Czech Technical University in Prague
Faculty of Mechanical Engineering
Department of Technical Mathematics

Doctoral thesis

NUMERICAL SIMULATIONS OF MICROSCALE ATMOSPHERIC FLOWS AND POLLUTION DISPERSION

Mgr. Viktor Šíp

Doctoral study programme: Mechanical Engineering

Branch of study: Mathematical and Physical Engineering

Supervisor: Doc. Ing. Luděk Beneš, Ph.D.

2016

Declaration

I hereby declare that the thesis submitted is my own unaided work. All direct or indirect sources used are acknowledged as references.

Abstract

A 3D finite volume solver aimed at solving atmospheric boundary layer flows and pollutant dispersion in domains with complex geometry was developed. It features two methods based on different sets of equations. First is based on the general equations of compressible flow, and employs low Mach preconditioning to accelerate its convergence. Second is based on the simplified equations of atmospheric boundary layer flow under the assumption of incompressibility and the Boussinesq approximation, and its solution relies on the artificial compressibility method. Reynolds-averaged Navier-Stokes approach is used for the turbulence modelling. The comparable accuracy and computational performance of both methods was demonstrated on example problems.

The effects of the vegetation were included in the fluid flow model and in the employed k-ϵ turbulence model. A detailed, physically based model of the dry deposition of aerosol particles on the vegetation surface was implemented to account for the filtering properties of the vegetation. The applicability of the models was assessed by comparison with the available field measurements. The developed methods were then utilized on the problem of a near-road vegetation barrier optimization.

A moment method for solution of the dispersion problems of particulates of a wide size range was further examined, and the dry deposition model was adapted for use with the method. The method was shown to provide a computationally efficient alternative to the simpler sectional method.

Keywords: Atmospheric boundary layer, RANS modelling, Pollution dispersion, Vegetation modelling, Moment method

Abstrakt

V této práci byl vyvinut 3D řešič postavený na metodě konečných objemů, zaměřený na problémy proudění v mezní vrstvě atmosféry a šíření zplodin v geometricky komplexních oblastech. Řešič obsahuje dvě metody založené na dvou sadách rovnic. První je založena na obecných rovnicích stlačitelného proudění, a pro urychlení konvergence využívá předpodmínění pro nízká Machova čísla. Druhá metoda je založena na zjednodušených rovnicích pro proudění v mezní vrstvě atmosféry za předpokladu nestlačitelnosti proudění a Boussinesqovy aproximace. Pro modelování turbulence byl nasazen přístup Reynoldsova průměrování Navier-Stokesových rovnic. Srovnatelná přesnost a výpočetní náročnost obou metod byla demonstrována na dvou ukázkových problémech.

Efekty vegetace byly začleněny do modelu proudění a použitého k-ϵ turbulentního modelu. Filtrační schopnost vegetace byla zachycena detailním, na fyzikálních principech postaveným modelem suché depozice aerosolových částic na povrch vegetace. Použitelnost modelů byla zhodnocena porovnáním s dostupnými měřeními. Vyvinuté metody byly poté využity na problém optimalizace vegetační bariéry u silniční komunikace.

Dále byla zkoumána momentová metoda, vhodná k řešení problémů šíření pevných částic velkého rozsahu velikostí. Model suché depozice byl adaptován pro použití s touto metodou. Bylo ukázáno, že metoda poskytuje výpočetně méně náročnou alternativu jednodušší sekční metodě.

Klíčová slova: Mezní vrstva atmosféry, RANS modelování, Šíření zplodin, Modelování vegetace, Momentová metoda

Acknowledgements

Contents

Nomenclature

Acronyms

ABL   Atmospheric Boundary Layer

BC   Boundary Condition

CFD   Computational Fluid Dynamics

DNS   Direct Numerical Simulation

JFNK   Jacobian-Free Newton-Krylov

LAD   Leaf Area Density

LAI   Leaf Area Index

LES   Large Eddy Simulation

ODE   Ordinary Differential Equation

PDE   Partial Differential Equation

PM   Particulate matter

RANS   Reynolds-Averaged Navier-Stokes

RHS   Right hand side

TKE   Turbulence Kinetic Energy

Greek Symbols

Symbol	Description	Units
$\boldsymbol{\Gamma}$	Local preconditioning matrix	
Γ	Lapse rate	$\mathrm{K\,m^{-1}}$
γ	Specific heat ratio, $\gamma = c_p/c_v$	1
Γ_d	Adiabatic lapse rate	$\mathrm{K\,m^{-1}}$
δ_{ij}	Kronecker delta	

$\boldsymbol{\epsilon}_i$	Perturbation vector	
ϵ	Turbulence kinetic energy dissipation	$\mathrm{m^2\,s^{-3}}$
θ	Potential temperature	K
κ	Von Kármán constant	1
λ	Mean free path of the particle in the air	m
μ, μ_T, μ_E	Dynamic viscosity (laminar, turbulent, effective)	$\mathrm{kg\,m^{-1}\,s^{-1}}$
ν, ν_T, ν_E	Kinematic viscosity (laminar, turbulent, effective)	$\mathrm{m^2\,s^{-1}}$
ρ	Density	$\mathrm{kg\,m^{-3}}$
ρ_p	Particle density	$\mathrm{kg\,m^{-3}}$
σ_g	Geometric standard deviation	1
$\boldsymbol{\tau}$	Viscous stress tensor	Pa
$\boldsymbol{\tau}^E$	Effective stress tensor	Pa
$\boldsymbol{\tau}^R$	Reynolds stress tensor	Pa
τ_p	Particle relaxation time, $\tau_p = (\rho_p C_C d_p^2)/(18\mu)$	s
Φ	Normal cumulative distribution function	1

Latin Symbols

Symbol	Description	Units
a	Speed of sound	$\mathrm{m\,s^{-1}}$
c	Pollutant concentration	$\mathrm{kg\,m^{-3}}$
C_C	Cunningham correction factor	1
C_d	Drag coefficient	1
c_p	Specific heat capacity at constant pressure	$\mathrm{J\,kg^{-1}\,K^{-1}}$
c_v	Specific heat capacity at constant volume	$\mathrm{J\,kg^{-1}\,K^{-1}}$
d_e	Size of a vegetation element	m
d_{gn}	Geometric mean size	m
d_p	Particle diameter	m
E	Total energy per unit mass	$\mathrm{m^2\,s^{-2}}$
e	Specific internal energy	$\mathrm{m^2\,s^{-2}}$

$\boldsymbol{f}$	Density of volume forces per unit mass	$\mathrm{m\,s^{-2}}$
$\boldsymbol{F}_j$	Inviscid flux	
$\boldsymbol{g}$	Gravitational acceleration vector	$\mathrm{m\,s^{-2}}$
g	Gravitational acceleration	$\mathrm{m\,s^{-2}}$
$\boldsymbol{I}$	Unit matrix	
k	Turbulence kinetic energy	$\mathrm{m^2\,s^{-2}}$
k_L	Heat conduction coefficient	$\mathrm{W\,m^{-1}\,K^{-1}}$
M	Mach number	1
M_k	k-th moment of a distribution	$\mathrm{m^k\,m^{-3}}$
$\boldsymbol{n}$	Normal vector	
$n_N(d_p)$	Aerosol size distribution function	$\mathrm{m^{-1}\,m^{-3}}$
N_T	Total number concentration	$\mathrm{m^{-3}}$
$n_V(d_p)$	Aerosol volume distribution function	$\mathrm{m^2\,m^{-3}}$
p	Pressure	Pa
Pr_T	Turbulent Prandtl number	1
$\boldsymbol{Q}$	Source term	
q	Density of heat sources per unit mass	$\mathrm{m^2\,s^{-3}}$
$\boldsymbol{R}_j$	Viscous flux	
R	Gas constant, $R = c_p - c_v$	1
Re	Reynolds number	1
Sc	Schmidt number	1
Sc_T	Turbulent Schmidt number	1
S_T	Total surface concentration	$\mathrm{m^2\,m^{-3}}$
St	Stokes number	1
$\boldsymbol{T}$	Stress tensor	Pa
T	Temperature	K
t	Time	s
$\boldsymbol{u}$	Velocity vector	$\mathrm{m\,s^{-1}}$

U	Velocity magnitude	$\mathrm{m\,s^{-1}}$
u_*	Friction velocity	$\mathrm{m\,s^{-1}}$
u_d	Deposition velocity	$\mathrm{m\,s^{-1}}$
u_s	Gravitational settling velocity	$\mathrm{m\,s^{-1}}$
V_T	Total volume concentration	$\mathrm{m^3\,m^{-3}}$
$\boldsymbol{W}$	State vector	
$\boldsymbol{x}$	Position vector	m
z_0	Roughness length	m

Chapter 1

Introduction

1.1 Motivation

In this work we deal with the numerical modelling of the flow and pollution dispersion in the atmospheric boundary layer. The *atmospheric boundary layer* (ABL) is the lowest part of the atmosphere, roughly below 2 kilometers, and it is the part of the atmosphere directly affected by the ground surface. Its interest to researches stems from the fact that the ABL is where humans live and work, and it is also where most human-made pollutant are released into the atmosphere. The modelling of the ABL has several important areas of applications: wind energy production, pollution dispersion, or problems of the urban flow.

Urban areas are of special interest regarding the ABL research and modelling. With the continuing urbanization, more and more people move to the densely packed metropolitan areas. The inhabitants' health is negatively affected by the increased air pollution in the urban zones, caused by the road traffic, industrial sources of air pollutants, or residential burning. Furthermore, urban planning and architecture has a direct impact on the wind comfort of the pedestrians. Heat stress and thermal comfort are another factors affected by the wind flow around the buildings. Detailed understanding of the processes in the ABL and our capability to accurately predict their effects is therefore crucial for mitigating the negative effects of the urbanization and improving the well-being of the urban inhabitants.

Modelling of the wind flow and pollutant dispersion in the urban areas can be classified as belonging to the general class of *microscale* problems of atmospheric flows. These focus on the phenomena occurring on the scales roughly below one kilometer. This distinguishes them from the *mesoscale* problems, occurring on scales of few kilometers to hundreds of kilometers, and *synoptic scale* problems on scales of thousands of kilometers.

Vegetation plays an important role in the microscale problems, and especially so in the urban areas. Urban vegetation, such as parks, street trees, roof gardens, or green walls, can block or slow down the air flow, and therefore can affect the wind and thermal comfort of the pedestrians. Urban vegetation was also proposed as a measure for mitigating the impacts of the air pollution due to its capability to act as an air filter. Indeed, the dry deposition process can capture particulate matter on the leaves of the vegetation, and thus reduce the pollution levels in the neighbourhood.

Numerical modelling is an indispensable tool in the effort of understanding the related phenomena. It can provide answers not obtainable by any other method. Field experiments, while necessary for validation, are generally expensive, and the measured data provide only limited view on the underlying physical processes. Furthermore, only flow around already existing buildings or other features can be investigated, limiting the role of the field experiments in the design process. Wind tunnel modelling, on the other hand, has to deal with the problem of scaling, which is - especially for vegetation - challenging. Moreover, even with the modern measurement techniques, wind tunnel studies cannot offer such detailed information about the flow fields as the computational fluid dynamics (CFD) can.

Several characteristics of the ABL and urban flow especially have to be taken into account when considering using a CFD solver for above outlined problems: Such flow are generally fully turbulent with very high Reynolds numbers, and low Mach numbers. Thermal and stratification effects can play important role in the flow. The geometry of the problems is often very complex, prohibiting the use of structured grids. Finally, the effects of the vegetation on the flow field and on the pollutant concentration has to be modelled in some way. This work is devoted to development and validation of such solver.

1.2 State of the art

1.2.1 Equations of the fluid flow

The task of the computational modelling of the fluid flow in the ABL starts with the choice of the physical model describing the equations. Various models and equation sets were employed. For urban flow application where the thermal and stratification effects can be neglected the Navier-Stokes equations of the incompressible flow consisting of four equations (non-divergence constraint and three velocity components in 3D) are often used (Balogh et al., 2012; Blocken et al., 2012; Vranckx et al., 2015). Compelling advantage of such approach is that the incompressible flow model is often present in the publicly available open source and commercial CFD solvers. However, this simple model allows investigation only of the flow in the neutrally stratified atmosphere without any heat sources. Considering that the urban heat island is mentioned as one of the major problems of modern cities (Rizwan et al., 2008), and that the atmospheric stratification has a direct influence on the diffusion and dispersion of the pollutants (Arya, 1999), the four-equation incompressible flow model is an oversimplification for our purpose.

More advanced class of models of the ABL flow can be labeled as variable density incompressible flow models using the Boussinesq approximation. These models generally use five equations (non-divergence constraint and velocity components as above, plus one additional equation, typically for temperature or potential temperature). Zeytounian (2003) provides a view on the history and applicability of the approximation and shows that the applications goes far beyond the ABL flows. Models using the Boussinesq approximation are widespread in the atmospheric research (e.g. Apsley and Castro, 1997b; Eidsvik and Utnes, 1997; Dupont and Brunet, 2008b; Mehta et al., 2014), and were also used at the author's department before (Bodnár, 2003; Beneš et al., 2011; Bodnár et al.,

2012). Strict focus of these models on the atmospheric research however limits their applicability and reuse elsewhere, and, conversely, calls for a dedicated solver for the ABL flows only. Furthermore, these models are limited to low Mach flows, and problems with mixed low and high speed flows are outside of their domain of applicability. Such mixed speeds problems may appear in the ABL, for example the flow around the fast moving tip of the wind turbine blade, or extreme weather events such as tornadoes, where Mach number can rise above 0.3.

Usage of the models of compressible flow, based on the equations of conservation of mass, momentum, and energy, is less common for atmospheric applications. Their main disadvantage is that the compressible flow solvers suffer from slow convergence when applied to the low speed flow problems due to the disparity of the acoustic and convective wave speeds. Nevertheless, few studies using such models exist. Duarte et al. (2014) studied moist atmosphere flows using the compressible Euler equations. They used an explicit temporal discretization, and the time step in their simulation was limited by the acoustic CFL condition. In their investigated cases the condition was not severely restrictive, but it makes the scheme impractical for small scale flows. Yang and Cai (2014) modelled atmospheric flow with an compressible Euler solver, although using the equation for potential temperature instead of the energy equation. They solved the time step restriction problem by employing an implicit temporal scheme together with the additive Schwarz preconditioner. Another approach was taken by Jafari (2014), who used the low Mach preconditioning technique of Weiss and Smith (1995) to develop an efficient solver for wind energy applications.

To summarize, the general formulation for the compressible flow offers some advantages over the ABL flow formulation, and the low Mach preconditioning techniques promise to overcome its drawbacks. In this work we plan to compare the two options to see whether the comparable performance can be obtained.

1.2.2 Turbulence modelling

Another question is the choice of the turbulence modelling methodology. The main approaches used today are the Reynolds-averaged Navier-Stokes (RANS) equations that allows the simulation of the mean values of the flow variables, and the Large eddy simulation (LES) which resolves the large scale motions and models only the subgrid turbulence motions. For the flow through the vegetation both RANS (Svensson and Häggkvist, 1990; Katul et al., 2004; Steffens et al., 2012; Gromke and Blocken, 2015; Vranckx et al., 2015) and LES (Su et al., 1998; Dupont and Brunet, 2008a; Dupont et al., 2011; Mueller et al., 2014) were used. Comparing both approaches in the context of urban applications (although without any vegetation present), Blocken (2015) noted that RANS is the usually preferred choice due to the lower computational costs, but also due to the lacking extensive validation and missing best-practice guidelines for the LES approach. In their review of CFD simulations of urban pollution dispersion, Tominaga and Stathopoulos (2013) found that LES gives more accurate results than RANS, but also that the difference between the approaches is smaller for the mean flow variables than for the pollutant concentration. Better accuracy of LES, but also higher computational costs were observed also for simulations with vegetation present (Gromke and Ruck, 2012).

3

The computational cost of the simulation was a significant factor in our decision, and we opted for the RANS approach in this work.

1.2.3 Computational method and mesh

Crucial component of any CFD simulation is the computational method employed and the mesh on which is the simulation performed. For the flows above a reasonably smooth terrain without obstacles a structured grid may be used (as e.g. in Bodnár et al., 2012), and one may thus benefit from the easier implementation of the algorithms as well as the higher order methods for structured grids, using both finite difference and finite volume method. This approach however becomes less feasible for the geometrically complex domains with obstacles. For such applications are the finite volume solvers based on the unstructured grids a better option, although they have an alternative, which is the Immersed Boundary Method (IBM). IBM uses a Cartesian mesh that does not conform to the obstacle or terrain surface, and instead adds a forcing to the cells that interact with the boundary. Advantages of the method are fast and easy grid generation and relatively simple incorporation of body motion, while its main disadvantage is more complicated implementation of the boundary conditions. IBM was originally developed for biological flows, but it was since applied to atmospheric flows as well (Jafari, 2014). In this work, we opt for the approach using body-fitted unstructured meshes, mainly because reliable grid generators of unstructured meshes are available and because the problem of body motion is irrelevant to our intended application.

1.2.4 Dry deposition

The effect of the vegetation on the pollutant dispersion and its filtration properties has been investigated by researchers in their effort to mitigate the negative effects of the air pollution in the urban areas. Overview of this effort can be found in the reviews by Litschke and Kuttler (2008) or Janhäll (2015). Aerosol particles are removed from the air by the vegetation via the dry deposition process. Models of the dry deposition have a long history in the field of large scale air quality models (Slinn, 1982; Zhang et al., 2001; Petroff et al., 2008b; Petroff et al., 2009; Petroff and Zhang, 2010), in which the flow inside the canopy is not explicitly resolved.

In small scale numerical simulations, however, the approach to the dry deposition varies. In some small scale dispersion studies the process was not considered at all, such as in (Buccolieri et al., 2011), where the authors cited the low filtering potential of the vegetation as the main reason. Other used a constant rate of pollutant deposition, ignoring the rich physical background of the dry deposition (Vranckx et al., 2015), or various models based on the underlying processes of different level of detail (Tiwary et al., 2005; Bruse, 2007; Steffens et al., 2012). However, there is currently no generally accepted and extensively validated dry deposition model for the microscale vegetation flow problems.

1.2.5 Summary

To summarize, it is apparent from the above overview that a number of works related to the microscale atmospheric flows exists. However, if we consider our intended application to urban flows and pollutant dispersion in areas with vegetation present, the existing solvers fall short of one or more of the following aspects:

- Capability to capture the thermal and stratification effects.

- Suitability for geometrically complex domains.

- Capability to model the effects of the vegetation on the fluid flow.

- Dry deposition model which takes into account the dependence of the deposition rate on the properties of the particles and the vegetation.

These shortcoming lead us naturally to the declaration of the aims of this work.

1.3 Aims of the work

The main objective of the work is *to devise, implement, and validate an efficient method for the solution of the air flow and pollution dispersion microscale problems with vegetation present*. The partial goals pursued in order to fulfill the main one are as follows:

- To develop a RANS-based CFD solver for microscale ABL flows working on unstructured grids.

    The solver requirements come from our need for an efficient solver (therefore it should be RANS-based) that is capable of solving the problems of urban flows with complex geometry.

- To compare the accuracy and performance of the schemes based on the compressible flow equations and on the Boussinesq approximation.

    This shall be performed in the developed framework.

- To implement and validate suitable vegetation models.

    Models of the flow through the vegetation as well as of the dry deposition of the aerosol particles on the vegetation should be included.

1.4 Structure of the work

The work is structured in the following way. Chapters 2 and 3 can be read as a description of the computational solver developed as a part of this work. In Chapter 2 we present the mathematical background of the fluid flow solver. The equations of the fluid flow models are introduced, and the RANS approach to the turbulence modelling is detailed. We follow with the discussion of the boundary conditions for ABL flows. The chapter concludes with the description of the employed models of the vegetation, discussing both the fluid flow and the dry deposition. In Chapter 3 we deal with the numerical methods

used to solve the mathematical equations. Methods of artificial compressibility and low Mach preconditioning, modifying the original equations, are described. The spatial and temporal discretization methods used in the finite volume solver are elaborated upon.

Following two chapters deal with the validation and application of the developed solver. In Chapter 4 we validate the physical models and their implementation. The accuracy and performance of the compressible flow and ABL flow formulation are compared, and the suitability of the vegetation models is assessed. Chapter 5 presents an application of the developed models: a CFD optimization of a near-road vegetation barrier.

In Chapter 6 we step aside and pursue an alternative approach for modelling the pollutant dispersion called the moment method. The method is useful when the behaviour of particles in a wide size range is of concern.

Finally, the work in concluded with Chapter 7, where the work is summarized, and future perspectives are given.

Note on the software implementation.   The models and numerical methods described in Chapters 2 and 3 were implemented in the developed CFD solver nicknamed *Atifes*. Atifes originated from the software platform used previously at the author's department for the simulation of the electric discharge in a high voltage electric fields (Karel, 2014), although the further development proceeded independently. Atifes is written in C++ language.

The moment method solver described in Chapter 6 was developed separately using the OpenFOAM platform.

Chapter 2

Mathematical models of the fluid flow in the ABL

The content of this chapter is an expanded version of the model description in: Šíp, V. and Beneš, L. (2016d). RANS solver for microscale pollution dispersion problems in areas with vegetation: Development and validation. *arXiv e-print*. https://arxiv.org/abs/1609.03427 (Submitted).

In this chapter we introduce the models of the fluid flow in the atmosphere. We present two sets of the flow equations on which our solver is based: the compressible flow equations and the equations of the atmospheric boundary layer flow. We continue with the description of the passive scalar equation used for the modelling of the pollutant dispersion. Problem of turbulence modelling is discussed, and the turbulence models implemented in the solver are presented. The chapter follows with the overview of the boundary conditions related to the atmospheric boundary layer flows. In the last section of the chapter we then present the employed models of the vegetation, both regarding its effect on the flow field and its function as a filter of the polluted air.

2.1 Fluid flow equations

We present two fluid flow models that form the basis of the developed solver. First, it is the model of the compressible flow in the conservative form. The model is derived directly from the conservation laws and employs no approximation specific to the flows in the atmosphere. Similar models are often present in the general purpose CFD solvers, but are rarely used for modelling of the atmospheric flows.

Second presented model employs the non-divergence constraint and the so called Boussinesq approximation. This model or some of its variations is often used in the atmospheric research due to its relative simplicity. However, it is generally unavailable outside of the specialized research community. When researchers want to use a general purpose CFD solver for problems of atmospheric flows, they may be tempted to choose the closely related model of incompressible flow. While the fully incompressible model have its use cases, it is unable to capture many phenomena appearing in the ABL flows and thus may be a worse option that the above mentioned model of compressible flow.

2.1.1 Equations of the compressible flow

The principal equations of fluid dynamics are derived from the three conservation laws: conservation of mass, conservation of momentum, and conservation of energy. These three laws give rise to the continuity equation, momentum equation, and energy equation respectively. Detailed derivation of the equations can be found elsewhere (e.g. Feistauer et al., 2003; Blazek, 2001) and is not pursued here.

Continutity equation

The law of the mass conservation states that the amount of mass in a closed system cannot change. For a fluid with the density $\rho(\boldsymbol{x}, t)$ and velocity $\boldsymbol{u}(\boldsymbol{x}, t)$, this can be expressed in the differential form,

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \boldsymbol{u}) = 0. \tag{2.1}$$

Momentum equation

The law of the conservation of momentum states that the rate of change of the momentum of a piece of fluid is equal to the force acting on it. The forces acting on the piece of fluid are *body forces*, acting on directly on the fluid, and *surface forces*, acting on the surface of a control volume.

The body forces per unit volume can be expressed as $\rho \boldsymbol{f}$. The body forces in the atmospheric simulations can include gravitational force or Coriolis force, and we will describe them later in greater detail.

The surface forces are expressed through the stress tensor $\boldsymbol{T} = (T_{ij})_{i,j=1}^3$, and then the momentum equation can be written in vector form as

$$\frac{\partial(\rho \boldsymbol{u})}{\partial t} + \operatorname{div}(\rho \boldsymbol{u} \otimes \boldsymbol{u}) = \rho \boldsymbol{f} + \operatorname{div} \boldsymbol{T}. \tag{2.2}$$

The stress tensor consists of an isotropic pressure component and a viscous stress tensor $\boldsymbol{\tau}$,

$$\boldsymbol{T} = -p\boldsymbol{I} + \boldsymbol{\tau}. \tag{2.3}$$

For Newtonian fluids the viscous stress tensor $\boldsymbol{\tau} = (\tau_{ij})_{i,j=1}^3$ has the form

$$\boldsymbol{\tau} = (\lambda \operatorname{div} \boldsymbol{u})\boldsymbol{I} + 2\mu \boldsymbol{D}, \tag{2.4}$$

where

$$\boldsymbol{D} = (D_{ij})_{i,j=1}^3, \quad D_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{2.5}$$

is the strain rate tensor, μ denotes the dynamic viscosity coefficient and λ is the second viscosity coefficient. The viscosity coefficients are tied together through the Stokes' hypothesis,

$$\lambda + \frac{2}{3}\mu = 0. \tag{2.6}$$

Using relation (2.3) in Eq. (2.2), we obtain the final form of the momentum equation,

$$\frac{\partial(\rho\boldsymbol{u})}{\partial t} + \text{div}\,(\rho\boldsymbol{u} \otimes \boldsymbol{u}) = -\nabla p + \rho\boldsymbol{f} + \text{div}\,\boldsymbol{\tau}, \tag{2.7}$$

with $\boldsymbol{\tau}$ specified by (2.4) and (2.6).

Energy equation

The energy equation is a mathematical representation of the law of conservation of energy. The law states that the changes in total energy inside the volume are caused by the power of forces acting on the volume and the heat transmitted to it.

The mathematical form of the law is written in terms of the total energy per unit mass,

$$E = e + \frac{|\boldsymbol{u}|^2}{2}, \tag{2.8}$$

where e is the specific internal energy.

The density of the heat sources per unit mass is denoted as q. The heat flux $\boldsymbol{q}$ is expressed through the *Fourier's law*,

$$\boldsymbol{q} = -k_L \nabla T. \tag{2.9}$$

Here k is the heat conduction coefficient and T is the temperature of the fluid. Then the energy equation can be written as

$$\frac{\partial(\rho E)}{\partial t} + \text{div}\,(\rho E\boldsymbol{u}) = \rho\boldsymbol{f} \cdot \boldsymbol{u} - \text{div}\,(p\boldsymbol{u}) + \text{div}\,(\boldsymbol{\tau}\boldsymbol{u}) + \text{div}\,(k_L\nabla T) + \rho q. \tag{2.10}$$

External forces

Two external forces are relevant for the atmospheric simulations. First, it is the gravitational force,

$$\boldsymbol{f}_g = (0, 0, -g)^T, \tag{2.11}$$

where g is the gravitational acceleration. Secondly, it is the Coriolis force,

$$\boldsymbol{f}_c = 2\boldsymbol{\Omega} \times \boldsymbol{u}, \tag{2.12}$$

with $\boldsymbol{\Omega}$ being the angular velocity vector. Vertical component, orders of magnitude smaller than other terms in the equations, is however often neglected, and the force is simplified to

$$\boldsymbol{f}_c = (f_c u_2, -f_c u_1, 0)^T. \tag{2.13}$$

Here

$$f_c = 2\Omega \sin\phi \tag{2.14}$$

is the Coriolis parameter, $\Omega = 2\pi\,\text{rad}/24\,\text{hours}$ is the angular velocity, and ϕ is the latitude.

The force acting can then be expressed as a sum of the gravitational and Coriolis force,

$$\boldsymbol{f} = \boldsymbol{f}_g + \boldsymbol{f}_c. \tag{2.15}$$

2.1.2 Thermodynamical relations

To close the system of equations (2.1, 2.7, 2.10), additional relations between p, ρ, T and e have to be specified. The dry air in the atmosphere can be considered to be an ideal gas, for which the equation of state has the form

$$p = \rho RT, \tag{2.16}$$

where $R > 0$ is the gas constant. The gas constant R can be expressed as a difference between the specific heat at constant pressure, c_p, and specific heat at constant volume, c_v,

$$R = c_p - c_v. \tag{2.17}$$

Furthermore, internal energy of an ideal gas is given by

$$e = c_v T. \tag{2.18}$$

Speed of sound in the fluid is defined as

$$a = \sqrt{\left(\frac{\partial p}{\partial \rho}\right)_s}, \tag{2.19}$$

where the derivative is taken at constant entropy. In ideal gas, this can be simplified to

$$a = \sqrt{\gamma \frac{p}{\rho}}. \tag{2.20}$$

2.1.3 Atmospheric boundary layer flow model

The second set of equations used in this work is derived specifically for flows in the atmospheric boundary layer. It is based on two major assumptions:

1. The flow can be approximated as incompressible.

2. *Boussinesq approximation*: perturbations of density from the background state in hydrostatic balance can be neglected everywhere except for the gravity term.

These points will be discussed here. Again, detailed derivation of the model is not pursued here, as it can be found elsewhere (Bodnár, 2003; Seinfeld and Pandis, 2006, chap. 16).

Before we proceed, let us introduce some terms and notation related to the atmospheric dynamics. *Potential temperature* θ is defined as temperature that a parcel of air would have if it were brought adiabatically to a reference pressure p_{ref} from its initial pressure p. In ideal gas the following formula can be derived,

$$\theta = T \left(\frac{p_{\mathrm{ref}}}{p}\right)^{R/c_p}. \tag{2.21}$$

Potential temperature will be useful when reasoning about the atmospheric stability in Sec. 2.1.3.

Hydrostatic equation relates the pressure and density in the atmosphere at rest. If we take Eq. (2.7), neglect the viscosity and assume steady, horizontal flow, the vertical velocity equation will reduce to

$$-\frac{1}{\rho}\frac{\partial p}{\partial z} = g. \tag{2.22}$$

This equation represents the balance between the forces resulting from the pressure gradient on one side and gravitational acceleration on the other side. When pressure and density satisfy this equation, the fluid is said to be in a *hydrostatic balance*.

Let pressure, density and potential temperature be decomposed into their background components dependent only on the height coordinate, denoted with subscript $_0$, and their fluctuations, denoted with superscript *,

$$
\begin{aligned}
p &= p_0 + p^*, \\
\rho &= \rho_0 + \rho^*, \\
\theta &= \theta_0 + \theta^*.
\end{aligned}
\tag{2.23}
$$

The background component represents the state of the air at hydrostatic balance, so that Eq. (2.22) is satisfied for p_0 and ρ_0. Background potential temperature is related to the background pressure and density through the equation of state (2.16) and through the potential equation formula (2.21).

Incompressibility assumption

The assumption of the incompressibility of the flow may simplify the flow equations. The flow is said to be incompressible if the material derivative of the density vanishes,

$$\frac{D\rho}{Dt} \equiv \frac{\partial \rho}{\partial t} + \boldsymbol{u} \cdot \nabla \rho = 0, \tag{2.24}$$

which together with the continuity equation (2.1) implies

$$\operatorname{div} \boldsymbol{u} = 0. \tag{2.25}$$

Flow in the atmospheric boundary layer can often be considered incompressible if some conditions are satisfied. Atkinson (1995) derives one such condition that states that the vertical length scale of the air circulation has to be significantly smaller than the density scale depth of the atmosphere,

$$L_z = \frac{u_z}{\operatorname{div} \boldsymbol{u}} \ll \frac{\alpha_0}{d\alpha_0/dz} = H_a \sim 8\,\mathrm{km}, \tag{2.26}$$

where $\alpha_0(z)$ is the background component of the specific volume of the fluid at hydrostatic balance. Baines (1995) states that for the flow forced by topography the divergence constraint (2.25) can be used if $|\boldsymbol{u}| < 100\,\mathrm{m\,s^{-1}}$ everywhere in the modelled domain and

$$\frac{N^2 H^2}{a^2} \ll 1. \tag{2.27}$$

Here $N = -\frac{g}{\rho}\frac{\partial \rho}{\partial z}$ is the buoyancy frequency, H is vertical length scale and a is the speed of sound. He further notes that this criterion is satisfied for $H < 10\,\mathrm{km}$.

When the incompressibility assumption cannot be used, the *anelastic approximation* is often employed instead,

$$\mathrm{div}\,(\rho \boldsymbol{u}) = 0. \tag{2.28}$$

However, in this work we restrict ourselves only to the incompressible assumption (2.25).

