




Universitetet
i Stavanger

FACULTY OF SCIENCE AND TECHNOLOGY

MASTER'S THESIS

Study programme/specialisation: Mechanical and Structural Engineering and Material Science Specialization: Mechanical Engineering	Spring semester, 2019 Open
Author: Ragnhild Glomnes Torgersen	 (signature of author)
Programme coordinator: Dimitros Pavlou Supervisor: Knut Erik Giljarhus	
Title of master's thesis: Evaluation of biomimetic approach to drag reduction of time trial helmet using computational fluid dynamics	
Credits: 30	
Keywords: CFD, biomimetic, sports aerodynamics, time trial helmet	Number of pages: 57 Stavanger, 15/06/2019 date/year

Abstract

Computational fluid dynamic (CFD) using OpenFOAM are performed to investigate aerodynamics of a time trial bicycle helmet. Mesh generation by the two softwares OpenFOAM and Pointwise are compared, to detect differences between them. A new type of helmet, inspired by the shell of an armadillo is presented. Results are compared with earlier wind tunnel experiment to validate if simulations without the cyclist is feasible. Simulations have focus on drag force and how mesh influence the results. Results indicates the same tendencies as wind tunnel experiment performed with a helmet attached to human. Computational time will be saved running simulations with only the helmet. Quality check gave better results for the mesh created with Pointwise than for OpenFOAM. Pointwise's properties converged with half of the amount of cells compared to OpenFOAM's, and were less time consuming. Pointwise generated proper layers despite coarse far field, while OpenFOAM require finer far field to properly generate layers.

The original helmet were modified by inverse kinematics, forcing a smaller frontal area. Results showed an reduction in air resistance at 22% and 13%.

Preface

This thesis is written as a final subject in my master degree program, Mechanical and Structural Engineering and Materials Science at the University of Stavanger (UiS). With help from my supervisor, associate professor Knut Erik Giljarhus, the process has been enjoyable. I am extremely grateful for the guidance, support and patience during the process.

I would like to use this opportunity to thank Farangis Bagheri who took her time to give me a private lesson in Pointwise. My fellow students for sharing ideas, knowledge and experience during the process. My boyfriend, friends and family for great support and understanding during this semester.

Contents

1	Introduction	1
1.1	Problem statement	2
1.2	Outline	2
2	Theory	3
2.1	Aerodynamic resistance	3
2.1.1	Cyclist performance	3
2.1.2	Drag Force	4
2.2	Computational fluid dynamics	4
2.2.1	Turbulence	5
2.2.2	Reynolds-averaged Navier-Stokes	6
2.2.3	Boundary conditions	9
2.2.4	Mesh quality	11
2.3	Softwares	13
2.3.1	OpenFOAM	13
2.3.2	Pointwise	14
2.3.3	Blender	15
2.4	Biomimetic	15
3	Computational setup	17
3.1	Geometry	17
3.2	Meshes	18
3.2.1	OpenFOAM mesh	18
3.2.2	Pointwise mesh	21

3.3	Solver settings	21
3.4	Mesh convergence study	22
3.4.1	OpenFOAM mesh	22
3.4.2	Pointwise mesh	26
3.4.3	Quality of the meshes	29
3.5	Head angles	30
3.5.1	Measurement of head angles	31
4	Evaluation of head angles	33
4.1	Results	33
4.2	Mesh quality	44
5	Biomimetic optimization of helmet	47
5.1	Armadillo	47
5.2	Modification of helmets	48
5.3	Evaluation of modified helmets	48
6	Conclusion	51
6.1	Further-work	52

List of Figures

2.1	Relationship between Reynolds number and drag coefficient together with flow regimes	6
2.2	Relationship between u^+ , y^+ and the three layers	10
2.3	Differences in face center and vector \mathbf{e} , responsible for skewness	12
2.4	Non-orthogonal, the angle between d_{CF} and S_f	13
2.5	Structure of OpenFOAM directory.	14
2.6	Biomimetic design approaches	16
3.1	Comparison of the original and processed geometry	18
3.2	Generated mesh before adjusting parameters	20
3.3	Generated mesh after adjusting parameters	20
3.4	Coarse OpenFOAM mesh	23
3.5	Medium OpenFOAM mesh	24
3.6	Fine OpenFOAM mesh	24
3.7	Residual and force plot for the simulation with OpenFOAM mesh	24
3.8	Pressure distribution over the helmet for simulation with OpenFOAM mesh	25
3.9	y^+ distribution over the helmet for the OpenFOAM mesh	25
3.10	Coarse Pointwise mesh	27
3.11	Medium Pointwise mesh	27
3.12	Fine Pointwise mesh	27
3.13	Residual and force plot for simulation with Pointwise mesh	28
3.14	Pressure distribution over the helmet	28
3.15	y^+ distribution over the helmet with Pointwise mesh	29
3.16	Wind-tunnel test performed at NTNU	30
3.17	Head posture and definition of head landmarks and metrics	31

4.1	Four of the eleven different head angles tested.	33
4.2	Air resistance from simulation and wind tunnel experiments	37
4.3	Streamlines of all angles meshed in OpenFOAM and Pointwise	39
4.4	Pressure-plots of all angles meshed in OpenFOAM and Pointwise	41
4.5	Velocity-plots of all angles	43
5.1	Here it is easy to see the bony plates in the shell of the armadillo	47
5.2	Constraints from helmet at 45 degrees	48
5.3	Three main steps of modifying the geometry	49
5.4	Velocity-plots of the modified helmets	50
5.5	Streamlines of the modified helmets	50
5.6	Pressure-plots of the modified helmets	50

Nomenclature

Abbreviations

CFD	Computational fluid dynamics
RANS	Reynolds-average Navier-Stokes
SST	Shear stress Transport
UiS	University of Stavanger

Greek letters

$(\delta\phi)_f$	Approximated gradient of quantity ϕ
β^*	Constant equal to 0.09
β_1	Constant equal to 0.075
β_2	Constant equal to 0.083
δ_{ij}	Kronecker delta
ε	Dissipation rate of turbulence
γ_1	Constant
γ_2	Constant
κ	von Karmans constant, equal to 0.41
μ	Viscosity
μ_τ	Eddy viscocity
ω	Rate of dissipation of k
ϕ	Constants used in blended functions
ϕ_1	Represent constants one used in blended functions
ϕ_2	Represent constants two used in blended functions
ϕ_N	Quantity in discrete control volume N
ϕ_P	Quantity in discrete control volume P
ρ	Density
$\sigma_{\omega,1}$	Constant equal to 0.5
$\sigma_{\omega,2}$	Constant equal to 0.856
$\sigma_{k,1}$	Constant equal to 0.8
$\sigma_{k,2}$	Constant equal to 1.0
τ	Reynolds stress at wall

τ_w	Shear stress at wall
τ_{ij}	Reynolds shear stress at the surface normal to the x,y or z direction in x, t or z-direction
τ_{xx}	Reynolds normal stress at the surface normal to the x-direction
τ_{xy}	Reynolds shear stress at the surface normal to the x direction in y- direction
τ_{xz}	Reynolds shear stress at the surface normal to the x direction in z- direction
τ_{yx}	Reynolds shear stress at the surface normal to the y direction in x- direction
τ_{yy}	Reynolds normal stress at the surface normal to the y-direction
τ_{yz}	Reynolds shear stress at the surface normal to the y direction in z- direction
τ_{zx}	Reynolds shear stress at the surface normal to the z direction in x- direction
τ_{zy}	Reynolds shear stress at the surface normal to the z direction in y- direction
τ_{zz}	Reynolds normal stress at the surface normal to the z-direction

Skewness

Latin letters

C_μ	Constant equal to 0.09
C_dA	Air resistance
$CD_{k\omega}$	Argument of arg1
F_1	Blending function
u_τ	Friction velocity
∂	Partal derivative
E	Constant equal to 9.8
e	Center-to-center vector
A	Frontal area
a_1	Constant equal to 0.31
C_D	Drag coefficient
C_f	Friction variation
d_{CF}	Center vector
div	Divergence
divU	Volumetric deformation
E	Efficiency
e_{ra}	Expansion rate
f	Face center

f'	Intersection of \mathbf{e} and face cell
F_2	Blended function
F_D	Drag force
grad	Gradient
H	Longest edge
h	Smallest edge
I	Turbulence intensity
k	Turbulent kinetic energy
L	Characteristics length
n_{ce}	Number of layers
P	Mean pressure
p	Pressure
p'	Pressure fluctuation
P_A	Linear aerodynamic resistance
P_{AT}	Aerodynamic resistance
P_{KE}	Kinetic energy
P_{PE}	Potential energy
P_{RR}	Rolling resistance
P_{tot}	Total power
P_{WB}	Wheel bearing friction losses
P_{WR}	Rotational aerodynamic resistance
R	Aspect ratio
Re	Reynolds number
Re	Reynolds number
S	Function presented equation 2.2.2
S_f	Face area vector
S_f	Normal vector of face
U	Mean velocity in x direction
U	Velocity
u	Instantaneous velocity in x direction
u'	Fluctuation velocity in x direction

u^+	Dimensionless velocity
U_∞	Free stream velocity
V	Mean velocity in y direction
v	Instantaneous velocity in y direction
ν	Kinematic viscosity
v'	Fluctuation velocity in y direction
W	Mean velocity in z direction
w	Instantaneous velocity in z direction
w'	Fluctuation velocity in z direction
x	X-direction
y	Y-direction
y^+	Dimensionless velocity
y_{cl}	Height of the cell layer furthest from the wall
y_{dc}	First layer thickness
z	Z-direction

1. Introduction

In 2017, Moviestar cyclist Mikel Landa Meana was one second away his first place at the podium at the overall wins at the worlds biggest bicycle race, Tour de France [1]. Margins after 3540 km at the bicycle seat are minimal, but still, so crucial. Athletes work out multiple hours a day and follow strict diets to be at their best during the highlight of their season. In addition, their equipment should be updated to gain as much advantage as possible. Most of the resistance a cyclist must overcome is caused by aerodynamic resistance. Kyle and Burke [2], referenced in Chowdhurys article [3], states that the aerodynamic drag contributes to 50-90% of the total resistance. Where variance is depended on velocity, body positioning and slope of the road. Imposing extra pressure on the equipments aerodynamic performance.

In the past years, science has become a common tool in cycling, and different studies of aerodynamics has been performed to improve the performance of the cyclists. Zdravkoich looked at how different ratios of the wheel-to-tire diameter affected the vortex shedding [4]. Parker et al. studied the effect of gap on aerodynamic drag formed by the riders leg and two different bicycle frames by wind tunnel experiments [5]. Browline et al. wind tunnel tested more then 200 fabrics at both cylinders and cyclist, in order to find a type of fabric that reduced frictional drag and induced a drag crisis at an earlier stage [6]. Helmets and head postures has been subject of lots of research. Time trial helmets are designed to meet safety standards and are at the same time, designed for speed. Sidelko compared aero helmets with regular road helmets and found that aero helmets had a drag reduction of up to 7.8% more [7]. The tear-drop time trail helmet where first used by Greg LeMond in 1989, and since then, research has been done in order to understand how helmet structures and head positioning influence the air resistance and how to optimize the helmet. Beaumont shows in his research that using a teardrop helmet in a head up position is the most effective. But is crucial when in head down position due to large flow separations and increased drag coefficient [8].

New techniques, adjustments and equipment can be tested by wind tunnel, experiments and computational fluid dynamics (CFD). Many benefits are gained by using CFD, as the software is quick and cheap compared to experiments. However, in order to assure that simulations are correct, a well created mesh must be made. A well made mesh will have a direct influence on how fast, how converged and how accurate the solution is [9].

1.1 *Problem statement*

The objective of this thesis is to study the effect of a cyclist helmet due to aerodynamic forces by the use of computational fluid dynamics. And state if the helmet without a cyclist gives valid results. Two different meshing-software will be used to generate two sets of meshes. These will be compared in order to assure that meshing is valid and to detect differences between the softwares. From the results and previous research, a suggestion for a new model will be given inspired by an armadillo.

1.2 *Outline*

Section 2 is a theory section explaining theories, equations and softwares used during the study. In section 3, computational setup and mesh convergence study for the OpenFOAM and Pointwise mesh are presented. Section 4 evaluate head angles and compare them with previous results. In section 5 a new optimized helmet based on biomimetic is presented. In chapter 6, conclusion and recommendation for further work is presented. In the end references used during the project are presented.

2. Theory

The following section represent relevant theory and governing equation for this thesis. It starts with the resistance experienced by the cyclist, theory about fluid flow, followed by how computational fluid dynamics works together with its governing equations, and what requires for simulations to perform optimal. Then further information about softwares used in the thesis is presented.

2.1 Aerodynamic resistance

2.1.1 Cyclist performance

In order to move the bicycle forward, the cyclist must create enough power to overcome the total resistance. The physics behind the motion of a cyclist is described by Martins validated mathematical model, which describes the power required:

$$P_{tot} = \frac{P_{AT} + P_{RR} + P_{WB} + P_{PE} + P_{KE}}{E} \quad (1)$$

where P_{tot} is the total power input from the rider. This is corrected by an efficiency E , which correct for the transfer from the crank to the wheels by the chain. The total power input accounts for aerodynamic resistance P_{AT} , rolling resistance from the contact between the tire and road P_{RR} , wheel bearing friction losses P_{WB} , potential energy change due to up or down hill riding P_{PE} and kinetic energy changes due to linear and rotational acceleration, P_{KE} [10].

Since aerodynamic resistance is of interest, it will be further investigated. The part can be described by:

$$P_{AT} = P_A + P_{WR} \quad (2)$$

where P_A is the linear component and P_{WR} is the rotational component. The rotational power is accounted for aerodynamic forces, that resist the rotation the wheels. While the linear component is the resistive force acting in the direction of the body. Power required to overcome the linear aerodynamic resistance can be described by the product of drag force and velocity [10]:

$$P_A = F_D U \quad (3)$$

where F_D is the drag force and U is the velocity.

2.1.2 Drag Force

An object is always exposed to pressure. When the object or fluid is stationary, pressure acts like a normal force at every surface. When an object or fluid is moving, there will be a non-uniform pressure force and viscous shear force at the surface of the object, this is known as drag force. The aerodynamic drag force, F_D plays a central role in the aerodynamic resistance. In other words, it describes the relation between the area of the bicycle and cyclist, the air density and the air velocity [11]:

$$F_D = \frac{1}{2} \rho C_D A U^2 \quad (4)$$

where ρ is density, C_D is the coefficient of drag, A is the frontal area and U is air velocity.

From the equations it can be seen that drag force can be reduced by reducing the air resistance drag coefficient C_d or the cross sectional area A . The two of them are closely linked and it is hard to reduce one without the other. Therefore, it is important to find a combination of them two that can be used to improve aerodynamic performance. It is possible to find this force by CFD, but in order to do so, a mathematical description of the fluid flow is needed. Further in the thesis $C_d A$ will be used to define air resistance [11].

2.2 Computational fluid dynamics

CFD, is computer-based simulation and analysis of the system involving fluid flow, heat transfer or other associated situations. The area of application is broad, and vary from aerodynamics to chemical processes, power plants and turbo-machinery [12]. The advantages of using CFD are many. For instance, it can be cheaper and less time consuming than performing an experiment, as a 3D model is sufficient for the simulations and not necessary with a physical model. The CFD software is built by three main elements, a pre-processor, a solver and a post-processor where the different CFD codes are stored. Codes construction are based on the numerical algorithms able to handle fluid flow problems. The pre-processor consist of input of the flow problem. This include definition of computational domain, grid generation, selection of physical phenomena to be modelled, definition of fluid properties and specification of boundary conditions [12]. Solver's mission is to solve the numerical algorithms. The CFD codes consist of discretisation techniques with is a suitable treatment for the key transport

phenomena, convection for transport due to fluid flow and diffusion for the variation of flow variable ϕ , which may be velocity, pressure or temperature from point to point. Then the post-processor present the result from the simulation. Examples might be visualization of the domain and grid and vector plots of wanted solution as velocity, pressure and temperature [12].

2.2.1 Turbulence

It is known for the human eye that the behavior of the fluid change as the velocity change. Scientifically the fluid changes as the Reynolds number changes, which is a dimensionless quantity describing the viscous behavior of all Newtonian fluids calculated from this formula [11]:

$$Re = \frac{U_{\infty} L \rho}{\mu} \quad (5)$$

where U is the free stream velocity, L is the characteristics length and “ μ ” is the fluid viscosity and ρ is the density of air. It is possible to predict if the fluids behavior is turbulent, laminar or in a transition between them, by using the Reynolds number. At low Reynolds number, the fluid flow is laminar. When laminar, layers in the flow will have a smooth behavior and systematically slide past each other. At high Reynolds number, it is observed that flow is turbulent. That means that the flow will behave in a random and chaotic way, in both time and space. The flow might also be in a transition phase, then the behavior will be a combination of these two [12]. The transition from laminar to turbulent begins as the value of the Reynolds number approaches the critical Reynolds number Re_{cr} [13]. For flow along a flat plate, critical Reynolds number is $Re_{cr} = 5 \cdot 10^5$, while along a sphere, it is $Re_{cr} = 4 \cdot 10^5$ [14]. Surface of the geometry is another factor determining the critical Reynolds number. A rougher surface and deformation of the geometry causes reduction in the critical value [15]. In Figure 3.17 the relationship between the Reynolds number and the drag coefficient is plotted together with flow regimes. As the transition from laminar to turbulent boundary layer takes place, drag crisis is seen in the critical flow regime. The turbulent boundary layer remains attached to the surface for a longer distance, and is therefore more resistant to flow separation compared with laminar layers. As the wakes is narrower, the drag is lower [12]. But the low drag does not sustain low. As Reynolds number keep increasing, the drag coefficient increases as well.

