

# Micro-scale wind resource assessment in complex terrain using CFD

*Study case: Sierra Morena, Chiapas*

**Masterarbeit**

*von*

**Jorge Ortigosa**

Erstgutachter:

Prof.Dr.-Ing.Paul Uwe Thamsen

Institut für Strömungsmechanik und Technische Akustik (TUB)

Mitberichter:

M.Sc. Moritz Mühlbauer

Institut für Strömungsmechanik und Technische Akustik (TUB)

Berater auf UNAM

Dr.-Ing.César Angeles Camacho

Instituto de Ingeniería (UNAM)

**Technische Universität Berlin**

4. April 2017

**Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.**

**Mexiko Stadt, den 4.April 2017**

.....  
**Jorge Ortigosa**

# Index

<b>Index.....</b>	<b>2</b>
<b>Figures.....</b>	<b>6</b>
<b>Tables.....</b>	<b>9</b>
<b>Nomenclature .....</b>	<b>10</b>
<b>Acknowledgements.....</b>	<b>17</b>
<b>Abstract (Deutsch) .....</b>	<b>18</b>
<b>Abstract (English) .....</b>	<b>19</b>
 <b>Chapter 1: Introduction.....</b>	 <b>20</b>
<b>1.1 Energy Access and Electrification.....</b>	<b>20</b>
<b>1.2 Wind Energy.....</b>	<b>21</b>
<i>1.2.1 Complex Terrain.....</i>	<i>21</i>
<i>1.2.2 Wind Resource assessment.....</i>	<i>22</i>
<i>1.2.3 Wind Assessment Software .....</i>	<i>22</i>
<b>1.3 Rural Electrification México.....</b>	<b>23</b>
<b>1.4 Wind Resource Chiapas.....</b>	<b>23</b>
<b>1.5 Objectives.....</b>	<b>24</b>
<b>1.6 Motivation.....</b>	<b>24</b>
<b>1.7 Methodology.....</b>	<b>25</b>
 <b>Chapter 2: Theoretical Background.....</b>	 <b>26</b>
<b>2.1 Atmospheric Boundary Layer (ABL) .....</b>	<b>26</b>
<i>2.1.1 Stability and Stratification.....</i>	<i>26</i>
<i>2.1.2 Velocity Vertical Profile.....</i>	<i>27</i>
<b>2.2 Turbulence.....</b>	<b>28</b>
<i>2.2.1 The Eddies.....</i>	<i>28</i>
<i>2.2.2 Turbulence Modelling.....</i>	<i>28</i>
<i>2.2.3 RANS Models.....</i>	<i>29</i>
<i>2.2.4 Near-Wall Physics.....</i>	<i>33</i>
<i>2.2.5 The Standard Wall Functions .....</i>	<i>36</i>
<b>2.3 The Canopy Model.....</b>	<b>37</b>
 <b>Chapter 3: Computational Fluid Dynamics (CFD) .....</b>	 <b>40</b>
<b>3.1 Computational Fluid Dynamics (CFD) .....</b>	<b>40</b>
<b>3.2 The Finite Volume Method (FVM) .....</b>	<b>41</b>
<i>3.2.1 Interpolation Schemes.....</i>	<i>42</i>
<i>3.2.2 Solvers.....</i>	<i>45</i>
<i>3.2.3 Pressure-Velocity Coupling.....</i>	<i>45</i>
<b>3.3 Relaxation.....</b>	<b>46</b>

3.4 Residuals.....	47
3.5 Convergence.....	47
<b>Chapter 4: Simulation Construction.....</b>	<b>48</b>
4.1 OpenFOAM.....	48
4.2 Pre-processing.....	49
4.2.1 The Domain.....	49
4.2.1.1 Dimensions.....	49
4.2.1.2 Meshing.....	49
4.2.1.3 Meshing Quality Guidelines.....	50
4.2.1.4 Meshing Tools.....	50
4.2.2 Boundary Conditions (BC) .....	51
4.2.3 Fluid Properties.....	55
4.2.4 Transport Properties.....	56
4.2.5 Canopies .....	56
4.2.6 The groovyBC utility.....	58
4.3 Solving.....	58
4.3.1 Discretization Schemes.....	59
4.3.2 Linear solvers.....	59
4.3.3 Application Solvers.....	60
4.3.4 Relaxation.....	60
4.3.5 Tolerance and Convergence.....	60
4.3.6 Time Step .....	61
<b>Chapter 5: Simulations Tests.....</b>	<b>62</b>
5.1 Horizontal Homogeneity (HH).....	62
5.1.1 First Node Position.....	63
5.1.2 Vertical Grading.....	67
5.1.3 The Aspect Ratio.....	69
5.2 Obstacles.....	70
5.2.1 The Canopy.....	71
5.3 Conclusions.....	75

<b>Chapter 6: Validation Process.....</b>	<b>76</b>
<b>6.1 The Bolund Experiment.....</b>	<b>76</b>
6.1.2 <i>The Bolund Hill</i> .....	76
6.1.3 <i>The Blind Comparison</i> .....	77
<b>6.2 The Simulation.....</b>	<b>78</b>
6.2.1 <i>The Domain</i> .....	78
6.2.2 <i>Boundary Conditions</i> .....	79
6.2.3 <i>Grid Sensitivity Analysis</i> .....	80
<b>6.3 Validation Cases.....</b>	<b>80</b>
6.3.1 <i>Reference Wind</i> .....	80
6.3.2 <i>Speed-Up Wind</i> .....	81
6.3.3 <i>Turbulent Kinetic Energy</i> .....	83
<b>6.4 Conclusions.....</b>	<b>85</b>
 <b>Chapter 7: Study Case: Sierra Morena.....</b>	 <b>86</b>
<b>7.1 Field Work.....</b>	<b>86</b>
<b>7.2 Village Choice.....</b>	<b>86</b>
<b>7.3 Wind Data .....</b>	<b>87</b>
<b>7.4 Atmospheric Conditions and Model Limitations .....</b>	<b>88</b>
7.4.1 <i>Turbulence</i> .....	88
7.4.2 <i>Atmospheric Stability</i> .....	88
7.4.3 <i>Coriolis Force</i> .....	89
<b>7.5 Terrain Input Data.....</b>	<b>89</b>
7.5.1 <i>Orography</i> .....	89
7.5.2 <i>Terrain Ruggedness</i> .....	90
7.5.3 <i>Land Cover</i> .....	91
<b>7.6 Wind Input Data.....</b>	<b>91</b>
<b>7.7 Domain.....</b>	<b>92</b>
<b>7.8 Simulations.....</b>	<b>93</b>
 <b>CONCLUSIONS.....</b>	 <b>97</b>
<b>References.....</b>	<b>99</b>
<b>Annex 1: Discretization Schemes and Solvers.....</b>	<b>110</b>
<b>Annex 2: Canopy Tests.....</b>	<b>119</b>

<b>Annex 3: Validation.....</b>	<b>129</b>
<b>Annex 4: Study Case (Sierra Morena) .....</b>	<b>146</b>
<b>Annex 5: Wind Data.....</b>	<b>158</b>
<b>Annex 6: Topography data processing.....</b>	<b>160</b>

# FIGURES

1.1: Electricity generation in 2050 according to the Energy for All Scenario.....	21
1.2: Mexico electrification rates by state 2010 (INEGI) .....	23
1.3: Tehuantepec Isthmus Wind Atlas (NREL) .....	24
2.1: Wind speed profile .....	26
2.2: Turbulences length and time scales. ....	28
2.3: Turbulence models application range. ....	29
2.4: SKE Coefficients.....	32
2.5: Shear stresses in the near-wall region.....	34
2.6: Normalized wind speed in the near-wall region.....	35
2.7: Wind speed profile entering a forested canopy.....	38
3.1: Control volume. ....	41
3.2: First order upwind scheme. ....	43
3.3: Central differencing scheme.....	43
3.4: QUICK scheme.....	44
3.5: Segregated procedure for a pressure-based solver.....	46
4.1: OpenFOAM general structure.....	48
4.2: OpenFOAM case structure.....	48
4.3: Domain example.....	49
4.4: SHM mesh steps (a) blockMesh mesh (b) castellated mesh (c) snapped mesh.....	51
4.5: Atmospheric Boundary Layer conditions dictionary. ....	55
4.6: Fluid properties dictionary. ....	55
4.7: Wind speed profile entering a forested canopy. ....	56
4.8: Porosity properties dictionary. ....	58
4.9: groovyBC velocity entry at the inlet. ....	58
5.1: TKE profiles on plain terrain. ....	62
5.2: Test domain. ....	63
5.3: First node position simulations $U_x$ relative errors. ....	64
5.4: First node position simulations epsilon relative errors. ....	64
5.5: First node position simulations TKE relative errors. ....	65
5.6: First node position simulations TKE relative errors (details). ....	65

5.7: First node position simulations epsilon relative errors at 50m.....	66
5.8: First node position simulations TKE relative errors at 50m. ....	66
5.9: $z_p/z_0=5$ test TKE relative error, a) whole profile, b) near-ground detail.....	67
5.10: Vertical grading epsilon relative errors. . ....	68
5.11: Vertical grading TKE relative errors. . ....	68
5.12: Vertical grading $U_x$ relative errors. . ....	68
5.13: Aspect ratio simulations TKE relative errors. . ....	69
5.14: Aspect ratio simulations $U_x$ relative errors.....	70
5.15: Aspect ratio simulations epsilon relative errors .....	70
5.16: Sensors positions.....	71
5.17: Leaf area density: a) real b) modelled.....	71
5.18: Canopy model simulations domain and canopy region.....	72
5.19: Canopy model simulations domain.....	73
5.20: Canopy models normalized velocity profiles.....	73
5.21: Canopy models intensity profiles. . . ....	74
6.1: The Bolund Hill. . . ....	76
6.2: The Bolund Experiment a) instrumentation b) and mast position.....	77
6.3: Bolund Hill orography file.....	78
6.4: Blind Comparison simulations domain.....	79
6.5: Refinement above the Bolund Hill.....	79
6.6: Reference wind a) velocity b) and intensity profiles.....	80
6.7: Bolund Hill elevation profile along Line B (see Fig.6.2).....	81
6.8: Speed-up over the Bolund Hill along Line B at 2m a.g.l.....	81
6.9: Speed-up over the Bolund Hill along Line B at 5m a.g.l.....	81
6.10: Speed-up over the Bolund Hill along Line B at 9m a.g.l.....	82
6.11: Speed-up along Line B at M3 at 9m a.g.l.....	82
6.12: Speed-up along Line B at M3.....	83
6.13: TKE Variation over the Bolund Hill along Line B at 2m a.g.l.....	83
6.14: TKE Variation over the Bolund Hill along Line B at 5m a.g.l.....	84
6.15: TKE Variation over the Bolund Hill along Line B at 9m a.g.l.....	84
6.16: TKE Variation along Line B at M3.....	85



<b>7.1: Sierra Morena a) met.station location b) view from the met.station.....</b>	<b>87</b>
<b>7.2: Sierra Morena a) met.station b) monthly averaged temperature and wind speed values.</b>	<b>87</b>
<b>7.3: Sierra Morena met.station a) wind rose and b) Weibull distribution.....</b>	<b>88</b>
<b>7.4: Sierra Morena area contour curves map. ....</b>	<b>89</b>
<b>7.5: Sierra Morena area RIX map. ....</b>	<b>90</b>
<b>7.6: Sierra Morena area a) TRI and b) slopes maps. ....</b>	<b>90</b>
<b>7.7: Sierra Morena domain. ....</b>	<b>92</b>
<b>7.8: Sierra Morena domain land cover. ....</b>	<b>92</b>
<b>7.9: Sierra Morena grid a) general view in y direction, b) numeric characteristics c) near ground meshing and d) near-ground added layers.</b>	<b>93</b>
<b>7.10: Wind speed at inlet patch using a) the OF entries and b) the groovyBC tool.....</b>	<b>93</b>
<b>7.11: Wind speed inlet profile using the groovyBC tool. ....</b>	<b>94</b>
<b>7.12: Sierra Morena a) wind speed and b) wind power density maps. ....</b>	<b>94</b>
<b>7.13: Sierra Morena wind speed a) at 10m, b) 20m and c)30m height. ....</b>	<b>95</b>
<b>7.14: Sierra Morena wind power density a) at 10m, b) 20m and c)30m height. ....</b>	<b>96</b>

# TABLES

<b>1.1: Electrification Rates worldwide 2014. ....</b>	<b>20</b>
<b>4.1: Pressure boundary conditions. ....</b>	<b>52</b>
<b>4.2: Outlet patch boundary conditions. ....</b>	<b>53</b>
<b>4.3: Side patches boundary conditions. ....</b>	<b>53</b>
<b>4.4: Top patch boundary conditions. ....</b>	<b>53</b>
<b>4.5: Inlet patch boundary conditions. ....</b>	<b>54</b>
<b>4.6: Ground patch boundary conditions. ....</b>	<b>54</b>
<b>4.7: Internal field boundary conditions. ....</b>	<b>55</b>
<b>4.8: SKE coefficients. ....</b>	<b>56</b>
<b>4.9: Governing equation terms. ....</b>	<b>59</b>
<b>4.10: Variable solvers. ....</b>	<b>59</b>
<b>4.11: Relaxation factors. ....</b>	<b>60</b>
<b>5.1: Test simulations settings. ....</b>	<b>63</b>
<b>5.2: First node position simulations values. ....</b>	<b>64</b>
<b>5.3: Vertical grading simulations values. ....</b>	<b>67</b>
<b>5.4: Aspect ratio simulations settings. ....</b>	<b>69</b>
<b>5.5: Aspect ratio simulations values. ....</b>	<b>69</b>
<b>5.6: Canopy models coefficients. ....</b>	<b>72</b>
<b>5.7: Canopy model simulations settings. ....</b>	<b>73</b>
<b>6.1: Blind Comparison cases boundary conditions.....</b>	<b>77</b>
<b>6.2: Grid Independency simulations settings. ....</b>	<b>80</b>
<b>7.1: Land cover colour code and roughness length. ....</b>	<b>91</b>
<b>7.2: Sierra Morena simulations inlet profiles settings. ....</b>	<b>91</b>

# Nomenclature

## Roman Symbols

<b>A</b>	<b>[-]</b>	Solver's matrix (Eq.3.9)
<b>a<sub>E</sub></b>	<b>[-]</b>	Coefficient in Eq.3.3
<b>a<sub>N</sub></b>	<b>[-]</b>	Coefficient in Eq.3.3
<b>a<sub>nb</sub></b>	<b>[-]</b>	Coefficient in Eq.3.4
<b>a<sub>p</sub></b>	<b>[-]</b>	Coefficient in Eq.3.3
<b>a<sub>s</sub></b>	<b>[-]</b>	Coefficient in Eq.3.3
<b>a<sub>w</sub></b>	<b>[-]</b>	Coefficient in Eq.3.3
<b>B</b>	<b>[-]</b>	Solver's matrix (Eq.3.9)
<b>B</b>	<b>[-]</b>	Velocity profile constant in the logarithmic layer (Eq.2.32)
<b>b'</b>	<b>[Pa]</b>	Continuity imbalance coefficient (Eq.3.11)
<b>c</b>	<b>[-]</b>	Constant in Prandtl relation (Eq. 2.16)
<b>c'</b>	<b>[-]</b>	Constant in Eq.2.18
<b>C<sub>1</sub></b>	<b>[s<sup>-1</sup>]</b>	Momentum source term viscous coefficient (Eq.2.48)
<b>C<sub>2</sub></b>	<b>[m<sup>2</sup>m<sup>-3</sup>]</b>	Momentum source term inertial coefficient (Eq.2.48)
<b>C<sub>D</sub></b>	<b>[-]</b>	Drag coefficient
<b>C<sub>o</sub></b>	<b>[-]</b>	Courant number (Eq.4.15)
<b>C<sub>p</sub></b>	<b>[kgm<sup>2</sup>K<sup>-1</sup>s<sup>-2</sup>]</b>	Heat Capacity
<b>C<sub>p</sub></b>	<b>[-]</b>	Power coefficient (Eq.7.3)
<b>C<sub>s</sub></b>	<b>[-]</b>	Smagorinsky-Lilli constant (Eq.2.40)
<b>C<sub>ε1</sub></b>	<b>[-]</b>	k-epsilon model constant
<b>C<sub>ε2</sub></b>	<b>[-]</b>	k-epsilon model constant
<b>C<sub>μ</sub></b>	<b>[-]</b>	k-epsilon model constant
<b>d</b>	<b>[m]</b>	Displacement height (Eq.2.46)
<b>E</b>	<b>[m<sup>-2</sup>]</b>	Porosity coefficient (Eq.4.12)
<b>E</b>	<b>[-]</b>	Velocity profile constant in the logarithmic layer (Eq.2.32)
<b>e</b>	<b>[-]</b>	Errors (Eq.5.1)
<b>E<sub>ij</sub></b>	<b>[Pa]</b>	Mean deformation rate matrix <i>ij</i> component

$\mathbf{e}_{ij}$	[Pa]	Fluctuating term of the $ij$ component of the mean deformation rate matrix
$\mathbf{F}$	[m <sup>2</sup> m <sup>-3</sup> ]	Forchheimer tensor (Eq.4.12)
$\mathbf{f}$	[m <sup>-1</sup> ]	Porosity coefficient (Eq.4.12)
$\mathbf{f}$	[m <sup>-1</sup> ]	Coriolis parameter (Eq.7.2)
$\mathbf{g}$	[ms <sup>-2</sup> ]	Gravitation acceleration
$\mathbf{G}_k$	[m <sup>2</sup> s <sup>-3</sup> ]	Turbulent kinetic energy per unit of mass creation rate
$\mathbf{H}_{\text{ref}}$	[m]	Reference height
$\mathbf{I}$	[-]	Turbulent Intensity
$\mathbf{I}_M$	[-]	Modelled turbulent intensity (Eq.6.7)
$\mathbf{I}_s$	[-]	Simulated turbulent intensity (Eq.6.7)
$\mathbf{k}$	[m <sup>2</sup> s <sup>-2</sup> ]	Turbulent kinetic energy per unit of mass
$\mathbf{K}$	[m <sup>2</sup> ]	Intrinsic Permeability tensor of a porous media
$\mathbf{k}^+$	[-]	Near-wall turbulent kinetic energy per unit of mass
$\mathbf{k}_s$	[m]	Equivalent sand-grain roughness (Eq.2.39)
$\mathbf{k}_s^+$	[-]	Near-wall equivalent sand-grain roughness (Eq.2.40)
$\mathbf{L}$	[m]	Length scale (Eq.4.10)
$\mathbf{L}$	[m]	Terrain characteristic length scale (Eq.7.2)
$\mathbf{l}_m$	[m]	Mixing length
$\mathbf{L}_{MO}$	[m]	Monin-Obukhov Stability Length
$\mathbf{l}_s$	[m]	Length scale (Eq.2.18)
$\mathbf{M}$	[-]	Solver's matrix (Eq.3.9)
$\mathbf{N}$	[-]	Solver's matrix (Eq.3.9)
$\mathbf{nut}$	[m <sup>2</sup> s <sup>-2</sup> ]	Turbulent kinematic viscosity
$\mathbf{P}$	[Pa]	Pressure
$\mathbf{p}'$	[Pa]	Pressure correction (Eq.3.11)
$\mathbf{P}_0$	[Pa]	Standard Reference Pressure
$\mathbf{P}_k$	[Pa]	Turbulent kinetic energy creation rate
$\mathbf{Q}$	[-]	Solver's matrix (Eq.3.9)
$\mathbf{R}$	[J.mol <sup>-1</sup> .K <sup>-1</sup> ]	Gas constant

<b>R<sub>Asp</sub></b>	<b>[-]</b>	Aspect Ratio
<b>Re</b>	<b>[-]</b>	Reynolds number
<b>R<sub>exp</sub></b>	<b>[-]</b>	Vertical Cell-to-cell expansion ratio (5.1.2)
<b>R<sup>N</sup></b>	<b>[?]</b>	Overall residual after “N” iterations (Eq.3.13)
<b>Ro</b>	<b>[-]</b>	Rossby number (Eq.7.2)
<b>R<sub>p</sub><sup>N</sup></b>	<b>[?]</b>	Residual at point “P” after “N” iterations (Eq.3.13)
<b>R<sub>s</sub></b>	<b>[-]</b>	Speed-Up error (Eq.6.6)
<b>R<sub>TKE</sub></b>	<b>[-]</b>	Turbulent Kinetic Energy error (Eq.6.7)
<b>S<sub>k</sub></b>	<b>[kgms<sup>-3</sup>]</b>	Turbulent kinetic energy per unit of mass source term (Eq.2.49)
<b>S<sub>u</sub></b>	<b>[kgm<sup>-2</sup>s<sup>-2</sup>]</b>	Momentum source term (Eq.2.48)
<b>S<sub>ε</sub></b>	<b>[ms<sup>-1</sup>]</b>	Turbulent kinetic energy dissipation rate source term (Eq.250)
<b>S<sub>Φ</sub></b>	<b>[?]</b>	Φ source term (Eq.3.1)
<b>u</b>	<b>[ms<sup>-1</sup>]</b>	Velocity or x-velocity component
<b>U</b>	<b>[ms<sup>-1</sup>]</b>	Mean velocity/ Mean x-velocity term
<b>u*,u<sub>τ</sub>,u<sub>f</sub></b>	<b>[ms<sup>-1</sup>]</b>	Friction velocity
<b>u<sup>′</sup></b>	<b>[ms<sup>-1</sup>]</b>	Velocity fluctuating term/ x-velocity fluctuating term
<b>u<sub>ABL</sub></b>	<b>[ms<sup>-1</sup>]</b>	Atmospheric Boundary Layer friction velocity
<b>u<sub>p</sub></b>	<b>[ms<sup>-1</sup>]</b>	Velocity at p-point (Eq.2.34)
<b>U<sub>REF</sub></b>	<b>[ms<sup>-1</sup>]</b>	Reference velocity
<b>U<sub>x</sub></b>	<b>[ms<sup>-1</sup>]</b>	Mean x-velocity term
<b>V</b>	<b>[ms<sup>-1</sup>]</b>	Mean y-velocity term
<b>v, v<sup>′</sup></b>	<b>[ms<sup>-1</sup>]</b>	y-velocity component, v-velocity component fluctuating term
<b>v<sub>s</sub></b>	<b>[ms<sup>-1</sup>]</b>	Velocity scale
<b>W</b>	<b>[ms<sup>-1</sup>]</b>	Mean z-velocity term
<b>W</b>	<b>[W/m<sup>2</sup>]</b>	Wind power density (Eq.7.3)
<b>w, w<sup>′</sup></b>	<b>[ms<sup>-1</sup>]</b>	z-velocity component, z-velocity component fluctuating term
<b>y<sup>+</sup></b>	<b>[-]</b>	Near-wall region distance
<b>y<sub>0</sub></b>	<b>[m]</b>	Aerodynamic roughness length (Eq.2.44)
<b>y<sub>p</sub></b>	<b>[m]</b>	Distance of p point above ground (Eq.2.34)
<b>z</b>	<b>[m]</b>	Height above terrain

$z_0$	[m]	Aerodynamic roughness length
$z_{\text{ground}}/z_g$	[m]	Ground height (Eq.4.1)
$z_p$	[m]	Distance of p point above ground
$z_{\text{ref}}$	[m]	Reference mast height (Ch.6)

## Greek Symbols

$\alpha$	[m <sup>2</sup> m <sup>-3</sup> ]	Leaf area
$\alpha$	[-]	Relaxation factor (Eq.3.12)
$\alpha$	[-]	Factor (Eq.5.2)
$\beta$	[-]	Porosity (Eq.2.47)
$\beta_d$	[-]	Fraction of the mean flow's kinetic energy dissipated rapidly after the wake (Eq.2.50)
$\beta_p$	[-]	Fraction of the mean flow's kinetic energy converted to TKE due to the canopy (Eq.2.49)
$\Gamma$	[m <sup>2</sup> s <sup>-1</sup> ]	Diffusion coefficient
$\delta$	[m]	Near-wall boundary layer thickness
$\Delta k_m$	[-]	Measured TKE increase (Eq.6.4)
$\Delta k_s$	[-]	Simulated TKE increase (Eq.6.2)
$\Delta S_M$	[-]	Measured Normalised Speed Up (Eq.6.3)
$\Delta S_s$	[-]	Simulated Normalised Speed Up (Eq.6.1)
$\Delta t$	[s]	Time step (Eq.4.15)
$\Delta x, \delta x$	[m]	Cell length (Eq.4.15)
$\delta y, \delta z$	[m]	Cell's measures (Eq.5.3)
$\delta \vartheta$	[m]	Near-wall inner boundary layer thickness/viscous length scale
$\varepsilon$	[m <sup>2</sup> s <sup>-3</sup> ]	TKE dissipation rate
$\varepsilon^+$	[-]	Near-wall TKE dissipation rate
$\Theta$	[K]	Potential temperature
$\vartheta$	[m <sup>2</sup> s <sup>-2</sup> ]	Kinematic viscosity
$\vartheta_t$	[m <sup>2</sup> s <sup>-2</sup> ]	Turbulent kinematic viscosity
$\vartheta_t^+$	[-]	Near-wall turbulent kinematic viscosity

$\kappa$	[-]	Karman constant
$\mu$	[Pa.s]	Dynamic viscosity
$\mu_{\text{effective}}$	[Pa.s]	Dynamic viscosity within the near-wall boundary layer (Eq.2.34)
$\mu_t$	[Pa.s]	Dynamic turbulent viscosity
$\rho$	[kgm <sup>-3</sup> ]	Density
$\sigma_k$	[-]	k-epsilon model constant
$\sigma_\epsilon$	[-]	k-epsilon model constant
$\tau$	[Pa]	Stress tensor
$\tau_{ij}$	[Pa]	Stress tensor <i>ij</i> component
$\tau_w$	[Pa]	Wall shear stress
$\Phi$	[?]	Fluid property (Eq.3.1)
$\Phi_{\text{(inlet)}}$	[?]	Fluid property at the inlet (Eq.5.1)
$\Phi_E$	[?]	Fluid property on point “E” (Eq.3.6)
$\Phi_e$	[?]	Fluid property on point “e” (Eq.3.6)
$\Phi_{\text{Face}}$	[?]	Fluid property on point a face (Eq.3.7)
$\Phi_{\text{nb}}$	[?]	Fluid property in neighbour cell (Eq.3.4)
$\Phi_P$	[?]	Fluid property on point “P” (Eq.3.6)
$\Phi_P$	[?]	Fluid property on point “P” (Eq.3.6)
$\Phi_P^n$	[?]	Fluid property on point “P” after “n” iterations (Eq.3.12)
$\Phi_P^{n-1}$	[?]	Fluid property on point “P” after “n-1” iterations (Eq.3.12)
$\Phi_W$	[?]	Fluid property on point “W” (Eq.3.67)
$\Phi_x$	[?]	Fluid property at a distance “x” from the inlet (Eq.5.1)
$\psi$	[-]	Atmospheric stability function

## Abbreviations

<b>ABL</b>	Atmospheric Boundary Layer
<b>BC</b>	Boundary Condition
<b>BM</b>	BlockMesh
<b>BZ-Model</b>	Bessel Expansion on a Zooming Grid-Model
<b>CDS</b>	Central Differencing
<b>CFD</b>	Computational Fluid Dynamics
<b>CONAGUA</b>	Comisión Nacional del Agua
<b>CP</b>	Centre Point
<b>CV</b>	Cell volume
<b>DNS</b>	Direct Numeric Simulations
<b>FDM</b>	Finite Difference Method
<b>FEM</b>	Finite Element Method
<b>FOU</b>	First Order Upwind
<b>FVM</b>	Finite Volume Method
<b>GAMG</b>	Geometric Algebraic Multigrid
<b>HH</b>	Horizontal Homogeneity
<b>HHTBL</b>	Horizontally Homogeneous Turbulent Boundary Layer
<b>IEA</b>	International Energy Agency
<b>INEGI</b>	Instituto Nacional de Estadística y Geografía.
<b>INIFAP</b>	Instituto Nacional de Investigaciones Forestales Agrícolas y Pecuarias.
<b>KAMM</b>	Karlsruhe Atmospheric Mesoscale Model
<b>LAD</b>	Leaf Area Density [ $\text{m}^2/\text{m}^3$ ]
<b>LAI</b>	Leaf Area Index [ $\text{m}^2/\text{m}^2$ ]
<b>LES</b>	Large Eddy Simulations
<b>Nb</b>	Neighbour cell
<b>NREL</b>	National Renewable Energy Laboratory
<b>N-S</b>	Navier-Stokes Equations
<b>OF</b>	OpenFOAM



<b>OMWC</b>	Observed Mean Wind Climate
<b>QUICK</b>	Quadratic Upwind Interpolation for Convective Kinetics
<b>RANS</b>	Reynolds Averaged Navier-Stokes
<b>RE</b>	Renewable Energy
<b>RIX</b>	Ruggedness Index
<b>SHM</b>	SnappyHexMesh
<b>SIMPLE</b>	Semi-Implicit Method for Pressure-Linked Equations
<b>SKE</b>	Standard k-epsilon model
<b>SOU</b>	Second Order Upwind
<b>SWT</b>	Small Wind Turbines
<b>TI</b>	Turbulent Intensity
<b>TKE</b>	Turbulent kinetic energy per unit of mass
<b>TRI</b>	Terrain Ruggedness Index
<b>WAsP</b>	Wind Atlas Analysis and Application Program
<b>WRF</b>	Weather Research and Forecasting

# Acknowledgements

To all the people, both at the TUB and at the UNAM, who helped me to carry out this work. Especially to Dr.Ing.César Angeles Camacho and Moritz Mühlbauer.

To all the people who support me during the data collection phase in Chiapas. Here I would like to remember all the people from the *Centro de Derechos Humanos Digna Ochoa* in Tonalá, with an especial mention to Nathaniel, the workers from *Reserva de la Biosfera La Sepultura* and the staff from *Junax*.

To my flatmates in Berlin.

To all my friends and relatives, especially to Andy and Lucia.

To my brother and my parents.

# Abstract

Der Zugang zur Energieversorgung wird weitgehend als wichtigster Motor für wirtschaftliche Entwicklung angesehen und ist für die meisten Menschen eine der Grundstützen zur Verbesserung ihrer Lebensqualität. Dennoch hatten im Jahr 2014 etwa 1.3 Milliarden Menschen keinen Zugang zu Elektrizität. Die meisten von ihnen leben in abgelegenen ländlichen Gegenden sogenannter 'Entwicklungsländer'. Also in Gegenden, wo es besonders schwierig ist, aufgrund der Beschaffenheit des Geländes und der großen Entfernungen, Verteilungsnetze auszubauen die eine Stromversorgung gewährleisten.

Netzferne Systeme, die auf regenerativen Energien basieren, wurden daher wiederholt als ein möglicher Lösungsansatz angebracht. Jedoch erfordert die Nutzung erneuerbarer Energien auf diese Weise neue Innovationen. Im Fall von Windenergie, stellen turbulente Phänomene, die durch komplexe Gelände hervorgerufen werden, eine große Herausforderung an die konventionellen Windressourcen Bewertung Softwares dar. Gleiches gilt für die Entwicklung von Windenergieprojekten in diesen Gegenden.

*Computational Fluid Dynamics* (CFD) könnten eine gute Alternative sein, um solche Probleme zu behandeln und eine genaue Bewertung der Windressourcen auf komplexen Terrain durchzuführen. Allerdings gibt es zur Anwendung von CFD in diesem Bereich noch keine klaren Richtlinien.

In der vorliegenden Arbeit wurden die Möglichkeiten, die die Open-Source CFD Software OpenFOAM (OF) bietet, in Betracht gezogen, um ein *micro-scale* Modell für die Simulation von Wind über komplexe Gelände innerhalb der atmosphärischen Grenzschicht (Atmospheric Boundary Layer - ABL) aufzubauen. Dafür wurden zuerst die Randbedingungen festgelegt und eine Turbulenz Modell gewählt, das SKE, während die geeignetsten OF Utilitys bestimmt wurden. Weiterhin wurden auf einer ebenen Fläche Tests durchgeführt, um möglichst realistische Werte für einige Simulationsparameter zu haben. Zudem wurden verschiedene Modelle getestet, um Fälle von waldbedeckten Gebiete bis hin zu porösen Volumina abzudecken. Das daraus resultierende Modell wurde anhand experimentell bestimmter Messwerte, nach der *Bolund Hill Blind Comparison*, überprüft. Dies zeigte, dass das Modell das generelle Verhalten von Wind, der sich über einen Hügel bewegt, korrekt vorhersagen kann. Die Genauigkeit der Vorhersagen ist jedoch stark von den verschiedenen Gegebenheiten abhängig und variiert sehr stark, oftmals auch über die zulässigen Grenzen hinaus, die für ein Windenergie Projekt nötig wären. Am Ende wurde das Modell in der Realität angewandt, in einer ländlichen Gemeinde an der Küste von Chiapas, um eine Bewertung des dort bestehende Windressourcen zu machen.

**Kennwörter:** Zugang zur Energieversorgung, Netzferne Systeme, Erneuerbare Energien Systeme, Windenergie, Windressourcen Bewertung, *Computational Fluid Dynamics*, OpenFOAM, Turbulenzmodellierung, Komplexe Gelände.

# Abstract

Access to energy is widely recognised as a major driver for economic development and one of the keystones for the improvement of people's quality of life. However, in 2014 there was still around 1.3 billion people without access to electricity. Most of them live in remote rural areas in developing countries where, terrain features and long distances make the supply of electricity through the expansion of the distribution grids very difficult. Renewable energy-based off-grid systems have been repeatedly pointed out as a possible solution. However, this application of renewable energy technologies requires new tools. In the case of wind energy, the turbulent phenomena that complex terrain make arise, pose a challenge to the traditional wind assessment software and therefore to the development of wind energy projects on these locations.

Computational Fluid Dynamics (CFD) may be a good alternative to overcome these problems and provide an accurate wind assessment on complex terrain although there are not clear guidelines for its use in that field. In the present work, the different options offered by the open source CFD software OpenFOAM (OF) were considered and a set of them were used to build a micro-scale model for wind flow on complex terrain. Firstly, after an extensive literature review, the standard  $k-\epsilon$  model, a RANS turbulence model, and the boundary conditions for the simulation of wind flow in the Atmospheric Boundary Layer (ABL) were chosen. Then, the most appropriate OF utilities for this purpose were identified and test were made on flat and homogeneous surfaces to find the most appropriate values of a few simulation parameters. Different models to represent forested canopies through porous volumes were also tested.

The resulting model was validated against experimental measurements following the *Bolund Hill Blind Comparison*. The results showed that, although the model was able to predict the general behaviour of the wind flow over the hill correctly, the accuracy of the predictions varies greatly with the location, in many cases beyond the acceptable limits for a wind energy project. Finally, the model was applied on a real case, to assess wind resource on a rural community on the coast of Chiapas, Southern Mexico.

**Key Terms:** Electricity access, Renewable Off-grid systems, wind energy, wind resource assessment, Computational Fluid Dynamics, OpenFOAM, turbulence modelling, complex terrain.

# Chapter 1

## Introduction

### 1.1 Energy Access and Electrification

Access to modern energy services<sup>1</sup> has been recognized by governments, international institutions and NGOs worldwide, as a fundamental keystone for the development of any community and the improvement of people's quality of life [1]. Consequently, electrification programs at different levels have been launched [2].

However, in 2014 there were still almost 1.3 billion people with no access to electricity worldwide, mostly in developing or emerging countries in Africa and Asia, but also, to a lesser degree, in Latin America and the Middle East.

Region	Population without electricity (millions)	Electrification rate %	Urban Electrification Rate %	Rural Electrification Rate
Africa	622	43%	68%	26%
Asia	620	83%	95%	74%
Latin America	23	95%	99%	82%
Middle East	18	92%	98%	78%
Transition economies & OECD	1	100%	100%	100%
<b>WORLD</b>	<b>1.285</b>	<b>82%</b>	<b>94%</b>	<b>68%</b>

Table 1.1: Electrification rates worldwide [3].

The bulk of them live in rural communities in remote and scarcely populated areas where topographic obstacles make the extension of the national grid technically difficult, costly<sup>2</sup> and inefficient in many cases. This has led to a growing interest in a decentralised approach to electricity generation based on off-grid schemes like small and local micro-grids or stand-alone systems as a more feasible and affordable solution for such communities [2,4,5,6]. Indeed, if universal electrification is to be achieved by 2030, up to a 54% of the world electricity by that time would have to be provided by decentralized power systems or off-grid systems.

---

1. The IEA's energy access definition can be found in [7] ( pag. 12).

2. E.g. In rural areas, far from the grid in India electrification can be from seven up to ten times more expensive than urban electrification due to higher transmission and distribution costs and transmission losses [8].

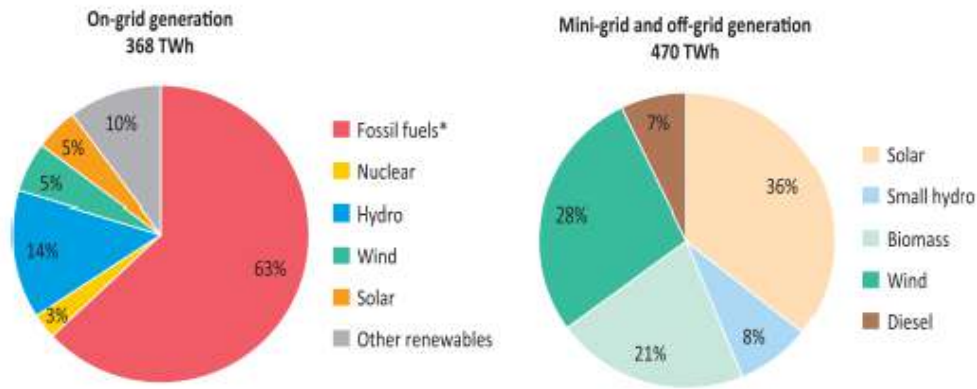


Figure 1.1: Electricity generation by source in 2050 according to [7].

Due to their steady development and the reduction of the production costs ([9] page-417/10.4, 333/8.10 & 430/10.15. [10]), renewable energy has been pointed out as a suitable and cost-effective alternative for off-grid systems, especially in rural areas, for several reasons [11,12,13], such as:

- Size and capacity flexibility
- Supply independency
- Location flexibility

Many feasibility studies on renewable energy-based off-grid systems (RE off-grid) have been made [14,2,15] and implementation examples can be found all over the world [16, 17, 18, 19]. They are also being extensively used in national electrification programs worldwide [20].

## 1.2 Wind Energy

The election of a certain technology for a RE-off-grid system depends on the resource availability, which must be previously assessed. Unlike solar radiation, which is present virtually everywhere, wind resource is strongly influenced by terrain topography, which may cause turbulence and frequent direction changes, especially in the kind of complex terrains typical of most of the communities lacking electricity access. Therefore, a proper wind resource assessment is of the utmost importance for the installation of wind energy off-grid systems in such locations [11,12].

### 1.2.1 Complex Terrain

Although there is no wide accepted definition for terrain complexity [21], in [22] a set of topographic features, according to which it might be established, are given:

- Roughness→ High and non-uniform roughness lengths values
- Obstacles→ Frequent obstacles (e.g. forest, small buildings, crops etc)
- Orography→ High ruggedness index

Indeed, different ruggedness indices calculated out of the terrain steepness, are used by well-established wind assessment software like WAsP to evaluate the terrain complexity and consequently as a performance indicator. [22, 23].

### 1.2.2 Wind Resource Assessment

For wind energy purposes, the models used for wind assessment can be roughly divided into two groups [24]:

- **Meso-scale models:** applied to estimate the overall wind resource over a whole region (1-20km), with resolutions of around 2.5km. They usually combine long-term meteorological data, national wind atlases and, if available, on-site measurements (e.g.KAMM [25], WRF[26]).
- **Micro-scale models:** used to predict wind patterns at specific location (0.1-1km) for siting purposes or calculate power output from wind turbines based on measurements from nearby meteorological stations with higher resolution.

The estimation of wind resource for the installation of wind-energy-based off-grid systems at a certain location would fall within the micro-scale models.

### 1.2.3 Wind Assessment Software

The use of computer models for the estimation of wind resource at a micro-scale level for wind farms has become a standard within the wind industry since the first wind programs for this purpose were developed in the 80s [27]. These early wind assessment software, such as WAsP [28], mostly relied on linear BZ-Models [139] based on a simplified version of the fluid flows equations and, although they yield good results on flat or moderately hilly terrain, they fail to correctly predict phenomenon such as flow detachment and recirculation which arise on complex terrain. These limitations together with the increasingly importance of wind energy and the decreasing availability of suitable sites in simple terrain, such as open spaces or low hills, have led to a growing demand of new wind assessment methods capable of more accurate predictions in order to assure the profitability of the investments and reduce uncertainty costs.

These new needs of the wind industry have boosted advances in the application of micro-scale models based on non-lineal numerical methods, like *Computational Fluid Dynamics* (CFD). So far many of the results of different CFD microscale models in complex terrain have been successfully validated against empirical data and proven more accurate than those of the linear models [29, 30]. In the last years, this development has reached the wind assessment industry and several CFD-based wind assessment software, like *WindSim* and *MeteoDyn*, have appeared and well-established software, like WAsP have included CFD tools in their new editions [31].

However, the use of CFD techniques for wind assessment still faces some problems. Their development is limited by the computing power available which for instance hinders the inclusion of meteorological models due to their complexity and the exploration of more demanding CFD techniques for wind flow simulation, such as large-eddy simulation (LES), which might render better results [32]. In addition to this, there is no agreement on a standard methodology for the assessment of wind resources on complex terrain for its application for wind energy purposes [30].

### 1.3 Rural Electrification México

According to the *Instituto Nacional de Estadística y Geografía*, INEGI 2010 Census [33], 3,5% of the Mexican population, around 3 million people, had no access to electricity in 2010. They are mostly concentrated in rural areas, especially in the southern States [34] such as Chiapas, which is located at the border with Guatemala, with a large rural population, a low average income, ca. 34316 pesos/year [35], where the electrification rate was around 95.9% in 2010, lower than the national average [33] (see Fig.1.2). Moreover, even where there is electricity, the supply isn't reliable and can be very expensive, to the extent that, in 2010, electricity and fuel conformed the bulk (ca.87%) of the average household expenditures [36].

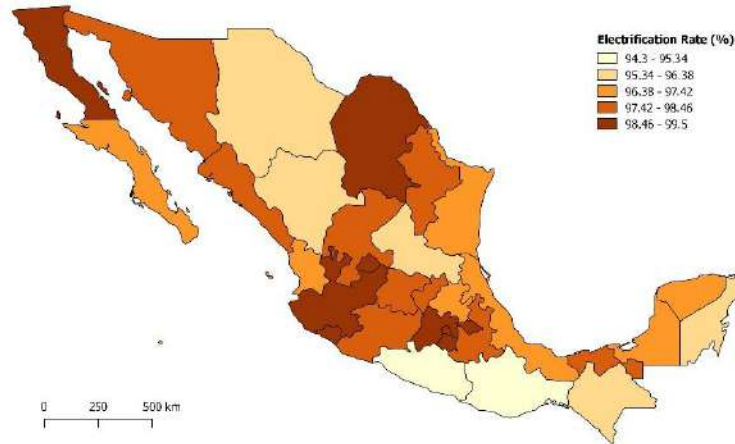


Figure 1.2: Mexico electrification rates by state 2010 [33]

These circumstances make Chiapas a good candidate for the installation of RE-Off-grid systems for rural electrification [37]. Indeed, several electrification plans based on renewable energy have been carried out in the country since the early 90s, although many of the first systems were poorly installed and operated and stopped working after a few years [38, 39].

### 1.4 Wind Resource Chiapas

The Tehuantepec Isthmus region in southern Mexico stretches across the border between the states of Oaxaca and Chiapas and is one of the areas with the highest wind energy potential in the country [40, 41] (see Fig.1.3). In 2012 there were 1250 MW wind power capacity installed in the area [42].

In the Tehuantepec Isthmus, the stronger winds blow mainly from October to February in the northeast-north direction. Therefore, it is more likely to find a good wind resource where the terrain features channel or speed up the winds from the northeast, especially at the ridges and hills with a height up to 1000m close to the coast. The *Sierra Madre de Chiapas* valleys close to Oaxaca are also mentioned as areas with potentially good wind resource.



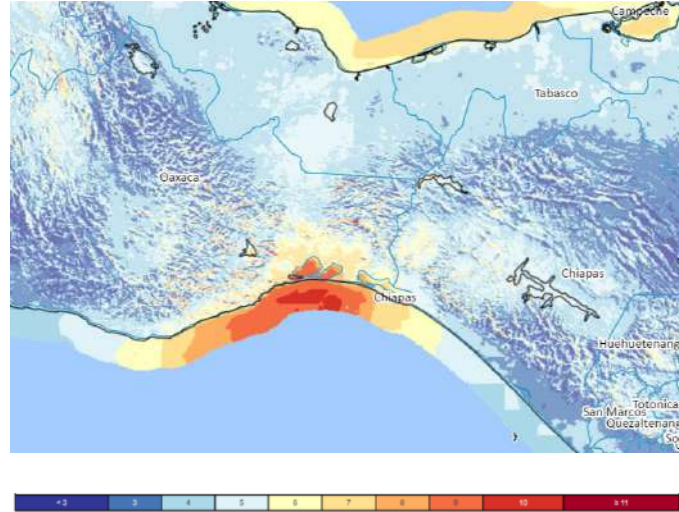


Figure 1.3: Tehuantepec Isthmus Wind Atlas (NREL)[40].

## 1.5 Objectives

The objective of the present work is the construction of a micro-scale CFD model for wind resource assessment in complex terrain, its validation against experimental measurements and its application to the assessment of wind resource in a rural community on the coast of Chiapas. A secondary objective is the test of a few of the model and simulation design features, which may be especially important for wind simulation over complex terrain, and thus provide guidelines and, to some degree, a methodology for this kind of simulations.

The CFD code chosen was the free and open-source software *OpenFOAM 2.3* (OF) in its version for Windows, which is written in C++. Unlike most of the commercial software's CFD tools, OF is not specifically designed for wind flow simulations. This means that it can be freely manipulated to create models for each study case, which makes it especially attractive for research purposes. Besides it is a free open source code, which means no licence is required to use it and therefore is accessible to anybody. However, for the same reason, not everything is well documented, and an extensive literature research was required.

Finally, it is not within the scope of the present work to change the turbulence model or to develop new utilities for OF but to use the model as it is and apply the utilities currently provided by OF.

## 1.6 Motivation

RE off-grid systems have been pointed out as a possible solution to the current lack of reliable electricity supply in many communities worldwide, especially in rural areas. The resource availability is essential for the choice of the technology to apply. For wind energy, the strong dependency between wind resource and terrain topography makes an accurate resource assessment even more important, especially on complex terrain. However, the traditional wind resource assessment techniques and tools, such as wind maps and linear equations-based assessment software, are focused on big wind parks on flat terrain, and are inappropriate for wind assessment over complex terrain at low heights, as it would be required for wind energy-based rural electrification projects.

A CFD-based micro-scale model may render better results in these cases. However, this is still a relatively new field, and, except for the commercial software's CFD tools, which doesn't allow a great deal of manipulation, no clear guidelines or methodologies have established.

## 1.7 Methodology

The methodology followed can be summarized as follows:

- A literature research on the general functioning and main features of the *finite volume method* (FVM) in CFD and in turbulent flow modelling was made and a turbulence model, the standard k-epsilon model, (SKE) was chosen as well as treatments for the solid walls and canopies.
- The OF utilities and tools which better suit the purpose of the present work were chosen
- Model and simulation design features, such as the vertical grading and the porosity coefficient were tested.
- The model was applied to simulate the wind flow above the Bolund Hill and the results yielded were compared with the experimental measurements from the *Bolund Blind Comparison* [28].
- Wind flow was simulated over the area surrounding a rural community on the coast of Chiapas. The results were used to assess wind resource in the area.

# Chapter 2

## Theoretical Background

In this section turbulence modelling within the atmospheric boundary layer is briefly introduced and the standard k-epsilon model (SKE) is presented, together with the treatment applied to solid boundaries and the canopy model.

### 2.1 Atmospheric Boundary Layer (ABL)

The *Atmospheric Boundary Layer (ABL)* is defined as the lowest layer of the atmosphere which is in contact with the earth's surface and, therefore, is influenced by the friction at the ground and by Earth's surface features, such as its orography and topography as well as by pressure and temperature gradients. This leads to a non-uniform wind velocity distribution across the layer, highly dependent on the height (see Fig.2.1). Above the ABL, is the so called *free* atmosphere, where earth's surface influence is less noticeable or inexistent and wind flows are only affected by pressure gradients and Coriolis forces. These are the so called *geostrophic winds* [43].

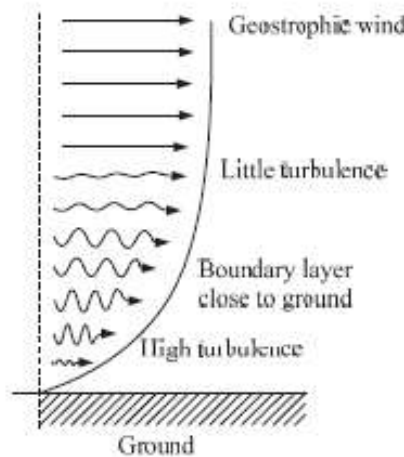


Figure 2.1: Wind speed profile [44].

The depth of the ABL varies in time and space with terrain orography, heat fluxes, evaporation or the day time, its upper limit ranging from 500m to around 3km above the Earth's surface [43]. The ABL itself is usually divided into several layers, being the *Surface Layer* the lowest one, comprising approximately the 10% of the ABL closest to the Earth's surface [44,45]. There the influence of terrain characteristics and temperature gradients is maximal, whereas that of the Coriolis forces is hardly noticeable.

#### 2.1.1 Stability and Stratification

The interaction with the Earth's surface leads to frequent changes in the direction and velocity of wind flows within the ABL and thus to the appearance of turbulence phenomena which makes it more instable than the upper layers [43].

The stability strongly depends on the vertical temperature profile which leads to the so called thermal stratification. The stratification of the atmosphere at a certain location is given by the *potential temperature*:

$$\theta = T \left( \frac{P_0}{P} \right)^{\frac{R}{c_p}} \quad (2.1)$$

- **Stable Atmosphere** ( $\frac{\partial \theta}{\partial z} > 0$ ): Heavier and cooler masses of air lay close to the ground and the warmer ones are above them. Typically, at night.
- **Unstable Atmosphere** ( $\frac{\partial \theta}{\partial z} < 0$ ): Warm and light air masses are close to the ground while cool and heavy are located at higher heights. This may lead to turbulent phenomena as the warm air masses tend to move upwards and the cool downwards, typically during the day time
- **Neutrally Stable** ( $\frac{\partial \theta}{\partial z} = 0$ ): the ABL isn't stratified, there are not great temperature differences between neighbouring air layers and density can be considered constant.

Solving flow governing equations become simpler if the thermal effects are neglected and the turbulence phenomena can be considered entirely of mechanical origin. However, this can only be made under the assumption of a neutrally stable atmosphere.

### 2.1.2 Velocity Vertical Profile

For a neutrally stable atmosphere above an unlimited and homogenous plain terrain, the mean vertical wind profile, based on the Prandtl boundary physics is given by the *logarithmic wind profile* [43]:

$$u(z) = \frac{u_{ABL}}{\kappa} \ln \left( \frac{z + z_0}{z_0} \right) \quad (2.2)$$

Here  $\kappa$  is the Von Karman constant and  $u_{ABL}$  is the ABL friction velocity. The wind velocity profile range between the free stream velocity at the upper limit of the ABL and 0 at the ground. The parameter  $z_0$  is the *aerodynamic roughness length*, which accounts for the effects of the land cover on the wind flow. Roughness length values are obtained empirically by fitting measured wind speed at different heights on the same spot to Eq.2.2 [28]. In a non-neutral atmosphere, an extra term must be introduced in Eq.2.2 to account for the thermal effects, such as heat and mass transfer, caused by the new temperature profile.

$$u(z) = \frac{u_{ABL}}{\kappa} \ln \left( \frac{z}{z_0} - \Psi \left( \frac{z}{L_{MO}} \right) \right) \quad (2.3)$$

$\Psi$  function depends on height  $z$  and  $L_{MO}$ , the so called *Monin-Obukhov Stability Length* which depends on the potential temperature  $\theta$  and describes the vertical mass exchange. In a neutral atmosphere, the condition  $[1/L_{MO}] < 0.04 \text{m}^{-1}$  is fulfilled [46].

## 2.2 Turbulence

Turbulence is a very complex phenomenon, whose effects are very difficult to describe accurately due to their wide range of length and time scales, which goes from a few millimetres to several kilometres and from one second to several days respectively (see Fig.2.2). It may be caused by ascending masses of warm air, the frictional drag over the ground, obstacles etc [43,47].

	Time scale					
	1 month	1 day	1 hour	1 minute	1 second	
	Weather systems, tropical cyclones					Meso Scale
250 km		Land-sea breeze, cloud cluster, thermal effects				
25 km			Thunder storms, micro climate, town			
2.5 km			Convection, tornadoes			Micro Scale
250 m				Dust devil, thermal		
25 m						
2.5 m						
25 cm						
2.5 cm						
2.5 mm						

Figure 2.2: Turbulences lengths and time scales [44].