Boussinesq approximation

We further employ the Boussinesq approximation. When using this approximation, density perturbations are assumed to be small compared to its background state,

$$\rho^* \ll \rho_0, \tag{2.29}$$

and it is assumed that the perturbations can be neglected everywhere in the flow equations except in the gravitational term.

With this assumption, following line of thought can be further pursued. Vertical momentum equation contains a large pressure gradient, mostly balanced by the gravity. Using the hydrostatic equation (2.22) and the assumption that the density perturbations are small compared to the background density, Atkinson (1995) derives the following approximation:

$$-\frac{1}{\rho}\frac{\partial p}{\partial z} - g \approx -\frac{1}{\rho_0}\frac{\partial p^*}{\partial z} + \frac{\theta^*}{\theta_0}g. \tag{2.30}$$

This approximation removes from the vertical momentum equation the large balanced terms and leaves only the small perturbations. Finally, further approximation consisting of replacing the background density by a reference constant value, e.g. the density at the ground level,

$$\frac{1}{\rho_0}\nabla p^* \approx \frac{1}{\rho_{\mathrm{ref}}}\nabla p^*, \tag{2.31}$$

allows us later to use a conservative finite volume scheme for the numerical solution of the equations.

System of equations

Finally, to remove the density from the momentum equation altogether, we use the kinematic viscosity, $\nu = \mu/\rho$, instead of the dynamic viscosity as in (2.7). One more approximation is used in the viscous terms,

$$\frac{1}{\rho}\mathrm{div}\,(\mu \nabla \boldsymbol{u}) \approx \mathrm{div}\,(\nu \nabla \boldsymbol{u}) \tag{2.32}$$

Using the ingredients outline above, we obtain the following equations describing the flow in the atmospheric boundary layer. First, it is the divergence constraint,

$$\mathrm{div}\,\boldsymbol{u} = 0. \tag{2.33}$$

Secondly, there are the momentum equations,

$$\frac{\partial \boldsymbol{u}}{\partial t} + \operatorname{div}\left(\boldsymbol{u} \otimes \boldsymbol{u}\right) = -\frac{1}{\rho_{\text{ref}}}\nabla p^* + \operatorname{div}\left(\nu\nabla\boldsymbol{u}\right) + \boldsymbol{f}^B, \tag{2.34}$$

where $\boldsymbol{f}^B = \boldsymbol{f}_g^B + \boldsymbol{f}_c$ is the sum of the gravitational force formulated in the Boussinesq approximation, $\boldsymbol{f}_g^B = (0, 0, \frac{\theta^*}{\theta_0}g)^T$ and of the Coriolis force $\boldsymbol{f}_c$ given by Eq. (2.13). Lastly, the energy equation is replaced by the potential temperature equation,

$$\frac{\partial \theta}{\partial t} + \operatorname{div}\left(\theta\boldsymbol{u}\right) = \operatorname{div}\left(\frac{k_L}{\rho_{\text{ref}}c_p}\nabla\theta\right) + \frac{q}{c_p}. \tag{2.35}$$

This equation shows that if we neglect the thermal diffusion and assume no heat source, the potential temperature is simply carried by the flow.

Stability of the atmosphere

Potential temperature θ is connected to the notion of stability of the atmosphere. Atmosphere is said to be statically *stable* if parcel of air after vertical displacement returns to its original position. It is *unstable* if accelerates in the direction of displacement and *neutral* if it stays at the new position.

This can be put more rigorously using the *adiabatic lapse rate*, which is the rate of decrease of the temperature in the adiabatic atmosphere,

$$\Gamma_d = -\frac{dT_a}{dz} = \frac{g}{c_p} \tag{2.36}$$

It can be then shown (see e.g. Holton, 2004) that in the sense described above the atmosphere is

- stable if $\partial T/\partial z > \Gamma_d$ or equivalently $\partial\theta/\partial z > 0$,

- neutral if $\partial T/\partial z = \Gamma_d$ or equivalently $\partial\theta/\partial z = 0$,

- unstable if $\partial T/\partial z < \Gamma_d$ or equivalently $\partial\theta/\partial z < 0$.

The form of the gravitational force in the momentum equation (2.34) allows to easily investigate effect of a vertical displacement. Recall that the potential temperature does not change after an adiabatic displacement. In the stable atmosphere, parcel of air displaced upwards has lower potential temperature than its surroundings, therefore the perturbation θ^* is negative and the force $\boldsymbol{f}_g^B$ is oriented downwards. Similarly, a downward displacement results in a positive θ^* and upward oriented gravitational force. The opposite holds true for the unstable atmosphere. In the neutral atmosphere, the potential temperature perturbation is always zero after an adiabatic displacement, and the displaced parcel stays at rest.

2.1.4 Flow in the neutrally stratified atmosphere

As a side note, we mention one further simplification of the model of the atmospheric flow that is not implemented in the developed solver. In neutrally stratified atmosphere without any heat sources and adiabatic walls, the potential temperature equation (2.35) will reduce to $\theta = const$. From this it follows that $\theta^* = 0$ and $\boldsymbol{f}_g^B = 0$. Assuming that the Coriolis force can be neglected, we can rewrite the equations (2.33), (2.34) and (2.35) to give us the usual form of the incompressible flow equations,

$$\text{div } \boldsymbol{u} = 0,$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \text{div } (\boldsymbol{u} \otimes \boldsymbol{u}) = -\frac{1}{\rho_{\text{ref}}} \nabla p^* + \text{div } (\nu \nabla \boldsymbol{u}). \tag{2.37}$$

Due to its availability in the commercial and open-source CFD solvers, such model is often used in the urban scale models (e.g. Balogh et al., 2012; Blocken et al., 2012; Vranckx et al., 2015). However, it is unable to capture the thermal or stratification effects that may be encountered in such situations.

2.1.5 Summary

We described two systems of equations, applicable to the problems of flows in the atmospheric boundary layer. First one is derived directly from the conservation laws, and consists of the continuity equation, the momentum equations and the energy equation,

$$\frac{\partial \rho}{\partial t} + \text{div } (\rho \boldsymbol{u}) = 0,$$

$$\frac{\partial (\rho \boldsymbol{u})}{\partial t} + \text{div } (\rho \boldsymbol{u} \otimes \boldsymbol{u}) = -\nabla p + \text{div } \boldsymbol{\tau} + \rho \boldsymbol{f},$$

$$\frac{\partial (\rho E)}{\partial t} + \text{div } (\rho E \boldsymbol{u}) = -\text{div } (p \boldsymbol{u}) + \text{div } (\boldsymbol{\tau} \boldsymbol{u}) + \text{div } (k_L \nabla T) + \rho q + \rho \boldsymbol{f} \cdot \boldsymbol{u}. \tag{2.38}$$

In the following text, we will use the name *compressible flow model* when referring to this system.

The second system is built on the assumption of incompressibility of the flow and on the Boussinesq approximation. The equations read

$$\text{div } \boldsymbol{u} = 0,$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \text{div } (\boldsymbol{u} \otimes \boldsymbol{u}) = -\frac{1}{\rho_{\text{ref}}} \nabla p^* + \text{div } (\nu \nabla \boldsymbol{u}) + \boldsymbol{f}^B,$$

$$\frac{\partial \theta}{\partial t} + \text{div } (\theta \boldsymbol{u}) = \text{div } \left(\frac{k_L}{\rho_{\text{ref}} c_p} \nabla \theta \right) + \frac{q}{c_p}. \tag{2.39}$$

We will call this system the *ABL flow model*.

Both systems are capable of capturing the rich dynamic of flows in the atmospheric boundary layer. This includes stratification and thermal effects, that may play significant role in the mesoscale flows affected by the topography and in the microscale urban flows.

Vector form of the equations

In the following text, a vector form of the equations will occasionally be more suitable to work with. Both systems (2.38) and (2.39) can be written as

$$A\frac{\partial \boldsymbol{W}}{\partial t} + \sum_{j=1}^{3}\frac{\partial \boldsymbol{F}_j(\boldsymbol{W})}{\partial x_j} = \sum_{j=1}^{3}\frac{\partial \boldsymbol{R}_j(\boldsymbol{W},\nabla\boldsymbol{W})}{\partial x_j} + \boldsymbol{Q}(\boldsymbol{W}), \qquad (2.40)$$

Here $\boldsymbol{W}$ is the state vector, $\boldsymbol{F}_j(\boldsymbol{W})$ are the inviscid fluxes, $\boldsymbol{R}_j(\boldsymbol{W},\nabla\boldsymbol{W})$ are the viscous fluxes, and $\boldsymbol{Q}(\boldsymbol{W})$ are the sources and sinks. In the compressible flow equations, these have the form

$$\boldsymbol{W} = (\rho, \rho u_1, \rho u_2, \rho u_3, \rho E)^T$$
$$\boldsymbol{F}_j = (\rho u_j, \rho u_j u_1 + \delta_{1j}p, \rho u_j u_2 + \delta_{2j}p, \rho u_j u_3 + \delta_{3j}p, \rho u_j(E + p/\rho))^T \qquad (2.41)$$
$$\boldsymbol{R}_j = (0, \tau_{1j}^E, \tau_{2j}^E, \tau_{3j}^E, \tau_{j1}^E u_1 + \tau_{j2}^E u_2 + \tau_{j3}^E u_3 - pu_j + (k_L + \mu_T c_p/Pr_T)(\partial T/\partial x_j))^T$$
$$\boldsymbol{Q} = (0, \rho f_1, \rho f_2, \rho f_3, \rho q + \rho \boldsymbol{f}\cdot\boldsymbol{u})^T.$$

In the ABL flow model the terms are

$$\boldsymbol{W} = (p^*, u_1, u_2, u_3, \theta)^T$$
$$\boldsymbol{F}_j = (u_j, u_j u_1 + \delta_{1j}p^*/\rho_{\text{ref}}, u_j u_2 + \delta_{2j}p^*/\rho_{\text{ref}}, u_j u_3 + \delta_{3j}p^*/\rho_{\text{ref}}, u_j\theta)^T \qquad (2.42)$$
$$\boldsymbol{R}_j = (0, \nu_E(\partial u_1/\partial x_j), \nu_E(\partial u_2/\partial x_j), \nu_E(\partial u_3/\partial x_j), (k_L/\rho_{\text{ref}}c_p + \nu_T/Pr_T)(\partial\theta/\partial x_j))^T$$
$$\boldsymbol{Q} = (0, f_1^B, f_2^B, f_3^B, q/c_p)^T.$$

Matrix $\boldsymbol{A}$ is an identity matrix in the compressible flow equations, and

$$\boldsymbol{A} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \qquad (2.43)$$

in the ABL equations, where it eliminates the pressure from the non divergence constraint. The vector forms of these and of other systems of equations used in this work are summarized in Appendix B.

2.2 Passive scalar equation

One of the important intended applications of the atmospheric flow solver is a modelling of a pollutant dispersion. This pollutant might be a leaked harmful gas, smoke from a residential heating, or a dust raised by the road traffic. The numerical simulation of the dispersion is significantly simplified if the pollutant can be modelled as *passive scalar*. This is possible under several conditions: the pollutant should be present only in small

amount so that the physical properties of the air are not altered. It should be chemically inactive and cause no significant thermodynamical effects, such as water vapour undergoing the phase transition to liquid form. If that is the case, the pollutant dispersion may be described by the passive scalar equation written in terms of the pollutant mass concentration c (in SI units given in $\mathrm{kg\,m^{-3}}$). The equation can be derived from the mass conservation law of the pollutant, which states that the change of the pollutant mass in a control volume is equal to the amount of pollutant that enters the volume via advection and diffusion, plus the amount of pollutant emitted to (or removed from) the control volume by the pollutant sources (or sinks). Its form is

$$\frac{\partial c}{\partial t} + \mathrm{div}\,(c\boldsymbol{u}) + \mathrm{div}\,(c\boldsymbol{u}_s) = \mathrm{div}\,(k_c \nabla c) + S_c. \qquad (2.44)$$

In this equation, the term $\mathrm{div}\,(c\boldsymbol{u}_s)$ represents the gravitational settling of the particles heavier than air, and is discussed in the following subsection. Coefficient k_c is the molecular diffusion coefficient of the pollutant, and S_c stands for the volume sources and sinks of the pollutant (given in $\mathrm{kg\,m^{-3}\,s^{-1}}$). The source term S_c may include physical sources and sinks of the material, such as pollutant emissions or sinks due to the deposition on the vegetation, discussed in Sec. 2.5.2.

2.2.1 Gravitational settling of the particles

Particles of a higher density than the surrounding air settle down due to the gravitational force acting on them. This is modelled by the term

$$\mathrm{div}\,(c\boldsymbol{u_s}) = -u_s \frac{\partial c}{\partial z}. \qquad (2.45)$$

in Eq. (2.44). Here $\boldsymbol{u}_s = (0, 0, -u_s)$ is the gravitational settling velocity vector oriented towards the ground. The settling velocity u_s of the particle of the diameter d_p and density ρ_p is given by the Stokes' equation,

$$u_s = (d_p^2 \rho_p g C_C)/(18\mu), \qquad (2.46)$$

where

$$C_C = 1 + 2\frac{\lambda}{d_p}\left(1.257 + 0.4\exp\left(-1.1\frac{d_p}{2\lambda}\right)\right) \qquad (2.47)$$

is the Cunningham correction factor and $\lambda = 0.066\,\mathrm{\mu m}$ is the mean free path of the particle in the air (Seinfeld and Pandis, 2006).

2.3 Turbulence modelling

In the atmospheric boundary layer the Reynolds number is above 10^7, and the flow is turbulent everywhere. This has to be reflected in the modelling techniques. Three approaches to modelling turbulent flow are generally used:

Direct Numerical Simulation (DNS) aims at simulation of the turbulent flow in its entirety, i.e. resolving all scales. Due to the range of the spatial scales of the turbulent flow, number of grid point needed in DNS scales as $Re^{9/4}$, which constitutes major difficulty from the computational performance viewpoint. DNS is however sometimes used for investigating the properties of the atmosphere under special conditions (Sun et al., 2015).

Large Eddy Simulation (LES) is based on the idea of resolving only the large eddies and modelling the small ones using a *subgrid scale model*. LES is able to capture the unsteady turbulent dynamics at significantly lower costs than DNS. It is used for many problems of atmospheric flows, including wind farm aerodynamics (Mehta et al., 2014), urban simulations (Blocken et al., 2011), pollutant dispersion (Fuka and Brechler, 2012), or vegetation flow (Dupont et al., 2011). Its computational costs are however still much higher than of the RANS approach.

Reynolds averaged Navier-Stokes (RANS) is the approach adopted in this work. It is based on the idea of a decomposition of the flow variables into their averaged and fluctuating components, which leads to the equations of the averaged variables that are then solved. It offers low costs compared to LES, but it is incapable of capturing the unsteady turbulent dynamics and provides only averaged quantities as a result. Its accuracy depends to a large extent on a turbulence model used.

In the following section we outline the idea of RANS approach and present the final form of the equations. Further details of can be found e.g. in (Wilcox, 1993) or (Blazek, 2001).

2.3.1 Reynolds and Favre averaging

The main idea of the RANS approach is to simulate the behaviour of the averaged quantities instead of their instantaneous values. The temporal *Reynolds average* of the quantity $a(\boldsymbol{x}, t)$ is defined as

$$\overline{a}(\boldsymbol{x}, t) = \frac{1}{T} \int_t^{t+T} a(\boldsymbol{x}, t') dt', \tag{2.48}$$

where the size of the time window T is much larger than the temporal scale of the turbulent processes T_1, but much smaller than the temporal scale of the mean flow variations T_2, i.e. $T_1 \ll T \ll T2$. The quantity can be then decomposed into the mean and the fluctuations,

$$a(\boldsymbol{x}, t) = \overline{a}(\boldsymbol{x}, t) + a'(\boldsymbol{x}, t). \tag{2.49}$$

When the density is present in the equations, it is often practical to use the *Favre* (or *mass*) *averaging*,

$$\widetilde{a}(\boldsymbol{x}, t) = \frac{1}{\overline{\rho}(\boldsymbol{x}, t)\, T} \int_t^{t+T} \rho(\boldsymbol{x}, t')\, a(\boldsymbol{x}, t') dt', \tag{2.50}$$

where $\overline{\rho}$ is the Reynolds averaged density. The decomposition then reads

$$a(\boldsymbol{x}, t) = \widetilde{a}(\boldsymbol{x}, t) + a''(\boldsymbol{x}, t). \tag{2.51}$$

Using the Reynolds averaging on the density and pressure and the Favre averaging on the other variables present in the equations, the compressible flow equations (2.38) can be transformed to their averaged form. For clarity, here we write the equations using the Einstein notation.

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{u}_j) = 0$$

$$\frac{\partial(\overline{\rho}\widetilde{u}_i)}{\partial t} + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{u}_j\widetilde{u}_i) = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}(\widetilde{\tau}_{ji} - \overline{\rho\widetilde{u_i''u_j''}}) + \overline{\rho}f_i, \qquad i = 1, 2, 3$$

$$\frac{\partial(\overline{\rho}\widetilde{E})}{\partial t} + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{E}\widetilde{u}_j) = -\frac{\partial}{\partial x_j}(\widetilde{p}\widetilde{u}_j) + \frac{\partial}{\partial x_j}\left(\widetilde{u}_i(\widetilde{\tau}_{ij} - \overline{\rho\widetilde{u_i''u_j''}})\right)$$

$$+ \frac{\partial}{\partial x_j}\left(k_L\frac{\partial \widetilde{T}}{\partial x_j} - \overline{\rho\widetilde{u_j''h''}}\right) + \overline{\rho}q + \overline{\rho}f_j\widetilde{u}_j \qquad (2.52)$$

Applying Reynolds averaging to the Boussinesq system of equations (2.39) yields

$$\frac{\partial \overline{u}_j}{\partial x_j} = 0$$

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\overline{u_ju_i}) = -\frac{1}{\rho_{\text{ref}}}\frac{\partial \overline{p^*}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\nu\frac{\partial \overline{u}_i}{\partial x_j} - \overline{u_i'u_j'}\right) + f_i^B, \qquad i = 1, 2, 3$$

$$\frac{\partial \overline{\theta}}{\partial t} + \frac{\partial}{\partial x_j}(\overline{\theta}\overline{u}_j) = \frac{\partial}{\partial x_j}\left(\frac{k_L}{\rho_{\text{ref}}c_p}\frac{\partial \overline{\theta}}{\partial x_j} - \overline{\theta'u_j'}\right) + \frac{q}{c_p} \qquad (2.53)$$

These set of equations are called the Reynolds averaged Navier-Stokes (RANS) equations. Strictly speaking, first set should be called Favre-Reynolds averaged equations, we will however use only the term RANS equations for simplicity.

2.3.2 Eddy viscosity hypothesis

The averaged equations (2.52) and (2.53) are, apart from the averaging operators, similar in form to the original equations (2.38) and (2.39). They however feature some additional correlations between the velocity fluctuations and the correlations between the velocity fluctuations and enthalpy or potential temperature fluctuations.

First of these terms is modelled using the *Boussinesq* (or *eddy viscosity*) *hypothesis* (do not confuse with the Boussinesq approximation, sec. 2.1.3). The hypothesis assume that there is a linear dependence between the Reynolds stresses $\boldsymbol{\tau}^R = (\tau_{ij}^R)_{i,j=1}^3$ and the mean rate of strain, specifically

$$\tau_{ij}^R = -\overline{\rho\widetilde{u_i''u_j''}} = 2\mu_T\widetilde{D}_{ij} - \left(\frac{2\mu_T}{3}\right)\frac{\partial \widetilde{u}_k}{\partial x_k}\delta_{ij} - \frac{2}{3}\overline{\rho}\widetilde{k}\delta_{ij} \qquad (2.54)$$

for the Favre averaged equations and

$$\tau_{ij}^R = -\rho_0\overline{u_i'u_j'} = 2\mu_T\overline{D}_{ij} - \frac{2}{3}\rho_0\overline{k}\delta_{ij} \qquad (2.55)$$

for the Reynolds averaged equations. Here $k = 1/2\,(\widetilde{u_i'u_i'})$ is the turbulent kinetic energy and μ_T is the so called *turbulent* (or *eddy*) *viscosity*. We further use an approximation

suitable for low Mach number flows, which is neglecting the last term on the right hand side of equations (2.54) and (2.55).

Similar assumption is made for the turbulent heat transfer vector, which is assumed to be proportional to the temperature gradient. We can write it in the following form,

$$\widetilde{\overline{\rho u_j'' h''}} = -\frac{\mu_T c_p}{Pr_T} \frac{\partial \widetilde{T}}{\partial x_j} \tag{2.56}$$

and

$$\rho_0 \overline{u_j' \theta'} = -\frac{\mu_T}{Pr_T} \frac{\partial \overline{\theta}}{\partial x_j}, \tag{2.57}$$

where Pr_T is the turbulent Prandtl number, usually taken as $Pr_T = 0.9$.

Naming $\mu_E = \mu + \mu_T$ as the effective dynamic viscosity and

$$\boldsymbol{\tau}^E = (\tau_{i,j}^E)_{i,j=1}^3 = \left(-\frac{2}{3}\mu_E \operatorname{div} \boldsymbol{u}\right)\boldsymbol{I} + 2\mu_E \boldsymbol{D} \tag{2.58}$$

as the effective stress tensor, Eq. (2.52) can be rewritten to

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{u_j}) = 0,$$

$$\frac{\partial(\overline{\rho}\widetilde{u_i})}{\partial t} + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{u_j}\widetilde{u_i}) = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial \widetilde{\tau_{ji}^E}}{\partial x_j} + \overline{\rho}f_i, \qquad i = 1, 2, 3$$

$$\frac{\partial(\overline{\rho}\widetilde{E})}{\partial t} + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{E}\widetilde{u_j}) = -\frac{\partial}{\partial x_j}(\widetilde{p}\widetilde{u_j}) + \frac{\partial}{\partial x_j}\left(\widetilde{u_i}\widetilde{\tau^E}_{ij}\right) + \frac{\partial}{\partial x_j}\left((k_L + \mu_T c_p/Pr_T)\frac{\partial \widetilde{T}}{\partial x_j}\right)$$
$$+ \overline{\rho}q + \overline{\rho}f_j\widetilde{u_j}, \tag{2.59}$$

which has the same form as (2.38) except for the averaging operators and the replacement of the viscosity and heat conduction coefficient by their effective counterparts. Similarly, Eq. (2.53) can be casted in the form

$$\frac{\partial \overline{u_j}}{\partial x_j} = 0,$$

$$\frac{\partial \overline{u_i}}{\partial t} + \frac{\partial}{\partial x_j}(\overline{u_j u_i}) = -\frac{1}{\rho_{\text{ref}}}\frac{\partial \overline{p^*}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\nu_E \frac{\partial \overline{u_i}}{\partial x_j}\right) + \overline{f_i^B}, \qquad i = 1, 2, 3$$

$$\frac{\partial \overline{\theta}}{\partial t} + \frac{\partial}{\partial x_j}(\overline{\theta u_j}) = \frac{\partial}{\partial x_j}\left((k_L/\rho_{\text{ref}}c_p + \nu_T/Pr_T)\frac{\partial \overline{\theta}}{\partial x_j}\right) + \frac{q}{c_p}, \tag{2.60}$$

analogous to Eq. (2.39). Here $\nu_E = \nu + \nu_T$ is the effective kinematic viscosity. What now remains to determine is the turbulent viscosity. In the RANS approach this problem is dealt with by a turbulence model. We describe the implemented turbulence models in Sec. 2.3.4.

We must note that the Boussinesq hypothesis used here have its limits and is not always suitable for atmospheric flows simulations. Garratt (1992, chap. 2) states that the assumption is suitable when the flow is dominated by small eddies, e.g. in neutral or stably stratified atmosphere, but fails if the flow is dominated by large eddies, e.g. in the highly convective unstable ABL. In such cases, it is necessary to use other approaches, such as the higher order RANS models or even LES.

2.3.3 Passive scalar equation

The same process consisting of the averaging and using the eddy viscosity hypothesis can be used also for the passive scalar equation (2.44). After averaging we obtain

$$\frac{\partial \overline{c}}{\partial t} + \frac{\partial}{\partial x_j}(\overline{cu_j}) - \frac{\partial}{\partial x_3}(\overline{c}u_s) = \frac{\partial}{\partial x_j}\left(k_c \frac{\partial \overline{c}}{\partial x_j} - \overline{c'u_j'}\right) + S_c. \tag{2.61}$$

The unknown turbulence fluxes are modelled using the eddy viscosity hypothesis,

$$-\overline{c'u_j'} = \frac{\nu_T}{Sc_T}\frac{\partial \overline{c}}{\partial x_j}, \tag{2.62}$$

where Sc_T is the turbulent Schmidt number. It their review, Tominaga and Stathopoulos (2007) noted that its choice has large influence on the accuracy of the results. However, the optimal value depends on the problem geometry and flow characteristics, and is distributed over the interval $[0.2, 1.3]$ for different problems of atmospheric flows. The choice should therefore be made on a case by case basis.

Since the turbulent diffusivity in the atmospheric flows is typically much larger than the molecular diffusivity, $k_c \ll \frac{\nu_T}{Sc_T}$, the molecular diffusivity can be neglected. Putting it together, we obtain the final form,

$$\frac{\partial \overline{c}}{\partial t} + \frac{\partial}{\partial x_j}(\overline{cu_j}) - \frac{\partial}{\partial x_3}(\overline{c}u_s) = \frac{\partial}{\partial x_j}\left(\frac{\nu_T}{Sc_T}\frac{\partial \overline{c}}{\partial x_j}\right) + S_c. \tag{2.63}$$

2.3.4 Turbulence models

Two turbulence models are implemented in the solver and referenced in the following sections: algebraic mixing length model, and standard k-ϵ model.

Please note that here and in the remaining text we will drop the averaging operators to make the equations easier to read. It will however be the averaged variables we will generally refer to.

Mixing length model

The mixing length model is a conceptually simple and computationally effective model. It is an algebraic model, that is, one does not need to solve any partial differential equation, and the turbulent viscosity is obtained from an algebraic relation. Unlike the standard k-ϵ model presented below, it limits the length scale in the upper parts of the ABL. Among its disadvantages its its incapability of capturing the behaviour in flows where separation occurs, or its assumption that the mixing length depends only on the geometry and not on the flow field or its history. Therefore, its applicability for ABL flows is limited, especially for complex geometries of urban flow problems.

The model employs the mixing length hypothesis of Ludwig Prandtl. The hypothesis postulates that in a turbulent flow a parcel of fluid retains its momentum for a distance l_m, called the *mixing length*, before it dissolves in the neighbouring fluid. Under this

hypothesis the turbulent viscosity in the ABL can be written as

$$\mu_T = \rho l_m^2 \left[\left(\frac{\partial u}{\partial z} \right)^2 + \left(\frac{\partial v}{\partial z} \right)^2 \right]^{1/2}. \tag{2.64}$$

The mixing length itself is specified using the model proposed by (Blackadar, 1962) (but see also (Estoque and Bhumralkar, 1969), where the model is extended by the stability effects, here neglected). The model gives the following formula for the mixing length:

$$l_m = \frac{\kappa(z + z_0)}{1 + \frac{\kappa(z+z_0)}{l_\infty}}, \tag{2.65}$$

where κ is the Von Kármán constant, usually set to value $\kappa \sim 0.41$, z is the vertical coordinate, z_0 is the roughness length (discussed more in Sec. 2.4.2 below), and

$$l_\infty = 0.00027 \frac{G}{f_c} \tag{2.66}$$

is the limit on the mixing length in the upper part of the boundary layer, where G is the geostrophic wind speed, and f_c the Coriolis parameter (2.14).

Standard k-ϵ model

Two equation turbulence models are commonly used in the engineering applications and they are often used in the atmospheric research as well. They model the transport of two turbulence related variables, which in the case of the k-ϵ model are the turbulent kinetic energy, $k = \frac{1}{2}(\overline{u_i' u_i'})$, and its dissipation ϵ. The standard k-ϵ model used here was proposed by Launder and Spalding (1974).

We chose to implement and use the model for the following reasons:

- it is still a relatively simple model, both from the implementation viewpoint as well as from the viewpoint of the computational performance,

- it is widely used in the literature,

- it was shown to perform adequately in the range of subproblems of atmospheric flows: urban flow (Tominaga and Stathopoulos, 2013), street canyon models (Koutsourakis et al., 2012) or flow over complex terrain (dos Santos et al., 2009; Abdi and Bitsuamlak, 2014). The model is generally shown to perform comparably to the related RNG k-ϵ model, although it poorly represents the separation flow near the upwind corner of the buildings (Tominaga and Stathopoulos, 2013).

The transport equations for both variables consist of the time derivative, advection, diffusion, its production and its dissipation:

$$\frac{\partial(\rho k)}{\partial t} + \text{div}\,(\rho k \boldsymbol{u}) = \text{div}\left(\left(\mu_L + \frac{\mu_T}{\sigma_k} \right) \nabla k \right) + P_k - \rho\epsilon, \tag{2.67}$$

$$\frac{\partial(\rho\epsilon)}{\partial t} + \text{div}\,(\rho\epsilon \boldsymbol{u}) = \text{div}\left(\left(\mu_L + \frac{\mu_T}{\sigma_\epsilon} \right) \nabla\epsilon \right) + C_{\epsilon_1} \frac{\epsilon}{k} P_k - C_{\epsilon_2} \rho \frac{\epsilon^2}{k}. \tag{2.68}$$

The production of the turbulent kinetic energy has the form

$$P_k = \tau_{ij}^R \frac{\partial u_i}{\partial x_j}.$$
(2.69)

The turbulent viscosity is coupled to the modelled variables through the relation

$$\mu_T = C_\mu \rho \frac{k^2}{\epsilon}.$$
(2.70)

Constants of the standard model are presented in Tab. 2.1. In this work we will however use a modified set of constants, presented under the title ABL flows in Tab. 2.1. This choice allows the model to correctly reproduce the horizontally homogeneous unperturbed ABL flows, and is discussed in more detail in Sec. 2.4.2.

	C_{ϵ_1}	C_{ϵ_2}	C_μ	σ_k	σ_ϵ
Standard k-ϵ	1.44	1.92	0.09	1.0	1.3
ABL flows	1.44	1.92	0.09	1.0	1.167

Table 2.1: Constants of the k-ϵ model

Since the turbulent viscosity in the ABL is typically much larger than the laminar viscosity, $\mu_T \gg \mu_L$, the latter could be usually left out of the equations (2.67) and (2.68) without introducing any significant error.

Two important aspects of the stratified atmospheric boundary layer flows are however not included in the the standard k-ϵ model as presented here and used in this work. First, it is the limit on the length scale of the turbulence eddies in the upper parts of the ABL, leading to the overprediction of the eddy viscosity compared to the observations. This effect can be safely ignored as long as we focus on the area of our interest, which are the urban areas in the lowest part of the ABL.

Second neglected effect is more substantial, and that is the effect of buoyancy in the stratified atmosphere on the turbulence, which promotes either generation of the turbulence kinetic energy (in the unstable atmosphere) or its suppression (in the stable atmosphere). One might argue that in the geometrically complex urban areas the turbulence generation by shear overshadows the buoyancy effects, and this simplification is thus justifiable for our purpose, but this is difficult to state with certainty. There are wind tunnel experiments (Uehara et al., 2000) and numerical investigations using LES (Tomas et al., 2016) showing that the atmospheric stratification can significantly affect the flow field as well as the pollutant dispersion even in the geometrically complex urban areas. However, without dedicated numerical experiments it is difficult to say how much of these effects should be attributed to the buoyancy generated turbulence (which is missing in our k-ϵ model), and how much would be properly captured by the flow equations, which do support stratification effect in our formulation. To conclude, this remains to be an issue that should be dealt with. Possible adaptations of the k-ϵ model aimed at solving the issue is presented in (Apsley and Castro, 1997a) or (Sogachev et al., 2012).

2.4 Boundary conditions for the ABL flows

2.4.1 Thermodynamic variables

In a typical simulation of the ABL flow we assume that the vertical profiles of the thermo-dynamic variables at the inlet correspond to the atmosphere at equilibrium, i.e. satisfying the hydrostatic balance equation (2.22). The three thermodynamic variables - pressure, density and temperature - are further related through the state equation (2.16). Of these three, we can prescribe the temperature profile to fit the measured data or the considered atmospheric conditions. As noted in section 2.1.3, the temperature profile determines the static stability of the atmosphere.