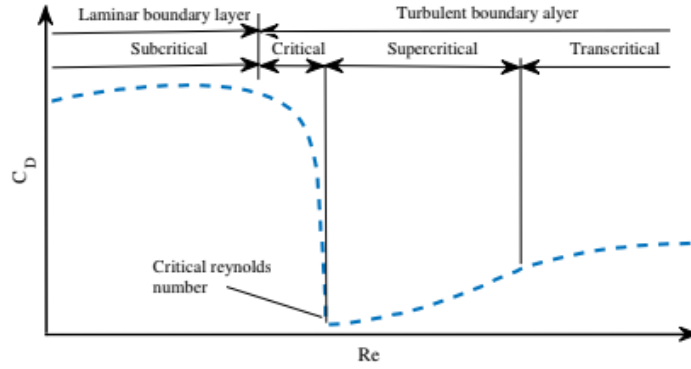


Figure 2.1: Relationship between Reynolds number and drag coefficient together with flow regimes

2.2.2 Reynolds-averaged Navier-Stokes

In engineering applications when turbulent is present, a description of turbulent flow is needed. In this thesis Reynolds-average Navier-Stokes (RANS) equations are used to achieve mathematical equations describing the turbulence. The method express the turbulence as random fluctuations on the mean flow by decomposing flow properties as velocities and pressure to the sum of mean and fluctuating component [12]:

$$u = U + u', v = V + v', w = W + w', p = P + p' \quad (6)$$

where u , v and w is the instantaneous velocity in respectively x , y and z direction, p is the pressure, capital letters denote the mean value and ' values denotes the fluctuation. The conservation laws of mass, momentum and energy establish the mathematical representation of the fluid flow. The equations are complex, both mathematically and physically. But their complexity can be reduce by using hypothesis, idealizations and simplifications appropriately, and then solved numerically. The original Navier-Stokes equations are used to describe the governing equations for an incompressible flow. But in order to take account for the turbulence, RANS equations are introduced which is as

follows: [12].

(7)

$$\text{div}\mathbf{U} = 0 \quad (8)$$

$$\frac{\partial U}{\partial t} + \text{div}(U\mathbf{U}) = -\frac{1}{\rho} \frac{\partial P}{\partial x} + \nu \text{div}(\text{grad}(U)) + \frac{1}{\rho} \left[\frac{\partial}{\partial x}(-\rho \overline{u'^2}) + \frac{\partial}{\partial y}(-\rho \overline{u'v'}) + \frac{\partial}{\partial z}(-\rho \overline{u'w'}) \right] \quad (9)$$

$$\frac{\partial V}{\partial t} + \text{div}(V\mathbf{U}) = -\frac{1}{\rho} \frac{\partial P}{\partial y} + \nu \text{div}(\text{grad}(V)) + \frac{1}{\rho} \left[\frac{\partial}{\partial x}(-\rho \overline{u'v'}) + \frac{\partial}{\partial y}(-\rho \overline{v'^2}) + \frac{\partial}{\partial z}(-\rho \overline{v'w'}) \right] \quad (10)$$

$$\frac{\partial W}{\partial t} + \text{div}(W\mathbf{U}) = -\frac{1}{\rho} \frac{\partial P}{\partial z} + \nu \text{div}(\text{grad}(W)) + \frac{1}{\rho} \left[\frac{\partial}{\partial x}(-\rho \overline{u'w'}) + \frac{\partial}{\partial y}(-\rho \overline{v'w'}) + \frac{\partial}{\partial z}(-\rho \overline{w'^2}) \right] \quad (11)$$

where t is time, ν is the kinematic viscosity and $\overline{u'}$, $\overline{v'}$ and $\overline{w'}$ is the average fluctuating velocity over time in respectively x , y and z direction, div denotes the divergence, and $\text{div } \mathbf{U}$ is the volumetric deformation given by

$$\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} + \frac{\partial W}{\partial z} \quad (12)$$

The time averaged terms, called the Reynolds stresses, must be modelled. They are split into six different stresses, three normal stresses

$$\tau_{xx} = -\rho \overline{u'^2}, \tau_{yy} = -\rho \overline{v'^2}, \tau_{zz} = -\rho \overline{w'^2} \quad (13)$$

and three shear stresses [12].

$$\tau_{xy} = \tau_{yx} = -\rho \overline{u'v'}, \tau_{yz} = \tau_{zy} = -\rho \overline{v'w'}, \tau_{xz} = \tau_{zx} = -\rho \overline{w'x'} \quad (14)$$

A model is required in order to calculate the Reynolds stresses. The Shear Stress Transport (SST) $k-\omega$ turbulence model is suitable for external aerodynamics as the model provides good calculations of both wall bounded flows and separated regions. The model is a hybrid which employs the $k-\omega$ in the boundary layer region close to a surface, and automatically switches to $k-\epsilon$ in free stream and mixing regions region further from the wall [16]. The model calculates Reynolds stresses using the Boussinesq equation, which takes account for the increase in turbulence with the proportional increase in rate of deformation [17].

$$\tau_{ij} = -\rho \overline{u'_i u'_j} = \mu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) - \frac{2}{3} \rho k \delta_{ij} \quad (15)$$

where i, j is denotes x, y or z direction, μ_t is the eddy viscosity, k is turbulence kinetic energy and δ_{ij} is the kronecker delta. The k-omega SST model introduces two additional transport equations, one for the turbulence kinetic energy k

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho u_j k)}{\partial x_j} = \tau_{ij} \frac{\partial u_i}{\partial x_j} - \beta^* \rho k \omega + \frac{\partial}{\partial x_j} \left[(\mu + \mu_t \sigma_k) \frac{\partial k}{\partial x_j} \right] \quad (16)$$

and for the rate of dissipation of the turbulent kinetic energy, ω

$$\frac{\partial(\rho \omega)}{\partial t} + \frac{\partial(\rho u_j \omega)}{\partial x_j} = \frac{\gamma}{\mu_t} \tau_{ij} \frac{\partial u_i}{\partial x_j} - \beta \rho \omega^2 + \frac{\partial}{\partial x_j} \left[(\mu + \mu_t \sigma_{\omega,1}) \frac{\partial \omega}{\partial x_j} \right] + 2\rho(1 - F_1) \sigma_{\omega,2} \frac{1}{\omega} \frac{\partial k}{\partial x} \quad (17)$$

where

$$F_1 = \tanh \left(\left(\min \left[\max \left(\frac{\sqrt{k}}{\beta^* \omega y}, \frac{500\mu}{\rho y^2 \omega} \right), \frac{4\rho \sigma_{\omega,2} k}{CD_{k\omega} y^2} \right] \right)^4 \right) \quad (18)$$

$$CD_{k\omega} = \max \left(2\rho \sigma_{\omega,2} \frac{1}{\omega} \frac{\partial k}{\partial x_i} \frac{\partial \omega}{\partial x_i}, 10^{-20} \right) \quad (19)$$

$$F_2 = \tanh \left[\max \left(2 \frac{\sqrt{k}}{\beta^* \omega y}, \frac{500\mu}{\rho y^2 \omega} \right) \right]^2 \quad (20)$$

$$\mu_t = \frac{\rho a_1 k}{\max(a_1 \omega, SF_2)} \quad (21)$$

$$S = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \quad (22)$$

This is a blended function saying that near the wall $F_1=1$ and the k- ω model is used, and further from the wall, $F_1=0$ the k- ϵ model is used. The constants are also blended.

$$\phi = F_1 \phi_1 + (1 - F_1) \phi_2 \quad (23)$$

meaning that ϕ_1 represent constant one and ϕ_2 represent constant 2. The constants are [17]:

$$\begin{aligned} \sigma_{k1} = 0.8 \quad \sigma_{\omega1} = 0.5 \quad \beta_1 = 0.075 \quad \gamma_1 = \frac{\beta_1}{\beta^*} - \frac{\sigma_{\omega1} \kappa^2}{\sqrt{\beta^*}} \quad a_1 = 0.31 \quad \kappa = 0.41 \\ \sigma_{k2} = 1.0 \quad \sigma_{\omega2} = 0.856 \quad \beta_2 = 0.083 \quad \gamma_2 = \frac{\beta_2}{\beta^*} - \frac{\sigma_{\omega2} \kappa^2}{\sqrt{\beta^*}} \quad \beta^* = 0.09 \end{aligned}$$

2.2.3 Boundary conditions

In the region between the wall and the free steam, a near wall region must be introduced. This is a transitional region called the boundary layer, where velocity will reach the free stream velocity. Here effect of viscosity are extremely important [12]. The boundary layer is also highly dependent on whereas the flow within is laminar or turbulence. At laminar boundary layers, momentum is transferred between the layers in a microscopic level, while in the turbulent layer the mixing happens over a larger scale. The flow behaviour and turbulence structure are different close to a wall compared to a free turbulent flow. Due to friction and damping of the turbulent velocity fluctuations, the fluid is stationary at the surface of the object, known as the no-slip condition [12]. Sine there is no velocity, the Reynolds number is zero. The flow is only influenced by viscous effects and shear stress at the wall, and is not dependent on free stream parameters. The mean velocity can be described by:

$$U = f(y, \rho, \mu, \tau_w) \quad (24)$$

where y is distance of the wall, ρ is density, μ is viscosity and τ_w is the wall shear stress. Dimensional analysis close to the wall gives a formula called "the law of the wall" [12]

$$u^+ = \frac{U}{u_\tau} = f\left(\frac{\rho u_\tau y}{\mu}\right) = f(y^+) \quad (25)$$

where u^+ is the dimensionless velocity, y^+ is the dimensionless distance from the wall, u_τ is the friction velocity given by

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \quad (26)$$

The region close to the wall is split into three layers in order to properly capture the effect of boundary layers.

- viscous sub-layer. $0 \leq y^+ < 5$. where

$$u^+ = y^+ \quad (27)$$

- buffer layer, $5 \leq y^+ < 30$

- log-law layer. $30 \leq y^+ < 500$ where

$$u^+ = \frac{1}{\kappa} \ln(\mathbf{E}y^+) \quad (28)$$

where $E \approx 9.8$ is a constant. In the viscous sub-layer the flow is assumed to be laminar no matter of the outside flow. Further away the turbulent fluctuations dominate, and the log-law layer function is used to describe the flow. In the buffer layer, the magnitude of viscous and turbulence stresses are approximately equal, making it to a complex velocity profile. To simplify it, the buffer layer is split into two parts, using the linear relation as viscous sub-layer in one part and the logarithmic function in the other. In Figure 3.17, the characteristics of u^+ as a function of y^+ is plotted and the different layers is stated [18] [19].

The SST $k-\omega$ model describe the behaviour of ω in the viscous sub-layer and the logarithmic region [19]. The model switch between the viscous and logarithmic region according to the value of y^+ . To obtain the outcome of the law of the wall requires an extremely fine mesh along the wall. This is obtained by using a viscous mesh which define layers along the surface. Thickness of the first layer may be calculated by:

$$y_{dc} = \frac{y^+ \mu}{u_\tau \rho} \quad (29)$$

where in this case,

$$u_\tau = \sqrt{\frac{\tau_\omega}{\rho}} \quad (30)$$

where τ_ω along the wall is given by:

$$\tau_\omega = \frac{C_f \rho U_\infty^2}{2} \quad (31)$$

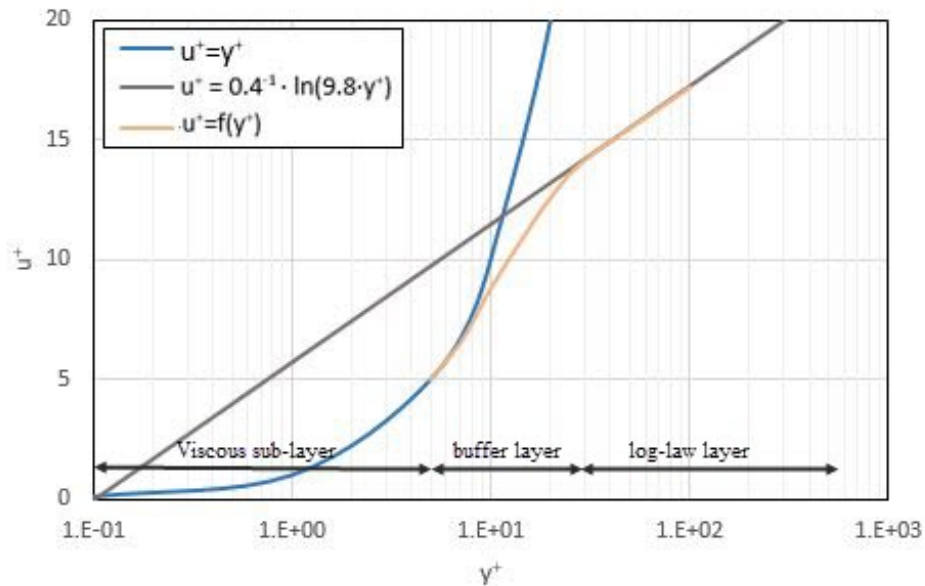


Figure 2.2: Relationship between u^+ , y^+ and the three layers

and C_f is the friction variation

$$C_f = \frac{0.027}{Re^{\frac{1}{7}}} \quad (32)$$

By adding these equations into each other, a final expression for the height of the first layer, y , can be presented.

$$y = \frac{1}{\sqrt{0.0135}} \frac{y^+ Re^{\frac{1}{4}} \mu}{U_\infty \rho} \quad (33)$$

In addition to the first layer thickness, other parameters must be defined. The expansion ratio e_{ra} which defines the rate of increase in height for the following layers and number of cell layers in the viscous mesh n_{ce} . This number should be sufficient in order to make an appropriate transition of the layer furthest from the wall to the background mesh. This can be calculated by:

$$n_{ce} = \frac{\ln y_{cl} - \ln y_{dc}}{\ln e_{ra}} \quad (34)$$

where y_{cl} is the height of the cells furthest from the wall in the viscous mesh, which is equal to the size of the background cells to ensure a smooth transition. With the exception of cases where n_{ce} is a non-integer number, then the value should be rounded downwards leaving y_{cl} a bit smaller than the cells in the background mesh.

To initiate values, k and ω must be specified at the inlet. Turbulent kinetic energy can be estimated by:

$$k = \frac{3}{2} (IU_\infty)^2 \quad (35)$$

where I is the turbulence intensity. Turbulence specific dissipation rate can be estimated by:

$$\omega = \frac{\sqrt{k}}{C_\mu L} \quad (36)$$

where $C_\mu = 0.09$ and where L is the reference length [20].

2.2.4 Mesh quality

The solution of a flow element, such as velocity, temperature, pressure etc, is defined as nodes inside each cell. This means that a larger number of cells leads to a more accurate solution. In order to

have a optimal mesh, the mesh should be finer in areas of variation in element [12]. Packing grids closely in regions where the flow quickly change, and using a more coarse distribution where the flow is homogeneous is a good combination in order to find the balance between needs of accuracy and computation time [21].

A fine mesh will often lead to a good quality of the mesh. Good quality can be measured by some main metrics; skewness, non-orthogonality, smoothness and aspect ratio.

Skewness is defined as deviation between the location of the face centers f , and f' which is where the center-to-center vector \mathbf{e} meets the face as illustrated in Figure 2.3. Skewness can affect the computational of flux between the cells and cause numerical diffusion [22].

$$\varepsilon = \frac{|f - f'|}{|\mathbf{e}|} \quad (37)$$

Value of skewness should be kept as close to zero as possible [23] [22].

Another threat that might cause numerical diffusion is non-orthogonality. Non-orthogonality is the angle between two cell-center vector \mathbf{d} and the normal- vector of the face S_f as seen in Figure 2.4 [24]. Normally the gradient at cell face are calculated by the differences between the neighboring cells divided by the distance between them. When the computed term require the product of the gradient with the face areas as shown in Equation 38, errors might occur if the normal face vector and cell-center vector are not aligned [22].

$$S_f \cdot (\Delta\phi)_f = |S_f| \frac{\phi_N - \phi_P}{|d_{CF}|} \quad (38)$$

where S_f is the face area vector and the $(\Delta\phi)$ is the approximated gradient Where ϕ denotes the quantity in discrete control volumes [23]. It is recommended to keep the angle as small as possible to

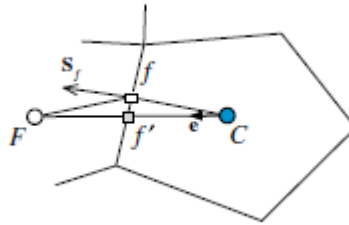


Figure 2.3: Differences in face center and vector \mathbf{e} , responsible for skewness

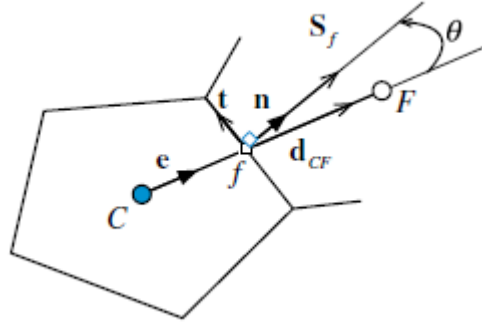


Figure 2.4: Non-orthogonal, the angle between d_{CF} and S_f

avoid numerical errors [22].

