

Evaluation of RANS turbulence models for flow problems with significant impact of boundary layers

Eric Furbo



UPPSALA
UNIVERSITET

Teknisk- naturvetenskaplig fakultet
UTH-enheten

Besöksadress:
Ångströmlaboratoriet
Lägerhyddsvägen 1
Hus 4, Plan 0

Postadress:
Box 536
751 21 Uppsala

Telefon:
018 – 471 30 03

Telefax:
018 – 471 30 00

Hemsida:
<http://www.teknat.uu.se/student>

Abstract

Evaluation of RANS turbulence models for flow problems with significant impact of boundary layers

Eric Furbo

This master's thesis was provided by the Swedish Defence Research Agency, FOI. The task is to test several RANS (Reynolds-averaged Navier-Stokes) models on two different case geometries and compare the results with LES and experimental data. The first is two dimensional, constructed for flow separation at a sharp edge. The second is three dimensional and flow separation occurs at a smooth surface. The models tested are implemented in the open source CFD (Computational Fluid Dynamics) program, OpenFOAM. OpenFOAM uses the finite volume method and the SIMPLE algorithm as solution procedure. The main flow features evaluated is the shape, position and size of the flow separation. Most of the models tested have problems describing the complex dynamics of flow separation in these particular cases. In addition to the simulations, the RANS k-epsilon turbulence model is presented and the RANS equations and the equation for the turbulent kinetic energy are derived from the Navier-Stokes equations. The theory behind wall functions is described and these equations together with the equations in the k-epsilon model are compared with the equations implemented in OpenFOAM.

Handledare: Mattias Liefvendahl
Ämnesgranskare: Gunilla Kreiss
Examinator: Tomas Nyberg
ISSN: 1401-5757, UPTEC F10061

Acknowledgements

I would like to thank Mattias Liefvendahl, my supervisor at FOI, for providing me with the subject of this thesis, and for using his time to guide and help me. I would also like to thank Gunilla Kreiss at Uppsala University, who is the examiner of this thesis.

Contents

1	Introduction	5
2	Theory of turbulence modelling	5
2.1	The Reynolds equation	6
2.2	The turbulent-viscosity hypothesis	7
2.3	The energy cascade	8
2.4	The equation for turbulent kinetic energy	8
2.5	Closure approximations	9
2.6	The $k - \epsilon$ model	10
2.7	Shear stress near walls	11
2.8	The log law	12
2.9	Turbulence modelling near walls	12
3	Implementation of turbulence models in OpenFOAM	14
3.1	The momentum equation	14
3.2	The k -equation	16
3.3	The ϵ -equation	18
3.4	The turbulent viscosity ν_T	18
3.5	Implementation of wall models	19
4	Numerical solution method	20
4.1	The finite volume method (FVM)	20
4.2	The SIMPLE algorithm	20
4.3	Under-relaxation	21
5	Different types of turbulence models used in our simulations	21
5.1	Linear eddy viscosity models	22
5.2	Reynolds stress transport models	22
5.3	Non-linear eddy viscosity models	22
6	The Pitz-Daily case	22
6.1	Different meshes	23
6.2	Boundary conditions for turbulent quantities	24
6.2.1	The k -equation	24
6.2.2	The ϵ -equation	24
6.2.3	The equation for the Reynolds stress tensor	24
6.2.4	The ω -equation	25
6.3	Mesh convergence for the Pitz-Daily case	25
6.4	Results from the Pitz-Daily case simulations	27
6.5	Results from simulations on the M_{t1} mesh with WFBC	27
6.6	Results from simulations on the M_r mesh, without WFBC	28
6.6.1	The $k - \epsilon$ model	29
6.6.2	The realizable $k - \epsilon$ model	29
6.7	Solution of the realizable $k - \epsilon$ model on the M_{t3} mesh	29
6.8	Discussion and conclusions of the Pitz-Daily results	30

7	The bump case	33
7.1	Previous Research	34
7.2	Boundary conditions for the turbulent quantities	35
7.3	Mesh generation	35
7.4	Short description of the physical experiment	36
7.5	Results from the bump case simulations	37
7.6	The realizable $k - \epsilon$ model	38
7.6.1	Results from simulation on the M_{r1} mesh	38
7.6.2	Results from simulation on the M_{r2} mesh	39
7.6.3	Results from simulation on the M_{u1} mesh	40
7.6.4	Results from simulation on the M_{u2} mesh	41
7.7	The RNG $k - \epsilon$ model	42
7.7.1	Results from simulation on the M_{r1} mesh	42
7.7.2	Results from simulation on the M_{u2} mesh	42
7.8	The LRR model	43
7.8.1	Results from simulation on the M_{r1} mesh	43
7.8.2	Results from simulation on the M_{u2} mesh	43
7.9	Comparison to LES and LDV data	44
7.10	Discussion and conclusions of the bump results	46
8	Summary and general conclusions	50

List of Figures

1	The Pitz-Daily case geometry	23
2	The two different types of meshes used for the Pitz-Daily case	23
3	Lines through the Pitz-Daily domain where data are taken.	25
4	Order of accuracy for the Pitz-Daily case.	26
5	Profiles of u_1/V_∞ from three different Pitz-Daily meshes.	26
6	Pitz-Daily results from models tested on the M_{t1} mesh.	28
7	Pitz-Daily results from models tested on the M_r mesh.	29
8	Pitz-Daily results from realizable $k-\epsilon$ on the M_{t3} mesh.	30
9	Comparison of Pitz-Daily results from different RANS models.	31
10	Comparison of Pitz-Daily results from different RANS models.	32
11	The bump case geometry.	33
12	The bump case inlet velocity profile.	34
13	The two different types of meshes used for the bump case	36
15	Results from the realizable $k - \epsilon$ model on the M_{r1} mesh.	38
16	Results from the realizable $k - \epsilon$ model on the M_{r1} mesh.	39
17	Results from the realizable $k - \epsilon$ model on the M_{r2} mesh.	39
18	Results from the realizable $k - \epsilon$ model on the M_{r2} mesh.	40
19	Results from the realizable $k - \epsilon$ model on the M_{u1} mesh.	40
20	Results from the realizable $k - \epsilon$ model on the M_{u2} mesh.	41
21	Results from the realizable $k - \epsilon$ model on the M_{u2} mesh.	41
22	Results from the realizable $k - \epsilon$ model on the M_{u2} mesh.	42
23	Results from the RNG $k - \epsilon$ model on the M_{r1} mesh.	42
24	Results from the RNG $k - \epsilon$ model on the M_{r1} mesh.	43
25	Results from the RNG $k - \epsilon$ model on the M_{u2} mesh.	43
26	Results from the RNG $k - \epsilon$ model on the M_{u2} mesh.	44
27	Results from the LRR model on the M_{r1} mesh.	44
28	Results from the LRR model on the M_{r1} mesh.	45
29	Results from the LRR model on the M_{u2} mesh.	45
30	Results from the LRR model on the M_{u2} mesh.	46
32	Comparison of bump results from RANS models, LES and LDV.	47
33	Comparison of bump results from RANS models, LES and LDV.	48
34	Bump results from the realizable $k - \epsilon$ model in the plane $x_1/H = 3.69$	49
35	Bump results from the realizable $k - \epsilon$ model in the plane $x_1/H = 3.69$	50
36	Bump results from the LRR model in the plane $x_1/H = 3.69$	51
37	Bump results from the RNG $k - \epsilon$ model in the plane $x_1/H = 3.69$	52

List of Tables

1	Meshes used for the Pitz-Daily case	24
2	Models tested on the M_{t1} mesh.	27
3	Models tested on the M_r mesh.	29
4	Results for the Pitz-Daily case.	30
5	Meshes used for the bump case	36
6	Tested turbulence models and meshes for the bump case.	37
7	Table of separation and re-attachment points.	45

1 Introduction

Computational prediction of flow separation from turbulent flows is a process of primary concern. The physical phenomena that arise is a subject of interest for many engineering components and systems. Streamlined car bodies, low-pressure turbine blades and highly loaded aircraft wings are some examples of where flow separation can have significant influence in ability of the device in question to perform effectively. Turbulent flows fluctuate on a broad range of time and length scales. This makes the simulation of such flows difficult and it is often necessary to model the turbulence in some way. RANS (Reynolds-average Navier-Stokes) are such model equations and are used to describe the flow field in this work.

The main objective with this master's thesis is to evaluate different RANS turbulence models. Three families of models have been tested; Linear eddy viscosity models, Reynolds stress transport models and Non-linear eddy viscosity models.

For the evaluation, we have two different geometries. The first domain is a two dimensional backward facing step, where the flow is separated due to the sharp edge of the step. (See figure 1.) Even though the geometry is simple it is not obvious where the re-attachment will take place. When evaluating the result, we mainly focus on the re-attachment point of the flow and the position of the recirculation region, arising from the separation. The simulations on this geometry are carried out on different meshes with different types of boundary conditions and the results are compared.

The second domain is a three dimensional hill, constructed to be very smooth without edges. On this geometry, the fluid flow becomes detached from the surface and instead takes the form of eddies and vortices. Despite the simple shape (see figure 11) of the hill, it is a challenging computational problem and experiments indicate that the flow in the recirculation zone is complex and strongly time-dependent.

On this second domain, the RANS turbulence models will be simulated using two different types of meshes. We mainly evaluate the results with respect to the location and shape of the separation zone and the velocity distribution in the near wake of the wall. Our results will be compared to results from experiments made by Byun and Simpson [1] and LES simulations made by Persson et al. [2]

In order to simulate the turbulent flow and separation, the open source program OpenFOAM was used.

2 Theory of turbulence modelling

The text in the following chapter is based on the books Turbulent Flows by Stephen B. Pope, [3] and Computational Methods for Fluid Dynamics by J. H. Ferziger and M. Perić, [4].

The fundamental basis of fluid dynamics are the Navier-Stokes equations. The incompressible form of these equations and the incompressible continuity equation are described as

$$\frac{Du_i}{Dt} \equiv \frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j} \quad (2.1)$$

and

$$\frac{\partial u_j}{\partial x_j} = 0, \quad (2.2)$$

where x_i ($i = 1, 2, 3$) are the cartesian coordinates, in this report also written (x, y, z) , u_i are the cartesian components of the velocity, t is the time, p is the pressure, ρ is the density and ν is the dynamic viscosity, defined as the viscosity μ divided by ρ .

Here and throughout this report, whenever the same index appears twice in any term, summation over the range of that index is implied. For example, the incompressible continuity equation:

$$\frac{\partial u_j}{\partial x_j} = \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} + \frac{\partial u_3}{\partial x_3} = 0. \quad (2.3)$$

2.1 The Reynolds equation

In the RANS (Reynolds averaged Navier-Stokes) approach to turbulence, all of the unsteadiness in the flow is averaged out and regarded as part of the turbulence. The flow variables, in this example one component of the velocity, are represented as the sum of two terms:

$$u_i(x_i, t) = \bar{u}_i(x_i) + u'_i(x_i, t), \quad (2.4)$$

where

$$\bar{u}_i(x_i) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T u_i(x_i, t) dt. \quad (2.5)$$

Here T is the averaging interval and must be large compared to the typical time scale of the fluctuations and u'_i is the fluctuation about the time averaged value.

If the flow is unsteady, time averaging cannot be used and it has to be replaced with *ensemble averaging*. The concept of this is to imagine a set of flows in which all of the variables that can be controlled (energy, boundary conditions etc.) are identical but the initial conditions are generated randomly. This will give flows that differ considerably from one another. An average over a large set of such flows is an ensemble average; In mathematical form written

$$\bar{u}_i(x_i, t) = \frac{1}{N} \sum_{n=1}^N u_{ni}(x_i, t), \quad (2.6)$$

where N is the number of members of the ensemble. The term *Reynolds averaging* refers to any of the processes above and applying it to the incompressible continuity equation gives

$$\frac{\partial \bar{u}_j}{\partial x_j} = 0. \quad (2.7)$$

Taking the mean of the incompressible momentum equation is not as straight forward because of the nonlinearity of the convective term. Taking the mean of the left hand side of equation (2.1) is written

$$\overline{\frac{D u_i}{D t}} = \frac{\partial \bar{u}_i}{\partial t} + \frac{\partial (\bar{u}_i u'_j)}{\partial x_j}. \quad (2.8)$$

Using decomposition (2.4) for the nonlinear term result in

$$\begin{aligned} \overline{u_i u'_j} &= \overline{(\bar{u}_i + u'_i)(\bar{u}_j + u'_j)} \\ &= \overline{\bar{u}_i \bar{u}_j} + \overline{u'_i \bar{u}_i} + \overline{u'_i \bar{u}_j} + \overline{u'_i u'_j} \\ &= \overline{\bar{u}_i \bar{u}_j} + \overline{u'_j \bar{u}_i} + \overline{u'_i \bar{u}_j} + \overline{u'_i u'_j} \\ &= \overline{\bar{u}_i \bar{u}_j} + \overline{u'_i u'_j}, \end{aligned} \quad (2.9)$$

since

$$\overline{u'_j \bar{u}_i} = \overline{u'_j \bar{u}_i} = 0. \quad (2.10)$$

If we use the result from equation (2.9) together with equation (2.8) we get

$$\frac{\overline{Du_i}}{Dt} = \frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} + \bar{u}_i \frac{\partial \bar{u}_j}{\partial x_j} + \frac{\partial (\overline{u'_i u'_j})}{\partial x_j}. \quad (2.11)$$

Since the mean velocity field is incompressible, (2.11) simplifies to

$$\frac{\overline{Du_i}}{Dt} = \frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial (\overline{u'_i u'_j})}{\partial x_j}. \quad (2.12)$$

Taking the mean of the rest of the terms in the momentum equation is simple since the spatial derivative commutes with the operation of taking average. The result is the *Reynolds* (or *RANS*) equation.

$$\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} - \frac{\partial \overline{u'_i u'_j}}{\partial x_j}. \quad (2.13)$$

Equation (2.13) can be rewritten as

$$\rho \left(\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} \right) = \frac{\partial}{\partial x_j} \left[-\bar{p} \delta_{ij} + \mu \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \rho \overline{u'_i u'_j} \right]. \quad (2.14)$$

The term in square brackets represents the sum of three stresses; $\bar{p} \delta_{ij}$ from the mean pressure field, the viscous stress from momentum transfer at molecular level and the stress term $-\rho \overline{u'_i u'_j}$, arising from the fluctuating velocity field. This term is called *Reynolds stresses*. (In this report, $\overline{u'_i u'_j}$ will also be referred to as Reynolds stresses.)

The Reynolds stresses are components of a symmetric second-order tensor. The diagonal components are normal stresses and the off-diagonal components are shear stresses. The *turbulent kinetic energy*, k is half the trace of the Reynolds stress tensor

$$k = \frac{1}{2} \rho \overline{u'_i u'_i} \quad (2.15)$$

and the isotropic stress is defined as $\frac{2}{3} k \delta_{ij}$. Then the deviatoric part is

$$a_{ij} = \overline{u'_i u'_j} - \frac{2}{3} k \delta_{ij}. \quad (2.16)$$

Because of the symmetry of the Reynolds stress tensor there are six independent elements of the tensor and therefore six more unknowns. To close the system, i.e. get equal number of unknowns and equations, we have to model the Reynolds stresses in some way.

2.2 The turbulent-viscosity hypothesis

The turbulent-viscosity hypothesis was introduced by Boussinesq in 1877 and is analogous to the stress-rate-of-strain relation for a Newtonian fluid. According to the hypotheses the relationship is

$$-\overline{u'_i u'_j} = \nu_T \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} k \delta_{ij}, \quad (2.17)$$

where the positive scalar field $\nu_T = \nu_T(x_i, t)$ is the *turbulent viscosity*. The turbulent-viscosity hypothesis substituted into equation (2.13) is

$$\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = \frac{\partial}{\partial x_j} \left[\nu_{eff} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] - \frac{1}{\rho} \frac{\partial}{\partial x_i} (\bar{p} + \frac{2}{3} \rho k), \quad (2.18)$$

where

$$\nu_{eff}(x_i, t) = \nu + \nu_T(x_i, t), \quad (2.19)$$

is the *effective viscosity*.

Equation (2.18) has the same appearance as the incompressible Navier-Stokes equation with \bar{u}_i and ν_{eff} in place of u_i and ν and with $\bar{p} + \frac{2}{3}\rho k$ modifying the pressure. The advantage with this model of is that it is fairly simple. Unfortunately, for many flows the accuracy of the model is poor, [3]. This shows that the physics of turbulence is different from the physics of the molecular processes that lead to the relation for the viscous stress in a Newtonian fluid. However, for simple shear flows, where the mean velocity gradients and turbulence characteristics develop slowly, the hypothesis is quite reasonable.

2.3 The energy cascade

Turbulence is considered to be composed of eddies of different sizes. The largest eddies of the flow are unstable and break up, transferring their energy to smaller eddies. These smaller eddies also break up and transfer energy to yet smaller eddies. This *energy cascade* continues until the Reynolds number $Re(l) \equiv u(l)l/\nu$ is sufficiently small so that the eddy motion is stable and molecular viscosity is effective in dissipating the kinetic energy. Here l and $u(l)$ are the characteristic length scale and velocity scale of these stable eddies.

This is of importance because it places the dissipation of energy at the end of the energy cascading process. The *rate* of dissipation, denoted ϵ , is determined by the first process in the sequence, which is the transfer of energy from the largest eddies. These eddies are characterized by the lengthscale l_0 , the velocity scale u_0 , the time scale $\tau_0 = l_0/u_0$ and have energy of $\frac{1}{2}\rho u_0^2$. Then the rate of transfer of energy can be supposed to scale as $u_0^2/\tau_0 = u_0^3/l_0$. Consequently, ϵ scales as u_0^3/l_0 , independent of ν .

