algorithm is thus essential in optimizing the CPU communications and minimizing the computational time.

Chapter 3

Concluding Remarks and Future Work

THe flow in a generator geometry was numerically simulated in OpenFOAM and CFX to investigate the effect of solvers, matrix manipulation algorithms, matrix preconditioners, turbulence models, boundary conditions and numerical schemes on the flow field.

The frozen rotor results prove to be dependent on both the relative position of the rotor pole and stator deflectors, as well as the radial position of the rotor-stator interface. The latter is a result of the intrinsic property of the frozen rotor concept in transferring wakes in the wrong direction across the interface [7], which is in turn caused by the added rotation terms in the rotating region of the domain. The flow field in frozen rotor simulations will, thus, always be dependent on the interface radial position. The problem vanishes in sliding grid solutions, as the transient flow in each time step is simulated with new boundary conditions.

The computational time required to simulate the same frozen rotor cases can be much longer with $overlapGgi$ compared to ggi boundary conditions. Although the mean characteristics of the flow field will be similar, the flow fields can demonstrate different levels of stability with ggi and $overlapGgi$. Depending on the problem at hand, the $cyclic$ boundary conditions can save considerable amounts of computational time compared to the $cyclicGgi$ boundary conditions.

When simulating a case in parallel mode, the choice of the number of CPUs and the communication level between CPUs is of high importance. The decomposition of the computational domain should be done in a way that assures that the maximum allowed number of the computational cells will be handled by each CPU, and that the communications between CPUs will be minimized. This is specially important for cases including any form of ggi interfaces, where the flow properties

at each face will be weighted by the properties in the shadow cells and their surrounding cells.

Appendices

Appendix A

Test Cases and Validation

The simplified geometry is designed to reduce the geometric complexities and to perform the computations on a simpler domain with less computational cells. The rotor poles and stator deflectors are replaced by purely radial blades. The number of rotor passages (36), the inlet and the outlet boundary conditions and the main dimensions of the computational domain are preserved as in the original geometry. The number of stator passages are reduced to a half, that is, from 72 to 36 to simplify the blocking system. The rotational speed of the rotor is reduced by ten times, that is, from $400rpm$ to $40rpm$. The interface radial position of $I : 2/3$ is chosen for the computations. Figure A.1 shows the the simplified geometry in the computational domain.

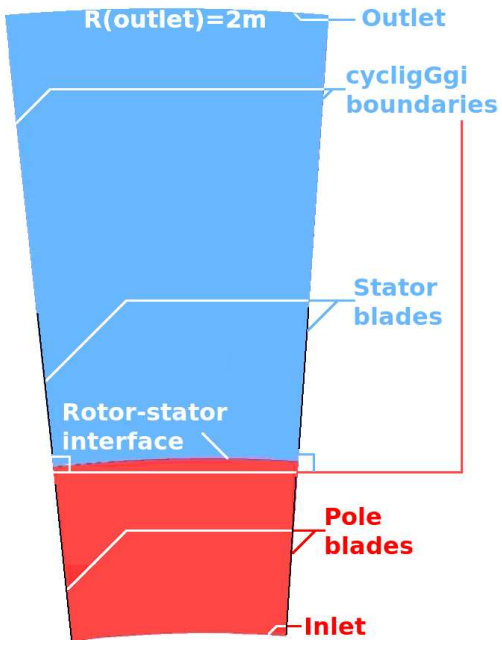

Figure A.1: The simplified geometry in the computational domain.

The simplified cases are simulated in 2D. The simulated cases include a 10^o sector as well as a 360° case. The cases are simulated in sliding grid mode. The results of the simulations follow in the sections below.

A.1 2D-sector

The 10^o sector cases are run in sliding grid modes. The focus in the 2D-sector simulations in sliding grid mode is mainly to investigate the effect of turbulence model, numerical schemes and different solution strategies on the results.