The efficiency and accuracy will also be affected by the aspect ratio, which measure the stretching of a cell [25]. It is calculated as the ratio between the cells maximum and minimum width.

$$R = \frac{H}{h} \quad (39)$$

where H is the longest edge and h is the shortest edge. The efficiency and accuracy will decrease as the aspects ratio increase. It will also lead to interpolation errors causing numerical diffusion [26]. In OpenFOAM the ratio should be kept as close as possible to one near surfaces [27].

Another thing to be aware of is sudden jumps in cell sizes. The large differences in cell volume will cost larger truncation errors which is the distinction between partial derivatives in the governing equations and their discrete approximations [25].

2.3 Softwares

Different softwares has been used in order to perform simulations, create meshes and modify the geometry.

2.3.1 OpenFOAM

OpenFOAM is a free, open source CFD software. It offers a wise range of feature able to solve complex fluid flows [28]. In addition it provides other important applications required for a full continuing mechanics modeling, like meshing utilities and the post-processing tool ParaView [29].

The OpenFOAM case dictionary consist of three main directories as seen in Figure 2.5. The constant

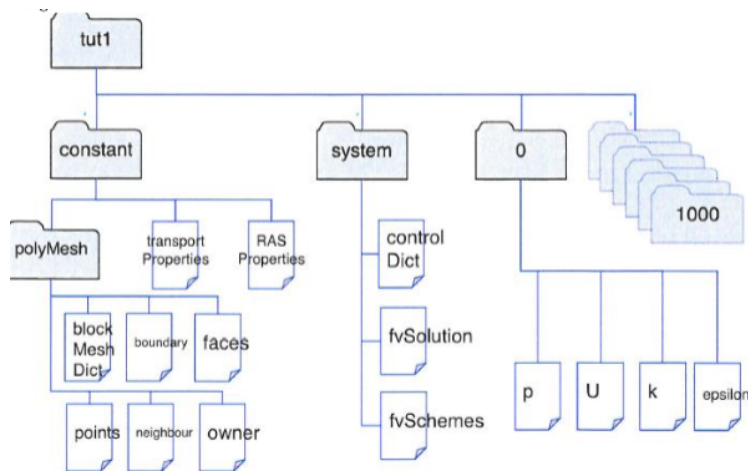


Figure 2.5: Structure of OpenFOAM directory.

folder include dictionaries with information about mesh, material properties, turbulence properties and etc. The system folder define number of iterations, time step size and solution control, and the 0 folder gives information about the initial flow fields and boundary conditions [30].

OpenFOAM has two different utilities for meshing. BlockMesh is the simplest one, which create a structured multiblock mesh after input information like point coordinates and limits for boundaries. The other one is snappyHexMesh, which is well suited for special cases of geometry. It creates unstructured mesh out of an already existing base mesh, eg. blockMesh. Cells are splitted before snapped around surfaces [27]. To avoid irregular cells along the boundary surface, layers created from hexahedral cells are aligned to the surface.

2.3.2 Pointwise

Pointwise is a software which generate meshes for computational fluid dynamics. It offers both structured and unstructured grid generation, together with their T-Rex technique which generates boundary layers in a point-by-point progress [9]. The software is able to import the model as an stl or cad file, and is well suited for cases containing complex geometries with viscous flows. With defined boundary conditions the mesh can be exported in the solvers CFD format [31].

2.3.3 *Blender*

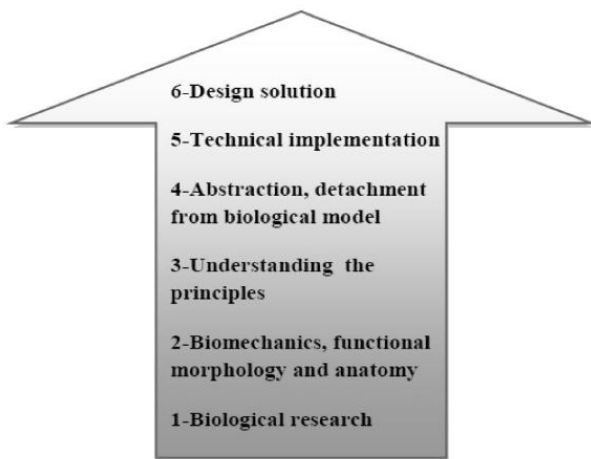
Blender is an open source for 3D creation. Usage for the software is mainly for animation of films or video games. This includes creating and modifying 3D models [32]. There is several categories of modifiers and utilities which blender serve is, but the most interesting is [33];

- Generate - Constructive tools changing the general appearance or add new geometry to an object.
- Sculpt mode - Add mass and smooth edges where wanted.
- Deform - Change the shape of an object without adding new geometry.
- Inverse Kinematics - Re-positions a whole chain length with movement of one bone.

Kinematics is the mechanics describing the motion of points, objects and groups of objects without reference to the cause of motions. Reaching an end position with forward kinematics can be calculated if starting position, length of chain and angles between bones are known. In inverse kinematic, the end position is the given data and configuration of the chain must be found in order to reach the new point [34]. Instead of starting with movement of the root bone and re-positioning of child bones, the last bone is positioned causing automatically positioning of its family bones [33].

2.4 *Biomimetic*

The study of mimicking nature and implement it in human problems is called biomimetic, or biomimicry. Nature has spent 3.85 billions years for evolution and optimization, making it the most experienced problem solver and a huge source for inspiration and innovation. Different professions have searched answers from nature, both architects and engineers have mimicked forms to create structures that are better and more efficient. The design process of biomimicry normally falls down into to categories: Defining human needs and seen how organisms or ecosystem solved theirs, or identifying characteristics in an organism and transfer it to a design as illustrated in Figure 2.6 [35].



(a) *Bottom-up mimicry*



(b) *Top-down mimicry*

Figure 2.6: *Biomimetic design approaches*

3. Computational setup

Computational Fluid Dynamic is used to create realistic simulations for studying the aerodynamic performance of helmets. Investigating drag using CFD, high-quality cells are essential. This may be challenging for some softwares when dealing with complex geometry. Mesh generation is crucial for how fast, how converged and how accurate simulations are, and is where the operator has the most influence in the simulation [36]. Computational setup for meshes created with Pointwise and integrated utilities in OpenFOAM will be performed in this chapter.

Helmet is the only object considered and tested in the simulations. This makes it possible to state differences caused by helmet shape without other disturbances, such as change in body position. In addition, computational time is reduced as the simulated object is smaller compared to the whole body of a cyclist. However, some inaccuracies might be present, since the cyclist is absent. Pressure gaps and velocity wakes in the transition from helmet to back of the cyclist which may vary with the head angle is not accounted for.

3.1 Geometry

Working with faceted geometries, such as stl files, in Pointwise require some processing in order to define the right topology. Faceted geometries consist of different shells made of groups of polygons defining the surfaces. Utilities like "on Database Entities" creates a domain on the selected geometry defining required surfaces and connectors [37]. The original helmet file consisted of two different shells, the helmet and a visor. In addition a sphere illustrating a head shaped figure had been added to the geometry, since this was also present in the experimental setup. In Pointwise, creating a proper mesh around the three shells is harder, or impossible, compared to meshing around the three peaces as one shell. This led to some modification in order to import the geometry into Pointwise as one shell. With utilities in blender, a new volume covering the old part were created. A plane adjusted with the "loop cut" command created a plane more like the shape of the helmet. The plane were extruded so that the helmet was captured inside the volume. "Shrimp wrap" command fitted the new geometry onto the nearest surface, creating the new geometry in figure 3.1b. The shape is just like the original, but with a more coarse surface. To properly compare the results, same geometry will be used in the OpenFOAM meshing. SnappyHexMesh utility in OpenFOAM is known for struggling with complex

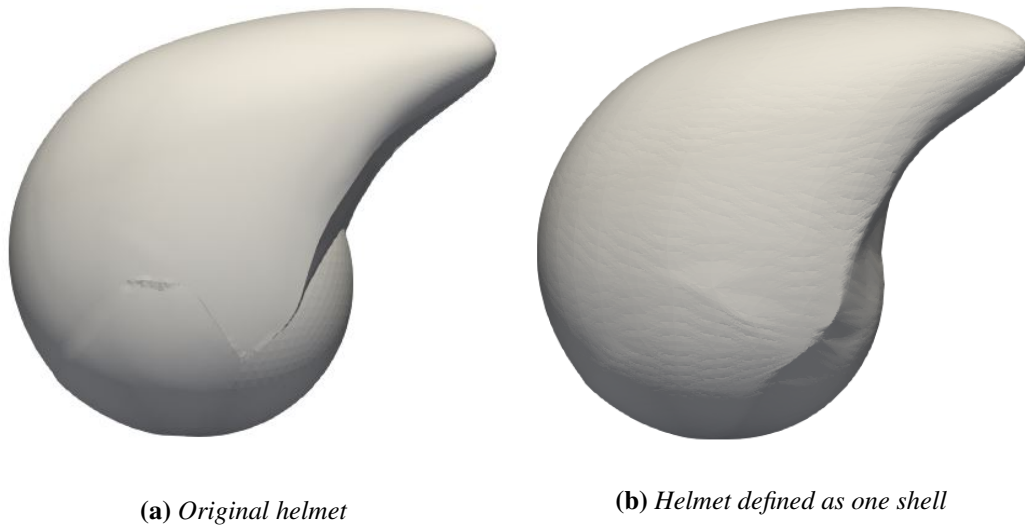


Figure 3.1: *Comparison of the original and processed geometry*

geometries, as refined mesh and proper layer generation are not as robust. Quality lacks as skewness and non-orthogonality would have a higher change of being present. The solver might be able to run with mesh quality errors, but may lead to decrease accuracy and increase in computational time for the solver [38]. Using the new geometry, will therefore benefit the mesh generation as some sharp edges under the helmet has flattened during the renewal.

3.2 Meshes

Process of mesh generation will be described in this section.

3.2.1 *OpenFOAM* mesh

Mesh generation can be done directly in OpenFOAM by using the two utilities `blockMesh` and `snappyHexMesh`. First computational domain is defined by `blockMesh`, which create a background mesh consisting purely of hexahedral cells filling the region. Then `snappyHexMesh` refines the existing mesh by splitting the hex mesh into smaller cells. Cells inside the geometry are removed and surrounding cells snaps onto the surface. Creating a grid fine enough close to the surface of the geometry, might be a challenge. To assure that drag resistance has proper values, boundary layers are created. Settings are defined in the `addLayerControls` utility inside the dictionary of `snappyHexMesh`. Some of the settings for the boundary layer, as first layer height and number of layers, are defined by equations

33 and 34. Other settings must be treated as parameters in order to obtain the best possible result [26].

With base in an already existing `snappyHexMeshdict`, where most of the parameters were recommendations from CFD direct [27], only small adjustments were required to create a sufficient mesh. The most important are the calculated values for first layer thickness and number of layers. The formulas are presented in the theory section, where equation 33 describes first layer height and equation 34 describes the number of layers. Reynolds number is calculated from equation 5, and is

$$Re = \frac{U_\infty L \rho}{\mu} = \frac{13.9 \cdot 0.32 \cdot 1.225}{1.81 \cdot 10^{-5}} = 302 \cdot 10^3 \quad (40)$$

Where the constants are: Length of the geometry, $L = 0.32m$, viscosity of air $\mu = 1.802 \cdot 10^{-5} \frac{kg}{ms}$, density of air $\rho = 1.225 \frac{kg}{m^3}$ and velocity $13.9 \frac{m}{s}$ [39].

Choosing a y^+ value at 1, which is recommended for turbulence [40] the first layer thickness is:

$$y_{dc} = \frac{1}{\sqrt{0.0135}} \frac{y^+ Re^{\frac{1}{4}} \mu}{U_\infty \rho} = y = \frac{1}{\sqrt{0.0135}} \frac{1 \cdot (302 \cdot 10^3)^{\frac{1}{4}} 1.805 \cdot 10^{-5}}{13.9 \cdot 1.225} = 2.253 \cdot 10^{-5} \quad (41)$$

In order to calculate the number of layers required, size of the surrounding cells must be specified. From Table 1, it is seen that a number of cells at 2,74 million are sufficient for further generation. This correlates to a grid-size at $2.679 \cdot 10^{-3}$ next to the geometry. Employing this and the first layer height into layers equation together with an expansion ratio of 1.2, number of layers for that specific mesh were defined.

$$n_{ce} = \frac{\ln y_{cl} - \ln y_{dc}}{\ln e_{ra}} = \frac{\ln 2.679 \cdot 10^{-3} - \ln 2.253 \cdot 10^{-5}}{\ln 1.2} \approx 26 \quad (42)$$

Other parameters controlling the mesh were adjusted and tested in order to create well generated layers. The most interesting where the parameters controlling the meshes ability to snap to surface and add layers. The parameters and their sub-dictionaries are as followed [27]:

snapControls

- `nSolveIter` - number of mesh displacement relaxation iterations
- `nRelaxIter` - maximum number of snapping relaxation iterations

addLayersControls

- nRelaxIter - number of snapping relaxation
- nLayerIter - overall max number of layer addition iterations

All listed parameters controls the number of iterations for mesh-generation. In the snapControl sub-dictionary, the nSolveIter value was increased to 100, and nRelaxIter increased to 50. In the addLayerControls dictionary, nRelaxIter and nLayerIter were increased to 100 and 50 respectively. The changes led to an decrease of the average y^+ from 7 to ≈ 1 . Figure 3.2 and 3.3 demonstrate the differences before and after the refinement respectively.

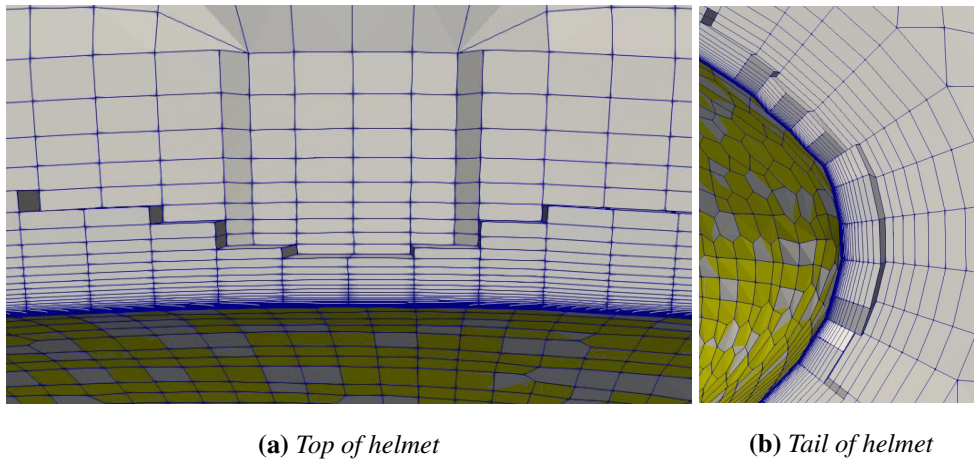


Figure 3.2: *Generated mesh before adjusting parameters*

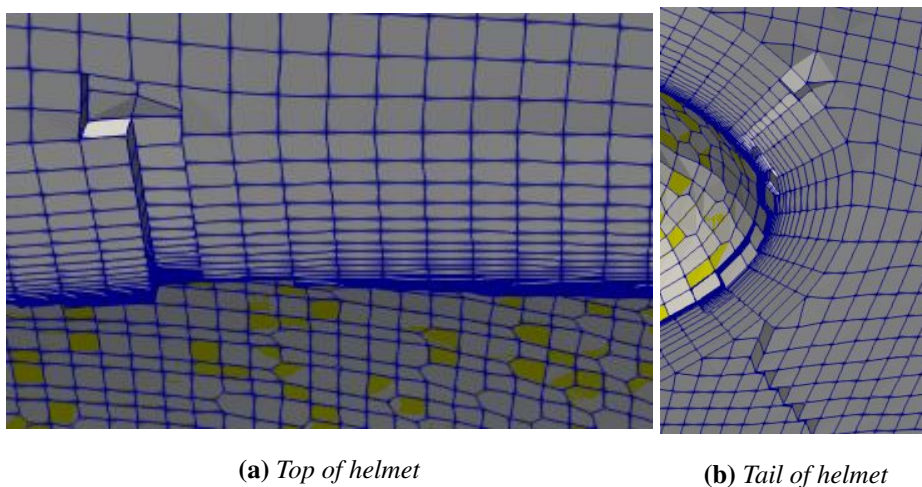


Figure 3.3: *Generated mesh after adjusting parameters*

3.2.2 Pointwise mesh

Mesh may be generated in an external software and exported to OpenFOAM for simulations. As explained in theory section, Pointwise is such a meshing software. Their hybrid meshing method is suited for this case, it is a rapid method for meshing of complex geometries with boundary layer resolution [41]. Domain is built by an unstructured mesh consisting of tetrahedrons and T-Rex is used to generate layers. Calculated value for first layer thickness is the same as in Equation 41 from Section 3.2.1, respectively $2.253e-5$ m. Since T-Rex automatically control the layer generation, calculation of number of layers is not necessary. In addition T-Rex automatically controls the layer expansion in sharp corners and a smooth transitions into the far field [21]. Since the transition to the far field might be rapid, a small, finer far field is defined behind the helmet in order to obtain better resolution of the wake.