### 2.2.1 The Eddies

These are the rotational flow structures characteristic of the turbulent flows. They are responsible for the high mixing rate inside the flow and of the mass, momentum and energy diffusion phenomena. Large eddies extract their energy from the mean flow, whose length and velocity scales they share. Provided a steady energy supply from the mean flow, as the viscous forces are negligible compared the inertial ones, eddies conserve their angular momentum while they increase their energy leading to a higher rotation rate and consequently to a smaller cross-section resulting in its breaking up into smaller eddies which undergo a similar process, setting up an *energy cascade*, between the mean flow and ever smaller eddies until the viscous effects cannot be neglected any longer and the eddies's energy is dissipated leading to the characteristic energy losses associated with turbulent flows. This occurs at the so-called *Kolmogorov microscale*.

### 2.2.2 Turbulence Modelling

A direct solution of the Navier-Stokes governing equations (N-S) (Eq.2.4,2.5,2.6,2.7) cannot be achieved for turbulent flows except in a few simple cases. For this reason, different procedures have been developed.

$$\text{div}(\mathbf{u}) = 0 \quad (2.4)$$

$$\frac{d(\rho u)}{dt} + \text{div}(\rho u \mathbf{u}) = -\frac{\partial p}{\partial x} + \text{div}(\mu \text{grad} u) \quad (2.5)$$

$$\frac{d(\rho v)}{dt} + \text{div}(\rho v \mathbf{v}) = -\frac{\partial p}{\partial y} + \text{div}(\mu \text{grad} v) \quad (2.6)$$

$$\frac{d(\rho w)}{dt} + \text{div}(\rho w \mathbf{w}) = -\frac{\partial p}{\partial z} + \text{div}(\mu \text{grad} w) \quad (2.7)$$

The use of one or other method depends on the input data, the scale and nature of the flow to be simulated (see Fig.2.3). These are a few of the most used and well-known models for the solution of turbulent flows:

- **Linear Models:** Based on the linearization of the N-S equations. The accuracy of their results decrease as the terrain complexity grows. It has been the model traditionally used by most of the commercial wind assessment programs like WAsP.
- **Large Eddy Simulation (LES):** This approach [48,49] is based on the spatially filtering of the flow equations to remove small scale eddies using a coarse grid, resolving only the large ones. The effects of the smaller eddies are later introduced through stresses in the momentum equations, boundary terms or modelled using other techniques [43]. The use of LES techniques for the solution of wind flows is however not feasible yet, although this might change soon thanks to the increasing computational power available [50,51,52]. According to [53,54] a hybrid LES-RANS scheme might become the predominant method for turbulent flow solution in a few years.
- **Direct Numerical Simulation (DNS):** The direct solution of the N-S equations requires a very fine grid to capture all the relevant scales in the flow, and a time-dependent solution over a period of time large enough to produce stable time averages of the flow variables. For these reasons, it demands high computational power and is only applied for low Reynolds numbers [55,56, 57, 58].

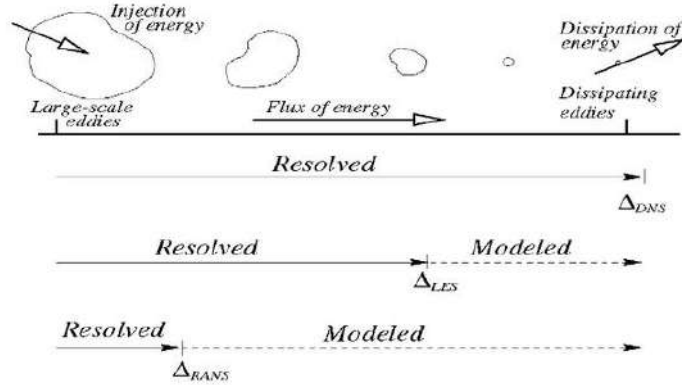


Figure 2.3: Turbulence models application range [59].

### 2.2.3 RANS Models

The basic idea behind non-linear RANS models is the decomposition of the flow variables into a mean and a fluctuating term to tackle the inherent chaotic nature of the turbulent flows. This is the so-called *Reynolds Decomposition*. For a variable  $\phi$ :

$$\phi(t) = \phi + \phi'(t) \quad \phi = \frac{1}{\Delta t} \int_0^{\Delta t} \phi(t) dt \quad (2.8)$$

Where  $\phi$  is the mean value of the flow property  $\phi$  over a time interval  $\Delta t$  and  $\phi'(t)$  the fluctuating term. If this decomposition is applied to the flow variables (e.g.  $\mathbf{u} = \mathbf{U} + \mathbf{u}'$ ) and then introduced into the N-S equations the result are the *Reynolds-Averaged Navier-Stokes equations* or *RANS* (Eq.2.9,2.10,2.11,2.12).

$$\text{div}(\mathbf{U}) = 0 \quad (2.9)$$

$$\frac{d(\rho U)}{dt} + \text{div}(\rho U \mathbf{U}) = -\frac{\partial p}{\partial x} + \text{div}(\mu \text{grad} U) + \left[ -\frac{\partial \overline{u'^2}}{\partial x} - \frac{\partial \overline{u'v'}}{\partial x} - \frac{\partial \overline{u'w'}}{\partial x} \right] \quad (2.10)$$

$$\frac{d(\rho V)}{dt} + \text{div}(\rho V \mathbf{U}) = -\frac{\partial p}{\partial x} + \text{div}(\mu \text{grad} V) + \left[ -\frac{\partial \overline{v'^2}}{\partial x} - \frac{\partial \overline{u'v'}}{\partial x} - \frac{\partial \overline{v'w'}}{\partial x} \right] \quad (2.11)$$

$$\frac{d(\rho W)}{dt} + \text{div}(\rho W \mathbf{U}) = -\frac{\partial p}{\partial x} + \text{div}(\mu \text{grad} W) + \left[ -\frac{\partial \overline{w'^2}}{\partial x} - \frac{\partial \overline{u'w'}}{\partial x} - \frac{\partial \overline{v'w'}}{\partial x} \right] \quad (2.12)$$

Here  $\mu$  is the *dynamic viscosity*. The RANS contain 9 new terms, the so-called *Reynolds Stresses* involving products of the fluctuating parts of the velocity components. There are 3 normal and 6 shear Reynolds stresses.

$$\tau = -\rho \begin{pmatrix} \overline{u'u'} & \overline{u'v'} & \overline{u'w'} \\ \overline{v'u'} & \overline{v'v'} & \overline{v'w'} \\ \overline{w'u'} & \overline{w'v'} & \overline{w'w'} \end{pmatrix} \quad (2.13)$$

Thus, RANS models don't model the turbulence itself but the effects it has on the mean flow. Despite this they provide a good balance between the computational resources required and the accuracy of the results. Besides they are robust, versatile [60] and well-established, with a lot of literature available, and they are applied in most of the CFD-based wind assessment software (Fluent, ANSYS CFX, Phoenix WindSim) [61].

## The Closure Problem

The N-S equations are a system of 4 equations with 4 unknown variables and therefore a *closed* set and solvable from a theoretical point of view (DNS). However, after the *Reynolds averaging* there are 6 new unknowns, the Reynolds Stresses. This is the so-called *Closure Problem*.

## The Standard k-ε model (SKE)

This two-equation model was developed by Jones and Launder in 1972 [62,63] and is based on the idea proposed by Boussinesq [64] that Reynolds stresses can be expressed as function of the mean velocities derivatives, analogous to the relation for the viscous stress in a Newtonian fluid.

$$\tau_{ij} = 2\mu_t \left[ \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] - \frac{2}{3} \rho k \delta_{ij} \quad (2.14)$$

Where  $\mu_t$  is the *dynamic turbulent viscosity* [Pa.s], interpreted as a measure of the additional viscosity due to the turbulence  $\mu_{Total} = \mu_t + \mu$  and  $k = \frac{1}{2} \overline{u'_i u'_i}$  is the *turbulent kinetic energy per unit of mass* [m<sup>2</sup>/s<sup>2</sup>] or TKE, which is the fluctuating term of the kinetic energy. If linear, this dependency implies that normal Reynolds Stresses are isotropic, and therefore the turbulence too.



The turbulent viscosity can be expressed as a function of the velocity ( $v_s$ ) and length scales ( $l_s$ ) of the flow according to the *Prandtl's mixing-length hypothesis* [65] as follows:

$$\frac{\mu_t}{\rho} = \vartheta_t = v_s l_s = l_m^2 \left| \frac{dU}{dy} \right| \rightarrow \tau_{xy} = \tau_{yx} = \rho l_m^2 \left| \frac{dU}{dy} \right| \frac{dU}{dy} \quad (2.15)$$

Where  $\vartheta_t$  is the *kinematic turbulent viscosity* and  $l_m$  is the *mixing length* which expression depends on the flow geometry and must be specified beforehand. This may lead to inaccuracies for poorly characterised complex flows ([59], pag.367). Prandtl also proposed to define  $v_s$ , and therefore  $\vartheta_s$  in terms of the turbulent kinetic energy  $k$  and a constant  $c$  as follows:

$$v_s = ck^{1/2} \Rightarrow \vartheta_t = ck^{1/2} l_m \quad (2.16)$$

This requires the TKE value. For this purpose, a TKE transport equation was obtained by multiplying each of the three velocity N-S equations (Eq.2.4,2.5,2.6,2.7) by the corresponding fluctuating velocity component and adding up the results, then doing the same for the RANS equations (Eq.2.9,2.10,2.11,2.12) and subtracting the resulting equations which yields:

$$\frac{\partial(\rho k)}{\partial t} + (\rho k \mathbf{U}) = \text{div} \left( -\overline{p' u'} + 2\mu \overline{u' e'_{ij}} - \rho \frac{1}{2} \overline{u'_i u'_i u'_j} \right) - 2\mu \overline{e'_{ij} e'_{ij}} - \rho \overline{u'_i u'_j} E_{ij} \quad (2.17)$$

In Eq.2.17 the first term on the left side represents TKE's change rate, the second TKE transport due to convection. On the right side, the first term of the divergence represents TKE transport due to pressure, the second the transport due to viscous stresses and the third the transport due to Reynold stresses. As for the two terms left, the first is the TKE's dissipation rate and the second the turbulent production, where  $E_{ij}$  is the mean deformation rate matrix.

However, in Eq.2.17 there are *unknowns*, like the TKE dissipation rate per unit of mass, commonly designated as  $\varepsilon = 2\vartheta \overline{e'_{ij} e'_{ij}}$ , where  $e_{ij}$  is the fluctuating component of the deformation rate matrix<sup>3</sup>. Besides,  $k$  alone isn't enough to characterize turbulent flows, for it doesn't distinguish between small and large eddies which have different dissipation rates. Therefore, a similar transport equation for  $\varepsilon$  is required.

Both variables are closely related and  $\varepsilon$  increase and decrease when  $k$  does so that no negative values for  $k$  will ever arise. At high Reynolds numbers,  $\varepsilon$  is proportional to  $v_s$  and  $l_s$  according to ([59], pag.370) and therefore  $\vartheta$  can now be expressed in terms of  $k$  and  $\varepsilon$  through Eq.2.17.

$$\varepsilon = c' \frac{v_s^3}{l_s} \rightarrow \vartheta_t = c' c \frac{k^2}{\varepsilon} = C_\mu \frac{k^2}{\varepsilon} \quad (2.18)$$

---

3. The dissipation rate is a parameter used for small eddies, however at high Reynolds numbers, the rate at which large eddies extract kinetic energy from the mean flow equals the rate of energy transfer through the flow to the smallest and dissipating eddies ([66] pag.70).



Where  $C_\mu$  is a dimensionless constant. Using these expressions, the  $\varepsilon$  and TKE transport equations can be written as follows:

$$\frac{\partial(\rho k)}{\partial t} + (\rho k \mathbf{U}) = \left( \frac{\mu_t}{\sigma_k} \text{grad} k \right) + 2\mu_t E_{ij} E_{ij} - \rho \varepsilon \quad (2.19)$$

$$\frac{\partial(\rho \varepsilon)}{\partial t} + (\rho \varepsilon \mathbf{U}) = \left( \frac{\mu_t}{\sigma_\varepsilon} \text{grad} \varepsilon \right) + C_{\varepsilon 1} 2\mu_t E_{ij} E_{ij} \frac{\varepsilon}{k} - C_{\varepsilon 2} \rho \frac{\varepsilon^2}{k} \quad (2.20)$$

Of the new model coefficients  $\sigma_k$  and  $\sigma_\varepsilon$  are usually assigned constant values, 1 and 1.3 respectively, while  $C_{\varepsilon 2}$ ,  $C_{\varepsilon 1}$  and  $C_\mu$  are given by the following equations:

$$C_{\varepsilon 1} = C_{\varepsilon 2} - \frac{\kappa^2}{\sqrt{C_\mu} \sigma_\varepsilon} \quad (2.21)$$

$$C_\mu = \frac{1}{\left( 0.5 \left( \left( \frac{\sigma_u}{u_*} \right)^2 + \left( \frac{\sigma_v}{u_*} \right)^2 + \left( \frac{\sigma_w}{u_*} \right)^2 \right) \right)^2} \quad (2.22)$$

Where  $\kappa$  is the *Karman* constant whose value will be taken as 0.41 and  $\sigma_u$ ,  $\sigma_v$  and  $\sigma_w$  are the variances of the velocity components. However, except when measurements are available and  $C_\mu$  can be fitted to a TKE profile, a set of values are provided, which may vary depending on the source:

	A ÉTS OpenFOAM RNG $k-\varepsilon$	B CENER Fluent $k-\varepsilon$	C Vattenfall OpenFOAM $k-\varepsilon$	D Risø DTU EllipSys3D $k-\varepsilon$	E CRES In-house $k-\omega$
$\kappa$	0.40	0.4187	0.4187	0.40	0.40
$C_\mu$ or $\beta_*$	0.0297	0.0297	0.033	0.03	0.033
$C_{\varepsilon 1}$	0.403	1.138	1.21	1.21	-
$C_{\varepsilon 2}$	1.68	1.92	1.92	1.92	-
$\sigma_k$	1.00	1.00	1.00	1.00	-
$\sigma_\varepsilon$	1.30	1.30	1.30	1.30	-
$\eta_0$	4.38	-	-	-	-
$\beta$	0.012	-	-	-	0.0275
$\alpha$	-	-	-	-	0.3706
$\sigma$	-	-	-	-	0.5
$\sigma_*$	-	-	-	-	0.5

Figure 2.4: SKE model coefficients [61].

## Model Limitations

The use of the  $k-\varepsilon$  model is limited to steady, incompressible, Newtonian, isotropic and fully turbulent flows in a neutrally stratified ABL where the effects of buoyancies, gravity and Coriolis forces can be neglected.

### 2.2.4 Near-Wall Physics

Most of the turbulence flows studied for engineering applications are bounded by one or more solid surfaces (planes, airfoils, pipes etc). The effects of solid boundaries on turbulent flows are twofold [67]:

- Anisotropic turbulent flow due to the damping of the velocity component normal to the wall.
- Increasing of the TKE production through the shearing mechanism in the flow.

These effects give rise to a  $\delta$ -thick boundary layer, where two main regions, the *inner* and *outer boundary* layers, may be distinguished. These can be divided in turn into several sublayers or regions regarding the flow characteristics in each of them ([59] pag. 264-270).

- **Inner Layer**( $y/\delta < 0.1$ )
  - *Viscous or laminar sublayer*: The closest region to the wall, here viscous forces (due to molecular viscosity) prevail over turbulent forces and dominate the shear stress resulting in a laminar-regime flow.
  - *Buffer sublayer*: Transition region between viscosity- and turbulence-dominated regions.
  - *Logarithmic sublayer*: Also, called *inertial* sublayer. At the upper border of the inner layer, here turbulent forces prevail resulting in an inertia-dominated flow.
- **Outer Layer**: Here the logarithmic layer continues. The viscosity effects become increasingly negligible until the flow can be considered fully inviscid.

The inner sublayer thickness is considered equal to the so called *viscous length scale* (Eq.2.23) and is used to scale magnitudes in the near-wall region, such as the distance from the wall  $y$ :

$$\delta_\vartheta = \frac{\vartheta}{u_\tau} \rightarrow y^+ = \frac{y}{\delta_\vartheta} = \frac{yu_\tau}{\vartheta} \quad (2.23)$$

Where  $u_\tau$  is the friction velocity at the wall. This relation makes  $y^+$  very useful to set the limits of the different regions of the *inner layer* ([59] pag.274). However not all the authors agree with this division ([32], [68],[66] pag.210). In OF it is considered that the logarithmic layer begins at  $y^+ = 10.97$  [69].

### The Law of the Wall

Given a 2D turbulent flow in the near-wall region, streamwise derivatives of any variable can be assumed to be much smaller than those in the direction normal to the wall and therefore can be neglected. The flow can also be considered in equilibrium, which means the TKE production ( $G_\kappa$ ) and TKE dissipation rate ( $\epsilon$ ) are equal [67].

If these assumptions are applied on the RANS equations (Eq.2.9,2.10,2.11,2.12) the result is:

$$\frac{d}{dy} \left( (\mu + \mu_t) \frac{du}{dy} \right) = 0 \Rightarrow (\mu + \mu_t) \frac{du}{dy} = cte \quad (2.24)$$

The corresponding  $\vartheta_t$  can be expressed according to the mixing length model as follows:

$$\vartheta_t = C_\mu \frac{k^2}{\varepsilon} = \kappa u_\tau y \quad (2.25)$$

This leads to new expressions for the wall shear stress and the RANS equation in the near-wall region:

$$\tau_w = (\mu + \mu_t) \frac{\partial u}{\partial y} \Big|_{y=0} \quad u_\tau = \sqrt{\frac{\tau}{\rho}} \quad (2.26)$$

$$(\mu + \mu_t) \frac{du}{dy} = \rho u_\tau^2 \quad (2.27)$$

From Eq.2.26 and Eq. 2.27 it can be derived that the sum of the viscous and turbulent stresses remains constant and equal to the wall shear stress throughout the near-wall region, although the values of the terms may change (see Fig.2.5) ([66] pag.69, [70], [67]).

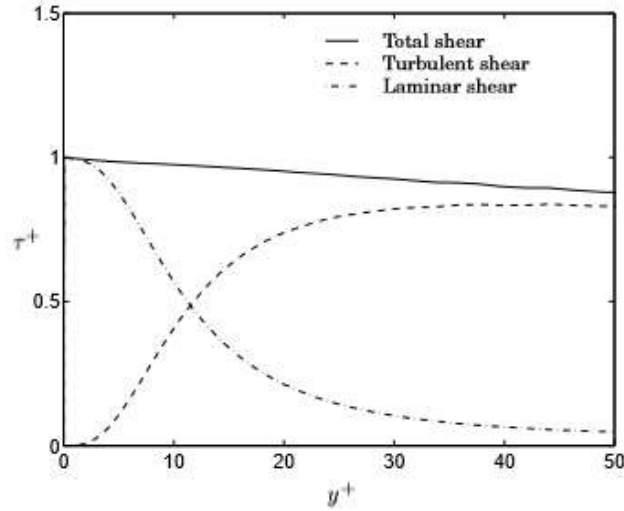


Figure 2.5: Shear stresses in the near-wall region [67].

To derive solutions in the near-wall region, flow variables are expressed in a non-dimensional form using  $\delta_\eta$ :

$$\varepsilon^+ = \frac{\varepsilon \vartheta}{u_\tau^4} \quad k^+ = \frac{k}{u_\tau^2} \quad \vartheta_t^+ = \frac{\vartheta_t}{\vartheta} \quad u^+ = \frac{u}{u_\tau} \quad (2.28)$$

The velocity expressions in the different sub-layers inside the near-wall region takes the following form ( [66], pag.69):

$$u^+ = \frac{u}{u_\tau} = f\left(\frac{\rho y u_\tau}{\mu}\right) = f(y^+) \quad (2.29)$$

This is the *law-of-the-wall* for the near-wall velocity distribution. It asserts that at high Reynolds number, the mean velocity profile inside the *inner* layer only depends on the viscous scales ( [71], [59] page-272). From Eq.2.29 expressions for each sub-layer of the near-wall region can be derived

- *Laminar sublayer*: Here  $\vartheta_t \ll \vartheta$  ( $\vartheta_t^+ \ll 1$ ) which means viscous shear stresses prevail over the turbulent ones. If applied on Eq.2.27 the following expression results:

$$u^+ = y^+ \quad (2.30)$$

- *Logarithmic sublayer*: Here  $\vartheta_t \gg \vartheta$  ( $\vartheta_t^+ \gg 1$ ). The turbulence effects prevail over the molecular viscosity. This allows to approximate Eq.2.27 as follows:

$$\frac{\partial u^+}{\partial y^+} = (1 + \vartheta_t^+) \quad (2.31)$$

Substituting  $\vartheta_t^+$  according to Eq.2.25 the result is the so-called *log-law*, where  $E$  and  $B$  are experimentally established constants which values for smooth terrain are 9.793 and 5.43 respectively [68].

$$u^+ = \frac{1}{\kappa} \ln(E y^+) = \frac{1}{\kappa} \ln(y^+) + B \quad (2.32)$$

$$\varepsilon^+ = \frac{\vartheta}{u_\tau \kappa y} \quad k^+ = \frac{1}{\sqrt{C_\mu}} \quad (2.33)$$

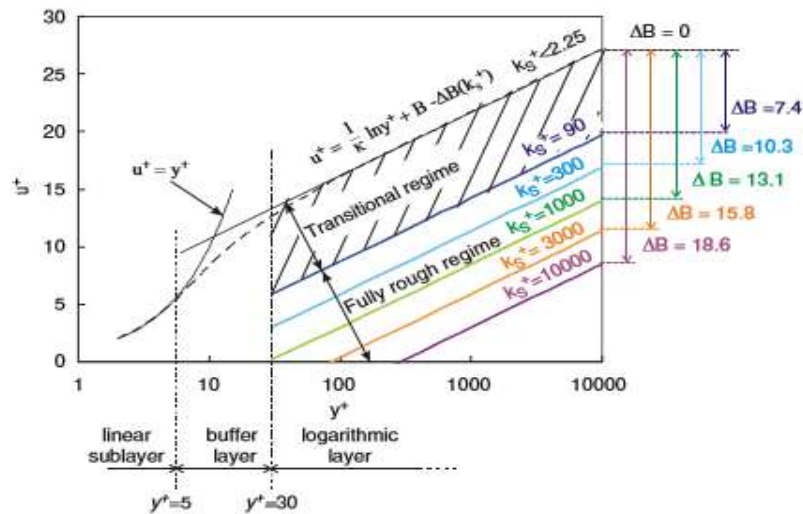


Figure 2.6: Normalized wind speed in the near-wall region [68].

## 2.2.5 The Standard Wall Functions

In the present work, the near-wall flow will be simulated using wall functions [63]. These are semi-empirical expressions which bridge the viscous or linear sub-layer, that remains unsolved, and approximate the mean flow characteristics between the wall and the turbulence-dominated region where the log-law region can be applied. They may reduce the results accuracy, but, on the other hand, they also reduce the computing power requirements and work well for high Reynolds numbers. They have the beneficial property of better numerical stability and a higher convergence speed ([67], [60] page-114).

The log-law region is characterized by a velocity profile (Eq.2.32), the equilibrium conditions and the near constancy of the Reynolds shear stress throughout it [59]. Thus, provided a fine enough mesh, the total shear stress at the first point of the mesh,  $P$ , should be equal to the wall shear stress. However, this is not always the case:

$$\tau_w = (\mu + \mu_t) \frac{\partial u}{\partial y} \Big|_w > (\mu + \mu_t) \frac{\Delta u}{\Delta y} = (\mu + \mu_t) \frac{u_P}{y_P} = \mu_{effective} \frac{u_P}{y_P} \quad (2.34)$$

This difference must be corrected to get the right shear stress at point  $P$ . In the present work this will be done by modifying the turbulent viscosity<sup>4</sup>. In the logarithmic sub-layer the wall shear is given by Eq.2.35. The comparison of this expression with that of the shear stress at  $P$ , yields the following expression for the modified turbulent viscosity:

$$\tau_w = \rho u_\tau^2 = \frac{\rho u_\tau u \kappa}{\ln(Ey^+)} \quad (2.35)$$

$$\mu_{effective} \frac{u_P}{y_P} = \frac{\rho u_\tau u_P \kappa}{\ln(Ey_P^+)} \rightarrow \mu_e = \frac{\rho u_\tau y_P \kappa}{\ln(Ey_P^+)} \quad (2.36)$$

The *modified turbulent viscosity*  $\mu_e$  accounts for the effects of surface roughness on the flow and depends on the position of the first cell above the terrain. Provided the equilibrium condition is fulfilled and that turbulent forces are much larger than viscous forces and therefore  $\tau \approx \tau_w$  ([59] page-378), the expressions for  $k$  and  $\varepsilon$  at  $P$  would be:

$$G_k = \tau_w \frac{\partial u}{\partial y} \Big|_{y=0} = \tau \frac{\partial u}{\partial y} = \frac{u_\tau^3}{\kappa y} = \varepsilon \quad (2.37)$$

$$\text{At } y = y_P \Rightarrow \varepsilon = \frac{u_\tau^3}{\kappa y_P} = \frac{k_P^{3/2} C_\mu^{3/4}}{\kappa y_P} \Rightarrow k_P = \frac{u_\tau^2}{\sqrt{C_\mu}} \quad (2.38)$$

## Roughness

Land cover<sup>5</sup>, such as plants or rocks and rough surfaces are characterized by a distinctive *roughness* value. Two are the most used roughness parameters:  $k_s$ , the *equivalent sand-grain roughness height* [72,73] and the *aerodynamic roughness length* ( $z_0$ ).

---

4. It can also be made through the addition of a source term to the momentum equation  $S_u = \tau_w$ .

5. Big buildings or urban areas must be explicitly introduced using programs) or substituted by objects with similar characteristics [52,75].

According to [74], roughness effects can be introduced into the wall functions through the addition of a parameter  $B$  into the velocity log law:

$$u^+ = \frac{1}{\kappa} \ln \left( \frac{y u_\tau}{\vartheta} \right) + B - \Delta B \left( \frac{u_\tau k_S}{\vartheta} \right) \quad (2.39)$$

However, the relation between  $\Delta B$  and  $k_S^+$  is not fixed and may change. For  $k_S^+ > 90$ , the laminar sublayer disappears, and the flow can be considered independent from the molecular viscosity. In this case  $\Delta B$  is given by Eq.2.40 where  $C_S$  is the Smagorinsky-Lilli constant [68]:

$$\Delta B = \frac{1}{k} \ln(1 + C_S k_S^+) \quad (2.40)$$

In a fully turbulent flow ( $C_S k_S^+ \gg 1$  with  $C_S > 0.2$ ) Eq.2.40 can be simplified and the velocity could be expressed as follows:

$$\Delta B = \frac{1}{k} \ln(C_S k_S^+) \rightarrow u^+ = \frac{1}{\kappa} \ln(E y^+) - \frac{1}{k} \ln(C_S k_S^+) \rightarrow \frac{u}{u_\tau} = \frac{1}{\kappa} \ln \left( \frac{E y}{C_S k_S} \right) \quad (2.41)$$

At the upper limit of the inertial sub-layer Eq.2.39 should be equal to the well-known wind profile equation (Eq.2.2). And therefore  $u_\tau = u_{ABL}$ . Under this assumption, the following expression relating the two roughness parameters  $k_S$  and  $z_0$  can be extracted:

$$\frac{y_p + y_0}{y_0} \approx \frac{y_p}{y_0} = \frac{E y_p}{C_S k_S} \quad (2.42)$$

$$E y_0 \approx C_S k_S \quad (2.43)$$

This approximation leads to a new expression for  $\mu_e$ . As for the relation between  $y_0$  and  $k_S$ , it depends on the values of  $E$  and  $C_S$  and different expressions have been found in the literature. In the present work,  $E$  is 9 and  $C_S$  is 0.3 like in [67,60].

$$k_S \approx 30 y_0 \quad (2.44)$$

$$\vartheta_t = \vartheta \left[ \frac{\kappa y^+}{\ln \left( \frac{y + y_0}{y_0} \right)} - 1 \right] \quad (2.45)$$

## 2.3 The Canopy Model

Wall functions are a valid method to model the effects of obstacles and rough functions on the flow up to  $z_0 = 0.1$  m. Higher  $z_0$  values imply the presence of small-obstacles, such as forests or small buildings. The effects of this kind of obstacles on the flow are only significant up to a vertical distance 3 times its height [76] and therefore they are not usually considered in wind assessment studies, for the hubs of the wind turbines are usually located above 50 m. However, they may have a great effect on the performance of the SWT used for rural electrification. For this reason, it may be worth applying a different treatment for such elements, considering them as obstacles with different properties [77,78]. This is the so-called canopy model.

The easiest approach is considering the canopies as solid bodies like in [79] and introduce a *displacement height* ( $d$ ), corresponding to canopy height, which displace the velocity profile an equivalent distance in the vertical direction.

$$u(z) = \frac{u_r}{\kappa} \ln \left( \frac{z-d}{z_0} \right) \quad (2.46)$$

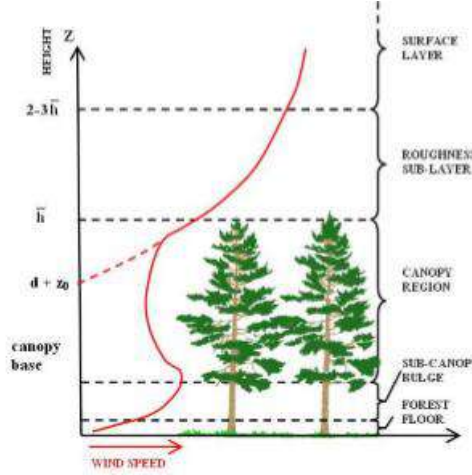


Figure 2.7: Wind speed profile entering a forested canopy

This approach isn't accurate, for considering a forest as a solid body, implies neglecting the effects that elements such as trunks, branches or leaves may have on the wind flow [77,78], like:

- Momentum absorption.
- Momentum and scalar properties are transported vertically by turbulent diffusion.
- Flow's kinetic energy conversion into turbulent kinetic energy at different length scales.
- Turbulent eddies are broken into small wake eddies by the foliage. Turbulence dissipation.

A more sophisticated approach considers forested areas as highly complex porous media to be represented as subdomains with a lower porosity value than air [80, 81,82, 83], being the porosity of a volume cell defined as follows:

$$\beta = \frac{\text{Volume of voids}}{\text{Total Volume}} \quad (2.47)$$

Forested canopies are also characterized by magnitudes such as the *drag coefficient*  $C_D$ , which ranges from 0.15 to 0.3, according to [84], the *leaf area*  $a$  [ $\text{m}^2/\text{m}^3$ ], the *leaf area density*, LAD [ $\text{m}^2/\text{m}^3$ ] [85,86], or the *leaf area index*, LAI [ $\text{m}^2/\text{m}^2$ ] [87,88] which values vary among species and are subjected to seasonal changes. Following this approach different CFD canopy models have been developed by adding extra terms to the transport equations which should account for the effects of vegetation on the flow variables. The extra term in the momentum equations (Eq.2.10,2.11,2.12) represent the momentum losses due to the drag forces.

$$S_u = -\rho C_1 \vec{U} - \rho C_2 |U| \vec{U} \quad (2.48)$$

Here  $C_1$  (1/s) and  $C_2$  (m<sup>2</sup>/m<sup>3</sup>) are respectively the so called *viscous* and *inertial* resistive coefficients. Although there is a wide agreement about the expression of the momentum sink term, there is neither a general expression nor a table which relates  $C_1$  and  $C_2$  with the canopy characteristic magnitudes.

As for the additional terms in the TKE and  $\varepsilon$  equations (for the  $k$ - $\varepsilon$  model), which represent the turbulent kinetic energy and energy dissipation rate increase, different expressions have been proposed [89]. For instance, those proposed in [90]:

$$S_k = \rho C_D \alpha (\beta_p |u|^3 - \beta_d k |u|) \quad (2.49)$$

$$S_\varepsilon = \rho C_D \alpha \frac{\varepsilon}{k} (C_{\varepsilon 4} \beta_p |u|^3 - C_{\varepsilon 5} \beta_d k |u|) \quad (2.50)$$

Where  $\beta_p$  is the fraction of the mean flow's kinetic energy converted to TKE due to the canopy and  $\beta_d$  is the fraction of TKE dissipated rapidly after the wake [84].



# Chapter 3

## Computational Fluid Dynamics (CFD)

In the present chapter, the general working of a CFD code using the *Finite Volume Method* (FVM) will be briefly described as well as a few of the most common discretization schemes and its characteristics.

### 3.1 Computational Fluid Dynamics (CFD)

CFD is defined as the technique of analysis of numerical models describing physical phenomenon related to fluid motion, heat transfer and associated phenomena, through computer-based simulations [66]. It has therefore numerous applications in different fields, from aerodynamics to biomedical engineering or meteorology.

CFD codes are based on a set of numerical versatile algorithms capable of solving the governing equations of fluid-related phenomena. Regardless of the code used, the whole simulation process can be divided into three general steps:

- **Pre-processing:** Here all the characteristics of the physical phenomena to be simulated must be converted into information that can be used by the code. This implies:
  - Domain definition and its division into cells or sub-domains (*meshing*).
  - Definition of flow and work fluid properties.
  - Set boundary conditions.
- **Solving:** The value of the flow variables is calculated at each point of the domain according to the governing equations. This can be done by different methods, however they all follow three basic steps:
  - The unknown values of the variables are approximated using a simple algorithm.
  - The approximations are substituted into the governing equations (*discretization*).
  - The resulting set of algebraic equations are solved iteratively using numerical methods.
- **Post-processing:** This last step is dedicated to the display of the results, visualization and manipulation tools.

In the first and second steps, different methods may be used, for instance the *Finite Difference Method* (FDM), the *Finite Element Method* (FEM), or the *Finite Volume Method* (FVM). Due to its versatility and simplicity, the FVM is used in most of the commercial CFD codes, like FLUENT or PHOENICS. Amongst its advantages are its low memory requirement and is relatively fast even for very large problems. There is an extended bibliography about it and has been tested in many occasions for all kind of cases. The FVM will be the method used throughout the present work in OF.

As for the latter step, in the present work *ParaView* and *gnuplot* were used for visualizing and plotting the results.

### 3.2 The Finite Volume Method (FVM)

FVM applies the conservation equation to each volume element, which makes it very convenient for its use in fluid dynamic, where variables such as mass, momentum and energy must fulfil the conservation principle.

For the sake of simplicity, the domain is divided through a Cartesian grid in most of the cases. This grid defines the volume boundaries but not the computational nodes, which are assumed to be in the cell volume (CV) centre as it is the most common assumption ([91] pag-84).

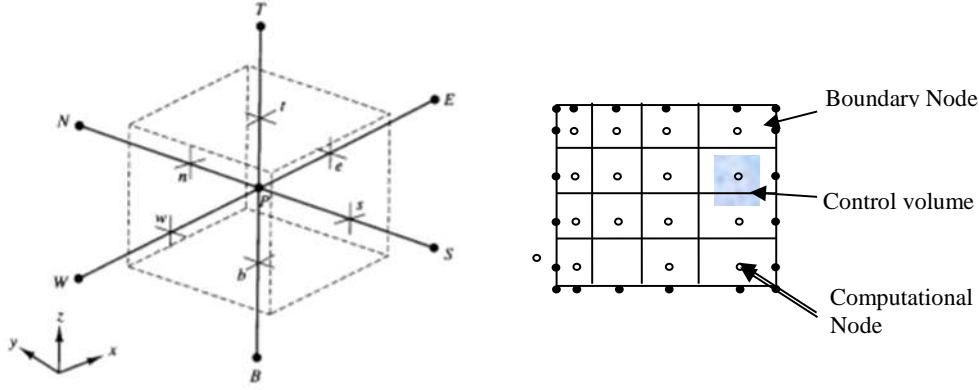


Figure 3.1: Control volume [92].

In fluid dynamics, the conservative form of the differential transport equation for any general variable or fluid property  $\phi$  can be written as follows [66]:

$$\frac{\partial(\rho\phi)}{\partial t} + (\rho\phi u) = (\Gamma grad\phi) + S_\phi \quad (3.1)$$

If integrated over a control volume it yields an equation, which corresponds to the  $\phi$  conservation equation over that finite volume. Then if the divergence theorem is applied to the convection and diffusion terms, the following expression results:

$$\frac{\partial}{\partial t} \left( \int_{CV} \rho\phi dV \right) + \int_A n(\rho\phi u) dA = \int_A n(\Gamma grad\phi) dA + \int_{CV} S_\phi dV \quad (3.2)$$

The FVM store  $\phi$  values at the CV centres by default. However, to convert Eq.3.2 into a solvable linear algebraic equation the values of  $\phi$  at the cell's faces are required. These values can be interpolated from the values at the nodes of the surrounding control volumes through different methods, the *discretization* or *interpolation* schemes. The final result is a linear equation in which the value of  $\phi$  at the centre of the control volume is expressed in terms of its value at the centres of the surrounding volumes, this is the *discretized* equation (Eq.3.3).

$$a_P \phi_P = a_W \phi_W + a_E \phi_E + a_N \phi_N + a_S \phi_S + b \quad (3.3)$$

$$a_P \phi_P = \sum_{nb} a_{nb} \phi_{nb} + b \quad (3.4)$$

Here, “ $nb$ ” stands for neighbour control volume. For the cells, adjacent to the boundaries of the domain, the values of  $\phi$  at the control volume faces are given by the own boundary conditions. If this procedure is applied to each cell of the domain a system of linear algebraic equations of the form  $Ax=b$  with as many equations and unknowns as cells<sup>6</sup>.

### 3.2.1 Interpolation Schemes

The choice of the interpolation scheme must consider the nature of the term and its physical meaning. In fluid flows, the relative importance of the two main transport mechanisms, convection and diffusion, is determinant to choose the interpolation scheme. The results also depend on the grid refinement. In almost every case the accuracy of the simulation will be improved by increasing the grid refinement [66].

In general, there is a set of fundamental requirements which, if fulfilled, guarantee physically meaningful results and stable iterative solutions:

- **Conservativeness:** The flux of a transported property leaving a control volume must be equal to the flux of that property entering the adjacent control volume.
- **Boundedness:** The values of the transported property and the coefficients of the linear equations must be within the limits imposed by the physical characteristics of the case.
- **Transportiveness:** The direction of mutual influence of the nodes of neighbouring control volumes is considered. In diffusion, each node is equally influenced by the all the surrounding nodes whereas in convective flows, the influence of the nodes upstream is more important.

The accuracy of the different schemes is given by the number of terms kept of a Taylor series development of function  $\phi(x+\Delta x)$  [66,91]:

$$\phi(x + \Delta x) = \phi(x) + \left( \frac{\partial \phi}{\partial x} \right)_x \Delta x + \left( \frac{\partial^2 \phi}{\partial x^2} \right)_x \frac{\Delta x^2}{2} + \dots \quad (3.5)$$

A common factor with the form  $\Delta x^n$ , can be extracted from these terms. The power  $n$  of  $\Delta x$  is the so-called *order of the difference approximation*, it determines the rate at which the error tends to 0 as the grid is refined and will be used to classify the different interpolation methods.

---

6. In [66] as in other works the expression resulting from the integration over the CV and the linear approximation or linearization of the system equations is simply referred as *discretized* equations and the process as *discretization*, however other authors distinguished between *discretization* and *linearization* and thus call the resulting equations *discretized* and *linearized* equations respectively. In the present work the terminology used in [66] is applied.

The higher the order the smaller the number of terms neglected and therefore the higher the accuracy of the result. However high order computing scheme might also be more unstable and produce oscillating solutions, especially on coarse grids ([91] page-101).

### First Order Upwind Scheme

It is the simplest and most used interpolation scheme and the only one which always satisfies the boundedness condition ([91] page-88). It assumes that the value of any variable  $\phi$  at a cell's face is the same as the value at the cell centre upstream. It's very stable and easy to use.

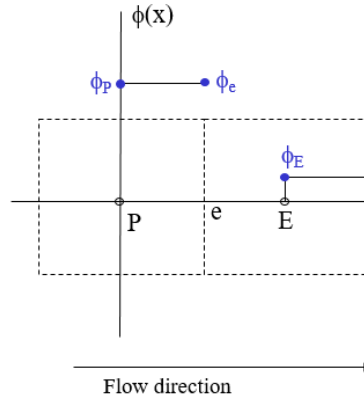


Figure 3.2: First order upwind scheme [92].

However, if the flow is not aligned with the grid lines, for instance in strong convective flows or flows with a high Reynolds number, it might produce wrong results<sup>7</sup>.

### Central Differencing Scheme (CDS)

The value of a variable at the control cell face is calculated through a linear interpolation between the values at the centres of the two cells sharing the face. For a uniform grid:

$$\phi_e = \frac{(\phi_P + \phi_E)}{2} \quad (3.6)$$

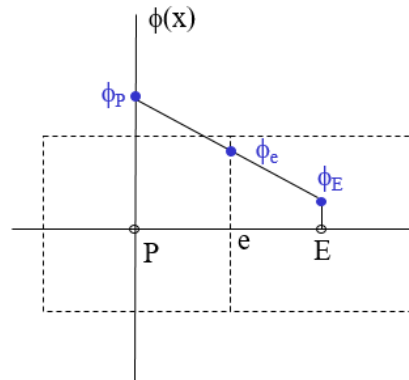


Figure 3.3: Central differencing scheme [92].

<sup>7</sup>These errors are commonly known as false diffusion because of their diffusion-like appearance and can be palliated to some extent by further refinement of the grid [66].

As a second order scheme, it is in general more accurate than the first order upwind scheme, although in occasions it may produce oscillatory solutions ([91] page-89). One of its drawbacks is that it doesn't satisfy *transportiveness*, for it doesn't consider flow direction and the value at a node is equally influenced by all the surrounding nodes, which makes it more appropriate for diffusive transport phenomena [66,92].

### Second Order Upwind Scheme

This scheme calculates the value of the transported property at the face of a cell using the centre values of the 2 neighbouring cells upstream. It is in general as stable as the first order upwind scheme and even more accurate, except in regions with strong gradients ([60] pag.678).

$$\phi_{Face} = \phi_P + \frac{\phi_E - \phi_P}{\phi_E - \phi_W} (\phi_P - \phi_W) \quad (3.7)$$

### QUICK Scheme

QUICK stands for *Quadratic Upwind Interpolation for Convective Kinetics* [93]. It approximates the variable profile between P and E using a quadratic curve which pass through the two closest nodes upstream and the closest node downstream of the face.

$$\phi_e = \frac{6}{8}\phi_P + \frac{3}{8}\phi_E - \frac{1}{8}\phi_W \quad (3.8)$$

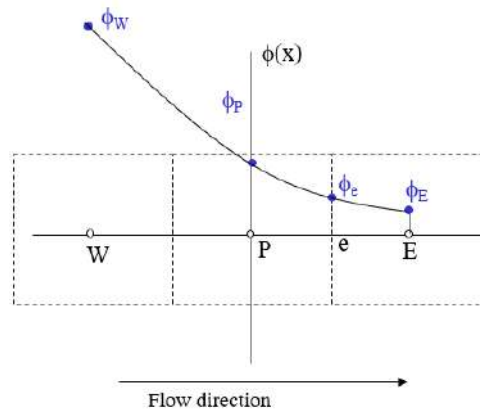


Figure 3.4: QUICK scheme [92].

QUICK produces, in general, the most accurate results. Except in regions with strong gradients, where some coefficients may become negative, which infringe *boundedness* and may lead to overestimated or underestimated results and thus to stability problems.

The schemes presented are defined at the cell centres, however velocities are commonly defined at the cell faces instead to avoid problems related to the oscillation of the pressure field. Thus, there are two superposed grids in the end, the original and another one centred on the cell faces of the first. The latter is the so called *staggered* grid [66].

### 3.2.2 Solvers

The discretization process yields a system of linear algebraic equations which must be solved. Due to the complexity and size of the system, iterative methods are preferred over direct methods due to their lower storage capacity requirements [66].

In OF the following terminology is used. The term *linear solver* or just *solver* is exclusively used for the numerical methods for solving a linear equations system. According to [66] any suitable matrix solution technique might be valid to solve the linear equation system. Given a system of linear equations for a variable  $\phi$ , the scheme of a general iterative method could be written as follows ([91] pag.109):

$$A\phi = B \quad (3.9)$$

$$M\phi^{n+1} = N\phi^n + Q \quad (3.10)$$

Whereas a set of equations and algorithms used to solve a certain problem is called *application solver*.

### 3.2.3 Pressure-Velocity Coupling

Linear solvers can be applied to any variable except pressure, for there is no pressure equation as such, and therefore the pressure field must be determined by other means. In incompressible flows the *pressure-velocity coupling* is applied, based on the principle that if the pressure field is correct the velocity field resulting from the solution of the momentum equations will satisfy continuity.

#### The Pressure-Based Solver Algorithm

To calculate the pressure field an iterative guess-and-correct procedure based on a specific algorithm is usually followed. In the present work, the *Semi-Implicit Method for Pressure-Linked Equations*, SIMPLE [94] will be used in every case. It is used by Fluent for steady-state calculations ([95] page-1487) and recommended by [69] for steady incompressible and turbulent flows. The application of this procedure is based on an expression for the corrected pressure value in each cell:

$$a_p p' = \sum_{nb} a_{nb} p' + b' \quad (3.11)$$

Here  $p'$  is the pressure correction,  $b'$  represents the continuity imbalance and the rest are coefficients which value is given by the grid and the flow field. This correction and is based on the imbalance of mass fluxes through the control volume faces.

Two general procedures may be distinguished for the solution of the discretized governing equations in a domain:

- *Segregated*: Governing equations are solved one after another, first the momentum equations, then the pressure-velocity coupling and finally the scalar variables equations.
- *Coupled*: Momentum equations, pressure-velocity coupling and scalar variable equations are solved in a coupled manner for each control volume.

The *segregated* procedure is used in most commercial FVM codes and is especially suitable for incompressible flows. However, although it converges slower than the coupled scheme, its memory requirements are lower [95]. It will be used throughout this work.

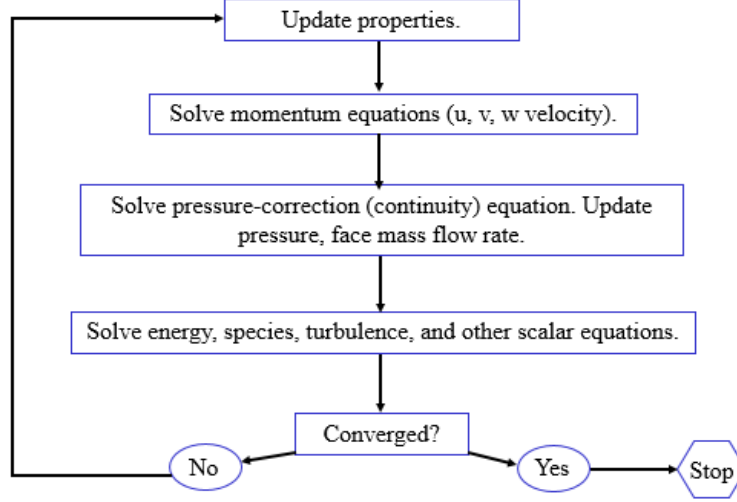


Figure 3.5: *Segregated* procedure for a *pressure-based* solver [92].

### 3.3 Relaxation

Relaxation techniques are commonly used to speed up or improve the stability of convergence processes. They consist in the modification of the results of an iteration before using them in the next one. The relaxation factors ( $\alpha$ ) must be carefully chosen, for a too high value might lead to a lack of stability and eventually to a divergence, while a too small value would reduce the convergence speed too much [92].

$$\phi_p^n = \phi_p^{n-1} + \alpha(\phi_p^{new} - \phi_p^{n-1}) \quad (3.12)$$

- **Overrelaxation ( $\alpha > 1$ ):** The convergence process is accelerated.
- **No Relaxation ( $\alpha = 1$ ):** Last predicted value is used in the new iteration.
- **Underrelaxation ( $\alpha < 1$ ):** Convergence process slowdown. Instability and divergence risk decrease.

### 3.4 Residuals.

They result from the substitution of the solution produced by the solver in each iteration in the discretized governing equations and the calculation of the difference between both sides of the equations. They provide a measure of the error in the solution. The smaller they are, the more accurate the solution.

The expression of the absolute residual for a variable  $\phi$  in the cell P after  $N$  iterations is given by Eq.3.13. If sum over all the computational cells of the domain, the result is an expression for the global residual.

$$R_P^N = \left| \sum_{nb} a_{nb} \phi_{nb}^N + b - a_P \phi_P^N \right| \rightarrow R^N = \sum_{cells} \left| \sum_{nb} a_{nb} \phi_{nb}^N + b - a_P \phi_P^N \right| \quad (3.13)$$

Residuals are used to monitor the whole computation process. Following the evolution of the residuals sum throughout the iterations is a common practice to know whether a solution is converging or not.

### 3.5 Convergence

Starting from an initial guess, flow variables are recalculated in each iteration. An iterative process is said to have converged when each governing equation is solved up to a predefined error called *tolerance*. In OF the convergence or termination criterion is based on the residuals of the flow variables. An iterative numerical process is said to be converging when the solutions residuals decrease iteration after iteration, eventually falling below the given tolerance. If correctly chosen, once the process has converged, the solution shouldn't change significantly in further iterations and it should satisfy the governing equations in every cell of the domain up to the set degree of accuracy. It may also occur that, although the residuals haven't yet met the termination criterion, they are no longer decreasing, in that case the solution is converged too.

#### Numerical Uncertainty

The numerical uncertainty of any CFD simulation is due to three different errors:

- **Round-off error:** It is a consequence of the finite precision of the computers. Its minimum value is given by the machine precision. It increases with grid refinement.
- **Iterative error:** Due to the non-linearity of the PDEs solved. Here are included the errors due to the turbulence model chosen, the pressure-velocity coupling and all the iterative sub-cycles. It increases with grid refinement.
- **Discretization error:** A consequence of the approximations made to transform the PDEs into a system of linear equations. It is the dominant error and decreases with the mesh refinement.

In principle, if the solution is converged up to machine accuracy, the iterative error can be considered equal to the round-off error and in general negligible. However due to the computational time required, normally a higher tolerance value is set, accepting a certain degree of iterative error.



# Chapter 4

## Simulation Construction

In this section the general structure of OF is described and the steps followed for the simulation of turbulent wind flow over complex terrain using the k-epsilon model are explained.

### 4.1 OpenFOAM

OpenFoam is a C++ library used to create executables for, either resolving a specific problem, the *solvers*, or perform a task involving data manipulation, the *utilities*. The general structure of OF can be divided into three as is shown in Fig.4.1.

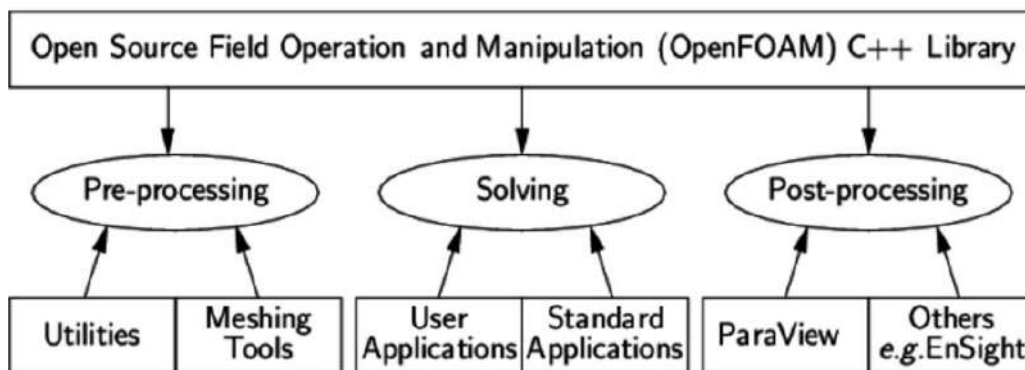


Figure 4.1: OpenFOAM general structure [69].

To simulate an OF case, data about the mesh, the boundary conditions, the fields and the fluid properties must be provided. All this data is stored in a case dictionary as follows:

```
<Case>
|-- 0
|   |-- epsilon
|   |-- k
|   |-- nut
|   |-- p
|   '-- U
|-- constant
|   |-- polyMesh
|   |   '-- blockMeshDict
|   |-- RASProperties
|   '-- transportProperties
'-- system
    |-- controlDict
    |-- decomposeParDict
    |-- fvSchemes
    '-- fvSolution
```

Figure 4.2: OpenFOAM standard case structure

## 4.2 Pre-processing

### 4.2.1 The Domain

It is a delimited part of the space where the simulation takes place. Its size and divisions are decisive factors affecting the outcome of the simulation.

#### 4.2.1.1 Dimensions

After an exhaustive literature research, the general recommendations found for the creation of a domain can be summarized as follows:

- The domain should comprise, not only the area of interest but also all the near topographical features that might influence the wind flow over this area and must be large enough to represent the energy containing motions ([59] page-346).
- The area of interest must be at a sufficient distance from the inlet and outlet patches to guarantee the flow is not affected by the boundary conditions.