Case Name	Turb. model	time sch.	Adv. Sch.	$P-U$ corr.	Linear solver p	Precond. for p
Base case	$k-\varepsilon$	Euler	linear Upwind	20	BiCGStab	none
50 p-U corrections	$k-\varepsilon$	Euler	linear Upwind	50	BiCGStab	none
Bakwd Euler	$k-\varepsilon$	bckwrd Euler	linear Upwind	20	BiCGStab	none
Upwind Adv	$k-\varepsilon$	Euler	upwind	20	BiCGStab	none
LimLinV Adv	$k-\varepsilon$	Euler	limited Lin.V 1	20	BiCGStab	none
Gamma Adv	$k-\varepsilon$	Euler	Gamma 1	20	BiCGStab	none
RNG KE turb	RNG $k-\varepsilon$	Euler	linear Upwind	20	BiCGStab	none
Rlzble KE turb	realiz. $k-\varepsilon$	Euler	linear Upwind	20	BiCGStab	none
Dgnl prec	$k-\varepsilon$	Euler	linear Upwind	20	BiCGStab	Diagonal
DILU prec	$k-\varepsilon$	Euler	linear Upwind	20	BiCGStab	DILU
PCG solver	$k-\varepsilon$	Euler	linear Upwind	20	PCG	none
Smooth Solver	$k-\varepsilon$	Euler	linear Upwind	20	Smooth Solver	none

Table A.1: Case properties: simplified geometry, 2D-sector, sliding grid with OpenFOAM.

The OpenFOAM cases are simulated with two sliding grid solvers in OpenFOAM-1.6-ext are selected, namely the distributed $pimpleDyMFoam$, using a combined "PISO-Simple" algorithm for pressure-velocity correction, and the in-house developed $transientSimpleDyMFoam$ using the "Simlpe" algorithm for correcting the pressure and the velocity fields. Table A.1 summarizes the properties of the sliding grid cases with simplified geometry. The computational domain in all cases includes $4.5k$ cells which give the best y^+ values. Three CFX cases are run to validate the results: 1^{st} and 2^{nd} order advection with standard $k-\varepsilon$, and 2^{nd} order advection with RNG $k-\varepsilon$.

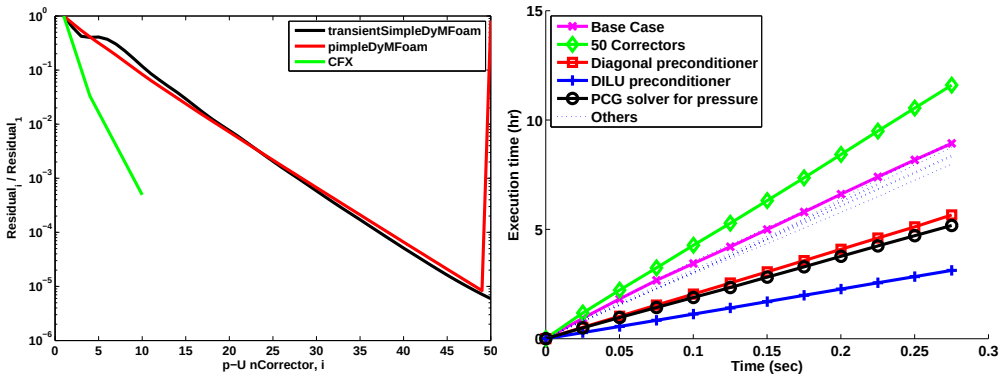

Figure A.2: Left: The average residual reduction after each pressure-velocity correction in the $transientSimpleDyMFoam$, $pimpleDyMFoam$ and CFX cases. Right: The computational times in the OpenFOAM cases.

During a transient simulation, the velocity and the pressure fields are corrected with each other a number of times. This is called the "outer correction". A number of inner iterations are performed during each outer correction loop to achieve low final residuals for the loop. The left side of Figure A.2 shows the average residual reduction after each pressure-velocity outer correction in a case with the OpenFOAM and CFX solvers. The OpenFOAM solvers reduce the residuals more or less equally after each pressure-velocity correction. CFX needs much less pressure-velocity corrections to achieve the same level of residual reductions as in OpenFOAM. The residual values for the $transientSimpleDyMFoam$ solver generally stay lower than those for the $pimpleDyMFoam$. Furthermore, the final pressure-velocity outer correction in the $pimpleDyMFoam$ is evaluated without underrelaxation, in contrast to the other outer corrections which are performed with underrelaxation. This causes the residuals in the final outer correction of the $pimpleDyMFoam$ solver to jump to a higher value close to the first outer correction. The implementation of the $pimpleDyMFoam$ depends on the OpenFOAM version.