3.3 Solver settings

In openFOAM there is no generic solver which may be appropriate to all cases, therefor a suited solver must be chosen. Dealing with an incompressible, turbulent flow, the SIMPLE (Semi-Implicit Method for Pressure Linked Equations) algorithm is a suitable choice to use for a steady-state simulation [42]. Governing equations together with the SST $k-\omega$ from Section 2.2.2 contributes to the solver. Inlet of the domain is imposed with a uniform horizontal velocity at 13.9m/s and zero gradient pressure. The outlets conditions are controlled by the surroundings and are imposed with ambient static pressure and mean velocity from internal field. Sides of the domain are imposed with a zero gradient for pressure and no slip for velocity. Helmet is imposed with slip for velocity and zero gradient for pressure.

The initial value required for the turbulence kinetic energy is estimated by:

$$k = \frac{3}{2}(IU_{\infty})^2 = \frac{3}{2}(0.005 \cdot 13.9)^2 = 7.245 \cdot 10^{-3} \quad (43)$$

and initial value for the turbulent specific dissipation rate is:

$$\omega = \frac{\sqrt{k}}{C_{\mu}L} = \frac{\sqrt{7.245 \cdot 10^{-3}}}{0.09 \cdot 0.32} = 2.955 \quad (44)$$

where L is the length of the helmet.

3.4 Mesh convergence study

To perform a mesh convergence study, some simple steps presented by Leap CFD team is followed in order to create a valid result [43].

1. Run the simulation on the initial mesh and ensure that residuals has convergence to an error at 10^{-4} . If not, refine the mesh and repeat the simulation.
2. Refine the whole meshes domain. If the monitored point varies from step 1, it means that step one was not accurate enough to capture the result. As visualized in table.
3. Repeat the refinement of mesh until a mesh independent solution is reached.

In addition to check the convergence of the residuals, the y^+ value must be controlled to assure a sufficient value.

The objective of this grid sensitive study is to find a combination of mesh and air resistance that don't require to long computational time, but still gives an reliable air resistance. Three main grids is created and compared in order to find an appropriate mesh in both cases.

Flipping equation 4, formula for air resistance, C_dA is obtained:

$$C_dA = \frac{2F_D}{\rho v^2} \quad (45)$$

Forces are gained from simulations and constants are already known. Drag coefficient is decisive in order to decide which of the meshes to keep.

3.4.1 OpenFOAM mesh

A coarser or finer grid can be created different ways. One option is to adjust the refinement parameters together with boundary layers in snappyHexMesh. But, as mentioned above in step 2 describing a mesh study, the whole domain should be refined. Changing the number of cells in each direction in blockMesh will lead to refinement of the whole mesh. This will impact the size of the grid surrounding the geometry, and require adjustment of number of layers for each case. The smaller the grid size, the fewer layers are required. From Table 1, it is seen that finer mesh provide less is the variation in drag. However, refinement of the mesh will have an impact on the computational time. From Table 1, it is

seen that air resistance varies a lot between coarse and medium mesh. While between medium and fine mesh values are equal. This makes the medium mesh the best suited one, due to lower amount of cells which correspond to lower computational time. The three meshes are illustrated in Figures 3.4, 3.5 and 3.6.

Table 1: Mesh properties for convergence test of OpenFOAM mesh

Mesh	points	$C_d A$ [m ²]	y^+
Coarse	1'032'010	0.0186	0.655
Medium	2'738'244	0.0159	0.614
Fine	5'580'052	0.0159	0.613

The difference in amount of cells are shown in Figures 3.4a, 3.5a and 3.6a which illustrates the domains. Less spacing is seen between cells in the fine mesh in Figure 3.6a, and more spacing between the cells in the course mesh in Figure 3.4a. Figure 3.4b illustrate the how poorly the layers transitions into the far field in the course mesh. While the medium and fine mesh in Figures 3.5b and 3.6b, layers are complete and transition smoother to into the surrounding grid. A small section from all the latter three figures, marked with a black box, are expanded in order to see how the layers have generated from the surface.

Figure 3.7a shows the residual plotted against the iterations for the medium mesh presented in Table 1 and shown in Figure 3.5. Here it is seen that after 6000 iterations, the residuals has a value at approximately 10^{-4} . With such a low source of error, forces will be stable and give valid results. This is seen in Figure 3.7b, where the force has stabilized just below 2N. Figure 3.8 illustrates the pressure distribution over the surface.

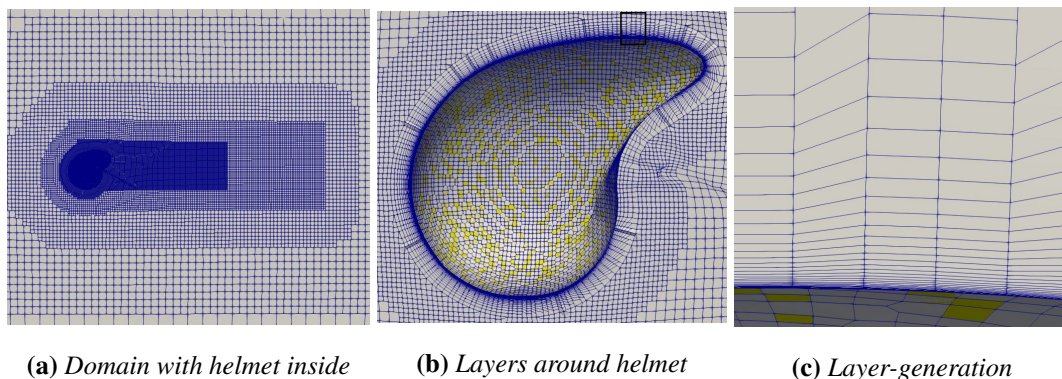


Figure 3.4: Coarse OpenFOAM mesh

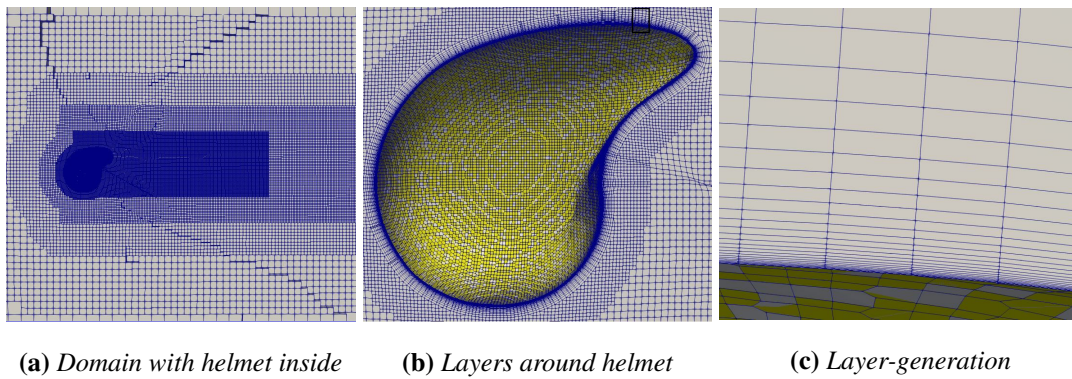


Figure 3.5: Medium OpenFOAM mesh

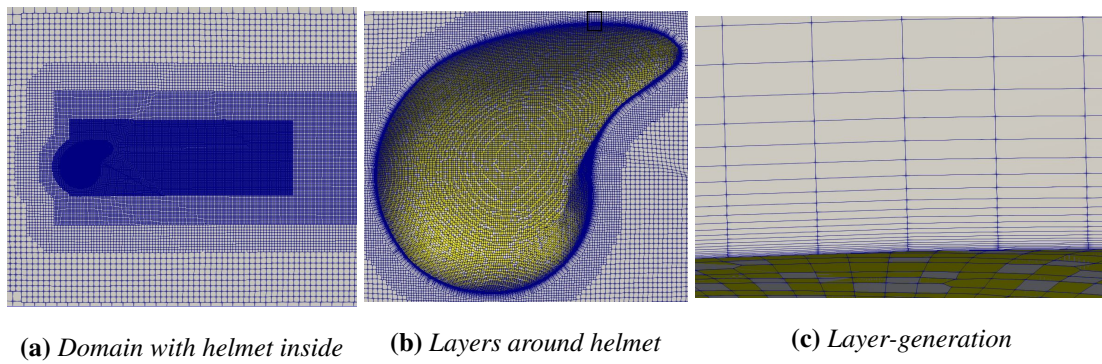


Figure 3.6: Fine OpenFOAM mesh

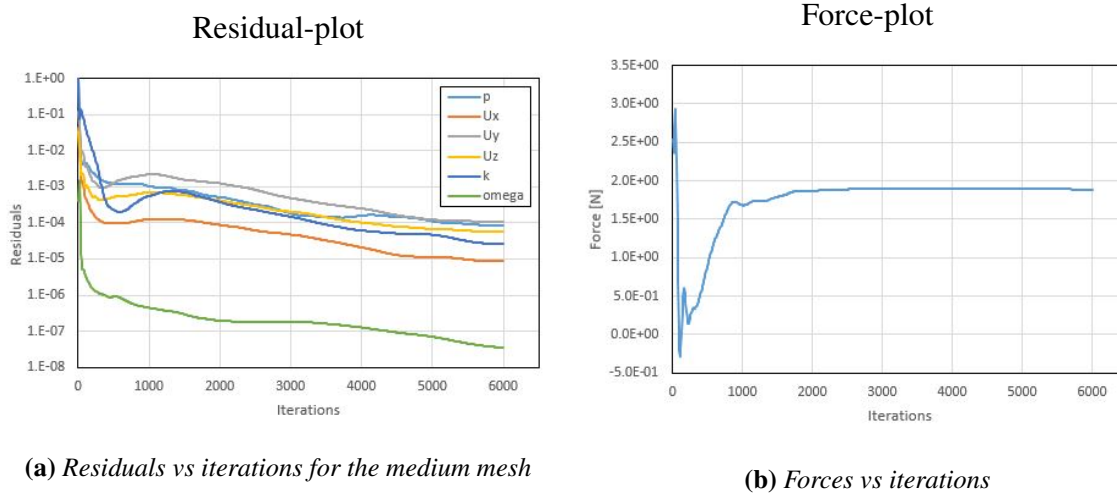
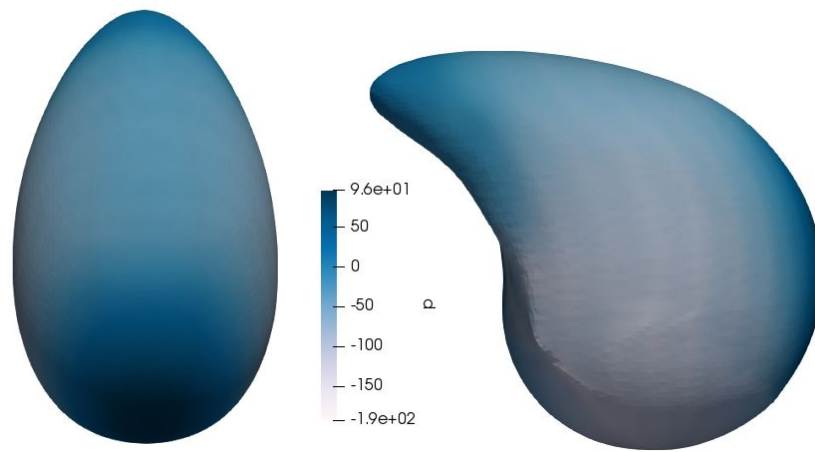


Figure 3.7: Residual and force plot for the simulation with OpenFOAM mesh

Table 1 indicates the average y^+ value for the three meshes. In Figure 3.9 the distribution of y^+ over the surface of the medium mesh is illustrated. The scale indicates that maximum value present is 1.1, which is located at the bottom edge of the visor.



(a) *Pressure distribution seen from top*

(b) *Pressure distribution seen from side*

Figure 3.8: *Pressure distribution over the helmet for simulation with OpenFOAM mesh*

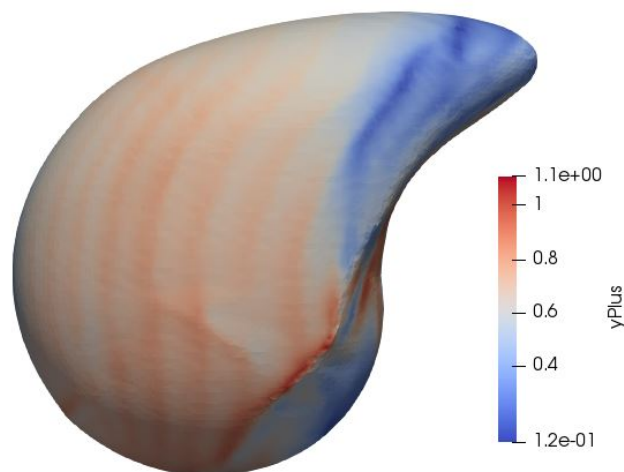


Figure 3.9: *y⁺ distribution over the helmet for the OpenFOAM mesh*

3.4.2 Pointwise mesh

Same procedure presented by Leap CFD is followed in this grid convergence study in Pointwise. Both residuals and y^+ value are inspected in order to assure that the initial grid is sufficient, before the domain is adjusted. Values for air resistance are observed in order to find a mesh with suited combination of accuracy and computational time. Three meshes are created and compared.

Finer or coarser grid in Pointwise are created by adjusting number of grid points positioned along domains connectors. More points create a finer grid, and less a coarser grid. Making an extra-coarse grid is almost impossible in Pointwise, as seen from Figure 3.10a. Most of the cells are generated close to the helmet due to layer generation or refined area, and the surrounding domain is extremely coarse. Figures 3.11a and 3.12a clearer illustrates the background domain and the refined area behind the helmet. Layers around the helmets are clearer illustrated in Figures 3.10b 3.11b, 3.12b. In all three figures, a smooth transition into the far field is present. From the figures of the helmet, a small section denoted with a black box is extracted, giving a clearer illustration of the generated layer seen in Figures 3.10c, 3.11c and 3.12c. From Table 2, it is seen that air resistance varies between coarse and medium mesh. In the medium and fine meshes, values are the same. Therefor the medium mesh is chosen, due to the lower amount of cells.

Figure 3.13a shows that after 6000 iterations the highest residuals has a value below 10^{-4} , and has converged nicely. This is reflected in the force result from Figure 3.13b, where forces have approximately the same value during the last 4000 iterations. Pressure plot of the helmet is shown in Figure 3.14. In addition to Table 2 where the average y^+ value is indicated, the y^+ values for the medium mesh is distributed over the helmets surface, illustrated in Figure 3.15. The scale indicates the maximum value at 1, which is located at the visors bottom edge, same place as in the OpenFOAM mesh in Figure 3.9.

Table 2: Mesh convergence, Pointwise

Mesh	cells	D_dA [m ²]	y^+
Coarse	507'516	0.0160	0.62
Medium	615'903	0.0157	0.62
Fine	1'168'025	0.0157	0.62

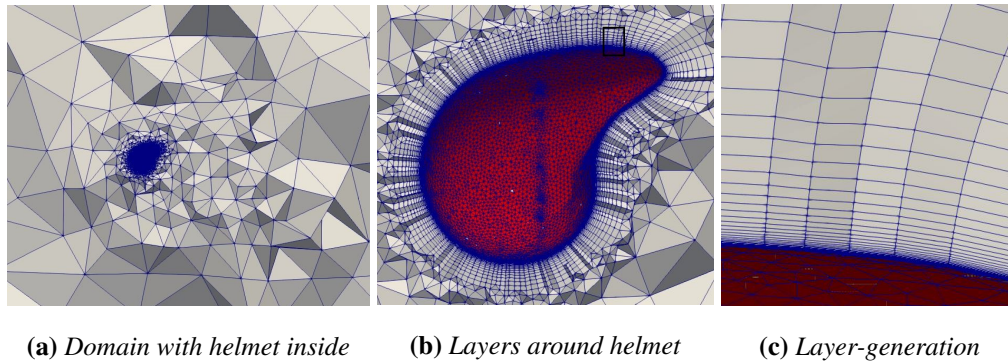


Figure 3.10: *Coarse Pointwise mesh*

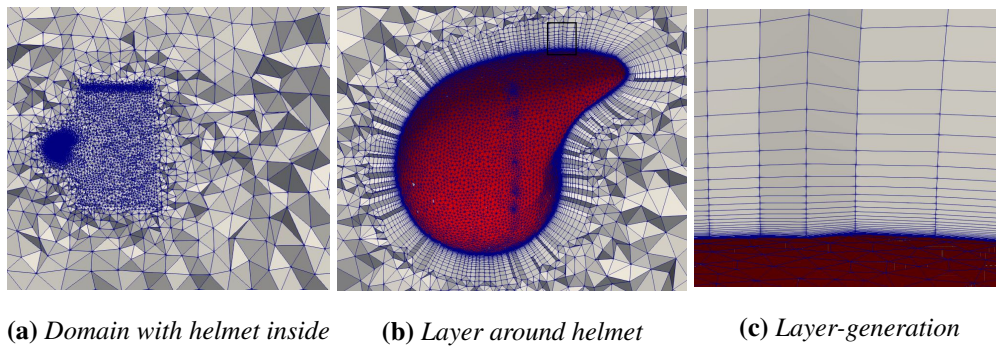


Figure 3.11: *Medium Pointwise mesh*

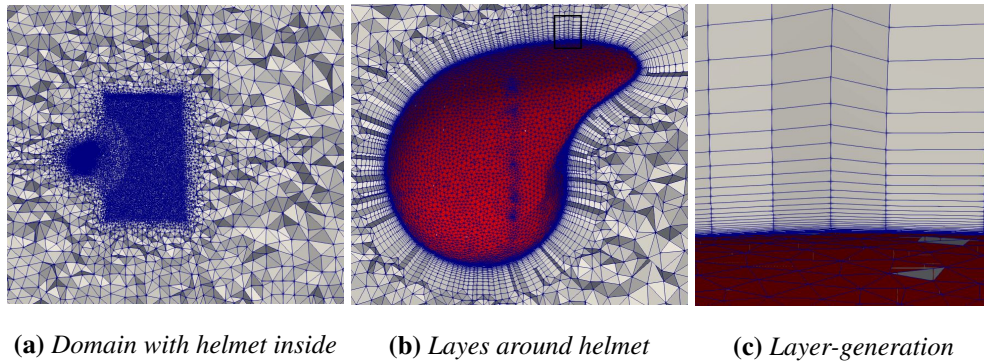
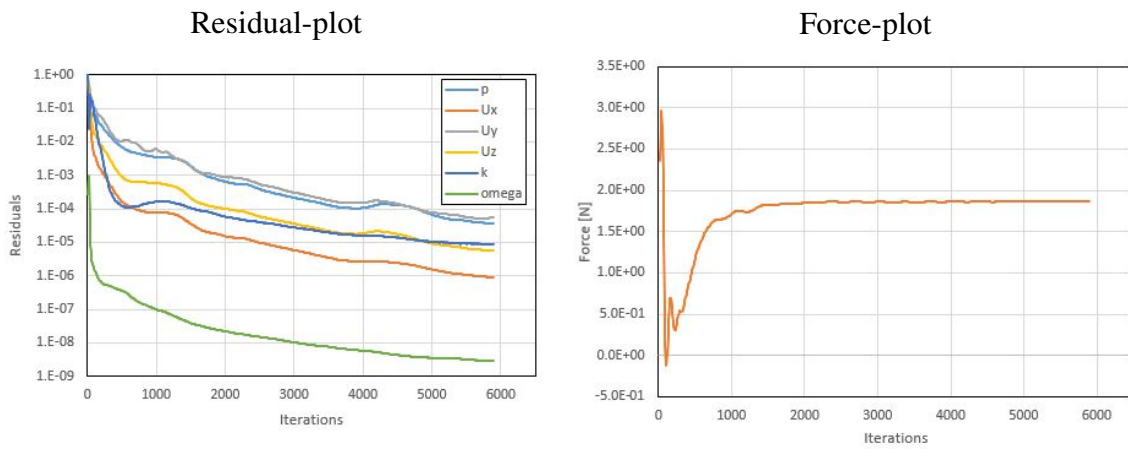