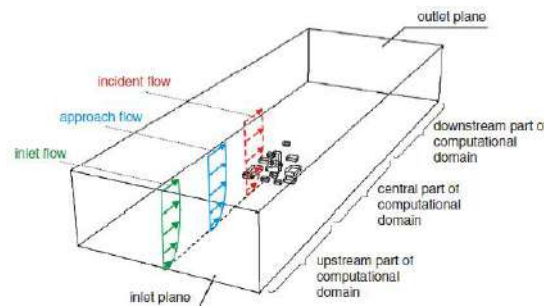


Figure 4.3: Domain example [68].

WAsP developers recommend a distance of at least 2-3 times the horizontal scale of the most significant terrain feature inside the area of interest [96, 97]. And in [98] a minimum area of 5km around each anemometer is recommended.

- As for the top patch a height from 500-1000m over the highest point of the terrain was recommended in different works [96,58] because it approximately corresponds to the mean depth of the Atmospheric Boundary Layer.

#### 4.2.1.2 Meshing

Meshing is the process of dividing the domain in blocks or cells for the application of FVM. The resolution of the mesh affects solution's accuracy and results themselves. The mesh resolution must be fine enough to capture the relevant physical phenomena ([59] page-346), especially in regions of high gradients such a near-wall zones. However, the higher the resolution the larger the number of cells and the higher the computational costs. Grid independency is achieved when the addition of more nodes to the domain doesn't produce any noticeable change in the results. At that point the spatial discretization errors can be neglected<sup>8</sup>.

---

8. The *Grid Convergence Index* (GCI) is proposed in [100] to quantify the result's dependency on the resolution. The GCI can be calculated using the *Richardson Extrapolation Method* (RE) [101].

Most of the works consulted recommend the division of the domain in several regions with different horizontal resolution ranging from 10x10 m cells for the regions of highest interest [99] to 20x20 m in the rest of the domain where the flow is predictable, and no high gradients are expected [102]. As for the vertical resolution, a vertical grading with an expansion ratio around 1.2 is commonly used [61], with cell thickness ranging from less than 1m at the surface [68], up to 5m or more at the top of the domain [96, 56].

#### 4.2.1.3 Meshing Quality Guidelines

The quality of the cells of the domain plays an important role in the accuracy of the results and the stability of the convergence. In general, nodes should be regularly distributed all over the domain, without large empty areas and irregular faces and cells must be avoided. These are a few of the most common cell defects:

- *Incorrectly orientated faces*
- *High Aspect Ratio Cells*
- *Non-Orthogonality*
- *Skewness*

Fulfilling all quality standards while preserving the terrain geometry features may be very difficult, if not impossible, therefore the presence of a certain number of irregular cells must be assumed. In OF the command *checkMesh* can be used to assess mesh quality.

#### 4.2.1.4 Meshing Tools

In the present work two meshing methods were used, the first one based on two OpenFOAM utilities, *blockMesh* and *snappyHexMesh* and the second one based on the open source meshing tool *cfMesh*. Each tool has its own advantages and drawbacks. The election of one of them depends on the case characteristics. In the present work both tools were used.

#### BlockMesh-SnappyHexMesh

*BlockMesh* is the basic OpenFOAM mesh generation utility. The input data for the mesh generation is given in a dictionary file called *BlockMeshDict* located in the *constant/polymesh* directory. It contains information about the size of the domain, its divisions, grading an initial refinement, its limiting surfaces or patches, names and properties. The command *blockMesh* reads this dictionary and decompose the given domain into a set of hexahedral blocks<sup>9</sup> according to the information provided. The resulting mesh is saved in the *constant/polymesh* directory.

*SnappyHexMesh* (SHM) is the standard utility offered by OpenFOAM to adapt a previously generated mesh to a surface given in triangulated surface format (*STereoLithography.stl*) file, located in the *constant/triSurface* folder. The information required to adapt the original mesh to the surface is specified in the *snappyHexMeshDict* inside the *system* folder (see example in [Annex 4](#)). This tool follows a four steps iterative process through which the mesh created with *blockMesh* is carved to adapt it to the given surface.

---

<sup>9</sup>. Hexahedral blocks are usually preferred over other kind of blocks, such as tetrahedral, in most of the works consulted, for they introduced smaller truncation errors and favour convergence [97].

In Fig. 4.4 the whole meshing process, from the original *blockMesh* grid to the final grid adapted to the terrain can be followed step by step.



Figure 4.4: *SnappyHexMesh* steps (a) *blockMesh* mesh (b) castellated mesh (c) snapped mesh.

### cfMesh

This is an open source meshing tool which may be a more user-friendly alternative to the traditional *blockMesh-snappyHexMesh* tandem [103]. Unlike SHM, *cfMesh* doesn't carve a pre-existing mesh following a surface but fills the space limited by a close surface given by one or more *.stl* files with hexahedral cells, according to the specifications provided by the user in a dictionary called *meshDict* inside the *system* folder. The resulting mesh is saved inside the *constant/polymesh* folder.

It has been found that *BM+SHM* offers a higher control degree over the size and grading of the mesh cells than *cfMesh*, allowing the user to set them at will all over the domain and in specific areas too. However, *cfMesh* clearly outperforms *BM+SHM*, when it comes to the addition of boundary layers, which, as it will be seen, are of the utmost importance for the simulations. In both cases, the boundary surfaces of the domain are assigned a type which determines the kind of boundary conditions which may be applied on them. In the present work, all the boundary surfaces were defined as *patches*, except for the solid boundary, corresponding to the terrain, which was defined as a *wall* [69].

#### 4.2.2 Boundary Conditions (BC)

The boundary conditions represent the influence of the surroundings that have been cut off by the computational domain. One of the most used set of boundary conditions for the modelling of turbulent wind flows in the ABL was proposed in [102]. This set is based on a few assumptions about the boundary layer:

- Shear stress is constant throughout the boundary layer.
- Vertical velocity is 0.
- Shear stress-driven flow. Pressure gradient is 0 in vertical and streamwise direction.
- The boundary layer is in equilibrium ( $G_K = \epsilon$ )

These assumptions together with the flow governing equations yield the k-epsilon model profiles for the ABL which will be used as inlet conditions [102]:

$$u = \frac{u_{ABL}}{\kappa} \ln \left( \frac{z - z_{ground} + z_0}{z_0} \right) \quad (4.1)$$

$$k = \frac{u_{ABL}^2}{\sqrt{C_\mu}} \quad (4.2)$$

$$\varepsilon = \frac{u_{ABL}^3}{\kappa(z - z_g + z_0)} \quad (4.3)$$

$$\mu_t = C_\mu \rho \frac{k^2}{\varepsilon} = \rho u_\tau \kappa z \quad (4.4)$$

These equations are an analytical solution of the k- $\varepsilon$  model governing equations [59], provided the following relation between the equations constants is satisfied:

$$\kappa = \sqrt{(C_{\varepsilon 2} - C_{\varepsilon 1})\sigma_\varepsilon \sqrt{C_\mu}} \quad (4.5)$$

$$\sigma_\varepsilon = \frac{\kappa^2}{(C_{\varepsilon 2} - C_{\varepsilon 1})\sqrt{C_\mu}} \quad (4.6)$$

These BC should yield a *horizontally homogeneous turbulent* ABL layer (HHTBL). This means that, on an ideal plain with a constant roughness mean, wind speed and turbulence quantities profiles should not change along the fetch.

Together with other boundary conditions this set of equations has become a standard in the CFD wind modelling and many commercial CFD codes, like Fluent or CFX use these profiles for wind flow simulations [32]. They will be used throughout the present work.

In OpenFOAM each flow variable of the k- $\varepsilon$  model, velocity ( $U$ ), turbulent kinetic energy ( $k$ ), TKE dissipation rate (*epsilon*), pressure ( $p$ ) and turbulent viscosity (*nut*), has a dictionary inside the 0 folder where the boundary conditions and initial values for that variable at each patch of the domain must be specified.

## Pressure

Pressure boundary conditions can be considered, to some extent, as static and independent of the site data and the other variables. The pressure BC, have a twofold objective, on one hand they set the flow direction through a pressure difference between inlet and outlet patches; on the other hand, they guarantee the no-flux through any other patch. This is achieved by imposing a 0 gradient condition at every patch except the outlet where a pressure value equal to that of the atmospheric pressure on that location is set. In this work this value is 1 atm in every simulation.

Patch	OF Entry	Value	Initial Value
<b>sides, top, bottom, inlet</b>	zeroGradient	-	-
<b>outlet</b>	uniformFixedValue	1atm	-

Table 4.1: Pressure boundary conditions

## Outlet

Here a 0 gradient condition is applied to  $k$  and  $\varepsilon$  using the OF entry `zeroGradient`. For the velocity, this same condition is usually imposed using the `inletOutlet`, which applies a 0 gradient condition to the velocity components normal to the boundary and a fixed value defined by the user, which is set to (0,0,0), to the rest. As for  $nut$ , it must be determined out of the values of the other variables using Eq.4.4. This can be indicated through the entry `calculated`. This entry requires an initial guess, which was set to 0 by default.

Variable	OF Entry	Value	Initial Value
$k, \varepsilon$	<code>zeroGradient</code>	-	-
$U$	<code>zeroGradient/inletOutlet</code>	-(0,0,0)	-/InternalField
$nut$	<code>calculated</code>	-	0

Table 4.2: Outlet patch boundary conditions.

## Sides

At the two lateral patches of the domain a symmetry boundary condition is imposed for all flow variables, except turbulent viscosity, using the entry `symmetry`. This entry imposes a 0 gradient for scalars variables, mirrors the tangential components of the vector variables and assigns a 0 value to the rest [69]. It seems the entry `slip` would yield the same results, being the only difference the way OF treats both BC in the pre- and post-processing. Whereas `slip` can be applied to *wall* patches and therefore wall post processing applications can be used on it, `symmetry` requires the patch to be defined as *symmetry* in the *polyMesh* folder. Finally, the  $nut$  is calculated out of the values of the other variables.

Variable	OF Entry	Value	Initial Value
$k, \varepsilon, U$	<code>slip/symmetry</code>	-	-
$nut$	<code>calculated</code>	-	0

Table 4.3: Side patches boundary conditions.

## Top

In some works on CFD simulations, a zero-shear stress condition is applied to every variable at the top patch through the `slip` entry to prevent the formation of a boundary layer [56]. However, in other works fixed values for  $k$ ,  $\varepsilon$ ,  $\mu_t$  and  $U$  are applied to create a balance between the upper boundary driving shear stress and the retarding shear stress at the surface and thus conserve the inlet profiles along the whole domain, ensure horizontal homogeneity and prevent them from decaying [32, 102]. Both options were tested, and the results compared, resulting that, imposing fixed values at the top yield stable profiles throughout the fetch, whereas, with the `slip` condition they decayed. For this reason, it was decided to apply fixed values at the top patch, using those of the inlet profile at the same height for that purpose

Variable	OF Entry	Value	Initial Value
$k, \varepsilon, U, nut$	<code>uniformfixedvalue</code>	Eq.4.1,4.2,4.3,4.4	-

Table 4.4: Top patch boundary conditions.

## Inlet

Here the profiles derived in [102] from the governing Eq.4.1,4.2,4.3,4.4 are imposed for  $U$ ,  $k$  and  $\varepsilon$ . For  $U$  and  $\varepsilon$  OpenFOAM offers two entries which correspond to the mentioned profiles, `atmBoundaryLayerInletVelocity` and `atmBoundaryLayerInletEpsilon` respectively. A constant value is set for TKE with the `uniformfixedValue` entry. These entries require additional information, like a friction velocity or a reference height.

Variable	OF Entry	Value	Initial Value
$k$	<code>uniformfixedValue</code>	Eq.4.2	<code>turbulentKE</code>
$\varepsilon$	<code>atmBoundaryLayerInletEpsilon</code>	Eq.4.3	<code>turbulentEpsilon</code>
$U$	<code>atmBoundaryLayerInletVelocity</code>	Eq.4.1	<code>InternalField</code>
$\nu_{\text{t}}$	<code>calculated</code>	-	0

Table 4.5: Inlet patch boundary conditions.

Together with the entries, initial guesses must be provided for  $k$ ,  $\varepsilon$  and  $U$ . Although these initial guesses may not alter the result of the simulation, if correctly chosen, they may speed-up the convergence. An initial value of (0,0,0) is commonly used for the velocity, while the constant value given by Eq.4.2 is used for  $k$ .

As for  $\varepsilon$ , whereas ANSYS Fluent Users Guide ([95] page-744) recommends using a value so that the resulting  $\nu_{\text{t}}$  is large, around two orders of magnitude, compared with the molecular viscosity. PHOENICS propose an initial value based on the mixing length (Eq.4.8).

$$\varepsilon = \rho C_{\mu} \frac{k^2}{\mu} \left( \frac{\mu_t}{\mu} \right)^{-1} \text{ with } \left( \frac{\mu_t}{\mu} \right)^{-1} \approx 0.01 \quad (4.7)$$

$$\varepsilon = \rho C_{\mu}^{3/4} \frac{k^{3/2}}{l} \quad (4.8)$$

## Ground

This patch corresponds to the terrain and is the only solid boundary of the domain. Here,  $U$  is set to 0, and the wall functions introduced in Ch.2 are applied to the other variables, through the corresponding OF entries. These wall functions are consistent with the governing equations of the flow under the assumption  $u_{\tau} = u_{\text{ABL}}$  [68]. As it was explained in Ch.2, the surface roughness effects are introduced through the turbulent viscosity expression (Eq.2.45), which requires a  $z_0$  value to be specified. The constant values used in the entries  $E$ ,  $\kappa$  and  $C_{\mu}$ , also must be specified. As for the initial guesses, the same values used at the inlet are assigned to  $k$  and  $\varepsilon$ , whereas 0 is used for  $\nu_{\text{t}}$  by default.

Variable	OF Entry	Value	Initial Value
$k$	<code>uniformfixedValue</code>	-	0
$\varepsilon$	<code>atmBoundaryLayerInletEpsilon</code>	Eq.2.37	<code>turbulentEpsilon</code>
$U$	<code>uniformfixedValue</code>	0	-
$\nu_{\text{t}}$	<code>nutkAtmRoughwallFunction</code>	Eq.2.45	0

Table 4.6: Ground patch boundary conditions.

## Internal Field

Here, values for each variable are assigned to the rest of the domain's cells. They will be used as initial values in the iterative process. They were chosen taking the OF tutorials as a reference.

Variable	InternalField
$k$	turbulentKE
$\epsilon$	turbulentEpsilon
$U$	(0,0,0)
$\nu_t$	0
$p$	1

Table 4.7: Internal field boundary conditions.

## Include

In *constant/include*, one or more dictionaries, such as *ABLconditions* (see Fig.4.5) and *inletConditions* can be written to store the values of the parameters required by the entries used to set the BC in the variables dictionaries.

```
Uref          13.9769;
Href          500;
z0            uniform 0.0003;
turbulentKE   0.928;
windDirection (1 0 0);
zDirection    (0 0 1);
zGround       uniform 0.0;
```

Figure 4.5: Atmospheric boundary layer conditions dictionary

These values can be called in the variables dictionaries using the entry *#Include*, so that it won't be necessary to change the value in each file but only once in the *include* folder.

### 4.2.3 Fluid Properties

In OpenFOAM air density and viscosity are introduced by means of the kinematic viscosity  $\nu$  in the dictionary *constant/transportProperties* (see Fig.4.6). The values of air density and dynamic viscosity at sea level are  $1.229\text{kg/m}^3$  and  $1.73\text{e-}5\text{ N.s/m}^2$  respectively [46]. However, it may be necessary to change them for simulations at higher altitudes. Inside the same dictionary, the mechanical characteristics of the fluid used must be specified at the entry *transportModel*. In this case the fluid is the air and it will be considered as a *newtonian* fluid.

```
transportModel Newtonian;

nu              nu [0 2 -1 0 0 0 0] 1.4076e-05;

//mu=1.73e-05kg/ms;density=1.229kg/m3;
```

Figure 4.6: Fluid properties dictionary.



#### 4.2.4 Transport Properties

In the *constant/RASProperties* dictionary, the model used, standard k- $\epsilon$  model in the present work, as well as the values of the coefficients of the governing equations must be specified. In absence of measurements to calculate them, the following values, used by the DTU for EllipSys3D are assumed:

$C_\mu[-]$	$C_{\epsilon 1}[-]$	$C_{\epsilon 2}[-]$	$\sigma_k[-]$	$\sigma_\epsilon[-]$
<b>0.03</b>	1.21	1.92	1.00	1.30

Table 4.8: SKE Coefficients.

#### 4.2.5 Canopies

For the simulation of wind flows through, above and around forested areas the porous approach explained in Ch. 2 is applied. To simulate flows through porous domains OF offers the *porousSimpleFoam*, a modified version of SIMPLE solver [104,105] which is based on the *Darcy-Forchheimer Law* for flow through porous mediums and therefore, only considers the effects of porosity on the momentum equation like FLUENT [95].

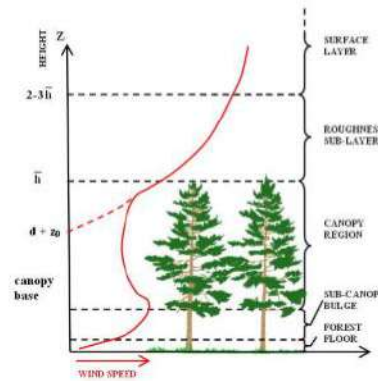


Figure 4.7: Wind profile entering a forest.

The introduction of extra terms in the  $k$  and  $\epsilon$  equations to account for the effects of the porous cells on these variables would imply the modification of the code, which is beyond the scope of the present work. A similar approach based on the characterization of the forested areas through a porosity value and the modification of the momentum equation is applied by the wind assessment software *WindSim*, which is based on the PHOENICS CFD code [56,106].

#### The Darcy-Forchheimer Law

According to [107] two different experimental laws can be applied for the momentum balance of a flow through a porous media, depending on the flow's Reynolds number.

$$Re = \frac{InertiaForce}{ViscousForce} = \frac{\rho v L}{\mu} \quad (4.9)$$

In Eq. 4.9  $L$  is the length scale. In slow flows with low Reynolds numbers [108,109] where inertial effects are considered negligible, the modelling of flows through porous media is typically based on the so-called *Darcy Law* [108,109,110].

This law states that, the pressure drop suffered by a fluid flowing through an isotropic porous media under negligible gravitational effects can be expressed as follows [111]:

$$\nabla p = \frac{-\mu}{K} U \quad (4.10)$$

Here  $K$  [ $\text{m}^2$ ] is the intrinsic permeability tensor of the porous media, which represents its ability to let the fluid to flow through it [112].

When velocity increases and the Reynolds number grows, inertial effects begin to prevail over the viscous effects. This was observed by P. Forchheimer, which included an extra non-linear term in the Darcy Law to account for these effects [113] and thus developed the so called *Forchheimer* or *Darcy-Forchheimer Law*:

$$\nabla p = \frac{-\mu}{K} U - F \rho |U| U \quad (4.11)$$

Here coefficient  $F$  is the so called Forchheimer tensor. If compared with the general momentum loss term expression (Eq.2.48), an apparently simple relation can be established between  $K$  and the viscous resistive coefficient  $C_1$ , and between  $F$  and  $C_2$ , the *inertial* resistive coefficient:

$$C_1 = \frac{\vartheta}{K} \quad C_2 = F \quad (4.12)$$

However, although  $C_2$  has been commonly related with the leaf area density ( $\alpha$ ), and the drag coefficient ( $C_D$ ) [125], after a long literature research no wide accepted expression has been found for the relation between  $C_D$ ,  $F$ ,  $K$  and the other parameters used to describe a forested area such as porosity ( $\beta$ ) or the leaf area index (LAI) [109,114]. Most of the expressions found have been experimentally obtained and differ greatly from one another.

## The Porous Region

In OF the information of the porous region must be specified in the *constant/porosityProperties* dictionary (see Fig.4.9). First, the region occupied by the porous cells is defined and then, its characteristics in terms of two parameters  $f$  and  $d$  are specified within the entry *DarcyForchheimerCoeffs*. Values for each of them must be provided for each of the space directions given by a coordinate system to be specified under the *coordinateSystem* entry.

```

porosity1
{
    type            DarcyForchheimer;
    active          yes;
    cellZone        porosity;

    DarcyForchheimerCoeffs
    {
        d    d [0 -2 0 0 0 0 0] (500 -1000 -1000);
        f    f [0 -1 0 0 0 0 0] (0 0 0);

        coordinateSystem
        {
            type        cartesian;
            origin       (0 0 0);
            coordinateRotation
            {
                type        axesRotation;
                e1           (-1 0 0);
                e2           (0 0 1);
            }
        }
    }
}

```

Figure 4.8: Porosity properties dictionary.

The relation between these parameters and those of the Darcy-Forchheimer Law can be obtained from the description of the OF porous media model given in source code files `DarcyForchheimer.T.H` and `DarcyForchheimerTemplate.T.C`, which can be found in GitHub (<https://github.com>) and it can be expressed as follows:

$$d = \frac{1}{K} \quad f = 2F \quad (4.13)$$

#### 4.2.6 The groovyBC utility

It is not within the scope of the present work to make changes in the code but use the tools and utilities provided by OF. However, this makes impossible to solve some problems related to the way the code is written and limits the application of OF. To mitigate this, OF users developed the *groovyBC* library which enables the user to set non-uniform boundary conditions without programming, by specifying values, gradients and formulas explicitly. For instance, instead of using the `atmBoundaryLayerInletVelocity` for the velocity at the inlet, an equivalent boundary condition may be implemented using *groovyBC* as follows:

```

inlet
{
    type groovyBC;
    value $internalField;
    variables "u_f=0.331;kf=0.41;z0=0.2;";
    valueExpression "(u_f/kf)*(log((pos().z+z0)/z0))*vector(-1,0,0)";
}

```

Figure 4.9: *groovyBC* velocity entry at inlet patch.

### 4.3 Solving

OpenFOAM offers a wide range of solvers and discretization schemes for different physical phenomena and the different equation terms. In the present work simplicity and low computer power requirements will be favoured (see example in [Annex 3](#)).

### 4.3.1 Discretization Schemes

In the *system/fvSchemes* dictionary the terms of the governing equations are classified according to their mathematical nature and each of them is assigned a discretization scheme (see [69] for a list of all the discretization schemes offered).

Convection	Diffusion
$\rho \vec{U}u, \rho \vec{U}v, \rho \vec{U}w, \rho \vec{U}k, \rho \vec{U}\epsilon$	$\mu \nabla u, \mu \nabla v, \mu \nabla w, \frac{\mu_t}{\sigma_\epsilon} \nabla \epsilon, \frac{\mu_t}{\sigma_k} \nabla k$

Table 4.9: Governing equations terms.

For the diffusion terms the *central differencing* (CDS) scheme is the most commonly used [68,61,115,116]. However, it is considered unsuitable for the convective terms, for it doesn't fulfill the *transportiveness* condition. For these terms the *first order upwind* scheme is the first option. It is robust, easy to use and provide quite stable and relatively accurate solutions [66], which makes it one of the most widely used schemes for convective terms [117, 61,116].

Nevertheless, its use is not recommended in complex problems [69] for it doesn't provide good results for high Reynolds numbers when the flow isn't aligned with the mesh. Although it is still a common practice to use it at the beginning of the convergence process and substitute it by a higher-order scheme after a few iterations [95]<sup>10</sup>. In the present work second order schemes, *second order upwind* [92], and QUICK [47,56, 119,118] are used. In general, they provide, despite their higher computing power requirements, more accurate results, although these might be unstable. Simulations of the wind flow over a 2D-symmetric hill were carried out using different combinations of these three methods and the results compared.

### 4.3.2 Linear solvers

A solver is assigned to each flow variable in the *solvers* sub-dictionary within the *fvSolution* dictionary. The solvers distinguish between symmetric (e.g. Pressure) and asymmetric matrices (e.g. velocity), which depends on the term of the equation being solved (see [69] for a list of the solvers and smoother offered by OF).

For the momentum and turbulence quantities the *smoothSolver* using the *GaussSeidel* smoother is recommended in [60] and [69], as a most reliable option, although it may be too slow. As for pressure, the *GAMG* is according to [69,47] the optimal choice. Here *GaussSeidel* is used again as smoother for the sake of simplicity although others might yield a faster convergence [120].

Variable	Solver	Smoother
$p$	GAMG	GaussSeidel
$U, k, \epsilon$	smoothSolver	GaussSeidel

Table 4.10: Variable solvers.

10. The ASME *Journal of Fluid Engineering* (JFE) (<http://fluidsengineering.asmedigitalcollection.asme.org/journal.aspx>) has indeed a policy of not publishing results from first order approximations [100].

Several simulations of the wind flow over a 2D-Hill were carried out using different solvers and smoothers without changing the interpolation schemes (see [Annex 1](#)). While no big differences could be seen in the results, the pairs `smoothSolver-GAMG` and `GAMG-GAMG` showed the lowest convergence times. The use of different smoothers yielded no variations either in the results or in the computing time. In the present work `GAMG` is used for pressure whereas the `smoothSolver` with `GaussSeidel` is used for the other variables.

### 4.3.3 Application Solvers

In the present work, SIMPLE is used in every simulation. This must be specified within the `system/fvSolution` and `system/controlDict` dictionaries. The SIMPLE algorithm only applies one correction to the pressure field by default [\[66\]](#) but OF offers the possibility to add an extra number of corrections for meshes with non-orthogonal cells, to be specified in the same dictionary.

### 4.3.4 Relaxation

The relaxation factors of each flow variable are specified in the `relaxationFactors` sub-dictionary within the `system/fvSolution` dictionary. The relaxation factors depend on the flow characteristics and, in practice, they must be determined on a trial-and-error basis. It is also usual to change them on the course of the simulation keeping them low at the beginning when the strongest values oscillations may take place and increasing them as the solution converge to speed it up. In [\[95\]](#) a default value of 0.5 for each variable is recommended. In the present work, as in [\[116,118\]](#), the following relaxation factors are used:

Variable	Relaxation Factor
$p$	0.3
$U, k, \epsilon$	0.7

Table 4.11: Relaxation factors.

### 4.3.5 Tolerance and Convergence

After each iteration of the *linear* or the *application* solver the residuals are calculated, if any of the following criterions is reached OF will put an end to the iteration process:

- $Residual < Tolerance$
- $[Current\ Residual/Initial\ Residual] < Relative\ Tolerance$
- $N^o\ Iterations > Max.N^o\ Iterations$ .

Tolerance and relative tolerance values are defined for each variable in the `fvSolution` dictionary and a residual control value is specified for SIMPLE. The maximum number of iterations is set to 1000 by default [\[69\]](#).

Values ranging from  $10^{-5}$  up  $10^{-8}$  for the tolerance of the *linear* solvers are suggested in different works [\[117\]](#). In the present work, the tolerance of the linear solvers for each variable is set to  $10^{-8}$  and the relative tolerance is set to 0.1. As for the SIMPLE tolerance, in [\[95,97,121\]](#) a value of  $10^{-5}$  is suggested for all the variables whereas in [\[47\]](#) a value of  $10^{-2}$  is proposed for pressure and  $10^{-3}$  for the others. In the present work, values of  $10^{-3}$  for pressure and  $10^{-4}$  for the other variables are applied in every case.

### 4.3.6 Time Step

Finally, in the *system/controlDict* dictionary the specific libraries to be used for the simulation must be declared as well as the start time, end time, time step and information about the write format, precision and compression. This information determines, together with the mesh resolution, the *Courant number* of the simulation defined as follows:

$$Co = \frac{\Delta t / U}{\Delta x} \quad (4.14)$$

Where  $\Delta t$ ,  $\Delta x$  and  $U$  are respectively the time step, the cell length and the velocity through that cell. In [\[69\]](#) it is said that a  $Co < 1$  is required to achieve temporal accuracy and numerical stability.

# Chapter 5

## Simulations Tests

In this chapter the influence of some domain parameters related with terrain complexity on horizontal homogeneity (HH) is tested to identify those settings which may yield the best results. Then the wind flow through and at the edge of a forested area is simulated using different values for the additional terms of the momentum equations and the results compared with experimental data.

The guidelines, boundary conditions and settings described in [Ch.4](#) are applied in each simulation. The same happens with the discretization schemes, linear solvers and relaxation and convergence values, unless stated differently.

### 5.1 Horizontal Homogeneity (HH)

As it was said earlier and has been reported by different authors [[32,68,102](#)], the inlet and boundary conditions described in [Ch.4](#) don't automatically yield horizontal homogeneity. In [Fig.5.1](#), the  $k$  profiles of wind flow over a plain and homogeneous terrain at different distances from the inlet are presented. As it can be observed in [Fig.5.1](#), the TKE profile changes over the fetch. This streamwise gradient may jeopardize the results of a simulation.

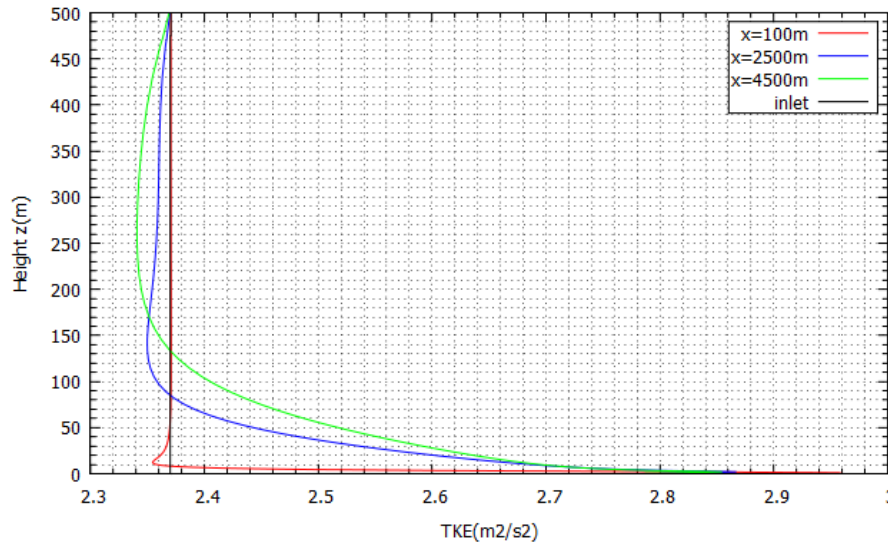


Figure 5.1: TKE profiles on plain terrain.

The relation between the grid resolution and the roughness index may be the cause of these gradients, according to [[68](#)]. To assess to what extent these gradients may be a result of the grid features and identify which values and settings may reduce or eliminate them, the following grid features will be tested:

- First node position
- Vertical grading
- Aspect Ratio

For this purpose, following a common procedure when studying turbulent flows [68,58], simulations of a 2D-flow are carried out in simple domain of 5000x100x50m with 500 cells in the x-direction, 5 in the y-direction and 50 in the z-direction created. In Table 5.1 the simulations settings used for each test are presented.

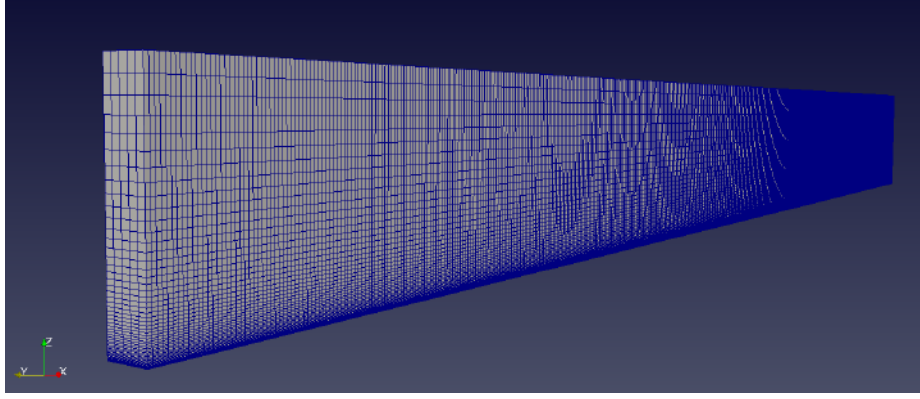


Figure 5.2: Test domain.

$H_{REF}(m)$	$U_{REF}(m/s)$	$Z_P (m)$	$K(m^2/s^2)$	$R_{EXP}$
500	16.91	0.5	2.37	1.076

Table 5.1: Test simulations settings.

For any variable  $\phi$ , changes along the fetch are expressed as relative errors respect to the inlet values:

$$e(x) = 100 \left| \frac{\phi(x) - \phi_{(inlet)}}{\phi_{(inlet)}} \right| \quad (5.1)$$

### 5.1.1 First Node Position

The location of the first node of the domain above the ground is a matter of utmost importance for it affects the application of the wall functions. These are a few general guidelines:

- Thickness of the first cell above the ground  $< 1m$  [56,68,58].
- First node of the domain above the ground must be within the inertial sublayer, that means  $y^+ > 30$  [67,102].
- The first node above the ground must fulfil  $y_p > k_s$ , for it wouldn't be physically meaningful to have cells whose centres are below the roughness height [60,68].

Whereas the second condition is easy to implement in wind simulations in the ABL, there may be a conflict between the first and the third requirements. On one hand the higher the  $y_p$  gets the easier it is to fulfil  $y_p > k_s$  for different  $k_s$  values, and the more accurate the approximation made in Eq.2.42 will be.



On the other hand, high  $k_S$  would require coarse meshes with the consequent resolution loss. These conditions also pose an additional problem when areas with different  $z_0$  within the same domain are considered.

To establish the optimum  $z_P/z_0$  or  $z_P/k_S$  ratio several tests are carried out with a constant  $z_P$  and  $z_0$  values varying between 0.001 and 0.75m. Table 5.1 and 5.2 show, respectively, the general settings and the specific to this case. The relative errors of  $U_x$ ,  $k$  and  $\varepsilon$  at a 4500km distance from the inlet are presented in Fig.5.3, 5.4 and 5.5.

$Z_0(\text{m})$	$K_S(\text{m})$	$Z_P/K_S$	$Z_P/Z_0$
0.001	0.03	16.66	500
0.005	0.15	3.33	100
0.01	0.3	1.66	50
0.05	1.5	0.33	10
0.1	3	0.17	5
0.5	15	0.03	1
0.75	22.5	0.02	0.66

Table 5.2: First node position simulations values.

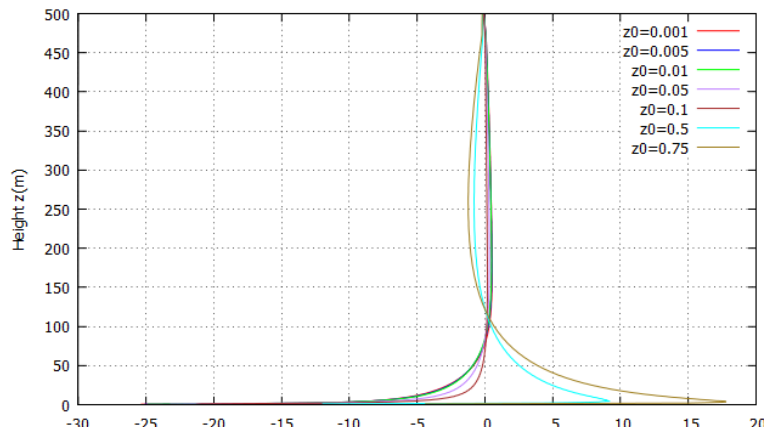


Figure 5.3: First node position simulations Ux relative errors.

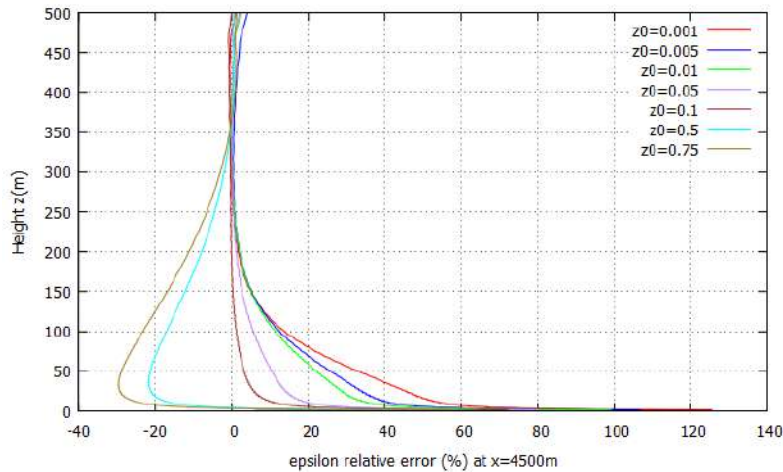


Figure 5.4: First node position simulations epsilon relative errors.

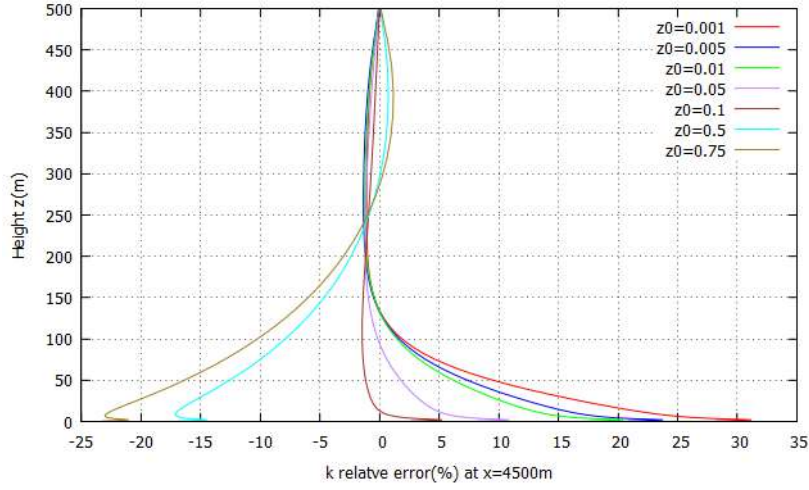


Figure 5.5: First node position simulations TKE relative errors.

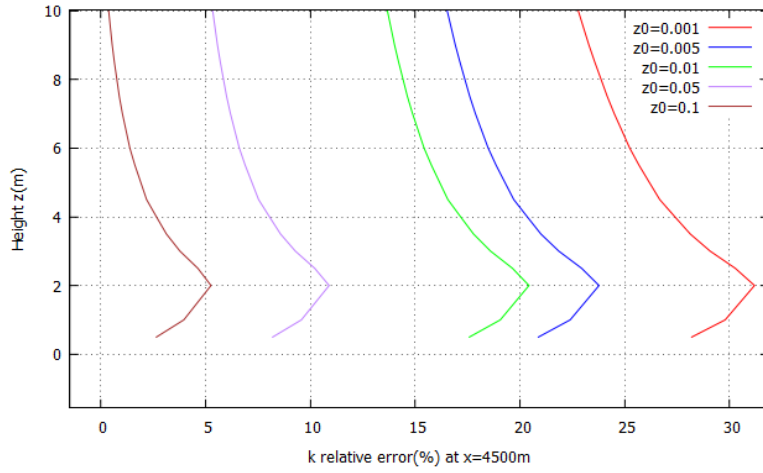


Figure 5.6: First node position simulations TKE relative errors (detail).

In every case the maximum value for the relative error is found at a peak or overshoot near the bottom of the domain (see Fig. 5.6) at the height of the second cell of the domain above the ground. Then it decreases with the height, becoming nearly 0 at the top of the domain. In all tests  $\varepsilon$  shows the highest relative errors, reaching values up to 100% near the bottom for  $z_0=0.001$ , whereas  $U_x$  shows the lowest, under 25% in every case.

In general,  $z_0=0.001\text{m}$  shows the highest relative errors, whereas, in every case the lowest relative errors correspond to  $z_0=0.1\text{m}$ , with  $z_p/z_0=5$ . In this case the  $k$  and  $U_x$  relative errors fall below 5% before reaching 5m height, whereas the  $\varepsilon$  relative error remains above 5% until around a height of 20m. However, here the condition  $k_S \sim 30z_0$ ,  $y_P > k_S$  is not fulfilled. Nevertheless, other values have been proposed for the  $k_S$ -  $z_0$  relation, like  $k_S \sim 7.5z_0$  or  $k_S \sim z_0$  ([58] pag.104). If  $k_S \sim z_0$  then  $y_P > k_S$  would also be fulfilled for  $z_0=0.1\text{m}$  and this would also explain the abrupt change in the shape of the relative error when  $z_0=0.5\text{m}$  or larger, for at those values  $y_P < k_S$ .

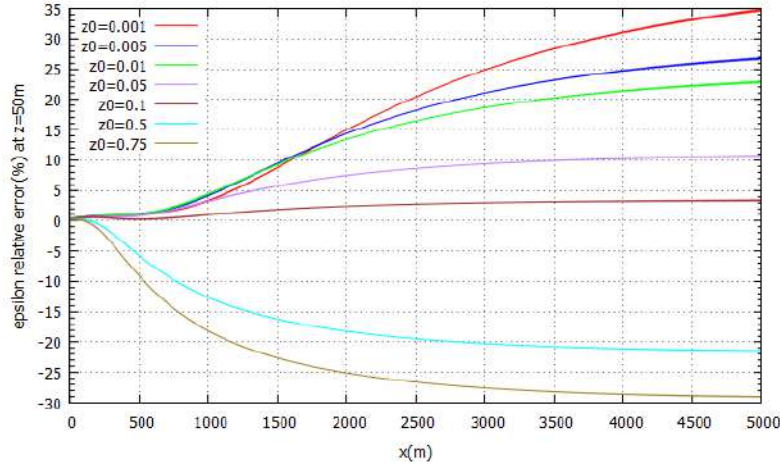


Figure 5.7: First node position simulations epsilon relative errors at 50m.

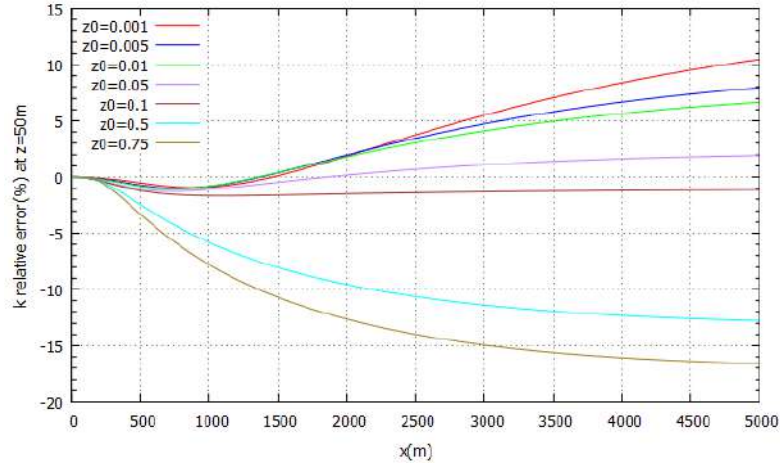


Figure 5.8: First node position simulations TKE relative errors at 50m.

In Fig.5.7 and 5.8 the relative error variation along the fetch at 50m height is presented. It can be observed that the relative error grows along the fetch for every  $z_0$  value. This is evident for  $k$  and  $\varepsilon$ , which show the highest variations along the fetch, while  $U_x$  absolute relative error is never above 4%. The  $z_0=0.1$ m case shows the lowest relative error, under 5% for all the variables.

### The near-ground overshoot

In all the tests, the highest relative error values for each variable were always at the bottom of the domain in the shape of a peak. A similar peak is described in [102], where a failure in the discretization process used to calculate the TKE production term is blamed for it. According to this explanation the TKE production term  $P_K$  is artificially multiplied by a factor  $\alpha$  (Eq.5.2) in the discretization process and thus the  $P_K=\rho\varepsilon$  equilibrium is broken.

$$\alpha = \left\{ \frac{1}{1 - \left( \frac{\Delta z}{2z} \right)^2} \right\}^2 \quad (5.2)$$

This explanation predicts the appearance of a peak in the TKE profile at the height of the second cell, where  $\Delta z/2z$  is maximal. This coincides with the results shown in Fig.5.6.

To test this explanation several tests were made changing  $z_p$ , and thus the position of the second cell keeping the relation  $z_p/z_0=5$ . Except for the number of cells in the vertical direction the settings in Table 5.1 were used in all the simulations:

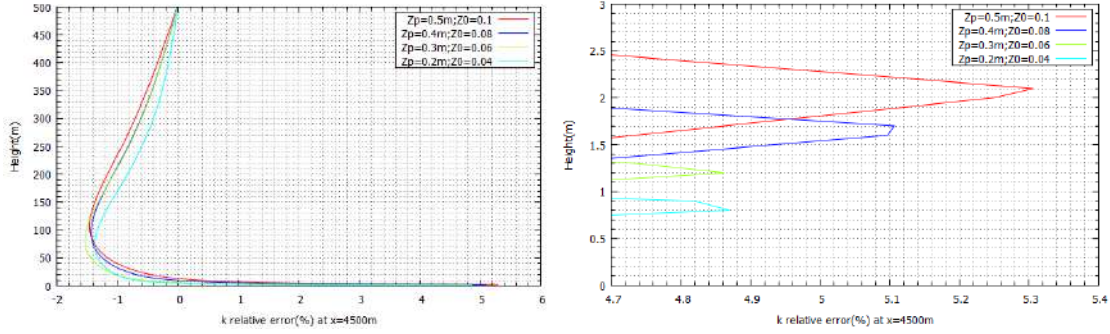


Figure 5.9:  $z_p/z_0=5$  test TKE relative error, a) whole profile b) near-ground detail.

Plots in Fig.5.9 b) show how the near-wall peak moves downwards with  $z_p$ . It also can be observed that the maximum relative error also decreases with  $z_p$  and gets closer to 0% with the height.

In [122] a higher vertical resolution is suggested as a solution; however, this would increase the number of cells and thus the computational power requirements. The solution suggested in [102] implies changing the code, what is out of the scope of this work. However, as it has been shown, an appropriate choice of the parameters values may reduce the TKE overshoot and make it vanish after a few meters.

### 5.1.2 Vertical Grading

A vertical expansion or cell-to-cell ratio,  $R_{exp}$ , between 1 and 1.2 is recommended in [95]. Four simulations using meshes with different  $R_{exp}$  values between 1 and 2 (see Table 5.3) were carried out to test the effects of the expansion ratio on the horizontally homogeneity. A  $z_0=0.05m$  and  $z_p=0.25m$  were applied.

N°z	$R_{EXP}$	Top Cell Thickness (m)	N° Cells
50	1,096	44,144	125000
75	1,055	26,627	187500
100	1,037	18,346	250000
150	1,0207	10,641	375000

Table 5.3: Vertical grading simulations values.

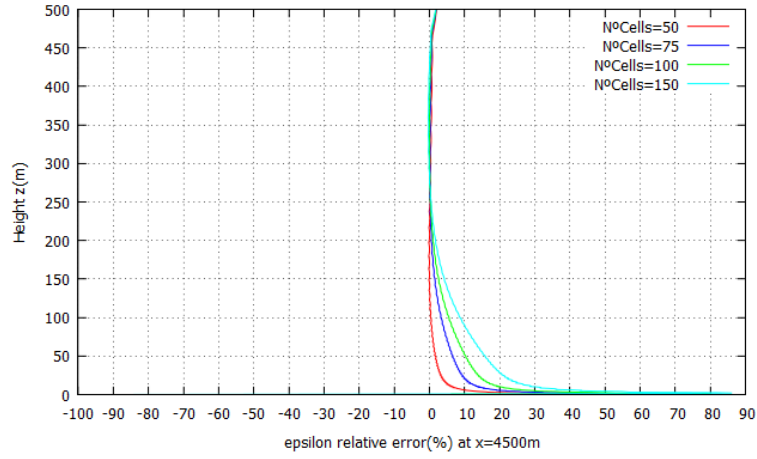


Figure 5.10: Vertical grading simulations epsilon relative errors.

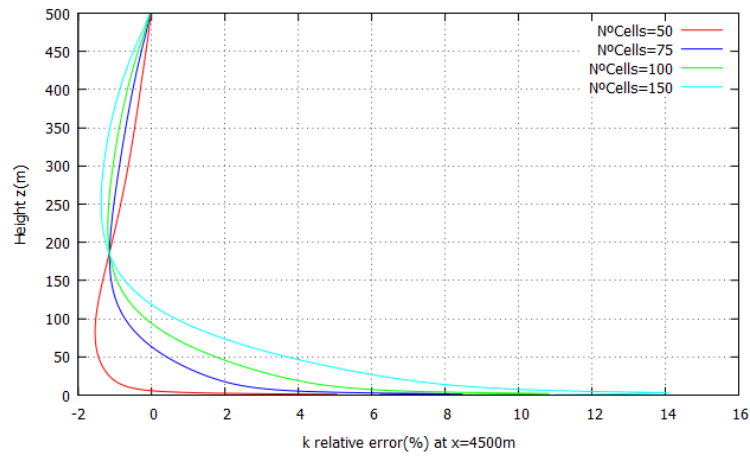


Figure 5.11: Vertical grading simulations TKE relative errors.

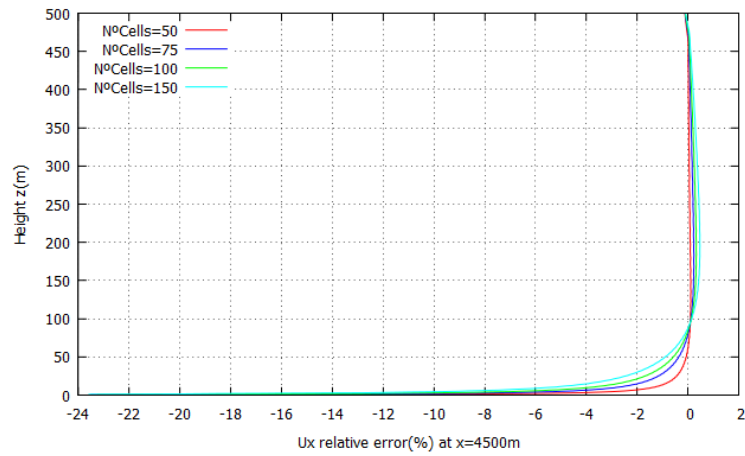


Figure 5.12: Vertical grading simulations Ux relative errors.

Again, the highest relative errors are at the bottom of the domain, and decrease with the height. Fig.5.10, 5.11 and 5.12 show that the length of these peaks grows with the number of cells, being the maximum relative error for  $\varepsilon$  near the bottom never under 50% and reaching values up to 80% for  $R_{EXP}=1.0207$ , while for  $k$  and  $U_x$  the maximum relative errors values are between 5% and 14%, and 20% and 23% respectively.

### 5.1.3 The Aspect Ratio

It is defined as the quotient between the maximum and minimum values of the distance between the cell centre and any of the face centroids or nodes ( $\delta x$ ,  $\delta y$  or  $\delta z$ ).

$$R_{Asp} = \frac{\max(\delta x, \delta y, \delta z)}{\min(\delta x, \delta y, \delta z)} \quad (5.3)$$

*High Aspect Ratio Cells* (HARC) can appear when meshing a complex terrain due to the need to achieve high near-wall resolution using the least possible number of cells. They may cause streamwise gradients and continuity problems. OF considers HARC all those cells with  $R_{Asp} > 1000$  [69].

Eight simulations with cells of different length in the x-direction ( $\Delta x$ ) were made to assess the effects of HARC on the HH. To change the maximum  $R_{Asp}$  without changing the number of cells in the x-direction 8 domains with different lengths were created (see Table 5.5). In every case the max.aspect ratio cells were the first above the ground.  $Z_P$  the resolution in both z and y-direction and  $Z_0$  were kept constant (see Table 5.4).

$H_{REF}(m)$	$U_{REF}(m/s)$	$Z_P (m)$	$Z_0 (m)$	$N^o_x$	$N^o_y$	$N^o_z$	$R_{EXP}$
500	16.91	0.25	0.05	500	5	50	1.096

Table 5.4: Aspect ratio simulations settings.

Length	5	25	50	75	100	125	135	140
Max. $R_{Asp}$	40.00	100.00	200	300	400	500	540	560

Table 5.5: Aspect ratio simulations values.

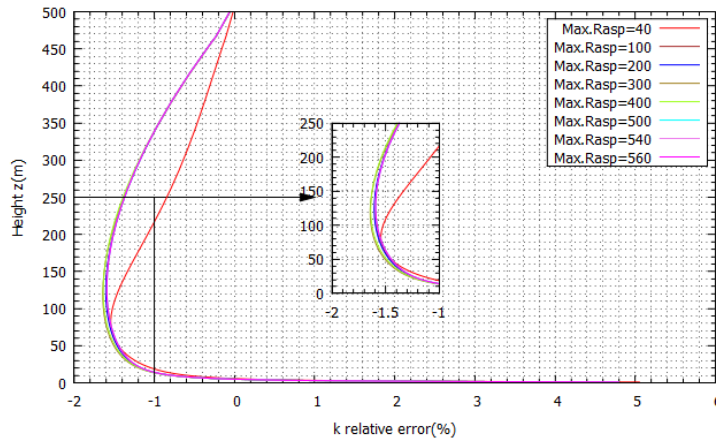


Figure 5.13: Aspect ratio simulations TKE relative errors.