The right side of Figure A.2 shows a comparison of the computational times in the OpenFOAM cases. Generally the computational times for the cases with same setups are equal for both OpenFOAM solvers. Using a diagonal or a DILU preconditioner for the pressure matrix results in respectively 40% or 80% reduction in the computational time. Using a PCG matrix solver for the pressure equation reduces the computational time by 40%. The correct choice of the linear solvers and preconditioners for the matrix manipulations can lead to a much faster convergence and save a lot of computational effort. The computational time is not directly proportional to the number of pressure-velocity outer corrections in the time step, as the number of inner iterations in each correction loop reduces after each correction.

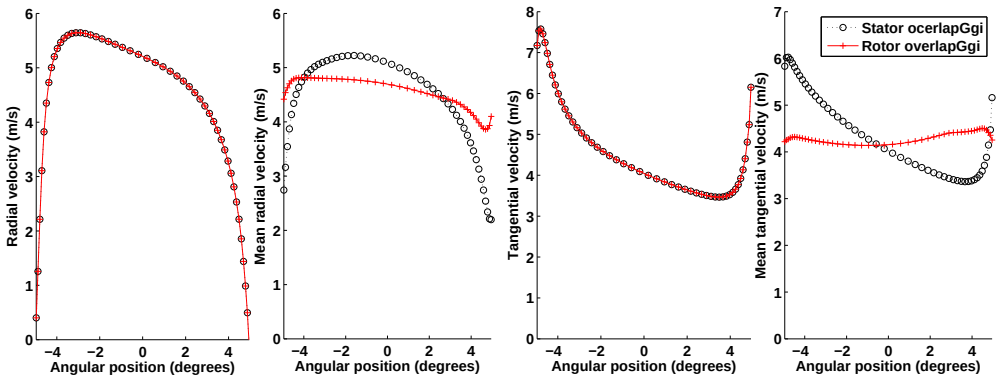

Figure A.3: The velocity profiles at the two sides of the interface in the 2D-sector case with $transientSimpleDyMFoam$ solver and RNG $k-\varepsilon$.

Figure A.3 shows the radial and the tangential velocities at the two sides of the interface in the case with $transientSimpleDyMFoam$ solver and RNG $k-\varepsilon$ turbulence model. The instantaneous velocities are plotted at the moment the two interfaces are in front of each other. The mean velocitis are averaged over an integer number of periods. The instantaneous radial velocities are decelerated by the rotor and the stator walls at the sides. The instantaneous velocities fall on each other at both side of the interface, while the time averaged velocities differ in shape and magnitude. This is because the velocities on the rotating side are more or less constant, while the stationary region experiences moving velocities at each point.

Figure A.4 shows the torque curves in the OpenFOAM and the CFX cases. The curves are not perfectly periodic, and the non-periodicity is associated with the non-periodic behaviour of the flow. The strongest variations in the periods are observed in the OpenFOAM cases with realizable $k-\varepsilon$ turbulence model. The general behavior and magnitudes of the torque curves are similar in both OpenFOAM solvers and CFX.