2.4 The equation for turbulent kinetic energy

In this section a differential equation describing the behavior of the turbulent kinetic energy, k , is derived. Starting by multiplying the incompressible Navier-Stokes equations with u_i and then taking the average of the result yields

$$\overline{\frac{\partial u_i}{\partial t} u_i} + \overline{u_j \frac{\partial u_i}{\partial x_j} u_i} = -\frac{1}{\rho} \overline{\frac{\partial p}{\partial x_i} u_i} + \nu \overline{u_i \nabla^2 u_i}. \quad (2.20)$$

Multiplying the Reynolds equation by \bar{u}_i gives

$$\overline{\frac{\partial \bar{u}_i}{\partial t} \bar{u}_i} + \overline{\bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} \bar{u}_i} = -\frac{1}{\rho} \overline{\frac{\partial \bar{p}}{\partial x_j} \bar{u}_i} + \nu \overline{\bar{u}_i \nabla^2 \bar{u}_i}. \quad (2.21)$$

Some rules due to averaging are

$$\overline{u_i u_j} = \bar{u}_i \bar{u}_j + \overline{u'_i u'_j} \quad (2.22)$$

$$\overline{\frac{\partial u_i}{\partial x_j}} = \frac{\partial \bar{u}_i}{\partial x_j} \quad (2.23)$$

$$\overline{\frac{\partial u_i}{\partial x_j} u_i} = \frac{\partial \bar{u}_i}{\partial x_j} \bar{u}_i + \overline{\frac{\partial u'_i}{\partial x_j} u'_i} \quad (2.24)$$

$$\overline{u_i u_j u_k} = \overline{u'_i u'_j u'_k} + \overline{u'_i u'_j} \bar{u}_k + \overline{u'_j u'_k} \bar{u}_i + \overline{u'_k u'_i} \bar{u}_j + \bar{u}_i \bar{u}_j \bar{u}_k. \quad (2.25)$$

Subtracting (2.21) from (2.20) we get

$$\rho \overline{\frac{\partial u'_i}{\partial t} u'_i} + \rho \left(\overline{u_j \frac{\partial u_i}{\partial x_j} u_i} - \overline{u_j} \frac{\partial \bar{u}_i}{\partial x_j} \bar{u}_i \right) = - \overline{\frac{\partial p'}{\partial x_i} u'_i} + \nu \overline{u'_i \nabla^2 u'_i} + \rho \frac{\partial \overline{u'_i u'_j}}{\partial x_j} \bar{u}_i. \quad (2.26)$$

From the averaging rules we have

$$\overline{u_j \frac{\partial u_i}{\partial x_j} u_i} - \overline{u_j} \frac{\partial \bar{u}_i}{\partial x_j} \bar{u}_i = \overline{u'_i \frac{\partial u'_j}{\partial x_j} \bar{u}_j} + \overline{\frac{\partial u'_i}{\partial x_j} u'_i u'_j} + \overline{\frac{\partial u'_i}{\partial x_j} u'_j \bar{u}_i} + \overline{u'_j u'_i} \overline{\frac{\partial u_i}{\partial x_j}}. \quad (2.27)$$

Using equation (2.26), equation (2.27), the chain rule for derivatives and the incompressibility of the mean velocity field, we end up with

$$\rho \left(\overline{\frac{\partial u'_i}{\partial t} u'_i} + \overline{u'_i \frac{\partial u'_j}{\partial x_j} \bar{u}_j} + \overline{\frac{\partial u'_i}{\partial x_j} u'_i u'_j} + \overline{\frac{\partial u'_i u'_j}{\partial x_j} \bar{u}_i} + \overline{u'_j u'_i} \overline{\frac{\partial u_i}{\partial x_j}} \right) = - \overline{\frac{\partial p'}{\partial x_i} u'_i} + \nu \overline{u'_i \nabla^2 u'_i} + \rho \frac{\partial \overline{u'_i u'_j}}{\partial x_j} \bar{u}_i. \quad (2.28)$$

The fourth term on the left hand side and the last term on the right hand side are equal and cancel out. Using the chain rule again we get

$$\frac{1}{2} \left(\overline{\frac{\partial u'_i u'_i}{\partial t}} + \overline{u'_j} \frac{\partial \overline{u'_i u'_i}}{\partial x_j} + \overline{\frac{\partial (u'_i u'_i u'_j)}{\partial x_j}} \right) = - \overline{u'_j u'_i} \overline{\frac{\partial u_i}{\partial x_j}} - \frac{1}{\rho} \overline{\frac{\partial p' u'_i}{\partial x_i}} + \nu \overline{\frac{\partial}{\partial x_j} \left(\frac{1}{2} \overline{\frac{\partial (u'_i u'_i)}{\partial x_j}} \right)} - \overline{\frac{\partial u'_i}{\partial x_j} \frac{\partial u'_i}{\partial x_j}}. \quad (2.29)$$

Using the definition of the turbulent kinetic energy according to equation (2.15) and the defintion of the dissipation rate of turbulent energy¹, given by

$$\epsilon = \nu \overline{\frac{\partial u'_i}{\partial x_j} \frac{\partial u'_i}{\partial x_j}}, \quad (2.30)$$

we end up with

$$\frac{\partial k}{\partial t} + \overline{u_j} \frac{\partial k}{\partial x_j} = - \frac{\partial}{\partial x_j} \left(\frac{1}{2} \overline{u'_i u'_i u'_j} + \frac{1}{\rho} \overline{u'_j p'} - \nu \overline{\frac{\partial k}{\partial x_j}} \right) - \overline{u'_j u'_i} \overline{\frac{\partial \bar{u}_i}{\partial x_j}} - \epsilon. \quad (2.31)$$

The sum of the two terms on the left-hand side, the unsteady term and the convection, is the material derivative of k that gives the rate of change of k following a fluid element. The first term on the right-hand side is known as the *turbulent transport* and is regarded as the rate at which turbulence energy is transported through the fluid by turbulent fluctuations. The second term on the right-hand side is called *pressure diffusion* and is another form of turbulent transport resulting from correlation of pressure and velocity fluctuations. The third term on the right-hand side represents the diffusion of turbulence energy caused by the fluids natural molecular transport process. The fourth term on the right-hand side is known as the *production* and represents the rate at which kinetic energy is transferred from the mean flow to the turbulence. Finally, ϵ is the dissipation rate of the turbulent kinetic energy.

2.5 Closure approximations

The left-hand side of equation (2.31) and the term representing the molecular diffusion are exact while production, dissipation, turbulent transport and pressure diffusion involve unknown correlations. To close the equation these terms have to be approximated. The standard way to

¹Here it is assumed that the turbulence is homogeneous.

approximate turbulent transport of scalar quantities is to use the *gradient-diffusion hypothesis*. In analogy with molecular transport processes, we say that $-\overline{u'_j \phi'} \sim \mu_T \frac{\partial \phi}{\partial x_j}$, where ϕ is some conserved scalar, p or k for example. Thus,

$$\frac{1}{2} \overline{u'_i u'_i u'_j} + \frac{1}{\rho} \overline{u'_j p'} = -\frac{\nu_T}{\sigma_k} \frac{\partial k}{\partial x_j}, \quad (2.32)$$

where σ_k is the *turbulent Prandtl number* for kinetic energy and is generally taken to be equal to unity. The gradient-diffusion approximation asserts that there is a flux of k down the gradient of k . Mathematically, it ensures that the solutions are smooth and that a boundary condition can be imposed on k everywhere on the boundary. Using this model for the turbulent transport and the pressure diffusion and using equation (2.17) for the production term, we end up with the following model transport equation for k :

$$\frac{\partial k}{\partial t} + \bar{u}_j \frac{\partial k}{\partial x_j} = \frac{\partial}{\partial x_j} \left[\left(\nu + \frac{\nu_T}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right] + \left[\nu_T \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} k \delta_{ij} \right] \frac{\partial \bar{u}_i}{\partial x_j} - \epsilon. \quad (2.33)$$

Since the mean velocity field is incompressible,

$$\frac{2}{3} k \delta_{ij} \frac{\partial \bar{u}_i}{\partial x_j} = \frac{2}{3} k \frac{\partial \bar{u}_j}{\partial x_j} = 0, \quad (2.34)$$

equation (2.33) reduces to

$$\frac{\partial k}{\partial t} + \bar{u}_j \frac{\partial k}{\partial x_j} = \frac{\partial}{\partial x_j} \left[\left(\nu + \frac{\nu_T}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right] + \nu_T \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \frac{\partial \bar{u}_i}{\partial x_j} - \epsilon. \quad (2.35)$$

According to section 2.3, the dissipation rate scales as u_0^3/l_0 . Therefore it is reasonable to model ϵ as

$$\epsilon = C_D k^{3/2} / l(x_i) \quad (2.36)$$

where $l(x_i)$ is the length scale of the turbulence and C_D is a closure constant.

In the turbulent-viscosity hypothesis, equation (2.17), the turbulent viscosity ν_T is introduced. To close the system of equations it has to be specified. Based entirely on dimensional arguments, the turbulent viscosity is given by

$$\nu_T = k^{1/2} l(x_i). \quad (2.37)$$

2.6 The $k - \epsilon$ model

To eliminate the need for specifying the turbulent length scale $l(x_i)$, in addition to the k -equation, a transport equation for one more turbulence quantity can be used. This type of models is called two-equation models and the standard one is the $k - \epsilon$ model. In this model, a transport equation is solved for ϵ . The exact equation for ϵ can be derived in a similar manner as the k -equation, but it is not a useful starting point for a model equation. As discussed earlier, this is because ϵ is best viewed as the turbulent energy flow rate in the beginning of the energy cascade, which is the transfer of energy from the largest eddies in the flow. In contrast, the exact equation for ϵ belongs to processes in the dissipative range, in the end of the cascade. So the standard model equation for ϵ is best viewed as being entirely empirical. The equation is

$$\frac{\partial \epsilon}{\partial t} + \bar{u}_j \frac{\partial \epsilon}{\partial x_j} = \frac{\partial}{\partial x_j} \left(\frac{\nu_T}{\sigma_\epsilon} \frac{\partial \epsilon}{\partial x_j} \right) + C_{\epsilon 1} \frac{\epsilon}{k} \left[\nu_T \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} k \delta_{ij} \right] \frac{\partial \bar{u}_i}{\partial x_j} - C_{\epsilon 2} \frac{\epsilon^2}{k}. \quad (2.38)$$

Due to the incompressibility of the mean flow field, this expression simplifies to

$$\frac{\partial \epsilon}{\partial t} + \bar{u}_j \frac{\partial \epsilon}{\partial x_j} = \frac{\partial}{\partial x_j} \left(\frac{\nu_T}{\sigma_\epsilon} \frac{\partial \epsilon}{\partial x_j} \right) + C_{\epsilon 1} \frac{\epsilon}{k} \nu_T \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \frac{\partial \bar{u}_i}{\partial x_j} - C_{\epsilon 2} \frac{\epsilon^2}{k}. \quad (2.39)$$

The diffusion term in the ϵ equation has the same benefits as the analogous term in the k equation.

Combining equation (2.36) and equation (2.37), the turbulent viscosity can be written as

$$\nu_T = C_\mu k^2 / \epsilon, \quad (2.40)$$

and therefore $l(x_i)$ is obtained from k and ϵ . Here C_μ is a model constant.

The equation for k and ϵ together with the specification of ν_T , form the $k - \epsilon$ turbulence model. This model is said to be *complete* since it does not require specifications such as the turbulent length scale $l(x_i)$.

The $k - \epsilon$ model consists of four components; two model equations are solved for k and ϵ . The turbulent viscosity is defined by $\nu_T = C_\mu k^2 / \epsilon$. The Reynolds stresses are found from the turbulent-viscosity hypothesis and the Reynolds equations are solved for \bar{u}_i and \bar{p} .

Standard values of the model constants of the $k - \epsilon$ turbulence model used in the model equations are:

$$C_\mu = 0.09, C_{\epsilon 1} = 1.44, C_{\epsilon 2} = 1.92, \sigma_k = 1.0, \sigma_\epsilon = 1.3, \quad (2.41)$$

[3].

2.7 Shear stress near walls

The total shear stress is the sum of the viscous stress and the Reynolds stress. Right at the wall, the no slip boundary condition $u_i(x_i, t) = 0$ implies that all Reynolds stresses are zero. Hence, all the wall shear stress is due to the viscous contribution. This is in contrast to the situation in free shear flows where the viscous stresses are everywhere negligible compared with the Reynolds stresses. Therefore, close to walls the viscosity ν and the wall shear stress τ_w are important parameters. From them, *viscous scales* are defined which are the appropriate velocity and length scales in the near wall region. These are the *friction velocity*

$$u_\tau \equiv \sqrt{\frac{\tau_w}{\rho}} \quad (2.42)$$

and the *viscous lengthscale*

$$\delta_\nu \equiv \nu \sqrt{\frac{\rho}{\tau_w}} = \frac{\nu}{u_\tau}. \quad (2.43)$$

The distance from the wall measured in viscous lengths or *wall units* is defined as

$$y^+ \equiv \frac{x_2}{\delta_\nu} = \frac{u_\tau x_2}{\nu}. \quad (2.44)$$

Different regions in the near-wall flow are defined based on y^+ . In the *viscous wall region* $y^+ < 50$, the viscosity contributes to the shear stress. Outside this region, the effect of viscosity is negligible. In the *viscous sublayer* $y^+ < 5$, the Reynolds shear stress is negligible compared with the viscous stress. [3]

2.8 The log law

We now consider the flow through a rectangular duct of height $h = 2\delta$, width b and length L . The duct is long ($L/\delta \gg 1$) and has a large aspect ratio ($b/\delta \gg 1$). The mean flow is in the x_1 -direction and the velocity statistics depend only on the x_2 coordinate. Then the viscous stress and the turbulence production are both determined by $\frac{d\bar{u}}{dx_2}$. Here \bar{u} is the x_1 -component of the mean velocity. On dimensional grounds we can write

$$\frac{d\bar{u}}{dx_2} = \frac{u_\tau}{x_2} \Phi \left(\frac{x_2}{\delta_\nu}, \frac{x_2}{\delta} \right) \quad (2.45)$$

where Φ is some non-dimensional function. The idea behind the two parameters is that δ_ν is the appropriate length scale in the viscous wall region and δ is the appropriate one in the outer layer. Close to the wall the function Φ should be entirely defined by the viscous scales, independent of δ . Mathematically this implies that the function $\Phi \left(\frac{x_2}{\delta_\nu}, \frac{x_2}{\delta} \right)$ tends asymptotically to a function of $\frac{x_2}{\delta_\nu}$ only, as $\frac{x_2}{\delta}$ tends to zero. Hence, equation (2.45) becomes

$$\frac{d\bar{u}}{dx_2} = \frac{u_\tau}{x_2} \Phi_1 \left(\frac{x_2}{\delta_\nu} \right), \text{ for } \frac{x_2}{\delta} \ll 1, \quad (2.46)$$

where

$$\Phi_1 \left(\frac{x_2}{\delta_\nu} \right) = \lim_{x_2/\delta \rightarrow 0} \Phi \left(\frac{x_2}{\delta_\nu}, \frac{x_2}{\delta} \right) \quad (2.47)$$

Using equation(2.44) and the definition

$$u^+ \equiv \frac{\bar{u}}{u_\tau} \quad (2.48)$$

equation (2.46) becomes

$$\frac{du^+}{dy^+} = \frac{1}{y^+} \Phi_1(y^+). \quad (2.49)$$

The integral of (2.49) is known as the *law of the wall*. As mentioned before, when y^+ is large the viscosity has little effect. Hence, the dependence of $\Phi_1(\frac{x_2}{\delta_\nu})$ on ν disappears and the value of Φ_1 is constant, denoted by κ^{-1} . Using this in equation (2.49) and integrating we end up with

$$u^+ = \frac{1}{\kappa} \ln y^+ + B, \quad (2.50)$$

where B is a constant. This relationship is known as the *log law* and κ is the von Kármán constant. The log law holds for $y^+ > 30$, $\frac{x_2}{\delta} < 0.3$. There are some variations in the values ascribed to the log-law constants but generally they are within 5 % of

$$\kappa = 0.41, \quad B = 5.2, \quad (2.51)$$

[3].

2.9 Turbulence modelling near walls

The profiles of \bar{u} and ϵ are steep near walls. To resolve them a substantial fraction of the computational effort must be devoted to the near-wall region. The idea of the *wall-function* approach is to apply the wall functions boundary conditions (WFBC) some distance away from

the physical wall so that the turbulence-model equations are not solved close to the physical wall. WFBC are applied at a location $x_2 = x_{2p}$ in the log-law region where y^+ is around 50. The subscript "p" indicates quantities evaluated at x_{2p} , e.g. $\bar{u}_p, k_p, \epsilon_p$. From direct numerical simulations (DNS) on wall-bounded type flows described above, it is shown that there is balance between production and dissipation in the region where y^+ is around 50, [3]. It holds that

$$-\rho \overline{u'_1 u'_2} = \tau_w = \rho u_\tau^2 \quad (2.52)$$

where the last step is according to definition (2.42). Since there is a balance between production and dissipation and since we near walls can neglect all velocity gradients except $\frac{\partial \bar{u}}{\partial x_2}$, the dissipation can be written

$$\epsilon = P = -\overline{u'_1 u'_2} \frac{\partial \bar{u}}{\partial x_2} = \frac{u_\tau^3}{\kappa y} \quad (2.53)$$

where the derivative with respect to x_2 is taken from equation (2.50). On the other hand, in the near-walls approximation and according to the turbulent-viscosity hypothesis we can write

$$-\overline{u'_1 u'_2} = u_\tau^2 = \nu_T \frac{\partial \bar{u}}{\partial x_2} = \frac{C_\mu k^2}{\epsilon} \frac{u_\tau}{\kappa x_2}. \quad (2.54)$$

Finally, combining equation (2.53) and (2.54), we end up with the following expression for ϵ near walls:

$$\epsilon_p = \frac{C_\mu^{3/4} k_p^{3/2}}{\kappa x_{2p}}. \quad (2.55)$$

When this type of boundary condition is used for ϵ , Neumann boundary condition is applied to k . The boundary conditions for the pressure and velocity fields are not changed.

The production term, P and y^+ can also be calculated according to wall function theory to be set in the near wall region. If, equation (2.54) and equation (2.53) are combined we can write

$$P_p = \nu_T \frac{C_\mu^{1/4} k_p^{1/2}}{\kappa x_{2p}} \frac{\partial \bar{u}(x_{2p})}{\partial x_2} \quad (2.56)$$

and

$$y_p^+ = \frac{C_\mu^{1/4} k_p^{1/2} x_{2p}}{\nu}. \quad (2.57)$$

3 Implementation of turbulence models in OpenFOAM

OpenFOAM is written in the object oriented programming language C++. In this section we will briefly describe some functions, classes and objects frequently occurring in the pieces of code that will follow in this chapter.

Each term in a partial differential equation (PDE) is represented in OpenFOAM code using the classes of static functions `finiteVolumeMethod` and `finiteVolumeCalculus`, shortened by a `typedef` to `fvm` and `fvc`, respectively. `fvm` and `fvc` contain static functions, representing differential operators that discretise the terms in the PDE.

Equations, and terms of equations are declared as `tmp<Type>` where `<Type>` is either `<fvVectorMatrix>` if the equation is a vector equation, like the momentum equation, or `<fvScalarMatrix>` if the equation is a scalar equation, like the ϵ -equation. The names indicate that the resulting discretized equations are stored as matrices. For more details, see [5].

3.1 The momentum equation

When using the incompressible solver `simpleFoam` in OpenFOAM, the implementation of the momentum equation can be found in the file `UEqn.H`. The equation is implemented as

```
// Solve the Momentum equation

tmp<fvVectorMatrix> UEqn
(
    fvm::div(phi, U)
    + turbulence->divDevReff(U)
);

UEqn().relax();

eqnResidual = solve
(
    UEqn() == -fvc::grad(p)
).initialResidual();

maxResidual = max(eqnResidual, maxResidual);
```

Listing 1: The file `UEqn.H`.

The first term in the implementation of the momentum equation is translated into

$$\frac{\partial}{\partial x_j} (\bar{u}_i \bar{u}_j) = \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} + \bar{u}_i \frac{\partial \bar{u}_j}{\partial x_i} = \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j}. \quad (3.1)$$

The last step is due to the incompressibility of the mean flow field.

The function `divDevReff()`, found in `UEqn.H` is defined in the file `kEpsilon.C` as

```
...
tmp<fvVectorMatrix> kEpsilon :: divDevReff( volVectorField& U) const
{
    return
    (
        - fvm::laplacian( nuEff(), U)
        - fvc::div( nuEff() * dev( fvc::grad(U)().T()))
    );
}
...
```

Listing 2: definition of the function `divDevReff()`.