Knowing the temperature profile $T(z)$ and prescribing the reference pressure at the ground level p_{ref}, we can then integrate the hydrostatic balance equation (2.22) using the density expressed from the state equation. For the linear temperature profile $T(z) = T_{\mathrm{ref}} - \Gamma z$ with the temperature at the ground T_{ref} and a constant lapse rate Γ this will give us the following formula for vertical profile of pressure (Jacobson, 2005, Sec. 2.4):

$$p(z) = \begin{cases} p_{\mathrm{ref}} \left(\frac{T_{\mathrm{ref}} - \Gamma z}{T_{\mathrm{ref}}} \right)^{\frac{g}{\Gamma R}} & \text{if } \Gamma \neq 0, \\ p_{\mathrm{ref}} \exp \left(-\frac{gz}{R T_{\mathrm{ref}}} \right) & \text{if } \Gamma = 0. \end{cases} \tag{2.71}$$

The density then may be calculated using the state equation.

2.4.2 Velocity profile

For the flow over rough surfaces the so called logarithmic wind profile law can be derived analytically and is well supported by the full scale measurements (Arya, 1999, chap. 4). It expresses the wind velocity as

$$u(z) = \frac{u_*}{\kappa} \ln \left(\frac{z + z_0}{z_0} \right), \tag{2.72}$$

where u_* is the friction velocity, κ is the von Kármán constant and z_0 is the surface roughness length. Friction velocity can be set to the measured value, while the roughness length depends on the modelled surface such as grass, bushes, trees, or buildings, and is related to the height of the obstacles. Its typical values are given in Tab. 2.2.

2.4.3 Boundary conditions for the k-ϵ model

A reasonable requirement on the model is its capability to sustain the inlet profiles inside a domain without any obstacles. Special care must be taken to achieve this horizontal homogeneity of the flow over a flat topography without any obstacles. In the ABL, the wind velocity profile is sustained by the balance of the shear stress acting at the top of the domain and the retarding shear stress acting at the ground. Reynolds stresses, modelled using the k-ϵ turbulence model, maintain this balance. Compatibility of the boundary conditions for the turbulence variables with the model and the wall functions must be considered. Such compatible inlet profiles and wall functions for the standard k-ϵ model were proposed by Richards and Hoxey, 1993, and these are described below.

Surface	z_0 [m]
Snow	10^{-3}
Uncut grass	0.05
Full-grown root crops	0.1
Tree covered	1
Low-density residential	2

Table 2.2: Surface roughness lengths for various surfaces. From (Seinfeld and Pandis, 2006).

Vertical profiles of k and ϵ

The inlet profiles are given as follows: wind velocity is prescribed by the log-wind profile (2.72), turbulent kinetic energy is set to

$$k(z) = \frac{u_*^2}{\sqrt{C_\mu}} \tag{2.73}$$

and turbulent dissipation to

$$\epsilon(z) = \frac{u_*^3}{\kappa(z + z_0)}. \tag{2.74}$$

If the flow described by these profiles is to be in equilibrium, the constants of the standard k-ϵ model has to be set such that

$$\sigma_\epsilon = \frac{\kappa^2}{(C_{\epsilon_2} - C_{\epsilon_1})\sqrt{C_\mu}}. \tag{2.75}$$

This precludes the use of the of usually employed constants. The constants satisfying this relation are listed in Tab. 2.1.

It shall be noted that this formulation of the inlet profiles allows only a constant turbulent kinetic energy profile, which may not be satisfactory when measured values are available and indicating different profile. To overcome this, Gorlé et al. (2009) adapted the constants of the k-ϵ model so that the use of a turbulence kinetic energy profile of a form $k(z) = \sqrt{A \ln(z + z_0) + B}$ is possible. Furthermore, Parente et al. (2011) introduced changes to the turbulence model to accommodate arbitrary turbulence kinetic energy profiles. None of these improvements are however implemented in our solver yet.

Wall functions

To capture the behaviour of the flow near the solid walls, one would need restrictively small cells at the near-ground layer. To avoid that, wall functions are often used. They are based on the applicability of the law of the wall, which states that the flow in near wall region is divided into the viscous sublayer and the logarithmic region, in which the flow behaves according to the universal properties of the near wall flow. Our code implements the wall functions in the following way. In the near ground cell with the center at height

z_p, the flow is retarded by the wall shear stress τ_w^R. The wall shear stress is calculated using the velocity and TKE in the near-wall cell (Apsley, 1995),

$$\tau_w^R = \frac{\kappa C_\mu^{0.25} k^{0.5} U}{\ln\left(\frac{z_p + z_0}{z_0}\right)}. \tag{2.76}$$

This term is added to the momentum equations of the near-wall cells as a source term $\tau_w^R(A/V)$, where A is the face area and V is the volume of the cell. With the wall shear stress already added, the boundary condition for velocity is then implemented as a slip wall.

The turbulent kinetic energy production and the dissipation are prescribed following (Richards and Hoxey, 1993) in the formulation by Parente et al. (2011):

$$P_{k,w} = \frac{(\tau_w^R)^2}{\rho \kappa C_\mu^{0.25} k^{0.5}(z_p + z_0)}, \tag{2.77}$$

$$\epsilon_w = \frac{C_\mu^{0.75} k^{1.5}}{\kappa(z_p + z_0)}, \tag{2.78}$$

As the dissipation ϵ is prescribed, Eq. (2.68) is not solved in the near-ground cells. The boundary condition for TKE is set as $k = 0$ at the wall.

2.5 Models of the vegetation

2.5.1 Fluid flow and turbulence

The models of the vegetation canopy flow have to deal with a number of associated difficulties. First, the structure of the vegetation is notably complex: it consist of thick trunks, number of branches and myriad of small elements, such as leaves or needles. All of these are mingled together in an inhomogeneous three dimensional mixture of varying density. Small elements are flexible and can change their orientation and shape under the varying wind speed. The flow through the vegetation is typically fully turbulent.

To make things even more complicated, the options for an experimental investigation of the canopy flow are limited. Only small plants or small plant parts can be investigated in a wind tunnel in real scale. Scaled models either represent a vegetation without any foliage, or some small scale foliage replacement have to be carefully designed. Appropriateness of such replacement may be questioned. Outdoor campaigns, on the other hand, are expensive and challenging due to the impossibility of controlling the atmospheric conditions.

To obtain a model useful for the desired purpose, the level of the vegetation parameterization has to be carefully considered. Most applications focused on the urban-scale vegetation flows employ the averaging approach originally developed by Wilson and Shaw (1977). Under this approach, individual vegetation elements are not resolved and the canopy is modelled as a horizontally homogenous porous block via additional terms in the momentum and turbulence equations. Details of this approach, used in this work, will be given in the following text.

While more detailed modelling approaches are useful for validating the parameterization in the coarser models, they are generally unsuitable for efficient solvers of the complex urban flow problems. We will mention two examples of alternative approaches. Endalew et al. (2009) simulated the airflow through the 3D canopy resolved to the level of branches, and left only the leaves parameterized as a porous volumes. Looking at much smaller scale, a model of a flow around a single leaf aimed at investigation of the stomatal transpiration was presented in (Defraeye et al., 2014).

Description of the vegetation

Under the employed assumption of the horizontally homogeneous vegetation, the vegetation is described by its *leaf area density* (LAD) profile. LAD profile (given in $\mathrm{m^2\,m^{-3}}$ or simply $\mathrm{m^{-1}}$) is defined as a total one-sided leaf area per unit volume and can vary with the height of the vegetation block. Its integral over the height of the canopy specifies the *leaf area index* (LAI),

$$\mathrm{LAI} = \int_0^h \mathrm{LAD}(z)dz, \tag{2.79}$$

which, in other words, is the ratio of the leaf area per ground area. Both LAI and LAD profile can be measured by the direct (destructive) methods, or indirect optical methods. When only the LAI is known, once can use the analytical formulas that were devised to reconstruct the LAD profile. One such empirical relation is given in (Lalic and Mihailovic, 2004). They propose

$$\mathrm{LAD}(z) = L_m \left(\frac{h - z_m}{h - z} \right)^n \exp\left(n \left(1 - \frac{h - z_m}{h - z} \right) \right), \tag{2.80}$$

$$n = \begin{cases} 6 & \text{if } 0 \leq z < z_m, \\ 0.5 & \text{if } z_m \leq z \leq h, \end{cases}$$

where h is the height of the vegetation, z_m is the height where the LAD attain its maximum, and L_m is the maximal value of LAD. This value is chosen so that LAI fits the known value. The authors have shown that this profile can reasonably approximate the measured profiles of various full grown trees, both deciduous and conifers.

k-ϵ vegetation model

Wilson and Shaw (1977) summarized the effects of the vegetation on the air flow in four points:

1. It extracts the momentum from the mean flow due to the aerodynamic drag of the vegetation elements.

2. The extracted energy is converted to the turbulence kinetic energy in the wakes formed behind the obstructions.

3. The energy of the large-scale turbulent motions is transformed into smaller scale turbulent motions, enhancing the turbulent dissipation in the canopy.

4. The TKE production is increased due to the heat transfer between the plant surface and the air.

The last phenomenom is often neglected as having little effect, however, all other mechanisms should be reflected in the vegetation model.

In the following description, we will for simplicity assume that the ABL flow equations (2.39) are used. All models can however be formulated also for the compressible flow model (2.38). When using the k-ϵ turbulence model, the above mentioned effects may be modelled through the additional source terms in the momentum equations (2.34) and k-ϵ equations (2.67) and (2.68) (Katul et al., 2004),

$$\left[\frac{\partial \boldsymbol{u}}{\partial t}\right]_{veg} = \boldsymbol{S}_u, \quad \left[\frac{\partial \rho k}{\partial t}\right]_{veg} = \rho S_k, \quad \left[\frac{\partial \rho \epsilon}{\partial t}\right]_{veg} = \rho S_\epsilon. \tag{2.81}$$

The momentum sink caused by the form drag is given by

$$\boldsymbol{S}_u = -(C_d \text{LAD}\, U)\boldsymbol{u}, \tag{2.82}$$

where C_d is the drag coefficient. Typical values are $0.1 \leq C_d \leq 0.5$ (Katul et al., 2004; Endalew et al., 2009). Viscous drag is considered negligible relative to the form drag, and is not included in the momentum sink. The source term in the TKE equation reads as

$$S_k = C_d \text{LAD}(\beta_p U^3 - \beta_d U k). \tag{2.83}$$

The positive part of the term represents the energy converted from the mean flow kinetic energy to the turbulence kinetic energy, and parameter β_p is the fraction of the converted energy. The negative part reflects the short-circuiting of the Kolgomorov cascade. Finally, the dissipation term was formulated as

$$S_\epsilon = C_d \text{LAD}\left(C_{\epsilon_4} \beta_p \frac{\epsilon}{k} U^3 - C_{\epsilon_5} \beta_d U \epsilon\right) \tag{2.84}$$

based on the dimensional analysis.

Several models fitting this general form were proposed (Svensson and Häggkvist, 1990; Green, 1992; Liu et al., 1996; Katul et al., 2004), differing in the terms included as well as in the choice of the constants. Kenjereš and ter Kuile (2013) compared four such models together with a momentum-only model (i.e. $S_k = S_\epsilon = 0$) and concluded that the model described by Katul et al. (2004) ($\beta_p = 1.0, \beta_d = 5.1, C_{\epsilon_4} = C_{\epsilon_5} = 0.9$) provides the best agreement to the experimental measurements and is sufficiently numerically stable. With the choice $C_{\epsilon_4} = C_{\epsilon_5}$ the Eq. (2.84) can be rewritten to simpler

$$S_\epsilon = C_{\epsilon_4} \frac{\epsilon}{k} S_k. \tag{2.85}$$

2.5.2 Dry deposition

The aerodynamic blocking of the flow is not the only effect of the vegetation. Due to the mechanism of the dry deposition, green canopies serve as a sink of the atmospheric aerosols. This phenomenom attracts increasing research interest, especially in relation to

the problem of reducing urban air pollution (Litschke and Kuttler, 2008; Janhäll, 2015). It is the urban environment and its inhabitants that suffer most from the increased particulate matter concentration. Among the most important sources of the particulate matter pollution is the road traffic, industrial processes, or residential burning.

Filtration properties of the vegetation are investigated using the experimental as well as the numerical approaches. Modelling of the dry deposition is a complicated matter, as it is a complex process depending on the physical and chemical properties of the aerosol, micrometeorological conditions, or vegetation surface properties. Petroff et al. (2008a) identifies four main mechanisms of the dry deposition:

- *Brownian diffusion*, affecting mainly small particles with diameter $d_p < 0.1\,\mu$m.

- *Interception*, occurring when a particle following the streamline passes too close to the obstacle and gets captured on it.

- *Impaction*, i.e. a collision of the particle which does not follow the streamline with the obstacle due to the inertia of the particle. Impaction is further differentiated into the *inertial* and the *turbulent impaction* by its cause.

- *Sedimentation*, which stands for the collision of the particle with the obstacle due to the downward motion of the particle caused by the gravitational force. Sedimentation is the dominant process for particles sizes $d_p > 10\,\mu$m.

Other processes are also mentioned by some authors, such as thermophoresis, caused by the temperature gradients, or electrophoresis, caused by the electric charge (Litschke and Kuttler, 2008), however, they are usually neglected in the dry deposition models due to their small effect.

The collection efficiency of the vegetation is often described through the *deposition velocity* u_d (usually given in $\mathrm{cm\,s^{-1}}$). Deposition velocity is defined as a ratio of the mass flow rate towards the leaf surface and the particulate matter concentration in the atmosphere. The dry deposition may then be modelled as a sink term in the passive scalar equation (2.44),

$$\left[\frac{\partial c}{\partial t}\right]_{veg} = -\mathrm{LAD}u_d c. \tag{2.86}$$

In (Litschke and Kuttler, 2008) the authors reviewed published values of the measured deposition velocities for different plant species, wind speeds, and particles sizes. They concluded that the values differ by up to four orders of magnitude, with most of them concentrated in the range $[0.01, 10]\,\mathrm{cm\,s^{-1}}$.

Detailed deposition velocity models have a long history in the regional air quality models (Slinn, 1982; Zhang et al., 2001; Petroff et al., 2008b; Petroff et al., 2009; Petroff and Zhang, 2010). In the regional models the flow through the vegetation is not explicitly resolved, and the deposition velocity is calculated at some height above the canopy. In this framework the resistance formulation of (Slinn, 1982) is often used,

$$u_d(z) = u_s + \frac{1}{R_A(z) + R_S}, \tag{2.87}$$

where u_s is the gravitational settling velocity, R_A is the aerodynamic resistance and R_S is the surface layer resistance. The aerodynamic resistance reflects the turbulent transport of the particles between the height z and the canopy, and the surface layer resistance models the deposition processes named above, i.e. diffusion, interception and impaction. Sedimentation is acting in parallel via the settling velocity u_s.

In the microscale CFD models the flow through the vegetation is modelled directly, and the aerodynamic resistance R_A can be left out. The surface resistance R_S is typically derived under some simplified assumptions on the flow through the canopy, which is in the regional models often represented by a single parameter: the friction velocity above the canopy u_*. Such models are unlikely to represent well the deposition velocity in small vegetation patches, typical for urban vegetation, as opposed to the continuous forest cover.

In (Petroff et al., 2008b) the authors developed a an alternative approach consisting of solving a 1D aerosol transport equation inside the canopy, which allowed detailed specification of the canopy morphology. The method was originally designed to deal with the vegetation consisting of cylindrical, needle-like obstacles, and was later extended to planar obstacles such as broadleaves (Petroff et al., 2009). In their method, the authors provided expressions for the deposition velocities associated with each of the underlying processes acting inside the canopy. [1] These expressions are also suitable for the use in the CFD models, and will be described here.

The model assumes that the physical mechanisms of the dry deposition act independently, and the deposition velocity can thus be written as a sum of the deposition velocities of all processes,

$$u_d = 2(u_{BD} + u_{IN} + u_{IM} + u_{TI} + u_{SE}), \tag{2.88}$$

where the subscripts of the velocities on the right hand side stand for the Brownian diffusion, interception, inertial impaction, turbulent impaction, and sedimentation respectively. The factor 2 is included because the deposition velocities of the underlying processes are expressed for two-sided LAD in the original formulation, whereas here we use the one-sided LAD.

The deposition velocities associated with each process are summarized in Tab. 2.3. Formulas presented here are derived for the canopies with the leaf (or needles) of equal size, although other leaf size distributions are also considered in the original papers. In the table, d_p is the particle diameter, ρ_p is the particle density, and d_e is the dimension of the vegetation element (i.e. needle diameter or leaf width).

The contribution of the Brownian diffusion is expressed in terms of the Schmidt and Reynolds numbers, $Sc = \nu/D_B$ and $Re = Ud_e/\nu$, where $D_B = (C_C k_b T)/(3\pi\mu d_p)$ is the Brownian diffusion coefficient, $k_b = 1.380\,648\,52 \times 10^{-23}\,\mathrm{J\,K^{-1}}$ is the Boltzmann constant and

$$C_C = 1 + 2\frac{\lambda}{d_p}\left(1.257 + 0.4\exp\left(-1.1\frac{d_p}{2\lambda}\right)\right)$$

is the Cunningham correction factor with $\lambda = 0.066\,\mu\mathrm{m}$ being the mean free path of the particle in the air. The interception deposition velocity depends on the ratio of the particle

---

[1]Note that in the original article the term *collection velocity* was used for these deposition velocities to distinguish them from the deposition velocity acting above the vegetation canopy.

	Needles	Broadleaves
u_{BD}	$UC_B Sc^{-2/3} Re^{n_B-1}$	
u_{IN}	$2Uk_x\frac{d_p}{d_e}$	$\frac{1}{2}Uk_x\frac{d_p}{d_e}\left(2+\ln(\frac{4d_e}{d_p})\right)$
u_{IM}	$Uk_x E_{IM}$	
u_{TI}	$u_f K_{TI1}\tau_p^{+2} \quad$ if $\tau_p^+ < 20$	
	$u_f K_{TI2} \quad$ if $\tau_p^+ \geq 20$	
u_{SE}	$k_z g\rho_p C_C d_p^2/(18\mu_a)$	

Table 2.3: Deposition velocities expressions given in (Petroff et al., 2008b; Petroff et al., 2009)

size and the vegetation element size d_p/d_e, and differs for the needle and broadleaf elements. The deposition velocity associated with the impaction is calculated from the impaction efficiency,

$$E_{IM} = \left(\frac{St}{St+\beta}\right)^2$$

where $St = \tau_p U/d_e$ is the Stokes number, and $\tau_p = \frac{\rho_p C_c d_p^2}{18\mu}$ is the particle relaxation time. The turbulent impaction contribution is given in terms of the nondimensionalized particle relaxation time, $\tau_p^+ = \tau_p u_f^2/\nu$, and the local friction velocity u_f (where $u_f^2 = -\overline{u_x' u_z'}$ in the coordinate system rotated so that the x-axis is aligned with the horizontal flow direction).

Coefficients of the model are given in Tab. 2.4. Constants C_B and n_B are derived

	Needles	Broadleaves
C_B	0.467	0.664
n_B	0.5	
β	0.6	0.47
K_{TI1}	$3.5 \cdot 10^{-4}$	
K_{TI2}	0.18	

Table 2.4: Deposition velocity model coefficients

for a laminar boundary layer around the vegetation elements, expected in the denser canopies. Parameters k_x and k_y stand for the ratios of the leaf surface projected on the plane perpendicular to the flow direction and on the horizontal plane respectively to the total leaf surface. Authors of the original papers derived these coefficients based on the angular distribution of the leaves in the canopy. Their values are presented in Tab. 2.5.

Fig. 2.1 shows the deposition velocity as a function of the particle size and demonstrates its dependency on the wind speed and element size. In general, the deposition

	Needles		Broadleaves	
	k_x	k_z	k_x	k_z
Horizontal	0.20	0.32	0.00	0.50
Planophile	0.24	0.27	0.14	0.43
Plagiophile	0.27	0.22	0.22	0.34
Erectophile	0.30	0.13	0.27	0.21
Vertical	0.32	0.00	0.32	0.00
Extremophile	0.26	0.19	0.19	0.30
Uniform	0.27	0.20	0.20	0.32

Table 2.5: Projection coefficients k_x and k_z according to (Petroff et al., 2008b; Petroff et al., 2009). Planophile refers to leaf orientation distribution with horizontal leaves most frequent, plagiophile to most frequent oblique leaves, erectophile to most frequent vertical leaves, extremophile to oblique leaves least frequent, uniform to all leaves equally frequent.

velocity is minimal for particle sizes between $0.1\,\mu m$ and $1\,\mu m$. Below that, the Brownian diffusion is dominant, above that, all other processes play significant role with the sedimentation eventually becoming the driving force behind the deposition.

Several effect of possible importance are not captured by this model. Real leaves may have a limited capacity to capture the aerosol particles, which is decreasing when the leaf gets dirty, and which is possibly refreshed by a cleaning effect of a rainfall. This effect is not included in the model, since no theoretical models or experimental measurements of this phenomena are known to the author. Changing orientation of the leaves due to the fluid-structure interaction is also not modelled nor parametrized, but it may be taken into account by changing the leaf angle distribution parameters.
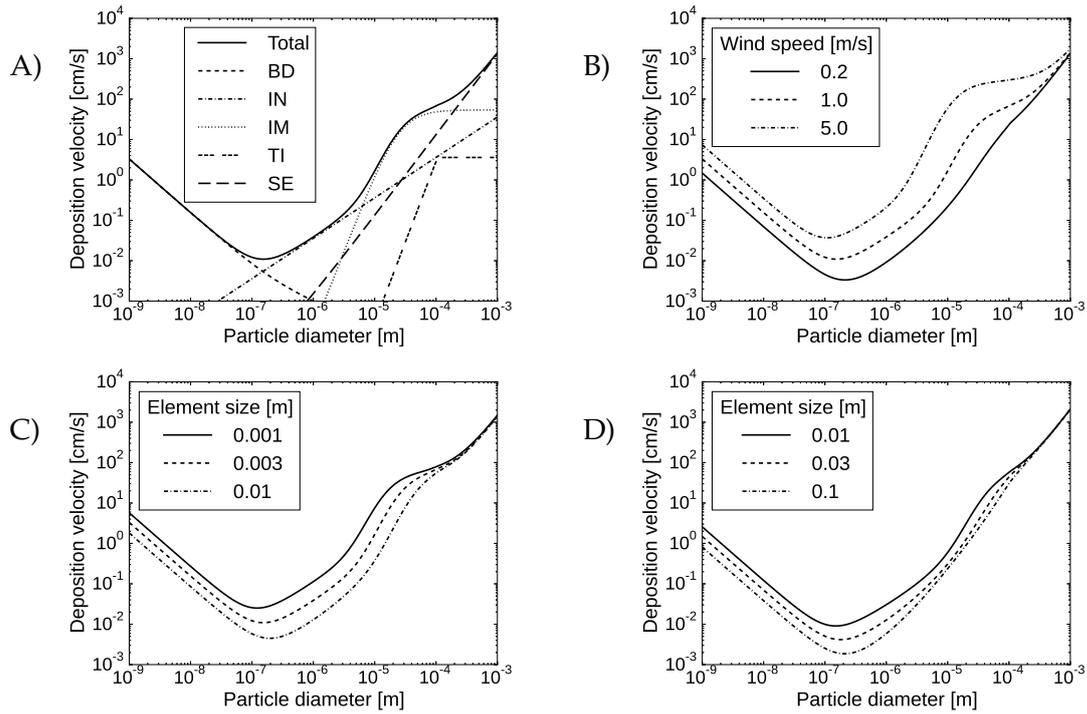
Figure 2.1: Examples of the deposition velocity dependence on particle size. (A) Contributions of all processes. Parameters of the model: needle-like elements, $\rho_p = 1000\,\mathrm{kg\,m^{-3}}, d_e = 3\,\mathrm{mm}, U = 1\,\mathrm{m\,s^{-1}}, u_f = 0.1\,\mathrm{m\,s^{-1}}$, plagiophile leaf distribution. Shortcuts refer to those in Eq. (2.88). (B) Dependence on the wind speed. Parameters as in panel A, except for the varying U. (C) Dependence on the vegetation element size for needle-like elements. Parameters as in panel A, except for the varying d_e. (D) Dependence on the vegetation element size for broadleaf elements. Parameters as in panel A, except for the element type and the varying d_e.

Chapter 3

Numerical methods

The content of this chapter is an expanded version of the numerical methods description in: Šíp, V. and Beneš, L. (2016d). RANS solver for microscale pollution dispersion problems in areas with vegetation: Development and validation. *arXiv e-print*. https://arxiv.org/abs/1609.03427 (Submitted).

In this chapter, we describe the numerical methods used for implementation of the mathematical models discussed in the previous chapter. The main characteristics of the developed solver are following:

- The solver is based on a finite volume method on an unstructured grids.

  Computational domains for the urban flow simulations can be very complex. As such, the usage of the structured grids would be very cumbersome, if not impossible.

- Fully implicit temporal discretization.

  Explicit time stepping schemes would introduce a major constraint on the length of the time step. Implicit schemes as implemented here allows more efficient implementation.

- Two implemented fluid flow solvers based on the compressible flow equations and ABL flow equations.

  Both sets of equations described previously are implemented. The method of artificial compressibility is used for the pressure resolution when ABL flow equations are used.

The chapter is structured as follows: In Sec. 3.1 we describe the artificial compressibility method for ABL flow equations, and in Sec. 3.2 we discuss the related low Mach preconditioning for the compressible flow equations. Spatial and temporal discretization of the modified equations are detailed in Sections 3.3 and 3.4 respectively. Section 3.5 elaborates on how are the ingredients put together to make the computational solver.

3.1 Artificial compressibility

Let us start with the question of the pressure resolution. The system of the compressible flow equations (2.38) consists of five equations, from which the density, velocity components and total energy can be calculated. From these, using the state equation, the pressure can be calculated. That is not the case for the system of ABL equations (2.39). While the velocity components and potential temperature can be calculated, the pressure is coupled to the rest through the momentum equations and the non-divergence constraint,

$$\operatorname{div} \boldsymbol{u} = 0.$$

Two main approaches are in use for this or similar system. First is the class of the *pressure-based* or methods, prominently represented by the SIMPLE method of Patankar and Spalding (1972). These methods solve the Poisson equation for pressure corrections, and use this correction to obtain a divergence-free velocity field. Second approach is the so-called *artificial compressibility* proposed by Chorin (1967). With this approach, an additional pressure derivative is added to the non-divergence constraint, thus obtaining a hyperbolic system for which the conceptually simple time stepping methods may be applied. It is this simplicity that was the main reason for our choice to use the artificial compressibility method in our solver. The details of the method are given here.

In our formulation, a time derivative of the pressure fluctuation is added to the non-divergence constraint,

$$\frac{1}{\beta} \frac{\partial p^*}{\partial t} + \operatorname{div} \boldsymbol{u} = 0 \tag{3.1}$$

where $\beta > 0$ is the artificial compressibility parameter. Physically, this means that the pressure waves with finite artificial speed of sound are introduced in the system. This way, a prognostic equation for the pressure (or in our case for the pressure fluctuation) appears. The choice of the artificial compressibility parameter β does not affect the steady-state solution, but it can have strong impact on the convergence properties. Some authors recommended for $\sqrt{\beta}$ to be equal to some representative convective velocity, however, the optimal value is generally case dependent (Muldoon and Acharya, 2007).

In a vector form, the equations now read

$$\boldsymbol{\Gamma} \frac{\partial \boldsymbol{W}}{\partial t} + \sum_{j=1}^{3} \frac{\partial \boldsymbol{F}_j}{\partial x_j} = \sum_{j=1}^{3} \frac{\partial \boldsymbol{R}_j}{\partial x_j} + \boldsymbol{Q}, \tag{3.2}$$

where

$$\boldsymbol{\Gamma} = \begin{pmatrix} 1/\beta & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{3.3}$$

Recall that the state vector of the ABL equations is $\boldsymbol{W} = (p^*, u_1, u_2, u_3, \theta)^T$.

It was noted by Turkel (1985) that the resulting preconditioned equations are not symmetrizable. He proposed a generalized formulation, which (if extended for the potential temperature equation) reads as

$$\boldsymbol{\Gamma} = \begin{pmatrix} 1/\beta & 0 & 0 & 0 & 0 \\ \alpha u_1/\beta & 1 & 0 & 0 & 0 \\ \alpha u_2/\beta & 0 & 1 & 0 & 0 \\ \alpha u_3/\beta & 0 & 0 & 1 & 0 \\ \alpha\theta/\beta & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{3.4}$$

For $\alpha = 0$ we obtain the original formulation, while a choice $\alpha = 1$ results in the symmetrizable system. This is the value we will use in the further work, unless noted otherwise.

This modification obviously changes the solution to the equations. However, the modified system of equations can be used directly if we are interested only in the steady-state solutions. For such solutions the time derivative vanishes and we are left with the non-divergence constraint satisfied again. Unsteady solutions have to be dealt with via the pseudo time stepping. A preconditioned time derivative in a pseudo time τ is added to the original equations,

$$\boldsymbol{\Gamma}\frac{\partial \boldsymbol{W}}{\partial \tau} + \boldsymbol{A}\frac{\partial \boldsymbol{W}}{\partial t} + \sum_{j=1}^{3}\frac{\partial \boldsymbol{F}_j}{\partial x_j} = \sum_{j=1}^{3}\frac{\partial \boldsymbol{R}_j}{\partial x_j} + \boldsymbol{Q}. \tag{3.5}$$

Recall that the matrix $\boldsymbol{A}$ given by Eq. (2.43) is an identity matrix with the empty first row. In every physical time step, we advance the solution in the pseudo time until the pseudo time derivative $\partial \boldsymbol{W}/\partial \tau$ vanishes. Then we are left with the solution to the original equations.