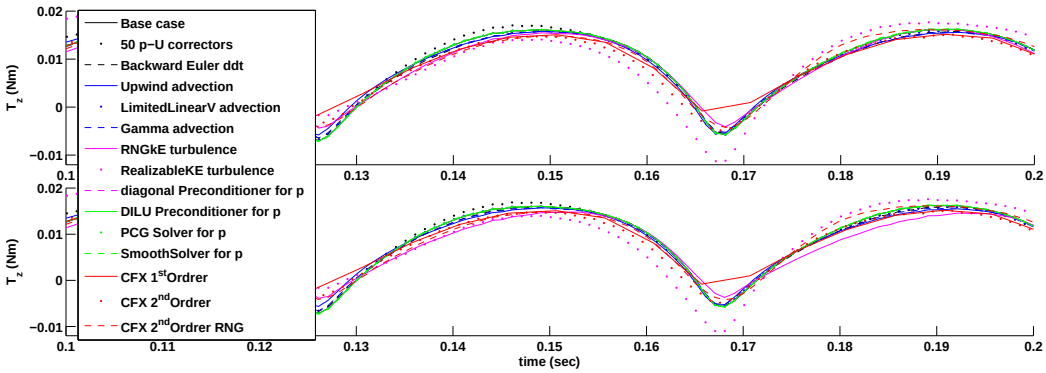

Figure A.4: The computed torque curves in the 2D-sector cases in OpenFOAM and CFX. Top: $transientSimpleDyMFoam$**. Bottom:** $pimpleDyMFoam$**.**

Case Name	Torque (Nm)		
	OpenFOAM		CFX
	$pimple DyMFoam$	$transientSimple DyMFoam$	
Base case	0.0084	0.0084	
50 p-U corrections	0.0083	0.0084	
Bakwd Euler	0.0084	0.0085	
Upwind adv.	0.0087	0.0087	
limLin. V1	0.0083	0.0084	
Gamma adv.	0.0083	0.0084	
RNG KE turb	0.0083	0.0083	
Rlzble KE turb	0.0078	0.0079	
Dgnl. prec.	0.0084	0.0084	
DILU prec.	0.0084	0.0084	
PCG solver	0.0084	0.0084	
Smooth solver	0.0083	0.0084	
1^{st} Order adv.			0.0088
2^{nd} Order adv.			0.0085
RNG+2^nd Order adv.			0.0084

Table A.2: The mean torques in the 2D-sector OpenFOAM and CFX cases.

Table A.2 shows the mean torques in the $transientSimpleDyMFoam$, $pimpleDyMFoam$ and CFX cases, averaged over a number of periods. Generally the mean torque values are close in the OpenFOAM and the CFX cases. However, certain numerical shcemes and turbulence models result in slightly different values. In the presented CFX cases, the choice of the turbulence model plays a minor role in the average torques obtained. A stronger variation in the CFX results is observed by changing the advection scheme. The first-order advection scheme gives a 5% difference in the mean torque compared to the CFX cases with the second-order advection scheme. The OpenFOAM cases generally demonstrate consistent results with the CFX cases. The OpenFOAM cases with realizable $k-\varepsilon$ turbulence model and upwind advection scheme show respectively about 6% and 4% deviation from the rest of the cases.

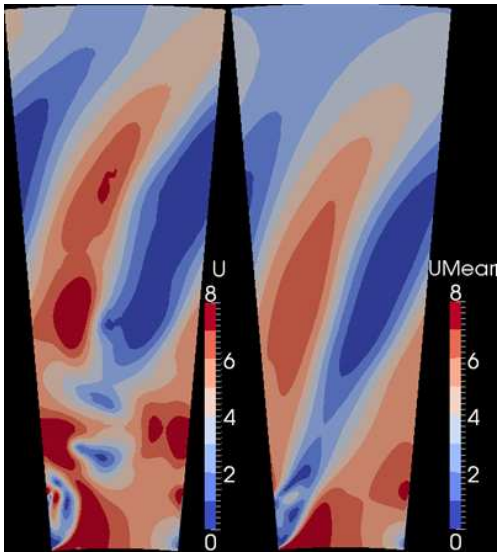

Figure A.5: The velocity fields in the stator region in the $transientSimpleDyMFoam$ 2D-sector case with RNG $k-\varepsilon$ turbulence model. Left: the instantaneous velocities. Right: the time averaged velocities.

Figure A.5 shows the instantaneous and the time averaged velocities in the stator region in the $transientSimpleDyMFoam$ case with the RNG $k-\varepsilon$ turbulence model. The instantaneous velocity field is different from the averaged velocity field, which means that the flow is unsteady in the computaional domain. A recirculation region, characterized by low velocities is built up at the downstream side of the stator blades.