Translated, the first term of the `divDevReff` is written

$$-\frac{\partial}{\partial x_j} \left(\nu_{eff} \frac{\partial \bar{u}_i}{\partial x_j} \right). \quad (3.2)$$

The second term of the `divDevReff` is the divergence of the effective viscosity times the deviatoric part of the transposed gradient of \bar{u}_i . This term is written

$$-\frac{\partial}{\partial x_j} \left[\nu_{eff} \left(\frac{\partial \bar{u}_j}{\partial x_i} - \frac{1}{3} \frac{\partial \bar{u}_k}{\partial x_k} \delta_{ij} \right) \right], \quad (3.3)$$

which simplifies to

$$-\frac{\partial}{\partial x_j} \left(\nu_{eff} \frac{\partial \bar{u}_j}{\partial x_i} \right), \quad (3.4)$$

due to incompressibility. Then the total `divDevReff` is

$$-\frac{\partial}{\partial x_j} \left[\nu_{eff} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right]. \quad (3.5)$$

The last part of the implementation is the gradient of p . Translated into mathematics, the equation that is implemented is

$$\bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial}{\partial x_j} \left[\nu_{eff} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right] = -\frac{\partial p}{\partial x_i} \quad (3.6)$$

If this result is compared to (2.18) the difference between the implemented equation and the equation according to theory is that there is no term with the time derivative of \bar{u}_i or any term with k , the turbulent kinetic energy. The connection to previous time step in the implemented equation is through the line

```
...
UEqn().relax();
...
```

Listing 3: The connection to previous time step.

in the file `UEqn.H`. The function `relax()` is used to scale the solution from the previous time iteration before it is employed in the current iteration. The theory behind relaxation of the solution is presented in section 4.3.

The k term is incorporated in the pressure term as $\tilde{p} = \bar{p} + \frac{2}{3}k$ and the $\frac{1}{\rho}$ factor in front of the pressure term in the RANS equations is dropped in OpenFOAM. So if the true mean pressure field is sought for, one has to take this in consideration.

3.2 The k -equation

The implementation of the incompressible k -equation in the file kEpsilon.C is

```
...
// Turbulent kinetic energy equation
tmp<fvScalarMatrix> kEqn
(
    fvm::ddt(k_)
    + fvm::div(phi_, k_)
    - fvm::Sp(fvc::div(phi_), k_)
    - fvm::laplacian(DkEff(), k_)

    ==
    G
    - fvm::Sp(epsilon_/k_, k_)
);

kEqn().relax();
solve(kEqn);
bound(k_, k0_);
...
...
```

Listing 4: Implementation of the k -equation.

The terms G and $DkEff()$ are defined in the same file.

```
...
volScalarField G("RASModel::G", nut_*2*magSqr(symm(fvc::grad(U_))));
...
DkEff() = nut_ + nu()
...
```

Listing 5: Definition of the production term and the function Dkeff().

The first term on the left hand side in the implementation of the k -equation is the time derivative of k . The second term is the divergence of the velocity times k

$$\frac{\partial(\bar{u}_j k)}{\partial x_j} = k \frac{\partial \bar{u}_j}{\partial x_j} + \bar{u}_j \frac{\partial k}{\partial x_j}. \quad (3.7)$$

The third term is the source term and is written $-k \frac{\partial \bar{u}_j}{\partial x_j}$. The last term of the left hand side is written $-\frac{\partial}{\partial x_j} \left(\nu_{eff} \frac{\partial k}{\partial x_j} \right)$. The G term is the production term and the OpenFOAM implementation of it is a bit difficult to understand. The symmetric part of the gradient of \bar{u}_i is

$$\frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right), \quad (3.8)$$

and the translation of the OpenFOAM code of G is therefore

$$\begin{aligned}
G &= 2\nu_T \left| \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right|^2 \\
&= \frac{\nu_T}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \frac{\partial \bar{u}_j}{\partial x_i} + 2 \frac{\partial \bar{u}_i}{\partial x_j} \frac{\partial \bar{u}_j}{\partial x_i} \right) \\
&= \nu_T \frac{\partial \bar{u}_i}{\partial x_j} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right).
\end{aligned} \tag{3.9}$$

The last term in the implementation of the k -equation is just ϵ . Putting all the pieces together we end up with

$$\frac{\partial k}{\partial t} + k \frac{\partial \bar{u}_j}{\partial x_j} + \bar{u}_j \frac{\partial k}{\partial x_j} - k \frac{\partial \bar{u}_j}{\partial x_j} - \frac{\partial}{\partial x_j} \left(\nu_{eff} \frac{\partial k}{\partial x_j} \right) = \nu_T \frac{\partial \bar{u}_i}{\partial x_j} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \epsilon \tag{3.10}$$

which simplifies to

$$\frac{\partial k}{\partial t} + \bar{u}_j \frac{\partial k}{\partial x_j} - \frac{\partial}{\partial x_j} \left(\nu_{eff} \frac{\partial k}{\partial x_j} \right) = \nu_T \frac{\partial \bar{u}_i}{\partial x_j} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \epsilon. \tag{3.11}$$

Here

$$\nu_{eff} = \nu + \nu_T. \tag{3.12}$$

The only difference between this equation and equation (2.35) is that ν_T in the last term on the left side is not divided by σ_k . But since $\sigma_k = 1$ in the $k - \epsilon$ model the equations are the same.

3.3 The ϵ -equation

The implementation of the incompressible ϵ -equation in the file kEpsilon.C is

```

...
// Dissipation equation
tmp<fvScalarMatrix> epsEqn
(
    fvm::ddt(epsilon_)
    + fvm::div(phi_, epsilon_)
    - fvm::Sp(fvc::div(phi_), epsilon_)
    - fvm::laplacian(DepsilonEff(), epsilon_)
    ==
    C1_*G*epsilon_/k_
    - fvm::Sp(C2_*epsilon_/k_, epsilon_)
);
epsEqn().relax();
epsEqn().boundaryManipulate(epsilon_.boundaryField());
solve(epsEqn);
bound(epsilon_, epsilon0_);
...

```

Listing 6: Implementation of the dissipation equation.

with DepsilonEff() defined in the same file as

```

...
DepsilonEff() = nut_/SigmaEps + nu()
...
```

and the production term G defined in the same way as in the k -equation. The terms are very similar to the ones in the k -equation with ϵ instead of k . The translation of the OpenFOAM code of the ϵ -equation is

$$\frac{\partial \epsilon}{\partial t} + \bar{u}_j \frac{\partial \epsilon}{\partial x_j} - \frac{\partial}{\partial x_j} \left((\nu + \frac{\nu_T}{\sigma_\epsilon}) \frac{\partial \epsilon}{\partial x_j} \right) = C_1 \frac{\epsilon}{k} \nu_T \frac{\partial \bar{u}_i}{\partial x_j} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - C_2 \frac{\epsilon^2}{k}. \quad (3.13)$$

The difference between the implemented ϵ equation and equation (2.39) is that the viscosity is added to $\frac{\nu_T}{\sigma_\epsilon}$ in the last term on the left hand side in the above equation.

3.4 The turbulent viscosity ν_T

To have a complete turbulence model the turbulent viscosity ν_T must be defined. This is done in kEpsilon.C.

```

...
nut_ = Cmu_*sqrt(k_)/(epsilon_ + epsilonSmall_);
...
```

This implementation is the same as equation (2.40) except for the term epsilonSmall_ which is added to avoid division by zero.

3.5 Implementation of wall models

When wall functions boundary conditions are used, ϵ is calculated in a special way in the near wall region, according to equation (2.55). This is implemented in OpenFOAM in the file epsilonWallFunctionFvPatchScalarField.C in the following way

```

...
// Set epsilon and G
    forAll(nutw, faceI)
    {
        label faceCellI = patch().faceCells()[faceI];

        scalar yPlus = Cmu25*y[faceI]*sqrt(k[faceCellI])
                      /nuw[faceI];

        epsilon[faceCellI] = Cmu75*pow(k[faceCellI], 1.5)
                           /(kappa_*y[faceI]);

        if (yPlus > yPlusLam)
        {
            G[faceCellI] =
                (nutw[faceI] + nuw[faceI])
                *magGradUw[faceI]
                *Cmu25*sqrt(k[faceCellI])
                /(kappa_*y[faceI]);
        }
        else
        {
            G[faceCellI] = 0.0;
        }
    }
...

```

As we can see, the production term G is also updated according to wall function theory, but only if $yPlus$ is bigger than $yPlusLam$. $MagGradUw$, nuw , $nutw$, $Cmu25$ and $Cmu75$ are all defined in epsilonWallFunctionFvPatchScalarField.C. Here $faceI$ is the index for the face next to the boundary and $faceCellI$ is the index of the cell that have one face with index $faceI$. $Cmu25$ and $Cmu75$ have values $C_\mu^{1/4}$ and $C_\mu^{3/4}$, respectively. The value of C_μ is set to be 0.09 according to $k - \epsilon$ theory but can be changed. If the above piece of code is translated into mathematics we find that ϵ and the production term are implemented as

$$\epsilon = \frac{C_\mu^{3/4} k_p^{3/2}}{\kappa y_p} \quad (3.14)$$

and

$$G = \frac{(\nu_{Tw} + \nu_w) C_\mu^{1/4} k^{1/2}}{\kappa y}, \quad (3.15)$$

respectively. The difference compared to theory is that in OpenFOAM the effective viscosity is used when calculating the production term, rather than only the turbulent viscosity, according to equation (2.56). the subscript "w" indicates that the viscosity and turbulent viscosity are calculated at the wall.

4 Numerical solution method

4.1 The finite volume method (FVM)

The solution domain is subdivided into a finite number of small control volumes (polyhedra) and the conservation equations are applied to each control volume. FVM uses the integral form of a general convection-diffusion equation for a quantity as its starting point which is described as

$$\frac{d}{dt} \int_V \phi dV + \int_S \phi \mathbf{u} \cdot \mathbf{n} dS = \int_S \Gamma(\nabla \phi) \cdot \mathbf{n} dS + \int_V q dV \quad (4.1)$$

where ϕ is some flow variable, k for example, V is a control volume, S its bounding surface, \mathbf{n} unit normal, Γ diffusion coefficient and q some external term.

The centroid of each control volume is assigned to be the computational node at which the variable values are to be calculated. Interpolation is used to express variable values at the control volume surface in terms of the centroid nodal values. Surface and volume integrals are approximated using suitable discretization methods. As a result, one obtains an algebraic equation for each control volume, in which a number of neighbour nodal values appear. One advantage with FVM is that it can handle complex geometries, however one disadvantage compared to other computational methods is that methods of higher order than 2nd, are more difficult to develop in 3D. A more detailed description of the Finite Volume Method can be found in chapter 4 of [4].

4.2 The SIMPLE algorithm

When the RANS approach for turbulence is used, a stationary problem arises. RANS and many other methods for steady problems in computational fluid dynamics can be regarded as unsteady problems until a steady state is reached. If an implicit method is used in time, the discretized momentum equations at the new time step are non-linear. Due to this and that the underlying differential equations are coupled, the equations system resulting from discretization cannot be solved directly. Iterative solution methods are the only choice.

The momentum equations are usually solved sequentially for each component. The pressure used in each iteration is obtained from the previous time step and therefore the computed velocities normally do not satisfy the discrete continuity equation. In order for the velocities to fulfill this equation one must modify the pressure field. This can be done by solving a discrete Poisson equation for the pressure.

After solving this new equation for the pressure the final velocity field at the new iteration is calculated. This new velocity field satisfies the continuity equation, but the velocity and pressure fields do not satisfy the momentum equations. Therefore, the procedure described above is iterated until a velocity field is obtained that satisfies both the momentum and continuity equations. These iterations, made to obtain a solution which satisfies all of the equations, are called *outer* iterations.

Methods of this kind which first construct velocity fields that do not satisfy the continuity equation and then correct them are known as *projection methods*. The SIMPLE algorithm is such a method, and it is the solving procedure used in OpenFOAM for our computations. A more detailed description of the SIMPLE algorithm can be found in chapter 7 of [4].

4.3 Under-relaxation

It is often necessary to limit the change in each variable from one outer iteration to the next, because a change in one variable changes the coefficients in the other equations, which may slow or prevent convergence. This type of limiting of each variable is called under-relaxation. One under-relaxation technique that is widely used is presented below.

On the n :th outer iteration, the algebraic equation for a generic variable ϕ , at point P , can be written

$$A_P \phi_P^n + \sum_l A_l \phi_l^n = Q_P, \quad (4.2)$$

where Q contains all the terms that do not depend explicitly on ϕ^n . Q and the coefficients A_l may involve ϕ^{n-1} . This equation is linear and the system of equations in the whole solution domain is assumed to be solved iteratively (inner iterations). If we allow ϕ to change by as much as equation (4.2) requires in the early outer iterations, it could cause instability. Therefore, we allow ϕ_n to change according to

$$\phi^n = \phi^{n-1} + \alpha_\phi (\phi^{new} - \phi^{n-1}) \quad (4.3)$$

where ϕ^{new} is the result of equation (4.2) and the under-relaxation factor satisfies $0 < \alpha_\phi < 1$. Replacing ϕ^{new} in equation (4.3) by

$$\phi_P^{new} = \frac{Q_P - \sum_l A_l \phi_l^n}{A_P}, \quad (4.4)$$

leads to a modified equation at node P

$$A_P^* \phi_P^n + \sum_l A_l \phi_l^n = Q_P^*, \quad (4.5)$$

where A_P^* and Q_P^* are modified main diagonal matrix elements and source vector components given by

$$A_P^* = \frac{A_P}{\alpha_\phi} \quad (4.6)$$

and

$$Q_P^* = Q_P + \frac{1 - \alpha_\phi}{\alpha_\phi} A_P \phi_P^{n-1}. \quad (4.7)$$

This kind of under-relaxation has a positive effect on many iterative solution methods since the diagonal dominance of the matrix A is increased. Optimum under-relaxation is problem dependent and the selection of under-relaxation factors is therefore largely empirical.

In OpenFOAM, under-relaxation factors are set for all flow variables and can also be changed during run-time.

5 Different types of turbulence models used in our simulations

In this work, we use three different types of turbulence models. They are presented below.

5.1 Linear eddy viscosity models

Except the $k - \epsilon$ model presented in section 2.6, four other models that use the linear eddy viscosity hypothesis are used.

The $k - \omega$ model use the turbulence frequency of the large eddies ω , to model the turbulence. A variation to the standard $k - \omega$ model is the $k - \omega$ SST (shear stress transport) model. This model can handle turbulence that is not close to local equilibrium. This occurs when the production of turbulent energy departs significantly from the dissipation rate ϵ .

The RNG $k - \epsilon$ model includes a modification to the transport equation for ϵ stemming from renormalization group theory.

The normal turbulent stresses are positive by definition. In the standard $k - \epsilon$ model, they can become negative due to the definition of the turbulent viscosity. The realizable $k - \epsilon$ model use certain mathematical constrains on the normal stresses to avoid this.

5.2 Reynolds stress transport models

These types of models determine the turbulent stresses directly by solving a transport equation for each stress component. This requires the solution of six additional coupled equations, together with an equation for ϵ . The LRR (Launder-Reece-Rodi) model used in this work is of this type.

5.3 Non-linear eddy viscosity models

These models relate the turbulent stresses to the time averaged velocity gradients algebraically with higher order quadratic and cubic terms. This feature additional coefficients which can then be calibrated to experimental data. Models of this type used in this work are the non linear $k - \epsilon$ model and the Lien cubic $k - \epsilon$ model, [6].

6 The Pitz-Daily case

The Pitz-Daily case is a tutorial case in OpenFOAM. The background to this case is an experiment that was carried out by Robert W. Pitz and John W. Daily. The aim of their experiment was to study combustion of a fluid mixture of air and propane in a turbulent mixing layer. [7] The main objective with the choice to study this case was because the case geometry is two-dimensional, rather simple and it is obvious where the flow separation will occur. The objectives with this study was to evaluate the different turbulence models with respect to the prediction of the center of the developed recirculation area, the reattachment point and the amount of turbulent kinetic energy produced in the turbulent area. The models were mainly tested on two different types of meshes: The mesh that comes with the tutorial and one mesh constructed to be used without WFBC. The meshes are described and discussed in section 6.1. The geometry of the case is a backward facing step and it is the sharp edge of the step that gives rise to the mixing layer and the recirculation zone. The outflow region is contracted but this is not of great importance since the essential phenomenon of the flow occur earlier. The geometry can be seen in figure 1.

A cartesian coordinate system is used for the simulations, and in the domain coordinates (x_1, x_2) , the velocity field is $\mathbf{u} = (u_1, u_2)$. The x_1 -axis is parallel to the floor of the domain, pointing downstream. The x_2 -axis is normal to the floor, pointing upward and the origin is fixed in the upper right corner of the step. The top and bottom patch of the domain is set to have no slip boundary conditions, i.e. $\mathbf{u} = 0$ at these boundaries. The inlet velocity is $V_\infty = 10$ m/s, pointing in the positive x_1 -direction.

The Reynolds number of this flow, based on the inlet velocity $V_\infty = 10$ m/s, the height of the step, $H = 0.0254$ m and the kinematic viscosity of air, $\nu = 10^{-5}$ m²/s is $Re = 25400$.



Figure 1: The Pitz-Daily case geometry.

6.1 Different meshes

As for computational mesh, mainly two types were used. The mesh that comes with the tutorial case, with a total of 12225 cells, was used when WFBC were applied. In OpenFOAM, the two dimensional meshes are represented as three dimensional. The primary refinement of this mesh is around the centre line to better resolve the shear layer there. From the simulation using the standard $k - \epsilon$ model with WFBC, y^+ varies from 0.7 up to 26. This mesh will in the following be referred to as M_{t1} since it comes with the tutorial.

The second mesh used for the Pitz-Daily simulations was designed to be used without WFBC for the turbulent quantities. To resolve the steep profiles of k and ϵ near walls this mesh had to be more refined near the upper and lower wall of the domain compared to the M_{t1} mesh. In the simulation using the standard $k - \epsilon$ model, the y^+ value for this mesh is below 1, except at the wall patches very close to the inlet. This mesh will in the following be referred to as M_r since it is refined near physical walls.

In addition to these, two more meshes were generated and used when trying to demonstrate mesh convergence for the Pitz-Daily case. These two meshes have the same distribution of cells as the M_{t1} mesh. The first has twice as many cells as the M_{t1} mesh in each direction. This mesh will in the following be referred to as M_{t2} . The second has twice as many cells as the M_{t2} mesh in each direction. This mesh will in the following be referred to as M_{t3} . For an overview of the mesh characteristics, see table 1.

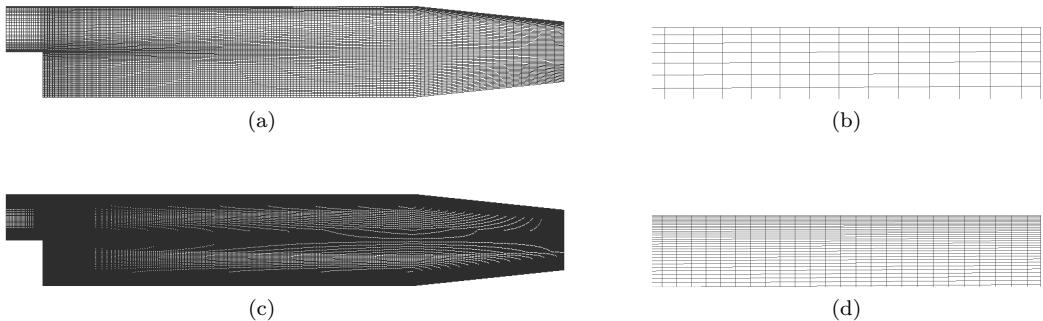


Figure 2: (a) The M_{t1} mesh. (b) Zoomed in at the top boundary of the M_{t1} mesh. Notice that there is almost no refinement of the mesh in the x_2 direction near the boundary. (c) The M_r mesh. (d) Zoomed in at the top boundary of the M_r mesh. Notice the refinement of the mesh in the x_2 direction near the boundary.