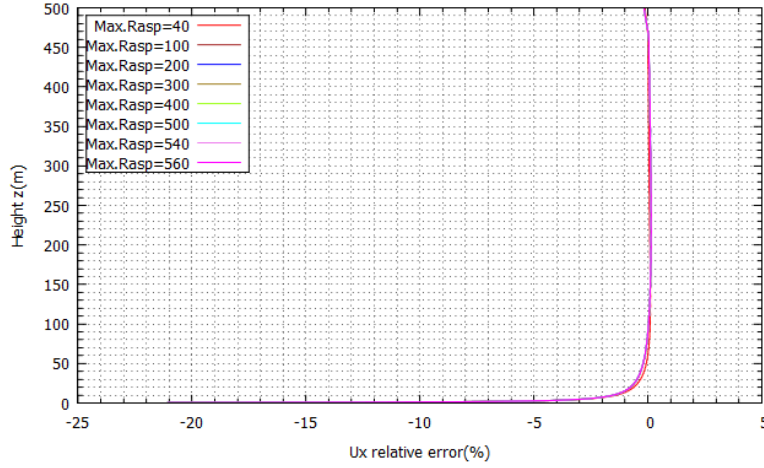


Figure 5.14: Aspect ratio simulations  $U_x$  relative errors.

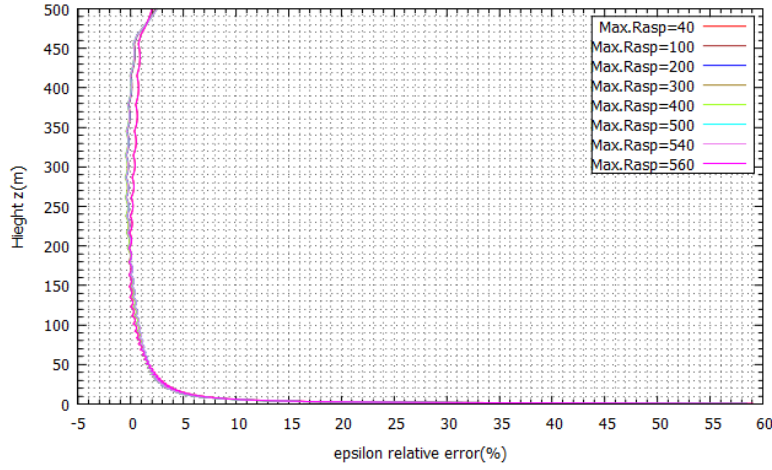


Figure 5.15: Aspect ratio simulations  $\epsilon$  relative errors.

The relative error of  $k$ ,  $\epsilon$  and  $U_x$  at 9/10<sup>th</sup> of the domain is presented in Fig. 5.13, 5.14 and 5.15. The maximum relative errors are in every case at a peak near the bottom and decrease with the height. Apparently, the  $R_{Asp}$  increase didn't produce any change on the relative error of any of the variables. However, for  $R_{Asp} > 560$ , the convergence process stops after around 60 iterations. This sudden stop was due to *time step continuity errors*, which is the OF denomination for conservation problems.

## 5.2 Obstacles

As no validation study for the use of a porous canopy model to represent forested areas in OF was found, it was decided to test and validate this model against real data, using a few different expressions for the extra terms in the momentum. This was made following the steps explained in [83], which used experimental data from [123] for this purpose. Unfortunately, only data for the wind flow entering a forest is provided and no experimental data was found for wind leaving a forest.

### 5.2.1 The Canopy

This campaign was carried out in a Sitka spruce forest at Harwood Forest, in Northumberland, England (55° 12.5' N 2° 02' W), surrounded by pasture, which extends uninterrupted for at least 2km in the prevailing wind direction (SW). Measurements were taken at four 15m high towers located upstream from the forest edge and inside the forest itself as it is shown in Fig. 5.16.

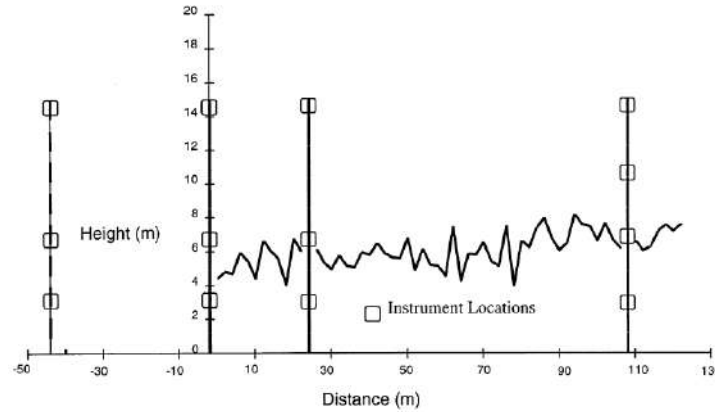


Figure 5.16: Sensors position [123].

Although the tree's heights range from 5.5m to 8m a 7.5m high ( $h$ ) uniform height for the canopy was assumed in the simulations [83]. The forest was assigned a 2.15 leaf area index (LAI) and an enhanced  $z_0$  of 0.563m, calculated according to [124], whereas that of the surrounding past was set to 0.0028m.

In [123] a drag coefficient value of 0.2 is considered. However, neither a leaf area density profile ( $\alpha$ ) nor a value for the porosity ( $\beta$ ) is given. In [83] a  $\alpha$  profile is proposed (Fig. 5.17a) but it cannot be applied in OF, for in the *porosityProperties* dictionary doesn't exist the possibility to impose the Darcy-Forchheimer coefficients  $f$  and  $d$  (Eq.4.13) changing values. To imitate the  $\alpha$  profile from [83] to some extent, the canopy was divided into three blocks and different  $\alpha$  values were assigned to each of them (Fig. 5.17 b). As for  $\beta$ , a value of 29% is assumed for the Spruce [106].

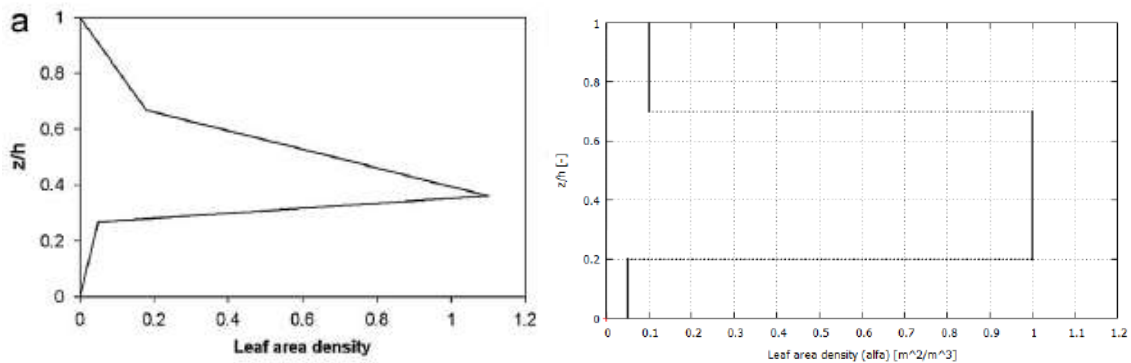


Figure 5.17: Leaf area density: a) real [83] b) modelled.



Different expressions have been proposed for the relation between these magnitudes and  $f$  and  $d$ . Of all the expression proposed in the literature four of the most used were chosen to create four different cases. A simulation in which the canopy is represented by the enhanced  $z_0=0.5625\text{m}$  was also carried out. In Table 5.6, the values assigned to the coefficients of Eq.5.4 and Eq.5.5 in each case are presented.

$$\nabla p = \frac{-\mu}{K} U - F\rho|U|U \quad (5.4)$$

$$S_u = -\rho C_1 \vec{U} - \rho C_2 |U| \vec{U} \quad (5.5)$$

Model	C1[s <sup>-1</sup> ]	C2(=F) [m <sup>2</sup> /m <sup>3</sup> ]	K[m <sup>2</sup> ]	d(=1/K)[m <sup>-2</sup> ]	f(=2F) [m <sup>-1</sup> ]
1 [83,125]	-	$\alpha C_d$	-	0	(0.01,0.2,0.02)
2 [119]	-	$\frac{1}{4} \left[ \frac{3}{2\beta} - 1 \right]^2$	-	0	4.3522
3 [83,125, 106]	$\frac{\vartheta}{K}$	$\alpha C_d$	$C \frac{\beta^2}{1 - \beta^2}$	2356.51	(0.01,0.2,0.02)
4 [119,106]	$\frac{\vartheta}{K}$	$\frac{1}{4} \left[ \frac{3}{2\beta} - 1 \right]^2$	$C \frac{\beta^2}{1 - \beta^2}$	2356.51	4.3522

Table 5.6: Canopy models' coefficients.

## The Domain

A simulation domain of 2000x100x500m domain, with a 100m long canopy representing the forest, beginning at 300m downstream from the inlet was created (Fig.5.18 and 5.19). It was divided into 5 cells in y-direction, 200 in the x-direction and 71 along the z-axis. In [83] the use of 2.5m long cells near the forest edge and cells with increasing length towards the inlet and outlet patch, up to 7m and 20m respectively is recommended as well as a 0.2m first cell thickness ( $z_p=0.1\text{m}$ ) and a  $R_{EXP}=1.066$ . These recommendations were followed except for  $z_p$ , whose value was set to 0.015m according to the previous findings.

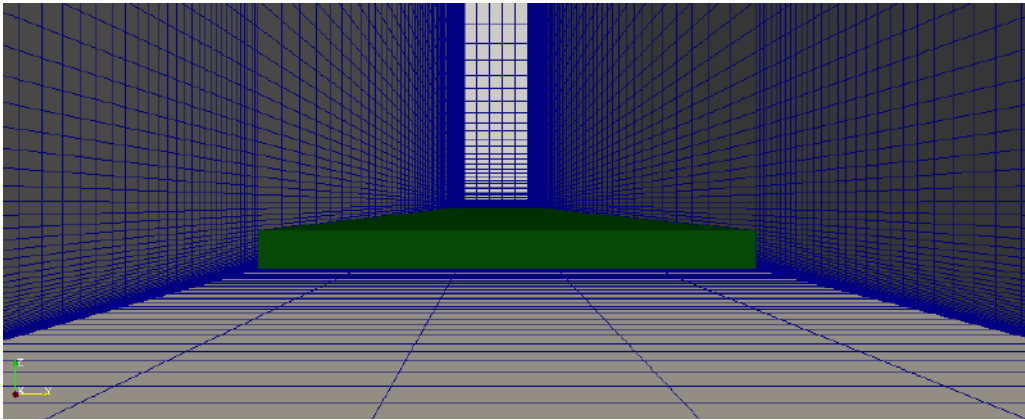


Figure 5.18: Canopy model simulations domain and canopy region.

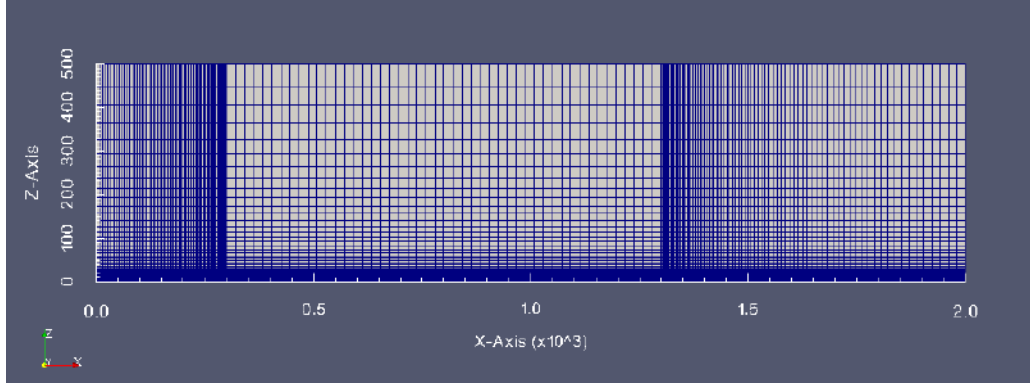


Figure 5.19: Canopy model simulations domain.

### Boundary Conditions

The same BC used in the previous test were applied in this case, except at the forest ground. According to [125], if the forest is dense enough, as it's in this case, the shear stress on the ground is negligible compared with the drag forces, and therefore a full-slip boundary condition can be imposed using the symmetry entry. The velocity measured by at Mast 1, 6.28m/s is used as reference for the entering wind profile. For the calculation of the experimental TKE values, only  $\sigma_u/u^*$  and  $\sigma_w/u^*$  values were available. As in [83],  $\sigma_v/u^*$  was assumed to be 2.1. Finally, the QUICK interpolation scheme, recommended in [83] was applied to all the convective terms.

$H_{REF}(m)$	$U_{REF}(m/s)$	$Z_P(m)$	$K(m^2/s^2)$	$R_{EXP}$
15	6.28	0.015	0.52	1.066

Table 5.7: Canopy model simulations settings.

In Fig.5.20 the results of the normalized velocity at the masts positions are presented together with the experimental measurements.

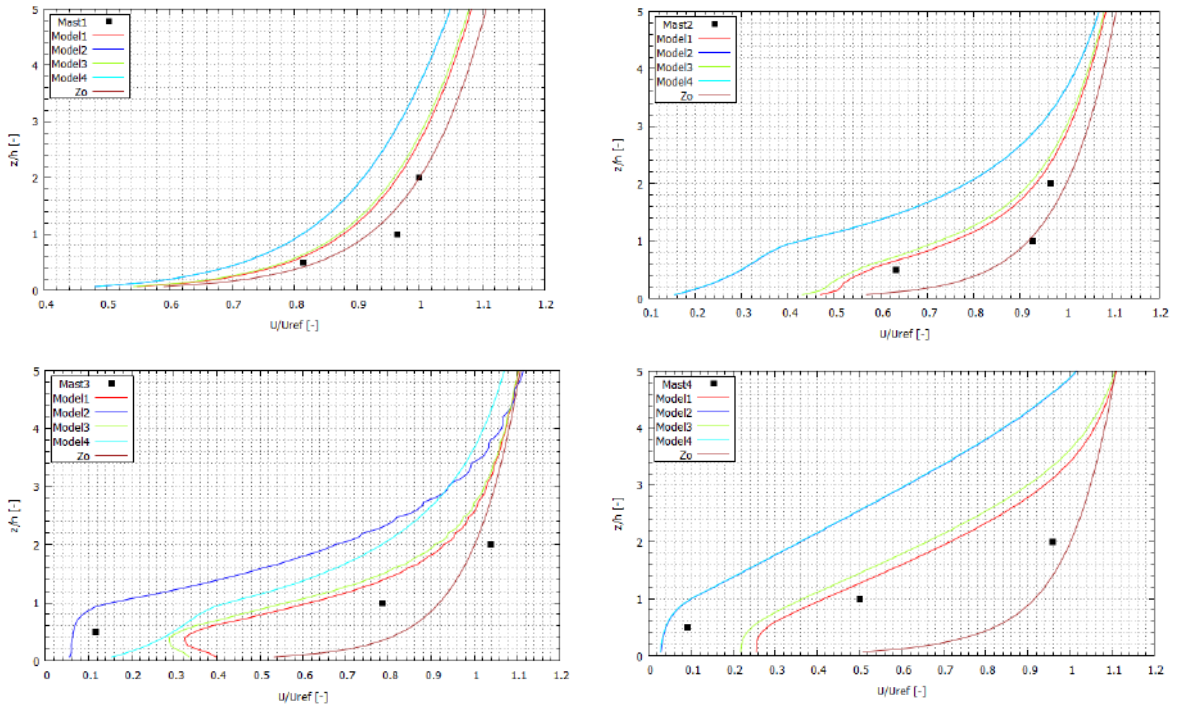


Figure 5.20: Canopy models normalized velocity profiles.

At a first sight, models 1 and 3 seem to better resemble the shape of the velocity profile at the masts. They show the wind deceleration before entering the canopy (mast 1 and 2) below  $h$ , and by mast 4 they have already adapted to the new mean and the new canopy surface. However, they underpredict wind speed above  $h$  and overpredict it below  $h$  inside the canopy (mast 3 and 4). No significant change derived from the use of the linear term of the Darcy-Forchheimer equation (Eq.4.12), except for model 4 at Mast 3, can be appreciated. This agrees with fact that inside forested areas Reynolds number can be assumed so high that inertia prevails over viscous effects, which can then be neglected, leading to one-term expressions for the momentum losses [126]. Finally, the simulation with an enhanced roughness length yielded good results outside and above the canopy, with a relative error, always under 20%, but failed to predict the wind deceleration inside the canopy.

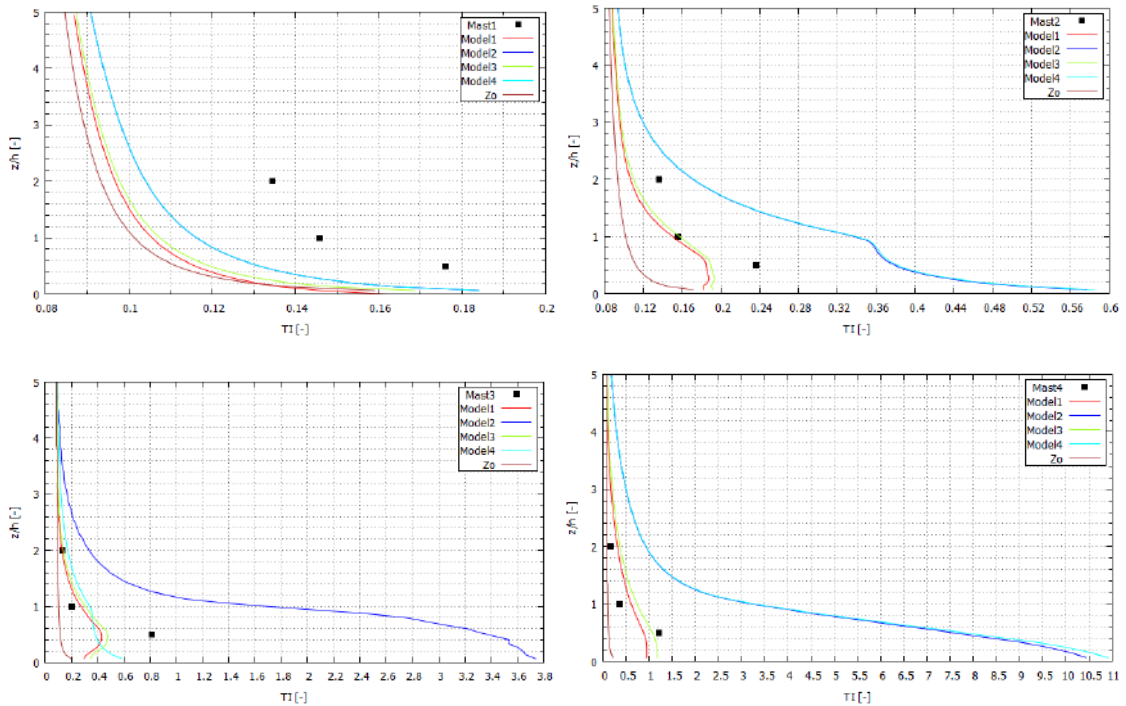


Figure 5.21: Canopy models' intensity profiles.

In Fig. 5.21 the results for the turbulent intensity at the mast positions are presented. Whereas models 4 and 3 failed to reflect the TI profiles, model 1 and 3 showed a better agreement with the experimental data, especially inside the canopy (masts 3 and 4) and at  $z/h > 1$ . Nevertheless, they underpredict the turbulent intensities at masts 1 and 2, and at  $z/h < 1$  at masts 3 and 4.

In general, wind speed was better predicted than turbulent intensity outside the canopy, while the opposite happened inside. Whereas wind speed predictions below  $h$  showed a higher accuracy than above  $h$  (up to two times the predicted values). Such a difference cannot be found for the intensity.

As it is said in [83], the assumed  $\sigma_v/u_*$  value may, partially explain, the disagreement between the models and the experimental data. To check this point, simulations were carried out using different  $\sigma_v/u_*$  values, resulting in a better agreement between the models and the experimental data for lower  $\sigma_v/u_*$  values (see Annex 2). Other possible reasons for these differences might be the election of  $\alpha$  and  $C_D$  values. For this reason, further tests were made using model 3 equations and different  $C_D$  values  $\alpha$  profiles, which adjust better to (Fig.5.17 a))

## 5.3 Conclusions

From the analysis of the results of the previous tests the following conclusions could be drawn:

- The results suggested that the height of the centre of the first cell above the ground must be larger than the roughness length, keeping it however as low as possible. A factor 5 between both seemed to minimize the streamwise gradient effect.

$$z_p > z_0 \text{ \& \; } \frac{z_p}{z_0} \approx 5 \quad (5.6)$$

- Increasing the number of cells in the vertical direction tends to amplify the relative errors in the first cells above the ground and may lead to the slowdown of the convergence process.
- Except for  $k$  and only for low values, the cell's aspect ratio doesn't have any significant effect on the flow below 560. Above this number, it provokes the simulation to stop abruptly.
- None of the tested canopy models provided satisfactory results. Nevertheless, those based on  $\alpha$  and  $C_D$  showed a better agreement with the experimental data outside and inside the canopy. A more accurate election of the  $\alpha$  and  $C_D$  values led to better results. (see Annex 2). The inclusion of the linear term of the Darcy-Forchheimer equation didn't have any significant effect on the result. Finally, the enhanced  $z_0$  model provided the most accurate results outside and above the canopy, although it failed to predict velocity and TI values inside it.

# Chapter 6

## Validation Process

In the present chapter, the settings presented in [Ch. 4](#) and the findings of [Ch. 5](#) were applied to simulate the wind flow over the Bolund Hill. Then the results were compared with experimental measurements following the *Bolund Blind Comparison Experiment* [\[127\]](#).

### 6.1 The Bolund Experiment

Many measurement campaigns to provide data for the validation of numerical flow models have been carried out in the last few decades. Perhaps, the most well-known of them is the *Askervein Hill Experiment* [\[128,129,130\]](#), which results have been extensively used for validating linear models. However, due to its gentle slopes, the Askervein Hill can hardly be considered complex terrain and thus, phenomena such as flow separation and recirculation zones isn't likely to occur. For this reason, it was decided to use the measurement campaign carried out by the Wind Energy Department of the Technical University of Denmark (DTU) at the Bolund Hill [\[127\]](#). Due to the more abrupt terrain features and abrupt roughness changes which may give rise to complex phenomena Bolund Hill offers an appropriate scenario to test turbulence models.



Figure 6.1: The Bolund Hill [\[46\]](#).

#### 6.1.2 The Bolund Hill

The Bolund hill is a 12m height, 130m long and 75m width hill located in Roskilde Fjord, Denmark (55.7035°N, 12.0982°E). It lays in the middle of a small bay and is completely surrounded by water except for a few hours a day, when a narrow land strips emerges linking it to the mainland. The hill is flat and covered with grass on the top, with high slopes, up to 40° to the north, east and south and a vertical escarpment to the west with a sharp crest. Coriolis effects can be neglected due to the hill height [\[127\]](#) and the long upstream fetch to the west and east makes it easier to achieve a HHTBL before the flow reaches the hill.

Between December 2007 and February 2008 wind speeds and direction and turbulence data were collected using 60 sensors, distributed on ten masts located along the predominant wind directions (see [Fig.6.2](#)). This was the first phase of the Bolund Experiment.

The cross of the two-line A and B marks the hill's centre point, CP where the origin of an  $x$ - $y$  coordinate was set, with  $x$  and  $y$ -axis corresponding to the west-east and north-south directions respectively. The vertical axis, corresponding to the positive  $z$ -direction.

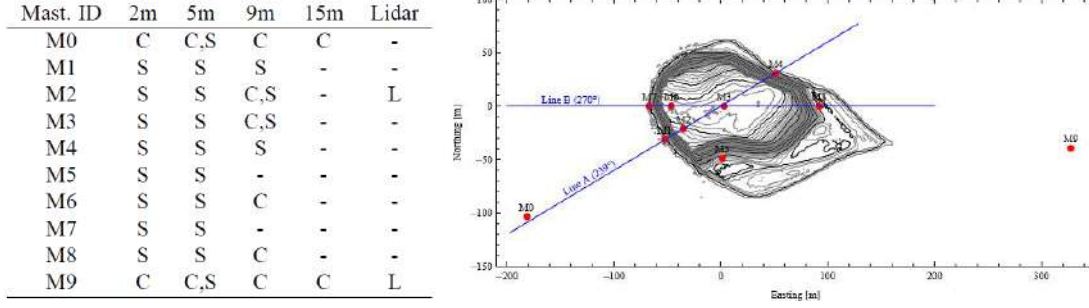


Figure 6.2: The Bolund Experiment a) instrumentation b) and mast position [46].

### 6.1.3 The Blind Comparison

In the Bolund Experiment four simulation cases, referred as the *Blind Comparison* [30], are defined in four different directions with specific boundary conditions for each one based on the experimental data, so that models can be tested under the same conditions and thus easily compared. Measurements at M0 and M9, considered undisturbed, were used to produce the boundary conditions (see Table 6.1) to be applied at the inlet of the simulation domains by fitting them to the inlet profiles described in Ch.4 (Eq.4.1,4.2,4.3,4.4).

Case	Wind Direction(°)	Reference Mast	$k(m^2/s^2)$	$U_f(m/s)$
1	270	M0	0.928	0.4
2	255	M0	0.928	0.4
3	239	M0	0.928	0.4
4	90	M9	1.45	0.5

Table 6.1: Blind Comparison cases boundary conditions.

The participants were asked to use the same air properties and coefficient values. While air density and viscosity values were provided in [46],  $C_\mu$  was calculated out of the TKE data according to Eq.2.22.

The comparison takes place between the normalised Speed Up ( $\Delta S$ ) and the TKE increasing ( $\Delta k$ ) of the measurements and the simulation results which are calculated as follows:

$$\Delta S_S(z_{ref}) = \frac{u(z_{ref}) - u_{ref.mast}(z_{ref})}{u_{ref.mast}(z_{ref})} \quad (6.1)$$

$$\Delta k_S(z_{ref}) = \frac{k(z_{ref}) - k_{ref.mast}(z_{ref})}{u_{ref.mast}^2(z_{ref})} \quad (6.2)$$

$$\Delta S_M = \frac{\langle \overline{u(z_{ref})} / u_{*0} \rangle - \langle \overline{u_{ref.mast}(z_{ref})} / u_{*0} \rangle}{\langle \overline{u_{ref.mast}(z_{ref})} / u_{*0} \rangle} \quad (6.3)$$



$$\Delta k_M = \frac{\langle \overline{k(z_{ref})} / u_{*0}^2 \rangle - \langle \overline{k_{ref.mast}(z_{sonic})} / u_{*0}^2 \rangle}{\langle \overline{u_{ref.mast}(z_{ref})} / u_{*0} \rangle^2} \quad (6.4)$$

The predicted wind characteristics can be obtained at any height, however this cannot be done with the experimental data. In that case the reference velocity at heights different from that of the sensors must be calculated by fitting the wind speed from the sonic anemometer at the reference mast according to the following expression:

$$\langle \overline{u_{ref.mast}(z)} / u_{*0} \rangle = \langle \overline{u_{ref.mast}(z_{sonic})} / u_{*0} \rangle + \frac{1}{\kappa} \ln \left[ \frac{z}{z_{ref}} \right] \quad (6.5)$$

For the quantification of the wind speed and the TKE error the following expressions are use, where  $I = \frac{\sqrt{k}}{s_0}$  is the turbulent intensity, the sub index 0 meaning the reference mast:

$$R_S = 100(\Delta S_S - \Delta S_m) \quad (6.6)$$

$$R_{TKE} = 100 \left( \frac{(I_S - I_{S.inlet}) - (I_m - I_{m.inlet})}{I_{m.inlet}} \right) \quad (6.7)$$

## 6.2 The Simulation

### 6.2.1 The Domain

The Bolund Experiment team provides the topography data for the simulations in several files in GRD-ASCII and Map-formats. After subtracting the CP coordinates, the result is a 290x250m map with a 25cm horizontal resolution with SW and NE corners at (-98,-132) and (192,118) respectively (see Fig.6.3) . The sea level, 0.75m, was also subtracted from all the heights.

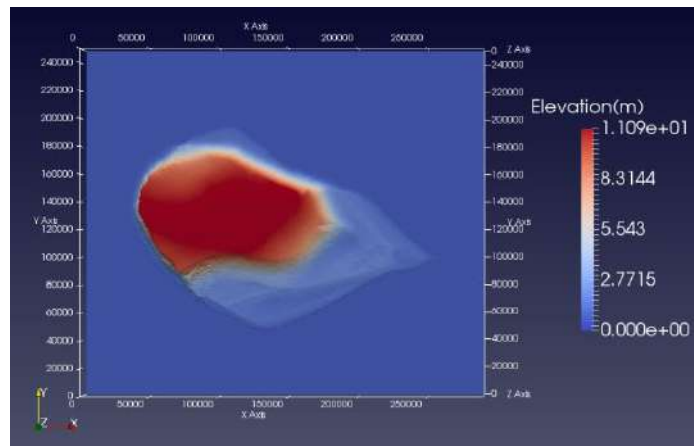


Figure 6.3: Bolund Hill orography file. Here coordinate's origin is set at (-98,-132). Paraview has no concept of units, and use the units in which the data is given by default. In this cases mm.

A domain 1000m long in the wind direction (the domain is rotated for each case), 500m in the direction perpendicular to the wind and 500m height, centred in CP was created with *cfMesh* (see Annex 3 for the *meshDict*) (see Fig.6.4).

This ensures upstream and downstream fetches of around 4 to 5 times the hill's length. The domain was divided in several areas with different resolutions. The finest resolution was applied in the first 30m above the bottom patch over the whole domain. Cells, first twice and then four times as long as the first ones were used between 30m and 80m, and then, cells 8 times the size of the first ones were used up to the top patch. In the first 20m above the bottom patch cell's thickness is 3 times their length, this ratio changes with the height, becoming 2 at the top patch.

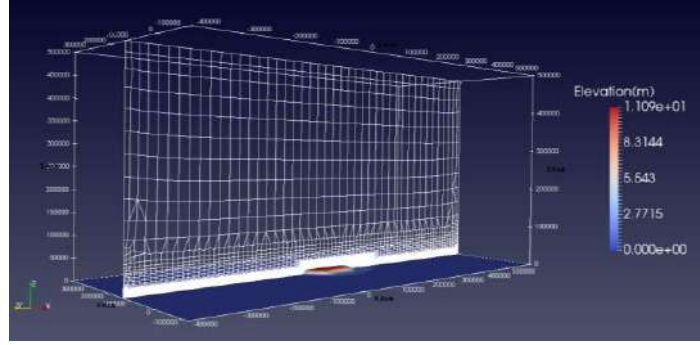


Figure 6.4: Blind Comparison simulations domain.

As for the first cell thickness, it was set to  $z_0=0.015\text{m}$  for the hill and  $z_0=0.0003$  for the surrounding water [46]. For these  $z_0$ , according to the findings presented in Ch.5, a  $z_p=0.075\text{m}$  and a  $z_p=0.0015\text{m}$  must be imposed above the hill and the water surface respectively. However, the maximum aspect ratio must remain below 560 (see Ch.5). This means that the first cells above the water should be less than 2m long. This might lead to an unnecessarily high number of cells and consequently to an increasing computing time. In the end, it was decided to add 11 boundary layers above the water, with  $z_p=0.027\text{m}$ , 18 times the ideal thickness, and 5 boundary layers above the hill with  $z_p=0.075\text{m}$ . In both cases the expansion ratio was set to 1.2.

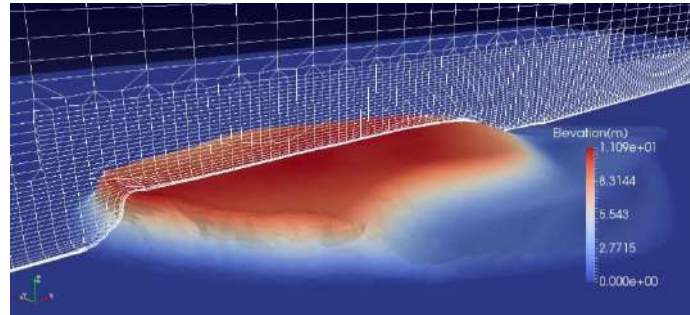


Figure 6.5: Refinement above the Bolund Hill.

## 6.2.2 Boundary Conditions

The boundary conditions described in Ch.4 were used in each simulation, applying the general information provided in [30] and the specific conditions of each case (Table 6.1) (see Annex 3 for the files).

The second order upwind scheme was used for the convective derivatives terms in every simulation. However, as some bounding and continuity errors occurred in the first iterations, it was decided to use first order Upwind schemes in the first 200 iterations and higher relaxation factors in the first 300, before applying the settings described in Ch.4.



### 6.2.3 Grid Sensitivity Analysis

Before simulating a case, a grid sensitivity analysis was carried out for Case 1. The wind flow was simulated using 3 different grids with an increasing resolution value in the first 30 m above the bottom patch (see Table 6.2.) and their results compared with the experimental data.

Resolution(m)	Time(s)	N°Cells
4	12114	1066011
3	13954	1828279
2	57281	4131970

Table 6.2: Grid Independency tests settings.

The mesh didn't manage to resemble the hill's shape for resolutions lower than 4m, for this reason that was the lowest resolution considered in this analysis. The minimum resolution was set at 2m, for no solution was reached for 1m after ca.80 hours of simulation.

The simulations results (see Annex 3) are not conclusive, for not trend can be appreciated. Therefore, it is not possible to say whether grid independency has been achieved or not. However, the 3m-resolution mesh seemed to produce the best results and for this reason it was decided to use it in the validation case.

## 6.3 Validation Cases

Of all four cases, it was decided to simulate only case 1, for it is the west side of the hill which presents the vertical escarpment that may produce the complex turbulence phenomena which may pose a challenge to the  $k-\varepsilon$  model.

### 6.3.1 Reference Wind

Before comparing the results, the predicted velocity and intensity profiles at the reference mast, M0 (see Fig.6.6) must be analyzed. Here and from now on, measurements from sonic and cup anemometers are represented by squares and triangles respectively.

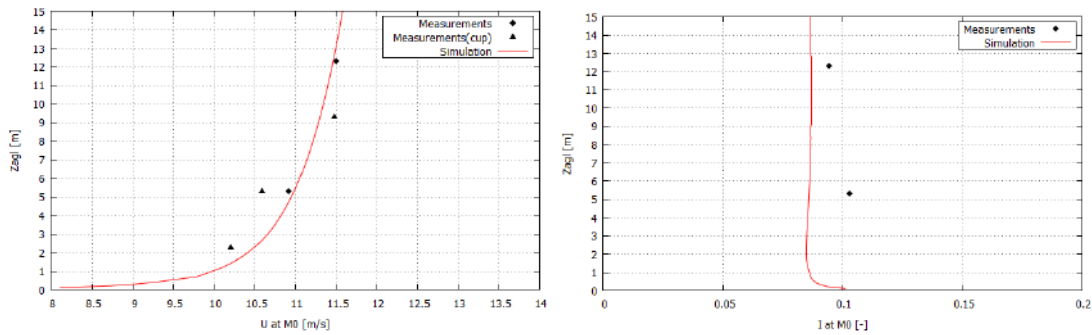


Figure 6.6: Reference wind a) velocity b) and intensity profiles.

According to [30] the profiles can be considered acceptable if they don't differ more than a  $\pm 15\%$  in the case of the velocity and  $\pm 30\%$  for the turbulence intensity  $I$ .

In the present work, errors (see [Annex 3](#)) were on average around 24% for the velocity and 12.1% for the intensity. However, the mean velocity average error drops to 1.3% if only the sonic anemometers are considered.

### 6.3.2 Speed-Up Wind

In [Fig.6.8,6.9](#) and [6.10](#), the speed-ups derived from the experimental measurements and those from the simulation's predictions at 2, 5 and 9m above ground level (a.g.l) along line B (270°, Case 1) are presented. The corresponding errors can be found in [Annex 3](#).

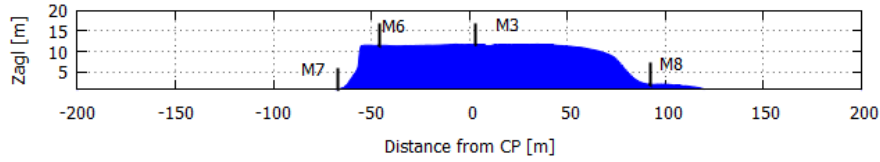


Figure 6.7: Bolund Hill elevation profile along Line B (see [Fig.6.2](#)).

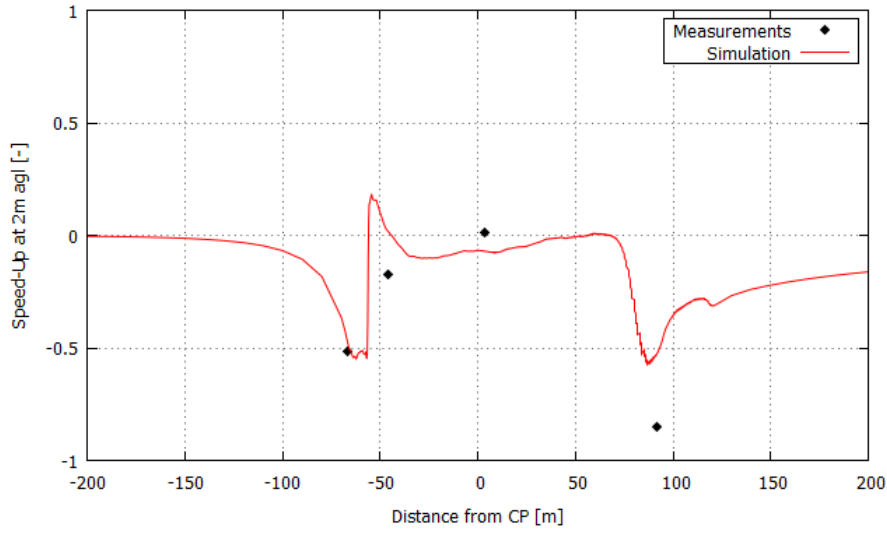


Figure 6.8: Speed-up over the Bolund Hill along Line B at 2m agl.

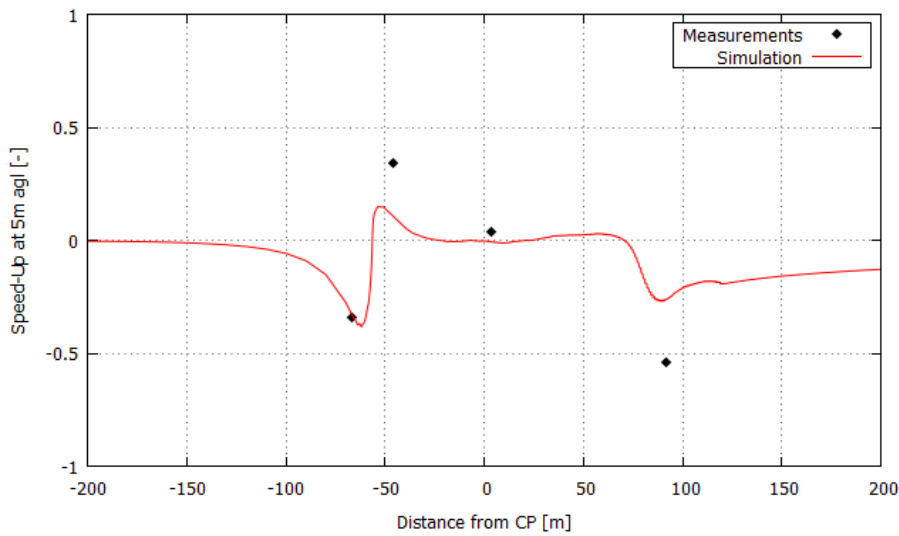


Figure 6.9: Speed-up over the Bolund Hill along Line B at 5m agl.

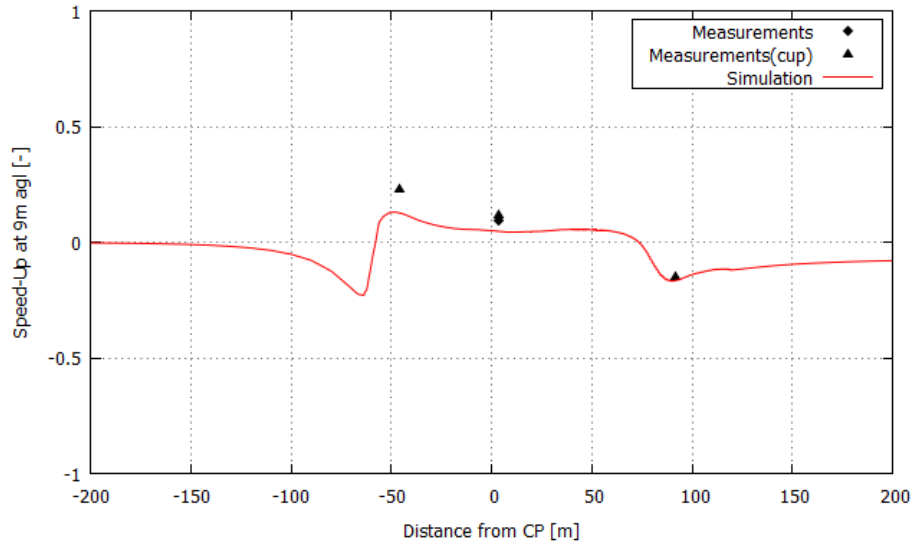


Figure 6.10: Speed-up over the Bolund Hill along Line B at 9m agl.

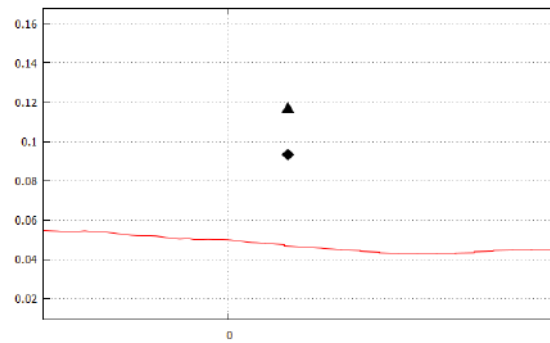


Figure 6.11: Speed-up along Line B at M3 at 9m agl.

The model could predict the general features of the speed-up of the wind flow over the Bolund Hill, although the accuracy of the prediction changes greatly with position. At the base of the escarpment (M7) the model predicts the speed-down with an absolute error of 4.6% at 2m and 1% at 5m a.g.l., while, at the edge of the escarpment (M6), although the model predicts the speed-up it doesn't match the experimental data at 2m a.g.l and at 5 and 9m a.g.l it greatly overestimated the speed-up with absolute errors around 18% and 23% respectively. At M3, close to the CP, the simulation produced very accurate results, with absolute errors between 4 and 8%. Finally, on the lee side of the hill (M8), the flow speed-down is predicted at each height, although it is greatly underestimated at 2 and 5m a.g.l. At 9m a.g.l the absolute error is ca.2.1%.

As expected the better predictions were obtained at M7 and M3, where the wind is less disturbed, with absolute mean errors around 3% and 6% respectively. The accuracy of the predictions increases with the height, for the sensors at 9m a.g.l are less affected by the turbulence phenomena than those at 2m a.g.l (see Fig.6.12). Thus, the mean absolute error are around 15% at 2m a.g.l, approximately 14% at 5m a.g.l and 6% at 9m a.g.l.

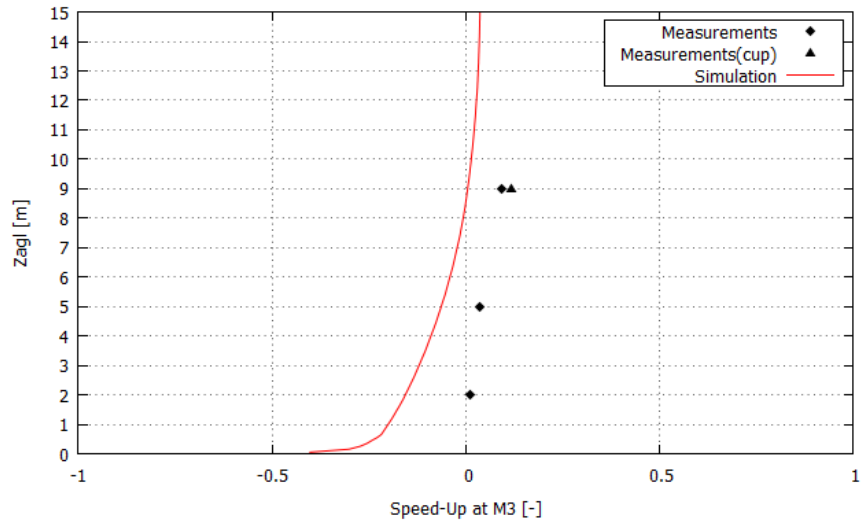


Figure 6.12: Speed-up along Line B at M3.

### 6.3.3 Turbulent Kinetic Energy

In Fig.6.13,6.14 and 6.15 the TKE variations derived from the experimental data and those predicted for Case 1 at 2,5 and 9m a.g.l are displayed. The corresponding errors can be found in Annex 3.

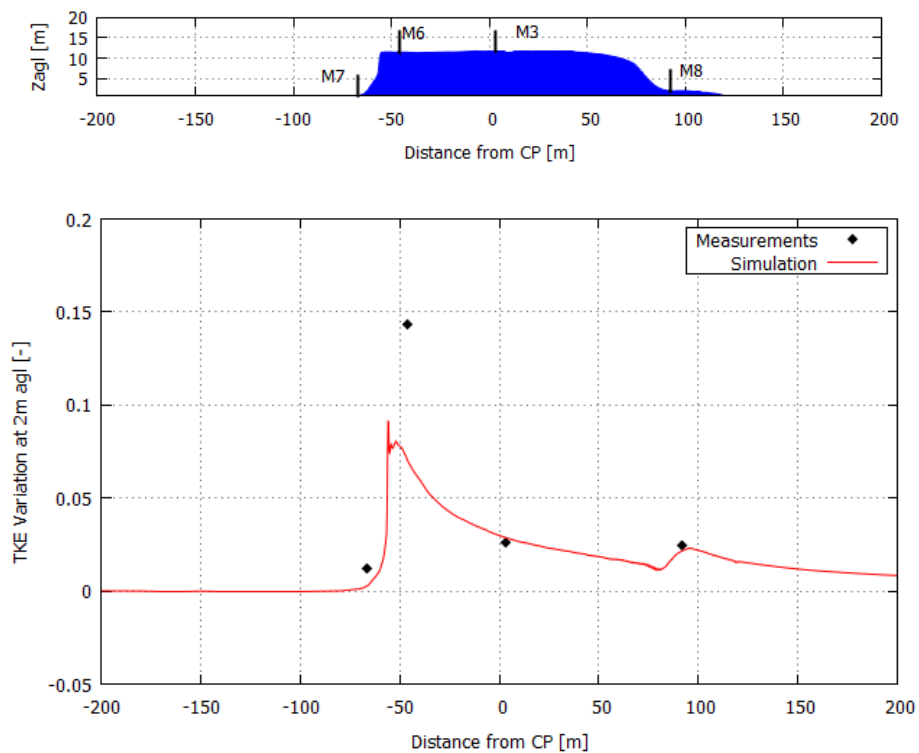


Figure 6.13: TKE Variation over the Bolund Hill along Line B at 2m agl.

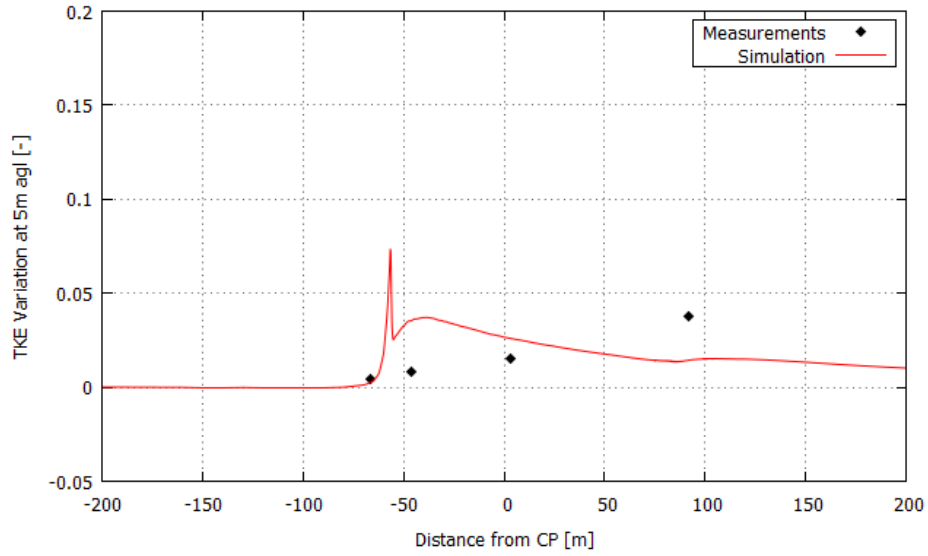


Figure 6.14: TKE Variation over the Bolund Hill along Line B at 5m agl.

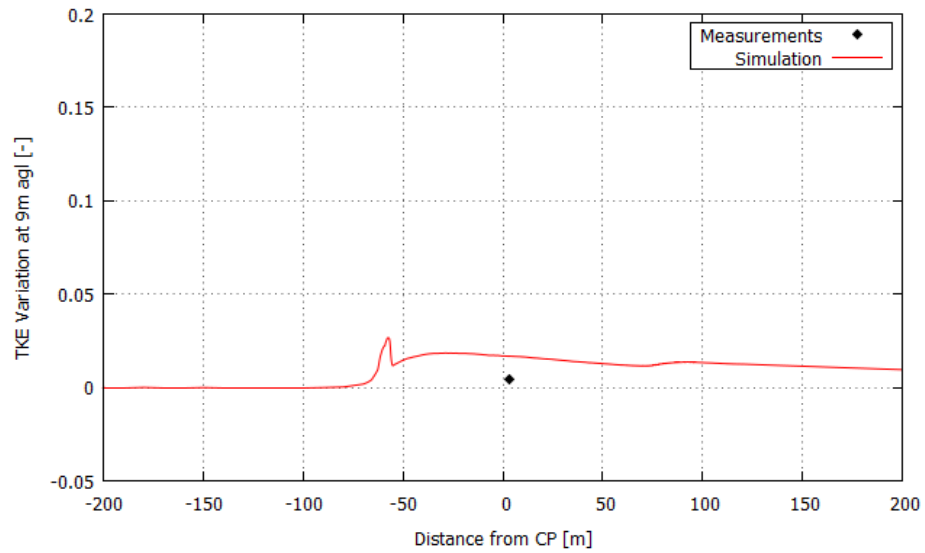


Figure 6.15: TKE Variation over the Bolund Hill along Line B at 9m agl.

The model successfully captured the general features of the TKE variation of the wind flow over the hill. It predicted the TKE peaks at the top of the escarpment (M6) and on the lee side of the hill (M8), as well as the progressive reduction of TKE variations with height. However, it underestimates and overestimates the TKE variations at different positions, with errors larger than those of the speed-up. And, unlike the speed-up errors, in this case, TKE variation errors increase with height (see [Fig.6.16](#)).

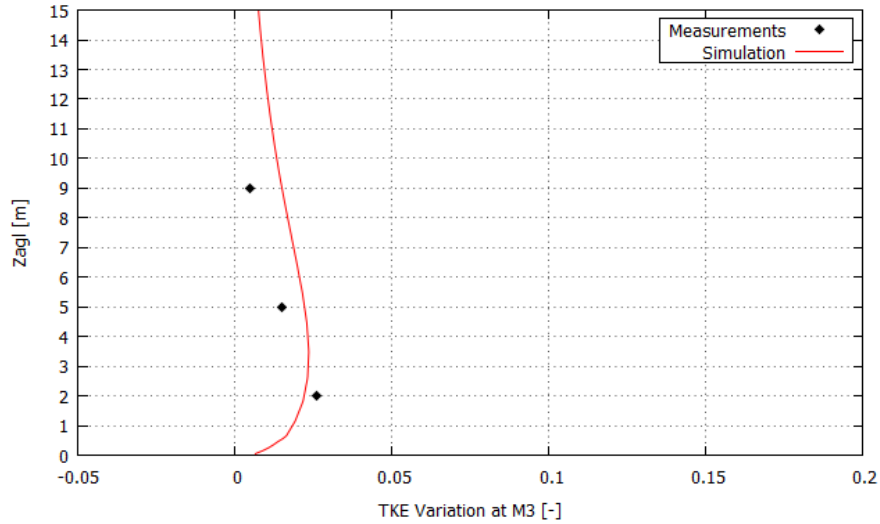


Figure 6.16: TKE Variation along Line B at M3.

## 6.4 Conclusions

The mean absolute speed-up error<sup>11</sup> for Case 1 is 12.3%, smaller than the mean absolute error of the models submitted to the organizers of the *Blind Experiment* [30] which is 15.8% for all cases and 17.3% for Cases 1 and 3 alone.

It is also smaller than the mean absolute error of all the two-equations RANS models, 13.6% for all cases and 15.1% for Cases 1 and 3, which, in general, were the ones to yield the best results. However, it is higher than that of the best performing model submitted, which is 10.2% for all cases and 11.4% for Cases 1 and 3.