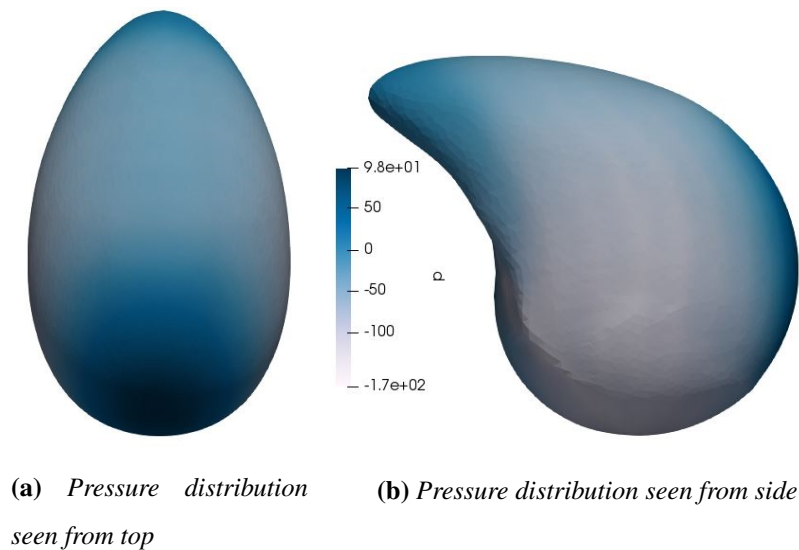
Figure 3.12: *Fine Pointwise mesh*



(a) Residuals vs iterations for the medium mesh

(b) Forces vs iterations

Figure 3.13: Residual and force plot for simulation with Pointwise mesh



(a) Pressure distribution seen from top

(b) Pressure distribution seen from side

Figure 3.14: Pressure distribution over the helmet

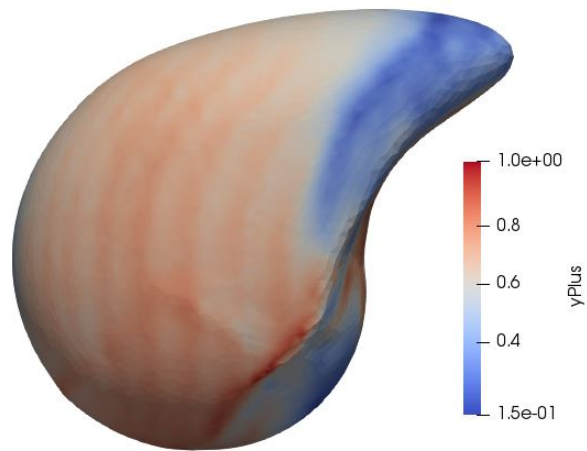


Figure 3.15: y^+ distribution over the helmet with Pointwise mesh

3.4.3 Quality of the meshes

CheckMesh command in OpenFOAM checks quality of the mesh and for topological errors. Simulation may run with quality errors in the mesh, but these errors might effect the solution by either tamper the accuracy or slow the solver. For the meshes created in Pointwise, quality can be checked in Pointwise before exported, or in OpenFOAM after it is exported. A quality check in OpenFOAM will ensure that nothing happened to the mesh during the exporting. The highest values non-ortagonality, skewness and aspect ratio are presented in Tables 3.4.3 and 3.4.3.

- Non-orthogonality faces $> 70 = 13$
- Max non-orthogonality = 82.0
- Max skewness = 0.99
- Max aspect ratio = 136.3

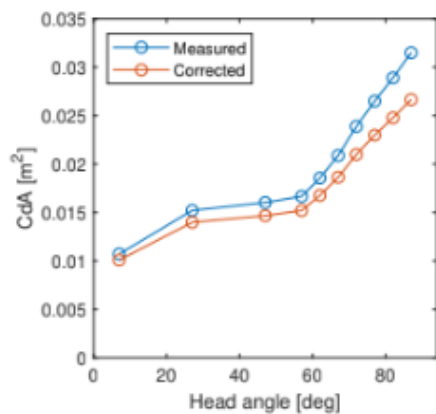
- Max non-orthogonality = 58.8
- Max skewness = 0.73
- Max aspect ratio = 140.0

For the OpenFOAM meshes, the non-orthogonality is a bit high, but there are few faces above 70 degrees. Some quality flaws must be expected and is hard to avoid when meshing complex geometries. However, Table 3.4.3 had lower values concerning the non-orthogonality.

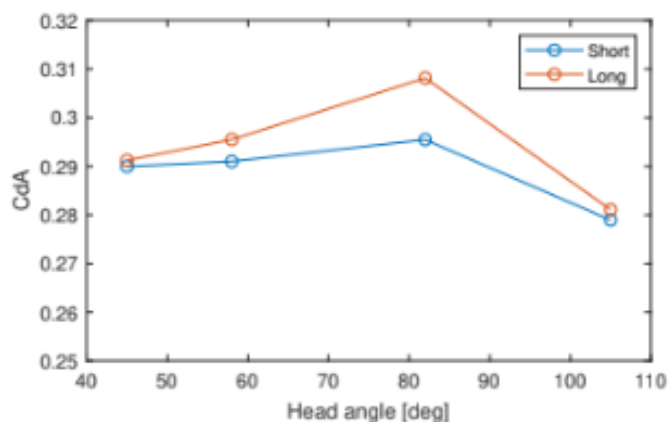
3.5 Head angles

Rider's head position will normally be around 45 degree. At this angle pupils will be in the middle of the eyes, providing good visibility of the road. This will be referred to as the normal position during the thesis. The head orientation might be change during the race. The rider might need to look down or to the sides, or head may fall as riders get tired. This may be crucial if a long tail helmet as in Figure 3.1b is used. In this study, further simulations of downward tilting of the head will be inspected, as it is the most exposed situation over time.

Studies performed at NTNU compared drag resistance for a helmet attached to a human and a helmet attached at a sphere mounted at a pole [44]. Result are presented in figure 3.16. Due to differences in areas, as helmet attached to a cyclist is bigger, values are not comparable only the tendencies. Ten angles were tested with the sphere, while the human experiment only had four angles tested. Both tests showed same tendencies, increase in resistance with increase in head angle.



(a) Air resistance vs head angle tested with sphere in wind-tunnel



(b) Air resistance vs head angle tested with human in wind-tunnel

Figure 3.16: Wind-tunnel test performed at NTNU

3.5.1 Measurement of head angles

In order to compare results from simulation and result from the NTNU's wind tunnel tests, it is important that the head angles are equal in both cases. To achieve this, it is necessary to have an expression for the angle. In the report from NTNU, a notion of head-tilt based on the Eye-Ear Line (E-E Line) was adopted. The E-E Line is used to describe the head tilt, or head angle, and is defined as a straight line passing through the eye and ear as seen in Figure 3.17 [45]. With this definition, the angle will be large when the rider is looking down, and small when looking straight forward. Even though this is a good way to measure the angle, it is still an approximation. There is therefore reason to assume that the head angle contain an uncertainty of ± 1 -2 degrees or more.

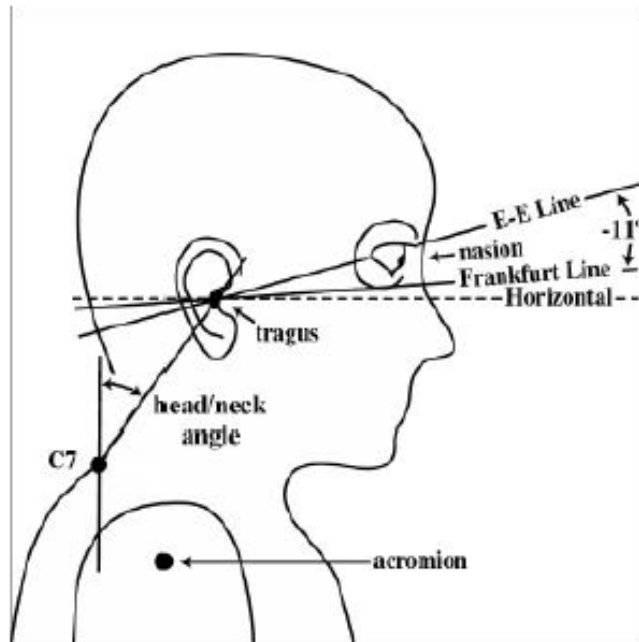


Figure 3.17: Head posture and definition of head landmarks and metrics

4. Evaluation of head angles

Angles tested in this section are approximately the same as in Lilands test for helmet attached to a sphere [44]. Figure 4.1 illustrates four of the angles, in addition seven more where tested. All angles are meshed and simulated based on the converged meshes created in OpenFOAM and Pointwise. Comparing the result from the two sets, possible differences may be manifested.

Helmets are orientated to a different angle in blender before imported to OpenFOAM, while in Pointwise, helmets were rotated directly in the software. Previous mesh set-up in both OpenFOAM and Pointwise where re-run in order to create new meshes for the oriented angles.

4.1 Results

Frontal area is listed in Table 3 and 4, showing an expected increase with growth in angle. Calculated air-resistance increases with angle growth up to 67 degrees for both sets of simulations. This is also illustrated in Figure 4.2 where results are plotted together with the wind tunnel tests performed at NTNU. The left y-label corresponds to the grey, blue and yellow lines, representing wind tunnel result of the helmet attached to a sphere and results from the simulations with Pointwise and OpenFOAM meshes respectively. While the right y-label represent the result of the human experiment. Results from the wind tunnel experiment with a sphere showed almost no increase in resistance from 45 to 58 degrees, but a steep almost linear increase during the seven measuring points from 58 to 87 degrees. The four measuring points in the human experiments had an increase in air resistance between 45 and 58 degrees, and an even steeper increase from 58 to 87 degrees. At 105 degree head angle, Liland

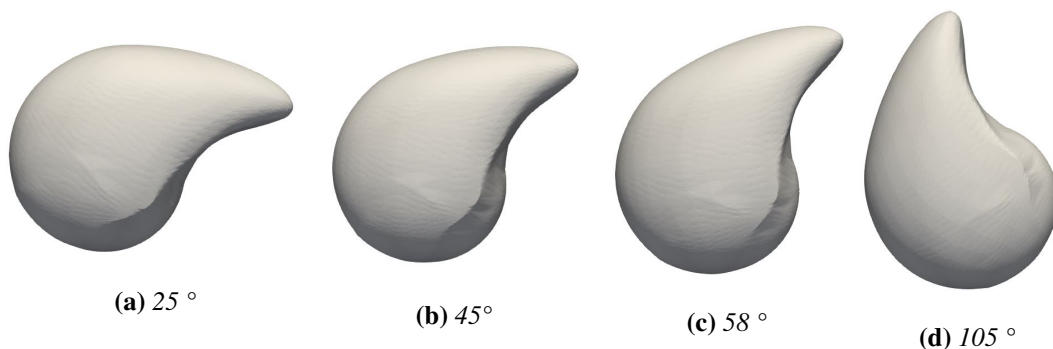


Figure 4.1: Four of the eleven different head angles tested.

Table 3: OpenFOAM results

Angle [deg]	$C_d A [m^2]$	y^+	Area [m^2]	Angle [deg]	$C_d A [m^2]$	y^+	Area [m^2]
7	0.0057	0.50	0.0356	⋮	⋮	⋮	⋮
25	0.0066	0.51	0.0367	72	0.0263	0.61	0.0446
45	0.0160	0.61	0.0380	77	0.0253	0.58	0.0459
58	0.0235	0.61	0.0402	82	0.0242	0.56	0.0475
62	0.0261	0.65	0.0416	87	0.0283	0.59	0.0488
67	0.0272	0.64	0.0431	105	0.0246	0.51	0.0532
⋮	⋮	⋮	⋮				

Table 4: Pointwise results

Angle [deg]	$C_d A [m^2]$	y^+	Area [m^2]	Angle [deg]	$C_d A [m^2]$	y^+	Area [m^2]
7	0.0062	0.52	0.0356	⋮	⋮	⋮	⋮
25	0.0076	0.54	0.0367	72	0.0248	0.62	0.0446
45	0.0158	0.62	0.0380	77	0.0250	0.60	0.0459
58	0.0225	0.64	0.0402	82	0.0252	0.58	0.0475
62	0.0244	0.64	0.0416	87	0.0257	0.58	0.0488
67	0.0252	0.63	0.0431	105	0.0266	0.55	0.0532
⋮	⋮	⋮	⋮				

stated that other elements affect the measurement due to substantial change in body position.

Results of the simulations performed in openFOAM have a fast increase in air resistance from 45 to 67 degrees, where it reach a peak of $C_d A$ at $0.027 m^2$. After 67 degrees, a strange behavior is seen in air resistance. Despite increase in area, diagram indicate random decrease and increase in air resistance. One reason for the rapid change in air resistance might be if Reynolds number is in the middle of the critical regime. It is hard to predict where the critical regime occurs. Shape of geometry is one factor affecting the critical Reynolds number, another factor which may have an affection is the coarse surface of the new geometry. Reynolds number calculated in Equation 3.2.1 gave a value at 302'000. As mentioned in theory Section 2.2.1, critical Reynolds number for a sphere is $4 \cdot 10^5$. This might give an indication of where in the flow regime in Figure 2.1 the flow is. Differences in geometry would most likely cause the present of turbulence earlier, and transfer the drag crisis to the left. If

Reynolds number is within the transition zone, variation in drag coefficient may be present with small variations of the Reynolds number. Which may explain the growth and reduction in air resistance for the simulations with the OpenFOAM mesh.

Until 58 degrees, simulations performed with Pointwise mesh has approximately same increase in air resistance as simulations with OpenFOAM-mesh. Then the increase gradually reduces up to 67 degrees and stabilize at $C_d A$ equal to $0.025 m^2$, despite the fact that frontal area is increasing. This clarify that are other aspects in addition to area affects the air resistance, like flow separations.

The air flow over and around the helmet has a massive influence on the aerodynamics. Boundary layers are formed around the helmet and change correspondingly with change in head angle. Streamlines in Figure 4.3 illustrates how air flows along the geometry. Flow separation in the figures shows a clear connection to the air resistance in Tables 3 and 4. At small angles up to 45 degrees, flow moves smoothly with the surface, causing less flow separations and corresponding low air drag. Already at 45 degrees, separations of flow is apparent as seen in Figure 4.3e and 4.3f. At larger head angles, more separations are present, again corresponding to an increase in drag force. Velocity changes are caused by variations in geometry. Comparing Figures 4.3 and 4.5 illustrated that flow separations in the streamlines are depended on change in velocity. In areas of high velocity, a higher share of separation is present. This is especially true for the plots created by the OpenFOAM mesh, which clearer illustrates this behavior. The velocity plots in Figure 4.5 clearly illustrate differences from simulations with OpenFOAM mesh and Pointwise mesh. The wake behind the helmet is better illustrated and more differences are present in results with the OpenFOAM mesh. At 45 degrees, Figure 4.3e illustrate a high velocity zone parallel to a low velocity zone behind the helmet. While Figure 4.3f illustrate one zone with reduced velocity. Same tendencies are present in figures representing larger head angles. But for them, the high velocity zone has decreased, now representing a small zone behind the helmet as in Figure 4.5g. The same smaller area is present in the plots with Pointwise mesh's results, but as an small area with velocity at approximately free flow velocity as in Figure 4.5h. This may affect the air resistance and explain why the values are higher in the OpenFOAM results.