Types of meshes for the Pitz-Daily case		
Mesh	Number of cells	Range of y^+
M_{t1}	12225	$0.7 < y^+ < 26$
M_r	75600	$0.7 < y^+ < 33$
M_{t2}	48900	$0.2 < y^+ < 12$
M_{t3}	195600	$0.1 < y^+ < 5$

Table 1: Meshes used for the Pitz-Daily case.

6.2 Boundary conditions for turbulent quantities

6.2.1 The k -equation

All of the models used except the LRR model use the k -equation to describe the turbulence. The values of k are set to a constant value on the inlet patch. k is computed according to equation (2.15), where the inlet turbulence is assumed to be isotropic and the fluctuations are estimated to be 5 % of the inlet velocity, $V_\infty = 10$ m/s. Therefore, the value of k on the inlet is $0.375 \text{ m}^2/\text{s}^2$.

For the top and bottom patch, different boundary conditions were used depending upon which mesh was used. When using WFBC, the M_{t1} mesh was used. For k this is set to be "kqRWallFunction" in OpenFOAM and simply acts as a Neumann boundary condition. For simulations using the M_r mesh the value of k is set to the constant value of $0.375 \text{ m}^2/\text{s}^2$.

According to theory, $k \rightarrow 0$ at the wall. But because of the ϵ/k term in the ϵ -equation and since ϵ is finite at the wall, $k = 0$ could not be used. This could have been avoided by choosing another turbulence model better suited for this type of boundary condition on k . But since the computational work was done in parallel with the theoretical studies, all of the theory was not taken into consideration when setting up the simulations. This type of boundary condition is not explained in the OpenFOAM documentation either.

For the outlet patch, Neumann conditions were used for both meshes.

6.2.2 The ϵ -equation

All of the models except the $k - \omega$ and $k - \omega$ -SST use the ϵ -equation to describe the turbulence. On the inlet patch, the value of ϵ is set to be $14.855 \text{ m}^2/\text{s}^3$, computed according to

$$\epsilon = \frac{C_\mu^{3/4} k^{3/2}}{l}, \quad (6.1)$$

where l is the turbulent length scale estimated to be 10 % of the inlet width and C_μ is specified according to equation (2.41). [5]

For the top and bottom patch, different boundary conditions were used depending upon which mesh was used. For the simulations using WFBC, the M_{t1} mesh was used. For ϵ this is set to be "epsilonWallFunction" and is calculated according to equation (2.55). For the simulations using the M_r mesh, ϵ is set to the constant value of $14.855 \text{ m}^2/\text{s}^3$. For the outlet patch, Neumann boundary conditions were used for both meshes.

6.2.3 The equation for the Reynolds stress tensor

For the LRR model, boundary conditions for the Reynolds stress tensor have to be set up. As described in section 2.1, the Reynolds stress tensor is symmetric and consists of six independent

elements. Therefore, conditions for these six elements have to be set up. The inlet condition for the Reynolds stress tensor is computed from the boundary conditions for k and u_i . The boundary conditions on the top and bottom patch is set to be "kqRWallFunction" and for the outlet Neumann boundary conditions were used.

6.2.4 The ω -equation

The $k - \omega$ and $k - \omega$ SST models use ω to describe the turbulence. The boundary values for ω are calculated according to $\omega = \epsilon/k$ from the values of k and ϵ presented in section 6.2.1 and 6.2.2. When using the mesh M_{t1} , "omegaWallFunction" is used for the top and bottom patch. It acts as the "epsilonWallFunction" but for ω instead of ϵ .

6.3 Mesh convergence for the Pitz-Daily case

A numerical method is said to be convergent if the solution of the discretized equations tends to the exact solution of the differential equation as the grid spacing of the mesh tends to zero. For non-linear problems like the Navier-Stokes equations, which are strongly influenced by boundary conditions, the stability and convergence of a method are difficult to prove analytically. Therefore convergence is usually checked using numerical experiments, i.e. repeating the calculation on a series of successively refined grids. For sufficiently small cell sizes of the mesh we usually find that the solution does converge to a grid-independent solution. The rate of convergence is governed by the order of the principal truncation error component. [4] The terms in the momentum equation of the Pitz-Daily case simulations are all discretized with second order methods except the convection term, which is discretized with the first order Upwind scheme. Therefore, we assume that the error in the solution will halve as the mesh spacing is halved. The solution variable on the coarsest mesh is denoted U_h and the solution with halved mesh spacing (in both x_1 and x_2 direction), is denoted $U_{h/2}$ (and so on for further halving of the mesh spacing.) In our calculations, U is the u_1 component of the velocity. Then the order of accuracy of the numerical method is given by the formula

$$2^q \approx \frac{|U_h - U_{h/2}|}{|U_{h/2} - U_{h/4}|}, \quad (6.2)$$

where q is the order of accuracy of the numerical scheme. As seen from equation (6.2), at least three solutions are needed on successive refined meshes.

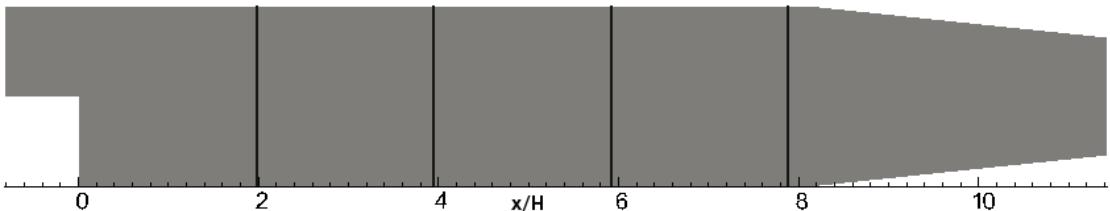


Figure 3: Lines through the Pitz-Daily domain where data are taken.

In order to investigate if there is a grid independent solution for the Pitz-Daily case we simulated the $k-\epsilon$ turbulence model without using WFBC for the turbulent variables. This is because WFBC can destroy the accuracy since it is dependent of the y^+ value. The simulations were carried out on three meshes and these are described in section 6.1. The accuracy q was evaluated according to equation (6.2) along four different lines through the domain. The lines

are shown in figure 3 and q for the different lines are shown in figure 4. As seen in this figure, q

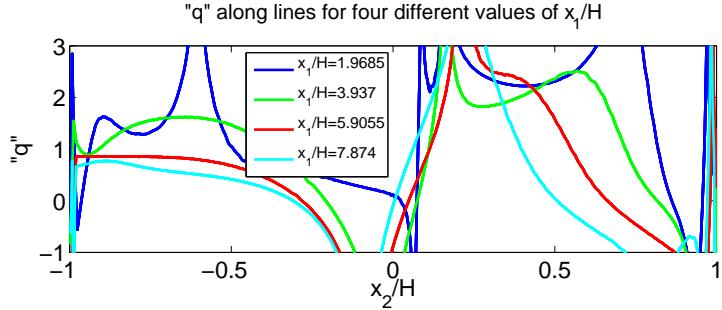


Figure 4: Order of accuracy "q" for different values of x_1/H .

varies a lot and are not around one as it should be according to theory. The peaks appear where the velocity profiles cross, and the difference tends to zero. The velocity profiles are shown in figure 5. For values of x_2/H below zero, q is quite close to one, for some values of x_1/H .

We also tried to estimate the accuracy by looking at how the x_1/H coordinate of the center of the recirculation zone changed when using finer meshes. The result for q we got when looking at this single point was, $q \approx 1.47$.

One way to get better results could be to compute q from simulations on further refined meshes.

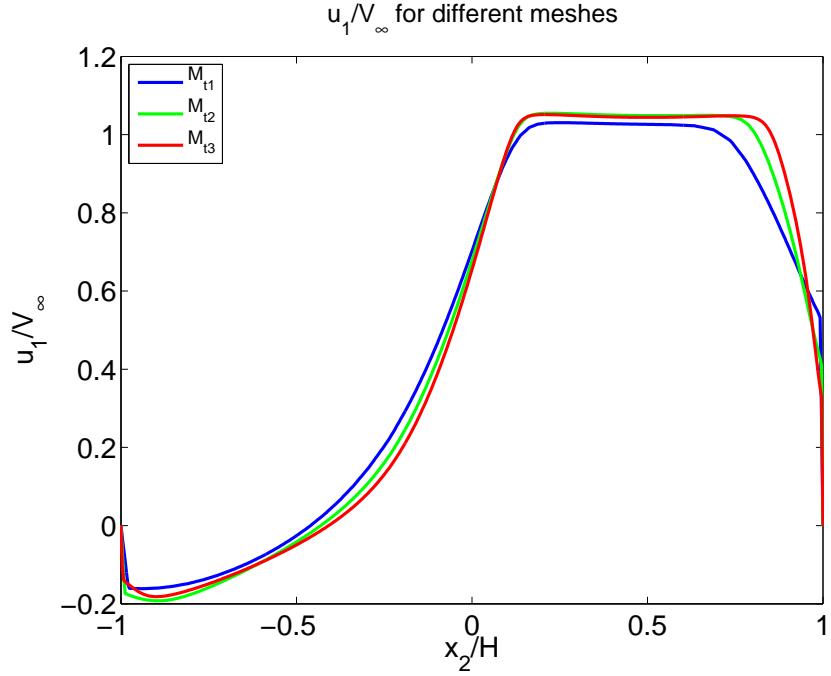


Figure 5: u_1/V_∞ at $x_1/H = 1.9685$ for three different meshes.

6.4 Results from the Pitz-Daily case simulations

The models that converge to a stationary solution are shown in table 2. As convergence criterion, the residuals had to be less than 10^{-5} for all variables except the pressure where the residuals had to be less than 10^{-6} to consider having a converged solution.

Models tested on the M_{t1} mesh	
Description	Comments
Standard $k - \epsilon$ model	Convergence reached after approximately 3 000 iterations.
$k - \omega$ SST two-equation model	Convergence reached after approximately 2 000 iterations.
LRR (Launder-Reece-Rodi) Reynolds stress transport model	Convergence reached after approximately 4 500 iterations.
Renormalization group $k - \epsilon$ model	Convergence reached after approximately 5 000 iterations.
$k - \omega$ two-equation model	Convergence reached after approximately 2 000 iterations.
Lien cubic $k - \epsilon$ model	Convergence reached after approximately 27 000 iterations. Had to fine tune relaxation factors to get convergence.
Non-linear Shih $k - \epsilon$ model	convergence reached after approximately 11 500 iterations.
Realizable $k - \epsilon$ model	convergence reached after approximately 8 500 iterations.

Table 2: Models tested on the M_{t1} mesh. All models were used with WFBC for the turbulent variables.

6.5 Results from simulations on the M_{t1} mesh with WFBC

All the models that converged manage to produce the recirculation zone that is expected due to the sharp edge of the backward facing step. The x_2/H coordinate for the center of the recirculation zone is also more or less equal for the different models. However, this value is strongly dependent of the height H of the step. The x_1/H coordinate for the center of the eddy vary between models and there are roughly three different regions where the center occurs. Around $x_1/H = 2.8$ for the $k - \epsilon$, $k - \omega$, $k - \omega$ SST and the LRR models, around $x_1/H = 3.6$ for the realizable $k - \epsilon$ and RNG $k - \epsilon$ models and around $x_1/H = 6.8$ for the Nonlinear $k - \epsilon$ Shih and Lien cubic $k - \epsilon$ models. The maximum turbulent kinetic energy k also varies between models. The Nonlinear $k - \epsilon$ Shih and the Lien cubic $k - \epsilon$ produce maximum turbulent kinetic energy that is about half as big as all the other models. We collected numerical values from the center of the recirculation region, reattachment point and maximum turbulent kinetic energy in table 4. The recirculation region that appears together with the separation, re-attachment point and the turbulent kinetic energy produced by all the models are shown in figure 6.

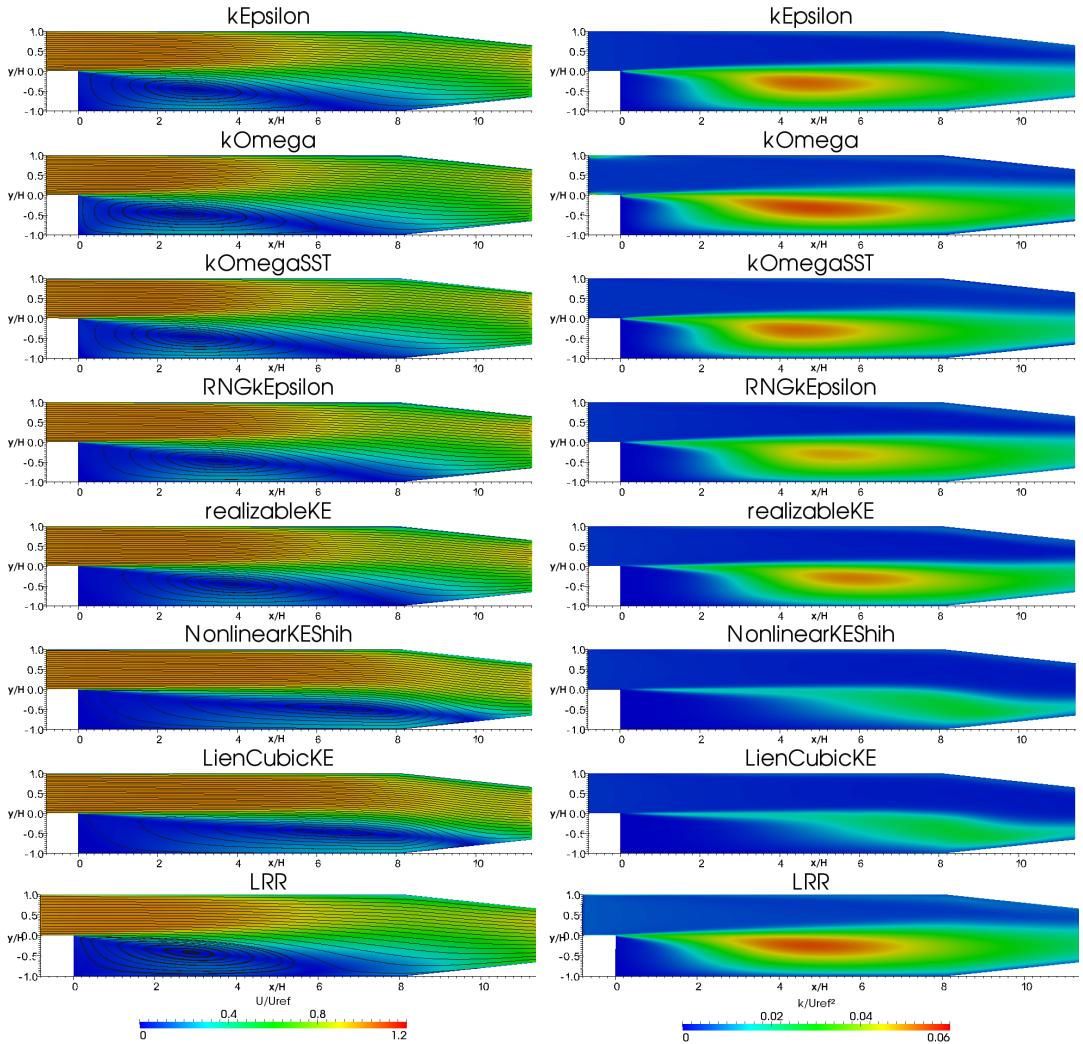


Figure 6: Pitz-Daily results from the models tested on the M_{t1} mesh. The left figures show streamlines and $|\mathbf{u}|/V_\infty$. The right figures show k/V_∞^2 .

6.6 Results from simulations on the M_r mesh, without WFBC

Both the models that converged manage to produce the recirculation zone that is expected due to the sharp edge of the backward facing step. The x_2/H coordinate for the center of the recirculation zone differ more between the models on this mesh than on the M_{t1} mesh. The x_1/H coordinate for the center of the recirculation zone is in the same region for the two models. The center of the recirculation region, reattachment point and maximum turbulent kinetic energy can be found in table 4.

Models tested on the M_r mesh	
Description	Comments
Standard $k - \epsilon$ model	Convergence reached after approximately 3 000 iterations.
Realizable $k - \epsilon$ model	convergence reached after approximately 8 500 iterations.

Table 3: Models tested on the M_r mesh. Both models were used with fixed value boundary conditions for the turbulent variables.

6.6.1 The $k - \epsilon$ model

The result from this model when simulating on the M_r mesh is a bit different compared with the results on the M_{t1} mesh. The recirculation region is smaller in the x_1 direction and the center of the eddy has moved backwards together with the re-attachment point. Now the flow is re-attached around $x_1/H = 5.90$ and the center of the eddy is located at $(x_1/H, x_2/H) = (2.61, -0.45)$. The separation, re-attachment point and the eddy that appear can be seen in figure 7 (a). This model also produces more turbulent kinetic energy near the walls. The turbulent kinetic energy is shown in figure 7 (b). A detailed picture of the "peaks" of k can be seen in figure 10.

6.6.2 The realizable $k - \epsilon$ model

As for the $k - \epsilon$ model, the result on the M_r mesh is different compared with the result on the M_{t1} mesh. The recirculation region is smaller in the x_1 direction and the center of the eddy has moved backwards together with the re-attachment point. Now the flow is re-attached around $x_1/H = 6.22$ and the center of the eddy is located at $(x_1/H, x_2/H) = (2.49, -0.37)$. The separation, re-attachment point and the eddy that appear can be seen in figure 7 (c). This model also produces more turbulent kinetic energy near the walls. The turbulent kinetic energy is shown in figure 7 (d). A detailed picture of the "peaks" of k can be seen in figure 10.

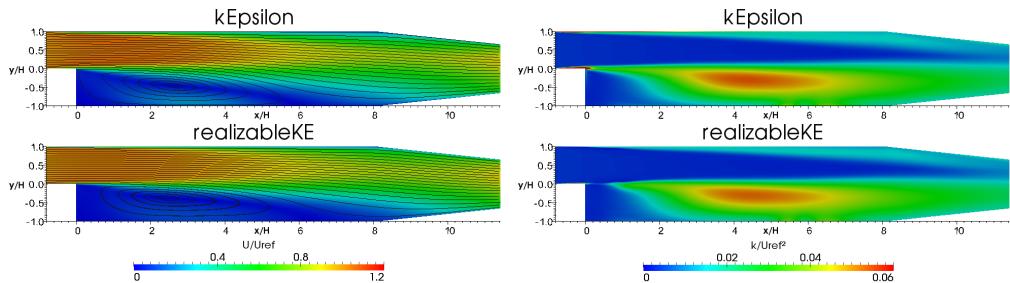


Figure 7: Pitz-Daily results from the tested models on the M_r mesh. The left figures show streamlines and $|\mathbf{u}|/V_\infty$. The right figures show k/V_∞^2 .

6.7 Solution of the realizable $k - \epsilon$ model on the M_{t3} mesh

To have data to compare the solutions from the M_{t1} mesh and the M_r mesh with, we ran the realizable $k - \epsilon$ model on the M_{t3} mesh, with WFBC. Compared with the solution of this model

on the M_{t1} mesh, the center of the recirculation region is moved back to occur at $(x_1/H, x_2/H) = (3.29, -0.44)$. The re-attachment point is the same as on the M_{t1} mesh. The separation, re-attachment point, turbulent kinetic energy and the recirculation region can be seen in figure 8.