A.2 2D-complete

The complete case is designed to investigate the computational performance of the ggi interface. The rotor and the stator consist of radial blades as in the sector cases. The rotational speed of the rotor is $40rpm$, which together with 36 stator blades gives a period of rotor movement in front of each stator stage of about $0.0417s$. The choice of the turbulence model and the numerical setup is performed by the knowledge obtained from the sector computations. The rotor-stator interface boundary condition is set to ggi and the outlet boundary is set closer to the stator deflectors, in order to reduce the number of cells in the computational domain. There are a total of 100440 cells in the domain which keeps the y^+ values in the interval between 30 and 100 with the standard $k-\varepsilon$ turbulence model. The inlet and outlet boundary conditions are the same as in the sector simulations. The computation is performed using the $transientSimpleDyMFoam$ solver, with the first-order $Euler$ time scheme and $LimitedLinearV1$ advection scheme. 50 pressure-velocity outer corrections loops are used in each time step and all matrices are solved utilizing the $BiCGStab$ linear solver and $DILU$ preconditioning. The computational domain is divided between 16 CPUs.

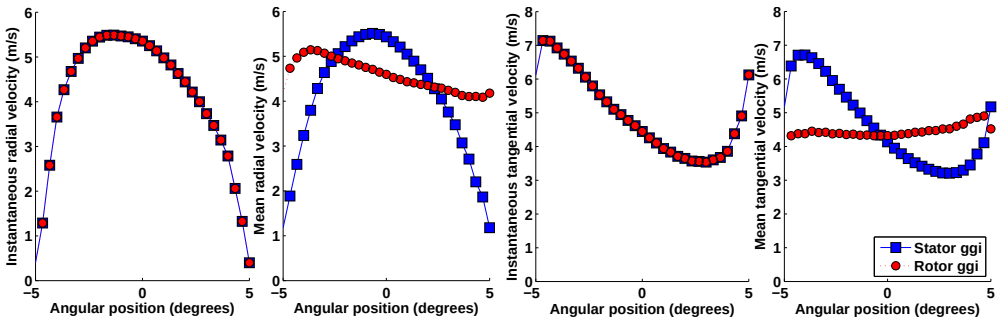

Figure A.6: The velocity profiles at the two sides of the interface in the 2D-complete case.

The residuals of the pressure equation decrease roughly by one order of magnitude after each ten corrections of the pressure and velocity fields in each time step. The residuals in other equations decrease twice as fast. Figure A.6 shows the radial the and tangential velocities at the two sides of the interface. The instantaneous velocities fall on each other at both sides of the interface, while the time averaged velocities differ in shape and magnitude.

Figure A.7 shows the computed torque curves over the last few periods. The curves are not perfectly periodic in time. The diagram shows the torque curves after having covered more than 100 periods. The

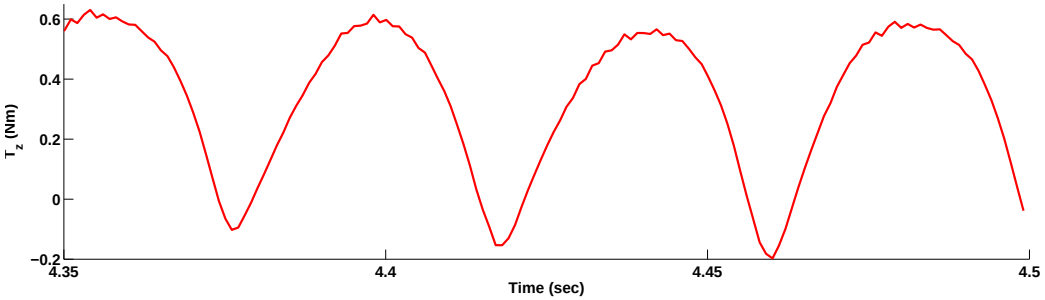

Figure A.7: The computed torque curves for the 2D-complete case.

time-averaged axial torque on the rotor (with 36 blades) over the period of $3 - 4.5$ seconds (one whole revolution, corresponding to 36 periods) is 0.322 (Nm), which for a single rotor blade would be 0.0089 (Nm). This means a difference of about 7% with the corresponding sector case (50 pressure-velocity corrections, $transientSimpleDyMFoam$). It should be noticed that the mesh resolution is adjusted and that the outlet boundaries are not located in the same radii in the two cases.