In the developed solver, the generalized artificial compressibility for steady state problems is implemented in the following way. We note that

$$\boldsymbol{\Gamma}\frac{\partial \boldsymbol{W}}{\partial t} = \begin{pmatrix} \frac{1}{\beta}\frac{\partial p^*}{\partial t} \\ u_1\frac{1}{\beta}\frac{\partial p^*}{\partial t} + \frac{\partial u_1}{\partial t} \\ u_2\frac{1}{\beta}\frac{\partial p^*}{\partial t} + \frac{\partial u_2}{\partial t} \\ u_3\frac{1}{\beta}\frac{\partial p^*}{\partial t} + \frac{\partial u_3}{\partial t} \\ \theta\frac{1}{\beta}\frac{\partial p^*}{\partial t} + \frac{\partial \theta}{\partial t} \end{pmatrix} \tag{3.6}$$

and that from Eq. (3.1) $\frac{1}{\beta}\frac{\partial p^*}{\partial t} = -\operatorname{div}\boldsymbol{u}$, so that the equations can be rewritten as

$$\frac{\partial \boldsymbol{W}}{\partial t} + \sum_{j=1}^{3}\frac{\partial \widetilde{\boldsymbol{F}}_j}{\partial x_j} = \sum_{j=1}^{3}\frac{\partial \boldsymbol{R}_j}{\partial x_j} + \widetilde{\boldsymbol{Q}}, \tag{3.7}$$

where

$$\widetilde{\boldsymbol{F}}_j = \begin{pmatrix} \beta u_j, \\ u_j u_1 + \delta_{1j} p^*/\rho_{\text{ref}}, \\ u_j u_2 + \delta_{2j} p^*/\rho_{\text{ref}}, \\ u_j u_3 + \delta_{3j} p^*/\rho_{\text{ref}}, \\ u_j \theta \end{pmatrix}, \tag{3.8}$$

i.e. the first term is multiplied by β, and

$$\widetilde{\boldsymbol{Q}} = \boldsymbol{Q} + \begin{pmatrix} 0 \\ u_1 \operatorname{div} \boldsymbol{u} \\ u_2 \operatorname{div} \boldsymbol{u} \\ u_3 \operatorname{div} \boldsymbol{u} \\ \theta \operatorname{div} \boldsymbol{u} \end{pmatrix}. \tag{3.9}$$

With these equations we use the timestepping schemes as described in Sec. 3.4 for the state vector $\boldsymbol{W} = (p^*, u_1, u_2, u_3, \theta)^T$. For the unsteady problems we follow the same route, and we end up with a similarly modified system

$$\frac{\partial \boldsymbol{W}}{\partial \tau} + \boldsymbol{A} \frac{\partial \boldsymbol{W}}{\partial t} + \sum_{j=1}^{3} \frac{\partial \widetilde{\boldsymbol{F}}_j}{\partial x_j} = \sum_{j=1}^{3} \frac{\partial \boldsymbol{R}_j}{\partial x_j} + \widetilde{\boldsymbol{Q}}. \tag{3.10}$$

3.2 Low Mach preconditioning

Applying the methods designed for compressible flows to low Mach number flows - such as the flows in the atmospheric boundary layer - is notoriously difficult. Main problem here stems from the disparity of the acoustic wave speed, $a+U$, and the convective speed, U. Maximal timestep is limited by the former (due to the CFL condition when using explicit schemes or due to the stiffness of the equations when using the implicit schemes), but the system mainly evolves on the timescale given by the latter. This results in a slow convergence of the computational solvers. Preconditioning schemes try to remove this disparity by multiplying the time derivative by a matrix designed to equalize the acoustic and convective speeds. In other words, the matrix $\boldsymbol{A}$ in the vector form of the equations (2.40), equal to identity matrix in the non-preconditioned form, is replaced by a preconditioning matrix $\boldsymbol{\Gamma}$. In the actual implementation we work with its inverse, so the equations can be written as

$$\frac{\partial \boldsymbol{W}}{\partial t} + \boldsymbol{\Gamma}^{-1} \sum_{j=1}^{3} \frac{\partial \boldsymbol{F}_j}{\partial x_j} = \boldsymbol{\Gamma}^{-1} \sum_{j=1}^{3} \frac{\partial \boldsymbol{R}_j}{\partial x_j} + \boldsymbol{\Gamma}^{-1} \boldsymbol{Q}. \tag{3.11}$$

Variety of different formulations of the preconditioning matrix were proposed. Turkel (1987) extended his incompressible flow preconditioning discussed in section 3.1 also to

the low speed compressible flows. Van Leer et al. (1991) devised an optimal precon-
ditioner in a sense that it achieves the lowest attainable ratio of maximal and minimal
wave speeds. Choi and Merkle (1993) and Weiss and Smith (1995) derived their precon-
ditioners from the Navier-Stokes equations expressed in terms of the primitive variables
(p, u, v, w, T). Weiss-Smith preconditioner was found to be most robust (Colin et al., 2011;
Jafari, 2014), and it is the one implemented in the software. If the Navier-Stokes equations
are written in terms of conservative variables, the preconditioning matrix inverse reads
(Turkel, 1999)

$$\mathbf{\Gamma^{-1}} = \boldsymbol{I} - \frac{(1-\epsilon)(\gamma-1)}{a^2} \begin{pmatrix} \frac{U^2}{2} & -u_1 & -u_2 & -u_3 & 1 \\ u_1\frac{U^2}{2} & -u_1^2 & -u_1u_2 & -u_1u_3 & u_1 \\ u_2\frac{U^2}{2} & -u_1u_2 & -u_2^2 & -u_2u_3 & u_2 \\ u_3\frac{U^2}{2} & -u_1u_3 & -u_2u_3 & -u_3^2 & u_3 \\ h\frac{U^2}{2} & -u_1h & -u_2h & -u_3h & h \end{pmatrix}, \tag{3.12}$$

where $h = E + \frac{p}{\rho}$ is the enthalpy, γ is the specific heat ratio, and ϵ is the preconditioning
parameter. The choice of this parameter is discussed in (Turkel, 1999). It should be
proportional to the square of the Mach number M^2 for the preconditioner to be efficient,
however, to avoid singularity at $M = 0$ in the original formulation, ϵ should be limited
from below. Furthermore, it is desirable that the preconditioning is disabled for the
supersonic flow. The parameter is therefore set as

$$\epsilon = \min(1, \max(KM_{\text{ref}}^2, M^2)), \tag{3.13}$$

where $K = 0.5$ and M_{ref} is some reference Mach number, e.g. the free stream Mach
number.

As is the case with the artificial compressibility approach for ABL flow equations,
the low Mach preconditioning changes the pseudo-time behaviour of the system, and as
such, it is in the described form useful only for steady state problems. For the unsteady
problems, dual time stepping (Eq. (3.5)) would have to be employed. However, this
is not yet implemented in the software, and for the unsteady problem presented in this
work (Sec. 4.1) the non-preconditioned set of equations is used, since the computational
performance was satisfactory in that case.

3.3 Spatial discretization

As discussed in the introduction, the problems of the flow in urban areas often bring the
need to deal with geometrically complex geometries, for which a structured grid is hard
or impossible to construct. The finite volume method, which we employ in the solver,
allows us to use the unstructured grids, and it is thus a natural choice for these problems.

The systems of RANS equations of the compressible flow (2.59) and of the flow in the
ABL (2.60), as well as the equations of the k-ϵ model (2.67), (2.68) and the passive scalar
equation (2.44) can all be rewritten into the following form:

$$\frac{\partial \boldsymbol{W}}{\partial t} + \sum_{j=1}^{3} \frac{\partial \boldsymbol{F}_j(\boldsymbol{W})}{\partial x_j} = \sum_{j=1}^{3} \frac{\partial \boldsymbol{R}_j(\boldsymbol{W}, \nabla\boldsymbol{W})}{\partial x_j} + \boldsymbol{Q}(\boldsymbol{W}). \tag{3.14}$$

Here $\boldsymbol{W}$ is the state vector, $\boldsymbol{F}_j(\boldsymbol{W})$ are the inviscid fluxes (including the pressure contribution), $\boldsymbol{R}_j(\boldsymbol{W}, \nabla \boldsymbol{W})$ are the viscous fluxes, and $\boldsymbol{Q}(\boldsymbol{W})$ are the sources and sinks. The full form of all terms for the mentioned systems of equations is given in Appendix B.

The equations are spatially discretized using the cell centered finite volume method (see e.g. Blazek, 2001; LeVeque, 2004). The computational domain is divided into a finite number of non-overlapping volumes V_i. We allow the computational cells to be of any shape, with the restriction that the faces of the cells have to be either triangular or quadrilateral. The number of faces of any cell is not restricted. The restriction on the shape of the faces is present because of the current implementation of the calculation of the gradient on the cell faces, described in Sec. 3.3.2.

We can then integrate over the control volumes, and with the use of the divergence theorem, the equations for each volume can be written as

$$\frac{\partial \boldsymbol{W}_i}{\partial t} = \frac{1}{|V_i|} \left(-\sum_{j=1}^{3} \int_{\partial V_i} (\boldsymbol{F}_j n_j - \boldsymbol{R}_j n_j) dS + \int_{V_i} \boldsymbol{Q} dx \right), \quad (3.15)$$

where $\boldsymbol{W}_i$ is the mean value of $\boldsymbol{W}$ in cell V_i, $|V_i|$ is the volume of the cell, ∂V_i is its boundary surface, and $\boldsymbol{n} = (n_1, n_2, n_3)$ is the surface normal oriented outwards.

As in (Blazek, 2001), we assume that the fluxes are constant at every individual face, and we evaluate them at the centers of the faces. Sources are assumed to be constant in each cell, and they are evaluated at the centers of the cells. Then ,if the volume V_i is enclosed by N_f faces marked as f_s with the area $|f_s|$ and normals $\boldsymbol{n}_{f_s}$, the equation can be rewritten to

$$\frac{\partial \boldsymbol{W}_i}{\partial t} = \frac{1}{|V_i|} \left(-\sum_{j=1}^{3} \sum_{s=1}^{N_f} (\boldsymbol{F}_{f_s,j} n_{f_s,j} |f_s| - \boldsymbol{R}_{f_s,j} n_{f_s,j} |f_s|) \right) + \boldsymbol{Q}_i, \quad (3.16)$$

where $\boldsymbol{F}_{f_s,j}$ and $\boldsymbol{R}_{f_s,j}$ are the inviscid and viscous fluxes at the face f_s.

3.3.1   Inviscid fluxes

Numerical flux AUSM$^+$-up

The numerical flux AUSM$^+$-up is used both for the compressible flow and ABL equations. The flux belongs to the AUSM (Advection Upstream Splitting Method) family of numerical fluxes. At the core of the method lies the idea that the inviscid fluxes consists of two parts: convective and pressure fluxes, first of which is convected by the velocity, while the second is governed by the acoustic wave speed. Principle of upwinding based on the underlying physical process is employed for each part separately. The original version named AUSM (Liou and Steffen, 1993) and later improved version AUSM$^+$ (Liou, 1996) were designed for transsonic and supersonic flows, and suffered of a loss of accuracy and reduced convergence rate at lower Mach numbers. This was improved by the AUSM$^+$-up version (Liou, 2006), which introduced rescaling of the terms so that the speed of sound and convective speed are of the same order. Resulting formulation is robust and efficient for all speeds, therefore suitable also for low Mach atmospheric flows.

Here we provide a brief overview of the flux, for details we refer to the original paper. The implementation of the flux differs for the two sets of equations we use. We first describe the flux as it is used in the compressible flow formulation (for which it was originally formulated), and then comment on the changes needed when the ABL equations are used.

Compressible flow equations. Following the notation in the original paper, let us denote the inviscid flux over the face f by $\boldsymbol{F}_{1/2} = \sum_{j=1}^{3} \boldsymbol{F}_{f_s,j} n_{f_s,j}$ and the velocity normal to the face by $u = \boldsymbol{u} \cdot \boldsymbol{n}_f$. The inviscid term on each face is calculated from the state vector on the left side of the face $\boldsymbol{W}_L$ and on the right side of the face $\boldsymbol{W}_R$. The sides are marked so that the right cell is in the direction of the face normal $\boldsymbol{n}_f$. The orientation of this normal, however, is set arbitrarily, except on the boundary, where it is outwards oriented.

The basic idea of the approach is to split the inviscid flux into two parts, namely into the convective and the pressure flux,

$$\boldsymbol{F}_{1/2} = \boldsymbol{F}^c_{1/2} + \boldsymbol{P}_{1/2}. \tag{3.17}$$

The convective flux is expressed using the oriented mass flow $\dot{m}_{1/2}$ through the face f,

$$\boldsymbol{F}^c_{1/2} = \dot{m}_{1/2} \boldsymbol{\psi}_{L/R}, \tag{3.18}$$

where the advected vector $\boldsymbol{\psi} = (1, u_1, u_2, u_3, h)^T$ is determined by an upwind rule,

$$\boldsymbol{\psi}_{L/R} = \begin{cases} \boldsymbol{\psi}_L & \text{if } \dot{m}_{1/2} \geq 0, \\ \boldsymbol{\psi}_R & \text{if } \dot{m}_{1/2} < 0. \end{cases} \tag{3.19}$$

The mass flow is given as

$$\dot{m}_{1/2} = a_{1/2} M_{1/2} \begin{cases} \rho_L & \text{if } M_{1/2} \geq 0, \\ \rho_R & \text{if } M_{1/2} < 0, \end{cases} \tag{3.20}$$

where the speed of sound at the interface is calculated as an average of the left and right states, $a_{1/2} = (a_L + a_R)/2$. The interface Mach number is calculated from the left and right Mach numbers $M_{L/R} = \frac{u_{L/R}}{a_{1/2}}$ as

$$M_{1/2} = \mathcal{M}^+_{(4)}(M_L) + \mathcal{M}^-_{(4)}(M_R) + M_p. \tag{3.21}$$

The split Mach numbers $\mathcal{M}^{\pm}_{(4)}$ are polynomials of the indicated degree, defined as

$$\mathcal{M}^{\pm}_{(2)}(M) = \pm \frac{1}{4}(M \pm 1)^2 \tag{3.22}$$

$$\mathcal{M}^{\pm}_{(4)}(M) = \begin{cases} \frac{1}{2}(M \pm |M|) & \text{if } |M| \geq 1, \\ \mathcal{M}^{\pm}_{(2)}(M)(1 \mp 16\beta_a \mathcal{M}^{\mp}_{(2)}(M)) & \text{otherwise.} \end{cases} \tag{3.23}$$

The pressure diffusion term M_p is defined as

$$M_p = -\frac{K_p}{f_a} \max(1 - \sigma \overline{M}^2, 0) \frac{p_R - p_L}{\rho_{1/2} a_{1/2}^2}, \tag{3.24}$$

where $\rho_{1/2} = (\rho_L + \rho_R)/2$ and the $\overline{M}^2 = \frac{1}{2}(M_L^2 + M_R^2)$ is the mean local Mach number. The factor f_a is given as $f_a = M_o(2 - M_o)$, where $M_o^2 = \min(1, \max(\overline{M}^2, M_\infty^2))$, and M_∞ is the reference Mach number.

The pressure flux is given as

$$\boldsymbol{P}_{1/2} = (0, \; n_{f,1} p_{1/2}, \; n_{f,2} p_{1/2}, \; n_{f,3} p_{1/2}, \; 0)^T, \tag{3.25}$$

where

$$p_{1/2} = \mathcal{P}_{(5)}^+(M_L) p_L + \mathcal{P}_{(5)}^-(M_R) p_R + p_u. \tag{3.26}$$

The fifth degree polynomial split functions are given as

$$\mathcal{P}_{(5)}^\pm(M) = \begin{cases} \frac{1}{M}(M \pm |M|) & \text{if } |M| \geq 1, \\ \mathcal{M}_{(2)}^\pm(M)((\pm 2 - M) \mp 16\alpha_a M \mathcal{M}_{(2)}^\mp(M)) & \text{otherwise.} \end{cases} \tag{3.27}$$

Finally, the velocity diffusion term has the form

$$p_u = -K_u \mathcal{P}_{(5)}^+(M_L) \mathcal{P}_{(5)}^-(M_R)(\rho_L + \rho_R) f_a a_{1/2}(u_R - u_L). \tag{3.28}$$

The parameters and constants of the numerical flux are set to $\alpha_a = \frac{3}{16}(-4 + 5f_a^2)$, $\beta_a = \frac{1}{8}$, $K_p = 0.25$, $K_u = 0.75$, and $\sigma = 1$.

ABL flow equations. Similar approach is taken also for the ABL flow model. As in (3.17), the flux is split into the convective flux $\boldsymbol{F}_{1/2}^c$ and the pressure flux $\boldsymbol{P}_{1/2}$. The convective flux is calculated as

$$\boldsymbol{F}_{1/2}^c = a_{1/2} M_{1/2} \boldsymbol{\psi}_{L/R}. \tag{3.29}$$

The original ABL system does not support the pressure waves, however, it does when the artificial preconditioning approach is used. These artificial waves then have the numerical speed of sound $a = \sqrt{\beta + u^2}$. The interface speed of sound is then calculated as before, $a_{1/2} = (a_L + a_R)/2$, as are the left and right Mach numbers, $M_{L/R} = \frac{u_{L/R}}{a_{1/2}}$. The advected vector is $\boldsymbol{\psi} = (\beta, u_1, u_2, u_3, \theta)^T$. The interface Mach number is calculated using Eq. (3.21), where the pressure diffusion term depends on the difference of pressure fluctuations,

$$M_p = -\frac{K_p}{f_a} \max(1 - \sigma \overline{M}^2, 0) \frac{p_R^* - p_L^*}{\rho_{\text{ref}} a_{1/2}^2} \tag{3.30}$$

with f_a and $\overline{M}$ as detailed above.

The pressure flux is given by

$$\boldsymbol{P}_f = \left(0, \; \frac{n_{f,1} p_{1/2}}{\rho_{\text{ref}}}, \; \frac{n_{f,2} p_{1/2}}{\rho_{\text{ref}}}, \; \frac{n_{f,3} p_{1/2}}{\rho_{\text{ref}}}, \; 0\right)^T, \tag{3.31}$$

and $p_{1/2}$ is calculated as

$$p_{1/2} = \mathcal{P}_{(5)}^+(M_L)p_L^* + \mathcal{P}_{(5)}^-(M_R)p_R^* + p_u \tag{3.32}$$

with the velocity diffusion term

$$p_u = -K_u \mathcal{P}_{(5)}^+(M_L)\mathcal{P}_{(5)}^-(M_R)2\rho_{\text{ref}}f_a a_{1/2}(u_R - u_L). \tag{3.33}$$

This formulation of the numerical flux was found to prevent the artificial pressure oscillations (known as "checkerboarding") that appeared when using simpler fluxes such as the central flux. However, to achieve that, the constant K_p had to be increased to the value 1.0, compared to 0.25 used for the compressible flow formulation. Still, this value falls into the allowed interval ($K_p \in [0, 1]$) given in the original paper.

Second order scheme

Using the neighbouring cell values of the state vector as $\boldsymbol{W}_L$ and $\boldsymbol{W}_R$ in the inviscid flux calculation results only in first order accuracy in space. Second order scheme is obtained using the process of linear reconstruction. Let us consider cell c with faces f_s, $s = 1, \ldots, N_f$. Value of each variable ϕ from the state vector $\boldsymbol{W}$ at face f_s is obtained by the first order Taylor expansion,

$$\phi_{f_s} = \phi_c + \psi_c (\nabla\phi)_c \cdot \boldsymbol{p}_{f_s}. \tag{3.34}$$

where the subscript $_c$ denotes the value at the center of the cell c, ψ_c is the limiting coefficient at cell c and $\boldsymbol{p}_{f_s}$ is the vector from the cell center to the center of the face f_s (see Fig. 3.1).
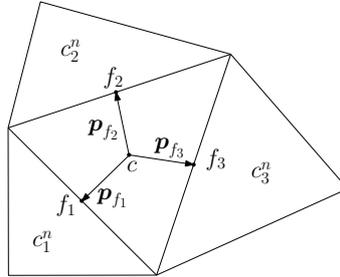
Figure 3.1: Depiction of one cell and its neighbours used in the second order reconstruction. For simplicity only 2D cells are shown here, however the algorithm is implemented for 3D meshes.

The gradient is computed using the least square reconstruction:

$$(\nabla\phi)_c = T_c \cdot \widetilde{\phi}_c, \tag{3.35}$$

where

$$\widetilde{\phi}_c = \begin{pmatrix} \phi_{c_1^n} - \phi_c \\ \phi_{c_2^n} - \phi_c \\ \vdots \\ \phi_{c_{N_f}^n} - \phi_c \end{pmatrix}, \tag{3.36}$$

41

is the vector of the differences between the values at the cell c and its neighbouring cells c_s^n (for $s = 1 \ldots, N_f$). Further,

$$T_c = (M^T M)^{-1} M^T, \tag{3.37}$$

in which

$$M = \begin{pmatrix} \Delta x_1 & \Delta y_1 & \Delta z_1 \\ \vdots & \vdots & \vdots \\ \Delta x_{N_f} & \Delta y_{N_f} & \Delta z_{N_f} \end{pmatrix}, \tag{3.38}$$

with $(\Delta x_s, \Delta y_s, \Delta z_s)$ being the distance between the centers of the cells c and c_s^n.

The limiting coefficient ψ_c is calculated using the limiter by Venkatakrishnan (1995), designed to obtain better convergence properties in the near-constant regions of the flow solution. It does so by introducing a smooth alternative to a min function and by suppressing the effects of the limiter in the regions of uniform flow.

3.3.2 Viscous fluxes

To evaluate the viscous fluxes $\boldsymbol{R}_{f_s,j}$, the gradients of the flow variables at the faces of the computational cells are needed. They are calculated using the so called *diamond cell method* (Coudière et al., 1999) in the formulation given in (Karel, 2014, Sec. 3.1), where the details of the method, here omitted for brevity, can be found. We will outline the method for quadrilateral faces, the triangular faces are treated as a limit case of a quadrilateral faces with one edge vanishing, i.e. with two vertices overlapping.

The method is as follows. For each face f (given by vertices ABCD) we construct an associated diamond cell by extruding two pyramids to the centers of gravity of the two neighbouring cells, denoted as L and R (Fig. 3.2). The gradient at the face is then
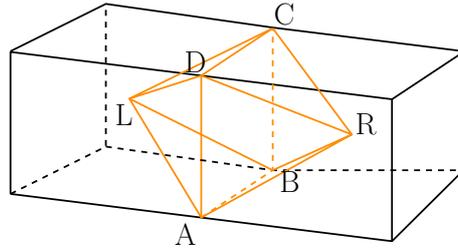
Figure 3.2: Diamond cell associated with the face ABCD. Points L and R denote the centers of gravity of the left and right cell respectively.

calculated using the Gauss-Green theorem, and the following expression is obtained:

$$(\nabla \phi)_f = \frac{1}{3|V_f|} ((\phi_A - \phi_C) \boldsymbol{n}_{BRDL} |f_{BRDL}| + (\phi_B - \phi_D) \boldsymbol{n}_{ALCR} |f_{ALCR}| + (\phi_R - \phi_L) \boldsymbol{n}_f |f|). \tag{3.39}$$

Here $\boldsymbol{n}_{BRDL} = (\boldsymbol{n}_{BRD} + \boldsymbol{n}_{DLB})/2$, $\boldsymbol{n}_{ALCR} = (\boldsymbol{n}_{ALC} + \boldsymbol{n}_{CRA})/2$ are expressed in terms of normals to the planes given by three points. Contributions are weighted by the face

areas, where $|f_{BRDL}| = |f_{BRD}| + |f_{DLB}|$ and $|f_{ALCR}| = |f_{ALC}| + |f_{CRA}|$. Term $|V_f|$ stands for the volume of the diamond cell associated with the face f. Finally, the variable values at the centers of gravity ϕ_L and ϕ_R are taken as the mean values in the left or right cells, and the values at vertices are calculated as an average of their neighbouring cells using a distance weighting procedure. We refer to (Karel, 2014) for further details.

3.3.3 Boundary conditions

The concept of *ghost cells* is applied to deal with the boundary conditions. For each cell at the boundary, a mirrored ghost cell is created outside of the computational domain. Then, every time the fluxes through the boundary are to be evaluated, the state vectors at the ghost cells are first filled with the values according to the boundary condition specified and the value on the inner side of the domain boundary. This way, the fluxes through the boundary faces can be calculated the same way as for the inner faces.

Let us describe the implementation of the often used boundary conditions. In the following paragraphs, c denote the inner cell at the boundary and c' its ghost counterpart. Outwards oriented normal of the boundary face is denoted as $\boldsymbol{n}$. The values of the variable ϕ at the inner and ghost cells are denoted as ϕ_c and $\phi_{c'}$ respectively.

Dirichlet BC, i.e. $\phi = \phi_{prescribed}$ at the boundary. The ghost cell is filled with the prescribed value, $\phi_{c'} = \phi_{prescribed}$.

Homogeneous Neumann BC, i.e. $\frac{\partial \phi}{\partial \boldsymbol{n}} = 0$ at the boundary. The value in the ghost cell is extrapolated from inside, $\phi_{c'} = \phi_c$.

In the following text we will often drop the "homogeneous" attribute, and use only "Neumann BC" when talking about this boundary condition. This should not cause any confusion, as the inhomogeneous Neumann boundary conditions (i.e. with $\frac{\partial \phi}{\partial \boldsymbol{n}} = q$ for some nonzero q) are not used in this work.

Slip wall BC represents a solid wall in the inviscid flow. There is no flow across the boundary, $\boldsymbol{u} \cdot \boldsymbol{n} = 0$, but the flow parallel to the wall is unrestricted. In this case, the velocity in the ghost cell is then calculated as $\boldsymbol{u}_{c'} = \boldsymbol{u}_c - 2(\boldsymbol{u}_c \cdot \boldsymbol{n})\boldsymbol{n}$. Assuming adiabatic wall, the density and the total energy (for the compressible flow equations) or the pressure fluctuation and the potential temperature (for the ABL flow equations) are set using the Neumann BC.

No-slip wall BC represents a solid wall in the viscous flow. The velocity of the flow is zero at the boundary, $\boldsymbol{u} = 0$, due to the friction exerted by the wall. The velocity in the ghost cell is calculated as $\boldsymbol{u}_{c'} = -\boldsymbol{u}_c$. If the wall is adiabatic, the density and the total energy or the pressure fluctuation and the potential temperature are as in the slip wall case set using the Neumann BC.

Wall functions used for turbulence modelling were described in Sec. 2.4.3. If the wall functions formulation prescribes a value in the inner cell, the value in the ghost cell is irrelevant, as the equations are not solved for the inner cell. This is in the implementation achieved by zeroing all fluxes to the inner cell, as well as the volume sources in the inner cell.

3.4 Temporal discretization

3.4.1 Time stepping in physical time

Unless the time stepping in pseudo time is employed, the problem is after the spatial discretization reduced to a system of ordinary differential equations (ODE) in the form

$$\frac{\partial \boldsymbol{Y}}{\partial t} = \boldsymbol{G}(\boldsymbol{Y}, t), \tag{3.40}$$

where the global state vector $\boldsymbol{Y} = (\boldsymbol{W}_1^T, \boldsymbol{W}_2^T, \dots, \boldsymbol{W}_N^T)$ is a concatenation of the state vectors $\boldsymbol{W}_i$ in all cells. The system is complemented by a given initial condition $\boldsymbol{Y}(t = 0)$.

Two general classes of numerical method exist for the solution of such ODEs in discrete time. The *explicit methods* calculate the state in the next time step directly from the current state (and possible some past states). Their computational demands for one time step are low, however, this is offset by the need to use small time steps lengths to maintain numerical stability, especially for stiff systems. Their overall computational demands may thus be prohibitively high. The *implicit methods* may allow arbitrarily large time step lengths, but one has to solve a system of linear or nonlinear (as in our case) equations in every time step. Nevertheless, even with this obstacle the implicit methods could be more efficient than the explicit methods. In our solver, two implicit time-stepping schemes are implemented: backward Euler method and BDF2 method.

Backward Euler method

Let $\boldsymbol{Y}_n$ stand for the discretized solution at the time t_n and let $\Delta t_n = t_{n+1} - t_n$ stand for the time step length. The implicit Euler method can be written as

$$\boldsymbol{Y}_{n+1} - \boldsymbol{Y}_n = \Delta t_n \boldsymbol{G}(\boldsymbol{Y}_{n+1}, t_{n+1}), \quad n = 0, 1, \dots \tag{3.41}$$

The method is unconditionally A-stable and of a first order of accuracy. As such, it is better suited for steady-state problems where the temporal evolution of the solution is of no interest. In that case, the timestep Δt_n is continuously increased to achieve faster convergence to steady-state solution, for which $\boldsymbol{G}(\boldsymbol{Y}, t) = 0$.

BDF2 method

Second order method from the BDF (Backward Differentiation Formula) family is better suited for unsteady problems than the first order Euler method. We use the variable step formulation (Eckert et al., 2004) to allow for time step adjustment when needed. The method is then given as follows:

$$\frac{1 + 2\alpha_n}{1 + \alpha_n} \boldsymbol{Y}_{n+1} - (1 + \alpha_n)\boldsymbol{Y}_n + \frac{\alpha_n^2}{1 + \alpha_n}\boldsymbol{Y}_{n-1} = \Delta t_n \boldsymbol{G}(\boldsymbol{Y}_{n+1}, t_{n+1}), \quad n = 0, 1, \dots \tag{3.42}$$

where $\alpha_n = \Delta t_n / \Delta t_{n-1}$. The BDF2 method is also unconditionally A-stable.

3.4.2 Time stepping in pseudo time

Things are more complicated when we deal with the solution of the system (3.10) with the added derivative in pseudo time. We write its spatially discretized form as

$$\frac{\partial \boldsymbol{Y}}{\partial \tau} + \hat{\boldsymbol{A}}\frac{\partial \boldsymbol{Y}}{\partial t} = \boldsymbol{G}(\boldsymbol{Y}, t),$$

where $\hat{\boldsymbol{A}}$ is a square matrix of the same dimensions as $\boldsymbol{Y}$ containing the repeated matrices $\boldsymbol{A}$ on its diagonal.

In this case, we discretize the derivative in physical time t using the BDF2 method to achieve high temporal precision [1] , and the derivative in pseudo time τ is discretized using simpler backward Euler method. In every physical time step, we set our initial value $\boldsymbol{Y}_{n+1,0} = \boldsymbol{Y}_n$, and then iterate using the scheme

$$\frac{\boldsymbol{Y}_{n+1,k+1} - \boldsymbol{Y}_{n+1,k}}{\Delta \tau} + \hat{\boldsymbol{A}}\frac{\frac{3}{2}\boldsymbol{Y}_{n+1,k+1} - 2\boldsymbol{Y}_n + \frac{1}{2}\boldsymbol{Y}_{n-1}}{\Delta t_n} = \boldsymbol{G}(\boldsymbol{Y}_{n+1,k+1}, t_{n+1}), \quad k = 0, 1, \dots$$
(3.43)

until the derivative in the pseudo time vanishes. After the final iteration k_{final} we set $\boldsymbol{Y}_{n+1} = \boldsymbol{Y}_{n+1,k_{final}}$, and continue with the next step in physical time.

3.5 Solution process

3.5.1 Nonlinear system solution

Every iteration of the time stepping schemes (3.41), (3.42) and (3.43) leave us with the system of nonlinear equations that needs to be solved. This problem can be reformulated to finding a vector $\boldsymbol{Y}$ such that

$$\boldsymbol{H}(\boldsymbol{Y}) = 0,$$
(3.44)

for some function $\boldsymbol{H}$ that is derived from the time stepping scheme and the function $\boldsymbol{G}$. The sought vector $\boldsymbol{Y}$ is either the value $\boldsymbol{Y}_{n+1}$ in the next physical step for the schemes (3.41), (3.42), or the value $\boldsymbol{Y}_{n+1,k+1}$ in the next step in pseudo time for scheme (3.43).

This problem is solved by the PETSc library (Balay et al., 2015) using the Jacobian-free Newton-Krylov (JFNK) method (see e.g. Knoll and Keyes, 2004). JFNK method solves the nonlinear system (3.44) through a sequence of Newton iterations, marked by the superscript index, expressed as linear systems

$$\boldsymbol{J}(\boldsymbol{Y}^m)\boldsymbol{\Delta}^m = -\boldsymbol{H}(\boldsymbol{Y}^m) \qquad m = 0, 1 \dots$$
(3.45)

where the matrix $\boldsymbol{J}(\boldsymbol{Y}) = \left(\frac{\partial (H(\boldsymbol{Y}))_i}{\partial Y_j}\right)_{i,j=1}^{N}$ is the Jacobian of the nonlinear function and vector $\boldsymbol{\Delta}^m = \boldsymbol{Y}^{m+1} - \boldsymbol{Y}^m$. The Jacobian is however generally not known and its numerical evaluation would be very costly, as we will discuss in Sec. 3.5.2 . The JFNK method overcomes this by not actually forming the Jacobian (hence the name Jacobian-free). This is made possible by solving the inner linear systems using a Krylov method,

---

[1]To simplify the following description, we will assume here that the time step in physical time is constant, however, variable time step as in Eq. (3.42) is also possible to use.

whose algorithm requires only an ability to calculate the matrix-vector product instead of the knowledge of the full matrix of the linear system in question. This matrix-vector product is provided by an approximation

$$J(Y)U \approx \frac{H(Y + \epsilon U) - H(Y)}{\epsilon}, \tag{3.46}$$

where ϵ is a small perturbation. This way, only two evaluations of H (or equivalently of the original RHS function G) are needed for the calculation of one matrix-vector product.

In our solver the GMRES method (Saad and Schultz, 1986) is employed for the solution of the inner linear systems. With the default settings, its iteration process is restarted after every 30 iteration steps.

3.5.2 Linear system preconditioning

To accelerate the solution process of the inner linear systems a preconditioner is used. The general idea of preconditioning is to replace the original linear system

$$Ax = b \tag{3.47}$$

by the system

$$PAx = Pb, \tag{3.48}$$

where the preconditioner matrix $P \approx A^{-1}$ approximates the inverse of the original matrix, but can be obtained more easily. In our solver we use the ILU(k) preconditioner (see e.g. a review by Chan and van der Vorst, 2001), which seeks an approximate LU factorization of the matrix, $LU \approx A$, such that the nonzero elements of the calculated lower and upper factors are only at the filled positions of the original matrix A in case of ILU(0) preconditioner, or, in case of ILU(k), of the matrix LU given by ILU(k-1) decomposition.

To calculate the preconditioner matrix the Jacobian is needed to be known. In general, there are several options how to evaluate the Jacobian: *analytical evaluation* would be the best option from the computational performance viewpoint, however, it is possible only for simple problems. *Automatic differentiation* allows to calculate the derivatives of any function defined by a computer program with little computational overhead, but requires specialized software tooling.