Pressure plots of helmet's surroundings illustrates differences from the front to the back of the helmet in Figure 4.4. The pressure differential caused by flow separations often lead to increase in pressure drag. From figures, the dark red color illustrate large pressure at the front of the helmet and the blue lower pressure at the back. At 7 and 25 degree head angles, the pressure differentials are minimal.

At 45 degrees, more differential is present, and keep increasing with head angle. At 72 degree the under-pressure starts decreasing again for the results with OpenFOAM mesh, seen in Figure 4.4m, and may have influenced the stabilizing in air resistance. At 82 degrees in Figure 4.4m, the difference in under-pressure is significant lower compared 77 and 87 degrees, which corresponds with the drag resistance in Table 3. The under-pressure in results from simulations with Pointwise behind the helmet illustrates a more stable behavior.

The growth in air resistance for both sets of simulations reveal same tendencies as the results from wind tunnel, but with some differences. In addition to an offset of roughly 13 degrees, the growth is steeper than the growth of the sphere results. Determining the accuracy of head angle measurement is challenging. First measurement where made by an approximate E-E Line from section 3.5.1 and used as base to create the other angles. As height in eyes and ears varies, the line may be drawn differently in the experiments at NTNU compared to the simulations. Meaning that an inaccurate first measurement will give errors in the following angles. It can be speculated that one or both of the plots should be moved further to the left or right. The sphere is attached to a pole which also might have an impact on the results as flow separations may happened further from the helmets surface or be delayed.

In the wind tunnel results when helmet is attached to a human, a substantial growth is present already from 45 degrees. This corresponds more to simulations results which only has a slightly steeper increase in the same area. After 58 degrees, the experiment had an enormous increase in air resistance before the peak were reached at a head angle of 82 degrees, 15 degrees later than the simulations' peaks.

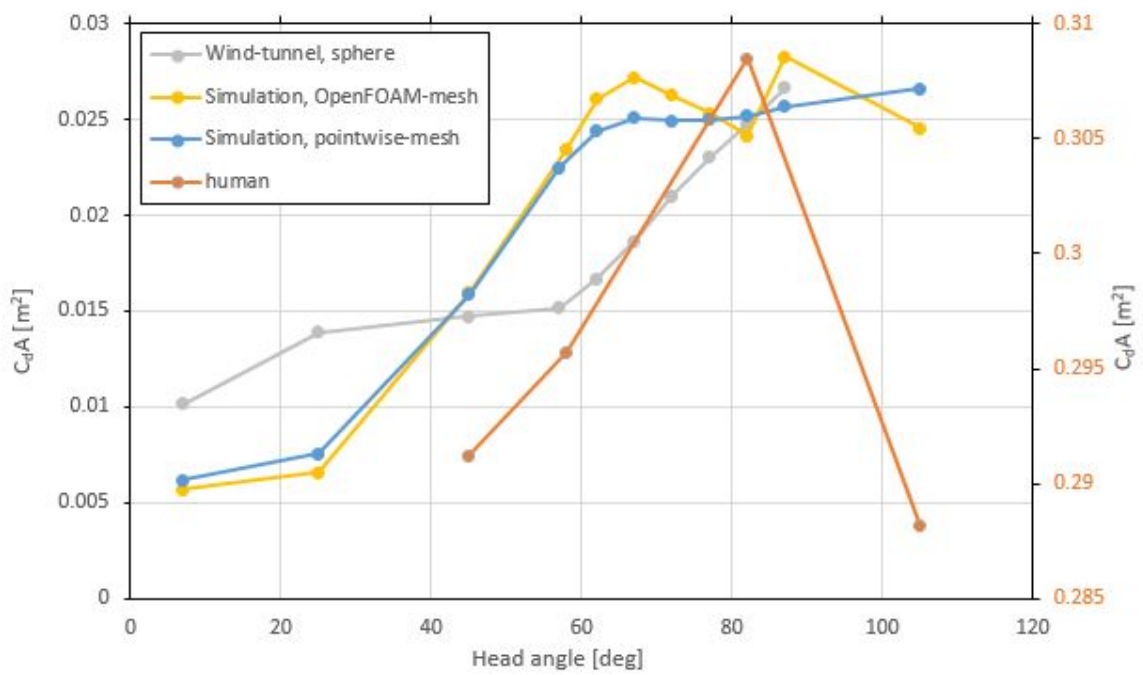


Figure 4.2: Air resistance from simulation and wind tunnel experiments



(a) 7°, *OpenFOAM*

(b) 7°, *Pointwise*



(c) 25°, *OpenFOAM*

(d) 25°, *Pointwise*



(e) 45°, *OpenFOAM*

(f) 45°, *Pointwise*



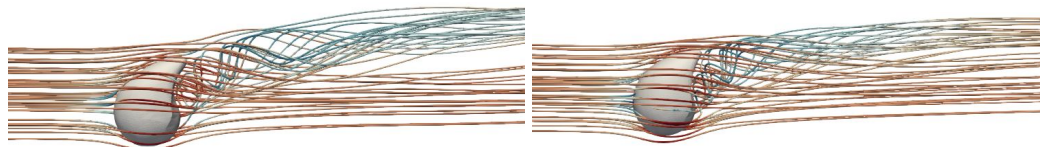
(g) 58°, *OpenFOAM*

(h) 58°, *Pointwise*



(i) 62°, *OpenFOAM*

(j) 62°, *Pointwise*



(k) 67°, *OpenFOAM*

(l) 67°, *Pointwise*

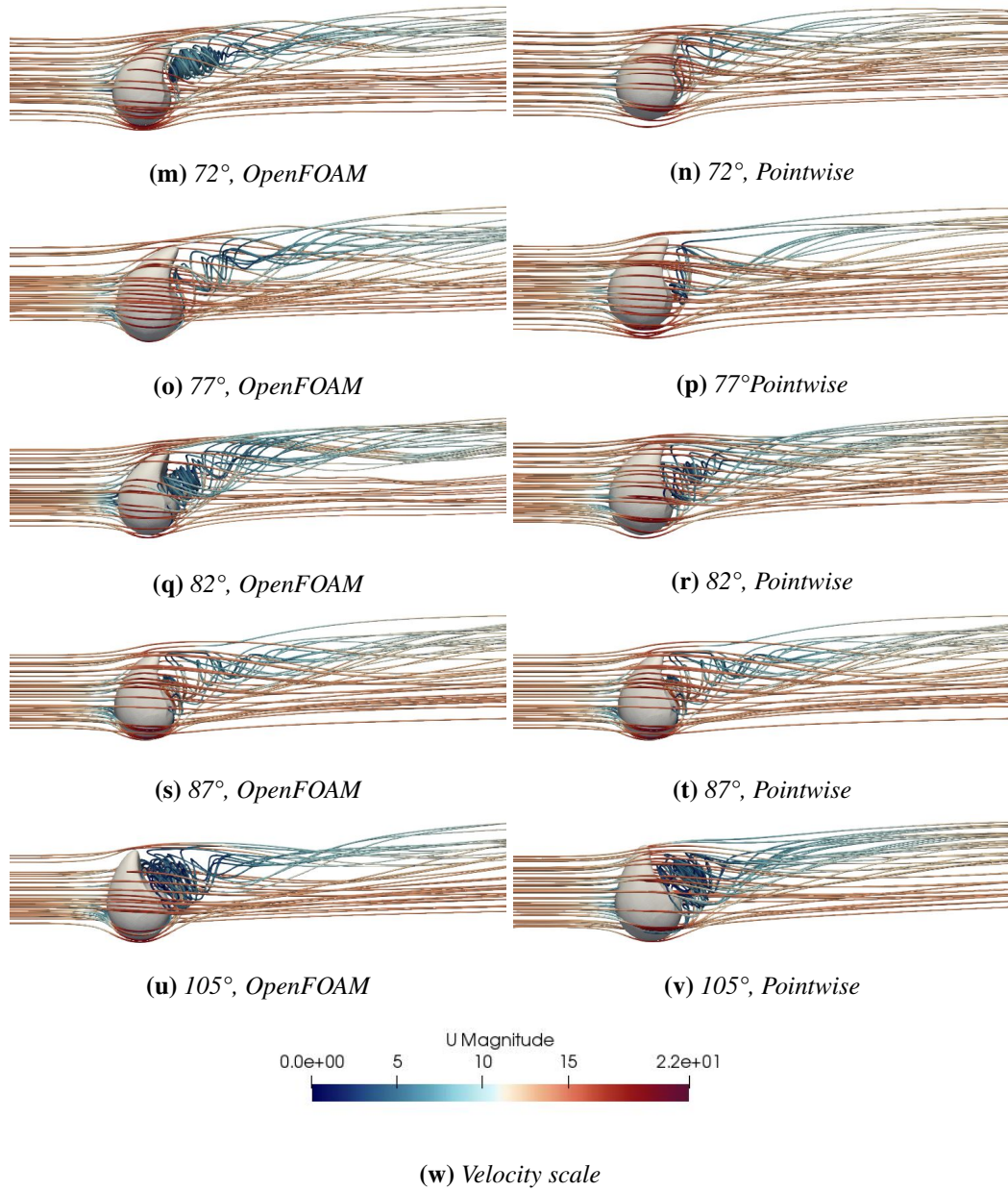
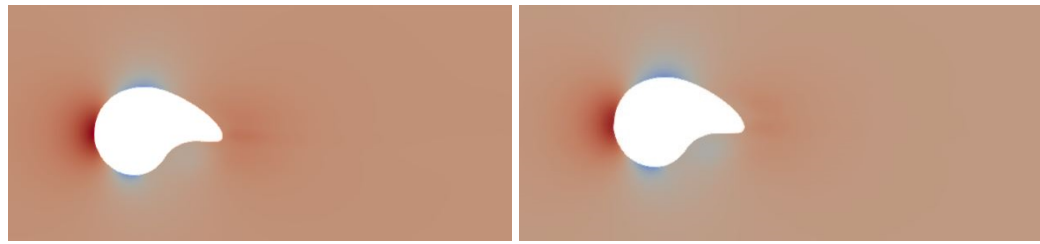
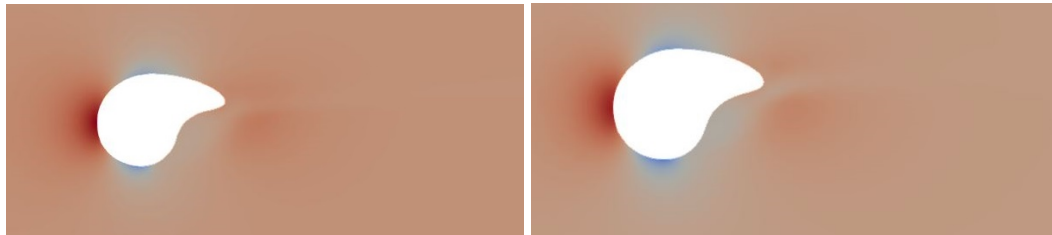