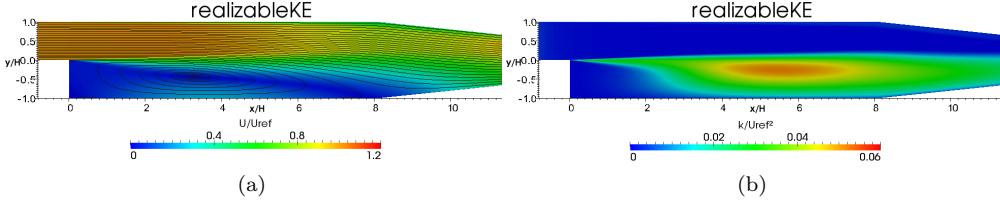


Figure 8: Results from realizable $k-\epsilon$ on the M_{t3} mesh. (a) Streamlines of the velocity. Colour represents $\frac{|\mathbf{u}|}{V_\infty}$. (b) The turbulent kinetic energy k , normalized by V_∞^2 .

Turbulence model	center of eddy	re-attachment point	$\max \frac{k}{V_\infty^2}$
$k - \epsilon$, M_{t1} mesh	2.88	-0.46	6.75
$k - \omega$, M_{t1} mesh	2.68	-0.46	6.82
$k - \omega$ SST, M_{t1} mesh	2.85	-0.46	7.12
RNG $k - \epsilon$, M_{t1} mesh	3.49	-0.46	7.66
realizable $k - \epsilon$, M_{t1} mesh	3.86	-0.45	7.82
Lien cubic $k - \epsilon$, M_{t1} mesh	6.75	-0.48	9.84
Nonlinear $k - \epsilon$ Shih, M_{t1} mesh	6.82	-0.48	9.85
LRR, M_{t1} mesh	2.91	-0.42	5.55
$k - \epsilon$, M_r mesh	2.61	-0.45	5.90
realizable $k - \epsilon$, M_r mesh	2.49	-0.37	6.22
realizable $k - \epsilon$, M_{t3} mesh	3.29	-0.44	7.82

Table 4: Center of recirculation zone, re-attachment points and maximum turbulent kinetic energy for the Pitz-Daily case geometry.

6.8 Discussion and conclusions of the Pitz-Daily results

The objective with the Pitz-Daily case simulations was to compare the results from several RANS models implemented in OpenFOAM. Most of the simulations were made on the mesh that comes with the tutorial. On this mesh, all simulations were made together with WFBC. In addition to these, Two models were run on another mesh, more refined close to the walls compared with the tutorial mesh. This was done to better resolve the turbulent quantities near walls, without using WFBC.

When comparing the results from the tutorial mesh using WFBC, the eight tested models are sorted into three groups with respect to the location of the recirculation zone. The three groups show significantly different prediction of the location of the center of recirculation zone in the streamwise direction.

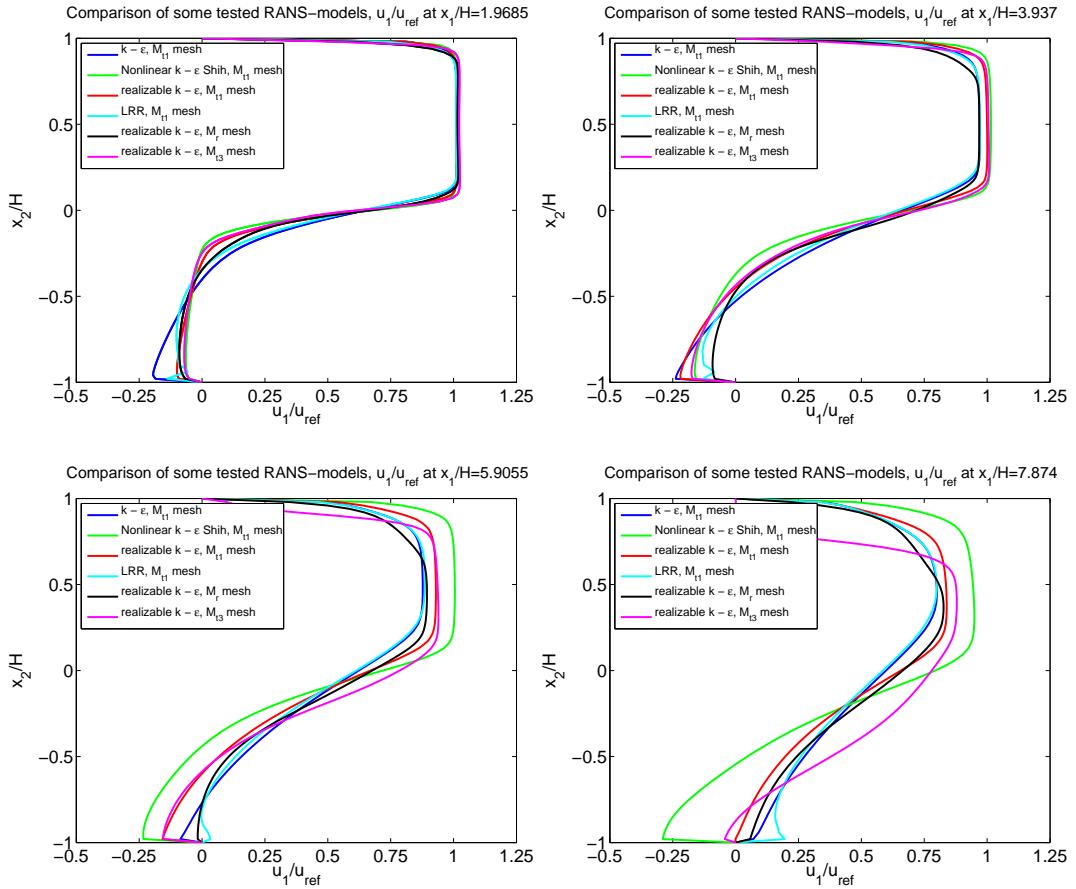


Figure 9: Comparison of u_1/V_∞ from different RANS models. Data are taken along four lines which can be seen in figure 3.

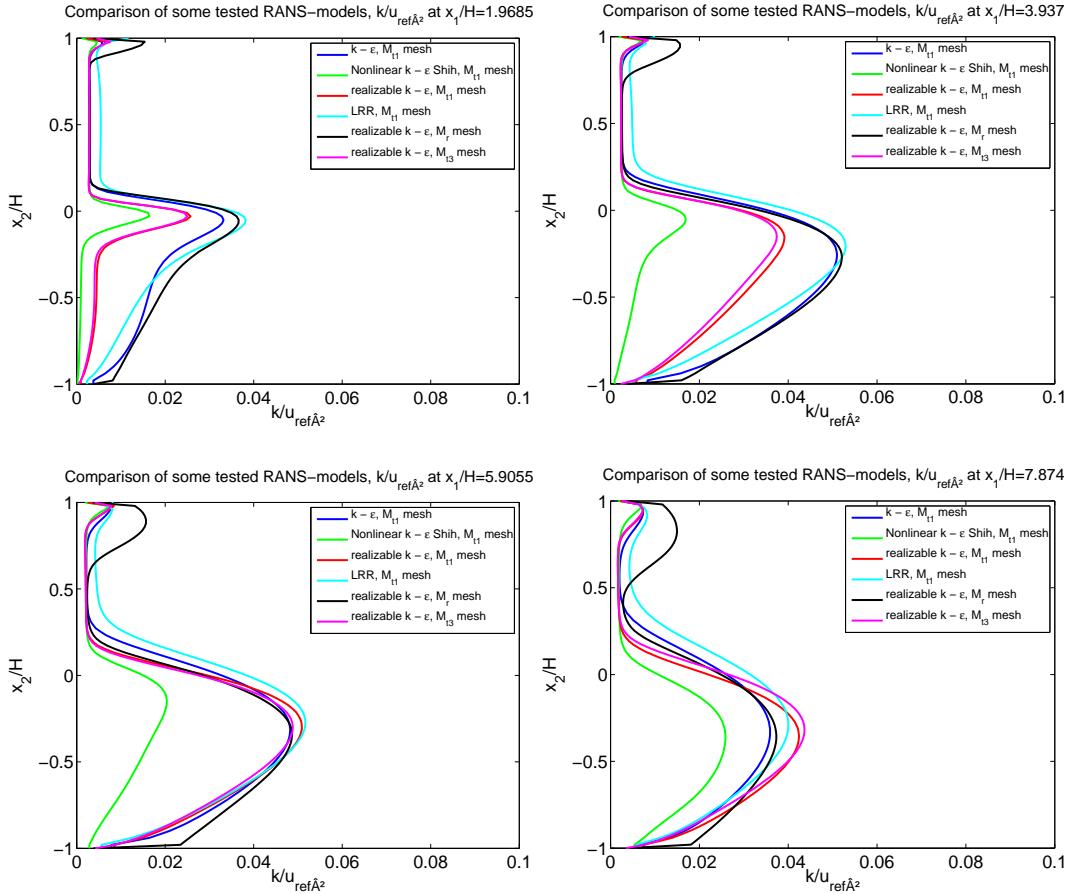


Figure 10: Comparison of k/V_∞^2 from different RANS models. Data is for four different values of x_1/H . Data are taken along four lines which can be seen in figure 3.

Better agreement between the results for the prediction of the recirculation zone from the models might be achieved if the turbulent shear layer around $x_2/H = 0$ is better resolved.

When looking at the produced turbulent kinetic energy, the models are sorted in to two groups, with the non-linear models producing about half as much turbulent kinetic energy as the other models.

The results from the M_r mesh could probably be improved by further refining the mesh near the lower wall just before the backward facing step. Here, y^+ is not below 1. This could also explain the maximum turbulent kinetic energy produced in this region, which can not be seen in the results from the M_{t1} mesh.

These results could probably also be improved further by using turbulence models better suited for this type of constant value boundary conditions of the turbulent variables.

We also tried to demonstrate mesh convergence for this case, but the results were not convincing. One reason for this might be that the most refined mesh was still too coarse.

The best performing models on this geometry are the linear eddy viscosity models with WFBC together with the M_{t1} mesh.

7 The bump case

The objective with the choice to study this case was because we wanted a geometry that was more computationally challenging than the Pitz-Daily case. Though the simple shape of the geometry, the flow in the recirculation region is complex and the flow prediction is challenging. Another aspect when choosing this case was because there are numerous results to compare with, both from LES (Large Eddy Simulations) and LDV (Laser Doppler Velocimeter) experiment. The geometry is a three-dimensional axisymmetric hill placed on the floor of a channel. The domain is shown in figure 11.

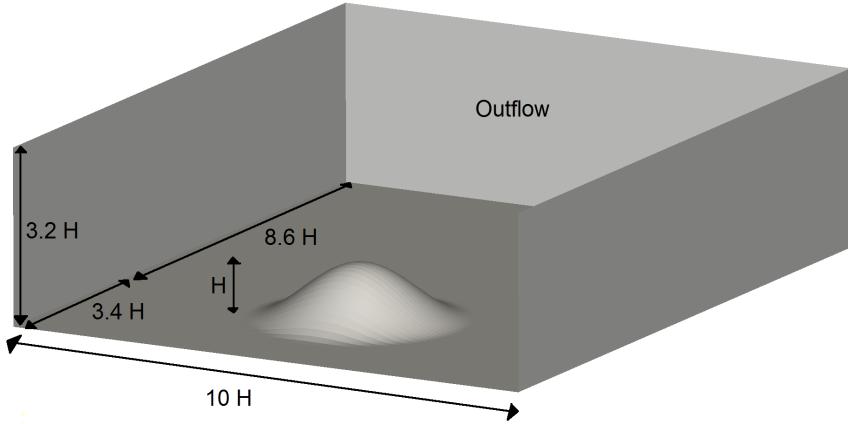


Figure 11: The bump case geometry seen from the inlet boundary.

The shape of the hill is defined by

$$\frac{x_2(r)}{H} = -\frac{1}{6.04844} \left[J_0(\Lambda)I_0\left(\Lambda \frac{r}{a}\right) - I_0(\Lambda)J_0\left(\Lambda \frac{r}{a}\right) \right] \quad (7.1)$$

where $\Lambda = 3.1926$, $a = 2H$, $H = 0.078$ m, is the height of the hill, a is the radius of the circular base of the hill and $r^2 = x_1^2 + x_3^2$. J_0 and I_0 are the Bessel function of the first kind and the modified Bessel function of the first kind, respectively. A Cartesian coordinate system was used for the simulations, and in the domain coordinates (x_1, x_2, x_3) , the velocity field is $\mathbf{u} = (u_1, u_2, u_3)$. The x_1 -axis is pointing downstream and the x_2 -axis is normal to the floor, pointing upward. The origin is fixed straight below the top of the hill, with $x_2 = 0$ corresponding to the tunnel floor.

The top and bottom patch seen in figure 11 are wall boundaries and this implies the use of no slip conditions, i.e. $u_i = 0$ at these boundaries. The left, right and outlet patches are used to limit the computational domain. Therefore homogeneous Neumann boundary conditions are used for these boundaries, hence $\frac{\partial \mathbf{u}}{\partial n} = 0$.

The inlet has a maximum flow velocity of $V_\infty = 27.5$ m/s. The velocity is not uniform, it is a profile with lower velocities near the top and bottom wall and higher in the middle of the inlet patch corresponding to a turbulent boundary layer with an approximate thickness of $0.3H$. The inlet velocity has only one non-zero component, that is in the x_1 -direction. The velocity profile can be seen in figure 12.

For the pressure p , Neumann boundary conditions are used for all patches except the outlet. Here, the pressure is set to be equal to zero.

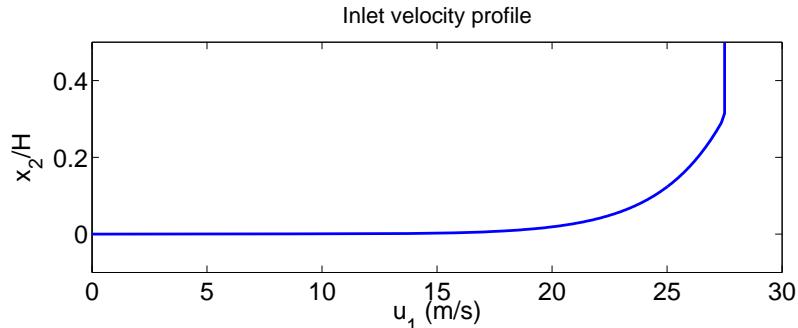


Figure 12: The bottom part of the velocity profile for the inlet patch of the computational domain. Note that the velocity profile is symmetric and has the same appearance near the top of the domain.

The Reynolds number for this case, based on V_∞ , the height of the hill, H and the kinematic viscosity of air, $\nu = 1.65 \cdot 10^{-5}$ m²/s, is $Re = 1.3 \cdot 10^5$.

7.1 Previous Research

This particular case has been extensively investigated by a number of computational research groups such as Garcia-Villalba et al. [8], Krajnović [9] and Bensow et al. [2]. Most of the previous research and the references therein for this case have relied on other turbulence models than RANS, such as DES (Detached Eddy Simulation), LES (Large Eddy Simulation) and hybrid LES-RANS. This master thesis builds upon results from the project course "Scientific Computing Advanced Course" at Uppsala University where this case was investigated, [10]. The primary aim of that project was to compare the performance of the different turbulence models implemented in OpenFOAM when simulating on one single wall refined mesh. One objective of the present work is to compare results from running RANS models on different meshes, both wall refined meshes and meshes generated on the basis of wall functions theory. The theory behind wall

functions is presented in section 2.9. Another objective of this work was to try some model that didn't manage to converge to a steady solution during the project course at Uppsala University. One of those models is the *LRR* Reynolds stress tensor model. We now try this turbulence model with better suited initial conditions for the Reynolds stress tensor.

7.2 Boundary conditions for the turbulent quantities

All the turbulence models that we used for the bump case, except the *LRR* model use the turbulent kinetic energy, k and the dissipation ϵ to describe the turbulence. The boundary conditions for k and ϵ are set to a constant value on the inlet patch. The values are $0.756 \text{ m}^2/\text{s}^2$ and $1.08 \text{ m}^2/\text{s}^3$ respectively.

For the top and bump patch, wall functions boundary conditions were used. For k it is set to be "kqRWallFunction" and simply acts as a Neumann boundary condition. For ϵ it is set to be "epsilonWallFunction" and is calculated according to equation (2.55). For the outlet, left and right patches, Neumann boundary conditions were used.

The *LRR* model needs boundary conditions for the Reynolds stress tensor. On the top and bump patch, it is set to be "kqRWallFunction". The inlet condition for the Reynolds stress tensor is computed from the boundary conditions for k and u_i . For the outlet, left and right patches, Neumann boundary conditions were used.

7.3 Mesh generation

Since many of the different RANS models implemented in OpenFOAM were tested and compared to LES and LDV data during the project in the course "Scientific Computing Advanced Course" [10], one of the objectives of this master's thesis was to run some of the best performing RANS models from this project course on meshes that are better suited for the RANS type turbulence models.

The mesh used in the course at Uppsala University consists of about 10^6 computational cells. It is a structured mesh that has been refined near the wall boundaries to better resolve the boundary layers and the flow near these regions. The number of cells in x_1 , x_2 and x_3 -direction is $69 \times 119 \times 119$ respectively. The y^+ value for this mesh is in the range between 1.9 and 4.1 for the bump patch. This mesh will in the following be referred to as M_{r1} since it is used with the bump case geometry and is refined near the wall boundaries.

The next mesh that was constructed has the same distribution of cells as the M_{r1} mesh. The only difference is that it has more cells in the x_2 direction, giving it better resolution in this direction. The number of cells in x_1 , x_2 and x_3 -direction is $69 \times 239 \times 119$ respectively. The y^+ value for this mesh is in the range between 0.8 and 2.2 for the bump patch. This mesh will in the following be referred to as M_{r2} .

Two other meshes were also constructed. They were constructed with the wall functions approach in mind, aiming for $y^+ \approx 30$ at the bump patch. The first has the same number and distribution of cells in the x_1 and x_3 direction as the M_{r1} mesh. But instead of being refined near walls in the x_2 direction, the cells are uniformly distributed. The number of cells in the x_2 direction for this mesh is 89. The y^+ value for this mesh is in the range between 37 and 112 for the bump patch. This mesh will in the following be referred to as M_{u1} since it has uniform distribution of cells in the x_2 direction.

The second mesh also has uniform distribution of cells in the x_2 direction. Compared to the M_{u1} mesh, this mesh has 109 cells in the x_2 direction. The refinement in this direction was made to lower the y^+ value to be closer to 30, to better agree with the wall functions theory. This mesh was also refined in the x_1 direction, to better resolve the separated flow behind the hill.

Types of meshes for the bump case		
Mesh	Number of cells	Range of y^+ for the bump patch
M_{r1}	977109	$1.9 < y^+ < 4.1$
M_{r2}	1962429	$0.8 < y^+ < 2.2$
M_{u1}	730779	$37 < y^+ < 112$
M_{u2}	1802969	$25 < y^+ < 95$

Table 5: Meshes used for the bump case together with values of y^+ . The y^+ values are from simulations with the realizable $k - \epsilon$ model.

The refinement in this direction also improved the ratio between height and width of the cells. The number of cells in x_1 , x_2 and x_3 -direction for this mesh is $139 \times 109 \times 119$ respectively. The y^+ value for this mesh is in the range between 25 and 95 for the bump patch. This mesh will in the following be referred to as M_{u2} .

The two different types of meshes are shown in figure 13. For an overview of the mesh characteristics, see table 5. The y^+ value for the different meshes, computed from simulations using the realizable $k - \epsilon$ model is visualized in figure 14.