We calculate the Jacobian using the *finite differencing*. Every element of the Jacobian is given by

$$J(Y)_{ij} = \frac{H(Y + \epsilon_j)_i - H(Y)_i}{\epsilon}, \tag{3.49}$$

where ϵ_j is a zero vector with ϵ at j-th position. This procedure is however very computationally demanding. Conceptually simple approach would be calculating the columns of the Jacobian one by one. This approach applied to the computational mesh with N cells and n_{vars} state variables in each cell would result in $n_{vars}N$ evaluation of the RHS function.

To prevent that, the procedure of matrix colouring (Gebremedhin et al., 2005) as implemented in PETSc package is employed. With matrix colouring, the effects of perturbations that do not interfere with each other may be evaluated at once (see Fig. 3.3).

That way the number of RHS function evaluations needed to construct the Jacobian is independent of the mesh size. It is proportionate to $n_{neighs}n_{vars}$, where n_{neighs} is the number of cells in the computational stencil of one cell. In our case, that is the number of second order neighbour cells (i.e. neighbours of neighbours).
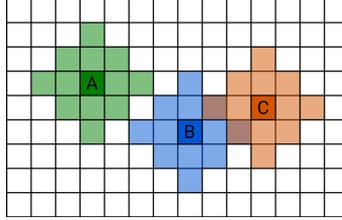
Figure 3.3: Computational stencils of three cells in 2D. The stencils of cells A and B do not overlap, and the effects of perturbation of one variable in each cell can be evaluated at once when using the matrix colouring. That is not possible for cells B and C, since two cells lie in the stencil overlap.

Furthermore, we can utilize the fact that we need the Jacobian only to compute the preconditioning matrix, that can be constructed only from an approximate Jacobian. With that in mind, we evaluate the Jacobian only once in every 20 iterations of the outer ODE solver. Such procedure is sometimes called *Jacobian lagging*, as the Jacobian used for the preconditioner calculation is lagged behind its actual value by several iterations.

The Jacobian evaluation interval was not rigorously optimized, but the used choice of 20 iterations in most cases proved to be a decent compromise between the computational demands of frequent Jacobian evaluation on one hand and increased number of Krylov iterations caused by the outdated preconditioner on the other hand.

3.5.3 Coupling of the fluid flow and turbulence solvers

With the RANS equations and the eddy viscosity assumption employed, the fluid flow solver and the turbulence solver are closely tied together. The turbulence variables are advected by the velocity field calculated by the fluid flow solver, which in turn is affected by the turbulent viscosity calculated from the turbulence variables. This close coupling provides good reason to solve all equations together as one large nonlinear system. However, to keep the resulting systems of linear equations relatively small, we solve the fluid flow equations separately from the turbulence equations. Specifically, in each time step we first solve the nonlinear system arising from the discretization of the fluid flow equations, and then (if the k-ϵ model is employed) the system of the turbulence equations using the computed values of the flow variables. On a computational mesh with N cells, this results in a sequences of two sparse linear systems, one of size $5N \times 5N$ (fluid flow equations) and one of size $2N \times 2N$ (k-ϵ equations).

The absence of an implicit coupling might result in an oscillatory behaviour in time. To prevent that, a *relaxation* of the turbulent viscosity might be employed. With relaxation, the turbulent viscosity in $n + 1$ step is calculated as

$$(\mu_T)_{n+1} = (1 - q)(\mu_T)_n + q(\tilde{\mu}_T)_{n+1}, \tag{3.50}$$

47

where $(\tilde{\mu}_T)_{n+1}$ is the value of turbulent viscosity computed by the turbulence model and $q \in [0; 1]$ is the relaxation parameter. Choice of $q = 1$ gives the unrelaxed scheme, while $q = 0$ would prevent any evolution in time. In effect, the relaxation slows down the evolution of the turbulent viscosity, making the solver more robust at the expense of the convergence rate. In the simulations later presented in this work, we generally use the unrelaxed scheme ($q = 1$). Only in the cases where it proved to be unsatisfactory due to the produced oscillations we use the relaxation.

3.5.4 Iteration process control

Default tolerances used in the program for each of the iteration processes employed are listed in Tab. 3.1. Solution of the inner linear systems solved by the GMRES solver is stopped when either the relative or the absolute L2 norm of the residual drops below the given tolerance. The same applies for the nonlinear systems solved by the JFNK method.

Parameter	Value
Linear system relative tolerance	10^{-3}
Linear system absolute tolerance	10^{-6}
Nonlinear system relative tolerance	10^{-2}
Nonlinear system absolute tolerance	10^{-6}
Target linear system solver iterations	$[50; 200]$
Target nonlinear system solver iterations	$[5; 20]$
Time step adaptation constant γ_1	1.4
After-failure reduction constant γ_2	0.25
Jacobian lag (ODE iterations)	20

Table 3.1: Default parameters of the iteration process

In the steady-state simulations we start with a small time step Δt_1, roughly on the order of one millionth of the time interval needed for the flow to cross the computational domain. The time step is continuously adapted. When both the number of the inner linear and outer nonlinear system solver iterations in one time step are below the lower end of the target interval, the length of the time step in the next iteration is increased by the factor γ_1, i.e. $\Delta t_{n+1} = \gamma_1 \Delta t_n$. Similarly, if the number of either linear or nonlinear system solver iterations exceeds the upper bound of the target interval, the time step length is decreased by the same factor, $\Delta t_{n+1} = \Delta t_n / \gamma_1$.

Due to the aggressive time step adapting policy that is constantly pushing for the largest attainable time step, it may happen that the nonlinear system solve fails. This may happen for a number of reasons, e.g. a stalled convergence of the inner linear solver or the outer nonlinear solver, or a divergence of the nonlinear solver. If that happens, the time step is reduced by a factor γ_2, the preconditioner is recomputed, and the iteration is repeated.

For the steady state simulations, the stopping criterion of the iteration process is

formulated on a case by case basis using the values of the residuals. For a state variable ϕ, its residual in time step n is defined as

$$Res_n^{\phi} = \|\boldsymbol{Y}_n^{\phi} - \boldsymbol{Y}_{n-1}^{\phi}\|_2, \tag{3.51}$$

where the state vector $\boldsymbol{Y}^{\phi}$ contains the values of the variable ϕ in all cells of the computational grid, $\boldsymbol{Y}^{\phi} = (\phi_1, \phi_2, \ldots, \phi_N)$. Recall that the state variable is one of the five variables $\rho, \rho u_1, \rho u_2, \rho u_3, \rho E$ in the compressible flow formulation, and one of $p^*, u_1, u_2, u_3, \theta$ in the ABL flow formulation.

3.5.5 Overview

The above described elements of the solution process are summarized here. In every physical time step the order of operations is as follows:

1. Every 20th iteration: Evaluate the Jacobian and compute the preconditioner.

2. One step of the fluid flow solver:

   (flow variables, $\mu_T)_n \rightarrow$ (flow variables)$_{n+1}$.

3. One step of the turbulence solver:

   (turbulence variables)$_n$, (flow variables)$_{n+1} \rightarrow$ (turbulence variables, $\mu_T)_{n+1}$.

4. Adapt the time step.

5. Evaluate the stopping criteria.

The iteration process continues until the convergence criteria are satisfied (for the steady solver) or the final time si reached (for the unsteady solver).

3.6 Solution of 2D problems

The solver is designed for three dimensional problems, but it can solve two dimensional problems just as well. In that case, we employ a pseudo 3D computational mesh, that is a 2D mesh extruded by one cell to the third dimension. The boundaries in the third dimension (i.e., the sides of the extruded domain) are then modelled as slip walls as described in Sec. 3.3.3, for which the parallel flow is unrestricted. For any passive scalar a homogeneous Neumann boundary conditions is prescribed.

This approach is more computationally demanding than a dedicated 2D solver could be, however, it allows us to use the same solver for 2D and 3D problems and thus avoid development of two separate ones, which was deemed more beneficial than the increased performance for 2D problems.

Chapter 4

Validation

In this chapter we present four test cases to validate the developed solver. First two cases (rising thermal bubble and flow around the hill) serve us to compare the accuracy and computational performance of the solvers based on the compressible flow equations and the ABL flow equations when used for the ABL flows without the vegetation. Third case - the forest canopy flow - shows whether the flow through the vegetation can be reproduced using our model. In the fourth case we asses the capability of the dry deposition model on the problem of the particle-laden flow through the hedgerow.

In the following text, we will use the shortened names "compressible solver" or "ABL solver" instead of lengthy (but more accurate) "solver based on the compressible/ABL flow equations" for brevity.

4.1   Rising thermal bubble

The first case serves to show that the solvers can properly capture the unsteady thermally driven flow in the ABL. The settings replicate the rising thermal bubble test case from (Giraldo and Restelli, 2008), which was based on the previous formulation of a similar test by Robert (1993).

4.1.1   Test case description

A bubble of hot air is placed in the atmosphere with a constant potential temperature. The air in the two dimensional domain is initially at rest, and the thermal effects force the bubble to rise through the environment. At the beginning, the unperturbed atmosphere has the potential temperature $\theta = 300\,\mathrm{K}$, and the bubble is created by increasing the potential temperature by

$$\theta^* = \frac{\theta_c}{2}\left(1 + \cos\left(\frac{\pi r}{r_c}\right)\right) \quad \text{when } r \leq r_c, \tag{4.1}$$

where $\theta_c = 0.5\,\mathrm{K}$, $r = \sqrt{(x - x_c)^2 + (z - z_c)^2}$, the center of the bubble is set to $(x_c, z_c) = (500, 350)\,\mathrm{m}$, and its diameter to $r_c = 250\,\mathrm{m}$. The initial velocity is set to zero, and the initial pressure is set using the barometric formula (2.71), so that the air is at hydrostatic balance.

The computational domain has size $[0, 1000]\,\mathrm{m} \times [0, 1000]\,\mathrm{m}$. Its boundary conditions are all set as *slip walls*, i.e. the velocity normal to the wall is set to zero, and for the velocity parallel to the wall as well as for all other variables the homogeneous Neumann BC is applied, $\frac{\partial \phi}{\partial \boldsymbol{n}} = 0$. The flow is modelled as inviscid. The employed mesh is Cartesian, with uniform resolution in both horizontal and vertical direction.

The evolution of the system is simulated for $t \in [0, 700]$ s. As we are interested in the temporal evolution, time stepping in the pseudo time (3.5) is necessary to use for the system of ABL equations. The length of one time step is set as $\Delta t = 1\,\mathrm{s}$.

4.1.2 Results

The analytical solution for the case is not known, so the results from the two developed solvers are compared with what we will call a reference solution from (Giraldo and Restelli, 2008). This reference solution was calculated by a Discontinuous Galerkin solver that uses 10th order polynomials and is based on the equations for density perturbations, momentum, and total energy perturbation. The solver is denoted by DG3 in the original paper. The reference results were obtained on the mesh with the spatial resolution of 5 m in both directions.

Mesh dependence

The dependence on the mesh resolution was assessed using the compressible solver running on four different meshes with spatial resolution of 20, 10, 5, and 2.5 m in both vertical and horizontal direction. Plot of the potential temperature perturbation at the final time is shown on Fig. 4.1, and its vertical profile at the centerline is shown on Fig. 4.2.

The vertical profile of the potential temperature perturbation reveals that even at the highest employed resolution of 2.5 m, the peak of the profile is slightly underpredicted compared to the reference solution calculated on mesh with resolution of 5 m. This should be however expected, as the reference solution was calculated using a high order Discontinous Galerkin method, compared to our second-order finite volume solver. Further refining of the computational mesh might increase the precision of our solution further, although at excessive computational cost: our finest mesh consisted of 160 thousands computational cells, halving the spatial resolution would further quadruple this number.

Comparison of the solvers

For the comparison of the two developed solvers we calculated the evolution of the bubble on the finest mesh tested above, i.e. with spatial resolution of 2.5 m.

Fig. 4.3 shows a side-by-side comparison of the potential temperature perturbation at the final time $t = 700\,\mathrm{s}$. At that time, the bubble has risen to the top of the domain, and its mushroom shape is partially affected by its interaction with the upper boundary. This
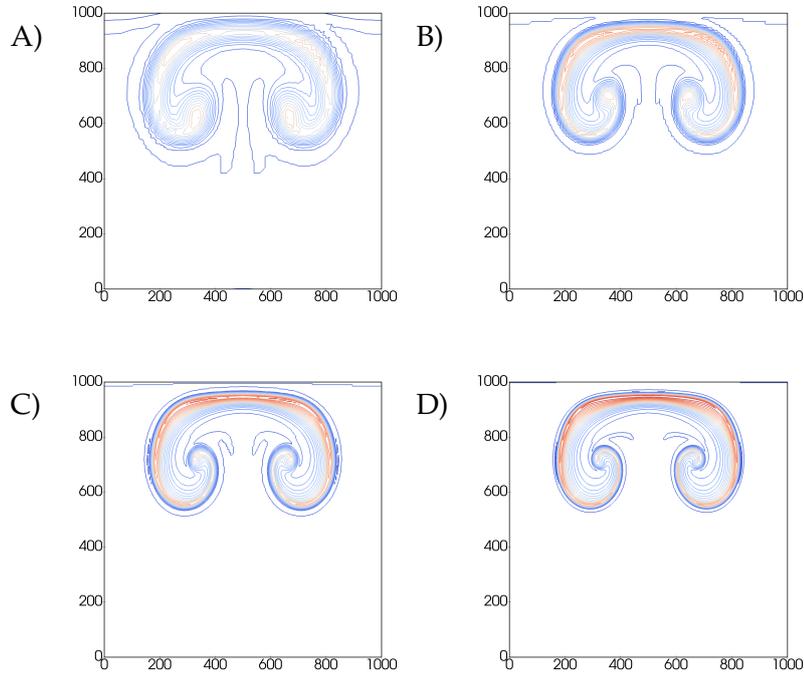
Figure 4.1: Mesh sensitivity of the warm bubble test case. Potential temperature perturbation at $t = 700\,\mathrm{s}$ calculated by the compressible solver for mesh resolutions (A) 20 m (B) 10 m (C) 5 m (D) 2.5 m. Interval between contours is 0.025 K.
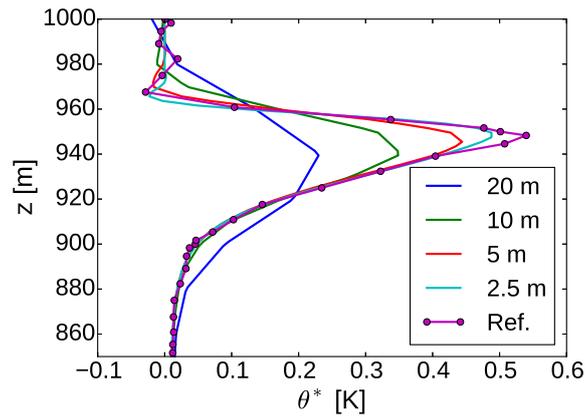
Figure 4.2: Mesh sensitivity of the warm bubble test case. Vertical profile of the potential temperature perturbation at $t = 700\,\mathrm{s}$ and $x = 500\,\mathrm{m}$ calculated by the compressible solver.
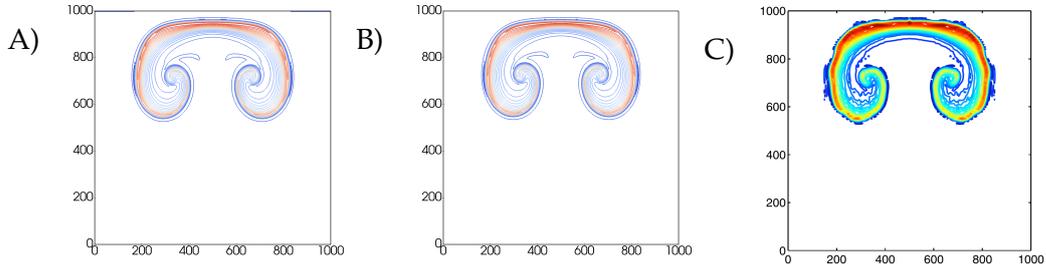
Figure 4.3: Warm bubble test case. Side-by-side comparison of the potential temperature perturbation at $t = 700\,\mathrm{s}$ obtained by the two developed solvers on mesh with the resolution of 2.5 m. (A) Compressible solver (B) ABL solver (C) Reference solution from (Giraldo and Restelli, 2008). Contour values are from -0.05 to 0.525 with an interval of 0.025.
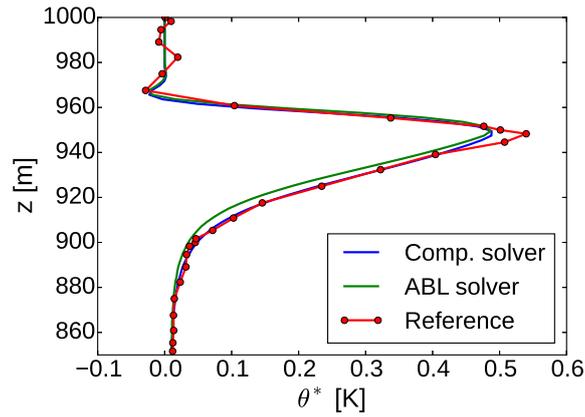
Figure 4.4: Warm bubble test case. Vertical profile of the potential temperature perturbation at $t = 700\,\mathrm{s}$ and $x = 500\,\mathrm{m}$.

	θ^* [K]		u_x [m/s]		u_z [m/s]	
	min	max	min	max	min	max
Comp. solver	-0.029	0.492	-1.941	1.941	-1.873	2.565
ABL solver	-0.029	0.491	-1.980	1.980	-1.855	2.565
Reference	-0.093	0.538	-2.081	2.081	-1.915	2.543

Table 4.1: Minima and maxima of potential temperature perturbation, horizontal velocity, and vertical velocity at time $t = 700\,\mathrm{s}$.

53

behaviour is captured well by both developed solvers, and the calculated solutions agree well with the reference one. The vertical profile of the potential temperature perturbation is shown in Fig. 4.4. The peak of the profile is underestimated by both solvers.

Furthermore, the ABL solver places the lower end of the bubble above the position given by the reference solution. This might be attributed to the simplifications made in our physical model, of which the most notable is the use of the reference density ρ_{ref} instead of the actual density ρ in the pressure term in the velocity equation (2.34). This shows that there are limits to its accuracy in the domains spanning more than few hundred meters in the vertical direction.

As a further comparison, Tab. 4.1 lists the minima and maxima of the selected variables at the final time. Both our developed solvers show very similar results. The underprediction of the maximal potential temperature perturbation, visible at Fig. 4.4, is again exhibited here. However, the overall qualitative as well as quantitative agreement of both our solutions with the reference one is demonstrated.

4.2 Flow over an isolated 2D hill

The flow over an isolated hill is among the most used testcases for the CFD solvers aimed at atmospheric boundary layer flows. In various configurations, it was often investigated through the wind tunnel experiments as well as numerical simulations. Here we present a comparison of the results obtained by our developed solvers with the experimental data from the RUSHIL wind tunnel study (Khurshudyan et al., 1981), obtained from the ERCOFTAC QNET-CFD test case database (ERCOFTAC, 2004). In addition to the flow data over 2D hill, the pollution dispersion over a hill ridge of the same shape is compared with the measured data.

4.2.1 Test case description

The performed numerical simulations reproduce the main aspects of the wind tunnel experiment. The 2D hill of a height h and a half-width a is described by the parametric equations

$$x = \frac{1}{2}\xi\left(1 + \frac{a^2}{\xi^2 + m^2(a^2 - \xi^2)}\right),$$
$$z = \frac{1}{2}m\sqrt{a^2 - \xi^2}\left(1 - \frac{a^2}{\xi^2 + m^2(a^2 - \xi^2)}\right) \quad \text{for } \xi \in [-a; a], \tag{4.2}$$

where $m = \frac{h}{a} + \sqrt{\left(\frac{h}{a}\right)^2 + 1}$. Two geometrical variants with different aspect ratios $n = a/h$ of 3 and 5 (marked in the following text as N3 and N5 respectively) are investigated. In both cases, the height of the hill is $h = 0.117$ m. Maximal slope of the N3 and N5 hills is 26° and 16° respectively. Shape of the hills is depicted on Fig. 4.5. The computation domain spans from -20h to 40h in the horizontal direction and from 0 to 13h in the vertical direction.

The boundary conditions are specified similarly as in the numerical simulation of the same problem by Castro and Apsley (1997). In the following text, D marks the depth of
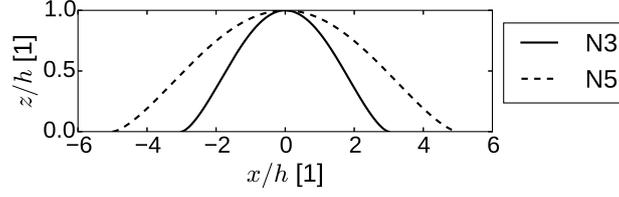
Figure 4.5: Shape of the N3 and N5 hills.

the boundary layer, defined such that the velocity given by the log profile wind profile at the top of the boundary layer is equal to free stream velocity, $u(D) = u_\infty$. At the inlet, the log wind profile was prescribed for the velocity inside the boundary layer, i.e. for $z < D$, and $u(z) = u_\infty$ for $z \geq D$. Turbulence kinetic energy was given by $k(z) = C_\mu^{-0.5} u_*^2 (1 - \frac{z}{D})$ for $z < 0.9D$ and extended as a constant above, and its dissipation was set to $\epsilon = (C_\mu^{0.75} k^{1.5})/(\kappa z)$. Inlet potential temperature was set to constant $\theta = 289\,\mathrm{K}$, and the homogeneous Neumann boundary condition (BC) was used for the pressure perturbation, i.e. $\partial p^*/\partial \boldsymbol{n} = 0$. At the outlet and at the top of the domain, zero pressure perturbation was prescribed, and all other variables were extrapolated from inside using the homogeneous Neumann BC. At the ground, wall functions, as described in Sec. 2.4.3, were employed, together the homogeneous Neumann BC for pressure perturbation and potential temperature.

Parameters of the boundary layer are as follows: friction velocity $u_* = 0.178\,\mathrm{m\,s^{-1}}$, von Kármán constant $\kappa = 0.4$, roughness length $z_0 = 0.157\,\mathrm{mm}$, free stream velocity $u_\infty = 4\,\mathrm{m\,s^{-1}}$. Depth of the boundary layer was thus $D = 1.258\,\mathrm{m}$.

Dispersion of the pollutant was studied using the flow field calculated by the ABL solver, which was then extended to 3D, so that the flow field represented the flow above the ridge of the same shape as the 2D hill. The domain was extended in the lateral direction to $[-8h; 8h]$. A point source of the pollutant was placed on the midplane of the domain at the upwind base (horizontal position of the source $x_s = -a$), at the summit ($x_s = 0$), or at the downwind base ($x_s = a$) of the hill. The height of the source was $h_s = h/4$ in every case. Zero mass concentration was prescribed at the inlet, and Neumann boundary conditions were used on all other boundaries.

4.2.2 Grid dependency study

First, the effect of the size of the near wall cell was investigated on the N3 case. Two computational grids, marked as G1 and G2, were used. Both grids had the same amount of cells, but differed in the near-wall cell size. The grids were structured, and expanded both vertically and horizontally away from the hill. Their parameters are summarized in Table 4.2, and the detail of the grid G1 around the hill is shown on Fig. 4.6. The dimensionless wall distance $z^+ = \frac{z u_*}{\nu}$ of the center of the near wall cell in the unperturbed flow is 30 for the grid G1 and 12 for the grid G2. Strictly speaking, the used wall functions are valid for $z^+ > 30$ and the grid G2 is therefore unsuitable. However, the friction velocity is expected to increase at the hill summit, so the investigation of the effect of the near wall cell was deemed necessary.

55

	G1	G2
Cells in horizontal direction	340	
Cells in vertical direction	100	
Minimum horizontal cell size	$0.06h$	
Horizontal expansion ratio	1.017	
Minimum vertical cell size	$0.043h$	$0.017h$
Vertical expansion ratio	1.020	1.033

Table 4.2: Grid parameters
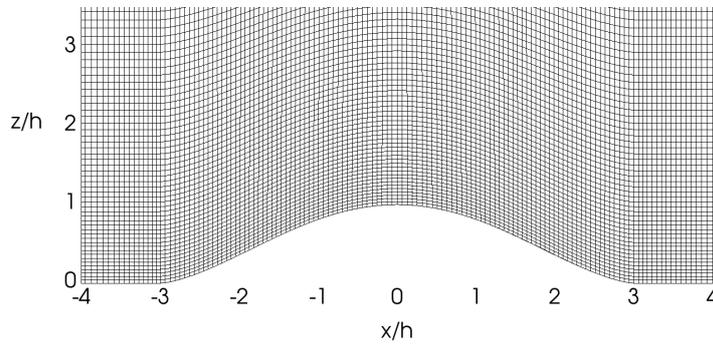
Figure 4.6: Detail of the computational grid G1 around the N3 hill.
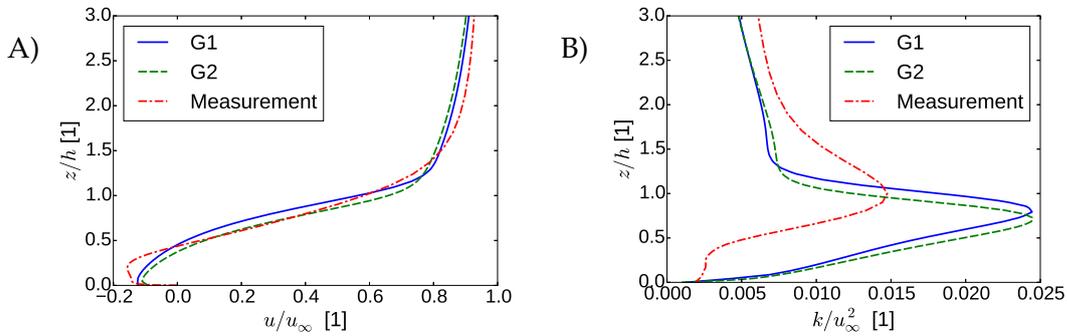
Figure 4.7: Vertical profiles at the downstream base of the N3 hill ($x = a$) for the investigated grids. (A) Non-dimensionalized horizontal velocity. (B) Non-dimensionalized TKE.

Fig. 4.7 shows the obtained vertical profiles of the velocity and the turbulent kinetic energy, compared with the measured values. The solutions on both grids have similar properties, and does not differ significantly. Separation of the flow is captured on both grids, as proved by the negative horizontal velocities near the ground. Similarity of the solutions holds also for other locations (not presented here). Based on this, in the further

simulations we will always use the grid G3 (or, for the N5 hill, a grid with the same near-ground cell size).

For the pollutant dispersion, the grid was extruded to 3D. The number of cells in the lateral direction was 75. The grid was refined in the middle of the domain, so that the lateral size of the smallest cells was $0.051h$, and the cells were expanded to the sides with the expansion factor 1.066. Total number of cells for the pollutant dispersion calculations was 2.55 millions.

4.2.3 Results

Flow field

Figures 4.8 and 4.9 show the vertical profiles of the normalized horizontal velocity and the turbulent kinetic energy for both N3 and N5 hills, as calculated by the compressible solver and the ABL solver, compared with the measurements. First, we note that the results from both solvers are almost indistinguishable. That is expected: at the wind tunnel length scale below 1 meter, the ABL approximation are safely within the range of validity, and there is no reason the expect any differences. The agreement between the solvers therefore provides a partial indication of the correct implementation, at least in the sense that the possible programming errors are shared by the solvers.

The solution also reasonably agrees with the measurements. The calculated solution shows a flow separation for the N3 hill and no separation for the N5 hill, as was observed in the experiment (see also the velocity streamlines in Fig. 4.10). The reattachment point for the N3 hill is however closer to the hill in our computations ($x/h = 5.6$) than what was indicated by the measurement ($x/h = 6.4$). The solvers also fail to predict the small flow speedup at the top of the N3 hill (Fig. 4.8, top row, $x/a = 0$). A near ground increase in the turbulent kinetic energy downstream of the hill is reproduced for both geometrical variants (Fig. 4.8 and Fig. 4.9, bottom rows), however, the maximum of TKE is overpredicted for the separated flow. At the same time, the TKE is generally underpredicted further from the N5 hill.

Overall, the solution shows a good level of agreement, especially looking at the calculated flow field. The choice of the turbulence model is expected to have a very significant influence on the results, and more complex turbulence models (such as the Reynolds Stress Model) might provide better agreement even in the calculated TKE.

Pollutant dispersion

The calculated and measured concentrations are presented here in normalized form,

$$c^+ = \frac{cu_\infty h^2}{Q}, \tag{4.3}$$

where Q is the source intensity in $\mathrm{kg\,s^{-1}}$.

First, the influence of the choice of the turbulent Schmidt number Sc_T was assessed. Fig. 4.11 shows the ground level concentration profiles for the source at the downwind base of the N3 hill. The Schmidt number plays affects the dispersion of the pollutant further away from the hill, but the position and the value of the maximum near the source
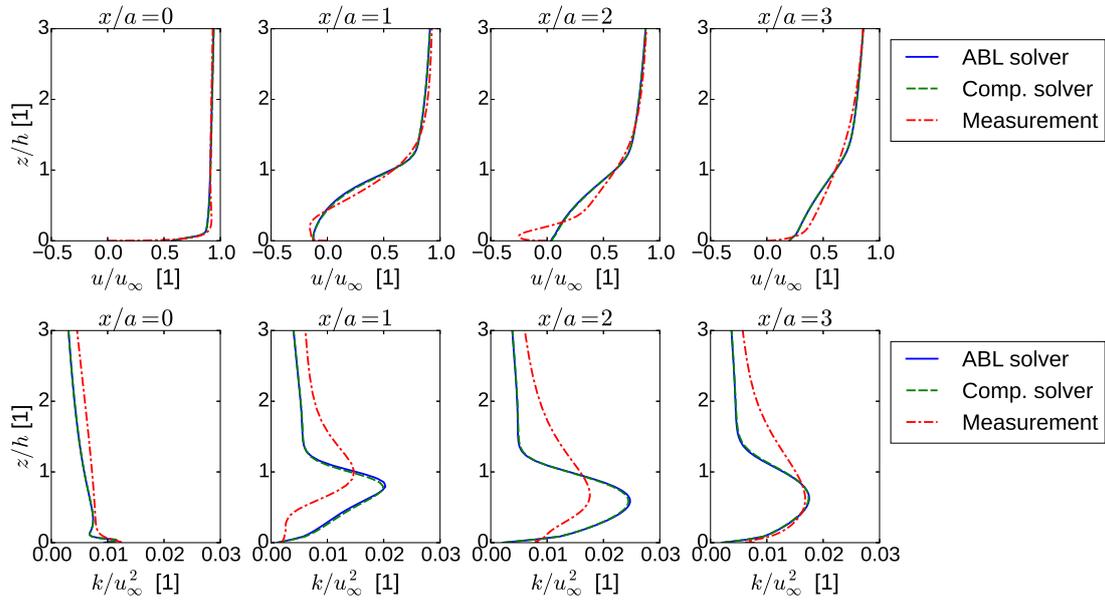
Figure 4.8: Flow around the N3 hill. Vertical profiles of the normalized horizontal velocity (top row) and the normalized turbulence kinetic energy (bottom row) at the hill summit ($x/a = 0$), at the downstream base ($x/a = 1$), and downstream from the hill ($x/a = 2$ and $x/a = 3$).
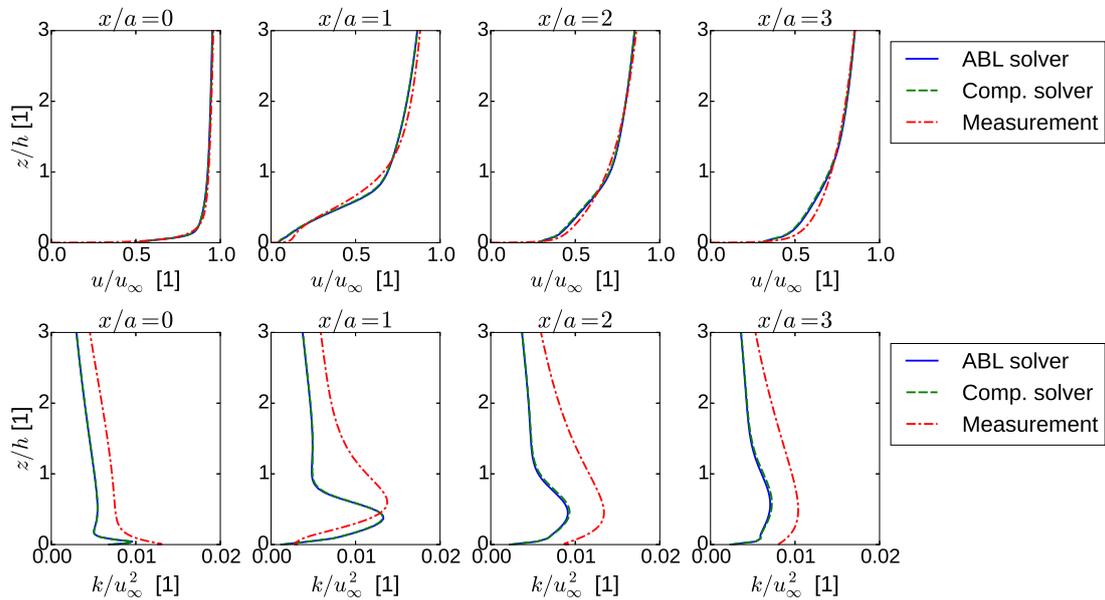
Figure 4.9: Flow around the N5 hill. Vertical profiles of the normalized horizontal velocity (top row) and the normalized turbulence kinetic energy (bottom row). at the hill summit ($x/a = 0$), at the downstream base ($x/a = 1$), and downstream from the hill ($x/a = 2$ and $x/a = 3$).
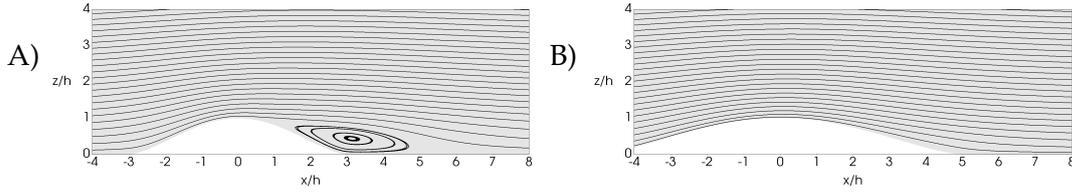
Figure 4.10: Velocity streamlines around the (A) N3 and (B) N5 hill as calculated by the ABL solver.

is relatively unaffected. The value $Sc_T = 0.7$ reproduces the rate of decay quite well, even if the absolute value of c^+ is underpredicted, and it is used in the further computations presented in this section.
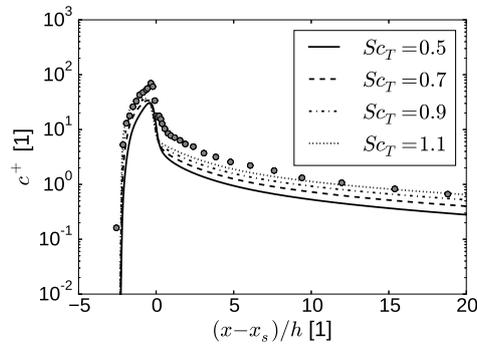
Figure 4.11: Flow around a hill. Ground level normalized concentration with source at the downwind base of the N3 hill for various turbulent Schmidt numbers. CFD simulations (lines) and measurements (symbols).