Figure 4.3: Streamlines of all angles meshed in OpenFOAM and Pointwise



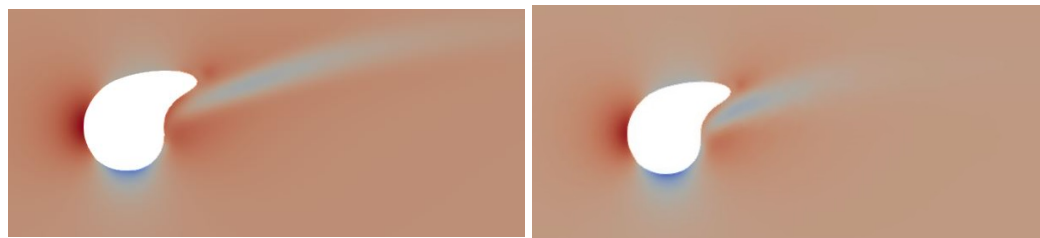
(a) 7°, *OpenFOAM*

(b) 7°, *Pointwise*



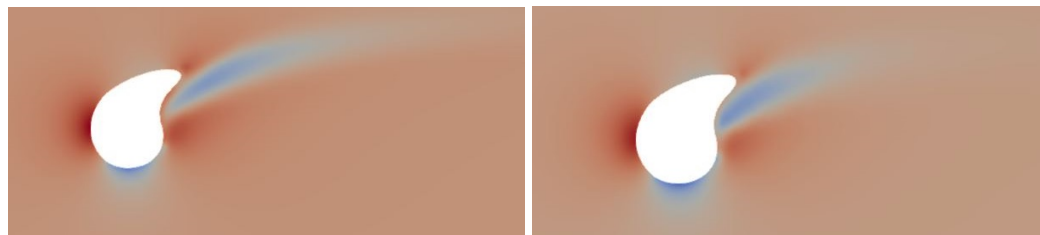
(c) 25°, *OpenFOAM*

(d) 25°, *Pointwise*



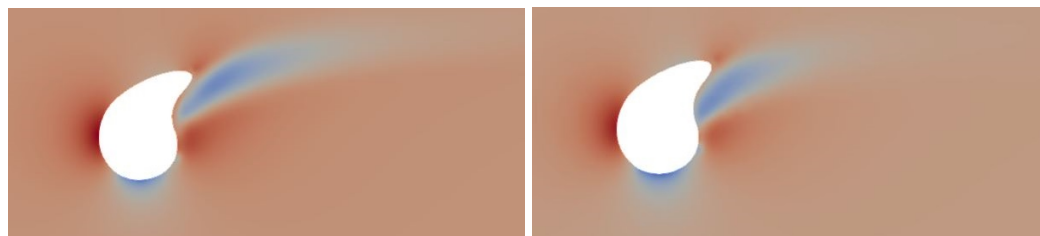
(e) 45°, *OpenFOAM*

(f) 45°, *Pointwise*



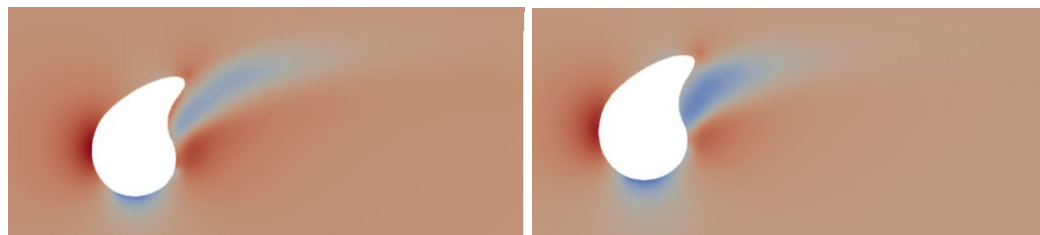
(g) 58°, *OpenFOAM*

(h) 58°, *Pointwise*



(i) 62°, *OpenFOAM*

(j) 62°, *Pointwise*



(k) 67°, *OpenFOAM*

(l) 67°, *Pointwise*

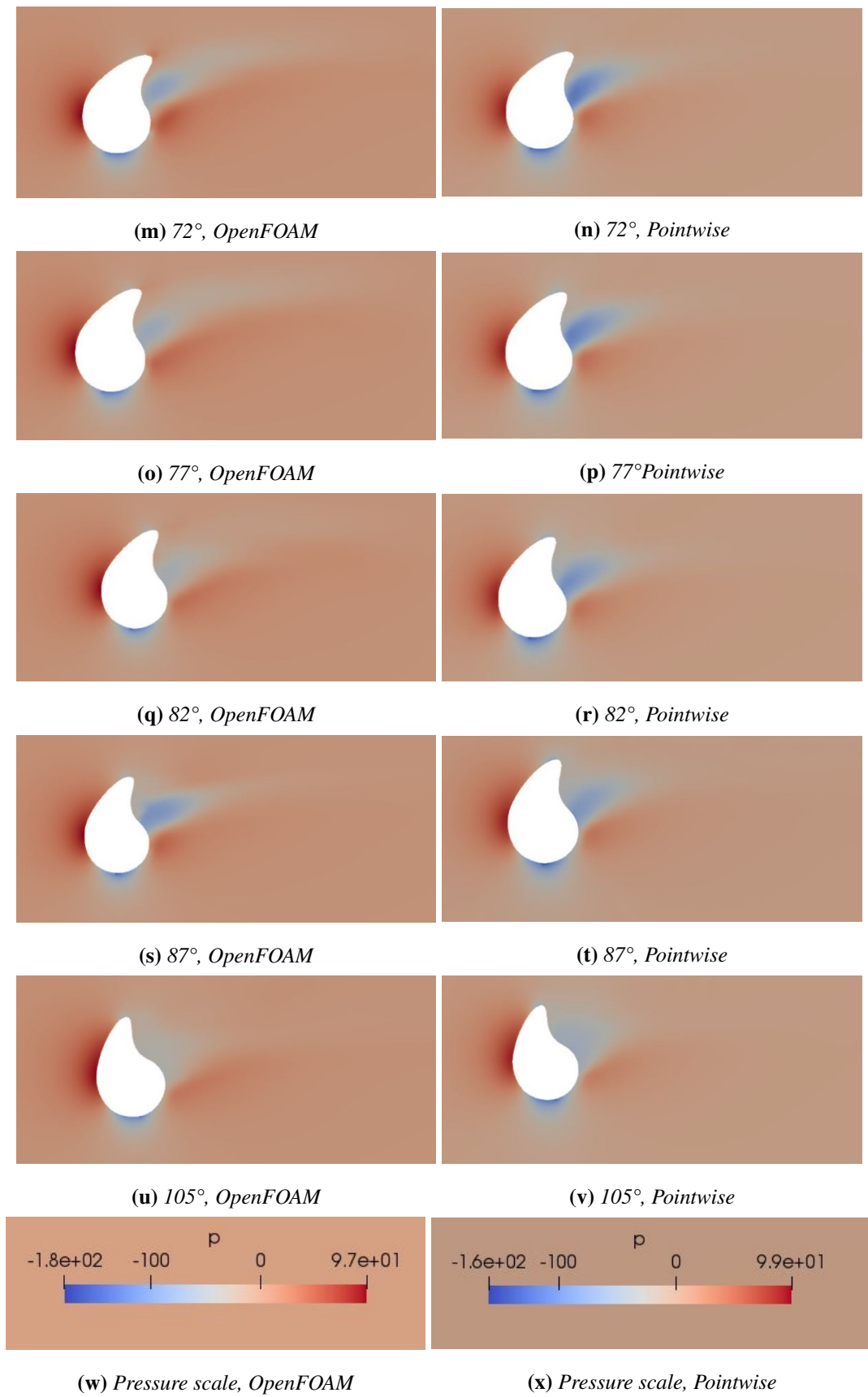
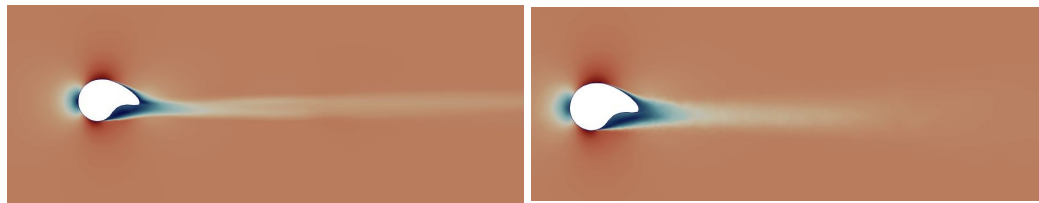
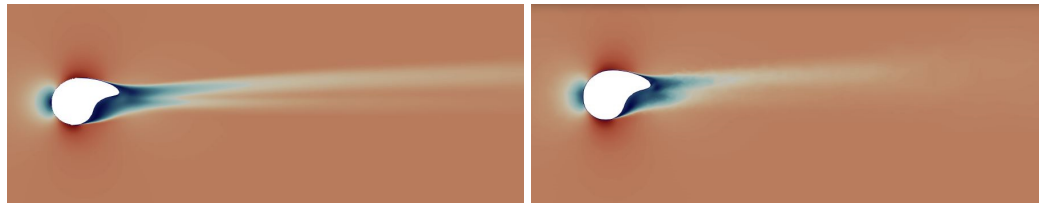


Figure 4.4: Pressure-plots of all angles meshed in *OpenFOAM* and *Pointwise*



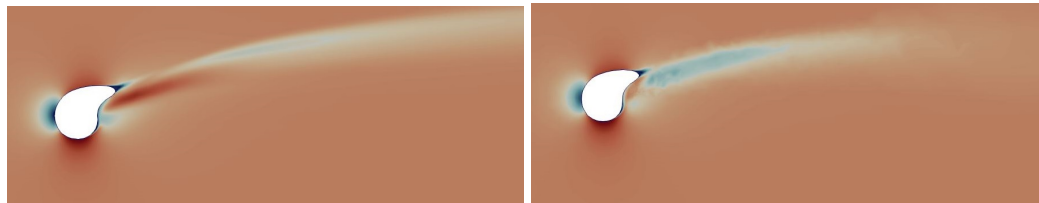
(a) 7°, *OpenFOAM*

(b) 7°, *Pointwise*



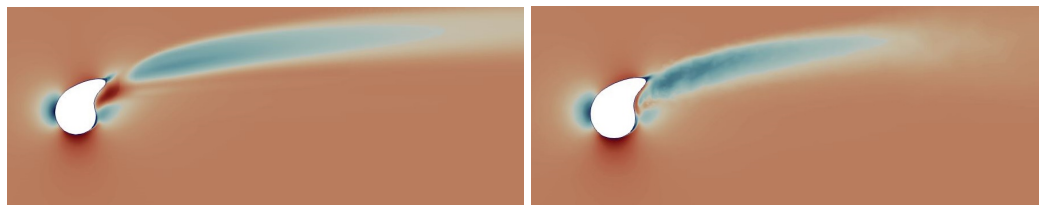
(c) 25°, *OpenFOAM*

(d) 25°, *Pointwise*



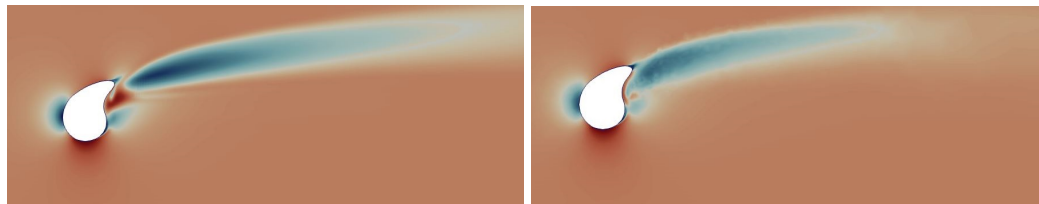
(e) 45°, *OpenFOAM*

(f) 45°, *Pointwise*



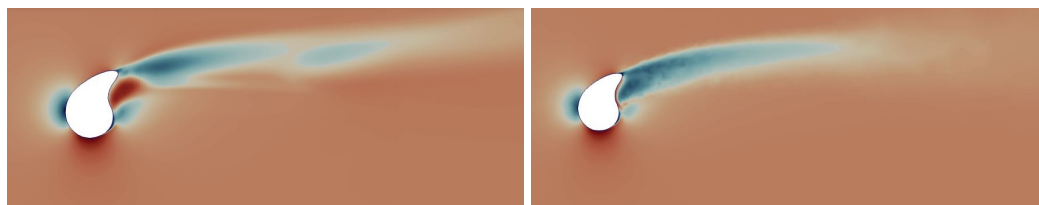
(g) 58°, *OpenFOAM*

(h) 58°, *Pointwise*



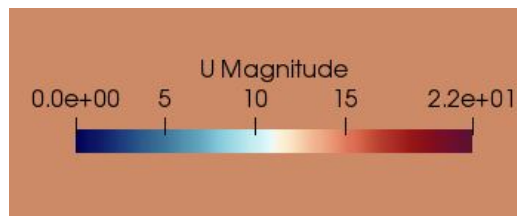
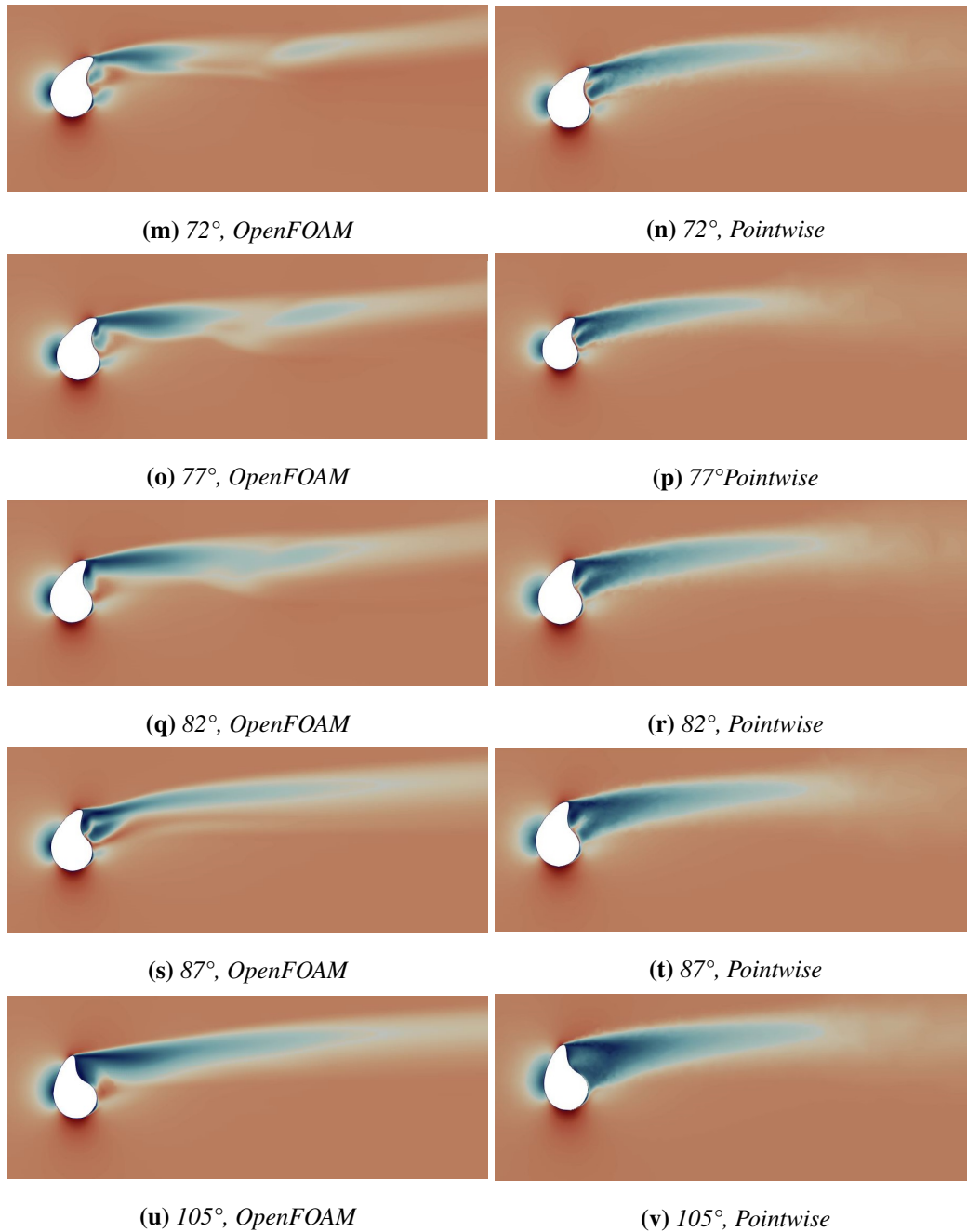
(i) 62°, *OpenFOAM*

(j) 62°, *Pointwise*



(k) 67°, *OpenFOAM*

(l) 67°, *Pointwise*



(w) *Velocity scale for both OpenFOAM and Pointwise*

Figure 4.5: *Velocity-plots of all angles*

4.2 Mesh quality

Different advantages and disadvantages are discovered when meshing with the software Pointwise and utilities in OpenFOAM. The latter one requires longer time in order to create a finish mesh. However, after geometry is inserted and specifications are set, the software creates the mesh it self OpenFOAM which can save the programmer from some work. In Pointwise, the controller must in addition to specify conditions, initialize steps during the process. However, time required until the mesh is finished, will be less compared to OpenFOAM. In addition the operator will have the advantage of visualizing the mesh during the generation. Pointwis' method of refining areas require less cells compared to OpenFOAM's, which in addition lead to less computational time when solving the simulation.

Results from the simulations are presented in table 3 and 4. Average y^+ value for all cases are of a reasonable value, below one, and varies from 0.5 to 0.65 in the OpenFOAM meses and 0.52 to 0.64 in Pointwise meshes.

There is no errors present in the Pointwise-meshes and requirements for both non-orthogonality and skewness where met in all cases. Highest value from all of the simulations with Pointwise mesh and in which simulation they occurred are listed:

- Max non-orthogonality = 62, found in the mesh for angle 77 degrees
- Max skewness = 0.73, same for all cases
- Max aspect ratio = 162.5, found in the mesh for angle 72 degrees

In OpenFOAM meshes, more quality lacks were found. Amount of lacks expanded with increase in head angle, but with some exceptions. Largest values and where they were found are listed below:

- Max non-orthogonality = 79.5 found in the mesh for angle 105 degrees.
- Non-orthogonality faces $> 70 = 26$ found in mesh for angle 105.
- Max skewness = 1.8 found in mesh for angle 67.
- Max aspect ratio = 209.3, found in the mesh for angle 58 degrees.

The behavior in the far field is clearer seen in the OpenFOAM mesh, compared to the Pointwise mesh. The velocity-stream plots of simulations created with OpenFOAM mesh show more turbulent and chaotic behavior in the streamlines. Differences is especially clear between figures 4.3e and 4.3f, 4.3m and 4.3n and 4.3o and 4.3p, while other figures like 4.3i and 4.3j, 4.3s and 4.3t and 4.3u and 4.3v showed more of the same tendencies and less differences between the figures. This is especially clear to see in figure 4.3 and 4.5. The wake is clearer in the far field at the OpenFOAM mesh, and catches the behavior of the wake. The far field is poorer in the Pointwise mesh compared to the OpenFOAM mesh.

It is hard to tell if the mesh quality have had an impact on the solution. Variances after 62 degrees in the OpenFOAM might been caused by lack of accuracy in the solution due to poor mesh generation, as values grow but due to values of max non-orthogonality the probability is low.

5. Biomimetic optimization of helmet

Previous research and results from simulations states that increase in head angle leads to increase in drag resistance. The growth in air resistance is already significant after 13 degrees tilting, which may be crucial in a race. Liland [44] and Beaumont [8] compared long tailed helmet to a short tailed helmet. Both studies showed that the long tailed helmet had less air resistance at small head angles, while the short tailed helmet had the least air resistance at larger head angles.

5.1 Armadillo

The design process for optimization of the helmet is in the Top-Down mimicry category. The design problem is to create a helmet with a long tail when the head is at normal position, and a short tail when the head is tilted downwards. Taking inspiration from the armadillo, such a helmet might be crated. Armadillo is an animal living in America and is known for its rare shell. The shell is made from bony plates covered in a leathery keratinous skin. This gives the armadillo the opportunity to curl into a ball for protection when its exposed to danger [46]. Figure 5.1 from national geographic clearly illustrate the nine banded armadillo shell [47].

Inspired by the armadillo, a helmet who adapts the tail with head angle, by either expanding or shrinking as the head is tilting downward or upward respectively will be explored. Creating this will avoid the tail pointing upwards and the corresponding large frontal area with growth in head angles.



Figure 5.1: Here it is easy to see the bony plates in the shell of the armadillo

5.2 Modification of helmets

As seen from the result of the previous chapter, the increase in resistance happened between 45 and 63 degrees of head tilting. From Tables 3 and 4 it is seen that the area grows with increase in angle which is illustrated in Figure 4.1. Using the helmet at 45 degree as foundation, other head angles were modified using inverse kinematics in blender, forcing the frontal area to be inside the black constraints from the 45 degree helmet, seen in Figure 5.2.

Figure 5.3 illustrates the original helmet at 58 degree, armature used during the inverse kinematic and the modified helmet. The chain-length for the armature in Figure 5.3b are 4 bones long. This keeps the first and largest chain-member fixed to ensure that the bottom of the helmet is un-moveable. The last member, on the outside of the helmet, is the handle controlling the movement of the armature. The modified helmet is illustrated in Figure 5.3c. It is possible to see the adjusted tail, now pointing backwards instead of upwards.