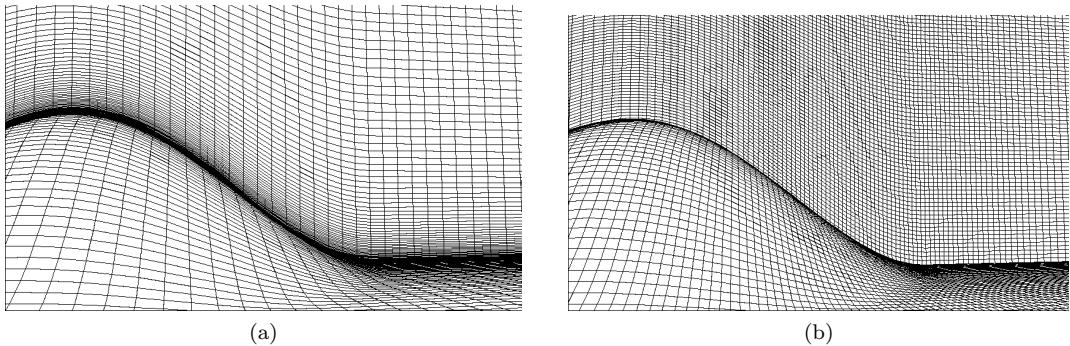


Figure 13: Two different types of meshes showing the cells on the surface of the hill and in the plane $x_3 = 0$. The positive x_1 direction is to the right. (a) The M_{r1} mesh. Notice the refinement of the mesh in the x_2 direction near the hill. (b) The M_{u2} mesh. Notice the uniform distribution of cells in the x_2 direction. Also notice the refinement in the x_1 direction compared to the M_{r1} mesh.

7.4 Short description of the physical experiment

Results from the LDV measurements by Byun and Simpson [1] states that there is no separation in front of the bump but that the flow decelerates there and then accelerates until the top of the bump. The mean flow on the lee side is closely symmetric around the centerline and complex vortical separation occurs downstream from the top and merge into large-scale turbulent eddies with two large streamwise vortices. The flow along the streamwise centerline at $x_1/H = 3.63$ is a downwashing reattachment flow. The LDV experiment shows, with resulting velocity vectors in the plane of $x_3/H = 0$, that the mean location of separation is at $x_1/H = 0.96$. A more detailed description of the results from the LDV experiment and comparison with our results is

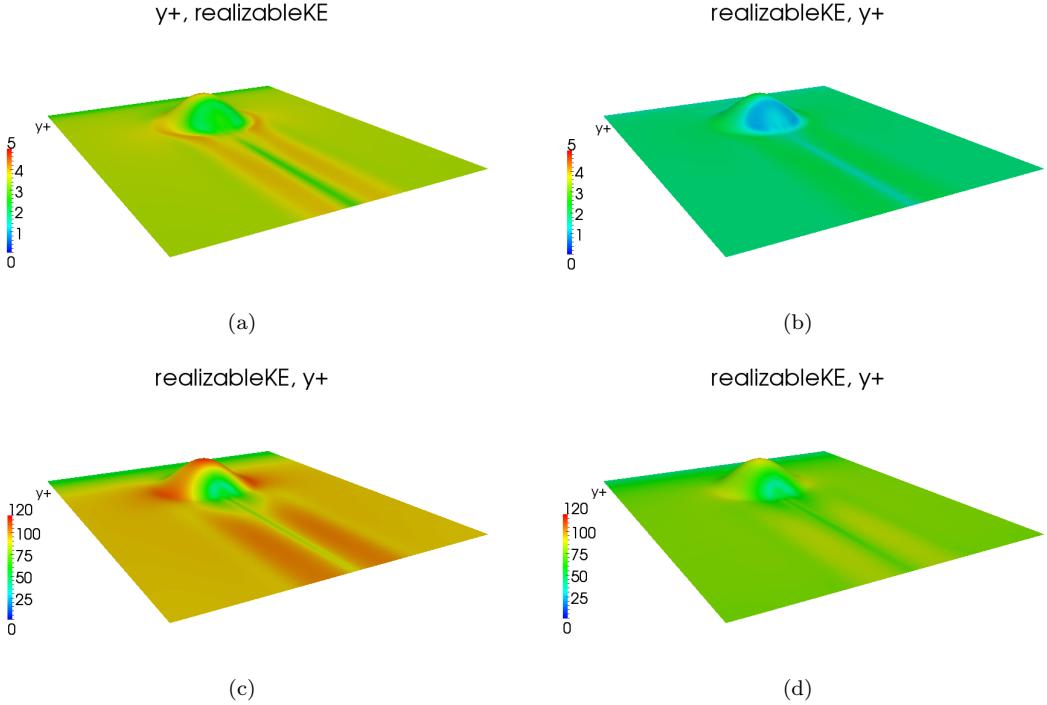


Figure 14: Plots of y^+ for the different meshes, computed from simulations with the realizable $k - \epsilon$ model. (a) The M_{r1} mesh. (b) The M_{r2} mesh. (c) The M_{u1} mesh. (d) The M_{u2} mesh.

discussed in 7.9.

7.5 Results from the bump case simulations

All of the simulations on the different meshes that converged capture the magnitude of the velocity quite reasonably in front of and on top of the bump, compared to the physical solution. However, behind the bump, separation of the flow is not found for all simulations. In general, the biggest difference is seen when comparing the results from the different meshes. When comparing the results from different turbulence models on the same mesh, the difference is small. The separation region is bigger for the wall refined meshes M_{r1} and M_{r2} , compared with the separation region found when using the M_{u1} and M_{u2} meshes. A summary of the models tested and on which mesh they were tested is shown in table 6. Comparisons to the LDV measurements and LES data are found in section 7.9.

Turbulence Model	Different meshes			
	M_{Br1}	M_{Br2}	M_{Bu1}	M_{Bu2}
realizable $k - \epsilon$	x	x	x	x
RNG $k - \epsilon$	x			x
LRR	x			x

Table 6: Tested turbulence models and meshes for the bump case.

7.6 The realizable $k - \epsilon$ model

This model was simulated on all the meshes described in section 7.3. All the meshes manage to capture the flow separation. However, the size of the wake differ. The wake produced by the simulations on the M_{r1} and M_{r2} meshes are thicker and the flow separation start further up on the leeward side of the hill. The flow is also re-attached further away, for larger values of x_1/H . Two vortices are produced on all the meshes, but they are much bigger on the M_{r1} and M_{r2} meshes. The turbulent energy produced on the two types of meshes also differ. For the M_{u1} and M_{u2} meshes, the maximum turbulent kinetic energy is less than half compared with k produced on the M_{r1} and M_{r2} meshes. This is consistent with the fact that there should be larger values of the turbulent kinetic energy where there are big eddies.

7.6.1 Results from simulation on the M_{r1} mesh

As seen in figure 15 (a), the flow layer near the bottom of the domain decelerates before the hill and then accelerates on the top to finally decelerate again on the leeward side of the hill. On this mesh, the realizable $k - \epsilon$ captures the flow separation. The flow separation occurs at $x_1/H = 0.51$ and is re-attached around $x_1/H \approx 2.24$. This can be seen in figure 15. Two vortices turn up on the leeward side of the hill and are almost symmetrically placed around the plane $x_3/H = 0$. This is shown in figure 16 (a).

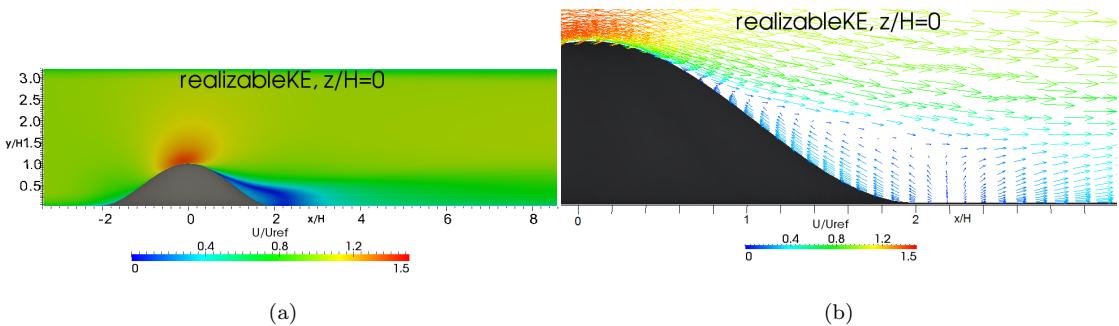


Figure 15: Results from the realizable $k - \epsilon$ model on the M_{r1} mesh. (a) The magnitude of the velocity at $x_3/H=0$. (b) The velocity field in the wake on the leeward side of the hill at $x_3/H=0$.

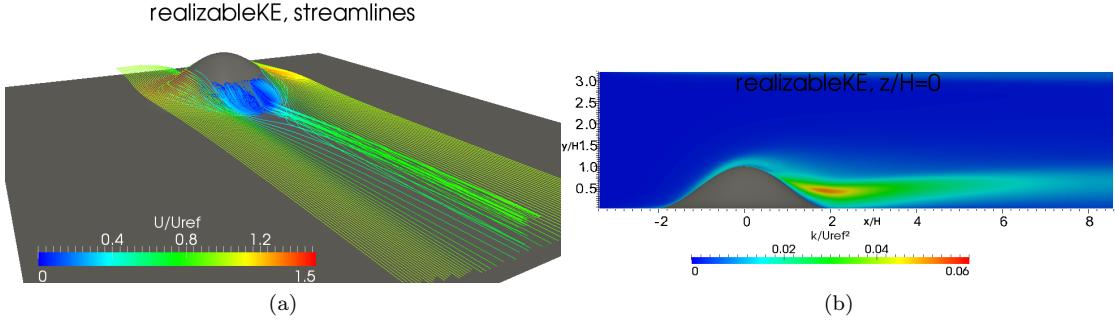


Figure 16: Results from the realizable $k - \epsilon$ model on the M_{r1} mesh. (a) Streamlines of the velocity in x_1 -direction through the line from $[0.125 \ 0.02 \ 0.15]$ to $[0.125 \ 0.02 \ -0.15]$. (b) The turbulent kinetic energy k , at $x_3/H=0$.

7.6.2 Results from simulation on the M_{r2} mesh

On this mesh, the realizable $k - \epsilon$ captures the flow separation. Flow separation occurs at $x_1/H = 0.40$ and is re-attached around $x_1/H = 2.32$. This can be seen in figure 15. Thus, the separation occurs earlier and the flow is re-attached later compared to the result on the M_{r1} mesh. The separation region is also thicker compared with the result on the M_{r1} mesh. Two vortices turn up on the leeward side of the hill and are almost symmetrically placed around the plane $x_3/H = 0$. This is shown in figure 16 (a).

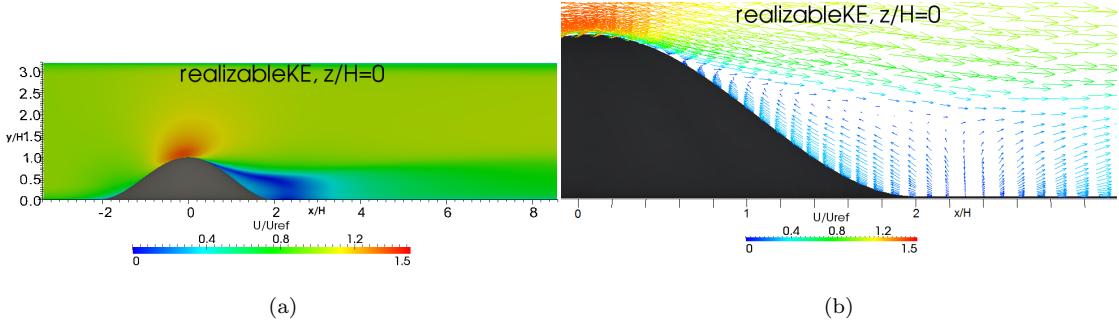


Figure 17: Results from the realizable $k - \epsilon$ model on the M_{r2} mesh. (a) The magnitude of the velocity at $x_3/H=0$. (b) The velocity field in the wake on the leeward side of the hill at $x_3/H=0$.

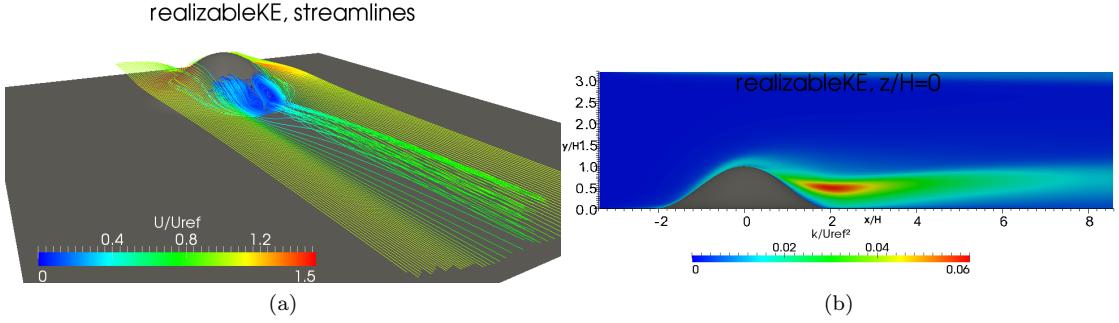


Figure 18: Results from the realizable $k - \epsilon$ model on the M_{r2} mesh. (a) Streamlines of the velocity in x_1 -direction through the line from $[0.125 \ 0.03 \ 0.15]$ to $[0.125 \ 0.03 \ -0.15]$. (b) The turbulent kinetic energy k , at $x_3/H=0$.

7.6.3 Results from simulation on the M_{u1} mesh

The separation region is captured on this mesh. However, the region is both thinner and smaller in the x_1 -direction compared with the results from the M_{r1} and M_{r2} meshes. Flow separation occurs at $x_1/H = 1.30$ and is re-attached at $x_1/H = 1.88$. This can be seen in figure 19. Two small vortices turn up on the leeward side of the hill and are almost symmetrically placed around the plane $x_3/H = 0$. This is shown in figure 20 (a).

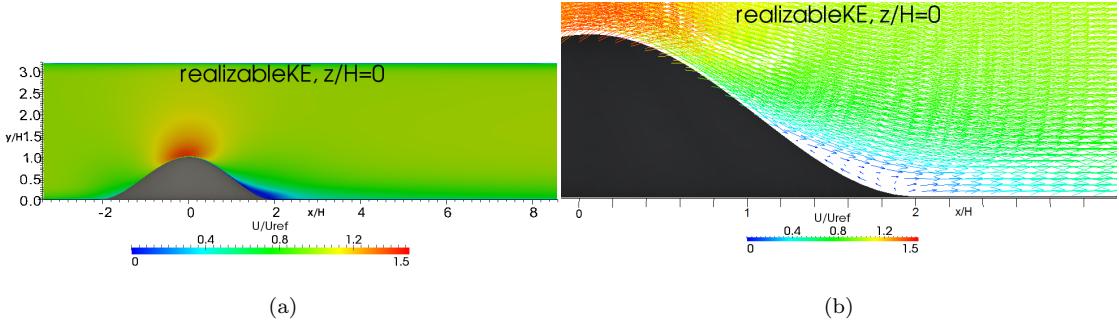


Figure 19: Results from the realizable $k - \epsilon$ model on the M_{u1} mesh. (a) The magnitude of the velocity at $x_3/H=0$. (b) The velocity field in the wake on the leeward side of the hill at $x_3/H=0$.

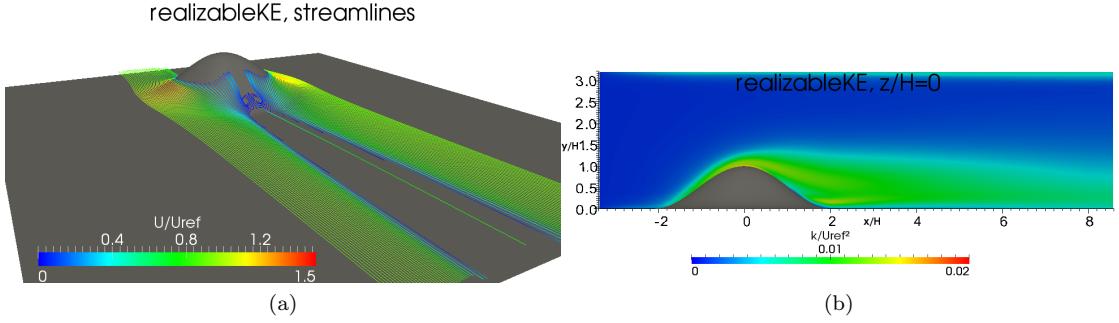


Figure 20: Results from the realizable $k - \epsilon$ model on the M_{u2} mesh. (a) Streamlines of the velocity in x_1 -direction through the line from $[0.125 \ 0.01 \ 0.15]$ to $[0.125 \ 0.01 \ -0.15]$. (b) The turbulent kinetic energy k , at $x_3/H=0$.

7.6.4 Results from simulation on the M_{u2} mesh

The separation region is captured on this mesh. However, as on the M_{u1} mesh, the region is thinner and smaller in x_1 -direction compared with the results from the M_{r1} and M_{r2} meshes. Flow separation occurs at $x_1/H = 1.43$ and is re-attached around $x_1/H = 1.68$. This can be seen in figure 21. Two small vortices turn up on the leeward side of the hill and are almost symmetrically placed around the plane $x_3/H = 0$. This is shown in figure 22 (a).

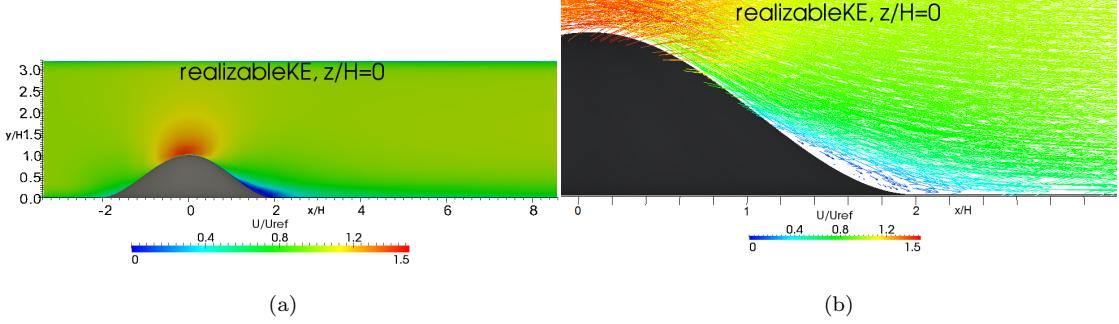


Figure 21: Results from the realizable $k - \epsilon$ model on the M_{u2} mesh. (a) The magnitude of the velocity at $x_3/H=0$. (b) The velocity field in the wake on the leeward side of the hill at $x_3/H=0$.

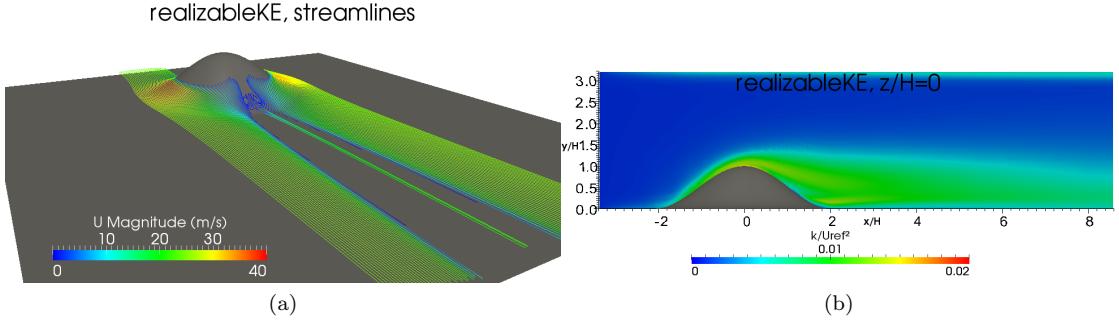


Figure 22: Results from the realizable $k - \epsilon$ model on the M_{u2} mesh. (a) Streamlines of the velocity in x_1 -direction through the line from $[0.125 \ 0.01 \ 0.15]$ to $[0.125 \ 0.01 \ -0.15]$. (b) The turbulent kinetic energy k , at $x_3/H=0$.