As for the TKE variations, the mean absolute error, 67.53% is larger than any of the errors yielded by the models used by the participants, except for the linearized models. If only the 2m a.g.l values are considered, the mean absolute error, now 50%, gets closer to those of the two-equation RANS models, around 40%. Nevertheless, it still lies far from the best results, around 20% for all cases and 30% for Cases 1 and 3, yielded, by the best two-equation RANS model.

Such errors would make the predicted values unacceptable as a base for the installation of a wind-farm. In addition to the errors that may have been committed in the construction of the simulation, like a too coarse grid refinement or an inappropriate vertical resolution, there are other factors which might explain, to some extent, these results, like for instance a suggested roughness length higher than the real one or the hill's complex topography. Indeed, if only the values above the turbulent stresses-dominated layer are considered (9m a.g.l), the speed-up mean absolute error drops to 3.4%. It also should be remembered that, although participants were asked to consider the sensors to be at 2,5 and 9m a.g.l these are not their real heights. Sensors at mast M8 are located at 1.8 and 4.7m a.g.l. for instance. Let alone the inaccuracy of the mast location or the position of the sensors. All these factors can lead to bigger, but also smaller, differences between measured and simulated values.

12. For the mean error calculations, only the sonic anemometer positions and measurements were considered [30].

# Chapter 7

## Study Case: Sierra Morena

In this last section, the k- $\epsilon$  model, with the settings explained in [Ch.4](#) and the findings made in [Ch.5](#), was used to simulate the wind flow over the area surrounding a rural community on the coast of Chiapas. Then the results were used to create a wind resource map of the region, based on which the feasibility of the installation of small wind turbines in the region was assessed.

### 7.1 Field Work

Between June and July 2015, 20 rural communities located in the valleys of the *Sierra Madre de Chiapas* and in the plains between the *Sierra Madre* and the Pacific Ocean were visited to collect first-hand information about the electrification rate and electricity consumption and needs of the inhabitants. Data on wind speed, temperature, pressure, solar radiation etc were also collected from 13 meteorological stations in the region, operated by different institutions and organizations like the *Comisión Nacional del Agua*, CONAGUA [131] and the *Instituto Nacional de Investigaciones Forestales Agrícolas y Pecuarias*, INIFAP [132].

However, these stations were equipped for weather monitoring and lack the appropriate equipment to sample data for wind assessment studies. They are usually equipped with just one cup anemometer and one vane, and therefore cannot, for instance, simultaneously provide information about the velocity changes in each direction, for, this requires a sonic anemometer. Moreover, with a few exceptions, only hourly mean values are recorded instead of 10-min or 15-min mean values.

### 7.2 Village Choice

The community of *Sierra Morena*, in the Villa Corzo municipality was chosen for the study case. These are the factors which led to this decision:

- Its relative isolated location within the *La Sepultura* Nature Reserve, at 1154m height up in one of the Sierra de Chiapas valleys
- According to the inhabitants, the electricity supply is not reliable, and, despite of being connected to the national grid, they suffer frequent cuts and blackouts.
- The complex topography of the surrounding area
- The proximity of a meteorological station.
- Relatively frequent wind speeds above 5m/s

Sierra Morena (16° 9'9.98"N, 93°35'27.87"W) is located at 1100 m height above sea level, on the bottom of a valley which begins at the city of Villaflores and extends to the west into the *Sierra Madre de Chiapas*. The village comprises around 60 buildings and its population is ca.160. Agriculture, mostly corn and beans but also coffee and palm, is the principal economic activity. The village is surrounded by low mountains covered with mixed forest.



Figure 7.1: Sierra Morena a) met.station location b) view from the met.station.

### 7.3 Wind Data

The closest meteorological station is located on a low hill out of the community (see Fig.7.1) ( $16^{\circ} 9'7.92''N$ ,  $93^{\circ}35'27.96''W$ ) and is equipped with a cup anemometer and a vane at 10m on the same mast height (see Fig.7.2). The terrain on which the mast stands is covered with low bushes and, except for a few small trees, there is no obstacle which may affect the measurements. However, the hill slope may speed up the flow resulting in artificially high velocity values.

The availability of only one wind data source prevents the validation of the results of the simulations and the simulation of wind flow over the same area in opposite directions.

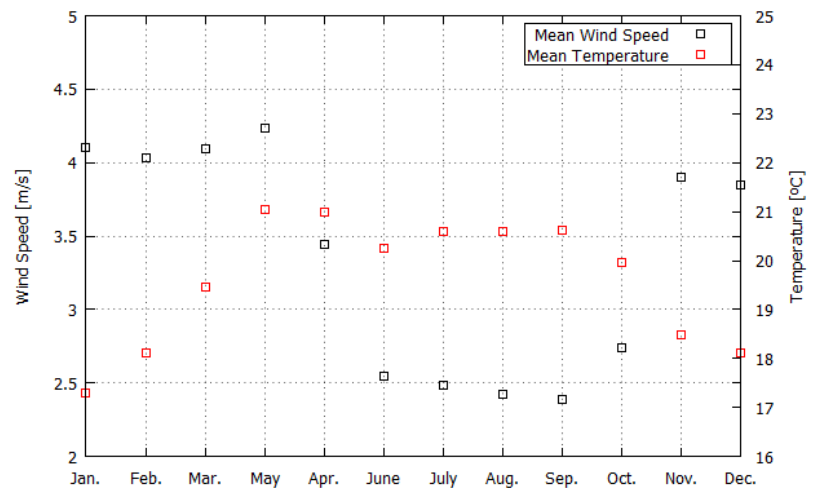


Figure 7.2: Sierra Morena a) met.station b) monthly averaged temperature and wind speed values.

Hourly mean wind speed and direction data was obtained from the period between 01/01/2012 to 09/08/2016. The WAsP OMWC (*Observed Mean Wind Climate*) tool for Wind Climate Analysis was used to analyse the wind data (see Annex 5).



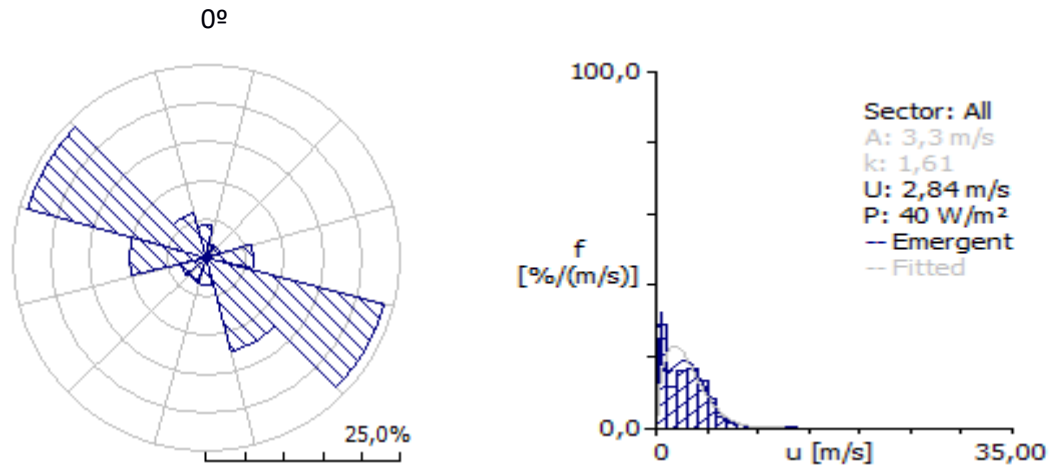


Figure 7.3: Sierra Morena met. station a) wind rose and b) Weibull distribution.

The wind rose in Fig. 7.3 a) shows that the main wind directions are given by sectors 120° and 300°, which coincide with the valley axis. This rose resembles a typical mountain-valley wind pattern ([44] pag. 117). The highest sectorial mean wind speed, around 3,5 m/s, was also registered in sectors 120° and 300°. Mean wind speed changes over the year, from around 2.5 m/s in summer and autumn, to more than 4 m/s in average between January and May (see Fig. 7.2 b)). In these months, if only the principal directions are considered it rises beyond 5m/s.

## 7.4 Atmospheric Conditions and Model Limitations

As it was said in Ch. 2, the use of the k-ε model is subject to a set of limitations and assumptions.

### 7.4.1 Turbulence

Reynolds number provides a ratio between the eddy and molecular viscosity. To apply a turbulence model, the first must be much larger than the former. At high Reynolds numbers the molecular viscosity effects can be neglected, the flow in the few hundred meters above the terrain can be considered inertial dominated and therefore Reynolds independent.

### 7.4.2 Atmospheric Stability

This model is only valid provided the atmosphere is neutrally stratified. This is guaranteed at high wind speeds,  $U > 5\text{m/s}$  and a *Monin-Obukhov Stability Length* (Eq. 7.1) larger than 250m [30].

$$L_{MO} = \frac{u_*^3}{\kappa \left(\frac{g}{\theta}\right) w' \theta'} \quad (7.1)$$

Due to the met. station's equipment, no information about wind speed changes in each direction is available and therefore  $L_{MO}$  couldn't be calculated. However, if a valley-mountain wind pattern is assumed, then the atmospheric stability conditions are different at the bottom of the valley and in the surrounding mountains, which makes the neutral stratified atmosphere less likely [133]. Only 16% of all the hourly averaged wind speed values were above 5 m/s. This percentage rises to around 35% (see Annex 5) if only 120° and 300° sectors are considered. Therefore, in order to guarantee a neutrally stratified atmosphere, these were the only values considered for the simulations

### 7.4.3 Coriolis Force

Although the effects of the Coriolis forces for the vertical direction may be neglected straight away [46], for the horizontal direction, the extent of its influence is usually assessed using the Rossby number, given by:

$$R_o = \frac{U}{fL} \quad (7.2)$$

where  $f$  is the Coriolis parameter, which value is estimated to be around  $10^{-4} \text{ s}^{-1}$  for mid-latitudes and  $L$  the terrain characteristic length. If Rossby number is much larger than one, Coriolis effects can be neglected. The same applies for turbulence length scales smaller than around 10km, as it happens in the present case, where, therefore, Coriolis forces can be neglected [22].

## 7.5 Terrain Input Data.

INEGI offers digital terrain data of elevation and land cover for the entire country. This data is available in different formats (.shp, .tif etc) and resolutions, from 5m up to 50m depending on the method used for the measurements.

### 7.5.1 Orography

As it was said, Sierra Morena lies on the bottom of a 2km wide valley of the Sierra Madre de Chiapas, which runs east to west. The lowest and highest points in the surrounding area, are at 950m and 1947m height respectively as it can be seen in Fig.7.4.

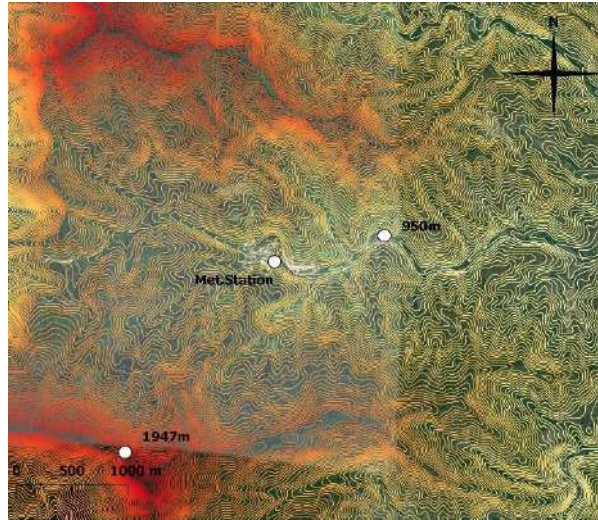


Figure 7.4: Sierra Morena area contour curves map.

### 7.5.2 Terrain Ruggedness

As it was explained in [Ch.1](#), ruggedness or steepness is one of the key features of a complex terrain and a performance indicator for linearized models [\[134,135\]](#). Two methods were used to analyse this property on the area surrounding Sierra Morena. First, the RIX index using the Map Editor WAsP tool, which calculates the percentage of terrain steeper than a critical slope along several lines or radius parting from a given point.

As in [\[134,135\]](#), 0.3 was taken as the critical slope, and the RIX was calculated along 24 3.5km long radius, parting from Sierra Morena.

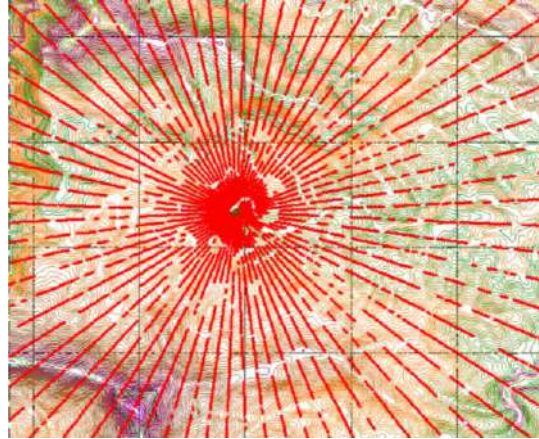


Figure 7.5: Sierra Morena area RIX map.

In [Fig. 7.5](#) the percentage of terrain steeper than 0.3, represented by the gaps along the lines, doesn't reach the 1% of the length of any line. However, this calculation depends on the length of the radius and the resolution of the terrain input data and therefore doesn't necessarily mean that the terrain is completely smooth, but only that most of it is.

The second ruggedness index used (see [Fig.7.6 a](#)) was the *Terrain Ruggedness Index* (TRI) proposed in [\[136\]](#). This index provides a measure of the terrain heterogeneity by calculating the elevation changes between a grid cell and its neighbours.

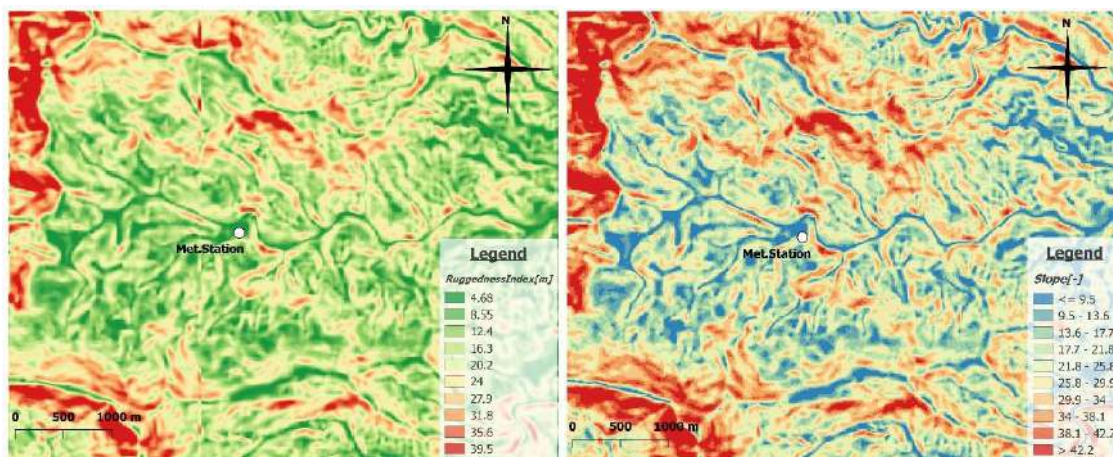


Figure 7.6: Sierra Morena area a) TRI and b) slopes maps.

According to the Riley classification [\[136\]](#), the terrain surrounding Sierra Morena falls within the lowest ruggedness level. However, as it happens with RIX, this result is also resolution dependant.

For this reason, it was decided to elaborate another map showing the terrain slope (see Fig.7.6 b)). There it can be observed that, in many locations, the slope is higher than 0.3, which is, according to [22,137] the threshold slope value for flow separation.

### 7.5.3 Land Cover

As no aerial photographs were found for the area and the topographical maps provided by the INEGI are incomplete and have a 50m resolution, far too low for micro-scale wind assessment purposes, it was decided to elaborate land cover maps with QGIS using satellite images (see Annex 6).




Land Cover	Colour	$Z_0(m)$
urban		0.5
grass/crops		0.05
forest		0.225
open field		0.15

Table 7.1: Land cover colour code and roughness length.

Despite the diversity of plant and trees and the different kinds of terrains, for the sake of simplicity, only four surface cover types were considered: urbanized, grass/crops, forest and open field, that is grass intermingled with bushes and scattered trees. Each of them was assigned a  $z_0$  value according to [28] (see Table 7.1). To model the forested areas, as no data is available about the leaf area density profile ( $\alpha$ ) or other characteristics of the existing trees species and only its averaged height, around 3m, is known, it was decided to use an enhanced  $z_0$ , which had yielded relative good results above the forested canopy (see 5.2.1), instead of a porous canopy.

## 7.6 Wind Input Data.

In the present case, wind in sectors 120° and 300° alone account for about the 50% of the all the data and including the highest wind speeds. For this reason, it was decided to carry out simulations in the mentioned directions exclusively. The wind data of both sector, with mean wind speed values above 5m/s, was fitted to the inlet profiles described in Ch.4, resulting in the following inlet values

Sector	$U_{ref}(m/s)$	$H_{ref}(m)$	$U_f(m/s)$	$K(m^2/s^2)$
120°	6.43	10	0.626	2.261
300°	5.66	10	0.551	1.750

Table 7.2: Sierra Morena simulations inlet profiles settings.



## 7.7 Domain

In [98] an interval between contour lines lower than 2m is recommended for the orography input data. Unfortunately, for the zone of interest only 15m-resolution terrain maps are available. A *.stl* file was created out of this maps using QGIS (see [Annex 6](#)).

The valley is approximately 2km width, which was taken as the domain width, and ends at a slope division 2 km up from the community. Down the valley there is no particular limit, however, in such a complex terrain, wind measurements become rapidly invalid with the distance from the measurement point [98,133]. Therefore, the domain length was set at 4km. As for its height, a 500m distance between the highest terrain point within the domain and the top patch was applied (see [Ch.4](#)). In this case the highest terrain point is at around 2000m, whereas the lowest point is at around 1000m, therefore the height of the domain will be around 1500m from the lowest point. Thus, a 4x2x1.5km domain using *blockMesh* (see [Annex 3](#)), centred on the met. station was created and then divided into two 2x2x1.5km domains, one for each wind direction. Their inlet patches were both located at the position of the met. mast.

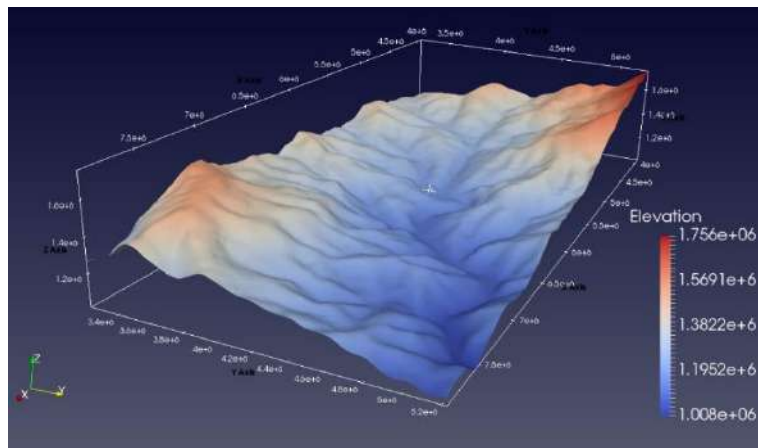


Figure 7.7: Sierra Morena domain.

Then land cover maps ([Fig.7.8](#)) created with QGIS (see [Annex 6](#)) were used to create sets of faces (*faceSets*) corresponding to the different land cover zones. These *faceSets* were used to define patches which will be assigned the corresponding  $z_0$  in the variables dictionaries.



Figure 7.8: Sierra Morena domain land cover.

A 3m horizontal resolution, like in the validation case, was applied over the whole domain. Given the small size of the areas with different land cover and the difficulties to apply different cell thicknesses in nearby regions, it was decided to use a uniform 1m thick first cell over the whole domain with a 1.2 expansion ratio, instead of different thickness for each region following the findings presented in Ch.5. For this reason, *snappyHexMesh* was used for the meshing instead of *cfMesh* (see Annex 4).

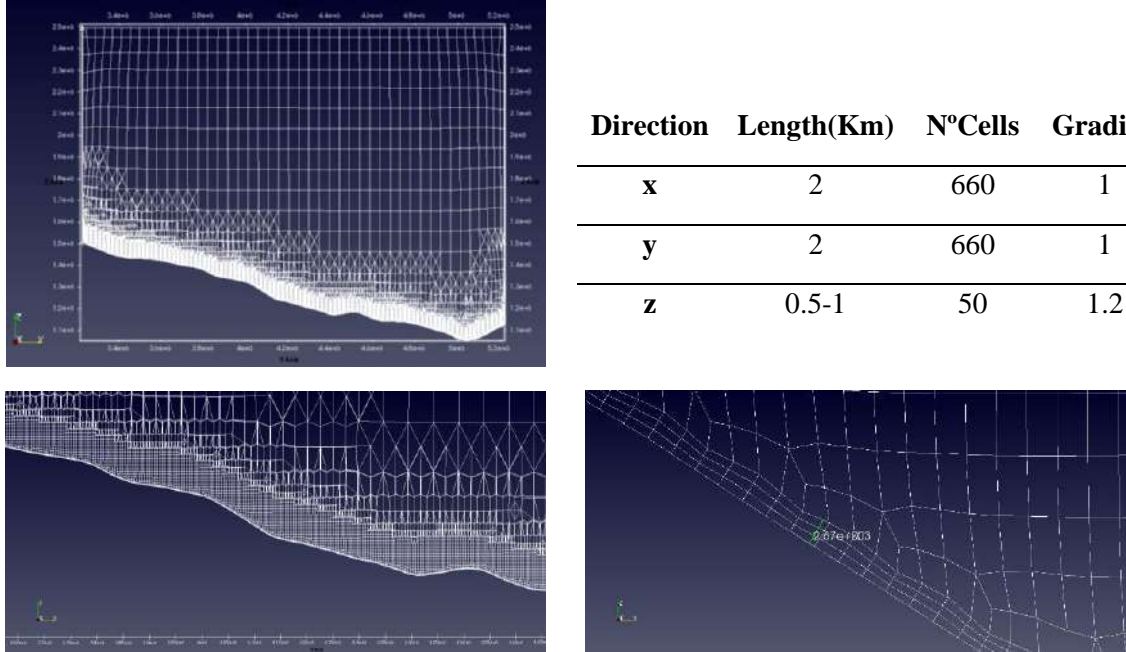


Figure 7.9: Sierra Morena grid a) general view in y direction, b) numeric characteristics c) near ground meshing and d) near-ground added layers.

## 7.8 Simulations

The BC described in Ch.4 were applied in the simulations (see Annex 4 for the files). However, the inlet OF *entries* for  $U$  and  $\varepsilon$ , *AtmBoundaryLayerInletvelocity* and *atmBoundaryLayerInletEpsilon* respectively, consider an uniform ground level at the inlet, *zGround*, which is not the case in most of the simulations on real terrain.

When used on irregular terrain these functions take the lowest point of the ground at the inlet as their *zGround*, and calculate  $U$  and  $\varepsilon$  at each cell of the inlet patch taking this value as reference. This leads to inlet profiles (see Fig.7.10 a)) where the velocity is the same at a certain height respect to *zGround*, no matter if the point is at 2 or at 3m above the terrain.

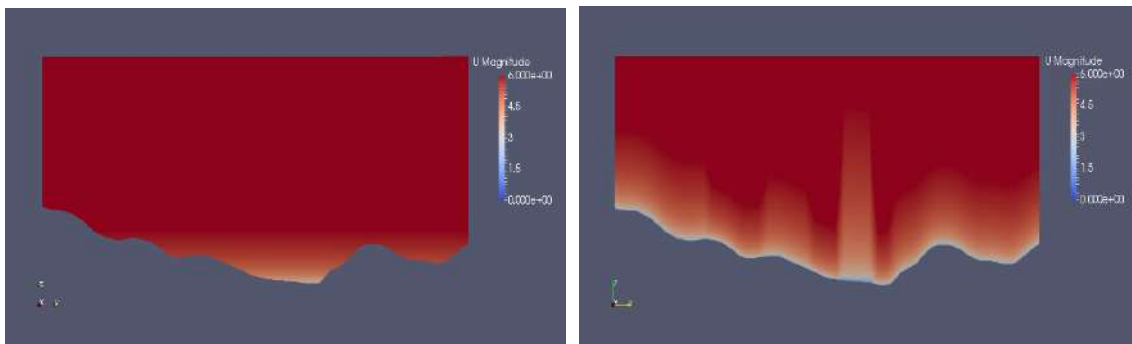


Figure 7.10: Wind speed at inlet patch using a) the OF entries and b) the *groovyBC* tool.

For this reason, the *groovyBC* utility presented in Ch. 4 was used to introduce a variable *zGround* value. A file containing the ground profile at the inlet was created (*InletLine.dat*) and then the *U* and  $\varepsilon$  profiles were written referred to this file, so that at each point *U* and  $\varepsilon$  are calculated using the height of that point over the closest terrain point in vertical direction (see Fig.7.11). This change yields non-uniform height *U* and  $\varepsilon$  distributions at the inlet as it can be seen in Fig.7.10 b).

```
inlet
{
  type groovyBC;
  value $internalField;
  variables "u_f=0.331;kf=0.41;yposition=pos().y;
  z0=(pos().y<3654600? 225:(pos().y>3918600? (pos().y<4141900? 225:(pos().y>4284200? (pos().y<4445300? 500:(pos().y>4515000? 225:150)):150)):150));";
  valueExpression "(u_f/kf)*(log((pos().z-profile(yposition)+z0)/z0))*vector(-1,0,0)";

  fields
  (
    yposition zGround
  );
  lookuptables (
  {
    name profile;
    outOfBounds clamp;
    fileName "$FOAM_CASE/constant/triSurface/InletLine.dat";
  }
  );
}
```

Figure 7.11: Wind speed inlet profile using the *groovyBC* tool.

Finally, a first order upwind interpolation scheme was applied, and the relaxation factors were kept at 3 during the first 100 iterations, to guarantee the stability of the process.

The simulations converged after around 400 iterations and about 2000s. The two domains were put together and wind speeds and wind power densities were calculated for three different heights a.g.l : 10, 30 and 50m. These were considered the most significant heights, for the small wind turbines used for rural electrification rarely reach the 30m height [138].

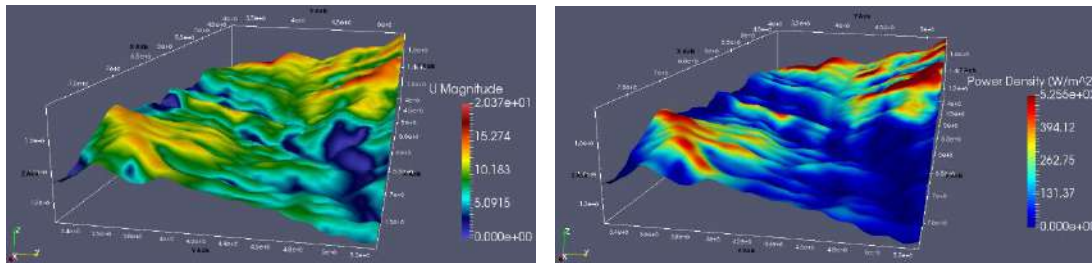


Figure 7.12: Sierra Morena a) wind speed and b) wind power density maps.

The wind power density was calculated according to the following formula:

$$W = \frac{1}{2} \rho C_p U^3 \quad (7.3)$$

Where  $C_p$ , the power coefficient, was given a value of 0.4 ([44] page-220). As for the density  $\rho$ , given the altitude of the valley where the community is located, it was decided to use a value of  $1\text{kg/m}^3$ , a bit lower than the standard density value at sea level.

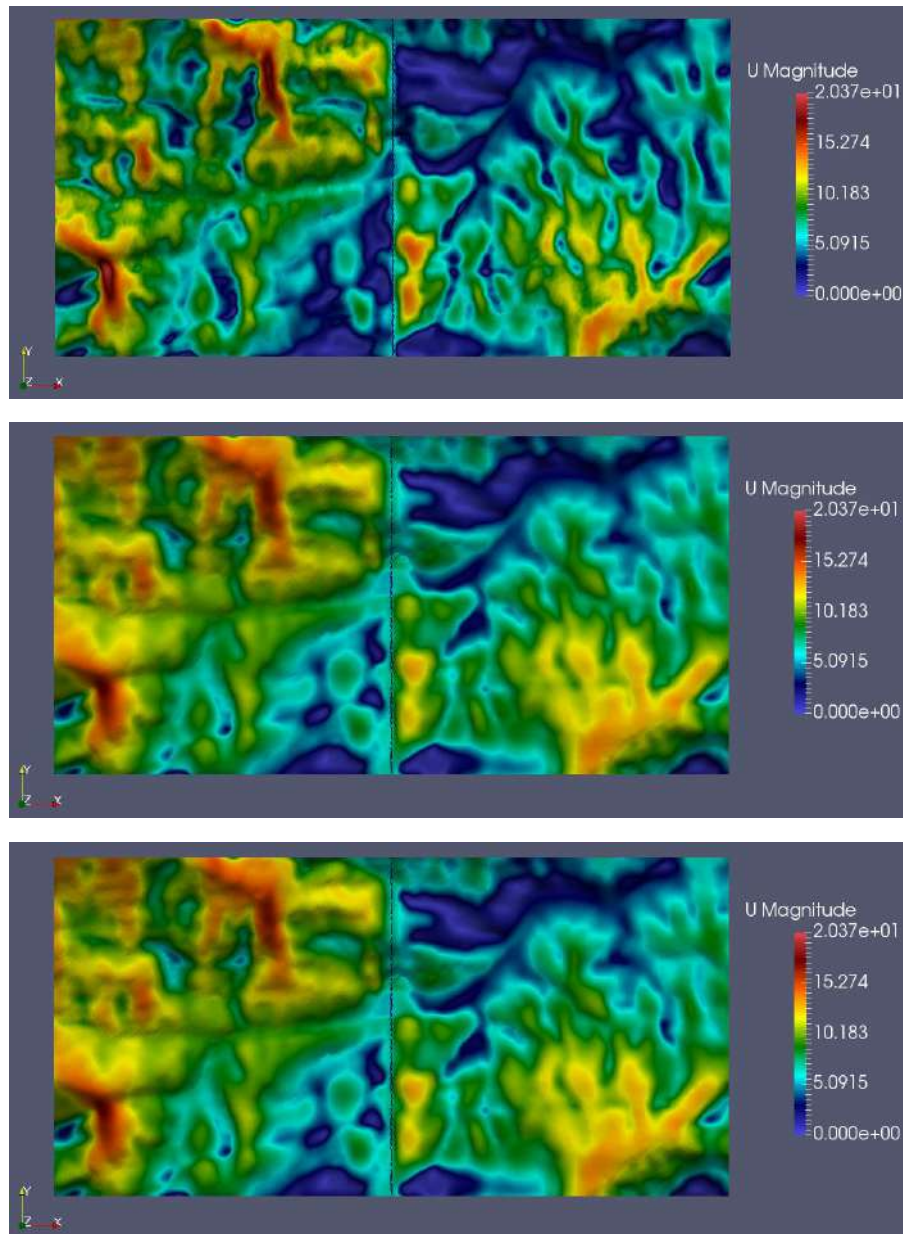


Figure 7.13: Sierra Morena wind speed a) at 10m, b) 20m and c) 30m height.

In general, the wind resource map seems to match the orographic and land cover maps. High wind speeds can be found near the top of the terrain elevations while the lowest are found at the valley bottom and in the areas with  $z_0$  changes. Moreover, the influence of the terrain features diminishes with the height as expected and the wind distribution becomes more uniform.

Due to the changing topographic features, wind resource is very variable within the domain, from very low wind speeds along the valley bottom downwards from Sierra Morena, to wind speeds about 12m/s in the peaks west from the community, at the slope division.



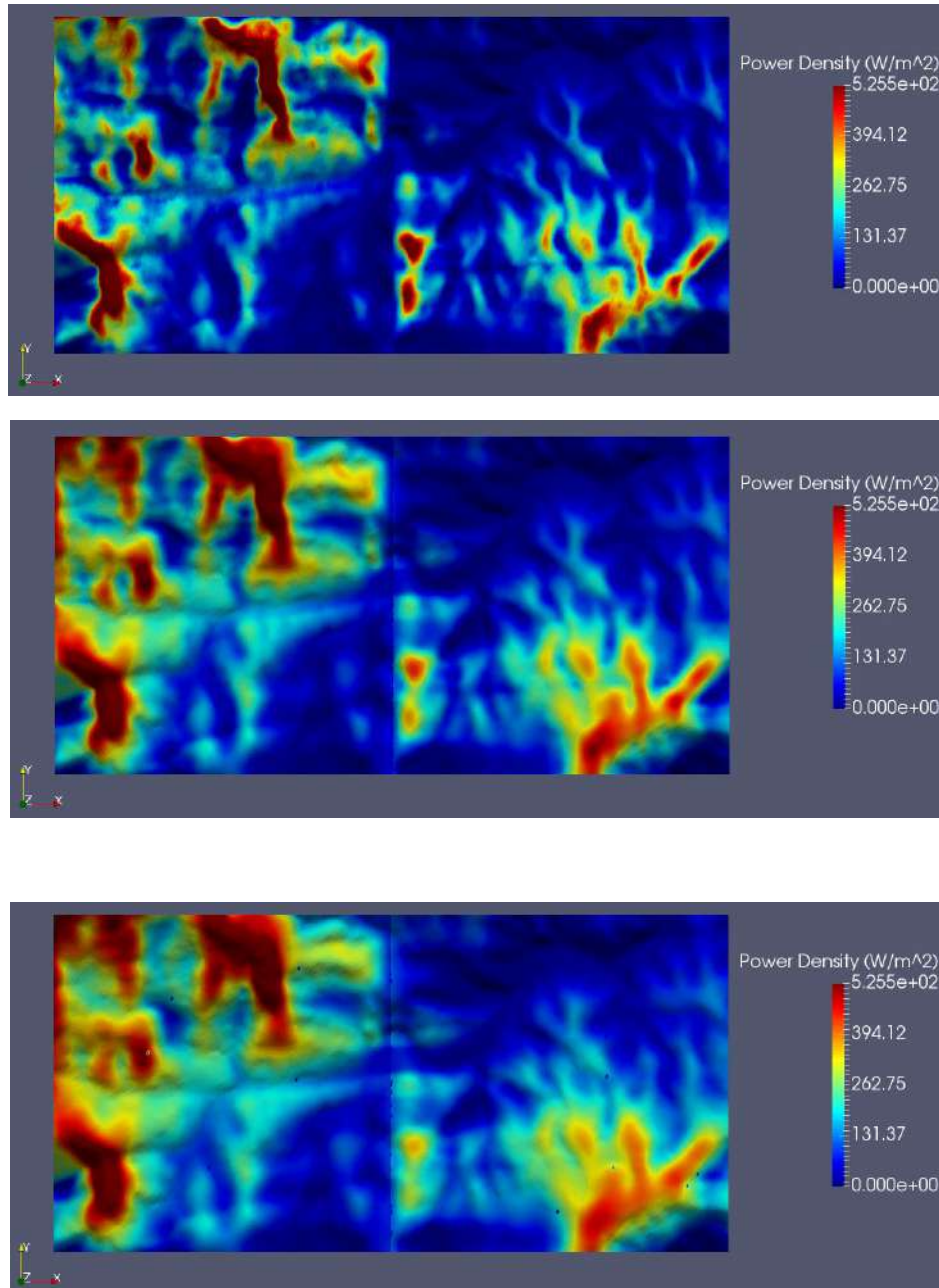


Figure 7.14: Sierra Morena wind power density a) at 10m, b) 20m and c)30m height.

Locations with wind speeds higher than 7m/s and an annual wind power density of at least  $400\text{W/m}^2$  at 50m a.g.l are the most suitable for the installation of wind turbines for commercial purposes. However, wind energy-based rural electrification installation may be feasible from 5.5 m/s wind speeds and  $200\text{W/m}^2$  at 50m a.g.l. [40]. In Fig.7.14 several areas which fulfil the latter conditions can be observed, especially west from Sierra Morena. But, as it was said before, only the two main wind directions and only the wind speed data above 5m/s (approx. 16% of all the wind data and 20% of the wind data in the two main directions) were used for the simulations. For this reason, these maps cannot be taken as an appropriate estimation of the wind energy potential of the region. Simulations were made using all the data available of the wind directions along the valley resulting in a power density map where, except for a few spots, the power density is never above  $200\text{ W/m}^2$ . Therefore, wind power within the domain could be classified as poor according to the NREL [40] classification table and, as a consequence, not suitable for the implementation of wind energy projects.

## CONCLUSIONS

After a wide literature review the  $k$ - $\epsilon$  standard model as well as a set of Wall Functions for the treatment of the solid boundaries were chosen to simulate wind flow over complex terrain. In addition to this, and, in agreement with most of the literature reviewed, the canopy model was adopted to represent forested areas.

The most appropriate OF utilities for the simulation of wind flows were identified and described in detail. Special attention was paid to the meshing process, due to its key role in the capture of the turbulent phenomena. A set of tests were made to identify the most appropriate mesh features, such as the first cell thickness, the mesh vertical grading or the max. aspect ratio, setting as a goal the achievement of a HHTBL. These tests showed that the degree of improvement the correct selection of this parameters may bring is limited and therefore it may be necessary to modify the wall functions source codes if better results are to be achieved.

Four different expressions for the coefficients of the extra momentum term based on the Darcy-Forchheimer Law were tested. None of them provide good results, nevertheless, it was observed that the wind and turbulent intensity profiles generated by those which Forchheimer coefficient was a function of the drag coefficient and the area density profile, manage to resemble the experimental profiles with the Darcy coefficient having very little or no influence at all on the results. Better results may be obtained through a more accurate representation of the forested canopy in terms of the area density profile and the drag coefficient, and by the modification of the code to introduce the corresponding extra terms in the TKE and  $\epsilon$  equations.

The model resulting of the previous tests was validated against the measurements of the *Bolund Experiment*. Meshing the simulation domain posed a challenge for the meshing tools used in this work, *snappyHexMesh* and *cfMesh*. This is because they were designed to mesh much simpler surfaces. For this reason, obtaining the desired accuracy required in the different parts of the domain, while keeping the overall number of cells relatively low is time-consuming and difficult to achieve. In the end, a balance between accuracy and the number of cells was found.

The model managed to predict the general features of wind flow over the hill, such as the speed-up or the pressure drop on the lee side of the hill, and even yield relatively accurate results for wind speed at heights above 5m. The errors, especially for the turbulent intensity, are too high to consider the results as a base for a wind farm project. Although the exceptionally steepness of the Bolund Hill and the position of the sensors, very close to the ground in some cases, may partially explain these results, it is thought that it is in the mesh, which features couldn't be completely controlled, and the OF wall functions, where the causes for these results lie and where future work must focus.

The model was also used to simulate wind flow over a real location in Chipas. Throughout the process which lead to the simulation, a few of the challenges of using a CFD tool to simulate wind on complex terrain, like terrain information processing, lack of appropriate wind information sources or the necessity of using non-uniform inlet profiles were identified and tackled. The resulting wind speed and wind power maps showed that the area may not be suitable for wind energy generation.

Finally, it may be necessary to remind that OpenFOAM is an open source code, and, like most of them, it is more complicated to use than commercial software. In addition to this, its utilities were not specifically designed for wind flow simulation, and therefore, they had to be adapted for this purpose. This may make it an interesting tool for research purposes, for it allows to gain insight in the functioning of each model, however it also leads to time-consuming preparations for each simulation and, in general, renders OF unpractical for systematic wind assessment. Nevertheless, despite all the drawbacks, the results yielded by OF look promising and it might be worth it to develop specific tools and utilities for wind flow simulation or procedures for wind and terrain data processing, which may simplify the whole process. In this respect a few guidelines have been presented in this work, which if pursued, may eventually lead to the developing of a user friendlier CFD-based software for wind assessment.

## References:

### Programs and useful directions:

INEGI: <http://www.inegi.org.mx/>

QGIS: <http://www.qgis.org/es/site/>

OpenFOAM: <http://www.openfoam.com/>

WAsP: <http://www.wasp.dk/>

WindSim: <https://windsim.com/>

CfdOnline: <https://www.cfd-online.com/>

OpenFOAMwikipedia: [https://openfoamwiki.net/index.php/Main\\_Page](https://openfoamwiki.net/index.php/Main_Page)

### General:

[1] United Nations (UN), 2012. *The Millennium Development Goals Report*.

<http://www.un.org/millenniumgoals/pdf/MDG%20Report%202012.pdf>

[2] International Bank for Reconstruction and Development (IBRD), The World Bank (WB) and the International Energy Agency (IEA). (2015). *Sustainable Energy for All. Global Tracking Framework 2015-Full Report*, Washington.

<http://www.se4all.org/sites/default/files/1/2013/09/GTF-2105-Full-Report.pdf>

[3] International Energy Agency (IEA), 2014. *World Energy Outlook 2014*. Paris

<http://www.iea.org/publications/freepublications/publication/WEO2014.pdf>

[4] Szabo, S., Bódis, K., Huld, T. & Moner-Girona, M. (2011). *Energy solutions in rural Africa: Mapping electrification costs of distributed solar and diesel generation versus grid extension*. Environmental Research Letters 6, 3: 034002.

[5] Breyer, Christian. (2012). *Identifying Off-Grid Diesel Grids on a Global Scale for economic Advantageous Upgrading with PV and Wind Power*. Presented at the 5th ARE Workshop Academia meets Industry, in Frankfurt on September 25, 2012. DOI: 10.13140/RG.2.1.4699.5365

[6] World Bank. 2005. *ESMAP 2005-2007 business plan - securing energy for poverty reduction and economic growth*. Washington, DC: World Bank.  
<http://documents.worldbank.org/curated/en/806051468175434798/ESMAP-2005-2007-business-plan-securing-energy-for-poverty-reduction-and-economic-growth>

[7] International Energy Agency (IEA), 2011. *World Energy Outlook 2011: Energy for All Case*. Paris

[http://www.worldenergyoutlook.org/media/weowebiste/2011/weo2011\\_energy\\_for\\_all.pdf](http://www.worldenergyoutlook.org/media/weowebiste/2011/weo2011_energy_for_all.pdf)

[8] Bhattacharyya S. Rural electrification through decentralised off-grid systems in developing countries. Green Energy Technol. 2013; 116.

[9] Global Wind Energy Council (GWEC), 2013. *Global Wind Energy Report 2013*. Brussels.

[http://www.gwec.net/wp-content/uploads/2014/04/GWEC-Global-Wind-Report\\_9-April-2014.pdf](http://www.gwec.net/wp-content/uploads/2014/04/GWEC-Global-Wind-Report_9-April-2014.pdf)

[10] Fraunhofer Institute for Solar Energy Systems (ISE). *Photovoltaics Report*. Freiburg 17<sup>th</sup> November 2014.

<https://www.ise.fraunhofer.de/content/dam/ise/de/documents/publications/studies/Photovoltaics-Report.pdf>

[11] Belward, A., Bisselink, B., Bodis, K., Brink, A., Dallemand, JF., De Roo, A., Huld, T., Kayitakire, F., Mayaux, P., Moner-Gerona, M., Thielen del Pozo, J., Szabo, S., Tromboni, U. and Willemen, L., 2012. *Renewable Energies in Africa-Current Knowledge*. Publication Office of the European Union, ISBN: 978-92-79-22331-0. DOI: 10.2788/1881.

[https://ec.europa.eu/jrc/sites/jrcsh/files/reqno\\_jrc67752\\_final%2520report%2520.pdf](https://ec.europa.eu/jrc/sites/jrcsh/files/reqno_jrc67752_final%2520report%2520.pdf)IRENA2013

[12] International Renewable Energy Agency (IRENA), 2015. *Off-Grid Renewable Energy Systems: Status and Methodological Issues*. Abu Dhabi, United Arab Emirates, 2015.

[http://www.irena.org/DocumentDownloads/Publications/IRENA\\_Off-grid\\_Renewable\\_Systems\\_WP\\_2015.pdf](http://www.irena.org/DocumentDownloads/Publications/IRENA_Off-grid_Renewable_Systems_WP_2015.pdf)

[13] International Energy Agency (IEA), 2013. *World Energy Outlook 2013*. Paris

<http://www.iea.org/publications/freepublications/publication/WEO2013.pdf>

[14] Khanh, Q. Nguyen, 2006. Alternatives to grid extension for rural electrification: Decentralized renewable energy technologies in Vietnam. *Energy Policy*, 2007, vol. 35, issue 4, pages 2579-2589.

[15] Gopakumar, G., Gregory, J., Iyer, V.R., Linn, C., Manuta, J., Smith, W.J., Wagley, S., Wicker, K., Wang, Y., Yamaguchi, T., Zhou, A., & Zhao, J., 2001. Off-Grid Renewable Energy Options for Rural Electrification in Western China. By the Center for Energy and Environmental Policy of University of Delaware Sponsored by National Renewable Energy Laboratory and Ministry of Agriculture People's Republic of China, 2001.

[http://www.ceep.udel.edu/wp-content/uploads/2013/08/2001\\_es\\_China\\_off-grid-energy-system-assessment\\_3-western-provinces.pdf](http://www.ceep.udel.edu/wp-content/uploads/2013/08/2001_es_China_off-grid-energy-system-assessment_3-western-provinces.pdf)

[16] Arriaga, M., Cañizares, C.A., Kazerani, M., (2014). *Northern Lights: Access to Electricity in Canada's Northern and Remote Communities*. *IEEE Power and Energy Magazine*, Vol. 12, Issue 4, pag. 50-59 July-Aug. 2014. DOI: [10.1109/MPE.2014.2317963](https://doi.org/10.1109/MPE.2014.2317963).

[17] Alliance for Rural Electrification (ARE), 2012. *The Potential of Small and Medium Wind Energy in Developing Countries: A guide for energy sector decision-makers*. 1<sup>st</sup> ed. Brussels, Belgium, June 2012.

[https://www.ruralelec.org/sites/default/files/are\\_small\\_wind\\_position\\_paper.pdf](https://www.ruralelec.org/sites/default/files/are_small_wind_position_paper.pdf)

[18] Alliance for Rural Electrification (ARE), 2013. *Best Practices of the Alliance for Rural Electrification*. 3<sup>th</sup> ed. Brussels, Belgium, September 2013.

[https://www.ruralelec.org/sites/default/files/are\\_best\\_practises\\_2013\\_final.pdf](https://www.ruralelec.org/sites/default/files/are_best_practises_2013_final.pdf)

[19] Yadoo, A., 2012. *Delivery Models for Decentralised Rural Electrification. Cases studies in Nepal, Peru and Kenya*. IIED, London (2012). <http://pubs.iied.org/pdfs/16032IIED.pdf>

[20] Sovacool, B.K., 2013. *Energy Access and Energy Security in Asia and the Pacific* (December 1, 2013). Asian Development Bank Economics Working Paper Series No. 383.



<https://www.adb.org/sites/default/files/publication/31154/ewp-383.pdf>

[21] International Energy Agency Wind (IEA Wind), (2010). *Wind Farms in Complex Terrain. 61<sup>st</sup> IEA Topical Expert Meeting*. April 6-10, 2010. POSTECH's POSCO International Center (POSCOIC), Pohang, South Korea.

[https://www.ieawind.org/task\\_11/TopicalExpert/61\\_Wind%20Farms%20in%20Complex%20Terrain.pdf](https://www.ieawind.org/task_11/TopicalExpert/61_Wind%20Farms%20in%20Complex%20Terrain.pdf)

[22] Petersen E.L., Mortensen, N.G., Landberg, L., Højstrup, J., Frank, H., (1998). *Wind Power Meteorology II: Siting and Models*. Wind Energy Journal, Vol.1, Issue 2, Dec.1998, pag. 55-72.

<http://www.ccpo.odu.edu/~klinck/Reprints/PDF/petersen2WindEnergy98.pdf>

[23] Mortensen, N.G., Tindal, A., Landberg, L. (2007). *Field Validation of the delta-RIX performance indicator for flow in complex terrain*. Proceedings of the 2008 European Wind Energy Conference and Exhibition, 2008.

[24] Badger, J. & Ejsing Jørgensen, H., (2011). *A high resolution global wind atlas - improving estimation of world wind resources*. in *Energy Systems and Technologies for the coming Century: Proceedings*. Danmarks Tekniske Universitet, Risø Nationallaboratoriet for Bæredygtig Energi, Roskilde, pp. 215-225. Denmark. Forskningscenter Risoe. Risoe-R, no. 1776(EN).

<http://irena.masdar.ac.ae/>

[25] Adrian, G. and Fiedler, F., 1991. *Simulation of unstationary wind and temperature fields over complex terrain and comparison with observations*. Beitr. Phys. Atmosph., 64:27-48, 1991.

[26] Michalakes, J., Dudhia, J., Gill, D., Klemp, J. and Skamarock, W. 1998. *Design of a next-generation regional weather research and forecast model: Towards Teracomputing*. World Scientific, River Edge, New Jersey, 1998, pp. 117-124.

<http://www.wrf-model.org/index.php>

[27] Jackson, P.S., Hunt, J.C.R., (1975). *Turbulent Flow over a low Hill*. Wind Energy Journal, Vol.101, Issue 430, Oct.1975, pag.929-955.

[28] Troen, I. and Petersen, E. L. 1989. *European Wind Atlas*. Risø National Laboratory, Roskilde, Denmark.

[29] Taylor, P.A. & Teunissen, H.W.1987. *The Askervein Hill Project: Overview and Background Data*. Boundary-Layer Meteorol (1987) 39: 15. doi:10.1007/BF00121863.

[30] Bechmann, A., Sørensen, N.N., Berg, J., (2011). *The Bolund Experiment, Part II: Blind Comparison of Microscale Flow Models*. Boundary-Layer Meteorology (2011) Volume 141,

[31] Mortensen, S.H., Hristov, Y., Knudsen, S.J., Oxley, G.S., 2012. *Validation of CFD wind resource mapping on complex terrain based on WTG performance data*. Presented at the EWEA 2012-European Wind Energy Conference & Exhibition, At Copenhagen in April 2012.

[32] Hargreaves, D.M. and Wright, N.G., 2007. *On the use of the  $k-\epsilon$  model in commercial CFD software to model the neutral atmospheric boundary layer*. Journal of Wind Engineering and Industrial Aerodynamics. 95(5), 355-369.

- [33] Instituto Nacional de Estadística y Geografía (INEGI), 2012. *Perfil sociodemográfico. Estados Unidos Mexicanos. Censo de Población y Vivienda 2010*.  
<http://www3.inegi.org.mx/sistemas/biblioteca/ficha.aspx?upc=702825047610>
- [34] World Bank. 2007. *Mexico - (CRL) Rural Electrification Project*. Washington, DC: World Bank. <http://documents.worldbank.org/curated/en/547771468049167553/Mexico-CRL-Rural-Electrification-Project>
- [35] Instituto Nacional de Estadística y Geografía (INEGI), 2012. *Perspectiva Estadística Chiapas. Diciembre 2012*.
- [36] Gobierno del Estado de Chiapas, Secretaría de Hacienda, 2010. *Encuestas de ingresos y gastos en los hogares (ENIGH). Chiapas 2010*.  
<http://ceieg.chiapas.gob.mx/home/wp-content/uploads/downloads/2012/01/Principales-resultados-de-la-ENIGH-Chiapas-2010.pdf>
- [37] Gerlach A-K., Gaudchau E., Cader C., Wasgindt V., Breyer, Ch. 2013. *Comprehensive country ranking for renewable energy based mini-grids providing rural off-grid electrification*. In: Proceedings of the 28th EU PVSEC. Paris; September 30–October 4, 2013.  
<http://dx.doi.org/10.4229/28thEUPVSEC2013-5DO.13.2>.  
[http://reiner-lemoine-institut.de/wp-content/publications/1\\_Comprehensive\\_country\\_ranking/Gerlach2013.pdf](http://reiner-lemoine-institut.de/wp-content/publications/1_Comprehensive_country_ranking/Gerlach2013.pdf)
- [38] International Renewable Energy Agency (IRENA), 2015. *Renewable Energy Prospects: México*. Abu Dhabi, United Arab Emirates .  
[http://www.irena.org/DocumentDownloads/Publications/IRENA\\_REmap\\_Mexico\\_report\\_2015.pdf](http://www.irena.org/DocumentDownloads/Publications/IRENA_REmap_Mexico_report_2015.pdf)
- [39] World Bank. 2016. *Mexico - Integrated Energy Services Project*. Washington, D.C. : World Bank Group.  
<http://documents.worldbank.org/curated/en/195401473187206344/Mexico-Integrated-Energy-Services-Project>
- [40] Elliot, D., Schwartz, M., Scott, G., Haymes, S., Heimiller, D., George, R. 2003. Wind Energy Resource Atlas of Oaxaca. National Renewable Energy Laboratory (NREL), Golden, CO.; 2003 Aug. Report No.: NREL/TP-500-34519.  
<http://www.nrel.gov/docs/fy03osti/34519.pdf>
- [41] International Energy Agency Wind (IEA Wind), 2014. *IEA Wind 2014 Annual Report*. Paris  
[https://www.ieawind.org/annual\\_reports\\_PDF/2014/2014%20AR\\_smallfile.pdf](https://www.ieawind.org/annual_reports_PDF/2014/2014%20AR_smallfile.pdf)
- [42] Secretariat of Energy (SENER, Secretaría de Energía), 2012. *Renewable Energies Perspective 2012-2016 (Prospectiva de Energías Renovables 2012-2016)*. Mexico City: 2012.
- [43] Stull, R. B., (1988): *An Introduction to Boundary Layer Meteorology*. Kluwer Academic Publishers, 666 pp.
- [44] Gasch, R., Twele, J.(Eds), (2014). *Windkraftanlagen: Grundlagen, Entwurf, Planung und Betrieb*. 5<sup>th</sup> ed. Berlin. Springer, 2014.
- [45] Petersen, Erik L. Mortensen, Niels G. Landberg, Lars. Højstrup, Jørgen. Frank, Helmut P., (1998) *Wind Power Meteorology. Part I: Climate and Turbulence*. Wind Energy Journal,

Volume 1, Issue S1, April 1998, Pages 25-45. DOI: 10.1002/(SICI)1099-1824(199804)1:1+<25::AID-WE4>3.0.CO;2-D.