It shall be noted here that the assumption of the isotropic turbulent diffusivity employed in our approach is inherently inaccurate in the ABL flows, where the horizontal and vertical diffusivities may significantly differ. However, since the lateral and vertical profiles of concentrations were not available for the presented cases, the effect of this inaccurate assumption could not be assessed.

Fig. 4.12 shows the ground level concentrations for all calculated source positions and for both hill shapes. Several discrepancies between the measured and calculated values are present, and deserve some commentary.

First, the calculated concentration of the pollutant released at the summit of the hill is well below the measured values for both hill shapes (Fig. 4.12, middle column). Cause of this error is unclear. On possible explanation may lie in the fact that the wind speed at the release point is higher at the summit than at the bases due to the flow speedup. Lower levels of the calculated turbulent diffusion at the summit would thus lead to the pollutant being advected faster, producing the observed underprediction.

Secondly, the concentration is underpredicted further away from the N3 hill for all source positions (Fig. 4.12, upper row). This may be caused by the overpredicted TKE
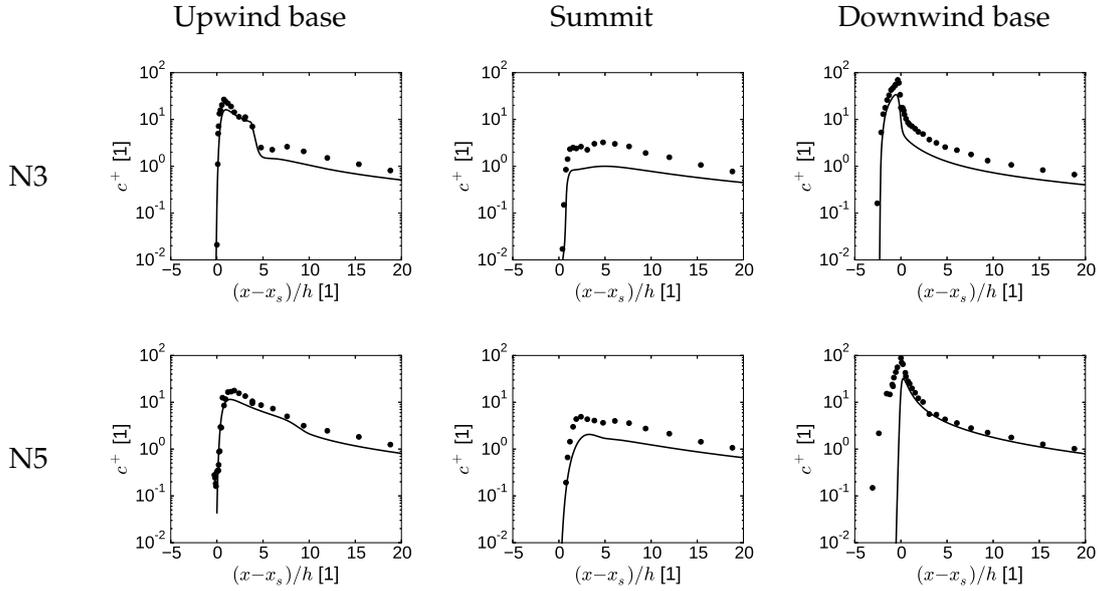
Figure 4.12: Flow around a hill. Ground level normalized concentration with the pollutant source at the upwind base (left column), at the summit (middle column), or at the downwind base (right column) of the N3 and N5 hills. Height of the source is $h_s = h/4$ in all cases. CFD simulations (lines) and measurements (symbols).
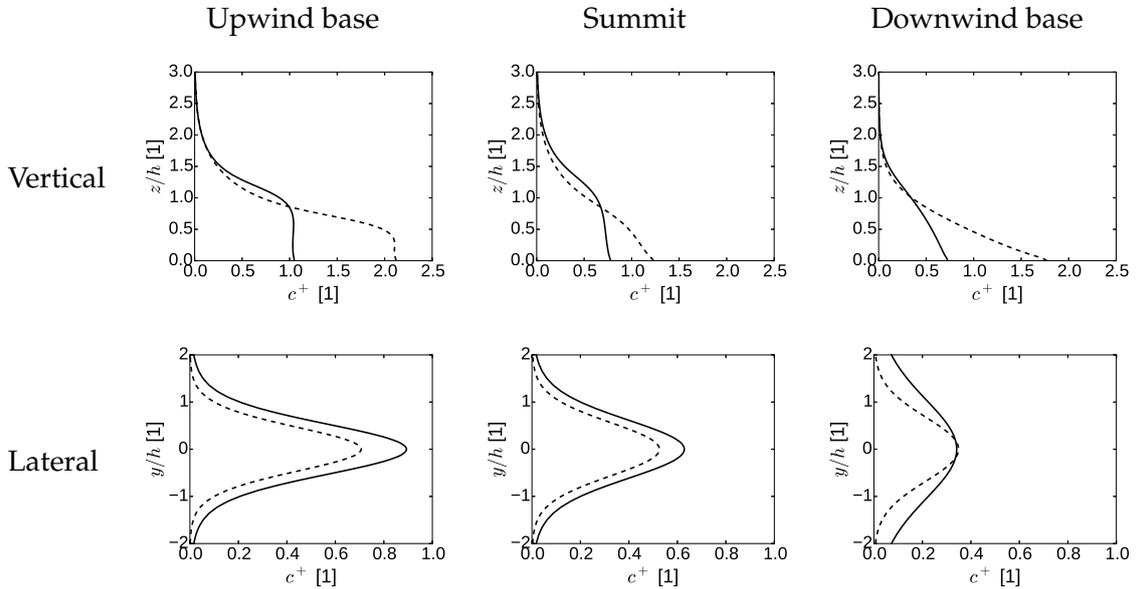
Figure 4.13: Flow around a hill. Profiles of the normalized concentration at $(x - x_s)/h = 10$. (Top row) Vertical profiles at the midplane. (Bottom row) Lateral profiles at height $z = h$. Hill shape: N3 (———), N5 (– – –).

close to the hill (see Fig. 4.8), and thus increased turbulent mixing in that area, leading to a faster dispersion of the pollutant.

And lastly, we note that the measurements show high concentration values upstream from the source placed at the downwind base of the N5 hill, which is not reproduced by the computation. Castro and Apsley (1997) speculated that this is caused by the flow separation occurring intermittently, which was not captured by the measurements (nor our RANS model). In that case, the source would be occasionally placed in the separation bubble, and the pollutant would be advected upstream. Such increase of the pollutant upstream of the release point may be observed on the concentration values for the release point at the downwind base of the N3 hill. Our calculation places the release point in the separation bubble, and the concentration measurements agree well with the calculations.

Fig. 4.13 shows the calculated vertical and lateral profiles of the concentration $10h$ downstream from the source. Measurements were not available. The vertical profile is clearly affected by the presence (or absence) of the separation bubble: its presence leads to the reduced concentration at the ground due to the pollutant being advected in the upper layer.

4.2.4 Performance study

As our interest ultimately lies in the full scale atmospheric simulations, computational performance of the solvers is evaluated on the same 2D hill problem in the N3 configuration, but scaled up to resemble the full scale problems. Two variants of the problem are investigated: one with the hill height $h^f = 11.7\,\mathrm{m}$ (i.e. scaled up by a factor of 100, marked as S100 in the following text), and the other with $h^f = 117\,\mathrm{m}$ (scale factor 1000, marked as S1000). First geometrical configuration might constitute an artificial near-road earth berm, second could represent a standalone hill in a flat terrain. The roughness length is scaled to keep the same ratio $\frac{z_0}{h} = \frac{z_0^f}{h^f}$ (Khurshudyan et al., 1981). Friction velocity in both cases is set to somehow arbitrary value $u_* = 0.445\,\mathrm{m\,s^{-1}}$ one might encounter in real situations. Reynolds number is therefore not kept equal to the value $Re = \frac{u_\infty h}{\nu} = 3.12 \cdot 10^5$ achieved in the wind tunnel experiment. After all, Reynolds numbers encountered in full scale atmospheric problems are on the order of $10^7 - 10^9$, which is difficult or impossible to reproduce in the wind tunnel experiments. However, in all situations discussed here - in the experiment as well as in the full scale simulations - is the Reynolds number high enough that the flow is fully turbulent, and due to the Reynolds number independence at high Reynolds numbers one might expect to obtain comparable results. Parameters of the scaled up simulations are summarized in Tab. 4.3. The stratification of the atmosphere is kept neutral as in the wind tunnel experiment.

We compared the performance of the ABL solver employing the classical artificial compressibility formulation (Eq. (3.3)) and the generalized artificial compressibility formulation (Eq. (3.4)). Both approaches were tested with four values of the parameter β: 10, 100, 1000, and 10000, and are in the following text marked as AC-C-β (the classical approach) and AC-G-β (the generalized approach). These were further compared with the compressible solver using no preconditioning (CN) and using the Weiss-Smith preconditioning (Eq. (3.12), marked CP).

All simulations were run on a single core of Intel Xeon E5520 processor. The iteration

	S100	S1000
Hill height h	11.7 m	117 m
Roughness length z_0	0.0157 m	0.157 m
Friction velocity u_*	0.445 m s^{-1}	0.445 m s^{-1}.
Free stream velocity u_∞	10 m s^{-1}	10 m s^{-1}
Reynolds number $Re = \frac{u_\infty h}{\nu}$	$7.8 \cdot 10^6$	$7.8 \cdot 10^7$

Table 4.3: Full scale simulations parameters

process was stopped when the u_1 or ρu_1 residual for the ABL and compressible solver respectively dropped below 10^{-5}. For simplicity, both residuals will be marked as u_1 residual in the following text. Residuals of the other variables behaved similarly as u_1 residual, which was considered representative of the convergence behaviour. To prevent the unphysical oscillation in time the relaxation of the eddy viscosity as discussed in Sec. 3.5.3 is employed with the parameter $q = 0.5$. The computational mesh denoted as G1 is used just as before, only scaled up to proper dimensions.

Results. First we look at how the different artificial compressibility formulations affect the solution process. Fig. 4.14 shows the evolution of the residuals for the performed simulations. We note that for lower values of β the solver employing classical formulation diverges in both geometrical variants. In these cases, the non-divergence constraint is far from being satisfied until the steady state is reached. Generally, this should not pose a problem, as we are interested only in this steady state solution, however, in this case of flow over a hill it has unpleasant consequences. Potential temperature advected by the velocity field with nonzero divergence locally raises by a significant amount, and through the gravity term in the momentum equation produces large vertical velocities. The solver cannot recover from this and the process ends with the divergence of the solver (Fig. 4.14 A, AC variants with $\beta = 10, 100$; Fig. 4.14 C, AC variant with $\beta = 10, 100, 1000$). Generalized formulation diverges only for the lowest tested $\beta = 10$, and comes out as a more robust choice in this situation. When both formulations converge, the solution time of the classical formulation is similar or higher compared to the generalized formulation, as visible in Tab. 4.4, which shows the CPU time needed by each variant to converge.

Evaluation of the results confirms that choice of the parameter β is crucial in obtaining optimal performance. While higher β should allow for faster convergence to divergence free velocity field, resulting systems of linear equations are badly conditioned, leading to slower solution of the nonlinear system in each pseudo-time step, and slower convergence rate overall. This is clearly demonstrated in Tab. 4.4. Furthermore, in the later stages of the iteration process when longer time steps are employed, the factorization produced by the ILU preconditioner is inaccurate. This results in a failure of the nonlinear solver and a necessary repetition of the step with a lower pseudo-time step, retarding the convergence to the steady state. We conclude that in this study, the elsewhere mentioned choice of the parameter $\beta = U^2$ (Muldoon and Acharya, 2007), corresponding to $\beta = 100$ here, indeed yields optimal results, but only in the generalized formulation, as the solver
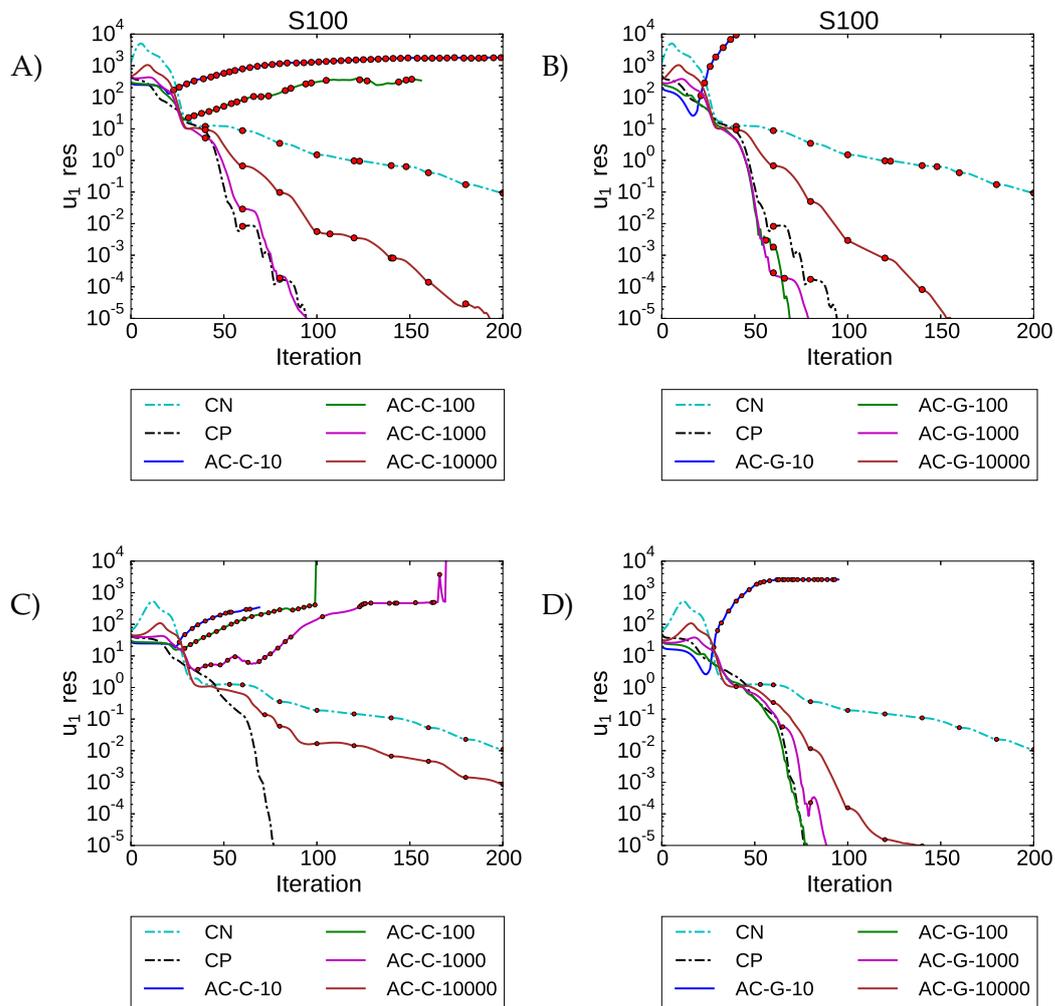
Figure 4.14: u_1 residuals for the flow around the hill in the (A,B) S100 and (C,D) S1000 scaled up geometry. Panels A and C show the convergence of the classical artificial compressibility (AC-C) formulation, panels B and D show the generalized artificial compressibility (AC-G) formulation. The convergence of the non-preconditioned (CN) and preconditioned (CP) compressible solver is shown in all panels. Red dots mark the iterations where the nonlinear system solve failed, and where the iteration was subsequently repeated with shorter time step. Full history of the slowly converging variants is not shown.

	S100	S1000
CN	6.42	5.57
CP	1.00	0.65
AC-C-10	Diverged	Diverged
AC-C-100	Diverged	Diverged
AC-C-1000	0.98	Diverged
AC-C-10000	1.63	3.67
AC-G-10	Diverged	Diverged
AC-G-100	0.61	0.54
AC-G-1000	0.76	0.74
AC-G-10000	1.33	1.09

Table 4.4: CPU times of the solution process of the flow around the hill for all calculated variants. Times are normalized by the solution time of the CP variant in S100 case (136 min 57 s).

does not converge with the classical formulation.

The compressible solver without preconditioning is stable, but it converges several times slower than the ABL solvers, as visible in Fig. 4.14 and Tab. 4.4. Preconditioning, however, removes this deficiency. With its use, the compressible flow solver converges in comparable time to the ABL solvers, slightly slower or faster depending on the choice of parameter β in the ABL solver.

4.3   Wind flow in and around a forest canopy

The described k-ϵ model of the vegetation flow was tested on a problem of a flow in and above a forest canopy. Dupont et al. (2011) presented field measurements and large eddy simulations of a flow over a maritime pine forest. The forest of an average height $h = 22$ m has a dense crown layer roughly 8 m thick and an open trunk space. The 41.5 m high measurement tower was located $9h$ from the edge of the forest in the north-west direction, while a homogeneous forest with a fetch greater than 1 km stood in the opposite direction from the tower. In addition to the tower, a smaller mast of height 13 m was located $4h$ from the edge of the forest. This configuration allowed to investigate both the flow over a homogenous forest as well as the edge effects based on the wind direction.

4.3.1   Numerical model

The flow was simulated using only the solver based on the ABL flow equations using the k-ϵ turbulence model. Two sets of k-ϵ model constants are investigated. Recall from

Sec. 2.4.2 that the model constants have to satisfy the relation

$$\sigma_\epsilon = \frac{\kappa^2}{(C_{\epsilon_2} - C_{\epsilon_1})\sqrt{C_\mu}}.$$

so that the model can reproduce balanced flow in an unperturbed atmosphere. This requirement led to a set of constants different from those used in the standard model. In their validation of the vegetation model we adopted (Katul et al., 2004), the authors changed the constant C_μ to 0.03 to provide a better match of the turbulent viscosity ν_T in the unperturbed atmosphere to the measured values of typical neutral ABL flows. To satisfy the above relation, constants σ_ϵ had to be changed accordingly as well. In the following numerical experiments we test the set of constants derived earlier for the ABL flows (marked as the default set) and this modified set (see Table 4.5).

	C_{ϵ_1}	C_{ϵ_2}	C_μ	σ_k	σ_ϵ
Default set	1.44	1.92	0.09	1.0	1.167
Modified set	1.44	1.92	0.03	1.0	1.92

Table 4.5: Tested set of constants of the k-ϵ model for the forest canopy flow.

The flow over a homogenous forest was investigated using a 1D model and the edge flow was examined using a 2D model (Fig. 4.15B).
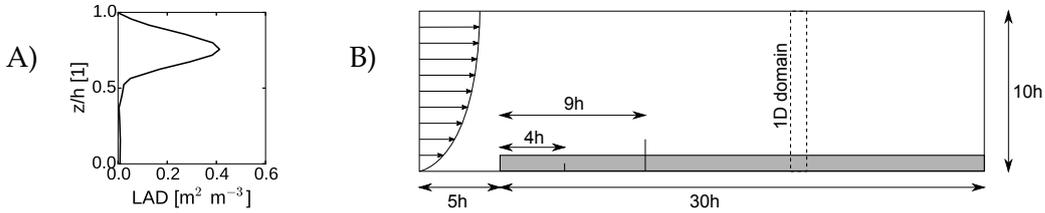
Figure 4.15: Flow over a forest canopy. (A) LAD profile of the pine forest. Vertical coordinate is normalized by the height of the forest $h = 22\,$m. (B) Sketch of the 2D domain. Positions of the measurement masts at $4h$ and $9h$ from the edge of the forest are marked by lines. Computational domain of the 1D model of the flow over a homogeneous forest is represented by the dashed lines.

Homogeneous forest. The 1D vertical model was constructed as follows. The flow was modelled between the ground and a height $10h$. At the ground the wall functions were prescribed with roughness length $z_0 = 0.03\,$m. Pressure perturbation and potential temperature were extrapolated from inside using homogeneous Neumann BC. The upper boundary was modelled as a slip wall. The flow was driven by prescribed horizontal pressure gradient acceleration $\frac{1}{\rho_{\text{ref}}}\frac{\partial p^*}{\partial x} = 0.001\,$m s^{-2}. The leaf area density of the canopy is pictured in Fig. 4.15A. The vegetation drag coefficient was set to $C_d = 0.26$. Atmosphere was considered to be neutrally stratified with potential temperature $\theta = 300\,$K.

The vertical interval was discretized by 100 cells. The cells inside the canopy (i.e. for $z < h$) had height $0.023h$, and the cells above were continuously expanded with an expansion factor 1.06.

Edge flow. To capture the behaviour of the flow over the edge of the forest a 2D model was employed (Fig. 4.15B). The size of the computational domain was chosen to allow the flow to stabilize before reaching the outlet. Dupont et al. (2011) evaluated that the adjustment region extends to around $22h$ from the forest edge in this case. Based on this, the computational domain was set to extend to $30h$ downstream from the edge of the forest, and $5h$ upstream. The height $10h$ is same as in the 1D model.

Boundary conditions at the ground were the same as in the 1D model. At the outlet zero pressure fluctuation was prescribed, and the homogeneous Neumann BC for all other variables was used. Log wind profile was prescribed at the inlet with the friction velocity $u_* = 0.23 \, \mathrm{m \, s^{-1}}$, and it was complemented by the turbulence inlet profiles described in Sec. 2.4. Potential temperature was set to $\theta = 300 \, \mathrm{K}$. Finally, the homogeneous Neumann BC was prescribed at the top of the domain for all variables except for the pressure, which was calculated so that the total pressure $p_0 = p + \frac{1}{2}\rho U^2$ was constant at the top boundary. The same leaf area density profile as in 1D case (Fig. 4.15A) and the same drag coefficient $C_d = 0.26$ were used.

The domain was discretized by 100 cells in vertical direction, using the same grading as in the 1D model. In horizontal direction 300 cells were used with width $0.023h$ at the edge of the forest, and expanding upstream with factor 1.05 and downstream with factor 1.011.

4.3.2 Results

Homogeneous forest. Fig. 4.16, left column, shows the vertical profiles of normalized horizontal and vertical velocities, Reynolds stresses and turbulence kinetic energy, together with the measured values for the homogeneous forest case. The values are normalized by a reference flow velocity u_{ref} and friction velocity u_*, both measured at the top of the tower, i.e. at height $z = 41.5 \, \mathrm{m}$.

Above the canopy the velocity profile has a typical logarithmic profile. The velocity is quickly reduced inside the canopy, and reaches a secondary maximum in the open trunk space. This is however reproduced only by the model with the modified constants, and not by the model with the default constants. In that case, the reduction of the velocity is not as extensive, and the horizontal velocity is overpredicted inside the canopy and the trunk space. The momentum fluxes are reduced to negligible values below the crown layer, signifying minimal momentum transfer between the flow above and below the crown layer. Turbulence kinetic energy is overpredicted by the model with default constants inside and above the canopy, while the modified constants model shows good agreement with the measurements.

Edge flow. The middle and the right columns of Fig. 4.16 show the vertical profiles of the same quantities as in the case of homogeneous forest at a distance $4h$ and $9h$ from the

| Homogeneous forest | Edge flow - $4h$ | Edge flow - $9h$ |
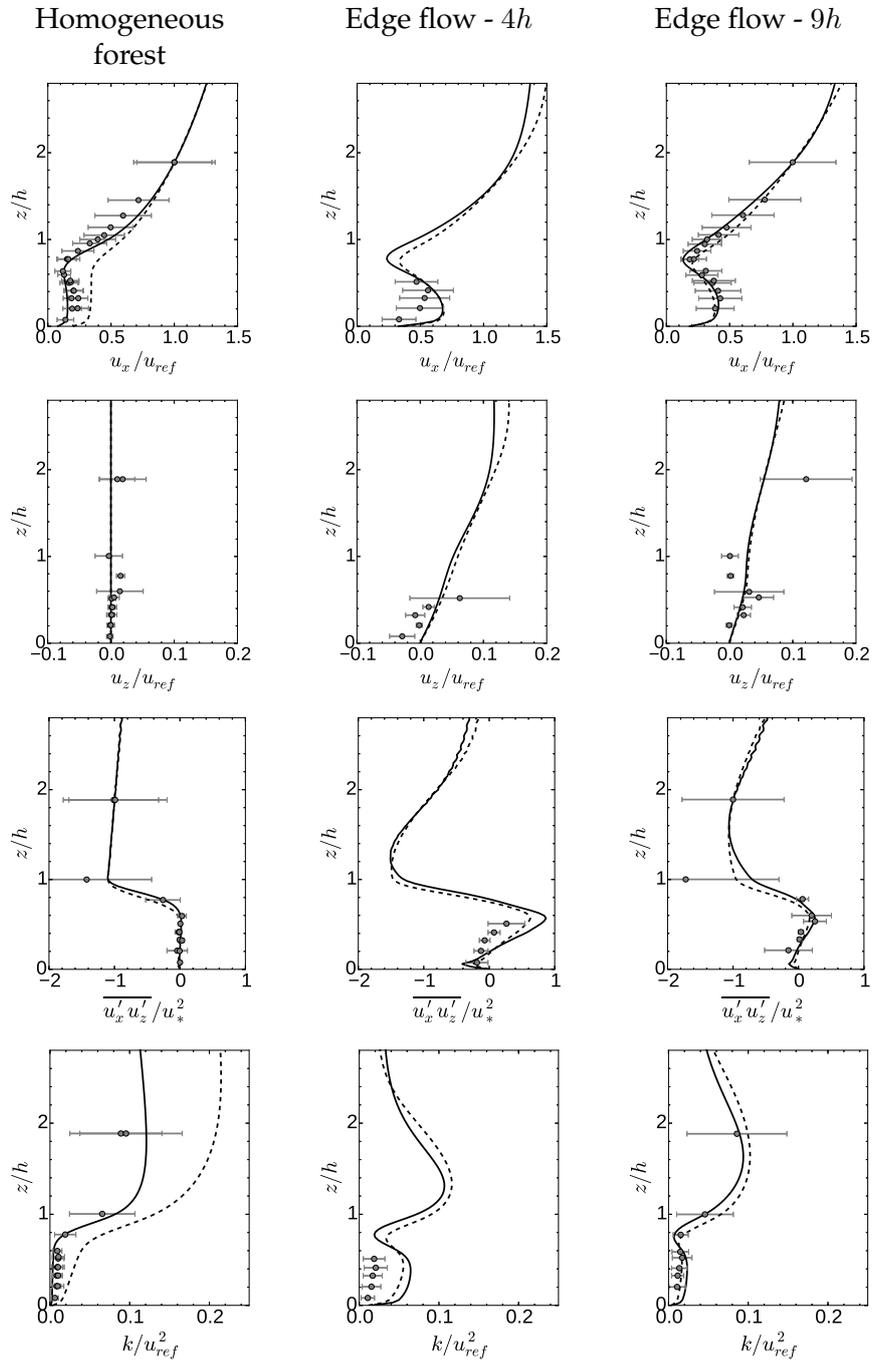
Figure 4.16: Vertical profiles of horizontal velocity u_x, vertical velocity u_z, Reynolds stresses and turbulence kinetic energy k inside and above the canopy. Values are normalized by the reference velocity u_{ref} or friction velocity u_* measured at $z = 41.5\,\mathrm{m}$. Dashed lines: default constants, Solid lines: modified constants, symbols: measurements by Dupont et al. (2011).

edge of the forest. The measured values at $4h$ are available only in the lower half of the canopy due to the smaller mast.

Compared to the homogeneous forest case, the secondary maximum of the horizontal velocity inside the trunk space is much more prominent. That is reproduced very well, especially at $9h$ from the edge. Upward motion caused by the deceleration of the flow, observed both at $4h$ and $9h$, is typical for the adjustment region close the the edge of the forest (Dupont and Brunet, 2008a). The positive momentum flux inside the canopy, noted as "striking" in (Dupont et al., 2011), is reproduced at $4h$ and especially well at $9h$. Turbulence kinetic energy inside the canopy is overpredicted at $4h$, but reaches a good agreement at $9h$.

4.3.3 Discussion

In the 2D edge flow case, the flow is well reproduced by the model with both sets of constants. In the 1D homogeneous flow case the default set performs considerably worse, however, main features of the flow are still captured. Arguably, the performance in the edge flow case is more relevant to the intended application of our model, which is mainly aimed at the problems of urban flows. In these settings, small, separated patches of vegetation are more typical than the continuous vegetation cover represented by the 1D case. Therefore, considering comparable performance of the models with both set of constants in the 2D case, better performance with the modified set of constants does not justify the change of the universally accepted constant $C_\mu = 0.09$, so often used by the atmospheric modelling community for the flows without the vegetation (Castro and Apsley, 1997; Hargreaves and Wright, 2007; Balogh et al., 2012; Vranckx et al., 2015) as well as with the vegetation present (Svensson and Häggkvist, 1990; Green, 1992; Kenjereš and ter Kuile, 2013; Gromke and Blocken, 2015). Therefore it is the default set of constants we have used in all further simulations performed by our solver.

4.4 Particle collection by a vegetation barrier

The dry deposition model described in Sec. 2.5.2 is tested on the problem of a hedgerow filtering particle-laden flow that was originally investigated in (Tiwary et al., 2005). In their field experiments, the authors measured concentrations of polystyrene particles of diameters between $0.8\,\mu\mathrm{m}$ and $15\,\mu\mathrm{m}$ upwind and downwind of the hawthorn hedge. From these measurement, the collection efficiency of the barrier was determined. The authors investigated the problem also numerically, using a detailed vegetation model. The same case was also numerically investigated in (Guo and Maghirang, 2012).