7.7 The RNG $k - \epsilon$ model

This model was simulated on the M_{r1} and M_{u2} meshes. Separation of flow is only found on the M_{r1} mesh. The turbulent energy produced on the two meshes also differ. On the M_{u2} mesh, the maximum turbulent kinetic energy is found on top of the hill and not on the leeward side of the hill, as on the M_{r1} mesh.

7.7.1 Results from simulation on the M_{r1} mesh

On this mesh, the RNG $k - \epsilon$ model captures the flow separation. Flow separation occurs at $x_1/H = 0.80$ and is re-attached around $x_1/H = 2.07$. This can be seen in figure 23 Two vortices turn up on the leeward side of the hill and are almost symmetrically placed around the plane $x_3/H = 0$. This is shown in figure 24 (a).

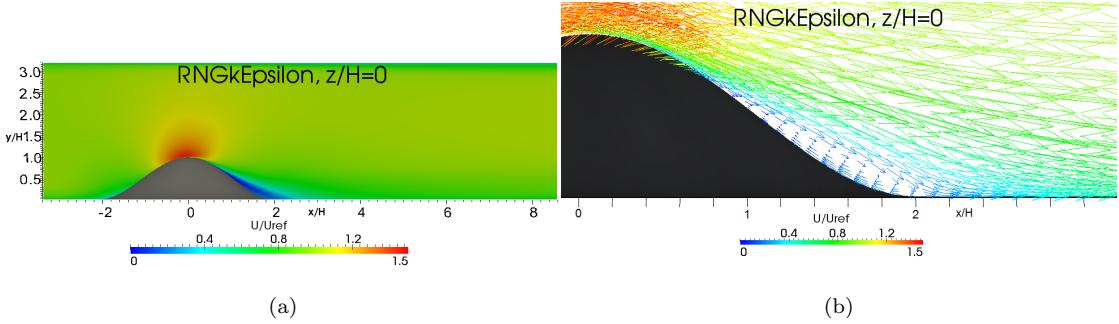


Figure 23: Results from the RNG $k - \epsilon$ model on the M_{r1} mesh. (a) The magnitude of the velocity at $x_3/H=0$. (b) The velocity field in the wake on the leeward side of the hill at $x_3/H=0$.

7.7.2 Results from simulation on the M_{u2} mesh

On this mesh, the RNG $k - \epsilon$ model does not capture any separation of flow and no vortices or eddies are captured.

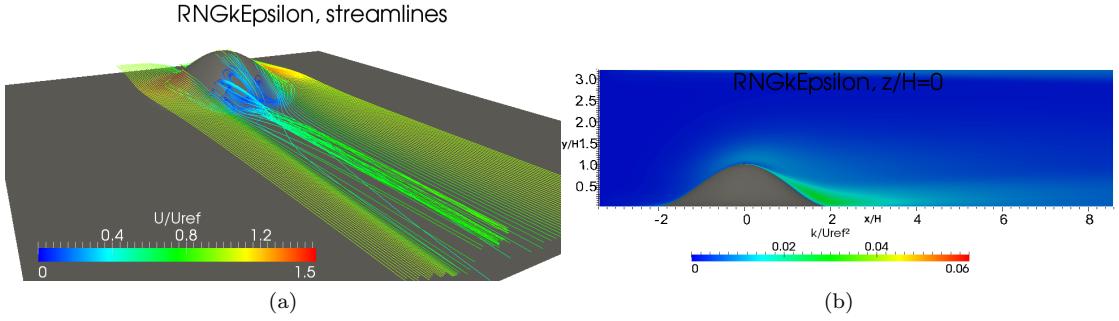


Figure 24: Results from the RNG $k - \epsilon$ model on the M_{r1} mesh. (a) Streamlines of the velocity in x_1 -direction through the line from $[0.125 \ 0.01 \ 0.15]$ to $[0.125 \ 0.01 \ -0.15]$. (b) The turbulent kinetic energy k , at $x_3/H=0$.

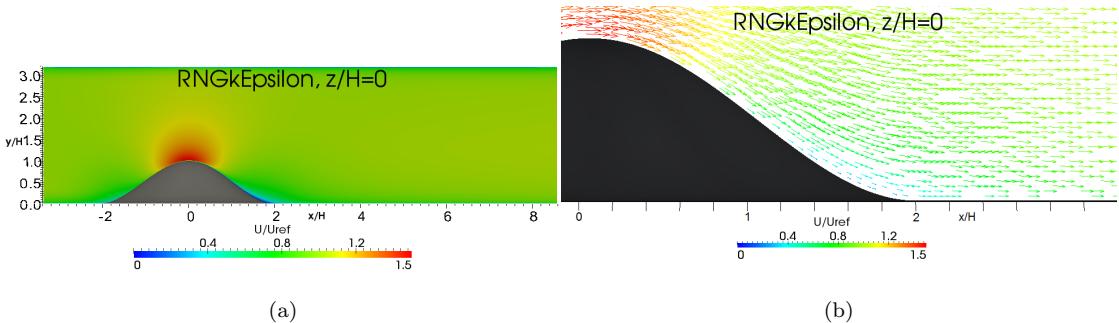


Figure 25: Results from the RNG $k - \epsilon$ model on the M_{u2} mesh. (a) The magnitude of the velocity at $x_3/H=0$. (b) The velocity field in the wake on the leeward side of the hill at $x_3/H=0$. Notice that there is no recirculation or separation of flow.

7.8 The LRR model

When testing this model during the project course [10], we did not manage to get a convergent, steady solution. This model was simulated on the M_{r1} and M_{u2} meshes. Like the RNG $k - \epsilon$ model, separation of flow is only found on the M_{r1} mesh.

7.8.1 Results from simulation on the M_{r1} mesh

On this mesh, the LRR model captures the flow separation. Flow separation occurs at $x_1/H = 0.77$ and is re-attached around $x_1/H 2.03$. This can be seen in figure 27. Two vortices turn up on the leeward side of the hill and are almost symmetrically placed around the plane $x_3/H = 0$. This is shown in figure 28 (a).

7.8.2 Results from simulation on the M_{u2} mesh

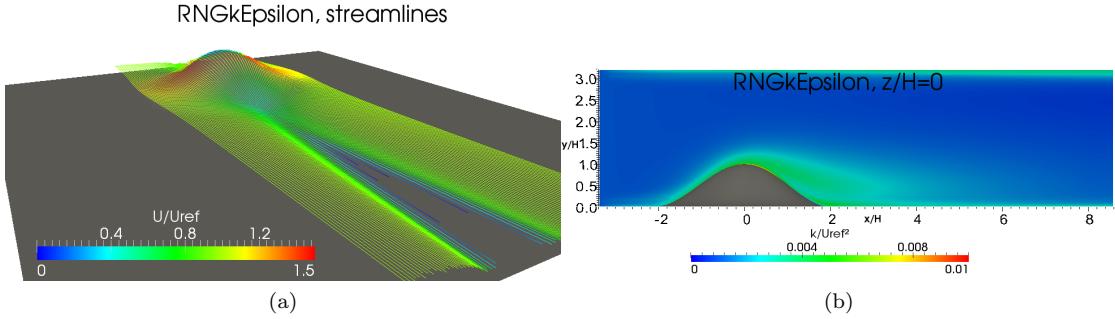


Figure 26: Results from the RNG $k - \epsilon$ model on the M_{u2} mesh. (a) Streamlines of the velocity in x_1 -direction through the line from [0.125 0.01 0.15] to [0.125 0.01 -0.15]. (b) The turbulent kinetic energy k , at $x_3/H=0$.

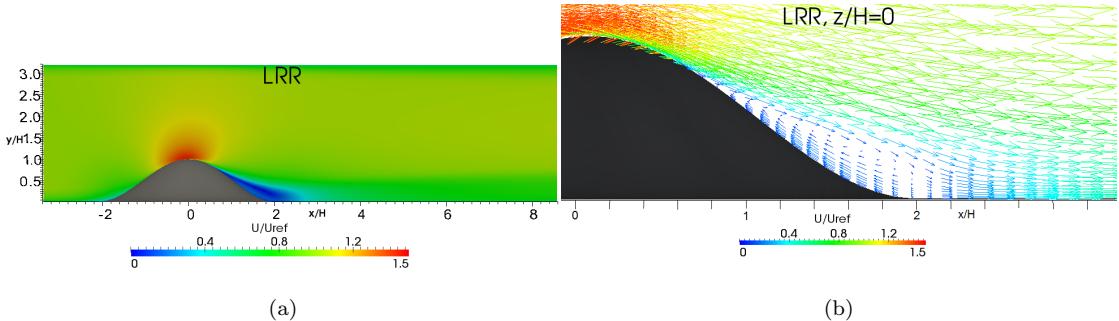


Figure 27: Results from the LRR model on the M_{r1} mesh. (a) The magnitude of the velocity at $x_3/H=0$. (b) The velocity field in the wake on the leeward side of the hill at $x_3/H=0$.

7.9 Comparison to LES and LDV data

Results of the LDV investigations of the flow around the hill were reported by Byun and Simpleso 2005, [1]. From their result, the flow in the $x_3 = 0$ plane, is separated at $x_1/H = 0.96$. Note that not all of our simulations have flow separation. The x_1 coordinate for the separation and re-attachement points for each simulation is shown in table 7. If we look at the results from the LDV-experiment in the plane $x_1/H = 3.69$, there are two vortices centred at $x_3/H = \pm 1.35$. Some similar results can be found from a couple of our simulations. Some of our simulations show strange results with vortices near and around the center line $x_3 = 0$. These results primarily occur on the M_{r1} and M_{r1} meshes and are not found in the LDV-experiment. The results from our simulations in the plane $x_1/H = 3.69$ can be seen in figure 34, 35, 36 and 37. The line plots in figure 32 and 33 show the x_1 and x_3 components of the velocity, respectively, at $x_1/H = 3.69$. Note that these plots show the velocity components behind the separation region. Therefore, u_1 is positive for all x_2/H values in each plot. However, there are still vortices in this region, but they appear in the x_1 plane. The lines where data are taken are shown in figure 31.

The results for the x_1 component of the velocity from our simulations are quite similar with the LDV and LES data except some differences around $x_3 = 0$.

If we look at the x_3 component there is a clear distinction between our results. The flow directions of the eddies are opposite compared with the LDV and LES data, for the results from

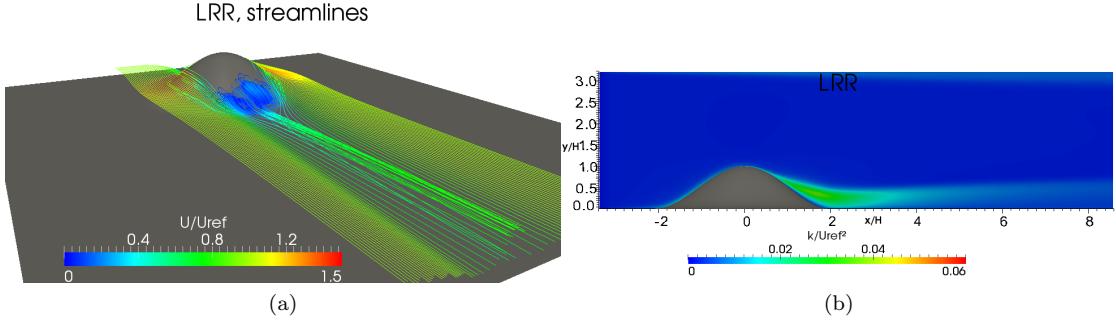


Figure 28: Results from the LRR model on the M_{r1} mesh. (a) Streamlines of the velocity in x_1 -direction through the line from $[0.125 \ 0.01 \ 0.15]$ to $[0.125 \ 0.01 \ -0.15]$. (b) The turbulent kinetic energy k , at $x_3/H=0$.

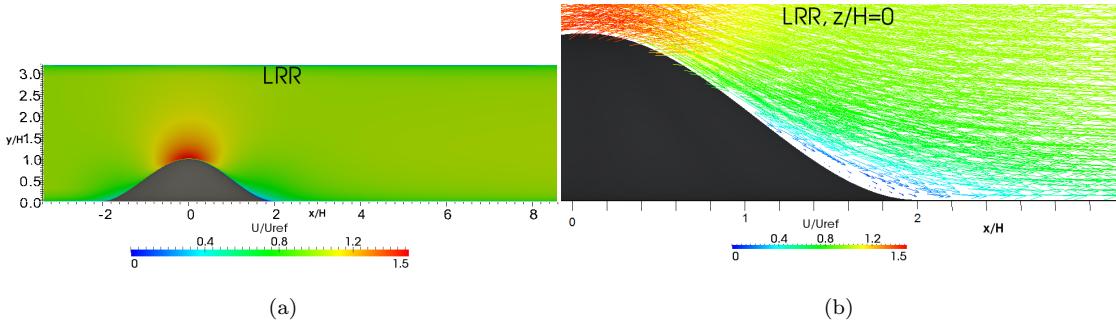
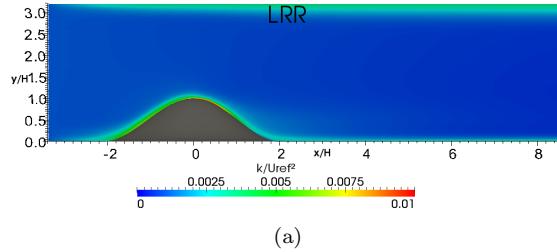


Figure 29: Results from the LRR model on the M_{u2} mesh. (a) The magnitude of the velocity at $x_3/H=0$. (b) The velocity field in the wake on the leeward side of the hill at $x_3/H=0$.

Model and mesh	Separation point, x_1/H	Re-attachment point, x_1/H
LDV	0.96	2.0
realizable $k - \epsilon$, M_{r1} mesh	0.51	2.24
realizable $k - \epsilon$, M_{r2} mesh	0.40	2.32
realizable $k - \epsilon$, M_{u1} mesh	1.30	1.88
realizable $k - \epsilon$, M_{u2} mesh	1.43	1.68
RNG $k - \epsilon$, M_{r1} mesh	0.80	2.07
RNG $k - \epsilon$, M_{u2} mesh	—	—
LRR, M_{r1} mesh	0.77	2.03
LRR, M_{u2} mesh	—	—

Table 7: Table of separation and re-attachment points for different models and LDV experiment. Note that no obvious re-attachment point is found in the LDV-experiment.

simulations on the M_{r1} mesh. The results from the M_{u2} mesh is consistent with the reference data in the direction of flow.



(a)

Figure 30: Results from the LRR model on the M_{u2} mesh. (a) Streamlines of the velocity in x_1 -direction through the line from [0.125 0.01 0.15] to [0.125 0.01 -0.15]. (b) The turbulent kinetic energy k , at $x_3/H=0$.

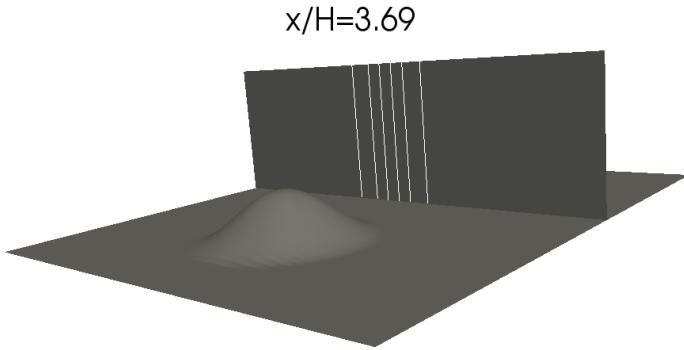


Figure 31: The plane $x_1/H = 3.69$ and lines where data are taken for the plots in figure 32 and 33.

7.10 Discussion and conclusions of the bump results

The results from the bump case were analysed and compared to both experiment- and LES-data. Simulations with RANS models were made on the bump geometry during a course at Uppsala University [10]. One of the aims with this work was to see if the results from some of the best performing models that was produced there, could be improved with better adjusted meshes.

Three different RANS models were tested on this case: the realizable $k - \epsilon$ model, the RNG $k - \epsilon$ model and the *LRR* Reynolds stress tensor model. Two new types of meshes were developed. The first type, more refined near the surface of the hill, compared with the mesh used at Uppsala University. This new mesh is the M_{r2} mesh and the mesh used at Uppsala University is the M_{r1} mesh. When analysing the results from M_{r2} mesh, no improvement could be seen. Rather the reverse, the shortcomings in the results from the work at Uppsala University became worse, with a larger separation region. Only the realizable $k - \epsilon$ model was tested on this new mesh. The second type of mesh was developed to be better suited when using WFBC. The primary aim with this mesh was to get $y^+ > 30$ near and around the separation region. Two meshes of this type were developed, the M_{u1} and M_{u2} mesh. The results from this type of mesh is very different compared to the results on the wall refined mesh. The results from both the *LRR* Reynolds stress tensor model and the RNG $k - \epsilon$ model on the M_{u2} mesh show no separation at all. A very thin separation bubble is only found from the realizable $k - \epsilon$ model. Among the tested models, the realizable $k - \epsilon$ model is also the one producing the thickest separation region on the

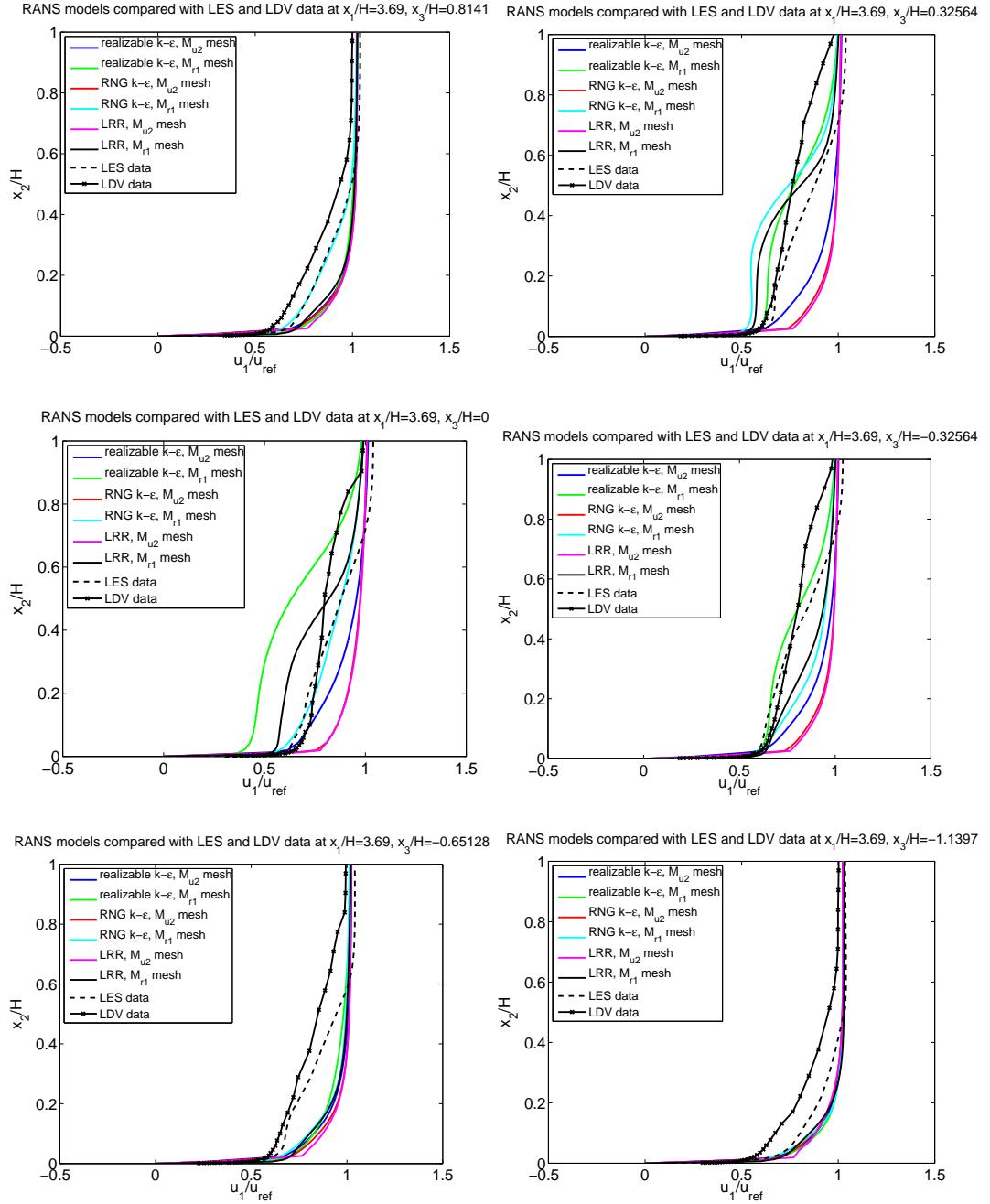


Figure 32: Comparison of RANS models, LES and LDV. Data is for six different values of x_3/H .