[46] Berg, J., Mann, J., Bechmann, A., (2011). *The Bolund Experiment, Part I: Flow over a Steep, Three Dimensional Hill*. Boundary-Layer Meteorology (2011) Volume 141, Number 2, Page 219. doi:10.1007/s10546-011-9636-y.

[47] Da Silva, J.M., 2013. *Development of procedures for the simulation of atmospheric flows over complex terrain, using OpenFOAM*. Master's Thesis in the Mechanical Engineering Department, School of Engineering, Polytechnic of Porto, Porto, Portugal, 2013. <http://recipp.ipp.pt/handle/10400.22/4691>

[48] Sagaut, P. (2001). *Large Eddy Simulations for Incompressible Flows*. 3<sup>th</sup> ed. Berlin. Springer, 2006.

[49] Geurts, B. (2004). *Elements of direct and large-eddy simulation*. RT Edwards, 2004.

[50] Chow, F. K., Street, R. L., (2004). *Evaluation of turbulence models for large-eddy simulations of flow over Askervein hill*. Preprints, 16th Symp. on Boundary Layers and Turbulence, Portland, ME, Amer. Meteor. Soc., 7.11.

[51] Chow, F. K., Street, R. L., (2009). *Evaluation of turbulence closure models for large-eddy simulation over complex terrain: Flow over Askervein Hill*. J. Appl. Meteor. Climatology, **48**, 1050–1065, doi:[10.1175/2008JAMC1862.1](https://doi.org/10.1175/2008JAMC1862.1)

[52] Rasouli, A., Hangan, H., (2013). *Microscale Computational Fluid Dynamics Simulation for Wind Mapping Over Complex Topographic Terrains*. Journal of Solar Energy Engineering 135(4):041005, Nov.2013. doi: 10.1115/1.4024124.

[53] Bechmann, A. and Sørensen, N. N., (2011). *Hybrid RANS/LES applied to complex terrain*. Wind Energy, 14: 225–237. doi:10.1002/we.414.

[54] Bechmann, A. and Sørensen, N. N., (2010). *Hybrid RANS/LES method for wind flow over complex terrain*. Wind Energy, 13: 36–50. doi:10.1002/we.346.

[55] Tryggvason, G., 2011. *Classical Turbulence Modeling*.

URL: <http://www3.nd.edu/~gtryggva/Turbulence-Intro.pdf>

[56] Crastro, G., (2007). *Numerical Simulations of the Atmospheric Boundary Layer*. Dottorato di Ricerca in Ingegneria Industriale XIX Ciclo, Università degli Studi di Cagliari, Facoltà di Ingegneria, 2007.

URL:

[https://windsim.com/documentation/papers\\_presentations/thesis/0702\\_Giorgio\\_Crasto.pdf](https://windsim.com/documentation/papers_presentations/thesis/0702_Giorgio_Crasto.pdf)

[57] Wallbank, T., (2008). *WindSim Validation Study: CFD Validation in Complex Terrain*.

[http://www.windsim.com/documentation/papers\\_presentations/thesis/080512trw%20WindSim%20Write%20Up%20-%20Validation%20study.pdf](http://www.windsim.com/documentation/papers_presentations/thesis/080512trw%20WindSim%20Write%20Up%20-%20Validation%20study.pdf)

[58] Stangroom, P., (2004). *CFD modelling of wind flow over terrain*. PhD Thesis, University of Nottingham, 2004. URL: <http://eprints.nottingham.ac.uk/10112/1/Stangroom.pdf>

[59] Pope, S.B., (2000). *Turbulent Flows*. 4<sup>th</sup> ed. Cambridge. Cambridge University Press, 2006.

[60] ANSYS FLUENT 14.0 Theory Guide. ANSYS Inc., 2011 : <https://es.scribd.com/doc/140163341/Ansys-Fluent-14-0-Theory-Guide>.



- [61] Sumner, J., (2012). *Towards improved RANS/ $k-\epsilon$  modelling of turbulent incompressible flows for wind energy applications*. Thèse de doctorat électronique, Montréal, École de technologie supérieure, 2012. URL: [http://espace.etsmtl.ca/1057/1/SUMNER\\_Jonathon.pdf](http://espace.etsmtl.ca/1057/1/SUMNER_Jonathon.pdf)
- [62] Launder B. E., Spalding, D. B., (1972). *Mathematical Models of Turbulence*. 169 S. m. Abb. London/New York 1972. Academic Press, 1972.
- [63] Launder, B.E., Spalding, D.B., (1974). [\*The numerical computation of turbulent flows\*](#). *Computer Methods in Applied Mechanics and Engineering* 3 (2): 269-289, 1974.
- [64] Boussinesq, J. (1877). *Essai sur la théorie des eaux courantes*. Mémoires présentés par divers savants à l'Académie des Sciences 23 (1): 1-680.
- [65] Prandtl, L., 1925. *Über die ausgebildete Turbulenz*. ZAMM, Vol. 5, pp. 136-139.
- [66] Versteeg, H. K., Malalasekera, W., (1995). *An introduction to Computational Fluid Dynamics: The Finite Volume Method*. Longman Scientific & Technical, 1995.
- [67] Bredberg, J., (2000). *On the Wall Boundary Condition for Turbulence Models*. Technical report, Chalmers University of Technology, 2000.
- [68] Blocken B., Stathopoulos T., Carmeliet J., (2007). *CFD simulation of the atmospheric boundary layer: wall function problems*. *Atmospheric Environment* 41(2): 238-252 , 2007. URL: [http://www.urbanphysics.net/2007\\_AE\\_CFD\\_ABL\\_simulation\\_Preprint.pdf](http://www.urbanphysics.net/2007_AE_CFD_ABL_simulation_Preprint.pdf)
- [69] OpenFOAM User's Guide: <http://www.openfoam.com/documentation/user-guide/>
- [70] Davidson, L. and Farhanieh, B., 1995. CALC-BFC: A Finite-Volume Code Employing Collocated Variable Arrangement and Cartesian Velocity Components for Computation of Fluid Flow and Heat Transfer in Complex Three-Dimensional Geometries. Rept. 95/11, Thermo- and Fluid Dynamics, Chalmers University of Technology, Göteborg, 1995.
- [http://www.tfd.chalmers.se/~lada/homepage\\_papers/CALC-BFC\\_a\\_finite-volume\\_code.html](http://www.tfd.chalmers.se/~lada/homepage_papers/CALC-BFC_a_finite-volume_code.html)
- [71] Kalitzin, G., Medic, G., Iaccarino, G., and Durbin, P., Near-Wall Behavior of RANS Turbulence Models and Implications for Wall Functions, *Journal of Computational Physics*, vol. 204, pp. 265–291, 2005.
- [72] Adams, T., Grant, C., Watson, H., *A simple algorithm to relate measured surface roughness to equivalent sand-grain roughness*. *Int. J. Mech. Eng. Mechatron.* 1(1), 66–71 (2012) <http://ijmem.avestia.com/2012/PDF/008.pdf>.
- [73] Nikuradse, J., 1993. *Stromungsgesetze in Rauhren Rohren*. Vol.361.V.D.I.Forschungshefte, pp.1–22.
- [74] Cebeci, T., Bradshaw, P., 1977. *Momentum Transfer in Boundary Layers*. Hemisphere Pub. Corp. Washington, 1977.
- [75] Zapka, M.J., Jatupatwarangkul, S., Tran, T., 2013. *Literature Review of External CFD*. Hawaii Natural Energy Institute Environmental Research and Design Laboratory School of Architecture, Honolulu, Hawaii, U.S.A, 2013.
- <http://www.hnei.hawaii.edu/sites/www.hnei.hawaii.edu/files/Final%20CFD%20Literature%20Review%20Report%20-%20D1%20-%20External%20CFD.pdf>
- [76] Mortensen, N., 2007. *Getting started with WAsP 9*. Risø-I-2571(EN), Risø National Laboratory, Roskilde, Denmark, 2007.

- [77] Finnigan, J.J., 2000. *Turbulence in plant canopies*. Annu. Rev. Fluid Mech. 32, 519–571.
- [78] Böhm, M., Finnigan, J.J., Raupach, M.R., 2013. *Turbulence in and above canopies*. Boundary-Layer Meteorol. (2013) 146: 393. doi:10.1007/s10546-012-9770-1.
- [79] Dellwik, E., Landberg, L. & Jensen, N.O., 2005. *WAsP in the forest*. Wind Energy, Vol 9, pp. 211-218. DOI: 10.1002/we.155.
- [80] Zeleti, Z.A., Haario, H., Hämäläinen, J., 2014. *Experimental Validation of porous medium modelling for air flow through forest canopy*. Presented at EWEA-2014 Conference, in Barcelona, Spain, March 2014.
- [81] Bitog, J. P., Lee, I. B., Hwang, H. S., Shin, M. H., Hong, S. W., Seo, I. H., 2012. *Numerical simulation study of a tree windbreak*. Biosystems Engineering, 111(1), 40e48.
- [82] Andersen, M.Q., Mortensen, K., Nielsen, D.E., Laursen, J., Bingöl, F. & Sogachev, A., 2009. *Development and Verification of CFD Models for Modeling Wind Conditions on Forested Wind Turbine Sites*. In *EWEC 2009 Proceedings*. vol. 6, EWEC, pp. 3758-3764. [http://orbit.dtu.dk/files/79351202/Development\\_and\\_Verification\\_of\\_CFD\\_Models.pdf](http://orbit.dtu.dk/files/79351202/Development_and_Verification_of_CFD_Models.pdf)
- [83] Dalpé, B. and Masson, C., 2009. *Numerical simulation of wind flow near a forest edge*. J. Wind Eng. Ind. Aerodyn. 97:228–241.
- [84] Katul, G., Mahrt, L., Poggi, D., Sanz, C., 2004. *One- and two-equation models for canopy turbulence*. Bound. Layer Meteor. 113, 81–109. doi:10.1016/ 0167-6105 (93) 90124-7.
- [85] Wilson, N.R., Shaw, R.H., 1977. *A higher order closure model for canopy flow*. Journal of Applied Meteorology 16:1197-1205. DOI: 10.1175/1520-0450(1977)016<1197:ahocmf>2.0.co;2.
- [86] Lalic, B. & Mihailovic, D.T., 2004. *An Empirical relation describing Leaf Area Density inside the forest for environmental modeling*. Journal of Applied Meteorology and Climatology, 43:641–645
- [87] Pekin, B., Macfarlane, C., 2009. *Measurement of crown cover and leaf area index using digital cover photography and its application to remote sensing*. Remote Sens. 2009, 1, 1298–1320.
- [88] Gromke, C., & Ruck, B., (2008). *Aerodynamic modelling of trees for small-scale wind tunnel studies*. Forestry, 81(3), 243–258.
- [89] Mochida, A., Tabata, Y., Iwata, I., Yoshino, H., 2008. *Examining tree canopy models for CFD prediction of wind environment at pedestrian level*. J. Wind Eng. Ind. Aerodyn., 96 (2008), pp. 1667–1677. <http://www.aij.or.jp/jpn/publish/cfdguide/JWEIAtree.pdf>
- [90] Sanz, C., 2003. *A note on k-epsilon modelling of vegetation canopy air flows*. Boundary-Layer Meteorology 108, pp. 191-197
- [91] Ferziger, J.H., Peric, M., 2002. *Computational Methods for Fluid Dynamics*. 3<sup>th</sup> ed. Berlin Heidelberg. Springer, 2002.
- [92] Bakker, A. Computational Fluid Dynamics Course. Lecture 5: Finite Volume Solvers. Dartmouth College 2001-2006. <http://www.bakker.org/dartmouth06/engs150/>

[93] Leonard, B. P., 1979. *A stable and accurate convective modelling procedure based on quadratic upstream interpolation*. Comp. Meth. Appl. Mech. & Eng. 19, p59-98.

[94] Patankar, S. V. and Spalding, D.B., (1972). *A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows*. Int. J. of Heat and Mass Transfer, Volume 15, Issue 10, October 1972, Pages 1787-1806.

[95] ANSYS 12.0 Fluent Users Guide.

<http://www.afs.enea.it/project/neptunius/docs/fluent/html/ug/node3.htm>

Launder B. E., Spalding, D. B., (1972). *Mathematical Models of Turbulence*. 169 S. m. Abb. London/New York 1972. Academic Press, 1972.

[96] The WAsP Team. *WAsP best practices and checklist*. Danmarks Tekniske Universitet (DTU) Wind Energy, February 2016.

[97] Franke, J., Hirsch, C., Jensen, A.G., Krüs, H.W., Schatzmann, M., Westbury, P.S., Miles, S.D., Wisse, J.A., Wright, N.G., 2004. *Recommendations on the use of CFD in wind engineering*. Proceedings of the International Conference on Urban Wind Engineering and Building Aerodynamics, in: van Beeck JPAJ (Ed.), COST Action C14, Impact of Wind and Storm on City Life Built Environment, von Karman Institute, Sint-Genesius-Rode, Belgium, 5 - 7 May 2004.

<https://www.kuleuven.be/bwf/projects/annex41/protected/data/Recommendations%20for%20CFD%20in%20wind%20engineering.pdf>

[98] Mortensen, N.G. and Petersen, E.L., 1997. *Influence of Topographical Input Data on the accuracy of Wind Flow Modeling in Complex Terrain*, At the European Wind Energy Conference 1997, in Dublin, Ireland, October 1997.

[99] Berge E., Gravdahl A. R., Schelling J., Tallhaug L. and Undheim O., 2006. *Wind in complex terrain. A comparison of WAsP and two CFD-models*. In: Proceedings from EWEC 27 [http://www.windchina.info/cwpp/files/1613/9821/8733/WAsPCFD\\_WAsP\\_WindSim\\_ewec\\_berge.pdf](http://www.windchina.info/cwpp/files/1613/9821/8733/WAsPCFD_WAsP_WindSim_ewec_berge.pdf)

[100] Roache, P. J., 1997. *Quantification of uncertainty in computational fluid dynamics*. Annual Review of Fluid Mechanics, 29, 123–160.

[101] Roache, P.J., Ghia, K.N. and White, F.M., 1986. *Editorial Policy Statement on the Control of Numerical Accuracy*. J. Fluids Eng 108(1), 2 (Mar 01, 1986). doi:10.1115/1.3242537.

[102] Richards, P.J., Hoxey, R.P., 1993. *Appropriate boundary conditions for computational wind engineering models using the k-ε turbulence model*. J. Wind Eng. Ind. Aerodyn. 46 & 47, 145–153.

[103] Antham, R., 2016. *Modelling of a chemical batch reactor*. Project work developed for the CFD with OpenSource Software Course at Chalmers University of Technology taught by Håkan Nilson, 2016.

[http://www.tfd.chalmers.se/~hani/kurser/OS\\_CFD\\_2015/RajukiranAntham/OpenFoam\\_Report.pdf](http://www.tfd.chalmers.se/~hani/kurser/OS_CFD_2015/RajukiranAntham/OpenFoam_Report.pdf)

[104] Gooya, R., 2014. *Description of porousSimpleFoam and adding the Brinkmann model to the porous models*. Project work developed for the CFD with OpenSource Software Course at Chalmers University of Technology taught by Håkan Nilson, 2014.

[http://www.tfd.chalmers.se/~hani/kurser/OS\\_CFD\\_2013/RezaGooya/Final%20OF.pdf](http://www.tfd.chalmers.se/~hani/kurser/OS_CFD_2013/RezaGooya/Final%20OF.pdf)

[105] Hafsteinsson, H.E, 2009. *Porous Media in OpenFOAM*. Chalmers University of Technology. Thecnical Report, 2009.

[http://www.tfd.chalmers.se/~hani/kurser/OS\\_CFD\\_2008/HaukurElvarHafsteinsson/haukurReport.pdf](http://www.tfd.chalmers.se/~hani/kurser/OS_CFD_2008/HaukurElvarHafsteinsson/haukurReport.pdf)

[106] Gravdahl, A.R., Crasto, G., Quinn, S., Mandas, N., Cambuli, F., 2006. *Canopy modelling with a CFD code*. EWEC, Athens, 2006.

[https://windsim.com/documentation/papers\\_presentations/0602\\_ewec/ewec\\_crasto.pdf](https://windsim.com/documentation/papers_presentations/0602_ewec/ewec_crasto.pdf)

[107] Nield, D.A., and Bejan, A., 2013. *Convection in Porous Media*. 4<sup>th</sup> ed. Springer, New York, 2013.

[108] Bear, J., 1972. *Dynamics of Fluids in Porous Media*. Elsevier, New York 1972

[109] Sobieski, W. & Trykozko, A., 2011. *Sensitivity Aspect of Forchheimer's Approximation*. Transp. Porous. Med. (2011) 89: 155. doi:10.1007/s11242-011-9760-7

[110] Darcy, P. G. 1856. *Les Fontaines Publiques de la Ville de Dijon*. Victor Dalmont, Paris, 1856

[111] Bear, J. and Bachmat, Y., 1986. *Macroscopic modelling of transport phenomena in porous media. 2: Applications to mass, momentum and energy transport*. Transport in Porous Media, 1:241–269, 1986. 10.1007/BF00238182

[112] Helmig, R., 1997. *Multiphase Flow and Transport Processes in the Subsurfaces : A Contribution to the Modelling of Hydrosystems*. Springer, 1997.

[113] Forchheimer, P., 1901. *Wasserbewegung durch Boden*. Zeitschrift des Vereines Deutscher Ingenieur, 45:1782-1788, 1901.

[114] Huang, H., Ayoub, J., 2008. *Applicability of the Forchheimer Equation for Non-Darcy Flow in Porous Media*. SPE Journal, 13(1): 112–122.

[115] Castro, F.A., Palma, J.M.L.M. & Silva Lopes, A., 2003. *Simulation of the Askervein Flow. Part I: Reynolds averaged Navier-Stokes Equations ( $k-\epsilon$  Turbulence Model)*. Boundary-Layer Meteorology (2003) 107: 501. doi:10.1023/A:1022818327584

[116] Pedruelo, X., 2009. *Modelling of wind flow over complex terrain using OpenFoam*. Master's Thesis at the Energy Systems at the Department of Technology and built environment, University of Gävle, Gävle, Sweden, June 2009.  
<http://www.diva-portal.org/smash/get/diva2%3A228936/FULLTEXT01.pdf>

[117] Sumner, J. & Masson, C., 2010. *k-epsilon Simulations of the Neutral ABL: Achieving Horizontal Homogeneity on Practical Grids*. AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, (p. 1373). Orlando.

[118] Walshe, J., 2003. *CFD Modelling of Wind Flow over complex terrain*. PhD Thesis for the award of Doctor of Philosophy of Loughborough University, Loughborough, United Kingdom, 2003.  
<https://dspace.lboro.ac.uk/dspace-jspui/handle/2134/7827>

- [119] Li, W., Wang, F. & Bell, S., 2007. *Simulating the sheltering effects of windbreaks in urban outdoor open space*. *Journal of Wind Engineering and Industrial Aerodynamics*, Vol 95, no. 7, pp. 533-549. DOI: [10.1016/j.jweia.2006.11.001](https://doi.org/10.1016/j.jweia.2006.11.001)
- [120] Behrens, T., 2009. *OpenFOAM's basic solvers for linear systems of equations. Solvers, preconditioners, smoothers*. Technical report, Technical University of Denmark, Lingby, Denmark, 2009.
- [http://www.tfd.chalmers.se/~hani/kurser/OS\\_CFD\\_2008/TimBehrens/tibeh-report-fin.pdf](http://www.tfd.chalmers.se/~hani/kurser/OS_CFD_2008/TimBehrens/tibeh-report-fin.pdf)
- [121] C. Peralta, C., Nugusse, H., Kokilavani, S.P., Schmidt, J. and Stoevesandt, B., 2013. *Validation of RANS solvers in OpenFOAM for the atmospheric boundary layer in complex terrain*. First Symposium on OpenFOAM in Wind Energy, IWES, Oldenburg, Germany, March 2013. <https://doi.org/10.1051/itmconf/20140201002>
- [http://www.itm-conferences.org/articles/itmconf/pdf/2014/01/itmconf\\_01002.pdf](http://www.itm-conferences.org/articles/itmconf/pdf/2014/01/itmconf_01002.pdf)
- [122] Parente, A., Gorlé, C., van Beeck, J., 2011. *A Comprehensive Modelling Approach for the Neutral Atmospheric Boundary Layer: Consistent Inflow Conditions, Wall Function and Turbulence Model*. *Boundary-Layer Meteorol* (2011) 140: 411. doi:10.1007/s10546-011-9621-5
- [123] Irvine, M.R., Gardiner, B.A., Hill, M.K., 1997. *The evolution of turbulence across a forest edge*. *Boundary-Layer Meteorol.* 84:467–496.
- [124] Jarvis, P. J., James, G. B., and Landsberg, J. J., 1976. *Coniferous Forest*. In J. L. Monteith (ed.), *Vegetation and the Atmosphere*, Vol. 2, Academic Press, London, pp. 171–240.
- [125] Katul, G.G., Mahrt, L., Poggi, D., Sanz, C., 2004. *One- and two-equations models for canopy turbulence*. *Bound-Lay. Meteorol.* 113:81-109, 2004.
- [https://nicholas.duke.edu/people/faculty/katul/katul\\_blm\\_k\\_epsilon.pdf](https://nicholas.duke.edu/people/faculty/katul/katul_blm_k_epsilon.pdf)
- [126] Kaimal J.C., Finnigan J.J. (1994). *Atmospheric boundary layer flows: Their structure and measurement*. Oxford University Press, New York, NY.
- [127] Bechmann, A., Berg, J., Courtney, M.S., Jørgensen, H.E., Mann, J., Sørensen, N.N., 2009. *The Bolund experiment: overview and background*. Risø DTU report Risø-R-1658(EN), pp 1-50. [http://orbit.dtu.dk/fedora/objects/orbit:81901/datastreams/file\\_4321515/content](http://orbit.dtu.dk/fedora/objects/orbit:81901/datastreams/file_4321515/content)
- [128] Taylor, P. A. and Teunissen, H. W., 1983. *ASKERVEIN '82: Report on the September/October 1982 Experiment to Study Boundary-Layer Flow Over Askervein, South Uist*. Research Report MSRB-83–8, Atmos. Environ. Service, Downsview, Ont., Canada (ASK82).
- [129] Taylor, P. A. and Teunissen, H. W., 1985. *The Askenein Hill Project: Report on the September/October 1983 Main Field Experiment*. Internal Report MSRB-84–6, Atmos. Environ. Service, Downsview, Ont., Canada (ASK83).
- [130] Taylor, P.A. & Teunissen, H.W., 1987. *The Askervein Project: Overview and Background data*. *Boundary-Layer. Meteorol.* (1987) 39: 15. doi:10.1007/BF00121863.
- [131] Comisión Nacional del Agua (CONAGUA): <http://www.gob.mx/conagua>

- [132] Instituto Nacional de Investigaciones Forestales, Agrícolas y Pecuarias (INIFAP): <http://www.gob.mx/conagua>
- [133] Ranaboldo, M., Ferrer-Martí, L., Velo, E., 2014. *Micro-scale wind resource assessment for off-grid electrification projects in rural communities*. A case study in Peru. Int. J. Green Energy 2014;11(1):75–90. [http://dx.doi.org/10\(1080/15435075\), 2013,769878](http://dx.doi.org/10(1080/15435075), 2013,769878).
- [134] Mortensen, N.G., Tindal, A., Landberg, L. (2007). *Field Validation of the delta-RIX performance indicator for flow in complex terrain*. Proceedings of the 2008 European Wind Energy Conference and Exhibition, 2008.
- [135] Bowen, A.J., Mortensen, N.G., 1996. *Exploring the limits of WAsP: the wind atlas analysis and application program*. Published in: Proceedings of the 1996 European Union Wind Energy Conference, vol. 1, 1996, pp. 584e587.
- [136] Riley, S.J., DeGloria, S.D., Elliot, R., 1999. *A terrain ruggedness index that quantifies topographic heterogeneity*. Intermount. J. Sci. 5, 23–27.
- [137] Wood, N., 1995. *The onset of separation in neutral, turbulent flow over hills*. Boundary-Layer Meteorol (1995) 76: 137. doi:10.1007/BF00710894
- [138] World Wind Energy Association (WWEA), 2013. Small Wind World Report Update 2013. Bonn, Germany. <http://small-wind.org/wwea-publishes-half-year-report-2013/>
- [139] Troen, I. (1990). *A high resolution spectral model for flow in complex terrain*. Ninth Symposium on Turbulence and Diffusion, Roskilde, April 30 - May 3, 1990, 417-420.



# Annex 1: Discretization Schemes and Solvers.

## Speed-Up and Lee-side phenomena

Terrain features, such as hills and ridges, may induce an acceleration on the flow usually referred as the Speed-Up, which reaches its maximum value at the top of the elevation. On the lee side of an elevation however, the flow slows down and if the slope is steep enough (ca.20° according to [27]) separated flow regions and recirculation zones may appear, which lead to the increasing of TKE. These two characteristic features of wind flow over complex terrain are of the utmost importance for wind energy purposes. The election of the appropriate discretization scheme, solver and smoother is essential for the modelling of such phenomena. To test the performance of a few of them 2D-flow simulations were carried out in a domain of 5000x100x1000m with a two-dimensional symmetric circular hill in its centre, which shape is given by Eq.1 (see [59]). These simulations offer the possibility to observe the above-mentioned phenomena (see Table 1) and are a common practice in works on turbulent flow over complex terrain [79,97,27,59].

$$\begin{cases} z(x) = H \cos^2 \left( \pi \frac{(x - x_0)}{2L} \right) \text{ for } x - x_0 \leq L \\ z(x) = 0 \text{ for } x - x_0 > L \end{cases} \quad (1)$$

While others may vary, the following parameters will remain unchanged in every simulation. Here  $x_0$  is the x-coordinate of the hill top from the inlet,  $H$  the height and  $L$  the radius of the circular base.

$N_y$	$N_z$	$R_{EXP}$	$x_0(m)$	$L(m)$	$H(m)$	$Z_P(m)$	$z_0$	$H_{REF}(m)$	$U_{REF}(m/s)$	$u_\tau(m/s)$
5	50	1.096	2500	240	80	0.5	0.1	500	16.91	0.753

Table 1:2D-Hill Simulation Domain features.

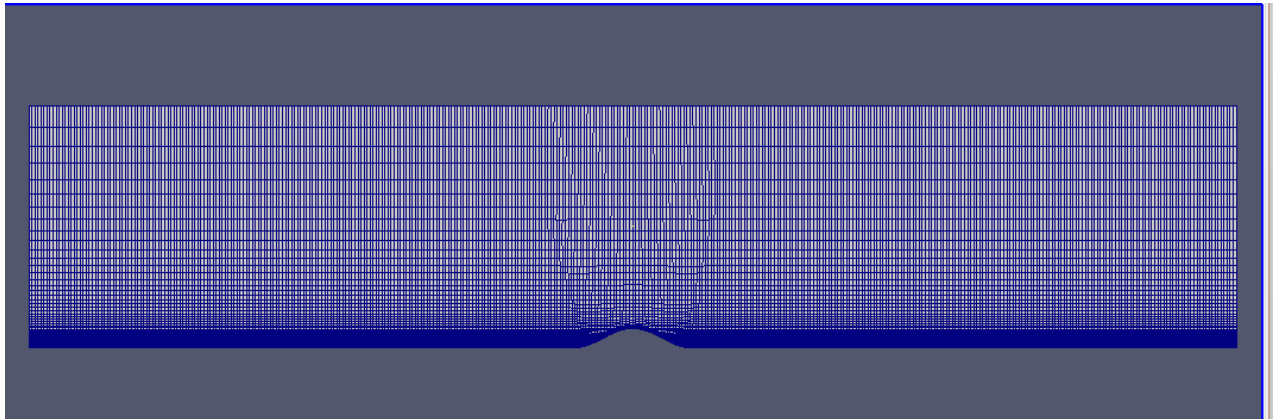


Figure 1: 2D-Hill Simulation Domain Mesh.

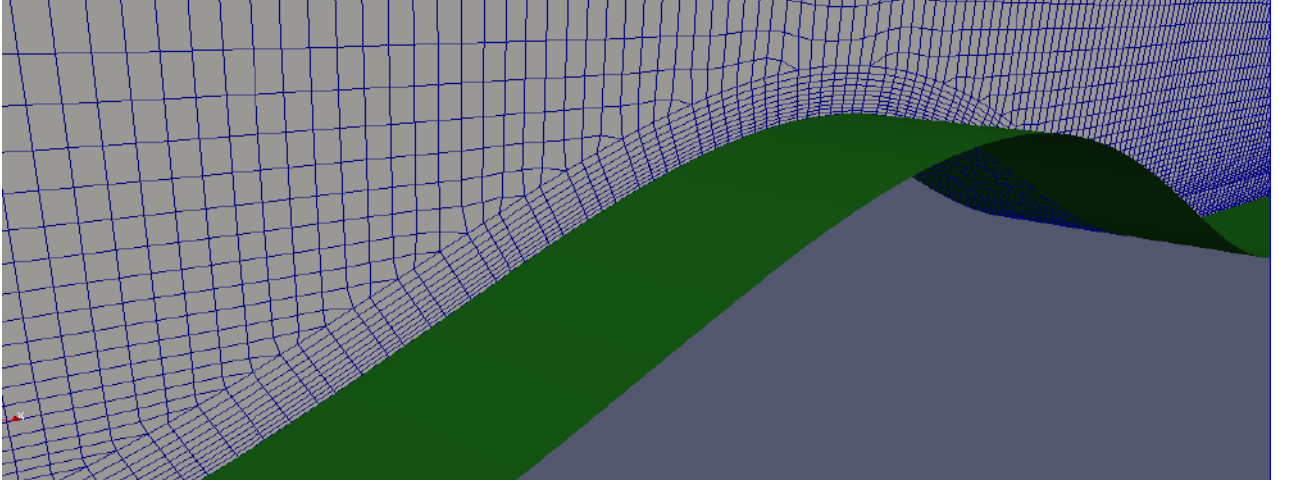


Figure 2: 2D-Hill near-ground meshing.

The mesh was created with *snappyHexMesh*. In order to guarantee the vertical resolution and grading as well as the position of the first node, a total of 10 layers ( $\sim 15\text{m}$ ) (Fig.2) were used. The mesh quality check didn't yield any errors.

Following [127], two indicators, the non-dimensional Speed-Up fraction and the normalized turbulent kinetic energy, defined in Eq. 2 and 3 respectively, were used to describe wind speed acceleration and turbulence increasing on the lee side. Both are evaluated at a certain reference height  $z_{ref}$ .

$$S(z_{ref}) = \frac{u(z_{ref}) - u_{inlet}(z_{ref})}{u_{inlet}(z_{ref})} \quad (2)$$

$$TKE_N(z_{ref}) = \frac{TKE(z_{ref})}{u_{inlet}^2(z_{ref})} \quad (3)$$

## Mesh Sensitivity

Mesh sensitivity and ultimately grid independency are usually tested by simulating flows in domains with meshes of changing resolution and comparing the results. For this purpose, simulations with 5 different resolution values in the x direction were carried out.

Mesh	1	2	3	4	5
$\Delta x(\text{m})$	12	10	8	5	2
N°Cells	118010	146870	183150	292640	731970

Table 2: Grid Independency Simulation Settings

The value of the highest horizontal resolution was chosen according to the most common values found in the literature (see Ch.4). The lowest resolution value was the smallest cell size for which *snappyHexMesh* was able to introduce the layers required to obtain the same vertical resolution in the other domains and at the same time fulfil the mesh quality standards.



The resulting speed-up fraction and the normalized turbulent kinetic energy values at a 10m<sup>1</sup> high above ground level (a.g.l) along the domain as well as the profiles of both magnitudes on the top of the hill for each of the 4 domains tested are displayed in Fig.3 and 4.

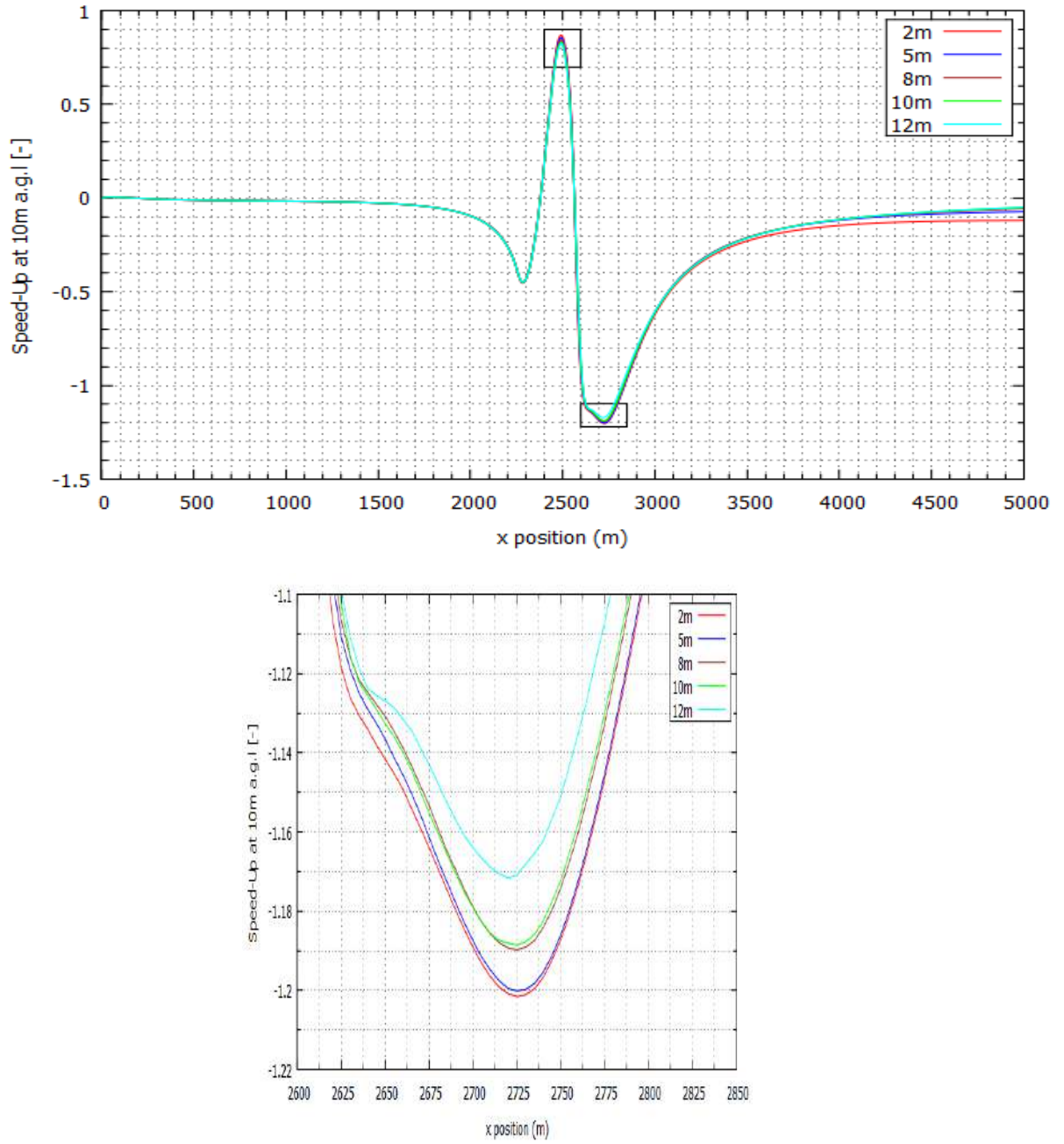


Figure 3: Speed-Up at 10m a.g.l for different horizontal resolutions.

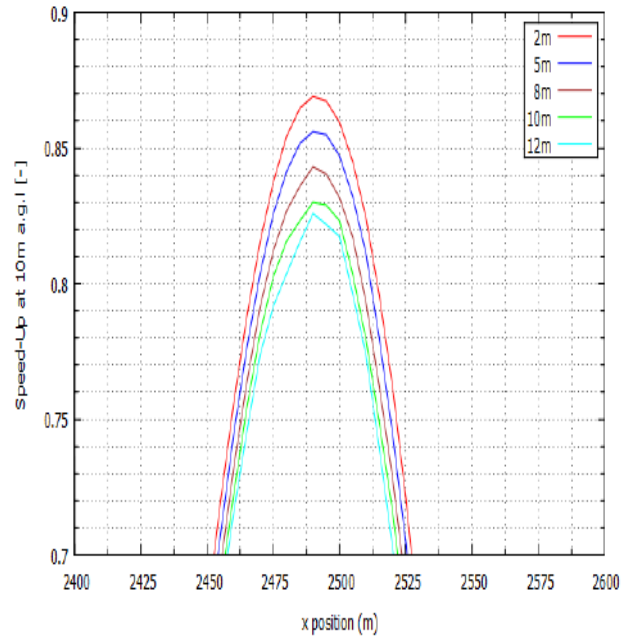


Figure 4: Detail of the Speed-Up at 10m a.g.l for different resolution.

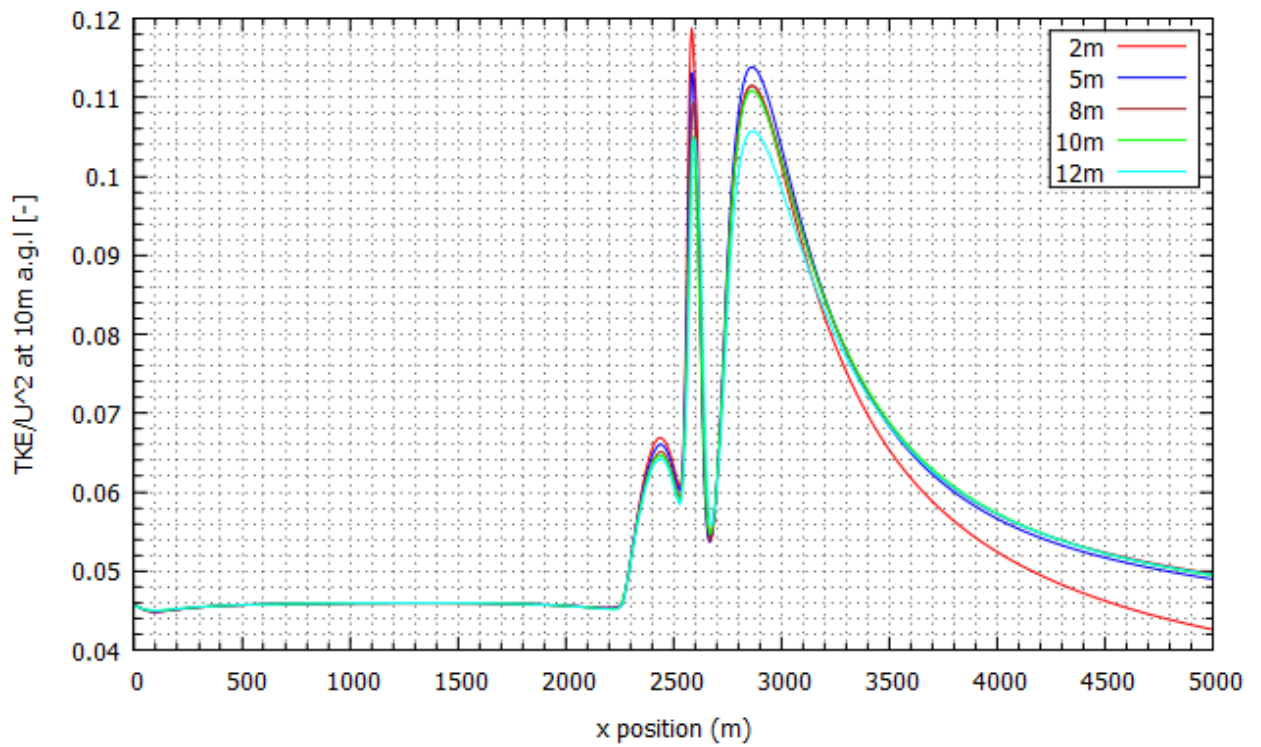


Figure 5: Normalized TKE at 10m a.g.l for different resolutions.

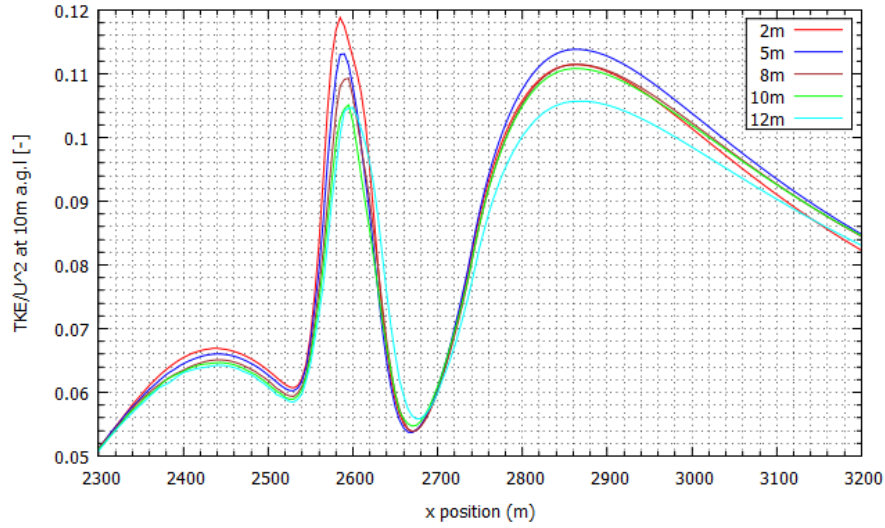


Figure 6: Normalized TKE at 10m a.g.l for different resolutions (detail).

In Fig. 3 and 4 it can be appreciated that the general shape of the speed-up at 10m a.g.l is similar for all the resolution values. They all reach their maximum Speed-Up at the hill top and their minimum at the downwind foot of the hill with negative values around -1, which may indicate the presence of a reverse flow and possibly a recirculation area. At the hill top the relative difference between the results of the domains with the highest (5) and the lowest (1) resolution is around 5.4% whereas the same magnitude is 2.6% at the hill foot on the lee side (see Table 3). As for the normalized TKE (Fig. 5 and 6), the maximum and minimum values are reached in every case on the lee side of the hill, the first near the top and the second near the base. The relative difference between case 1 and case 5 results at those two points, are ca.11.8% and ca.3.7% respectively (see Table 3). However, these differences disappear a few meters above the surfaces, as it can be observed in Fig. 7.

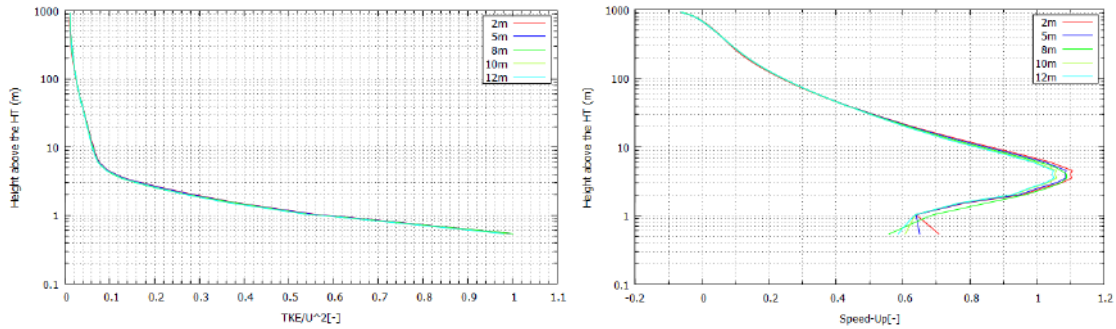


Figure 7: Comparison of the a) Speed-Up and b) Normalized TKE above the Hill.

Compared cases	Speed-Up	TKE/ $U_i^2$
<b>1-2.</b>	3.18%	0.69%
<b>1-3.</b>	5.91%	0.72%
<b>1-4.</b>	11.96%	1.26%
<b>1-5.</b>	27.84%	3.91%
<b>1-2.</b>	3.18%	6.93%
<b>2-3.</b>	2.12%	2.65%
<b>3-4.</b>	3.91%	6.36%
<b>4-5.</b>	13.13%	3.28%

Table 3: Averaged Speed-Up and Normalized TKE differences.

In Table 3, it can be observed that, the averaged relative differences among the results yielded by the different domains are larger for the Speed-Up fraction than for the normalized TKE, and in both cases, increase with the resolution difference between the domains considered. However, it cannot be concluded that they are approaching grid independency, for the relative differences between consecutive domains don't get smaller as the grid resolution increases.

Finally, the simulation times increases proportionally to the number of cells of the domain, as expected.

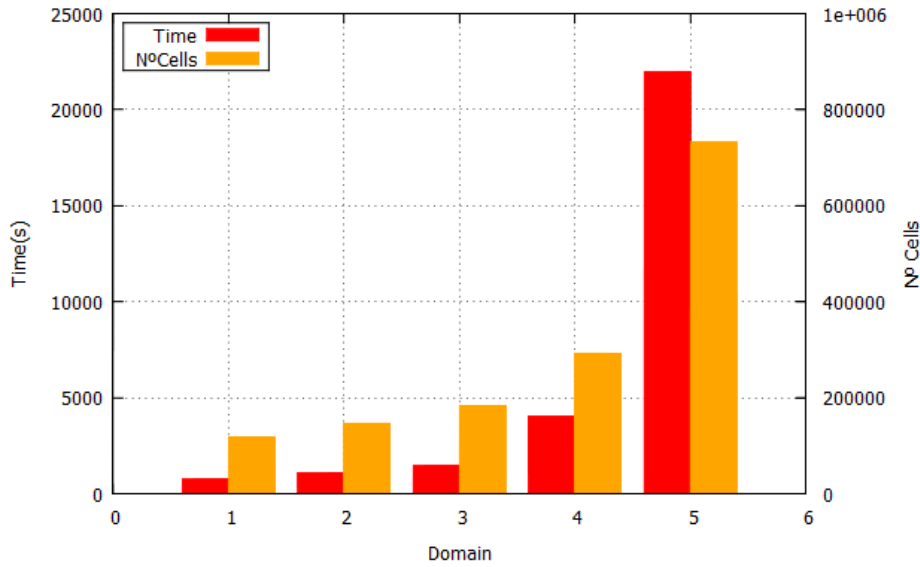


Figure 8: Simulation Time vs. Number of Cells.

## Discretization Schemes

Of the basic discretization schemes described in [Ch.3](#), only the first order upwind (FOU), the second-order upwind (SOU) and the QUICK schemes were tested, for they were the most commonly found in literature. As for the Hybrid and Power-Law Schemes, apparently OF doesn't offer the possibility to use them [\[69\]](#). Besides, according to [\[93\]](#) they might lead to incorrect results when used for oblique and recirculating flows, which are likely to appear on complex terrain.

Nine different simulations were carried out using different discretization schemes for the convective terms of the governing equations (see Table 4). The central differencing scheme (CDS) was applied for the diffusive terms in every case. Cases 8 and 9 crashed after a few iterations due to *time step continuity errors*, and in cases 5 and 6 convergence wasn't achieved for  $U_y$  and  $U_z$ . Domain features and simulations setting were those described in Table 1. A horizontal resolution of 8m was applied.

	Test	1	2	3	4	5	6	7	8	9
Convective Terms	U	FOU	SOU	SOU	SOU	QUICK	QUICK	FOU	FOU	QUICK
	k, $\epsilon$	FOU	SOU	QUICK	FOU	QUICK	FOU	QUICK	SOU	SOU

Table 4: Discretization schemes test cases.

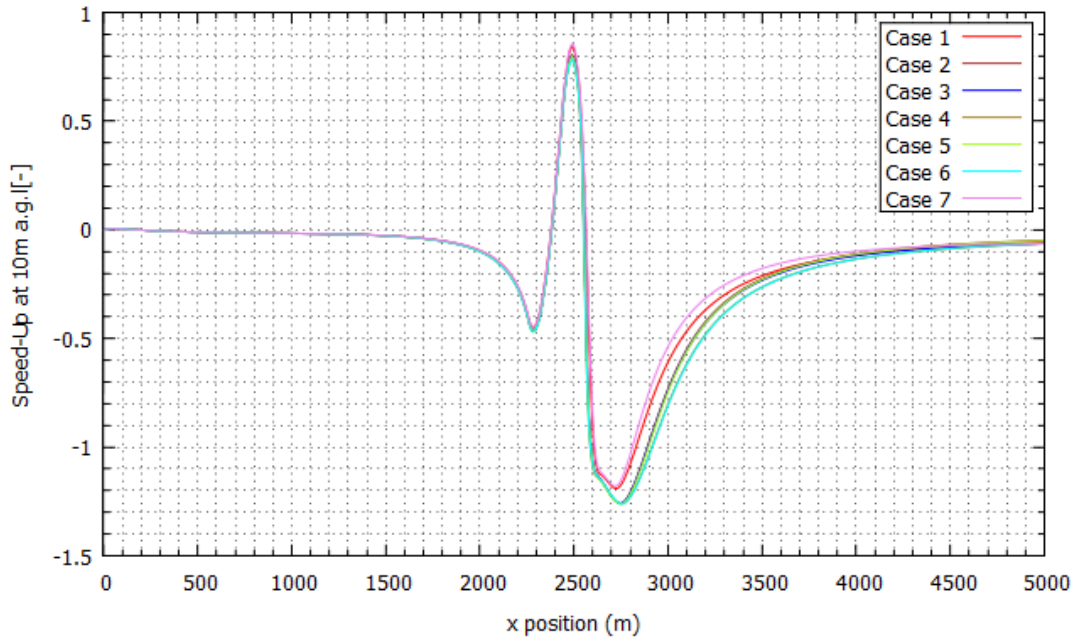


Figure 9: Speed-Up at 10m a.g.l for different discretization schemes.



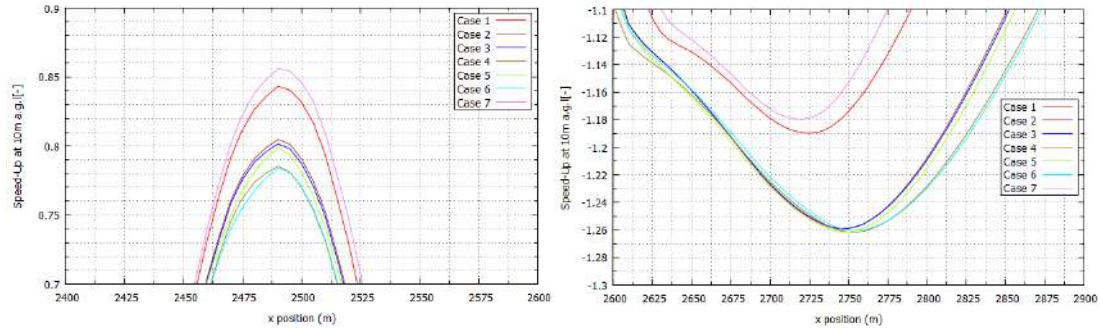


Figure 10: Speed-Up at 10m a.g.l for different discretization schemes at a) the hill top and at b) the hill foot on the lee-side.