In our study, we have constructed a 2D numerical model reproducing the experiment, and evaluated the influence of several parameters of the model on the results, namely of the drag coefficient C_d and of the properties of the leaves.

4.4.1 Numerical model

The vegetation barrier of width $w = 1.6\,\mathrm{m}$ and height $h = 2.2\,\mathrm{m}$ was placed inside the computational domain spanning $20w$ upwind and $40w$ downwind from the end of the

barrier and with height $10h$ (see Fig. 4.17, panel B). The barrier was porous, described by its leaf area density profile (Fig. 4.17, panel A), obtained from the original paper.
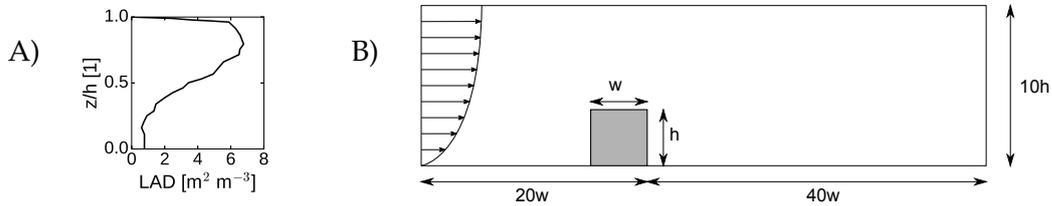
Figure 4.17: Flow through a vegatation barrier. (A) LAD profile of the vegetation. Vertical coordinate is normalized by the height of the barrier $h = 2.2$ m. (B) Sketch of the domain (not to scale).

The flow field was calculated using the solver based on the ABL flow equations employing the standard k-ϵ model extended by the vegetation model.

The boundary conditions for the flow equations were set as follows: At the inlet and at the top of the domain, log wind profile with $u_* = 0.198\,\mathrm{m\,s^{-1}}$ and $z_0 = 0.0189\,\mathrm{m}$ was prescribed. The reference velocity at $z = h$ was thus $u_{ref} = 2.3\,\mathrm{m\,s^{-1}}$. Potential temperature $\theta = 293$ K was set to a constant value to model the neutrally stratified atmosphere. Neumann BC was prescribed for the pressure. Profiles of the turbulence variables were given by equations (2.73) and (2.74). At the outlet, zero pressure fluctuation p^* was prescribed, and the homogeneous Neumann BC was used for all other variables. At the ground, the wall functions were used, together with the homogeneous Neumann BC for pressure fluctuation and potential temperature.

The transport and the collection of the particles of the diameters 0.875, 1.5, 2.75, 4.25, 6.25, 8.75, 12.5 and 15 μm and the density $\rho_p = 1050\,\mathrm{kg\,m^{-3}}$ was investigated using the passive scalar equation (2.44). Gravitational settling of the particles (Sec. 2.2.1) was included in the model, as well as the dry deposition on the vegetation (Sec. 2.5.2). The ambient background concentration was obtained by prescribing a concentration $1\,\mathrm{mg\,m^{-3}}$ at the inlet and at the top of the domain. At all other boundaries, the homogeneous Neumann BC was used for the particle concentration.

The unstructured computational mesh was generated using the *snappyHexMesh* generator from the OpenFOAM software package (Greenshields, 2015). The mesh, consisting of nineteen thousands cells, was refined around the vegetation barrier (Fig. 4.18). The largest cells in the domain were $0.8\,\mathrm{m} \times 0.73\,\mathrm{m}$ large and the smallest were $0.1\,\mathrm{m} \times 0.092\,\mathrm{m}$ large, so that the vegetation block itself was discretized into 16×24 cells.

4.4.2 Results

Influence of the drag coefficient. The flow through and around the barrier was calculated for four values of the drag coefficient C_d, spanning the interval from 0.15 to 0.5 of realistic drag coefficient values (Endalew et al., 2009; Katul et al., 2004). Fig. 4.19 shows the vertical profiles of the velocity magnitude behind the barrier normalized by the reference inlet velocity at height h, compared with the measured values.
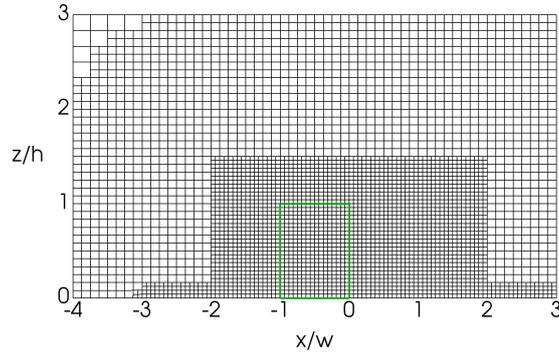
Figure 4.18: Computational mesh around the vegetation barrier.
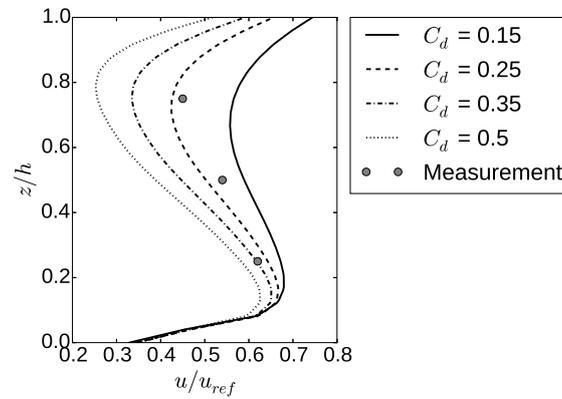
Figure 4.19: Vertical profiles of normalized velocity magnitude behind the vegetation barrier. Measured values taken from (Tiwary et al., 2005).

As expected, the largest slowdown is in all cases observed around $z/h = 0.8$, where the LAD profile attains its maximal value. Local maximum of the velocity profile is visible around $z/h = 0.15$, to where is the blocked flow deflected. Near-ground behaviour is affected mostly by the ground shear stress and is independent on the choice of the drag coefficient.

Choice of the drag coefficient $C_d = 0.25$ provides a reasonable agreement with the measured values at $z/h = 0.25$, 0.5 and 0.75. It is worth noting that the authors of the original paper (Tiwary et al., 2005) used the value $C_d = 0.5$ in their simulations and obtained a good agreement as well. This may be caused by the different vegetation model: while the source term in the momentum equation is the same in our and in their formulation, the authors of the original paper did not modify the turbulence equations to include the vegetation effects.

Parameterization of the leaves. Let us now turn to the filtering properties of the hedge-row. From the experiment, the filtering capacity was described via the *collection efficiency*

(CE), defined as

$$CE = \frac{c_{in} - c_{out}}{c_{in}}, \tag{4.4}$$

where c_{in} and c_{out} are the values of the mass concentration measured $0.1h$ upwind and $0.1h$ downwind from the barrier at height $0.75h$. The collection velocity generally falls into the range between 0 and 100%, but may reach negative values if the pollutant accumulates behind the barrier so that $c_{out} > c_{in}$.

In the adopted deposition velocity model, the vegetation is described by its type and typical size of the vegetation elements. Fig. 4.20, panel A, shows the calculated collection efficiencies when the vegetation elements are modelled as leaves with different diameters d_e. The increasing collection efficiency for particles of larger size, observed in the experiment, is clearly reproduced by our model. Furthermore, the CE increases for
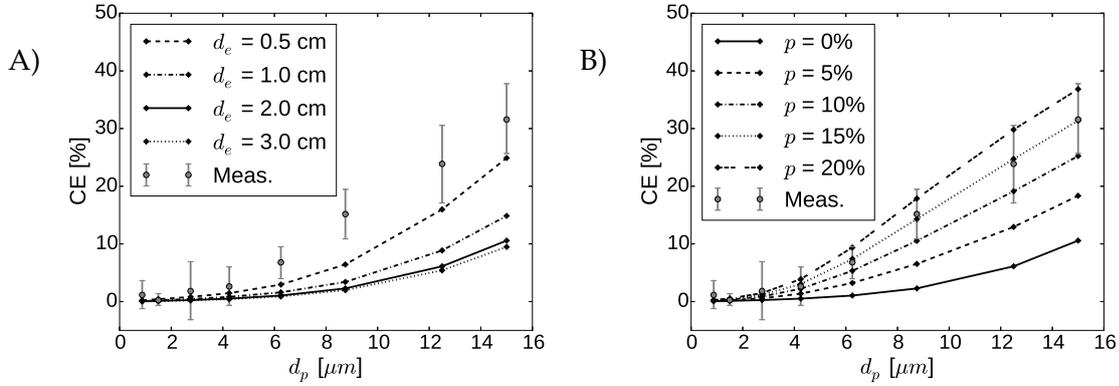
Figure 4.20: Collection efficiency and its dependency on the particle diameter and vegetation properties. (A) Leaves of different diameters d_e with smooth surface. (B) Thorny leaves, modelled as a mixture of smooth leaves of diameter $d_e = 2\,\text{cm}$ and needles of diameter $d_e = 0.5\,\text{mm}$ with the parameterized proportion p of the needle surface area to the total surface area. On both panels, the solid line references the same case of the vegetation with smooth leaves of diameter $d_e = 2\,\text{cm}$. Measured data taken from (Tiwary et al., 2005).

smaller leaf sizes. Tiwary et al. (2005) list that the range of the size of the hawthorn is between 1.1 and 3.2 cm. However, even when the leaf sizes are set to the half of the value at the lower end of the interval, the calculated collection efficiencies are still below the measured values. This may be attributed to the neglected fine needle-like collectors, such as the leaf hairs and thorns, which increase the deposition velocity (Beckett et al., 2000; Tiwary et al., 2005; Janhäll, 2015). To reflect this, we further modelled the vegetation as a mix of planar leaf elements of diameter $d_e = 2\,\text{cm}$ and fine needle-like collectors of diameter $d_e = 0.5\,\text{mm}$. The proportion of the surface area of the fine collectors to the total surface area is denoted as p, and the deposition velocity is calculated as

$$u_d = p u_d^{needle} + (1 - p) u_d^{leaf}, \tag{4.5}$$

where u_d^{needle} and u_d^{leaf} are the deposition velocities calculated for the needle and leaf elements respectively.

Comparison of the collection efficiencies calculated with this model and with the parameter p ranging from 0% to 20% is shown on Fig. 4.20, panel B. The CE rises with higher proportion of needles, reflecting the higher deposition velocity on the fine collectors. Best agreement with the measurement is obtained for $p = 15\%$.

Flow field and concentration field around the hedge. Further understanding of the deposition mechanisms at play may be obtained by inspecting Fig. 4.21, which shows the velocity field and the concentration of $15\,\mu m$ particles around the hedge. As in the numerical simulations of Tiwary et al. (2005) and Guo and Maghirang (2012), the flow deccelerates in the upper half of the hedge, and the blocked flow is partially redirected to the lower half, where the leaf area density is lower. Most of the particles are however removed near the upper edge of the vegetation block due to the large leaf area density. This results in a minimum in particle concentration at the downstream edge of the hedge. The concentration recovers to 90% of its inlet value at around $20w$ from the hedge (Fig. 4.22).
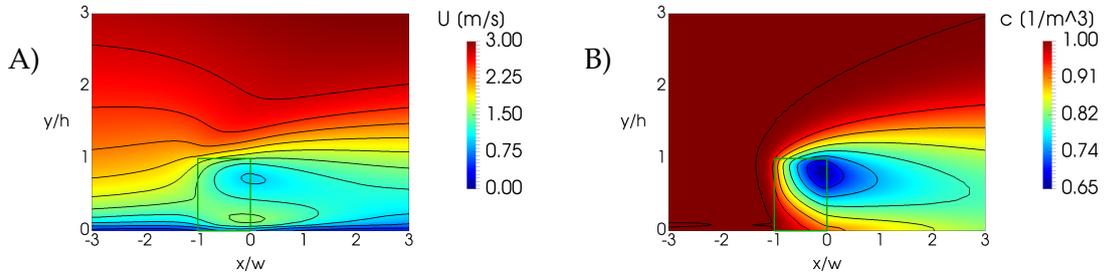
Figure 4.21: Results for $C_d = 0.25$ and mixed leaves and needles deposition velocity model with $p = 15\%$. (A) Velocity magnitude around the hedge. Contour interval is $0.25\,\mathrm{m\,s^{-1}}$. (B) Number concentration of particles of $d_p = 15\,\mu m$. Contour interval is $0.05\,\mathrm{m^{-3}}$.
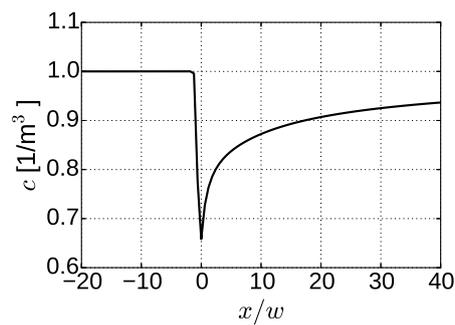
Figure 4.22: Horizontal profile of number concentration of particles of $d_p = 15\,\mu m$ at $z/h = 0.75$, calculated with $C_d = 0.25$ and mixed leaves and needles deposition velocity model with $p = 15\%$.

4.4.3 Conclusion

With some tuning of the vegetation parameters, good agreement with the measured results was obtained. The optimal drag coefficient $C_d = 0.25$ is within the range usually given as realistic for vegetation barriers. The mixed leaves-needles model is grounded in a realistic assumption, however, the exact proportion of the needle surface area is difficult to independently verify, as the author is not aware of any study on this topic.

For real-life applications where necessary measurements are not available, the estimation of the parameters of the vegetation might be a challenging matter. In that case, the inherent uncertainty must be managed by an appropriate method.

Chapter 5

Application: CFD optimization of a vegetation barrier

The content of this chapter is based on: Šíp, V. and Beneš, L. (2016b). CFD optimization of a vegetation barrier. In B. Karasözen, M. Manguoglu, M. Tezer-Sezgin, S. Göktepe and Ö. Ugur (Eds.), *Numerical mathematics and advanced applications - ENUMATH 2015*. Cham: Springer International Publishing. With permission of Springer.

In this chapter we present an application of the developed methods: a computational optimization of the near-road vegetation barrier, where the developed solver is employed to find an optimal properties of the barrier.

5.1   Introduction

Particulate matter (PM) pollution originating from the road traffic constitutes a significant health risk. Near-road vegetation barriers were proposed as a mean to the reduction of a harmful PM in the atmosphere. Due to the complexity of the problem, the assessment of the effectivity of the barriers and of its design is difficult without the computer simulations. In this section we employ the methods developed in earlier sections to numerically optimize the parameters of a near-road vegetation barrier in order to reduce the pollutant concentration behind the barrier. Specifically, the horizontal position of the barrier and its density is optimized.

5.2   Case settings

Fig. 5.1 shows the sketch of the model 2D problem. Four sources of pollutant, representing the road, were placed between 23 m and 42 m from the inlet at height 0.8 m. Vegetation block of height 15 m was placed downstream from the road.

The flow was calculated using the ABL flow model (2.60) employing the k-ϵ turbulence model, and the particulate matter dispersion was calculated using the passive scalar equation (2.63). Gravitational settling of the particles and the dry deposition was taken into account.
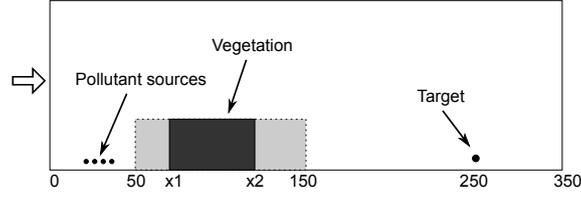
Figure 5.1: Sketch of the domain (not to scale).

We modelled the transport and dispersion of the particles of diameter $10\,\mu m$ and density $1000\,kg\,m^{-3}$. Each source of the pollutant had the intensity $1\,\mu g\,s^{-1}$. No resuspension of the particles fallen on the ground was allowed. The influence of the road traffic on the turbulence levels was represented by a source terms in the k and ϵ equations, using the model by Bäumer et al. (2005). Density of the traffic was set to 4 passenger cars and 1 heavy duty vehicle per minute in each of the four lanes.

The boundary conditions were set as follows:

Inlet and top Log wind profile with $u_{\text{ref}} = 5\,m\,s^{-1}$ at height $y_{\text{ref}} = 10\,m$ and $z_0 = 0.1\,m$. The temperature at the ground was set to $300\,K$ and above the ground is set so the atmosphere is under weakly stable stratification ($\partial T/\partial z = 0\,K\,m^{-1}$). Neumann BC was prescribed for the pressure. Profiles of the turbulence variables were given by equations (2.73) and (2.74).

Outlet Zero pressure fluctuation p^* was prescribed, and Neumann BC was used for all other variables.

Ground Wall function as described in Sec. 2.4.3.

Lateral walls Lateral wall were modelled as slip walls.

The computational mesh contained approximately 13 thousands computational cells, and is refined around the vegetation barrier. The largest cells in the domain, located in the upper part, were 4.3 m x 2.5 m large, while the smallest, located near the ground and around the vegetation, were 1.1 m x 0.63 m large.

5.3 Optimization

5.3.1 Method

General PDE-constrained optimization problem may be written in the following form:

$$\text{Find } \min_{\boldsymbol{p}} J(\boldsymbol{Y}, \boldsymbol{p}) \quad \text{subject to} \quad \boldsymbol{G}(\boldsymbol{Y}, \boldsymbol{p}) = 0 \tag{5.1}$$

and constrained by

$$p_i^{min} \leq p_i \leq p_i^{max} \quad i = 1...n, \tag{5.2}$$

$$g_j(p) \leq 0 \quad j = 1...m. \tag{5.3}$$

Here $J(\boldsymbol{Y}, \boldsymbol{p})$ is a cost function, $\boldsymbol{G}(\boldsymbol{Y}, \boldsymbol{p})$ is the system of steady-state PDEs, $\boldsymbol{Y}$ is the global state vector, and $\boldsymbol{p}$ is the vector of parameters. Allowed values of parameters are limited by p_i^{min} and p_i^{max}, while functions g_j represents nonlinear constraints. In the further text, we denote the lengths of the vectors $\boldsymbol{p}$ and $\boldsymbol{Y}$ by m and n respectively. The length of the state vector n is generally a large number, proportionate to the mesh size employed. The number of the parameters m, on the other hand, is typically much smaller: $n \gg m$.

To solve the optimization problem, the method of moving asymptotes (Svanberg, 2002) implemented in NLopt optimization package (Johnson, 2015) was employed. Since the method is gradient-based, the CFD solver has to facilitate the evaluation of not only the cost function at a given point in the parameter space, but also its derivatives with respect to the parameters. This was done via a direct sensitivity approach (Gunzburger, 2003). For each parameter vector $\boldsymbol{p}$ consider the the corresponding solution $\boldsymbol{Y}(\boldsymbol{p})$ of the steady-state PDE, so that

$$\boldsymbol{G}(\boldsymbol{Y}(\boldsymbol{p}), \boldsymbol{p}) = 0 \tag{5.4}$$

holds. We are interested in the value of the objective function $J(\boldsymbol{Y}, \boldsymbol{p}) = \hat{J}(\boldsymbol{p})$ and its derivatives with respect to the set of parameters $\frac{d\hat{J}}{d\boldsymbol{p}}$. We may use the chain rule to obtain

$$\underbrace{\frac{d\hat{J}}{d\boldsymbol{p}}}_{1 \times m} = \underbrace{\frac{\partial J}{\partial \boldsymbol{Y}}}_{1 \times n} \underbrace{\frac{\partial \boldsymbol{Y}}{\partial \boldsymbol{p}}}_{n \times m} + \underbrace{\frac{\partial J}{\partial \boldsymbol{p}}}_{1 \times m} . \tag{5.5}$$

The partial derivatives $\frac{\partial J}{\partial \boldsymbol{Y}}$ and $\frac{\partial J}{\partial \boldsymbol{p}}$ can be calculated by hand, when the cost function is defined in terms of the solution vector and the parameters. The term $\frac{\partial \boldsymbol{Y}}{\partial \boldsymbol{p}}$ is computed from a system of linear equations that is obtained by taking a derivative of Eq. (5.4) and using chain rule again,

$$\frac{d\boldsymbol{G}}{d\boldsymbol{p}} = 0 \tag{5.6}$$

$$\underbrace{\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{Y}}}_{n \times n} \cdot \underbrace{\frac{\partial \boldsymbol{Y}}{\partial \boldsymbol{p}}}_{n \times m} = - \underbrace{\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{p}}}_{n \times m} . \tag{5.7}$$

Terms $\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{Y}}$ and $\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{p}}$ are calculated using finite differencing. The i-th columns of the matrices are evaluated as

$$\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{Y}_i} = \frac{\boldsymbol{G}(\boldsymbol{Y} + \boldsymbol{\epsilon}_i, \boldsymbol{p}) - \boldsymbol{G}(\boldsymbol{Y}, \boldsymbol{p})}{\epsilon},$$
$$\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{p}_i} = \frac{\boldsymbol{G}(\boldsymbol{Y}, \boldsymbol{p} + \boldsymbol{\epsilon}_i) - \boldsymbol{G}(\boldsymbol{Y}, \boldsymbol{p})}{\epsilon}, \tag{5.8}$$

where $\boldsymbol{\epsilon}_i$ is a zero vector of appropriate length with ϵ at i-th position.

In summary, to compute the derivatives of the cost function with respect to the parameters, we need to:

- Calculate $\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{Y}}$ and $\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{p}}$ using finite differencing (Eqs. (5.8)). In total, $n + m + 1$ evaluations of function $\boldsymbol{G}$ is needed.

- Solve m systems of linear equations of size $n \times n$ (Eq. (5.7)).

- Perform matrix-vector product and summation in Eq. (5.5).

5.3.2 Case specific settings

In this study, the optimization cost function J was set to be the value of the pollutant concentration at $x = 250\,\text{m}$ from the inlet at height 2 m. Vector of parameters $\boldsymbol{p} = (x_1, x_2, \text{LAI})$ consisted of starting and end point of the vegetation block and its leaf area index. Following constraints were placed on the parameters:

- Position of the vegetation: $x_{min} \leq x_1 \leq x_2 \leq x_{max}$, where $x_{min} = 50\,\text{m}$ and $x_{max} = 150\,\text{m}$.

- Maximal leaf area index: $0.0 \leq \text{LAI} \leq \text{LAI}_{max}$ with $\text{LAI}_{max} = 9.0$.

- Maximal total amount of trees to be planted: $(x_2 - x_1)\text{LAI} \leq \text{VEG}_{max}$, where $\text{VEG}_{max} = 270.0$. That could represent for example a forest of length 30 m and LAI 9 or of length 100 m and LAI 2.7.

The leaf area density profile was set to be uniform for simplicity.

5.4 Results

Since our method searched only for a local minimum, three different initial points of the optimization process were used to rule out a possibility that only a local minimum in the vicinity of a initial position was found. The optimization procedure ended in the same point for all of the initial points. The initial configurations and corresponding solutions are listed in Tab. 5.1 and schematically depicted in Fig. 5.2.

Variant	Initial point	Solution	J (Initial)	J (Final)	#Evaluations
A	(90.0, 110.0, 4.5)	(50.0, 150.0, 0.810)	0.0407	0.0338	39
B	(80.0, 110.0, 6.75)	(50.0, 150.0, 0.810)	0.0419	0.0338	45
C	(60.0, 90.0, 8.1)	(50.0, 150.0, 0.810)	0.0402	0.0338	67

Table 5.1: Three initial variants and corresponding solutions. The initial and final points are listed in the form of the parameter vector $p = (x_1, x_2, \text{LAI})$.

Figure 5.2: Schematic depiction of the initial and final variants. The placement of the colored block corresponds to the position in the interval $[x_{min}; x_{max}] = [50; 150]$, the saturation of the block reflects the LAI of the vegetation.

The optimized variant represented a sparse vegetation block spanning the whole allowed interval. The obtained LAI = 0.81 lies well below the value given by the constraint on the maximal amount of trees planted, which allowed for a LAI = 2.7 for a block spanning the whole interval.

As evident from the Tab. 5.1, the cost function (i.e. the concentration behind the barrier) was reduced by 15% - 20% in all three cases. This reduction is further visible on Fig. 5.3, panel A, where the vertical profiles of the particle concentration at $x = 250$ m are shown. Three initial variants and the final variant are complemented by a variant with no vegetation present.
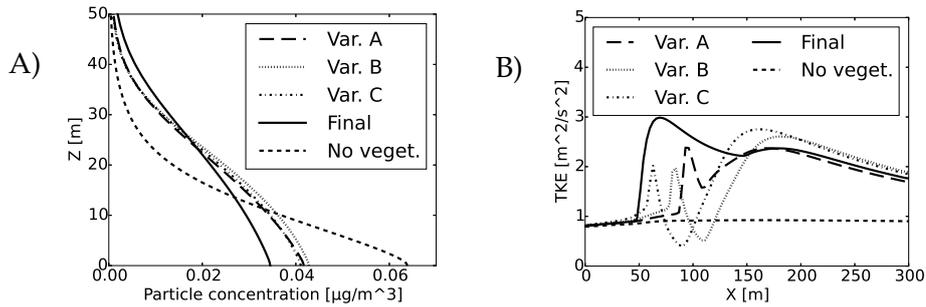
Figure 5.3: Optimization of the vegetation barrier. (A) Vertical profile of particle concentration at $x = 250$ m. (B) Horizontal profile of turbulence kinetic energy at height 10 m. Variant without any vegetation is included for reference.

	Variant A	Variant B	Variant C	Final variant
Deposition on the vegetation	2.88%	4.30%	5.51%	2.43%
Deposition on the ground	2.88%	2.95%	2.75%	2.32%

Table 5.2: Percentage of the injected pollutant deposited on the vegetation and on the ground.

Table 5.2 shows that less than 10% of the injected pollutant was deposed either on the ground or on the vegetation in all cases, and less than 5% in the optimized variant. The rest was redistributed to the higher layers of the atmosphere, where the higher velocity of the flow allowed for faster dilution. Therefore, the most important effect of the sparse vegetation here is the disturbance of the flow, leading to the increased levels of turbulence and increased turbulent diffusion, which results in faster redistribution to the higher layers. This is demonstrated on the right panel of Fig. 5.3, where the horizontal profiles of the turbulence kinetic energy are shown for all variants, and on Figures 5.4 and 5.5, where the velocity magnitude and turbulence kinetic energy plots are shown for the initial variant A and the optimized variant.
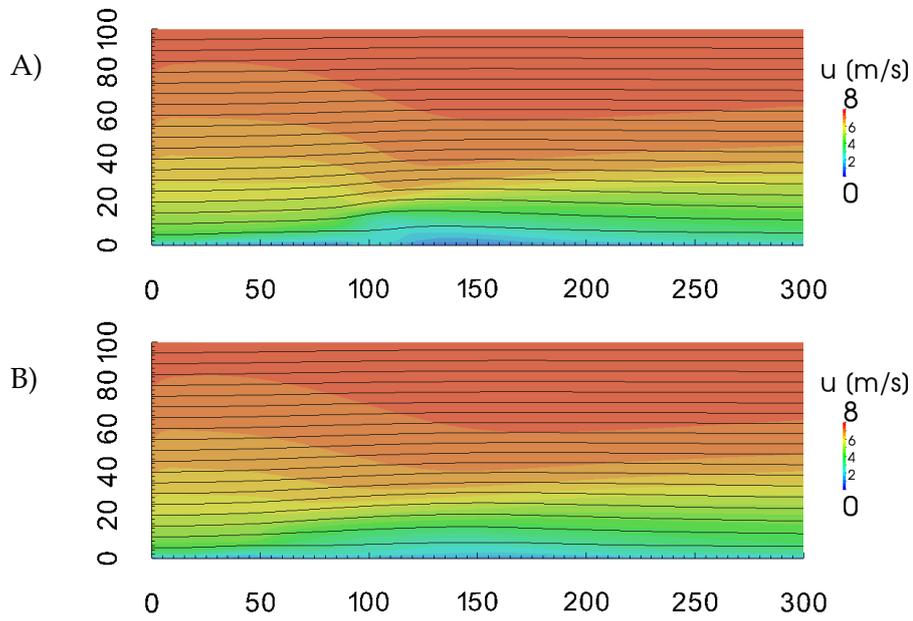
Figure 5.4: Plots of the velocity magnitude and velocity streamlines. (A) Initial variant A. (B) Optimized variant.
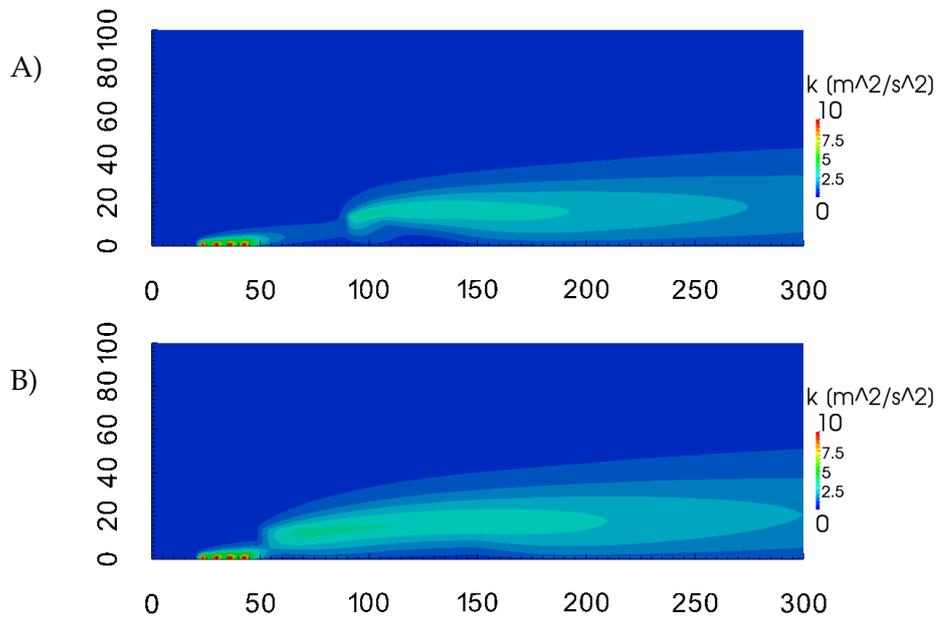
Figure 5.5: Plots of the turbulence kinetic energy. (A) Initial variant A. (B) Optimized variant.

5.5 Discussion

We have presented a method for optimization of the parameters of the vegetation barriers, and find an optimal barrier under the given constraints. Our discussion will focus on two aspects: on the obtained results first, and on the employed method itself second.

The optimized variant represented a sparse vegetation, and relied mainly on the dilution of the pollutant due to the flow disturbed by the vegetation. It is noteworthy that less pollutant was deposited inside the vegetation and on the ground in the optimized variant than in all the initial variants. This is a result of the given objective, which was to reduce the pollutant levels behind the barrier. Other reasonable objectives may give very different results; for example, we may rather prefer to maximize the amount of the pollutant deposited in the vegetation. These objectives were however not studied here.

As for the method itself, there are several shortcomings. First, it is suitable only for a limited number of parameters. In the current implementation when 100 parameters are optimized the amount of time for the CFD solution in every step of the optimization loop is roughly equal to the time needed for the gradient evaluation. For higher number of parameters it would be therefore more suitable to use the adjoint method for the gradient calculation. Secondly, our method optimizes only for a single target, while in reality we may be interested in several targets at once. To take that into account, multi-objective optimization should be employed. Lastly, optimization procedure searched only for the local minimum. Here we have used multiple initial points to assess whether we have found the global minimum, however, such approach is not sufficiently rigorous and could be difficult to apply when higher number of parameters is used.

Chapter 6

Moment method aerosol transport solver

The content of this chapter is based on: Šíp, V. and Beneš, L. (2016c). Dry deposition model for a microscale aerosol dispersion solver based on the moment method. *arXiv e-print*. https://arxiv.org/abs/1605.03397 (Submitted).

This (to a large extent self-contained) chapter is devoted to an alternative approach to the pollution dispersion modelling called the moment method. This method is useful when the behaviour of particles in a wide size range is of concern, as opposed to only particles of the same diameter. We describe the mathematical background of the moment method, and give a special attention the formulation of the dry deposition model. Applicability of the method is tested on two example problems of particle dispersion in the presence of a vegetation on small scales: a flow through a tree patch in 2D and a flow through a hedgerow in 3D.