5.3 Evaluation of modified helmets

Two of the original helmets were modified. Helmet at 58 degrees, as explained in Section 5.2, and helmet at 82 degrees. This will give an indication of how adjustments affects the air resistance in a region where air resistance is growing, and in a region where it has stabilized. From Table 5 it is specified that the y^+ value for the modified 58 degree helmet is a bit higher compared to previous in Table 3 and 4, but still at a reasonable level. The calculated frontal area confirms that area has

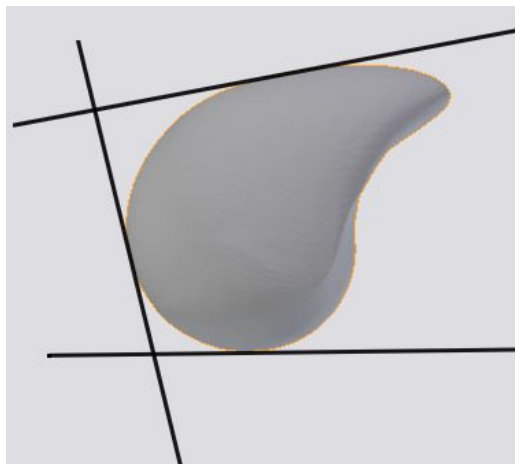


Figure 5.2: Constraints from helmet at 45 degrees

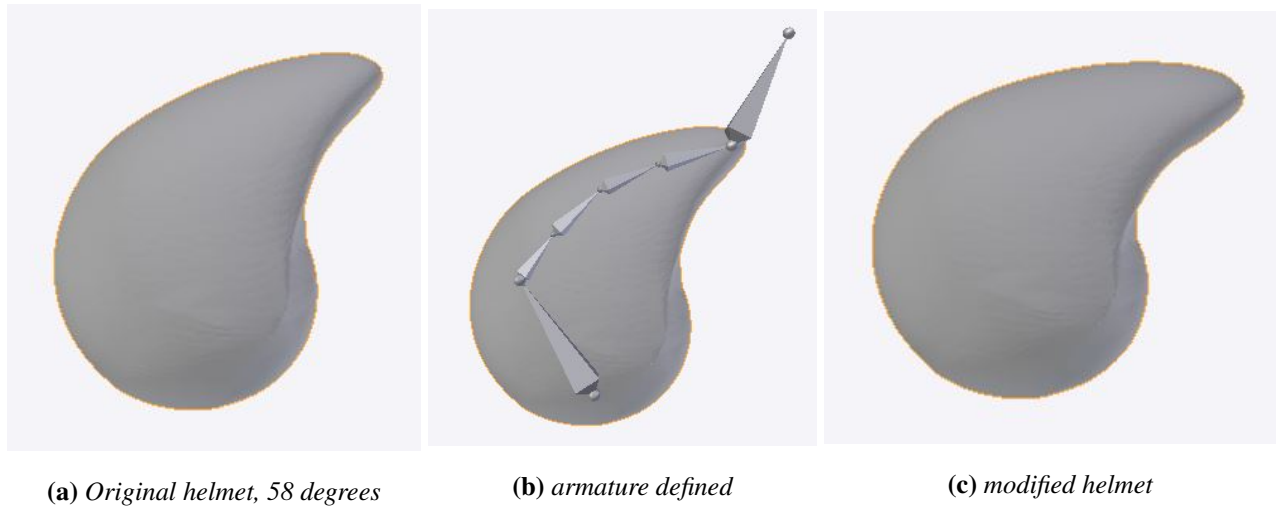


Figure 5.3: Three main steps of modifying the geometry

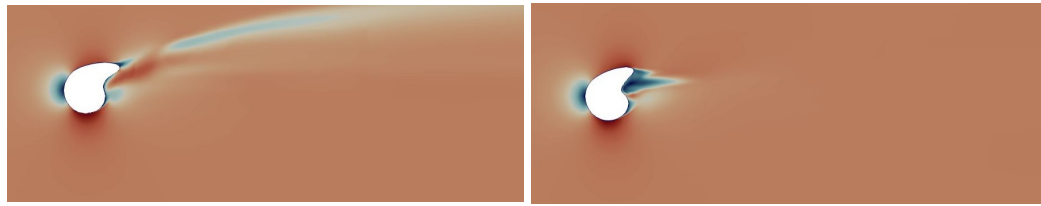
increased from previous values at 0.0402 m^2 and 0.0475 m^2 for 58 and 82 degrees angles respectively corresponding to 14.9% and 13.7% area reduction. The 58 degree helmet's C_dA value has decreased with 22% to 0.018 m^2 which is slightly above the air resistance for helmet at 45 degree at 0.016 m^2 . The air resistance for the modified 82 degree helmet also experienced improvement and decreased 13%, from the original value at approximately 0.027 m^2 to 0.0234 m^2 , approximately same values as the original air resistance for the 58 degree helmet.

The velocity plots illustrates clearly differences from the original helmets at same orientation, this is especially true for the 82 degree case. Figure 5.4a has lesser differences in the wake compared with 4.5g and 4.5h, and looks a bit more like 4.5e and 4.5f. Streamlines in 5.5a are closer packed then in 4.3g and 4.3h. The pressure wake in 5.6a is a bit smaller then 4.4g and 4.4h. All observations corresponds with the new, lower value for air resistance.

The biggest benefit of CFD simulations is the possibility of doing testing without the physical geometry. This advantage has been used in this section, where the modified helmet has been tested. Results illustrate that resources should be used to further investigation and examine if it is physical feasibly.

Table 5: Properties of modified helmet

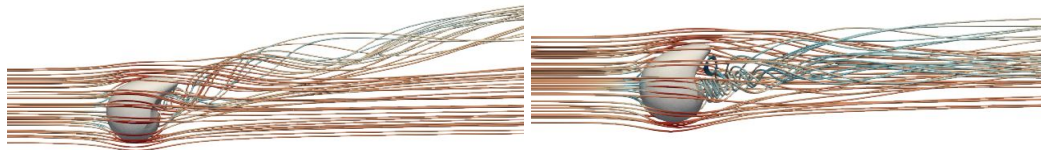
Angle	C_dA	y^+	Area
58	0.0175	1.49	0.0396
82	0.0234	0.58	0.0411



(a) *Modified 58 degree helmet*

(b) *Modified 82 degree helmet*

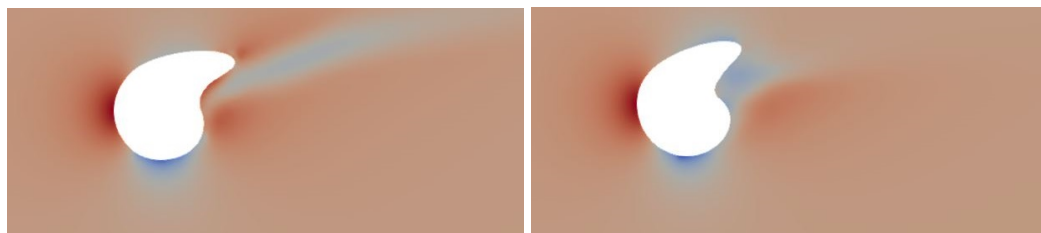
Figure 5.4: *Velocity-plots of the modified helmets*



(a) *Modified 58 degree helmet*

(b) *Modified 82 degree helmet*

Figure 5.5: *Streamlines of the modified helmets*



(a) *Modified 58 degree helmet*

(b) *Modified 82 degree helmet*

Figure 5.6: *Pressure-plots of the modified helmets*

6. Conclusion

In this thesis, a time trial helmets aerodynamics has been investigated in order to reveal if simulations of only the helmet is reliable. CFD is used to execute simulations, and two different meshing softwares were used to generate meshes and state differences between them. At the end, a new distinctive helmet were suggested, based on biomimetic and the behavior of an armadillo.

Results manifested that air resistance increased with increase in head angle and the corresponding expanding frontal area. This means, that when athletes looks down, extra power must be produced to overcome the additional resistance. Both sets of the executed simulations had approximately same tendencies and values until head angle reached 67 degree. After that, the results from the OpenFOAM mesh started altering while results from Pointwise stabilized.

Results revealed same tendencies as the wind tunnel experiments performed with helmet attached to a sphere and helmet attached to human. Despite an offset of 13 degrees, simulations had a similar slope in air resistance increase compared with sphere results. The human measurement had an increase in air resistance present already from first measuring point, 45 degree, corresponding with simulation results.

Generating mesh by utilities in OpenFOAM, the operator will have hands available for other tasks after the dictionary is set. However, utilities struggles with complex geometries and advantage of visualizing the mesh while it is generated is not possible, as it is in Pointwise. Overall the mesh generated in Pointwise had better quality and required less cells compared to the OpenFOAM mesh. Which were reflected in computational time.

Simulations of the helmet alone might be appropriate in order to check differences between geometries. The helmets' small scale geometry, will make the already time and cost saving CFD tolls more efficient. However, results gained from simulations will not be accurate and additional testing will be required to obtain precise results.

The armadillo inspired helmet showed positive results in air resistance with 22 and 13% improvement in air resistance. The frontal area were reduced and streamlines float nice around the helmet. It might be a viable option to explore for helmets producers.

6.1 *Further-work*

More wind tunnel experiments with helmet attached to a human should be performed, as it is give the most realistic measurement of how change in cyclist's head angle affects the air resistance. It would be interesting comparing results from simulations with more measured angles.

For future meshing with Pointwise, it would be recommended that the far field around surfaces are finer in order to obtain better resolution of the wake.

Further exploration of the armadillo helmet should be performed. Recreation of the armadillo helmet should be investigated in order to obtain the wanted behavior of the helmet and detect if the method is feasible.

References

- [1] Amaury Sport Organisation, “The history of the tour de france,” 2019, (). [Online]. Available: <https://www.letour.fr/en/history>. [Accessed: 2019-08-06].
- [2] C. R. Kyle and E. Burke, “Improving the racing bicycle,” *Mechanical Engineering*, vol. 106, pp. 34–35, 09 1984.
- [3] H. Chowdhury, F. Alam, and I. Khan, “An experimental study of bicycle aerodynamics,” *International Journal of Mechanical and Materials Engineering*, vol. 6, pp. 269–274, 08 2011.
- [4] M. Zdravkovich, “Aerodynamics of bicycle wheel and frame,” *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 40, pp. 55–70, 04 1992.
- [5] B. A. Parkera, M. E. Franke, and A. W. Elledge, “Bicycle aerodynamics and recent testing,” 1996. [Online]. Available: <https://arc.aiaa.org/doi/abs/10.2514/6.1996-557>. [Accessed: 2019-06-06].
- [6] L. Brownlie, C. Kyle, J. Carbo, N. Demarest, E. Harber, R. MacDonald, and M. Nordstrom, “Streamlining the time trial apparel of cyclists: the nike swift spin project,” *Sports Technology*, vol. 2, no. 1-2, pp. 53–60, 2009.
- [7] S. Sidelko, “Benchmark of aerodynamic cycling helmets using a refined wind tunnel test protocol for helmet drag research,” 2007. [Online]. Available: <https://dspace.mit.edu/bitstream/handle/1721.1/40486/191803811-MIT.pdf?sequence=2&isAllowed=y>. [Accessed: 2019-06-08].
- [8] F. Beaumont, R. Taiar, G. Polidori, H. Trenchard, and F. Grappe, “Aerodynamic study of time-trial helmets in cycling racing using cfd analysis,” *Journal of Biomechanics*, vol. 67, pp. 1 – 8, 2018.
- [9] Pointwise, “How meshing is better,” 2019. [Online]. Available: <https://www.pointwise.com/pointwise/>. [Accessed: 2019-02-05].
- [10] J. C. Martin, D. Milliken, J. E. Cobb, K. L. McFadden, and A. R. Coggan, “Validation of a mathematical model for road cycling power,” *Journal of Applied Biomechanics*, vol. 14, pp. 276–291, 08 1998.

-
- [11] F. White, *Fluid Mechanics*. McGraw-Hill series in mechanical engineering, McGraw Hill, 2011.
- [12] H. Versteeg and W. Malalasekera, “An introduction to computational fluid dynamics : the finite volume method / h. k. versteeg and w. malalasekera.” 04 2007.
- [13] C. C. Ngo and K. Gramoll, “Boundary layer theory - flow past a flatplate,” 2001. [Online]. Available: https://ecourses.ou.edu/cgi-bin/ebook.cgi?doc=&topic=fl&chap_sec=09.3&page=theory. [Accessed: 2019-05-27].
- [14] N. Beratlis, K. Squires, and E. Balaras, “Separation control and drag reduction using roughness elements,” 2017. [Online]. Available: <http://www.tsfpc-conference.org/proceedings/2017/2/397.pdf>. [Accessed: 2019-05-27].
- [15] H. Schlichting and K. Gersten, *Boundary layer Theory*. Springer, 2000.
- [16] N. J. Georgiadis* and D. A. Yoder, “Evaluation of modified two-equation turbulence models for jet flow predictions,” 2006. [Online]. Available: <https://www.grc.nasa.gov/WWW/wind/papers/AIAA-2006-0490-Georgiadis.pdf>. [Accessed: 2019-03-20].
- [17] R. Christopher, “The menter shear stress transportturbulence model the menter shear stress transportturbulence model,” 2015. [Online]. Available: <https://turbmodels.larc.nasa.gov/sst.html>. [Accessed: 2019-03-20].
- [18] C. E. Brennen, “An internet book on fluid dynamics,” 2006. [Online]. Available: <http://brennen.caltech.edu/fluidbook/basicfluidynamics/turbulence/lawofthewall.pdf>. [Accessed: 2019-04-04].
- [19] M. H. Nilsson, “A thorough description of how wall functions are implemented in openfoam,” 2017. [Online]. Available: http://www.tfd.chalmers.se/~hani/kurser/OS_CFD_2016/FangqingLiu/openfoamFinal.pdf. [Accessed: 2019-06-04].
- [20] OpenCFD, “k-omega shear stress transport (sst),” 2019. [Online]. Available: <https://www.openfoam.com/documentation/guides/latest/doc/guide-turbulence-ras-k-omega-sst.html>. [Accessed: 2019-06-06].
- [21] DE Editors, “Fast app: T-rex is fast, accurate boundary layer meshing from point-wise,” 2010. [Online]. Available: <https://www.digitalengineering247.com/article/>
-

-
- fast-app-t-rex-is-fast-accurate-boundary-layer-meshing-from-pointwise/. [Accessed: 2019-05-14].
- [22] J. Rhoads, “Effects of grid quality on solution accuracy,” in *OpenFOAM Workshop 2014:*, (Zagreb, Croatia), July 2014.
- [23] F. Moukalled, L. Mangani, and M. Darwish, *The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with OpenFOAM and Matlab*. Springer Publishing Company, Incorporated, 1st ed., 2015.
- [24] K. E. Giljarhus, *OpenFOAM tutorials*. University of Stavanger, January 2018.
- [25] ANSYS, “6.2.2 mesh quality,” 2009. [Online]. Available: <http://www.afs.enea.it/project/neptunius/docs/fluent/html/ug/node167.htm>. [Accessed: 2019-05-27].
- [26] R. Gullberg, “Computational fluid dynamics in openfoam, mesh generation and quality,” 2017. [Online]. Available: http://folk.ntnu.no/preisig/HAP_Specials/AdvancedSimulation_files/2017/project%20reports/CFD/Rebecca%20Gullberg%20-%20CFD_Mesh_Report.pdf. [Accessed: 2019-03-04].
- [27] CFD Direct, “Openfoam v6 user guide: 5.4 mesh generation with snappyhexmesh,” 20189. [Online]. Available: <https://cfdirect.com/openfoam/user-guide/v6-snappyhexmesh/>. [Accessed: 2019-03-02].
- [28] OpenFOAM, “About openfoam,” 2019. [Online]. Available: <https://www.openfoam.com/>. [Accessed: 2019-06-11].
- [29] B. H. Hjertager and K. E. Giljarhus, “What is openfoam?,” January 2018.
- [30] B. H. Hjertager and K. E. Giljarhus, “Openfoam cases: How to run an openfoam solver,” January 2018.
- [31] Applied CCM, “Pointwise,” 2018. [Online]. Available: <http://www.appliedccm.com/pointwise/>. [Accessed: 2019-05-13].
- [32] Blender, “Free software never looked this awesome,” 2019. [Online]. Available: <https://www.blender.org/features/>. [Accessed: 2019-03-06].
-

-
- [33] Blender, “Modelling, modifiers, introduction,” 2019. [Online]. Available: <https://docs.blender.org/manual/en/dev/modeling/modifiers/introduction.html>. [Accessed: 2019-03-06].
- [34] Lumen, “Basics of kinematics,” 2019. [Online]. Available: <https://courses.lumenlearning.com/boundless-physics/chapter/basics-of-kinematics/>. [Accessed: 2019-06-11].
- [35] M. S. Aziz and Y. El sherif, “Biomimicry as an approach for bio-inspired structure with the aid of computation,” 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1110016815001702#!>. [Accessed: 2019-06-02].
- [36] Pointwise, “Planning your grid,” 2018. [Online]. Available: <https://pointwise.com/doc/user-manual/getting-started/planning-your-grid.html>. [Accessed: 2019-05-05].
- [37] Pointwise, “On database entities,” 2018. [Online]. Available: <https://www.pointwise.com/doc/user-manual/create/on-database-entities.html>. [Accessed: 2019-05-05].
- [38] J. Guerrero, “Tips and tricks in openfoam,” 2016. [Online]. Available: <http://www.wolfdynamics.com/wiki/OFtipsandtricks.pdf>. [Accessed: 2019-03-06].
- [39] W. W Goddard, “Handbook of thermodynamic tables and charts,” *Physics Bulletin*, vol. 28, pp. 429–429, 09 1977.
- [40] J. Nobrega and H. Jasak, *OpenFOAM® Selected Papers of the 11th Workshop: Selected Papers of the 11th Workshop*. 01 2019.
- [41] Pointwise, “Applications of hybrid and unstructured meshing,” 2019. [Online]. Available: <https://www.pointwise.com/hybrid/>. [Accessed: 2019-06-05].
- [42] OpenCFD, “A.1 standard solvers,” 2018. [Online]. Available: <https://www.openfoam.com/documentation/user-guide/standard-solvers.php>. [Accessed: 2019-06-05].
- [43] Leap CFD team, “Tips & tricks: Convergence and mesh independence study,” 2012. [Online]. Available: <https://www.computationalfluidynamics.com.au/convergence-and-mesh-independent-study/>. [Accessed: 2019-04-20].
- [44] F. F. Liland, “Air resistance of time trial helmets at different head angles,” Master’s thesis, NTNU, 2018.

-
- [45] K. N. D. Ankrum, "Head and neck posture at computer workstations - what's neutral?," 2000. [Online]. Available: https://www.humanics-es.com/head_and_neck_posture_at_computer_workstations-ankrum.pdf.
- [46] H. Rhee, M.F.Horstemeyer, and A.Ramsay, "A study on the structure and mechanical behavior of the dasypus novemcinctus shell," 2012. [Online]. Available: {<https://www.sciencedirect.com/science/article/pii/S0928493110002493?via%3Dihub>}. [Accessed:2019-05-20].
- [47] NATIONAL GEOGRAPHIC, "These animals inspire better body armor for humans," 2019. [Online]. Available: <https://www.nationalgeographic.com/animals/2019/02/animals-armor-bioinspiration/>. [Accessed: 2019-06-10].