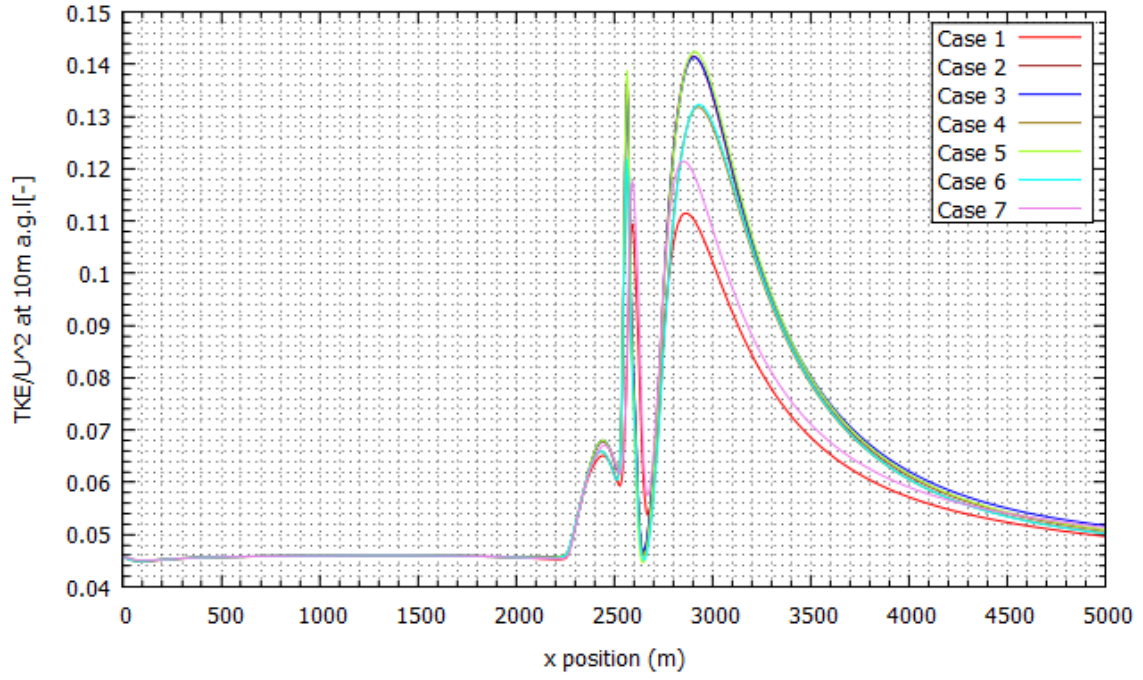


Figure 11: Normalized TKE at 10m a.g.l for different discretization schemes.

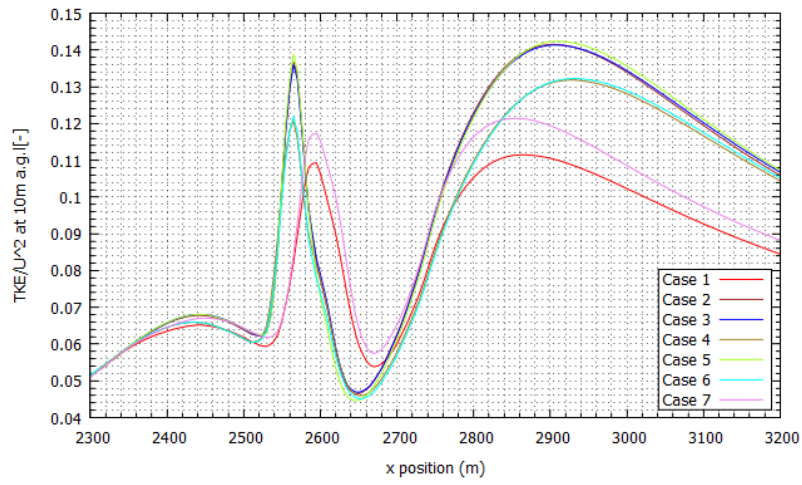


Figure 12: Normalized TKE at 10m a.g.l for different discretization schemes (detail).

In each case the maximum and minimum Speed-Up values are reached at the hill top and at the hill foot on the lee side respectively (see Fig.9 and 11). However, when the FOU scheme is used for the velocity terms (1 and 7), both, the predicted maximum and minimum Speed-Up values are clearly higher than in the other cases (Fig.10). Moreover, in cases 1 and 7 the Speed-Up minimum is shifted around 25m to the left compared with the others. This may be indicative of a smaller recirculation region, which would eventually lead to lower TKE values. The difference between cases 1 and 7 and the other cases is even higher for the normalized TKE. Not only their values are lower, but their maximums and minimums don't take place at the same position. This makes sense if the above-mentioned smaller recirculation region is considered. Cases using a FOU scheme for  $\varepsilon$  and  $k$  (1 and 4), predict lower normalized TKE values (Fig.11) than cases with second order schemes for all the terms (2,3 and 5). This might be explained by the well-known fact that the FOU scheme doesn't performance very well for high Reynolds numbers [66].

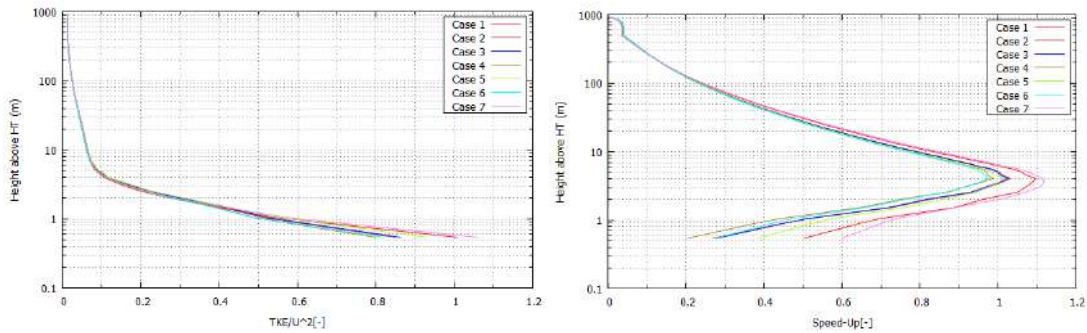


Figure 13: a) Speed-Up and b) Normalized TKE profiles above the Hill.

As it can be observed in Fig.13, the differences among the cases don't propagate upwards and at 10 a.g.l the are almost negligible.

The use of different discretization schemes also results into different simulation times (see Fig.14). Case 1, FOU scheme for all terms, shows the lowest value whereas Cases 5 and 6, both with QUICK scheme for velocity terms show the highest. Cases 3 and 7, which use the QUICK scheme for the  $k$  and  $\varepsilon$  terms, show higher simulation times than those which exclusively use FOU or SOU for all their terms. It may be necessary to remind that convergence wasn't achieved for  $U_x$  and  $U_z$  in Cases 5 and 6. This may be related to the lack of stability of QUICK schemes pointed out in [66]. In those cases, the time represented in Fig.14 is the time at which the residuals had already been relatively stable for at least 1000 iterations.

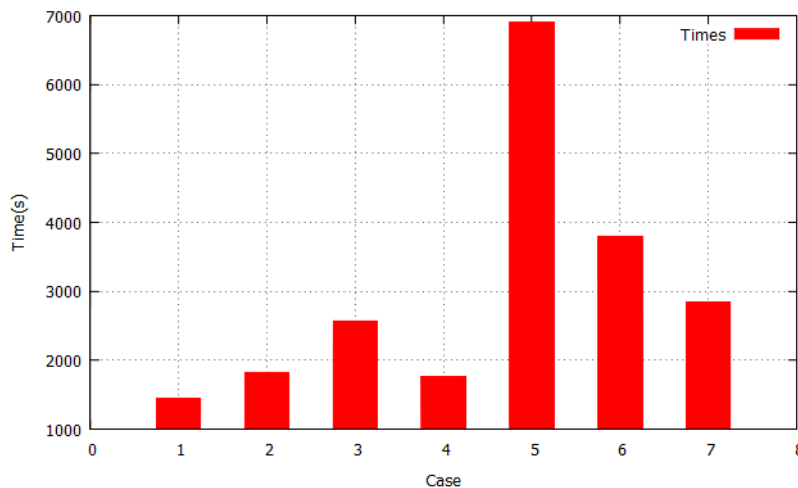


Figure 14: Convergence times for different discretization schemes.



# Annex 2: Canopy Tests

## Mesh Construction

Here are presented the *blockMeshDict* file used to create the domain for the canopy simulations and the *topoSetDict* file in which the canopy volume was defined:

### blockMeshDict:

```
/*-----*- C++ -*-----*\
| =====|
| \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O p e r a t i o n | Version: 2.3.0 |
| \ \ / A n d | Web: www.OpenFOAM.org |
| \ \ / M a n i p u l a t i o n |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    object       blockMeshDict;
}
// * * * * *

convertToMeters 1;

vertices
(
    ( 0 0 0)
    ( 300 0 0)
    ( 300 100 0)
    ( 0 100 0)
    ( 0 0 500)
    ( 300 0 500)
    ( 300 100 500)
    ( 0 100 500)//7

    ( 300 0 0)
    ( 1300 0 0)
    ( 1300 100 0)
    ( 300 100 0)
    ( 300 0 500)
    ( 1300 0 500)
    ( 1300 100 500)
    ( 300 100 500)//15

    ( 1300 0 0)
    ( 2000 0 0)
    ( 2000 100 0)
    ( 1300 100 0)
    ( 1300 0 500)
    ( 2000 0 500)
    ( 2000 100 500)
    ( 1300 100 500)//23
);

blocks
"(
    hex (0 1 2 3 4 5 6 7) (69 5 71) simpleGrading (0.35714 1 1666.66)
```

```

//Start cell width=0.03 m; z-dir Cell-to-cell expansion ratio=1.076
//Width of the end cell=50m
//X-dir. Start cell 7m end cell 0.25m. exp.ratio=1/1.015
hex (8 9 10 11 12 13 14 15) (48 5 71) simpleGrading (1 1 1666.66)
//Start cell width=0.03m; Cell-to-cell expansion ratio=1.076
//Width of the end cell=50m
hex (16 17 18 19 20 21 22 23) (83 5 71) simpleGrading (8 1 1666.66)
//Start cell width=0.03 m; Cell-to-cell expansion ratio=1.076
//Width of the end cell=50m
//X-dir. Start cell 7m end cell 0.25m. exp.ratio=1/1.015
);

edges
(
);

patches
(
    patch outlet3
    (
        (2 1 5 6)
    )
    patch outlet2
    (
        (9 10 14 13)
    )
    patch outlet
    (
        (17 18 22 21)
    )
    patch sides
    (
        (2 3 7 6)
        (0 1 5 4)
        (18 19 23 22)
        (16 17 21 20)
        (8 9 13 12)
        (10 11 15 14)
    )
    patch inlet
    (
        (3 0 4 7)
    )
    patch inlet2
    (
        (8 12 15 11)
    )
    patch inlet3
    (
        (16 20 23 19)
    )
    wall ground
    (
        (1 0 3 2)
    )

```

```

        (17 16 19 18)
    )

    symmetryPlane ground2

    (
        (9 8 11 10)
    )

    patch top

    (
        (7 6 5 4)
        (23 22 21 20)
        (15 14 13 12)
    )

);

mergePatchPairs
(
    (outlet3 inlet2)
    (outlet2 inlet3)
);

// ***** //

```

### topoSetDict:

```

/*-----*- C++ -*-----*\
| =====|
| \\ / F ield | OpenFOAM: The Open Source CFD Toolbox |
| \\ / O peration | Version: 2.3.0 |
| \\ / A nd | Web: www.OpenFOAM.org |
| \\ M anipulation |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    object       topoSetDict;
}

// * * * * * //

actions
{
    {
        name      porosity1;
        type      cellSet;
        action     new;
        source     boxToCell;
        sourceInfo
        {
            box (300 0 0) (1300 100 1.5);
        }
    }
    {
        name      porosity2;
        type      cellSet;
    }
}

```

```

        action new;
        source boxToCell;
        sourceInfo
        {
            box (300 0 1.5) (1300 100 5.25);
        }
    }
    {
        name    porosity3;
        type    cellSet;
        action new;
        source boxToCell;
        sourceInfo
        {
            box (300 0 5.25) (1300 100 7.5);
        }
    }
    {
        name    porosity1;
        type    cellZoneSet;
        action new;
        source setToCellZone;
        sourceInfo
        {
            set porosity1;
        }
    }
    {
        name    porosity2;
        type    cellZoneSet;
        action new;
        source setToCellZone;
        sourceInfo
        {
            set porosity2;
        }
    }
    {
        name    porosity3;
        type    cellZoneSet;
        action new;
        source setToCellZone;
        sourceInfo
        {
            set porosity3;
        }
    }
);

// *****

```

## Canopy Properties

Here is presented the model 3 *porosityProperties* file in which the values for the darcy-forchheimer parameters were defined according to table 5.6 (see [5.2.1](#)):

```

/*-----*- C++ -*-----*/
| ===== |
| \ \      / F ield      | OpenFOAM: The Open Source CFD Toolbox |
| \ \      / O peration  | Version: 2.3.0 |
| \ \      / A nd        | Web: www.OpenFOAM.org |
| \ \      / M anipulation | |
/*-----*- C++ -*-----*/

FoamFile
{
    version      2.0;
    format        ascii;
    class         dictionary;
    location      "constant";
    object        porosityProperties;
}
// *****

porosity1
{
    type          DarcyForchheimer;
    active        yes;
    cellZone      porosity1;

    DarcyForchheimerCoeffs
    {
        // the first two terms of the vectors are the values of d and f in the e1 and
        // directions.
        //The third term accounts for the value of d and f in the direction perpendicular
        // to e1 and e2.
        //In an isotropic media all three components should have the same value.
        //Negative coefficients have no meaning. They can be used to specify anysotropic
        // conditions.
        d      d [0 -2 0 0 0 0] (2356.51 2356.51 2356.51);
        f      f [0 -1 0 0 0 0] (0.01 0.01 0.01);

        coordinateSystem
        {
            type      cartesian;
            origin     (0 0 0);
            coordinateRotation
            {
                type      axesRotation;
                e1         (-1 0 0);
                e2         (0 0 1);
            }
        }
    }
}

porosity2
{
    type          DarcyForchheimer;
    active        yes;
    cellZone      porosity2;

    DarcyForchheimerCoeffs

```

```

{
    d    d [0 -2 0 0 0 0 0] (2356.51 2356.51 2356.51);
    f    f [0 -1 0 0 0 0 0] (0.2 0.2 0.2);

    coordinateSystem
    {
        type    cartesian;
        origin  (0 0 0);
        coordinateRotation
        {
            type    axesRotation;
            e1      (1 0 0);
            e2      (0 1 0);
        }
    }
}
porosity3
{
    type            DarcyForchheimer;
    active          yes;
    cellZone        porosity3;

    DarcyForchheimerCoeffs
    {
        d    d [0 -2 0 0 0 0 0] (2356.51 2356.51 2356.51);
        f    f [0 -1 0 0 0 0 0] (0.02 0.02 0.02);

        coordinateSystem
        {
            type    cartesian;
            origin  (0 0 0);
            coordinateRotation
            {
                type    axesRotation;
                e1      (1 0 0);
                e2      (0 1 0);
            }
        }
    }
}
// ***** //

```

## Further Canopy Simulations

According to the results presented in 5.2.1, the  $\alpha$ - and  $C_D$ -based canopy models seemed to provide better results than the other models tested. For this reason, it was decided to carry out further tests changing these parameters. A first test was made using Model 3 described in table 5.6 with a new leaf density profile, referred as Alfa 2, (Fig.1) which better fits the one proposed in [83].

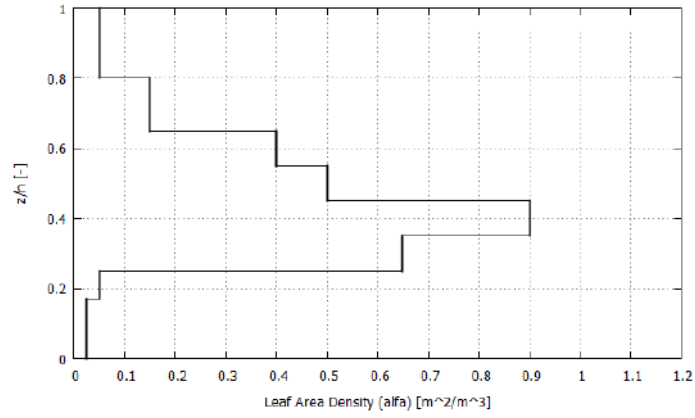


Figure 1: Leaf Density Profile.

In Fig.2 and 3, the profiles for the normalized wind speed and the turbulent intensity obtained at the four mast positions are displayed.

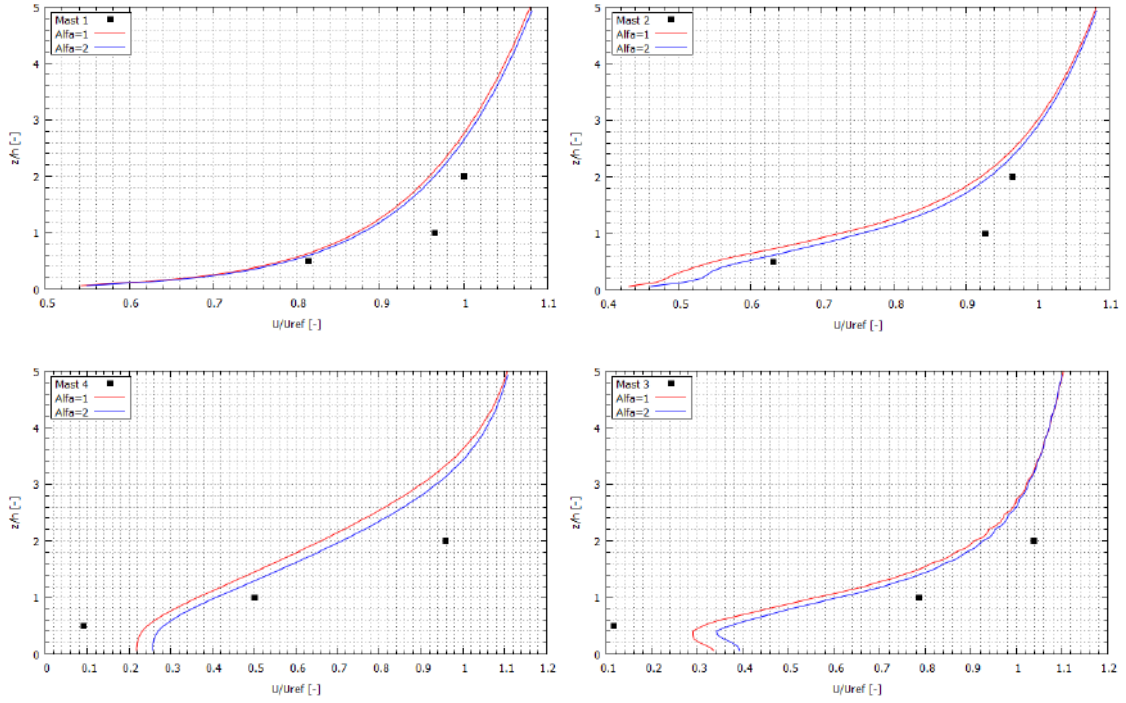


Figure 2: Normalized Velocity profiles with different  $\alpha$ .



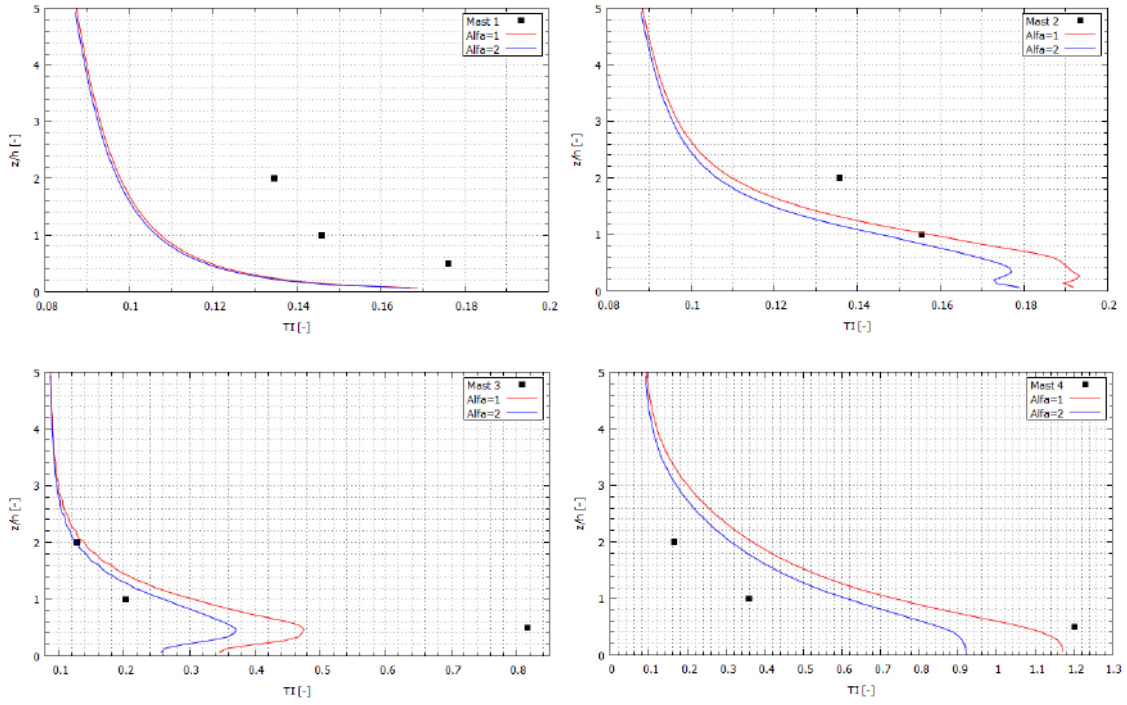


Figure 3: Turbulent Intensity profiles with different  $\alpha$ .

Although no great difference can be observed between the results of the simulations with the two different  $\alpha$  profiles, those of Alfa 2 seem to better adjust to the shape of the experimental data inside the forest at  $z/h > 1$ . However, at  $z/h < 1$ , wind speed and turbulent intensity values are still overpredicted.

A second series of simulations were made using the original  $\alpha$ -profile and applying three different values for the drag coefficient.

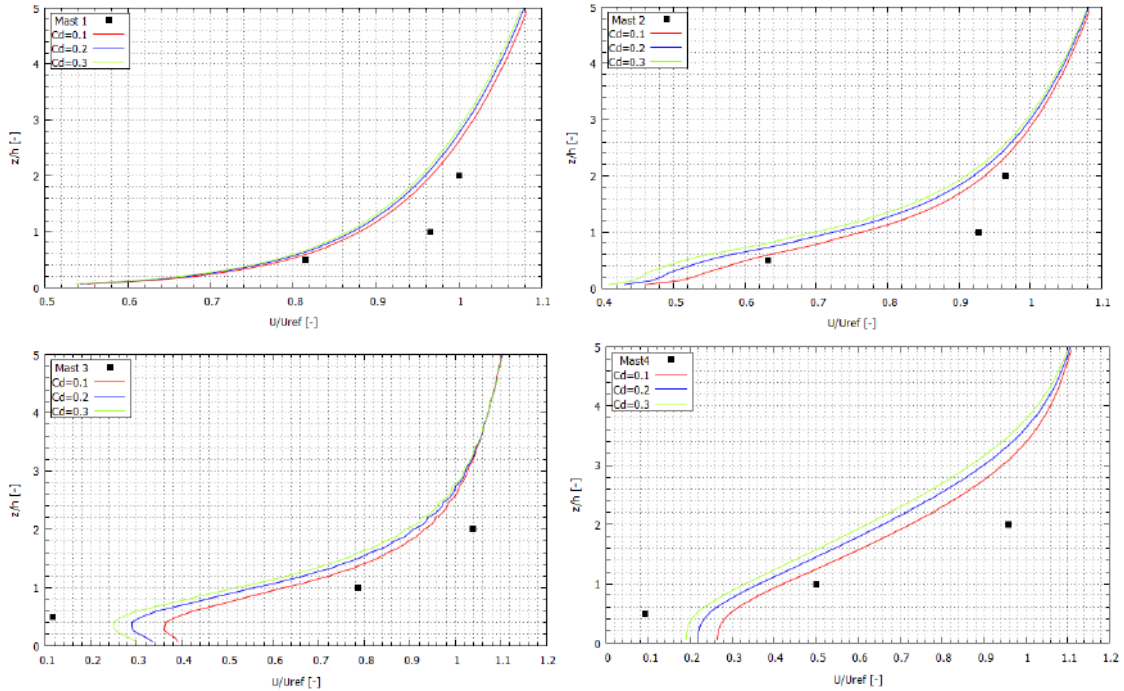


Figure 3: Normalized Velocity profiles with different Drag Coefficient.

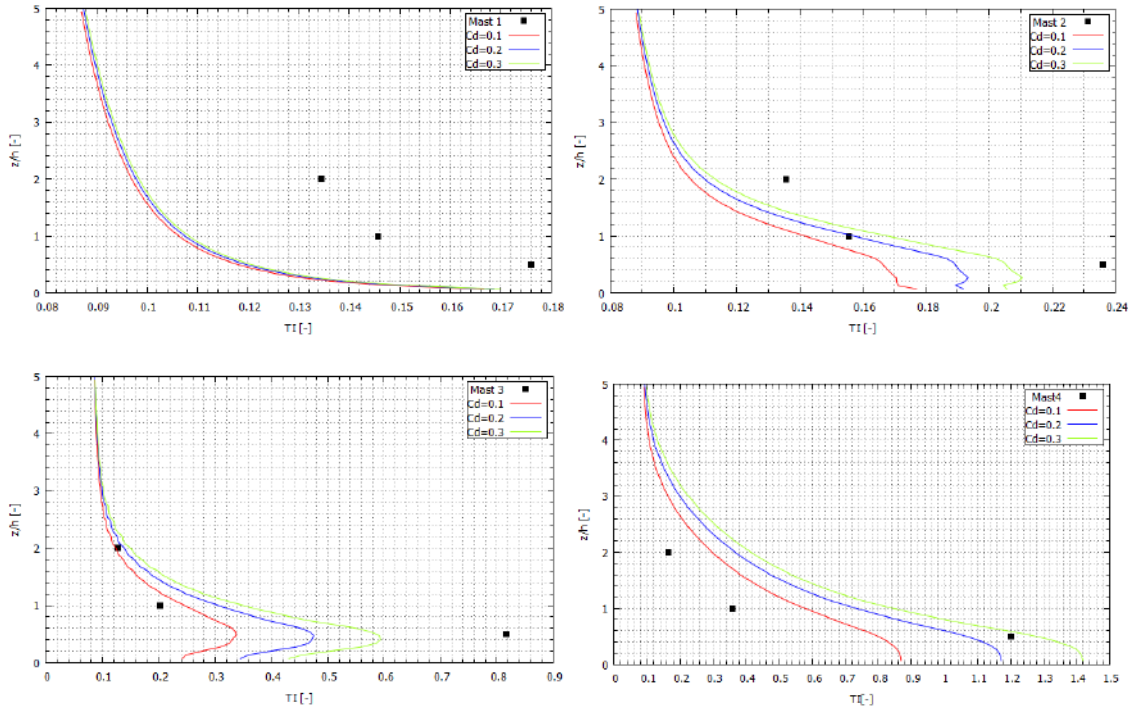


Figure 4: Turbulent Intensity profiles with different Drag Coefficient.

As expected, simulations with a higher drag coefficient show lower wind velocities and higher turbulent intensities, both out and inside the canopy and above and below  $h$ . The agreement between the simulations results and the experimental data seems to be better for  $C_d=0.1$  at every mast at  $z/h > 1$  whereas it underpredicts air flow deceleration and turbulent intensity inside the canopy, where the the  $C_D=0.3$  presents the best results. These results may suggest that a non-uniform drag coefficient could be the most appropriate input for the model.

Finally, in 5.2.1 the turbulent intensity of the experimental data was calculated assuming a value for  $\sigma_v/u^*$  of 2.1, following the indications in [83]. However, as in [83] no clear justification for this assumption is provided, it was decided to carry out further tests, changing  $\sigma_v/u^*$  value. The results are presented in Fig.5,6 y 7.

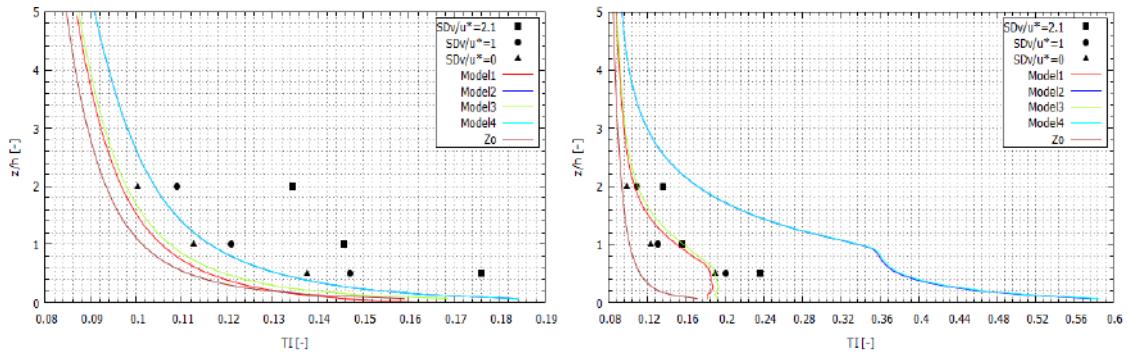


Figure 5: Turbulent Intensity profiles with different  $\sigma_v/u^*$  at a) mast 1 and b) mast 2.

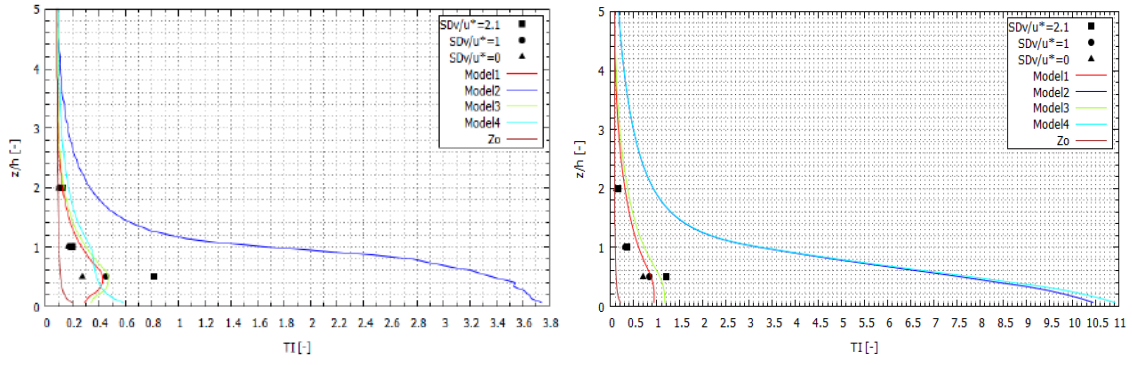


Figure 6: Turbulent Intensity profiles with different  $\sigma_v/u^*$  at a) mast 3 and b) mast 4.

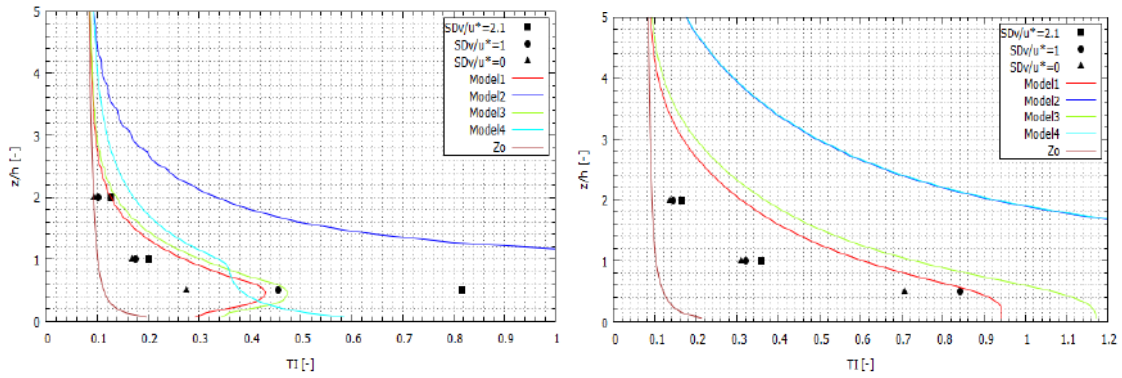


Figure 7: Detail of the Turbulent Intensity profiles with different  $\sigma_v/u^*$  at a) mast 3 and b) mast 4.

As it can be observed in the graphics above, Model 3 turbulent intensity profil fits the experimental data better as the  $\sigma_v/u^*$  value decrease. Although these results don't provide a conclusive proof about whether the  $\sigma_v/u^*$  assumed value was right or wrong, they show that further reasearch must be conducted on this topic.

# Annex 3: Validation

## Grid Independency Analysis

As it was explained in 6.2.3, before simulating a case, a grid sensitivity analysis was carried out using Case 1 input boundary conditions and using 3 different grids with an increasing horizontal resolution in the first 30m above the ground. In Table 1, the resolution, total number of cells of the meshes used along with the simulation times are presented:

Resolution(m)	Time(s)	N°Cells
4	12114	1066011
3	13954	1828279
2	57281	4131970

Table 1: Grid independency tests settings.

In Fig.1, at the reference mast hardly no difference can be appreciated between velocity and turbulence intensity profiles in the three different grids.

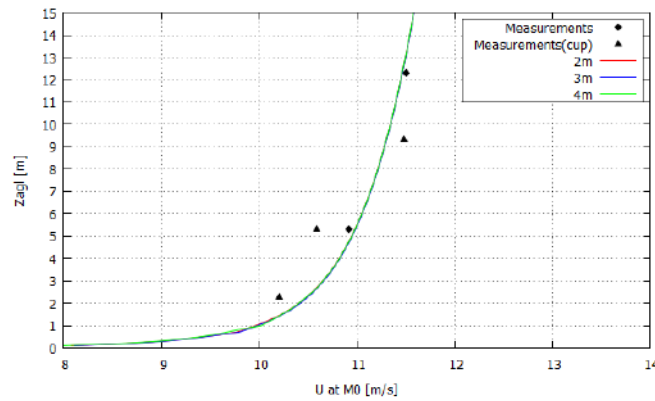


Figure 1: U at M0.

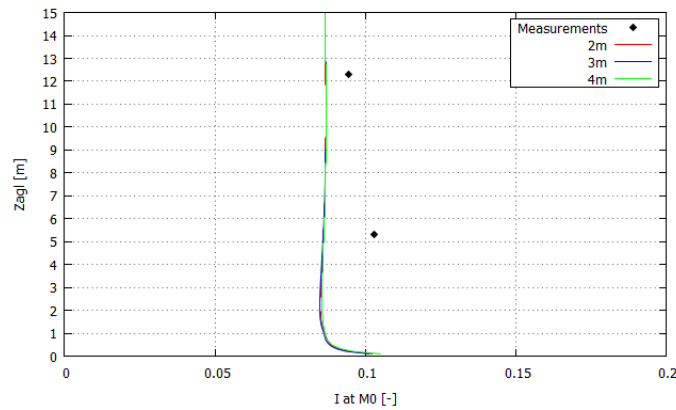


Figure 2: TI at M0.

However, that is not the case when it comes to the other masts, as it can be seen in Fig.3 to 8.

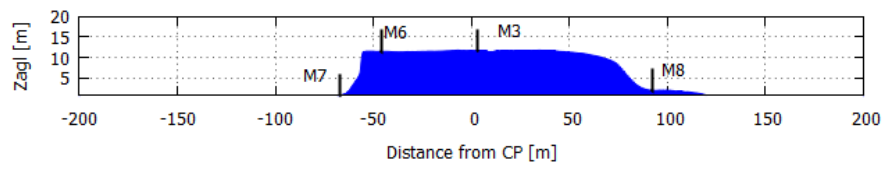


Figure 3: Bolund Hill elevation profile along Line B.

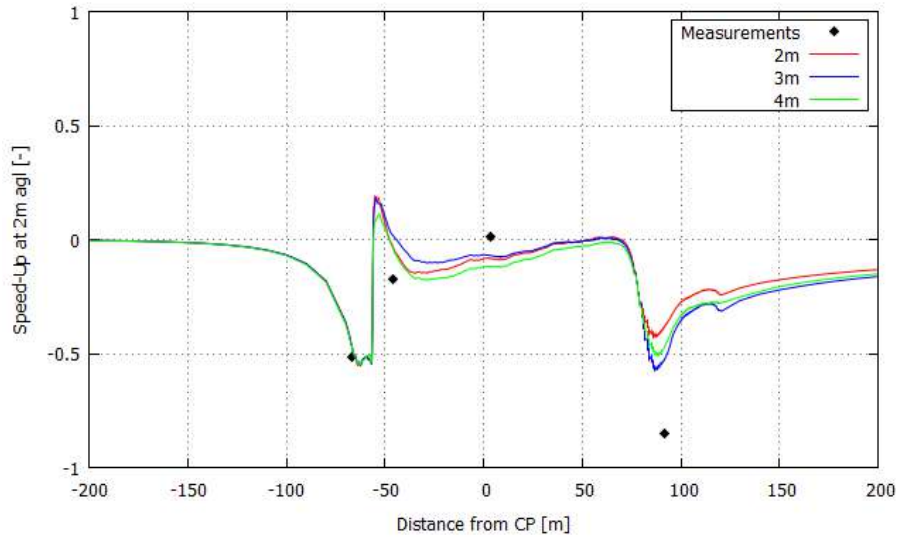


Figure 4: Speed-up over the Bolund Hill along Line B at 2m a.g.l.

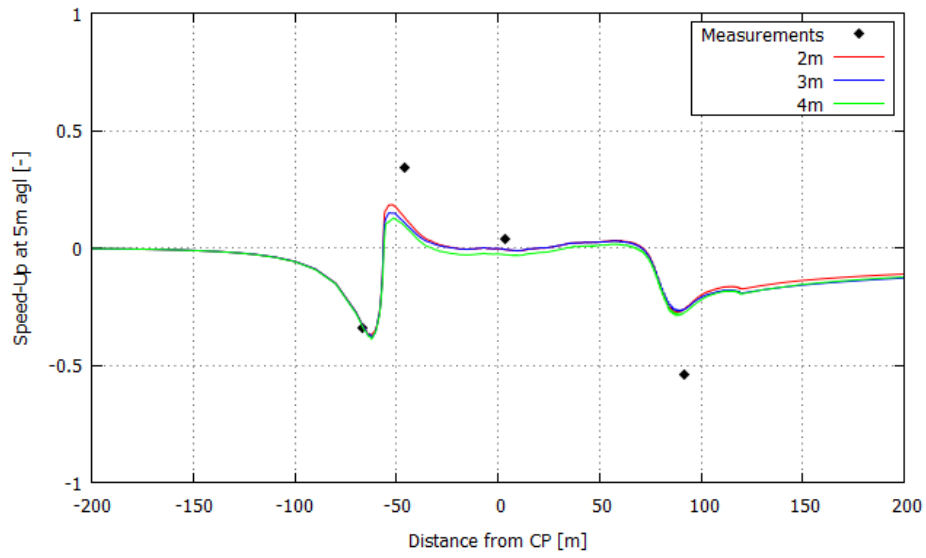


Figure 5: Speed-up over the Bolund Hill along Line B at 5m a.g.l.

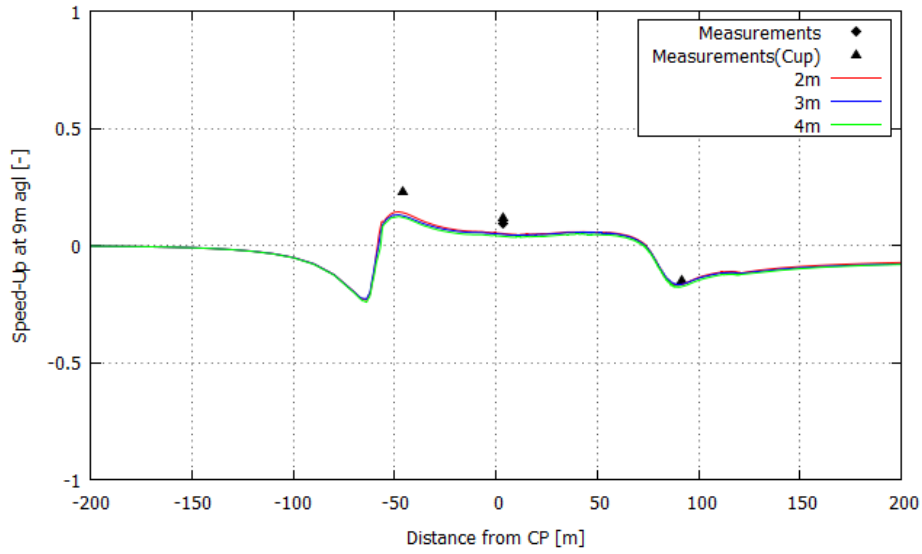


Figure 6: Speed-up over the Bolund Hill along Line B at 9m a.g.l.

As expected, no difference among the profiles can be appreciated before the wind flow meets the hill and the differences between the three different meshes decrease with the height, so that at 9m a.g.l the three Speed-up profiles can hardly be distinguished one from another. This may be explained by the fact that, as the turbulent phenomena takes place near the ground, it is there where a different resolution may be able to capture them or not. As the height above ground increases, the turbulent phenomena tend to disappear, and the mesh resolution has less and less influence. Although it is not clear, it seems that at 2m a.g.l (Fig.4) the 3m resolution results are closer to the measured values.

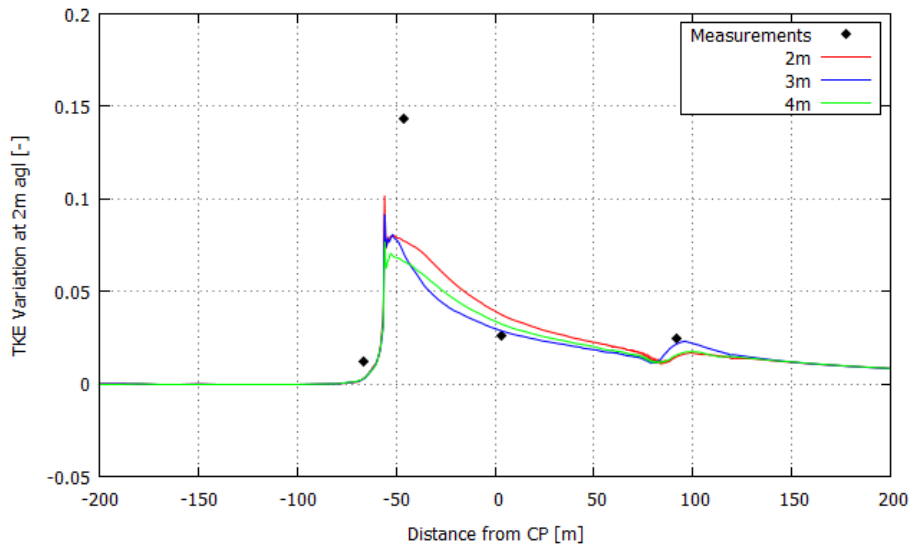


Figure 7: TKE Variation over the Bolund Hill along Line B at 2 m agl.

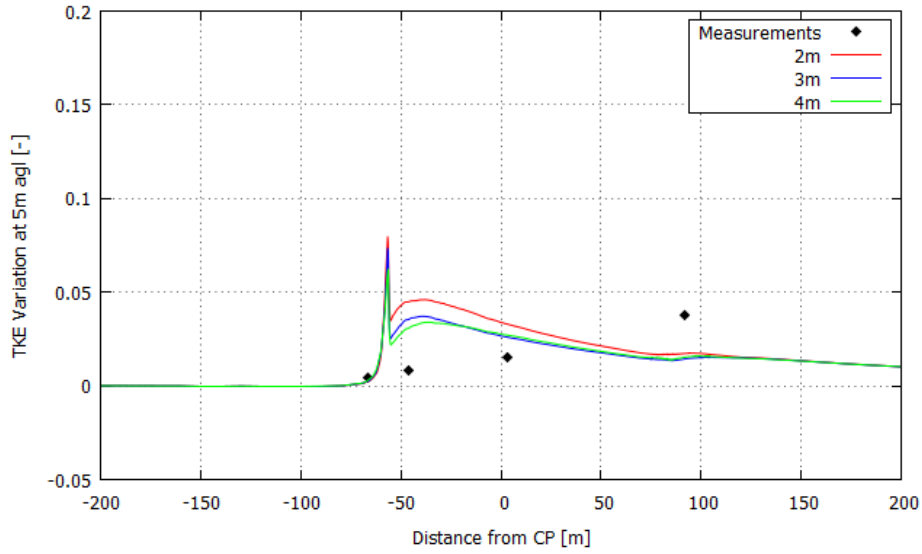


Figure 8: TKE Variation over the Bolund Hill along Line B at 5 m agl.

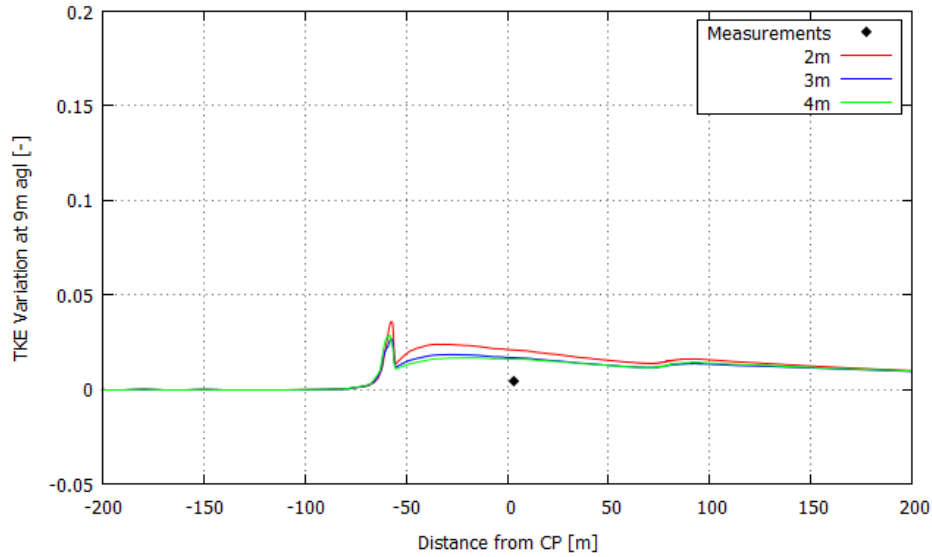


Figure 9: TKE Variation over the Bolund Hill along Line B at 9 m agl.

As Speed-up profiles, TKE variation profiles become increasingly similar with height, although, in this case, the 2m resolution profile keeps always a distance to the other two, even at 9m a.g.l. Although no one really adjust to the measured values, the mesh with a 3m resolution provided the best results at 2m a.g.l.

Finally, in Fig. 4 to 9 it can be observed that the profiles don't appear to follow any tendency or approach a certain value at each position as the resolution increases. This may indicate either, that grid independency has already been achieved and therefore any further resolution increase brings no improvement, or that none of the three grids is able to capture the turbulent phenomena that arise on the hill surface and further test with higher resolution values must be carried out.



## Mesh construction

The meshing of the domain was made using cfMesh. Here the corresponding *meshDict* file is presented:

```
/*-----*- C++ -*-----*\
| =====
| \\      / F ield      | cfMesh: A library for mesh generation
| \\      / O peration  |
| \\      / A nd        | Author: Franjo Juretic
| \\      / M anipulation | E-mail: franjo.juretic@c-fields.com
\*-----*/
FoamFile
{
    version    2.0;
    format     ascii;
    class      dictionary;
    location   "system";
    object     meshDict;
}

// * * * * *

surfaceFile "Bolund1LandNoSurroundings.stl";

maxCellSize 24000.0;

//Anisotropic meshing.
anisotropicSources
{
    boxSource
    {
        type box;
        centre (98000 132000 300000);
        lengthX 1000000;
        lengthY 500000;
        lengthZ 400000;

        //scaling factors in each direction. 0.5 means that the top cells are 2 times bigger
        than the bottom cells.

        scaleX 1;
        scaleY 1;
        scaleZ 2;
    }
    boxSource2
    {
        type box;
        centre (98000 132000 0);
        lengthX 1000000;
        lengthY 500000;
        lengthZ 122000;

        //scaling factors in each direction. 0.5 means that the top cells are 2 times bigger
        than the bottom cells.

        scaleX 1;
        scaleY 1;
        scaleZ 0.3333;
    }
}
```

```

}
}
objectRefinements
{
tail3
{
additionalRefinementLevels 3;
centre ( 145000 125000 0 );
lengthX 230000;
lengthY 160000;
lengthZ 62000;
type box;
}
tail2
{
additionalRefinementLevels 1;
centre (98000 132000 40000);
lengthX 1000000;
lengthY 500000;
lengthZ 40000;
type box;
}

}
localRefinement
{

    ground
    {

        additionalRefinementLevels 3;
        refinementThickness 50000;

    }

}
boundaryLayers
{
optimiseLayer 1;
untangleLayers 1;

optimisationParameters
{
nSmoothNormals 5;
maxNumIterations 10;
//Ratio between the maximum layer thickness and the estimated feature size.
//featureSizeFactor 0.9999;
//reCalculateNormals 1;
//relThicknessTol 0.1;
}
patchBoundaryLayers
{
Land
{
nLayers 5;
thicknessRatio 1.1;
maxFirstLayerThickness 150;
////active 1 or inactive 0.
allowDiscontinuity 1;
}
ground
{
nLayers 11;

```

```

thicknessRatio 1.2;
maxFirstLayerThickness 54;
////active 1 or inactive 0.
//allowDiscontinuity 1;
}
}
}
renameBoundary
{
    defaultName          fixedWalls;
    defaultType          wall;

    newPatchNames
    {
        inlet
        {
            newName      inlet;
            newType       patch;
        }

        outlet
        {
            newName      outlet;
            newType       patch;
        }
        ground
        {
            newName      ground;
            newType       wall;
        }
        Land
        {
            newName      Land;
            newType       wall;
        }
        sides
        {
            newName      sides;
            newType       patch;
        }
        top
        {
            newName      top;
            newType       patch;
        }
    }
}
}
// *****

```

## Boundary conditions

Here the files describing the boundary conditions for the validation simulation are presented:

### Velocity ( $U$ ):

```
/*-----*- C++ -*-----*\
| =====|
| \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O p e r a t i o n | Version: 2.3.0 |
| \ \ / A n d | Web: www.OpenFOAM.org |
| \ \ / M a n i p u l a t i o n |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        volVectorField;
    location     "0";
    object       U;
}
// * * * * *

#include        "include/initialConditions"

dimensions      [0 1 -1 0 0 0 0];

internalField   uniform $flowVelocity;

boundaryField
{
    #include "include/ABLConditions"

    outlet
    {
        type      zeroGradient;
    }

    inlet
    {
        type      atmBoundaryLayerInletVelocity;
        Uref      $Uref;
        Href      $Href;
        n         $windDirection;
        z         $zDirection;
        z0        $z0;
        value      $internalField;
        zGround    $zGround;
    }
    Surroundings
    {
        type      uniformFixedValue;
        uniformValue (0 0 0);
        value      uniform (0 0 0);
    }
    Land
    {
        type      uniformFixedValue;
        uniformValue (0 0 0);
        value      uniform (0 0 0);
    }
}
```

```

ground
{
    type            uniformFixedValue;
    uniformValue    (0 0 0);
    value           uniform (0 0 0);
}
top
{
    type            uniformFixedValue;
    uniformValue    (13.9769 0 0);
    value           uniform (0 0 0);
}

#include "include/sideAndTopPatches"
}
// *****

```

### Turbulent Kinetic Energy ( $k$ )

```

/*-----*- C++ -*-----*\
| =====|
| \\      / F ield      | OpenFOAM: The Open Source CFD Toolbox |
| \\      / O peration  | Version: 2.3.0                        |
| \\      / A nd        | Web:      www.OpenFOAM.org             |
| \\      / M anipulation|                                     |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        volScalarField;
    object       k;
}
// *****

#include      "include/initialConditions"

dimensions   [0 2 -2 0 0 0 0];

internalField   uniform $turbulentKE;

boundaryField
{
    #include "include/ABLConditions"

    outlet
    {
        type            zeroGradient;
    }
    Surroundings
    {
        type            kqRWallFunction;
        value           uniform 0.0;
    }
    inlet
    {
        type            uniformFixedValue;
        uniformValue    constant $turbulentKE;
    }
}

```

```

    ground
    {
        type            kqRWallFunction;
        value            uniform 0.0;
    }
    Land
    {
        type            kqRWallFunction;
        value            uniform 0.0;
    }
    top
    {
        type            uniformFixedValue;
        uniformValue     constant $turbulentKE;
    }
    #include "include/sideAndTopPatches"
}
// *****

```

### Turbulent Kinetic Energy dissipation rate/Epsilon ( $\epsilon$ ):

```

/*-----*- C++ -*-----*\
|=====|
|  \ \   /  F ield      | OpenFOAM: The Open Source CFD Toolbox
|  \ \   /  O peration   | Version:  2.3.0
|   \ \  /  A nd         | Web:      www.OpenFOAM.org
|   \ \  /  M anipulation |
\*-----*-*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        volScalarField;
    location     "0";
    object       epsilon;
}
// *****

dimensions      [0 2 -3 0 0 0 0];

#include         "include/initialConditions"

internalField    uniform $turbulentEpsilon;

boundaryField
{
    #include "include/ABLConditions"

    ground
    {
        type            epsilonWallFunction;
        Cmu              0.03;
        kappa            0.4;
        E                9.8;
        value            $internalField;
    }
    Surroundings
    {
        type            epsilonWallFunction;

```

```

        Cmu          0.03;
        kappa        0.4;
        E            9.8;
        value        $internalField;
    }
    Land
    {
        type          epsilonWallFunction;
        Cmu           0.03;
        kappa         0.4;
        E             9.8;
        value         $internalField;
    }

    outlet
    {
        type          zeroGradient;
    }

    inlet
    {
        type          atmBoundaryLayerInletEpsilon;
        Uref           $Uref;
        Href           $Href;
        z              $zDirection;
        z0             $z0;
        value          $internalField;
        zGround        $zGround;
    }

    top
    {
        type          uniformFixedValue;
        uniformValue   constant 0.0003;
    }
    #include "include/sideAndTopPatches"
}

// *****

```

### Turbulent Viscosity (*nut*):

```

/*-----*- C++ -*-----*\
| =====|
|  \ \    /  F ield      | OpenFOAM: The Open Source CFD Toolbox |
|  \ \    /  O peration  | Version: 2.3.0                        |
|   \ \  /   A nd        | Web:      www.OpenFOAM.org           |
|   \ \ /    M anipulation|                                     |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        volScalarField;
    location     "0";
    object       nut;
}
// *****

```



```

dimensions      [0 2 -1 0 0 0 0];

internalField    uniform 0;

boundaryField
{
    #include "include/ABLConditions"

    inlet
    {
        type          calculated;
        value          uniform 0;
    }

    outlet
    {
        type          zeroGradient;
    }

    ground
    {
        type          nutkAtmRoughWallFunction;
        z0            uniform 0.0003;
        value          uniform 0;
    }

    Surroundings
    {
        type          nutkAtmRoughWallFunction;
        z0            uniform 0.0003;
        value          uniform 0;
    }

    Land
    {
        type          nutkAtmRoughWallFunction;
        z0            uniform 0.015;
        value          uniform 0;
    }

    top
    {
        type          uniformFixedValue;
        uniformValue    constant 81.9590;
    }

    #include "include/sideAndTopPatches"
}

```

```
// ***** //
```

### Pressure ( $p$ ):

```

/*-----*- C++ -*-----*\
| =====|
| \\      / F ield      | OpenFOAM: The Open Source CFD Toolbox |
| \\      / O peration  | Version: 2.3.0 |
| \\      / A nd        | Web: www.OpenFOAM.org |
| \\      / M anipulation|
\*-----*/

```

```

FoamFile
{
    version      2.0;
    format       ascii;
    class        volScalarField;
    object       p;
}
// * * * * *

#include        "include/initialConditions"

dimensions      [0 2 -2 0 0 0 0];

internalField    uniform $pressure;

boundaryField
{
    inlet
    {
        type      zeroGradient;
    }

    outlet
    {
        type      uniformFixedValue;
        uniformValue    constant $pressure;
    }
    ground
    {
        type      zeroGradient;
    }
    Land
    {
        type      zeroGradient;
    }

    Surroundings
    {
        type      zeroGradient;
    }
    top
    {
        type      slip;
    }

    #include "include/sideAndTopPatches"
}

// *****

```

#### **Include (ABLConditions):**

```

Uref            13.9769;
Href            500000;
z0              uniform 0.0003;
turbulentKE     0.928;
windDirection   (1 0 0);
zDirection      (0 0 1);
zGround         uniform 0.0;

```

## Finite Volume Method dictionaries

Here the files describing the parameter and options affecting the application of the FVM are displayed:

### fvSchemes:

```
/*-----*- C++ -*-----*\
| ===== |
| \\\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\\ / O p e r a t i o n | Version: 2.3.0 |
| \\\ / A n d | Web: www.OpenFOAM.org |
| \\\ M a n i p u l a t i o n |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    object       fvSchemes;
}
// * * * * *

ddtSchemes
{
    default      steadyState;
}

gradSchemes
{
    default      cellLimited Gauss linear 1;
}

divSchemes
{
    default      none;
    div(phi,U)   bounded Gauss linearUpwind grad(U);
    div((nuEff*dev(T(grad(U)))) Gauss linear ;
    div(phi,epsilon) bounded Gauss linearUpwind grad(epsilon);
    div(phi,k)   bounded Gauss linearUpwind grad(k);
}

laplacianSchemes
{
    default      Gauss linear limited corrected 0.333;
}

interpolationSchemes
{
    default      linear;
}

snGradSchemes
{
    default      limited corrected 0.333;
}
```

```

fluxRequired
{
    default          no;
    p;
}

// *****

```

## fvSolutions:

```

/*-----*- C++ -*-----*\
| =====|
| \\      / F ield      | OpenFOAM: The Open Source CFD Toolbox|
| \\      / O peration  | Version: 2.1.x|
| \\      / A nd        | Web: www.OpenFOAM.org|
| \\      / M anipulation|
\*-----*\
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    object       fvSolution;
}
// *****
solvers
{
    p
    {
        solver      GAMG;
        tolerance    1e-7;
        relTol       0.1;
        smoother     GaussSeidel;
        nPreSweeps    0;
        nPostSweeps   2;
        cacheAgglomeration on;
        agglomerator  faceAreaPair;
        nCellsInCoarsestLevel 10;
        mergeLevels   1;
    }

    U
    {
        solver      smoothSolver;
        smoother     GaussSeidel;
        tolerance    1e-8;
        relTol       0.1;
        nSweeps       1;
    }

    k
    {
        solver      smoothSolver;
        smoother     GaussSeidel;
        tolerance    1e-8;
        relTol       0.1;
        nSweeps       1;
    }

    epsilon

```

```

        {
            solver            smoothSolver;
            smoother          GaussSeidel;
            tolerance         1e-8;
            relTol            0.1;
            nSweeps           1;
        }
    }
SIMPLE
{
    nNonOrthogonalCorrectors 0;
    residualControl
    {
        p                1e-3;
        U                1e-4;
        "(k|epsilon)"    1e-4;
    }
}
relaxationFactors
{
    fields
    {
        p                0.3;
    }
    equations
    {
        U                0.7;
        k                0.7;
        epsilon          0.7;
    }
}
potentialFlow
{
    nNonOrthogonalCorrectors 10;
}
cache
{
    grad(U);
}
// *****

```

### controlDict:

```

/*-----*- C++ -*-----*\
| ===== |
| \ \ / F ield | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O peration | Version: 2.1.x |
| \ \ / A nd | Web: www.OpenFOAM.org |
| \ \ / M anipulation |
\*-----*\
FoamFile
{
    version    2.0;
    format     ascii;
    class      dictionary;
    object     controlDict;
}
// *****
libs          ("libincompressibleRASModels.so");
libs          ("libgroovyBC.so");

```

```

application      porousSimpleFoam;

startFrom        latestTime;

startTime        40;

stopAt           endTime;

endTime          12100;

deltaT           1;

writeControl      timeStep;

writeInterval     20;

purgeWrite        0;

writeFormat       binary;

writePrecision    12;

writeCompression  uncompressed;

timeFormat        general;

timePrecision     6;

runTimeModifiable yes;

```

```
// ***** //
```

## Validation Errors

In Table 2 the differences between normalised speed-up and turbulent intensity predicted and measured values at masts M3, M6, M7 and M8, are presented as errors calculated according to Eq.6.6 and 6.7 (see 6.1.3). Two additional columns containing the TKE difference ( $R_{TKE}$ ) and Turbulent Intensity ( $R_I^*$ ) errors calculated adapting Eq.6.6 are included.