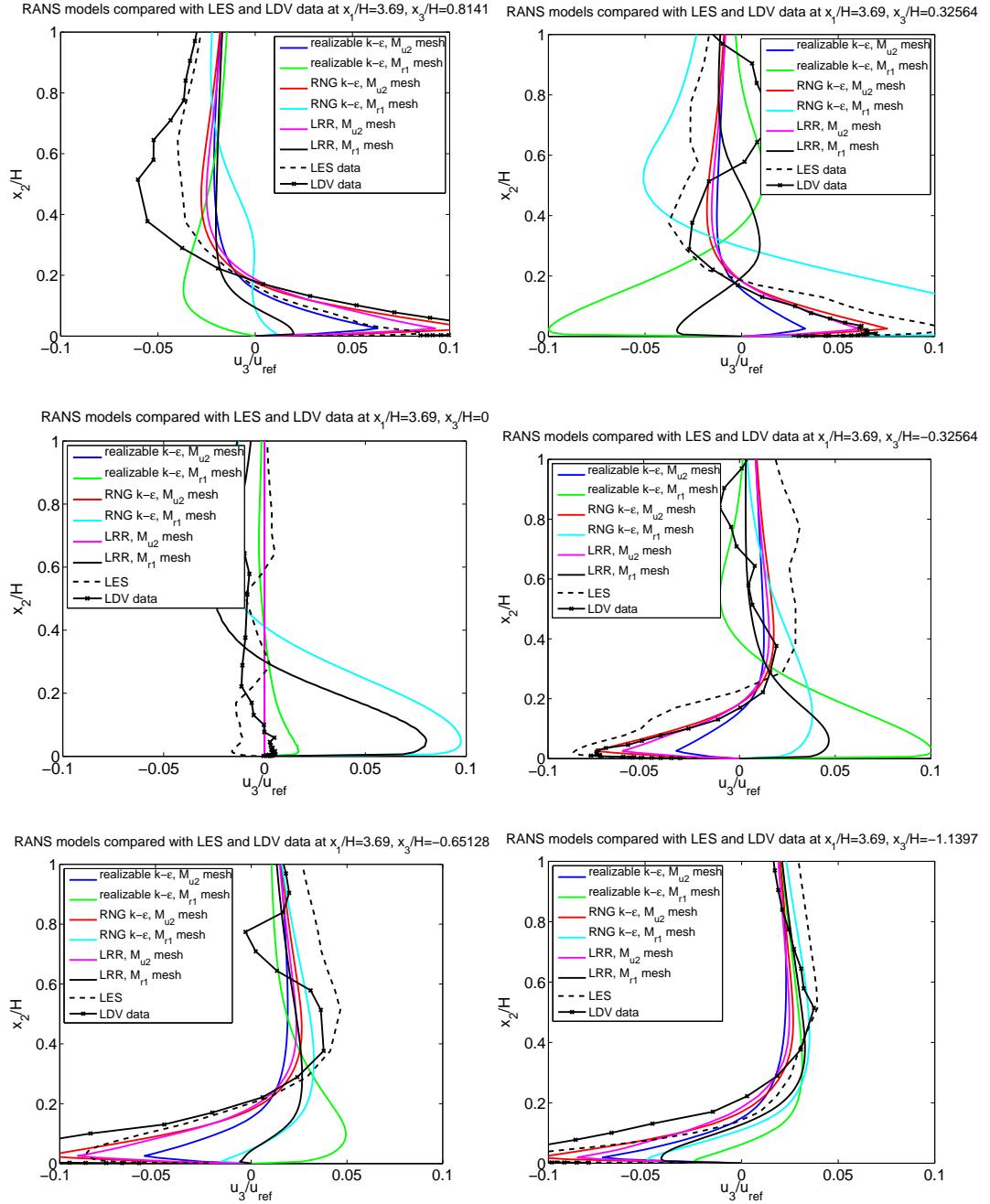


Figure 33: Comparison of RANS models, LES and LDV. Data is for six different values of x_3/H .

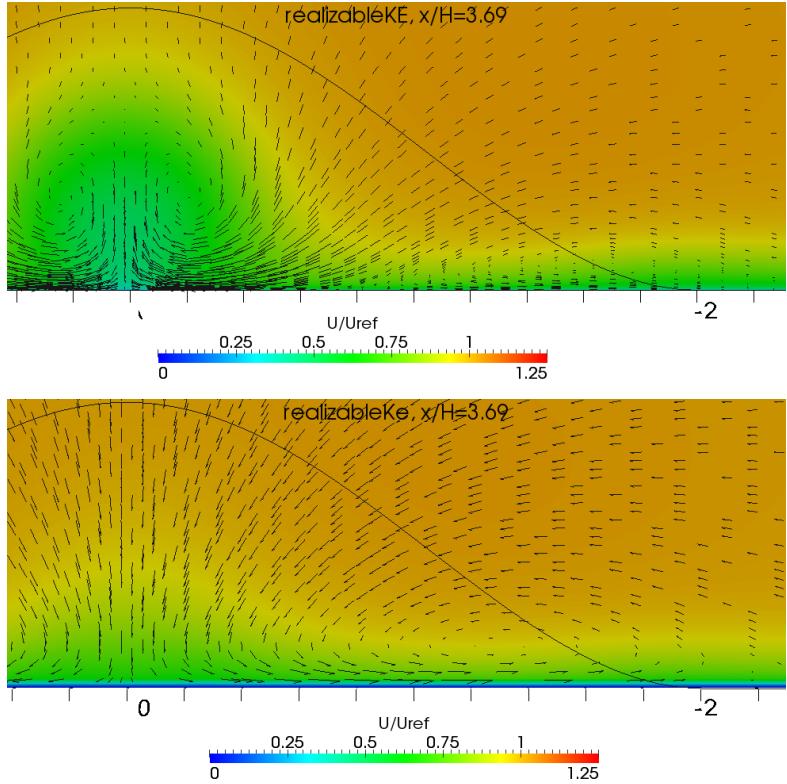


Figure 34: Plot of the magnitude and vector field of u_2 and u_3 in the plane $x_1/H = 3.69$. Results are from the realizable $k - \epsilon$ model on the M_{r1} and M_{u2} meshes respectively. The view is from behind the hill, in the negative x_1 direction.

M_{r1} mesh.

When comparing our results from the M_{r1} mesh with experimental and LES data in the plane $x/H = 3.69$ we find major differences. Our RANS results show a pair of counter-rotating vortices near the center line $x_3/H = 0$ that do not appear in the LES and experimental results. However, our results on the M_{u2} mesh show none or very small eddies in this region. These results can be seen in figure 34.

The meshes developed to be used together with WFBC are to coarse near the boundary to be able to resolve the thin separation zone, which is the biggest disadvantage with this type of mesh. One the other hand, It produces results in the plane $x_1/H = 3.69$ that coincide with the LES and LDV results. This might be because the mesh is adjusted for the boundary conditions used and the fact that the eddie motions occur far away from the wall; the coarseness of the mesh is not an issue here.

Improvements for the bump results could be to use a mesh that produces y^+ values below 1 together with constant boundary conditions for k and ϵ . This mesh should be refined enough to resolve the gradients of the turbulent quantities near walls.

Other improvements could be to use a mesh that is very refined in the region behind the hill where the separation occurs and less refined in regions where there is less turbulence. Using this typ of mesh, one should use boundary conditions that switch between WFBC type and constant values of k and ϵ , depending on the local value of y^+ . This approach could be useful since it

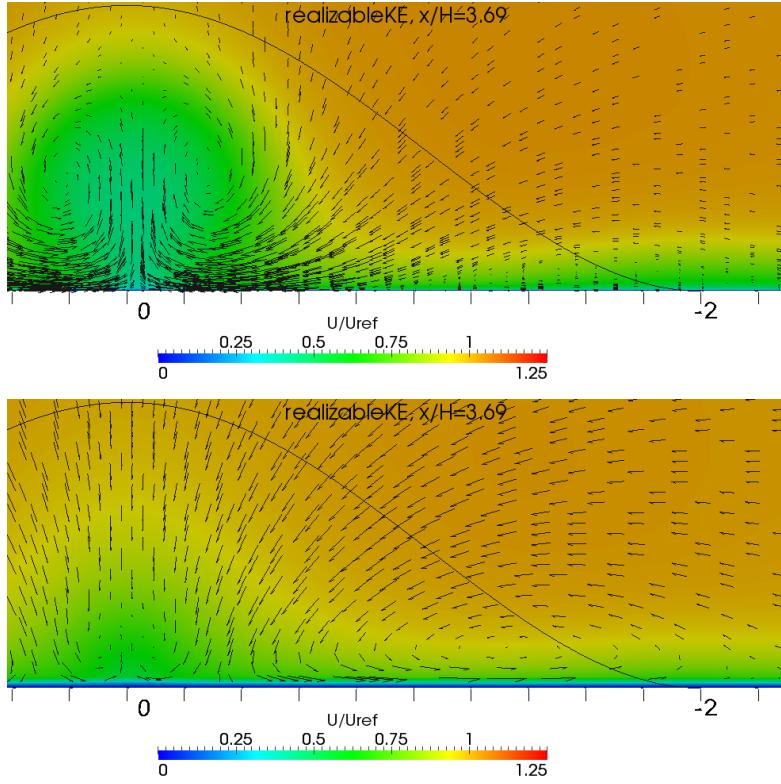


Figure 35: Plot of the magnitude and vector field of u_2 and u_3 in the plane $x_1/H = 3.69$. Results are from the realizable $k - \epsilon$ model on the M_{r2} and M_{u1} meshes respectively. The view is from behind the hill, in the negative x_1 direction.

reduces the computational effort, having refined mesh only where it is necessary.

The best performing model on the bump geometry is the realizable $k - \epsilon$ model with WFBC together with the M_{u1} mesh. The results from this simulation coincide with the LES and LDV results in the plane $x_1/H = 3.69$, and the separation region is also captured, but is too small in the streamwise direction.

8 Summary and general conclusions

In this work, the standard two equation $k - \epsilon$ turbulence model is presented. The RANS equations and the equation for the turbulent kinetic energy are derived from the Navier-Stokes equations. The theory of wall flow is also investigated together with turbulence modelling near walls. The implementation of the $k - \epsilon$ model and the near walls turbulence modelling in OpenFOAM are investigated. We found that the term $-\frac{\partial}{\partial x_j} \left(\left(\frac{\nu_T}{\sigma_\epsilon} \right) \frac{\partial \epsilon}{\partial x_j} \right)$ from the $k - \epsilon$ theory of [3] is implemented a bit different, namely $-\frac{\partial}{\partial x_j} \left((\nu + \frac{\nu_T}{\sigma_\epsilon}) \frac{\partial \epsilon}{\partial x_j} \right)$.

The RANS turbulence models were simulated on two different geometries and the result was analysed.

The results from the RANS turbulence models in this work depend on the meshes and on the

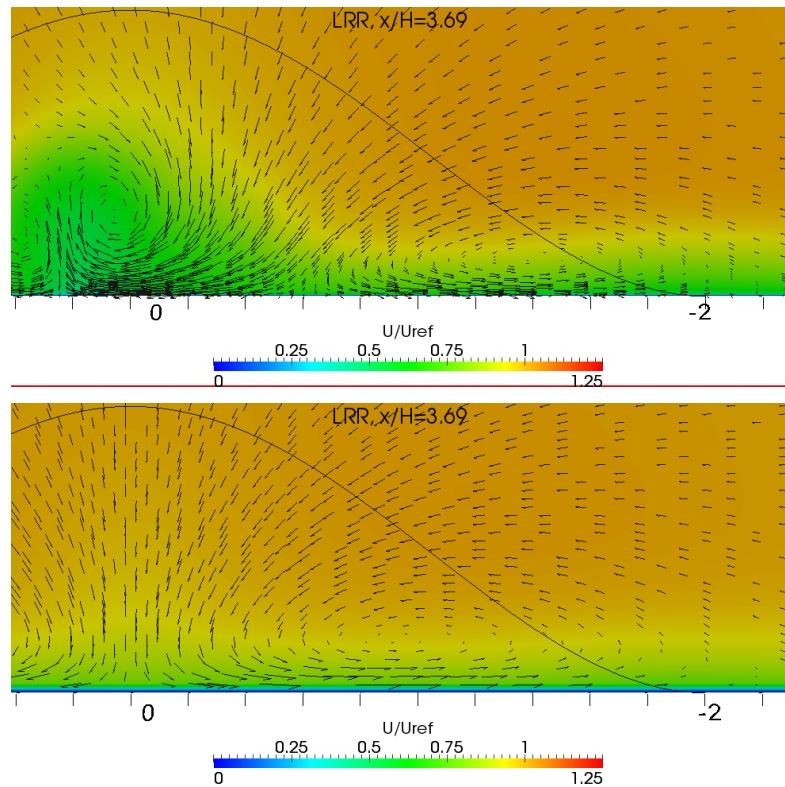


Figure 36: Plot of the magnitude and vector field of u_2 and u_3 in the plane $x_1/H = 3.69$. Results are from the *LRR* Reynolds stress tensor model on the M_{r1} and M_{u2} meshes respectively. The view is from behind the hill, in the negative x_1 direction.

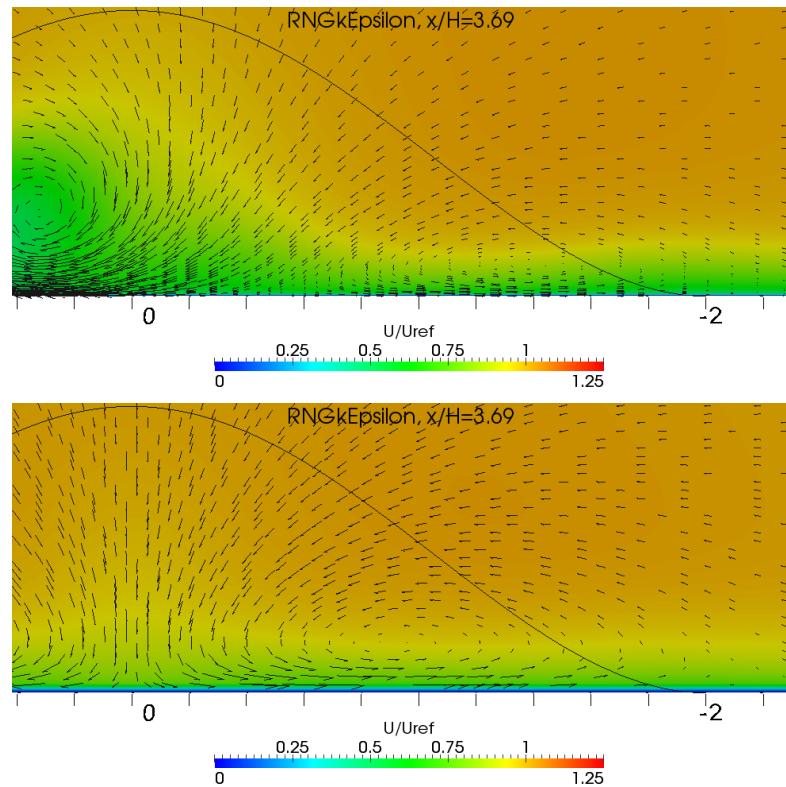


Figure 37: Plot of the magnitude and vector field of u_2 and u_3 in the plane $x_1/H = 3.69$. Results are from the RNG $k-\epsilon$ model on the M_{r1} and M_{u2} meshes respectively. The view is from behind the hill, in the negative x_1 direction.

type of boundary conditions used for the turbulent quantities. More work to fully understand which combination of mesh and type of boundary conditions to use could improve the results. Which type of turbulence models to use when not using WFBC could also be interesting and improve the results. Overall improvements could probably be achieved for both the cases if more effort was put to design meshes.

My recommendation from this study is to use the realizable $k - \epsilon$ model together with WFBC and a mesh that is suited for these types of boundary conditions.

References

- [1] Byun G. and Simpson R. L. Structures of three-dimensional separated flow on an axisymmetric bump. *43rd AIAA Aerospace Sciences Meeting and Exhibit*, 2005. Paper No. 2005-0113.
- [2] Bensow R. E., Fureby C., Liefvendahl M., and Persson T. Numerical investigation of the flow over an axisymmetric hill using LES, DES and RANS. *Journal of Turbulence*, 7(4):1–17, 2006.
- [3] Pope S. B. *Turbulent Flows*. Cambridge University Press, 2000.
- [4] Ferziger J. H. and Perić M. *Computational Methods for Fluid Dynamics*. Springer, 3rd edition, 2002.
- [5] OpenCFD Limited. *OpenFOAM - Programmer's Guide*, July 2009. version 1.6.
- [6] Casey M. and Wintergerste T., editors. *Best Practice Guidelines*. ERCOFTAC, 1.0 edition, 2000.
- [7] Daily J. W. and Pitz R. W. Combustion in a turbulent mixing layer formed at a rearward-facing step. *AIAA journal*, 21(11):1565–1570, 1983.
- [8] Garcia-Villalba M., Leschziner M. A., Li N., and Rodi W. Large-eddy simulation of separated flow over a three-dimensional axisymmetric hill. *J. Fluid Mech.*, pages 1–42, 2009.
- [9] Krajnović S. Large eddy simulation of the flow over a three-dimensional hill. *Flow Turbulence Combust*, 81:189–204, December 2008.
- [10] Furbo E., Harju J., and Nilsson H. Evaluation of turbulence models for prediction of flow separation at a smooth surface. Unpublished, may 2009.