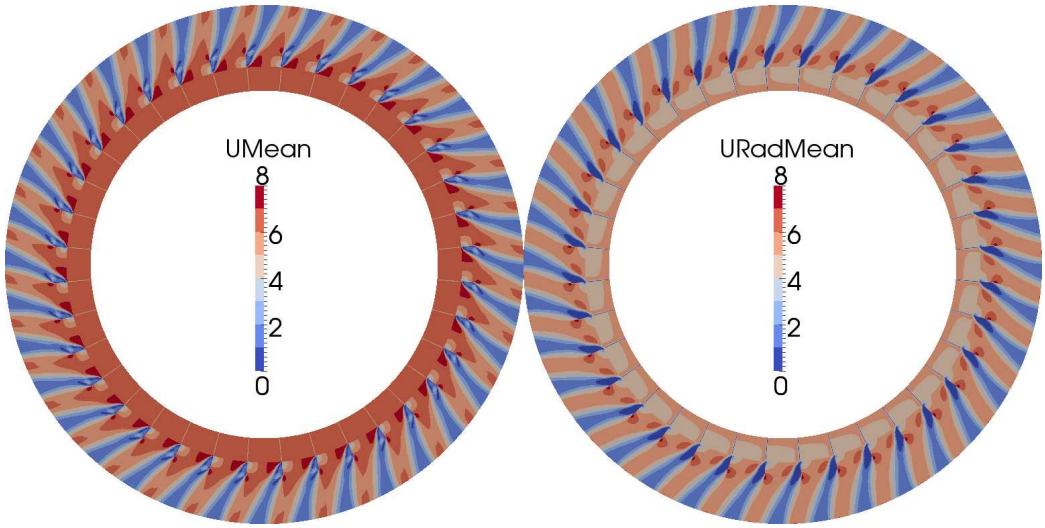

Figure A.8: The time averaged velocity contours for the 2D-complete case. Left: velocity magnitude. Right: radial velocity.

Figure A.8 shows the time averaged velocity magnitude and the time averaged radial velocity fields. In the stationary region mostly radial velocities are present and the tangential components are much smaller, compared to the rotating region where the tangential velocity is more important.

Bibliography

[1] S. Houde, C. Hudon, and P.B. Vincent. Simulation strategies of the cooling flow for large hydro-generators. *Hydropower & Dams*, 19, 2008.

[2] P. Moradnia. *CFD of air flow in hydro power generators*. Chalmers University of Technology, 2001.

[3] P. Moradnia and H. Nilsson. Cfd of air flow in hydro power generators for convective cooling, using openfoam. In *V European Conference on Computational Fluid Dynamics, ECCOMAS CFD 2010*, 2010.

[4] P. Moradnia and H. Nilsson. A parametric study of the air flow in an electric generator through stepwise geometry modifications. In *An ECCOMAS Thematic Conference, CFD and OPTIMIZATION*, 2011.

[5] P. Moradnia, V. Chernoray, and H. Nilsson. Experimental and numerical investigation of the cooling air flow in an electric generator. In *HEFAT 2011 International Conference on Heat Transfer, Fluid Mechanics and Thermodynamics*, pages 242–249, 2011.

[6] K. Toussaint, F. Torriano, J.F. Morissette, C. Hudon, and M. Reggio. Cfd analysis of ventilation flow for a scale model hydrogenerator. In *ASME Power*, 2011.

[7] H. Keck and M. Sick. Thirty years of numerical ow simulation in hydraulic turbomachines. *Acta Mech*, 201:211–229, 2008.