M3											
<i>Zagl</i>	<i>ΔSm</i>	<i>ΔSs</i>	<i>Rs (%)</i>	<i>I0m</i>	<i>ΔIm</i>	<i>ΔIs</i>	<i>R<sub>I</sub> (%)</i>	<i>R<sub>I</sub><sup>*</sup> (%)</i>	<i>Δkm</i>	<i>Δks</i>	<i>R<sub>TKE</sub> (%)</i>
2000,0000	0,0102	-0,155	<b>-16,560</b>	0,1141	0,083	0,105	<b>18,651</b>	<b>2,129</b>	0,0260	0,022	<b>-0,401</b>
5000,0000	0,0354	-0,063	<b>-9,822</b>	0,1028	0,057	0,097	<b>38,317</b>	<b>3,941</b>	0,0150	0,022	<b>0,706</b>
9000,0000	0,0932	0,005	<b>-8,784</b>	0,097	0,022	0,069	<b>48,836</b>	<b>4,722</b>	0,0046	0,015	<b>1,029</b>
9000,000	0,117	0,005	-	-	-	-	-	-	-	-	-
M6											
<i>Zagl</i>	<i>ΔSm</i>	<i>ΔSs</i>	<i>Rs (%)</i>	<i>I0m</i>	<i>ΔIm</i>	<i>ΔIs</i>	<i>R<sub>I</sub> (%)</i>	<i>R<sub>I</sub><sup>*</sup> (%)</i>	<i>Δkm</i>	<i>Δks</i>	<i>R<sub>TKE</sub> (%)</i>
1900,000	-0,172	-0,081	<b>9,041</b>	0,1148	0,281	0,195	<b>-74,099</b>	<b>-8,510</b>	0,1431	0,057	<b>-8,633</b>
4900,000	0,339	0,042	<b>29,693</b>	0,1031	0,033	0,123	<b>87,252</b>	<b>8,993</b>	0,0079	0,031	<b>2,334</b>
8900,000	0,233	0,082	<b>15,046</b>	-	-	-	-	-	-	-	-
M7											
<i>Zagl</i>	<i>ΔSm</i>	<i>ΔSs</i>	<i>Rs (%)</i>	<i>I0m</i>	<i>ΔIm</i>	<i>ΔIs</i>	<i>R<sub>I</sub> (%)</i>	<i>R<sub>I</sub><sup>*</sup> (%)</i>	<i>Δkm</i>	<i>Δks</i>	<i>R<sub>TKE</sub> (%)</i>
2050,000	-0,516	-0,470	<b>4,626</b>	0,114	0,043	0,015	<b>-25,020</b>	<b>-2,847</b>	0,012	0,003	<b>-0,900</b>
5050,000	-0,340	-0,328	<b>1,146</b>	0,103	0,018	0,013	<b>-5,377</b>	<b>-0,552</b>	0,004	0,002	<b>-0,177</b>
M8											
<i>Zagl</i>	<i>ΔSm</i>	<i>ΔSs</i>	<i>Rs (%)</i>	<i>I0m</i>	<i>ΔIm</i>	<i>ΔIs</i>	<i>R<sub>I</sub> (%)</i>	<i>R<sub>I</sub><sup>*</sup> (%)</i>	<i>Δkm</i>	<i>Δks</i>	<i>R<sub>TKE</sub> (%)</i>
1800,000	-0,853	-0,563	<b>28,928</b>	0,116	0,079	0,2893	<b>182,235</b>	<b>21,067</b>	0,024	0,020	<b>-0,410</b>
4700,000	-0,539	-0,277	<b>26,264</b>	0,104	0,117	0,2626	<b>140,786</b>	<b>14,576</b>	0,038	0,014	<b>-2,398</b>
8800,000	-0,150	-0,175	<b>-2,521</b>	-	-	-	-	-	-	-	-

Table 2: Validation case errors.

# Annex 4: Study Case (Sierra Morena)

## Mesh construction

Here the *blockMeshDict* and *snnapyHexMeshDict* files used to create the mesh for one of the two domains of the Study Case are presented:

### BlockMeshDict:

```
/*-----*- C++ -*-----*\
| =====|
| \ \ / F ield | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O peration | Version: 2.1.x |
| \ \ / A nd | Web: www.OpenFOAM.org |
| \ \ / M anipulation |
\*-----*\
FoamFile
{
    version      2.0;
    format        ascii;
    class         dictionary;
    object        blockMeshDict;
}
// * * * * * //
convertToMeters 1;

vertices
(
    ( 5895178 3223590 900000 )
    ( 7895178 3223590 900000 )
    ( 7895178 5223590 900000 )
    ( 5895178 5223590 900000 )

    ( 5895178 3223590 2000000 )
    ( 7895178 3223590 2000000 )
    ( 7895178 5223590 2000000 )
    ( 5895178 5223590 2000000 )

    ( 5895178 3223590 900000 )
    ( 7895178 3223590 900000 )
    ( 7895178 5223590 900000 )
    ( 5895178 5223590 900000 )

    ( 5895178 3223590 1550000 )
    ( 7895178 3223590 1550000 )
    ( 7895178 5223590 1550000 )
    ( 5895178 5223590 1550000 )

);

blocks
(
    hex ( 0 1 2 3 4 5 6 7 ) (100 100 50) simpleGrading (1 1 1)
    //first cell width=1000mm; last cell width= 97.01723m
    hex ( 8 9 10 11 12 13 14 15 ) (200 200 100) simpleGrading (1 1 1)
    //first cell width=1000mm; last cell width= 97.01723m
);

edges
(
);

patches
(
    patch top
    (
```



```

    (4 5 6 7)
  )
  wall ground
  (
    (0 3 2 1)
  )
  patch inlet
  (
    (0 4 7 3)
  )
  patch outlet
  (
    (1 2 6 5)
  )
  patch sides
  (
    (2 3 7 6)
    ( 0 1 5 4)
  )
);
mergePatchPairs
(
);

// *****

```

### **SnappyHexMeshDict:**

```

/*-----*- C++ -*-----*\
| =====|
| \ \ / F ield | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O peration | Version: 2.1.x |
| \ \ / A nd | Web: www.OpenFOAM.org |
| \ \ / M anipulation |
\*-----*\
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    object       snappyHexMeshDict;
}

// * * * * *

// Which of the steps to run
castellatedMesh true;
snap           true;
addLayers      true;

geometry
{
    ground1-30.stl
    {
        type triSurfaceMesh;
        name ground1-30;
    }
};

// Settings for the castellatedMesh generation.
castellatedMeshControls
{
    // Refinement parameters
    maxLocalCells 50000000;
    maxGlobalCells 50000000;
    minRefinementCells 0;
    maxLoadUnbalance 0.10;.
}

```

```

nCellsBetweenLevels 2;

// Surface based refinement

refinementSurfaces
{
    ground1-30
    {
        //Surface-wise min and max refinement level. Level means that the smallest cells will be
        //1/(2^level) the size of the normal ones.
        level (2 2);
    }
}

resolveFeatureAngle 30;

// Mesh selection
// ~~~~~~
locationInMesh (6895180 4223590 1650000);
allowFreeStandingZoneFaces false;
}

// Settings for the snapping.

snapControls
{
    //- Number of patch smoothing iterations before finding correspondence
    //- to surface
    nSmoothPatch 5;

    //- Relative distance for points to be attracted by surface feature point
    //- or edge. True distance is this factor times local
    //- maximum edge length.
    tolerance 10.0;

    //- Number of mesh displacement relaxation iterations.
    nSolveIter 100;

    //- Maximum number of snapping relaxation iterations. Should stop
    //- before upon reaching a correct mesh.
    nRelaxIter 6;
}

// Settings for the layer addition.
addLayersControls
{
    relativeSizes false;
    {
        ground1-30
        {
            nSurfaceLayers 1;
        }
    }
    expansionRatio 1.0;
    finalLayerThickness 1000;
    minThickness 1000;
    nGrow 1;
    featureAngle 30;
    nRelaxIter 3;
    nSmoothSurfaceNormals 1;
    nSmoothNormals 3;
    nSmoothThickness 10;
    maxFaceThicknessRatio 1.0;
    maxThicknessToMedialRatio 1.0;
    minMedianAxisAngle 130;
    nBufferCellsNoExtrude 0;
}

```

```

        nLayerIter 50;
        nRelaxedIter 20;
    }
    meshQualityControls
    {
        maxNonOrtho 65;
        maxBoundarySkewness 20;
        maxInternalSkewness 4;
        maxConcave 80;
        minVol 1E-13;
        minTetQuality 1e-30;
        minArea -1;
        minTwist 0.05;
        minDeterminant 0.001;
        minFaceWeight 0.05;
        minVolRatio 0.01;
        minTriangleTwist -1;
        minVolCollapseRatio 0.5;
        nSmoothScale 4;
        errorReduction 0.75;

        relaxed
        {
            maxNonOrtho 75;
        }
    }
    debug 0;
    mergeTolerance 1e-6;

    // ***** //

```

## Boundary conditions

Here the files describing the boundary conditions for the Study Case simulation are presented:

### Velocity ( $U$ ):

```

/*-----* C++ *-----*\
| ===== |
| \ \      / F ield      | OpenFOAM: The Open Source CFD Toolbox |
| \ \      / O peration  | Version: 2.3.0 |
| \ \      / A nd        | Web: www.OpenFOAM.org |
| \ \      / M anipulation |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        volVectorField;
    location     "0";
    object       U;
}
// ***** //

#include "include/initialConditions"

dimensions      [0 1 -1 0 0 0];

internalField    uniform $flowVelocity;

boundaryField
{
    #include "include/ABLConditions"

```

```

outlet
{
type      zeroGradient;
//type      inletOutlet;
//inletValue      uniform (0 0 0);
//value      $internalField;

}

inlet
{
type groovyBC;
value $internalField;
variables "u_f=0.626;kf=0.41;yposition=pos().y;z0=(pos().y<3654600?
225:(pos().y>3918600? (pos().y<4141900? 225:(pos().y>4284200? (pos().y<4445300?
500:(pos().y>4515000? 225:150)):150)):150));";
valueExpression "(u_f/kf)*(log((pos().z-
profile(yposition)+z0)/z0))*vector(1,0,0)";

fields
(
yposition zGround
);
lookuptables (
{
name profile;
outOfBounds clamp;
fileName "$FOAM_CASE/constant/triSurface/InletLine.dat";
}
);
}

forest
{
type      uniformFixedValue;
uniformValue      (0 0 0);
value      uniform (0 0 0);

}
grass
{
type      uniformFixedValue;
uniformValue      (0 0 0);
value      uniform (0 0 0);
}
ground
{
type      uniformFixedValue;
uniformValue      (0 0 0);
value      uniform (0 0 0);
}
ground1-30
{
type      uniformFixedValue;
uniformValue      (0 0 0);
value      uniform (0 0 0);
}
urban
{
type      uniformFixedValue;
uniformValue      (0 0 0);

```

```

        value            uniform (0 0 0);
    }
    top
    {
        //type            slip;
        type              uniformFixedValue;
        uniformValue      (12.3811 0 0);
        value            uniform (0 0 0);
    }

#include "include/sideAndTopPatches"
}

// *****

```

### Turbulent Kinetic Energy ( $k$ ):

```

/*-----*- C++ -*-----*\
| =====|
| \ \      / F ield      | OpenFOAM: The Open Source CFD Toolbox |
| \ \      / O peration  | Version: 2.3.0 |
| \ \      / A nd        | Web: www.OpenFOAM.org |
| \ \      / M anipulation| |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        volScalarField;
    object       k;
}
// *****

#include      "include/initialConditions"

dimensions   [0 2 -2 0 0 0 0];

internalField  uniform $turbulentKE;

boundaryField
{
    #include "include/ABLConditions"

    outlet
    {
        type            zeroGradient;
        //type           inletOutlet;
        //inletValue      uniform $turbulentKE;
        //value           $internalField;
    }
    ground
    {
        type            kqRWallFunction;
        value           uniform 0.0;
    }
    ground1-30
}

```



```

internalField    uniform $turbulentEpsilon;

boundaryField
{
    #include "include/ABLConditions"

    ground1-30
    {
        type            epsilonWallFunction;
        Cmu              0.03;
        kappa            0.4;
        E                9.8;
        value            $internalField;
    }
    ground
    {
        type            epsilonWallFunction;
        Cmu              0.03;
        kappa            0.4;
        E                9.8;
        value            $internalField;
    }
    grass
    {
        type            epsilonWallFunction;
        Cmu              0.03;
        kappa            0.4;
        E                9.8;
        value            $internalField;
    }
    urban
    {
        type            epsilonWallFunction;
        Cmu              0.03;
        kappa            0.4;
        E                9.8;
        value            $internalField;
    }
    forest
    {
        type            epsilonWallFunction;
        Cmu              0.03;
        kappa            0.4;
        E                9.8;
        value            $internalField;
    }
    outlet
    {
        type            zeroGradient;
        //type          inletOutlet;
        //inletValue     uniform $turbulentEpsilon;
        //value          $internalField;
    }
    inlet
    {
        type groovyBC;
        //value $internalField;
    }
}

```



```

    ///gradientExpression "0";
    ///fractionExpression "(time())>=4100 && time()<5100)? 1:((phi>0)? 0:1)";
    value $internalField;
    variables "u_f=0.626;kf=0.41;yposition=pos().y;z0=(pos().y<3654600?
225:(pos().y>3918600? (pos().y<4141900? 225:(pos().y>4284200? (pos().y<4445300?
500:(pos().y>4515000? 225:150)):150)):150));";
    valueExpression "(pow(u_f,3)/(kf*(pos().z-profile(yposition)+z0)))";
    fields
    (
    yposition zGround
    );
    lookupTables (
    {
    name profile;
    outOfBounds clamp;
    fileName "$FOAM_CASE/constant/triSurface/InletLine.dat";
    }
    );
    }

    top
    {
    //type slip;

    //type fixedGradient;
    //gradient uniform ???;

    type uniformFixedValue;
    uniformValue constant 0.0012;
    }
    #include "include/sideAndTopPatches"
}

// ***** //

```

### Turbulent Viscosity (*nut*):

```

/*-----*- C++ -*-----*\
| =====|
| \ \ / F ield | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O peration | Version: 2.3.0 |
| \ \ / A nd | Web: www.OpenFOAM.org |
| \ \ / M anipulation |
\*-----*/
FoamFile
{
    version 2.0;
    format ascii;
    class volScalarField;
    location "0";
    object nut;
}
// ***** //

dimensions [0 2 -1 0 0 0 0];

internalField uniform 0;

boundaryField
{

```

```

#include "include/ABLConditions"

inlet
{
    type          calculated;
    value         uniform 0;
}

outlet
{
    type          zeroGradient;
    //type        calculated;
    //value       uniform 0;
}
ground
{
    type          nutkAtmRoughWallFunction;
    z0            uniform 0.150;
    value         uniform 0;
}
ground1-30
{
    type          nutkAtmRoughWallFunction;
    z0            uniform 0.150;
    value         uniform 0;
}
forest
{
    type          nutkAtmRoughWallFunction;
    z0            uniform 0.225;
    value         uniform 0;
}
grass
{
    type          nutkAtmRoughWallFunction;
    z0            uniform 0.05;
    value         uniform 0;
}
urban
{
    type          nutkAtmRoughWallFunction;
    z0            uniform 0.5;
    value         uniform 0;
}
top
{
    //type        slip;
    type          uniformFixedValue;
    uniformValue  constant 128.2651;
}

#include "include/sideAndTopPatches"
}

// ***** //

```

### Pressure ( $p$ ):

```
/*-----*- C++ -*-----*\
| =====|
| \\      / F ield      | OpenFOAM: The Open Source CFD Toolbox|
| \\      / O peration  | Version: 2.3.0|
| \\      / A nd        | Web: www.OpenFOAM.org|
| \\      / M anipulation|
|-----*\
FoamFile
{
    version      2.0;
    format        ascii;
    class         volScalarField;
    object        p;
}
// * * * * *

#include          "include/initialConditions"

dimensions        [0 2 -2 0 0 0 0];

internalField     uniform $pressure;

boundaryField
{
    inlet
    {
        type          zeroGradient;
    }

    outlet
    {
        type           uniformFixedValue;
        uniformValue    constant $pressure;
    }
    ground
    {
        type           zeroGradient;
    }
    ground1-30
    {
        type           zeroGradient;
    }
    forest
    {
        type           zeroGradient;
    }
    grass
    {
        type           zeroGradient;
    }
    urban
    {
        type           zeroGradient;
    }
    top
    {

```

```

        type            slip;
    }

#include "include/sideAndTopPatches"

}

// ***** //

Include (ABLConditions):

Uref            12.3811;
Href            500000;
z0              uniform 150;
turbulentKE     2.2611;
windDirection   (1 0 0);
zDirection      (0 0 1);
zGround         uniform 0.0;

```

## Annex 5: Wind Data

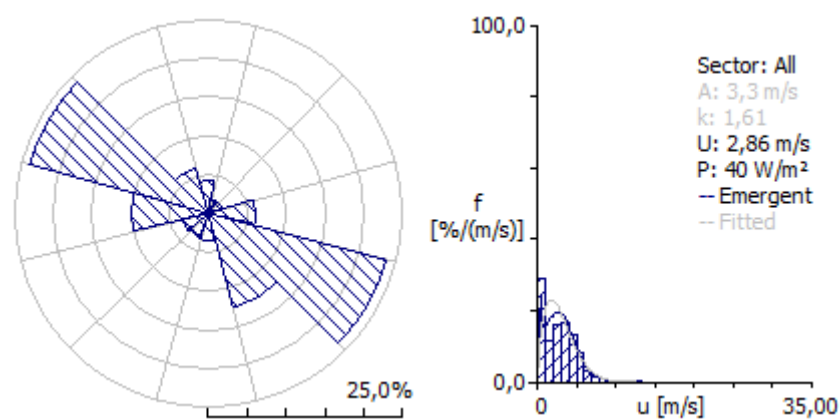
Here the results of the analysis of the wind data using OMWC are presented:

### 'SierraMorena' Observed Wind Climate

Produced on 19.11.2016 at 14:34:16 by licenced user: Unlicenced using WAsP version: 11.05.0028.

Site description: 'Untitled'; Position: 0,00°N 0,00°E; Anemometer height: 10,00 m a.g.l.

Parameter	Measured	Emergent	Discrepancy
Mean wind speed [m/s]	unknown	2,86	unknown
Mean power density [W/m <sup>2</sup> ]	unknown	40 W/m <sup>2</sup>	unknown



	0	30	60	90	120	150	180	210	240	270	300	330
A	0,4	0,5	0,6	3,2	4,0	3,8	0,8	0,8	0,9	2,9	3,8	4,1
k	0,45	10,34	1,52	1,83	2,09	1,46	0,92	1,14	1,08	1,91	2,59	2,46
U	0,95	0,49	0,58	2,82	3,52	3,44	0,83	0,78	0,85	2,57	3,40	3,63
P	89	0	0	29	49	71	3	1	2	21	37	47
f	4,2	1,8	1,8	6,1	23,5	12,5	3,4	3,2	3,5	10,0	24,0	6,0

U	0	30	60	90	120	150	180	210	240	270	300	330	All
1,0	807	1.000	991	320	115	254	759	779	715	239	99	302	290

2,0	67	0	3	75	118	140	151	146	192	164	111	16	113
3,0	26	0	0	178	193	127	63	64	75	248	213	69	160
4,0	18	0	2	186	218	118	18	10	14	185	237	176	166
5,0	16	0	3	130	170	111	6	1	3	96	203	224	134
6,0	8	0	0	72	98	101	0	0	0	52	104	140	79
7,0	10	0	0	29	51	56	1	0	1	13	28	58	33
8,0	9	0	0	7	20	39	1	0	0	1	5	12	13
9,0	6	0	0	2	11	23	1	0	0	0	0	2	6
10,0	7	0	0	1	5	16	0	0	0	0	0	1	3
11,0	5	0	0	0	2	8	0	0	0	0	0	0	2
12,0	4	0	0	0	0	5	0	0	0	0	0	0	1
13,0	3	0	0	0	0	2	0	0	0	0	0	1	0
14,0	3	0	0	0	0	0	0	0	0	0	0	0	0
15,0	0	0	0	0	0	1	0	0	0	0	0	0	0
16,0	1	0	0	0	0	0	0	0	0	0	0	0	0
17,0	3	0	0	0	0	0	0	0	0	0	0	0	0
18,0	2	0	0	0	0	0	0	0	0	0	0	0	0
19,0	0	0	0	0	0	0	0	0	0	0	0	0	0
20,0	1	0	0	0	0	0	0	0	0	0	0	0	0
21,0	1	0	0	0	0	0	0	0	0	0	0	0	0
22,0	0	0	0	0	0	0	0	0	0	0	0	0	0
23,0	1	0	0	0	0	0	0	0	0	0	0	0	0
24,0	1	0	0	0	0	0	0	0	0	0	0	0	0
25,0	0	0	0	0	0	0	0	0	0	0	0	0	0
26,0	1	0	0	0	0	0	0	0	0	0	0	0	0
27,0	0	0	0	0	0	0	0	0	0	0	0	0	0
28,0	0	0	0	0	0	0	0	0	0	0	0	0	0
29,0	0	0	0	0	0	0	0	0	0	0	0	0	0
30,0	0	0	0	0	0	0	0	0	0	0	0	0	0
31,0	0	0	0	0	0	0	0	0	0	0	0	0	0

32,0	1	0	0	0	0	0	0	0	0	0	0	0	0
------	---	---	---	---	---	---	---	---	---	---	---	---	---

#### All-sector statistics

	Weibull-A	Weibull-k	Mean speed	Power density
Source data	-	-	(not available from the file)	
Fitted	3,3 m/s	1,61	2,98 m/s	40 W/m <sup>2</sup>
Emergent	-	-	2,86 m/s	40 W/m <sup>2</sup>
Combined	3,2 m/s	1,48	2,86 m/s	40 W/m <sup>2</sup>

A and U are given in m/s, P in W/m<sup>2</sup> and the frequencies of occurrence in per mille and per cent (f).



## Annex 6: Topography data processing

Here the steps followed to process the topography data of the area surrounding Sierra Morena to use it for the wind flow simulations with OF are explained. In this process the programs *GoogleEarth* and *QGIS 2.18: Las Palmas*, a geographic information software (GIS) were used.

### Orography data processing

The orography data was obtained from the INEGI web page as a *.tif* file. The area comprised was bigger than that of the future domain. The resolution of the terrain file was 15m. This may be too low in terrain with abrupt height and slope variations [98].

Regardless the meshing tool used, *snappyHexMesh* or *cfMesh*, a *.stl* file of the terrain is required. Therefore, part of the information contained in the downloaded *.tif* has to be converted to a *.stl* format. The first step is opening the *.tif* as a raster layer in QGIS (Fig.1).

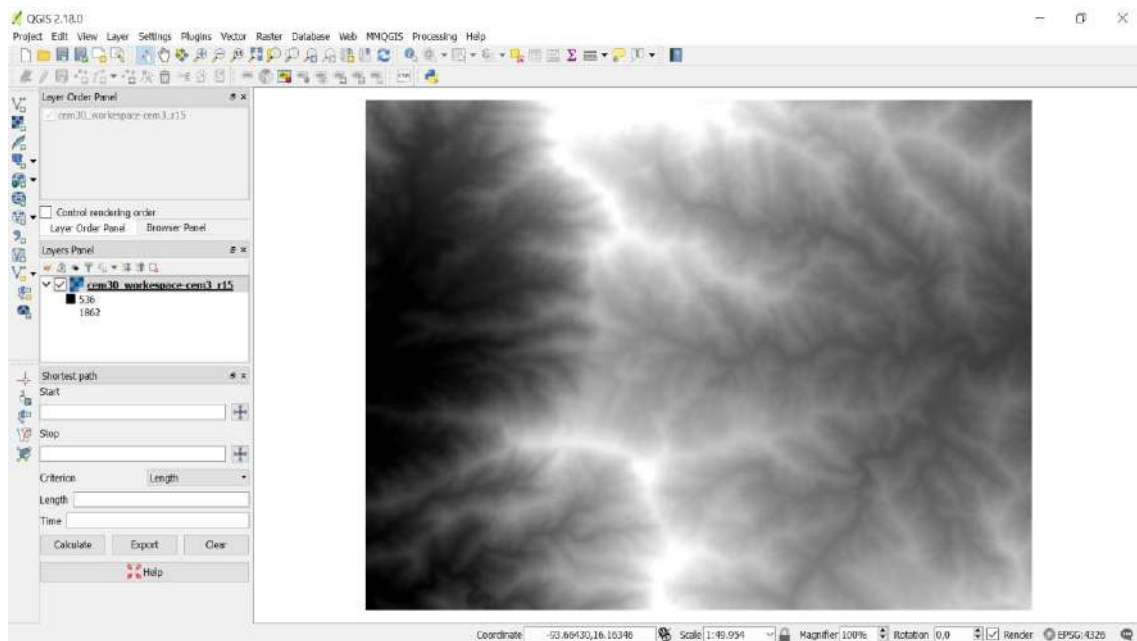


Figure 1: INEGI *.tif* file opened as a raster layer in QGIS.

Then, after being assigned a *Reference Coordinate System* (RCS), it can be converted into a *.stl* file using the *DEMto3D* tool within *Raster* menu. This tool is able to convert the *.tif* file into a *.stl* file and can be downloaded in *Plugins>Manage and Install Plugins*. The *DEMto3D* tool offers the option to specify the area to be converted using a square, defined by its corner's coordinates or a vector layer. It also provides the option to scale the map and modify its height. In this case the whole extension of the *.tif* was selected and no modifications were introduced, neither in the scale nor in the height.

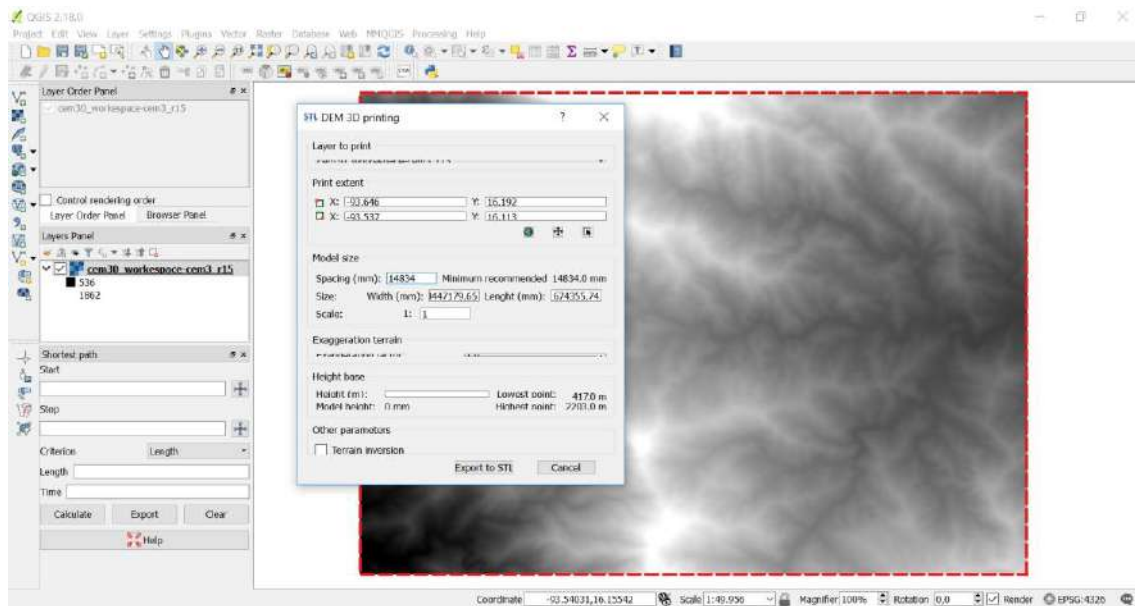


Figure 2: DEMto3D tool menu.

## Cover Land data processing

Here the procedure followed to translate the available land cover data of the interest area into information that can be read by OF is described. According to this land cover data different length roughness and porosity values can be assigned to each part of the domain.

Although INEGI also offers land cover data (forest, water bodies, roads, settlements etc) in .shp format, which can be opened with QGIS as vector layers (see Fig.3), the information provided is not enough, for there is areas to which no land cover data is assigned.

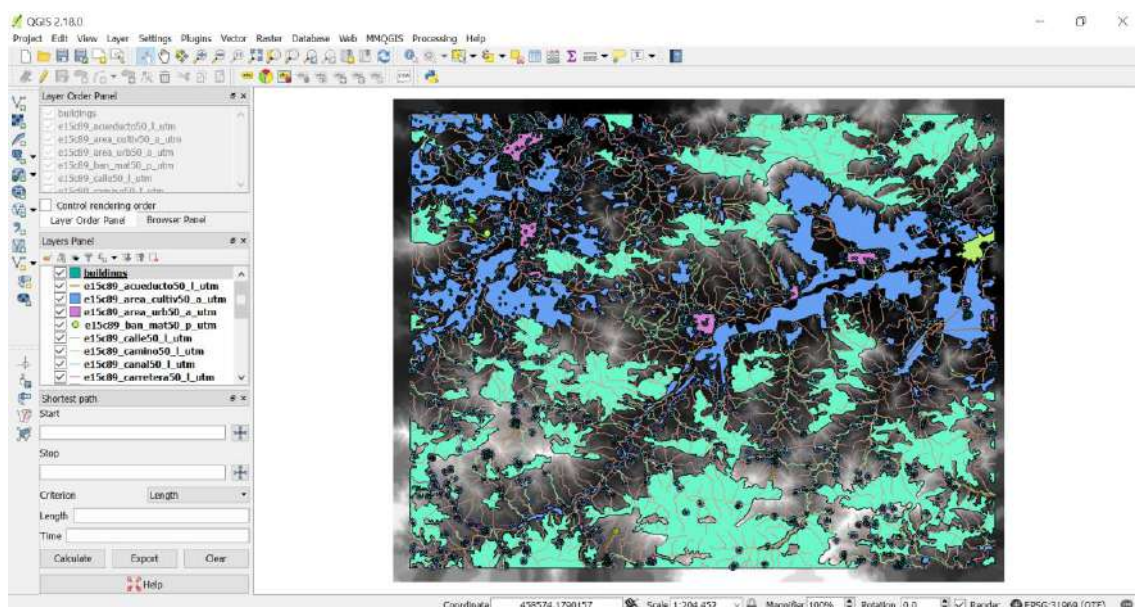


Figure 3: INEGI Land Cover vector layers.

For this reason, it was decided to elaborate land cover maps out of Google Earth satellite maps. Using the Google Earth's polygon option, polygons were drawn over those areas with different land cover. In the Study Case, three different sets of polygons were created, for forested areas, grass areas and urban areas (see Fig.4). There can be as many sets of polygons as types of land cover there are, so they cover the whole area, however that would require more precise information. The three polygon sets don't cover the whole area of interest and it was assumed that the area between the polygons was a mixture of grass, bushes and scattered small trees, which was referred to as open field in 7.5.3.

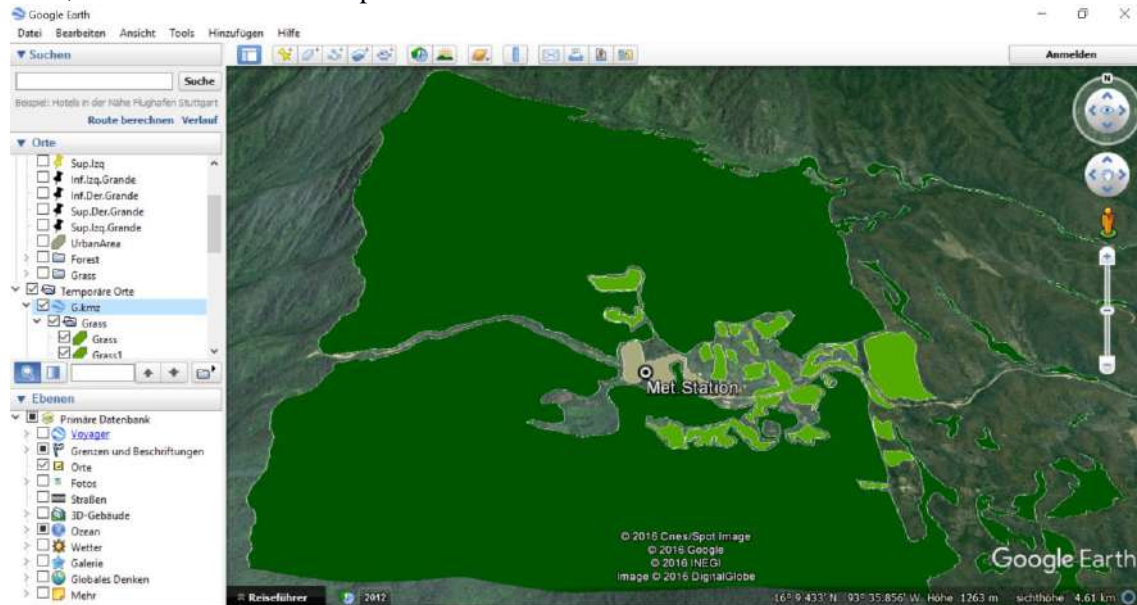


Figure 4: Land Cover polygons in Google Earth.

In QGIS the polygon sets were opened as vector layers (see Fig.5), assigned the same RCS as the elevation data and converted to *.shp* format. Then they were used to clip the *.tif* elevation file, resulting in a new *.tif* file which only contains the elevation of the area covered by the polygons for each polygon set (see Fig.6). This was done using the *clipper* option within the *Raster* menu, and selecting the corresponding *.shp* file as mask layer. It is also necessary to select the “No data value” option and assign a 0 value to those areas of the *.tif* map not covered by the selected *.shp* file. Finally these new *.tif* files can be converted into *.stl* files following the same procedure explained above. In the *DEMto3D* tool window the original *.tif* elevation map was selected as the *Print extent*.

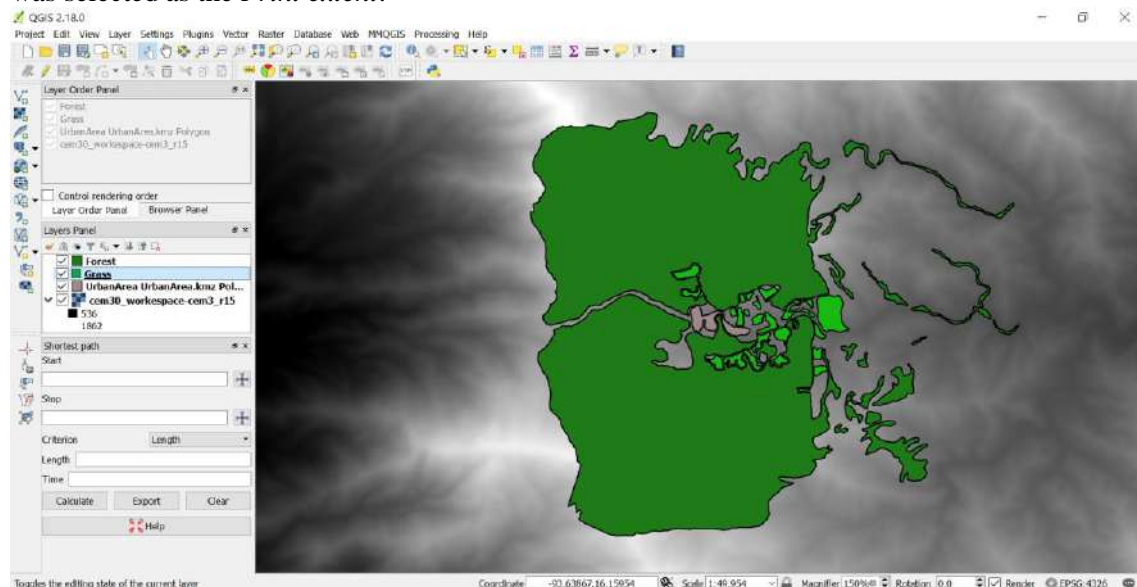


Figure 5: Land Cover polygons from Google Earth open in QGIS and converted into *.shp* files.

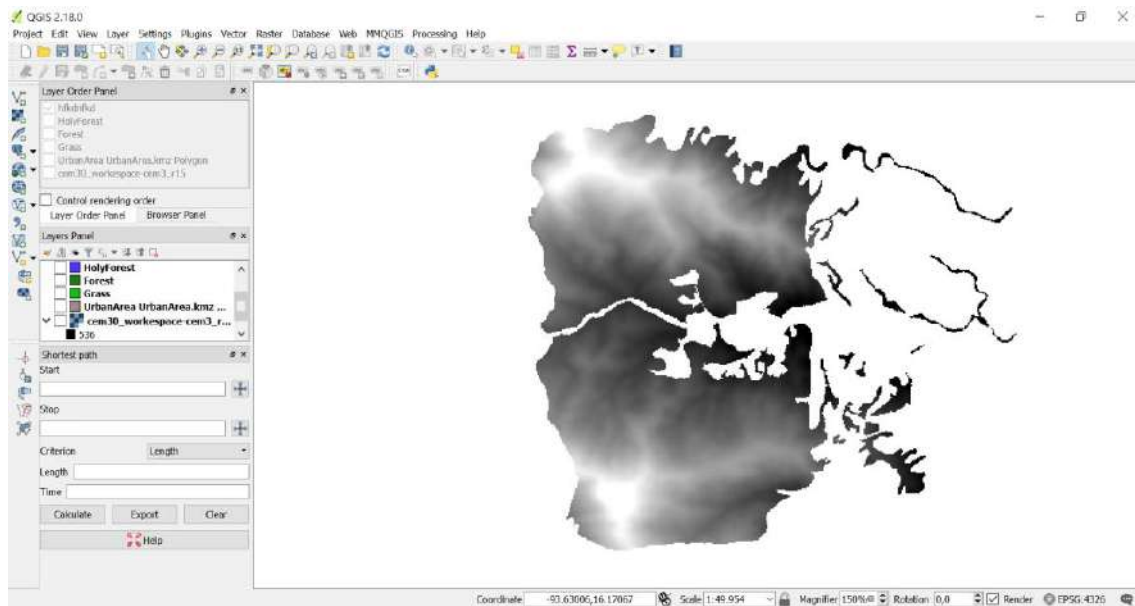


Figure 6: Forest .tif raster layer.

Once the corresponding .*stl* files for the grass-, forest-, and buildings-covered areas were created, these steps were followed to create patches in OF to which different roughness length could be assigned:

- The domain, which ground patch was the .*stl* file created out of the elevation data, was meshed according to the *snappyHexMeshDict* presented in [Annex 4](#).
- Through the following *topoSetDict* file the ground patch faces belonging to grass-, forest- or buildings-covered areas according to the land cover .*stl* files were grouped into three different faces sets, one for each kind of cover:

```

/*-----*- C++ -*-----*\
| =====|
| \\      / F ield      | OpenFOAM: The Open Source CFD Toolbox
| \\      / O peration  | Version: 2.3.0
| \\      / A nd        | Web: www.OpenFOAM.org
|  \\     M anipulation |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    object       topoSetDict;
}

// * * * * *
//

actions
(
    {
        name      UrbanPoints;
        type      pointSet;
        action     new;
        source     surfaceToPoint;
        sourceInfo
    }
)

```



```

    {
        file "./constant/triSurface/urban.stl";
        nearDistance 3000;
        includeInside false;
        includeOutside false;
    }
}
{
    name    UF;
    type    faceSet;
    action  new;
    source  pointToFace;
    sourceInfo
    {
        set UrbanPoints;
        option any;
    }
}
{
    name    GrassPoints;
    type    pointSet;
    action  new;
    source  surfaceToPoint;
    sourceInfo
    {
        file "./constant/triSurface/grass.stl";
        nearDistance 4000;
        includeInside false;
        includeOutside false;
    }
}
{
    name    GF;
    type    faceSet;
    action  new;
    source  pointToFace;
    sourceInfo
    {
        set GrassPoints;
        option any;
    }
}
{
    name    ForestPoints;
    type    pointSet;
    action  new;
    source  surfaceToPoint;
    sourceInfo
    {
        file "./constant/triSurface/forest.stl";
        nearDistance 3000;
        includeInside false;
        includeOutside false;
    }
}
{
    name    FF;
    type    faceSet;
    action  new;
    source  pointToFace;
    sourceInfo

```

```

        {
            set ForestPoints;
            option any;
        }
    }
);

// *****

```

However, these faces sets contain, not only ground patch faces but also a few faces perpendicular to the terrain which prevent these sets to be directly converted into a patch. To eliminate those faces, several operations had to be made using the ground patch and the land cover faces sets. For this purpose, the following *makeCellSets.setSet* file (which can be as well initiated from the windows terminal) was used:

```

//Urban
faceSet groundFaces new patchToFace ground1-30
faceSet groundFaces delete faceToFace UF
faceSet urbanFaces new patchToFace ground1-30
faceSet urbanFaces delete faceToFace groundFaces
//Grass
faceSet groundFaces1 new patchToFace ground1-30
faceSet groundFaces1 delete faceToFace GF
faceSet grassFaces new patchToFace ground1-30
faceSet grassFaces delete faceToFace groundFaces1
//Forest
faceSet groundFaces2 new patchToFace ground1-30
faceSet groundFaces2 delete faceToFace FF
faceSet forestFaces new patchToFace ground1-30
faceSet forestFaces delete faceToFace groundFaces2

```

Now the resulting face sets contain only ground patch faces and can therefore be used in turn to create patches for each land cover type. That was made through a *createPatchDict*:

```

/*-----*- C++ -*------*\
| =====|
| \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox
| \ \ / O peration | Version: 2.3.1
| \ \ / A nd | Web: www.OpenFOAM.org
| \ \ / M anipulation |
\*-----*/
FoamFile
{
    version 2.0;
    format ascii;
    class dictionary;
    object createPatchDict;
}
// *****
pointSync false;
patches
(
{
name urban;
patchInfo

```

```

{
type wall;
}
constructFrom set;
set urbanFaces;
}
{
name forest;
patchInfo
{
type wall;
}
constructFrom set;
set forestFaces;
}
{
name grass;
patchInfo
{
type wall;
}
constructFrom set;
set grassFaces;
}
);
// ***** //

```

The result of this procedure are three new patches, for three of the four types of land cover identified. They can be called in the variable files within the 0 folder, and therefore, different roughness length values can be assigned to them (see [Annex 4](#)). The original ground patch now only represents the area corresponding to the so-called open field areas (see Fig.7).



Figure 7: Land Cover patches.

The described procedure can be applied to forested areas when, as it happened in the Study Case (see [7.5.3](#)), the lack of available data about the leaf density profiles and other characteristics of the existing vegetation make impossible to apply the canopy model. In that case forested areas had to be modelled through an enhanced roughness length value [\[124\]](#). However, if the necessary information is available and therefore, the forest can be modelled as a canopy a different procedure has to be followed.

The steps followed would be the same until the point where the .tif file containing the elevation information of the forest-covered areas is created. After that a change has to be introduced in the conversion process to a .stl file. In the *DEMto3D* tool the mean height value of the trees to be represented must be typed where it says “*Height*”. This way the resulting .stl file would represent a terrain separated from the original by that same value (see Fig 8).

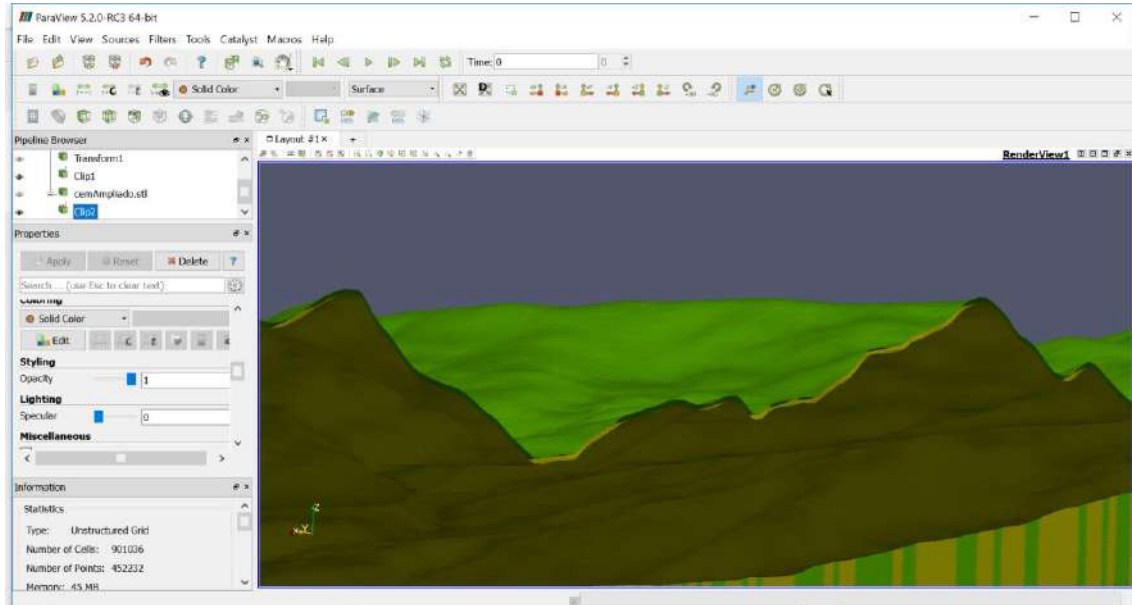


Figure 8: Elevation and forest .stl files.

This new forest .stl file will be used to select the cells of the mesh which belong to the canopy, through the following *topoSetDict*:

```
/*-----*- C++ -*-----*\
| =====|
| \ \ / F ield | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O peration | Version: 2.3.0 |
| \ \ / A nd | Web: www.OpenFOAM.org |
| \ \ / M anipulation |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    object       topoSetDict;
}

// * * * * *

actions
(
    {
        name      ForestPoints;
        type      pointSet;
        action     new;
        source     surfaceToPoint;
        sourceInfo
        {
```



```

        file "../constant/triSurface/forestHeight.stl";
        nearDistance 3000;
        includeInside true;
        includeOutside false;
    }
}
{
    name    CanopyCells;
    type    cellSet;
    action  new;
    source  pointToCell;
    sourceInfo
    {
        set ForestPoints;
        option any;
    }
}
);

// ***** //

```

It must be noticed that, in order to select those cells between the ground and the forest .stl file, within the *topoSetDict*, the option *includeInside* must be selected in the *surfaceToPoint* entry. Finally, once the *cellSet* has been created, it can be assigned darcy-forchheimer coefficients through a *porosityProperties* dictionary (see [Annex 4](#)).