6.1   Introduction

In the previous chapters, we have described the methods for modelling the pollutant dispersion, and dispersion of solid particles in particular. The assumption was that the investigated particles are of a known diameter. When behaviour of several classes of particles was of concern (such as in the validation case described in Sec. 4.4), multiple PDEs had to be solved. In real life applications, such as the modelling of a road traffic emissions dispersion, the situation is however more complex. The sizes of emitted particles form a continuous distribution, and an appropriate method must be used to deal with it.

The straightforward approach - so called *sectional* approach - is to divide the size range into a number of discrete bins and then model the appropriate number of scalar PDEs, i.e. one for each bin. Other option is to use the transport equation for the moments of the particle size distribution. Such approach can reduce the number of PDEs to be solved, and therefore reduce the computational demands. This class of methods, here referred to as the *moment method*, has been used for the simulation of the aerosol behaviour

for a long time (Whitby et al., 1991).

Usage of the moment method for the atmospheric aerosols modelling is widespread in large-scale models (e.g. Binkowski and Shankar, 1995; Pirjola et al., 1999; Jung et al., 2003), however, it is relatively uncommon in small-scale CFD models of atmospheric flows. Notable difficulty for the applications where the vegetation is present is the dry deposition model. Adapting the deposition velocity models to the moment method framework is not straightforward, since the mathematical formulation of the moment method requires all terms in the equation to be in the form of the power law of the particle size. Binkowski and Shankar (1995) simplified the problem by using the resistance model with Brownian particle diffusivity and settling velocity averaged over the particle size range. Bae et al. (2009) developed a deposition velocity model based on the model proposed by Raupach et al. (2001). This model, however, only includes the processes of Brownian diffusion, impaction and gravitational settling, and does not take into account the processes of interception and turbulent impaction, which play an important role in the dry deposition process (Petroff et al., 2008b).

This chapter has three primary aims:

- To present the mathematical basis of the moment method,

- to adapt the deposition velocity model for the moment method,

- and to asses the accuracy and performance of the developed methods by comparison of the moment method to the sectional approach.

This chapter is an exception from the rest of thesis in that the mathematical methods described here are implemented in the OpenFOAM framework, and not in our developed software as in the other chapters. The reason for this decision was the fact that the high-level OpenFOAM programming interface provided the necessary flexibility to implement the proposed methods, and allowed for a rapid development of the solver.

6.2 Mathematical formulation

6.2.1 Number concentration equation

The governing equation for the transport and the deposition of the aerosol particles of a diameter d_p in the flow field given by the velocity $\boldsymbol{u}$ can be formulated as

$$\frac{\partial n_N(d_p)}{\partial t} + \underbrace{\mathrm{div}\,(n_N(d_p)\boldsymbol{u})}_{\text{Convection}} = \underbrace{\mathrm{div}\,(D\nabla n_N(d_p))}_{\text{Diffusion}} - \underbrace{\mathrm{div}\,(n_N(d_p)\boldsymbol{u}_s(d_p))}_{\text{Gravitational settling}} - \underbrace{\mathrm{LAD}u_d(d_p)n_N(d_p)}_{\text{Dry deposition}},$$

(6.1)

where $n_N(d_p)$ is the number concentration of the particles with size in the infinitesimal interval from d_p to $(d_p + \mathrm{d}d_p)$, given in $\mathrm{m}^{-1}\,\mathrm{m}^{-3}$, and $D = \nu_T/Sc_T$ is the diffusion coefficient. This equation may be obtained by dividing Eg. (2.63) by the particle density and including the gravitational settling term (2.45) and the dry deposition sink term (2.86).

As the equation is formulated for the number concentration of the particles of given diameter d_p, the dependence of the settling velocity $\boldsymbol{u_s}$ and the deposition velocity u_d on

the diameter is explicitely indicated, while the dependencies on other flow parameters or particle properties are not.

6.2.2 Moment equations

The moment method is based on the idea that in order to model the size distribution of the particles, we can investigate the behaviour of the moments of the distribution. Moment of the distribution is defined as

$$M_k = \int_0^\infty d_p^k n_N(d_p) \mathrm{d}d_p, \tag{6.2}$$

where k is the order of the moment. Some moments have straightforward physical interpretation:

- $M_0 = \int_0^\infty n_N(d_p)\mathrm{d}d_p = N_T$ is the total number concentration,

- $M_2 = \int_0^\infty d_p^2 n_N(d_p)\mathrm{d}d_p = 1/\pi \cdot S_T$ is proportionate to the surface area concentration,

- $M_3 = \int_0^\infty d_p^3 n_N(d_p)\mathrm{d}d_p = 6/\pi \cdot V_T$ is proportionate to the volume concentration.

Assuming $n_N(d_p)$ is sufficiently smooth in space and time, moment equations are obtained by multiplying Eq. (6.1) by d_p^k, integrating over the whole size range and interchanging the derivatives and the integrals:

$$\frac{\partial M_k}{\partial t} + \underbrace{\mathrm{div}\,(M_k \boldsymbol{u})}_{\text{Convection}} = \underbrace{\mathrm{div}\,(D\nabla M_k)}_{\text{Diffusion}} - \underbrace{\int_0^\infty d_p^k \mathrm{div}\,(n_N(d_p)\boldsymbol{u}_s(d_p))\mathrm{d}d_p}_{\text{Gravitational settling}}$$
$$- \underbrace{\mathrm{LAD} \int_0^\infty d_p^k u_d(d_p)n_N(d_p)\mathrm{d}d_p}_{\text{Deposition}}. \tag{6.3}$$

Now we are left with the evaluation of the integrals in (6.3). This can be done easily if the multiplicative terms are in a form of a polynomial function of d_p. Such is the case with the gravitational term, if we take into account that gravity plays significant role only for larger particles, where the Cunningham correction factor C_C in (2.46) can be left out. Using this formula in the second term on the RHS of (6.3), the term can be rewritten as

$$-\mathrm{div}\left(\boldsymbol{g}\frac{\rho_p}{18\mu}\int_0^\infty d_p^{k+2}n_N(d_p)\mathrm{d}d_p\right) = -\mathrm{div}\left(\boldsymbol{g}\frac{\rho_p}{18\mu}M_{k+2}\right). \tag{6.4}$$

Here we introduced a dependence on the moment of a higher order. That necessitates that we either solve a separate moment equation also for this higher order moment, or that this moment can be calculated from the moments that we solve for.

The task of integrating the deposition term is more difficult and will be examined in the following section.

6.2.3 Deposition model for the moment method

Recall from Sec. 2.5.2 that under the employed dry deposition model from (Petroff et al., 2008b) and (Petroff et al., 2009) the deposition velocity may be written as

$$u_d = 2(u_{BD} + u_{IN} + u_{IM} + u_{TI} + u_{SE}). \tag{6.5}$$

The assumption of the parallel and independent acting is advantageous for adapting the model to the moment method, since it allows us to split the rightmost integral in Eq. (6.3) into integrals pertaining to the every physical process separately.

Even so, the analytical evaluation of some of these integral is not possible. It is therefore necessary to approximate the deposition velocity with an expression more amenable to the integration. Such approximation is detailed in Appendix C. Its result is formally written as

$$u'_d = 2(u'_{BD} + u'_{IN} + u'_{IM} + u_{TI} + u_{SE}), \tag{6.6}$$

where the prime marks an approximation of the original term. The major approximations made are:

- The Cunningham approximation factor is replaced by its size-dependent part in u_{BD}.

- An approximate power law term is used in place of a logarithm term in u_{IN} for broafleaf elements.

- The impaction efficiency in u_{IM} is approximated by a piecewise linear function of Stokes number.

Comparison of this model with the original for an exemplary set of parameters is shown on Fig. 6.1. The higher values of the deposition velocity for particles around $3\,\mu$m are the consequence of the inexact approximation to the inertial impaction term described in Appendix C.

The maximal difference of the deposition velocity given by the two models was determined by evaluating the deposition velocity for every combination of the parameters in the ranges expected in real-world situations ($\rho_p \in [500; 3000]\,\text{kg}\,\text{m}^{-3}$, $U \in [0, 10]\,\text{m}\,\text{s}^{-1}$, $d_p \in [10^{-3}, 10^2]\,\mu$m, and $d_e \in [0.5; 5]$ mm for needles or $d_e \in [1; 5]$ cm for broadleaves). Each interval was discretized using 50 points. Local friction velocity u_f was set to $0\,\text{m}\,\text{s}^{-1}$, as the turbulent impaction is implemented exactly and its contribution can only reduce the relative difference of the deposition velocities.

The largest relative difference $|u_d^{orig} - u_d^{approx}|/\min(u_d^{orig}, u_d^{approx})$ was found to be 1.29 for the needle model and 1.31 for the broadleaf model (Tab. 6.1). The maximum is obtained at the end of the expected ranges for all parameters except d_p.

This difference was considered acceptable considering that measured values shows much higher variability (Litschke and Kuttler, 2008).

With the approximated deposition velocity we may proceed with the integration of the last term in Eq. (6.3),

$$\left(\frac{\partial M_k}{\partial t}\right)_{deposition} = -\text{LAD} \int_0^\infty d_p^k u'_d(d_p) n_N(d_p) \mathrm{d}d_p = S_{dep,k}. \tag{6.7}$$
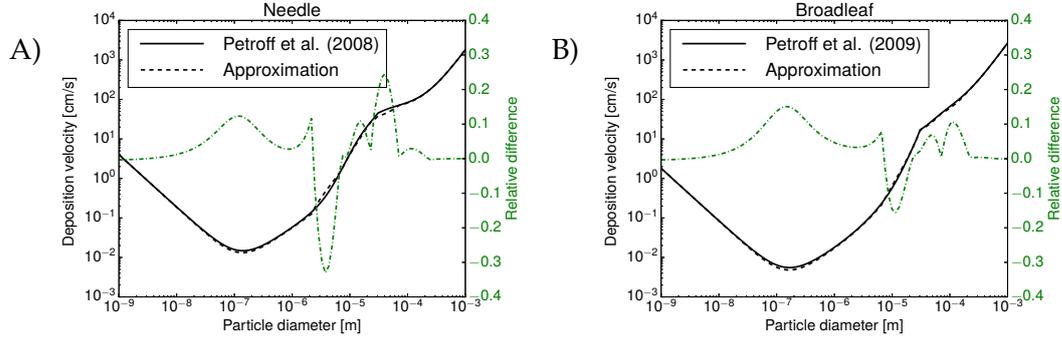
Figure 6.1: Comparison of the deposition velocities given by the original model and the approximation developed in this study for exemplary set of parameters: $\rho_p = 1300\,\mathrm{kg\,m^{-3}}, U = 1\,\mathrm{m\,s^{-1}}, u_f = 0.3\,\mathrm{m\,s^{-1}}$. (A) Needles, $de = 2\,\mathrm{mm}$, (B) Broadleaves, $de = 2\,\mathrm{cm}$. Relative difference is calculated as $(u_d^{orig} - u_d^{approx})/\min(u_d^{orig}, u_d^{approx})$.

	Needles	Broadleaves
Relative difference	1.29	1.31
u_d^{orig}	$0.385\,\mathrm{cm\,s^{-1}}$	$0.387\,\mathrm{cm\,s^{-1}}$
u_d^{approx}	$0.168\,\mathrm{cm\,s^{-1}}$	$0.168\,\mathrm{cm\,s^{-1}}$
ρ_p	$3000\,\mathrm{kg\,m^{-3}}$	$3000\,\mathrm{kg\,m^{-3}}$
d_p	$0.72\,\mathrm{\mu m}$	$2.33\,\mathrm{\mu m}$
U	$10\,\mathrm{m\,s^{-1}}$	$10\,\mathrm{m\,s^{-1}}$
d_e	$4.72\,\mathrm{mm}$	$4.92\,\mathrm{cm}$

Table 6.1: Maximal relative differences of the deposition velocities given by the original and the approximate model and the parameters of the model at the maximum.

Result of the integration is also given in Appendix C and is not repeated here for the sake of brevity. The sink term in the equation of k-th moment $S_{dep,k}$ depends on several moments of different orders. In the next section we will show that when we work with the lognormal distributions, all moments can be calculated from three known moments or arbitrary orders k_0, k_1 and k_2. In that case, the dependency of the sink term may be explicitly written as $S_{dep,k} = S_{dep,k}(M_{k_0}, M_{k_1}, M_{k_2})$.

6.2.4 Lognormal distribution

Before we move on to the description of the implementation, it is necessary to provide some assumptions on the particle size distribution. Size distributions of the atmospheric aerosols are often well fitted by a multimodal lognormal distribution (Seinfeld and Pandis, 2006). This is the distribution we will use from now on. We restrict ourselves only to the case of unimodal distribution, noting that the multimodal distribution can be modelled

by a superposition of several unimodal distributions.

Unimodal lognormal distribution can be described by three parameters: total number concentration N_T, geometric mean size d_{gn} and geometric standard deviation σ_g. Its probability density function is

$$n_N(\ln d_p) = \frac{N_T}{\sqrt{2\pi}\ln\sigma_g}\exp\left(-\frac{(\ln d_p - \ln d_{gn})^2}{2\ln^2\sigma_g}\right). \tag{6.8}$$

Knowing the three parameters, k-th moment can be calculated using the formula

$$M_k = N_T d_{gn}^k \exp\left(\frac{k^2}{2}\ln^2\sigma_g\right). \tag{6.9}$$

From the three moments of order 0, k_1 and k_2 the three parameters can be obtained using the relations

$$N_T = M_0, \tag{6.10}$$

$$d_{gn} = \overline{M}_{k_1}^{\frac{1}{r(k_2-k_1)}}\overline{M}_{k_2}^{\frac{r}{k_1-k_2}}, \tag{6.11}$$

$$\ln^2\sigma_g = \frac{2}{k_1(k_1-k_2)}\ln\left(\frac{\overline{M}_{k_1}}{\overline{M}_{k_2}^r}\right), \tag{6.12}$$

where $\overline{M}_k = \frac{M_k}{M_0}$ and $r = \frac{k_1}{k_2}$ (Whitby and McMurry, 1997).

For the incomplete higher order moments following holds:

$$M_k^-(x) = \int_0^x d_p^k n_N(d_p)\mathrm{d}d_p = M_k\Phi\left(\frac{\ln x - \ln d_{gn} - k\ln^2\sigma_g}{\ln\sigma_g}\right), \tag{6.13}$$

$$M_k^+(x) = \int_x^\infty d_p^k n_N(d_p)\mathrm{d}d_p = M_k\left(1 - \Phi\left(\frac{\ln x - \ln d_{gn} - k\ln^2\sigma_g}{\ln\sigma_g}\right)\right), \tag{6.14}$$

where Φ is the normal cumulative distribution function.

6.2.5 Choice of the moments

Now we turn our attention to the choice of the moments. For which orders we decide to solve the moment equation (6.3) is to a degree an arbitrary decision. When this problem is discussed in literature, cited reasons for a certain choice include the mathematical simplicity and ease of the formulation of the modelled processes or the physical interpretation of some moments (Whitby and McMurry, 1997; Binkowski and Shankar, 1995). Choices of the moments used in the field of atmospheric aerosol modelling in the selected literature are summarized in Tab. 6.2.

The recurrent usage of zeroth order moment brings substantial advantage, as it is equal to the total number concentration, and it is the order we will use as well. On the choice of the other moments authors differ.

To assess the influence of the choice of the moments, following numerical experiment was performed. We investigated the particle deposition in a 1D tube, spanning between 0

Reference	Moments
(Binkowski and Shankar, 1995)	0, 3, 6
(Pirjola et al., 1999)	0, 2, 3
(Jung et al., 2003)	0, 2, 3
(Koziol and Leighton, 2007)	0, 1, 2
(Bae et al., 2009)	0, 3, 6

Table 6.2: Choices of the moments in the selected literature

and 300 m. Homogenous vegetation block of $\mathrm{LAD} = 1.5\,\mathrm{m}^2/\mathrm{m}^3$ was placed between 100 a 150 m. Velocity of the air in the whole tube was set to constant $1\,\mathrm{m\,s}^{-1}$, unaffected by the vegetation. Source of the pollutant was placed at 50 m from the inlet with the intensity of number of particles $1\,\mathrm{s}^{-1}$ and the distribution parameters $\sigma_g = 0.7, d_{gn} = 3\mathrm{\mu m}$. The tube was discretized using 400 cells.

Beside the choices mentioned in Tab. 6.2, we tested also a variant with a negative order moment: 0, -1, 1. Non integer choices of the orders would also be possible to use, but we saw no advantage that such choice could bring.

Transport and the deposition of the pollutant was calculated by the sectional model based on the Eq. (6.1) and by the moment method based on the Eq. (6.3) (see section 6.2.6 for details on the implementation). To discard possible errors due to the inexact approximation of the deposition velocity, only the sedimentation contribution, adapted exactly, was taken into account. The numerical experiment is not meant to model any real-world situation, rather just demonstrate the behaviour of the moment method in a simple setting.

Number and volume concentration distribution behind the barrier (at 150 m) are shown on Fig. 6.2. As a reference, calculated distributions are complemented by the
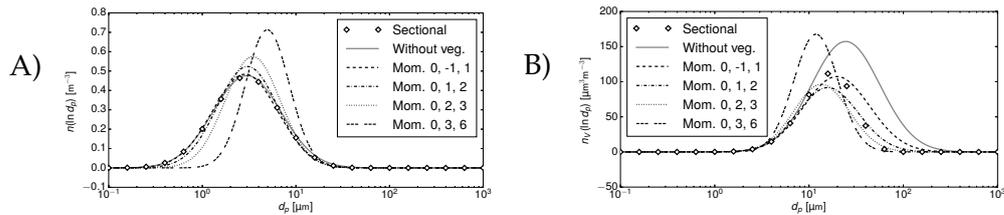
Figure 6.2: (A) Number concentration (B) Volume concentration

distribution for a case without the vegetation present. Evolution of the zeroth moment (equal to the number concentration) and the third moment (proportional to the volume concentration) through the vegetation block are shown on Fig. 6.3.

Effect of the vegetation, while small in number concentration, is significant in volume concentration. Only the variant using the moments of orders 0, -1, and 1 reproduces well the number concentration distribution, but overpredicts the peak of the volume concentration. Variants using the orders 0, 1, 2 and 0, 2, 3 produce result closer to the sectional model in volume concentration, but with larger differences in number concen-
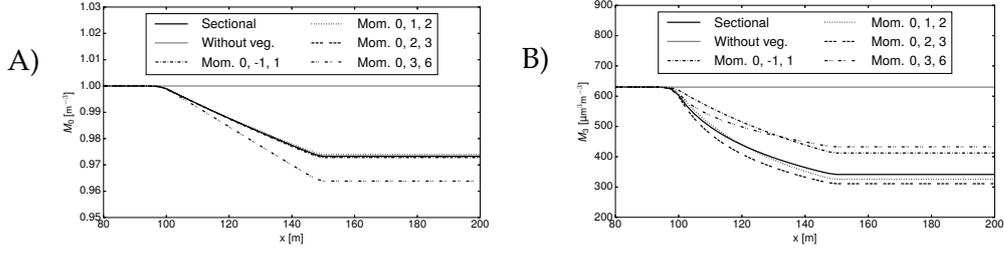
Figure 6.3: Evolution of the moments along 1-dimensional the tube. (A) Zeroth moment (B) Third moment.

tration. Variant using the orders 0, 3, 6 shows no advantages over the other variants.

Choosing between the orders 0, 1, 2 and 0, 2, 3, we opted for the latter variant, as the third moment is proportionate to the main quantity of interest - volume (and mass) concentration of the pollutant.

6.2.6 Numerical implementation

Both the sectional model and the moment model were implemented using the Open-FOAM platform (Greenshields, 2015). Second order upwind scheme was used for convective terms in Equations (6.1) and (6.3) and second order scheme based on the Gauss theorem was used for the diffusive terms. Residual levels of 10^{-5} were used to test for convergence of the steady state solver.

When using the moment method, we have to solve the discretized Eq. (6.3) for the three selected moments. These equations are coupled through the gravitational settling term and the deposition term, which depends on the moments of a different order than the one solved by the equation. The coupling is dealt with the following way. In every iteration, first the parameters of the lognormal distribution N, d_{gn} and σ_g are calculated using the Equations (6.10-6.12) from the values in the preceeding iteration. Three moment equations are then solved one after another with the coupling terms resulting from the deposition being treated explicitly.

Fully explicit treatment of the gravitational settling term (6.4) can result in numerical instability, unless low values of the relaxation factors are used. That would however lead to slower convergence, therefore we employed a semi-implicit treatment. Moment M_{k+2} in (6.4) is rewritten as $M_{k+2} = F_{k,2}M_k$ with

$$F_{k,m} = M_{k+m}/M_k = d_{gn}^m \exp\left(\frac{m(m+2k)}{2}\ln^2\sigma_g\right) \tag{6.15}$$

and the term $F_{k,2}$ is then treated explicitly and M_k implicitly.

Relaxation factors 0.95 were used both for the sectional equations and for the moment equations. For the first five iterations of the moment method the relaxation factors for the moment equations were however set to lower value 0.8, as the computations proved to be less stable at the beginning.

Calculation of the distribution parameters d_{gn} and $\ln^2\sigma_g$ via Eq. (6.11) and (6.12) includes the division of the moments, potentially very small far away from the source of

pollutant. To avoid this problem, small background concentration in the whole domain is set as an initial condition and used as a boundary condition where zero would be used otherwise.

6.3 Applications

Here we describe two example problems of microscale flows through and around the vegetation and assess the applicability of the developed moment method to the simulation of pollutant dispersion. Two vegetation elements that could be encountered in the urban settings are investigated in this test: small patch of full grown trees and a dense hedgerow.

   The flow field in both cases was precomputed by the finite volume CFD solver based on the equations for atmospheric boundary layer flows (2.39), as described in preceeding chapters. The solver utilizes the standard k-ϵ turbulence model presented in Sec. 2.3.4. Inlet profiles of velocity and the turbulence quantities, as well as the wall functions, are prescribed by the analytical expressions given by Richards and Hoxey (1993). Vegetation model for the momentum and k-ϵ equations described in Sec. 2.5.1 is employed. Turbulent Schmidt number was set to $Sc_T = 0.7$ in both cases, based on the analysis by Tominaga and Stathopoulos (2007).

   In both cases presented below, we simulated the dispersion of a coarse mode particles from a point or a line source. The coarse mode is chosen as the mode that contains, together with the accumulation mode, majority of the volume of the particles in the urban environment (Seinfeld and Pandis, 2006), but is affected more strongly by the dry deposition than the accumulation mode. The number distribution at the source is assumed to be lognormal with the parameters $d_{gn} = 0.86\mu$m and $\sigma_g = 2.21$, typical for the urban environment (Hinds, 1999).

   Evaluation of the developed moment method was based on the comparison with the results obtained by the sectional model. In the sectional model, Eq. (6.1) is solved for 41 particle sizes distributed uniformly between $0.01\,\mu$m and $100\,\mu$m. The interval is chosen so that the behaviour of the number distribution as well as the volume distribution can be captured by the sectional model.

6.3.1 Tree patch in 2D

First case investigates the filtering properties of a small patch of full grown conifer trees. A simplified 2D model is constructed as follows. The 30 meters wide and 15 meters high tree patch is represented as a horizontally homogeneous vegetation block. Pollutant source is placed 15 meters upstream from the vegetation, 5 meters above the ground.

   LAD profile of the vegetation is prescribed by a formula given by Lalic and Mihailovic
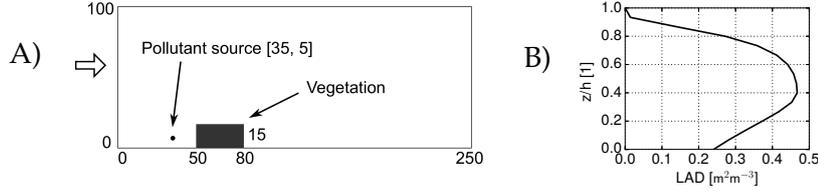
Figure 6.4: (A) Sketch of the domain. All dimensions in meters. Sketch is not to scale. (B) LAD profile of the vegetation.

(2004),

$$\mathrm{LAD}(z) = L_m \left(\frac{h - z_m}{h - z} \right)^n \exp\left(n \left(1 - \frac{h - z_m}{h - z} \right) \right), \tag{6.16}$$

$$n = \begin{cases} 6 & \text{if } 0 \leq z < z_m, \\ 0.5 & \text{if } z_m \leq < z \leq h, \end{cases}$$

where $h = 15$ m is the height of the trees, L_m is the maximum LAD, chosen so that leaf area index, LAI $= \int_0^h \mathrm{LAD}(z)\mathrm{d}z$, is equal to 5, and $z_m = 0.4h$ is the corresponding height of maximal LAD. The sketch of the domain and the LAD profile of the vegetation is shown on Fig. 6.4. Trees are modelled as generic conifers with $d_e = 2$ mm. The drag coefficient is chosen as $C_d = 0.3$ (Katul et al., 2004).

Intensity of the point source is set to a normalized value $1\,\mathrm{s}^{-1}$ in terms of number of particles. Since all terms in Eq. (6.1) and Eq. (6.3) are linear with respect to the number concentration, results can be simply scaled to other value of the source intensity if needed.

Inlet wind profile is set as logarithmic with $u_{\mathrm{ref}} = 10\,\mathrm{m\,s}^{-1}$ at height 20 m and $z_0 = 0.1$ m. For the number concentration in the sectional model and for all moments in the moment method the Neumann boundary conditions are used on the ground, at the top and at the outlet. No resuspension of the particles is allowed, i.e. any particle that falls on the ground stays on the ground indefinitely. Small value of the concentration and of the moments calculated from the lognormal distribution with the parameters $N_T = 10^{-6}\mathrm{m}^{-3}, d_{gn} = 0.86\,\mu\mathrm{m}, \sigma_g = 2.21$ is prescribed at the inlet.

Domain is discretized using a cartesian grid with 220 cells in horizontal direction and 100 cells in vertical direction, graded so that the grid is finer near the ground and around the tree patch. The near ground cells are 0.25 m high, and the vegetation block itself consists of 42 x 40 cells.

Flow field obtained by the CFD solver is shown on Fig. 6.5A. As visible, the vegetation block slows the wind down, but allows the air to pass through.

Results from the sectional and the moment model are compared in terms of the third moment of the particle size distribution, proportionate to the volume concentration of the particles. As we assume that the density is the same for particles of all sizes, third moment is also proportionate to the mass concentration of the particles.

Calculated field of the third moment by the moment method is shown on Fig. 6.5B. Fig. 6.5C shows the relative difference $(M_3^{mm} - M_3^{sec})/M_3^{mm}$ of the results obtained by
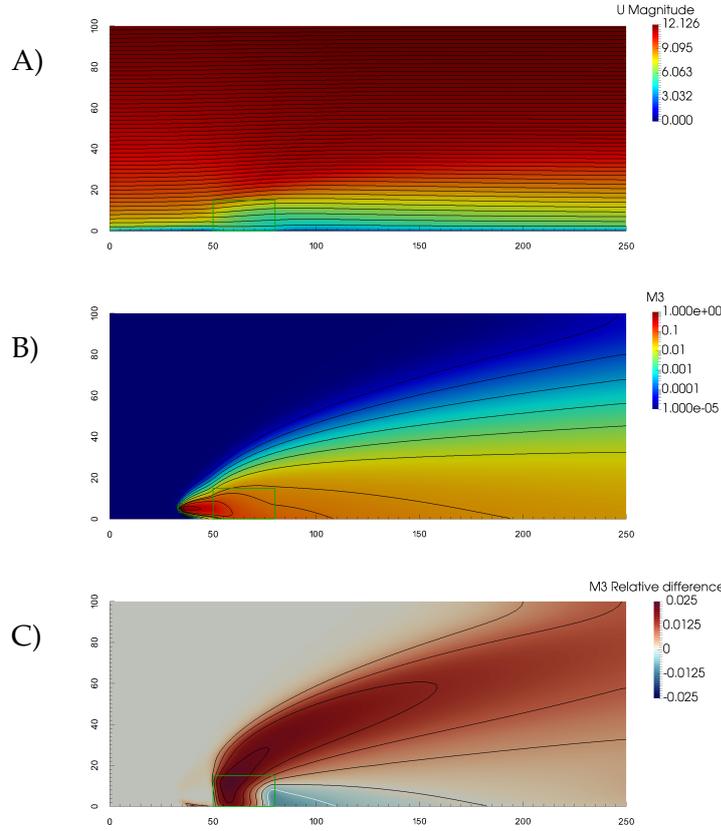
Figure 6.5: Results for the 2D tree patch case. Position of the tree patch is marked by a green rectangle. (A) Flow field. Shown are the streamlines, background is coloured by velocity magnitude (in $\mathrm{m\,s^{-1}}$). (B) Third moment of the size distribution calculated by the moment method (in $\mathrm{\mu m^3\,m^{-3}}$). (C) Relative difference $(M_3^{mm} - M_3^{sec})/M_3^{mm}$ of the third moment calculated by the moment method and the sectional approach.

the moment method, M_3^{mm}, and by the sectional model, M_3^{sec}. The source of the largest discrepancies between the two methods is the vegetation block. The relative difference raises up to 2.5% inside the vegetation block, and decreases with the increasing distance from the vegetation.

Further insights can be obtained from Fig. 6.6. It shows the volume concentration distribution at the top of the vegetation block in its first third, at the downstream edge of the tree patch, and at 120 m downstream from the tree patch, last two at height 2 m above ground. The vegetation has negligible effect on the particles smaller than $2\,\mathrm{\mu m}$, but significantly reduces the mass of the particles above $10\,\mathrm{\mu m}$. This is captured well both by the sectional and the moment method. At the downstream edge of the tree patch the moment method predicts lower peak of the volume concentration distribution than the sectional approach. The difference is slightly reduced by the mixing of the filtered air with the unfiltered air flowing above the vegetation further away from the tree patch.

It is noteworthy that the peak of the distribution, as calculated by the moment method,

91
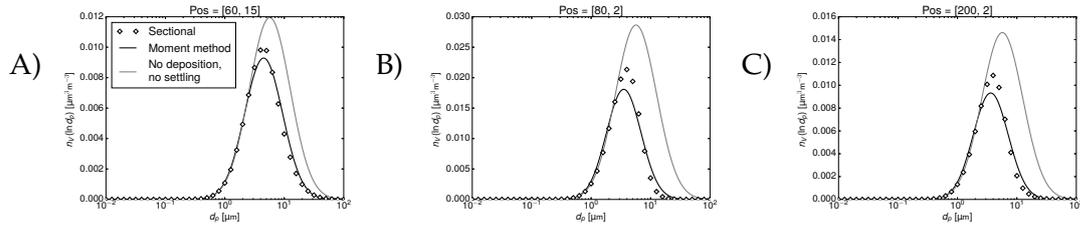
Figure 6.6: Results for the 2D tree patch case. (A) Volume concentration at [60; 15]. (B) Volume concentration at [80; 2]. (C) Volume concentration at [200; 2]. Discrete points calculated by the sectional method and the distribution calculated by the moment method are shown. For reference, the distribution calculated without the size dependent deposition and gravitational settling terms is shown as well.

is reduced even at the top of the vegetation block, where the total volume of the pollutant is predicted higher by the moment method (Fig. 6.6A). This may be explained by the fact that the size distribution at that point cannot be exactly fitted by the lognormal distribution anymore, and the moment method is thus bound to produce inaccurate results.

6.3.2 Hedgerow in 3D

Next we tested the method on a 3D model of a dense hedgerow placed near a line source of the pollutant. This case is a three dimensional extension of the 2D situation investigated in (Tiwary et al., 2005). The yew hedge is 10 m wide, 3.2 m deep and 2.4 m high. It is placed in the 40 m wide, 40 m long, and 20 m high computational domain. Two meters upstream from the hedge is a line source at height 0.5 m above ground. Intensity of the line source is set to a value $1\,\mathrm{s}^{-1}\,\mathrm{m}^{-1}$ in terms of number of particles, noting as in Sec. 6.3.1 that the results can be scaled if other value is desired.

Sketch of the domain is shown on Fig. 6.7A. Panel B shows the LAD profile of the hedge, taken from the original article. Vegetation is further described by the needle diameter is, $d_e = 3\,\mathrm{mm}$, and the vegetation drag coefficient which is set to $C_d = 0.5$ as in (Tiwary et al., 2005).
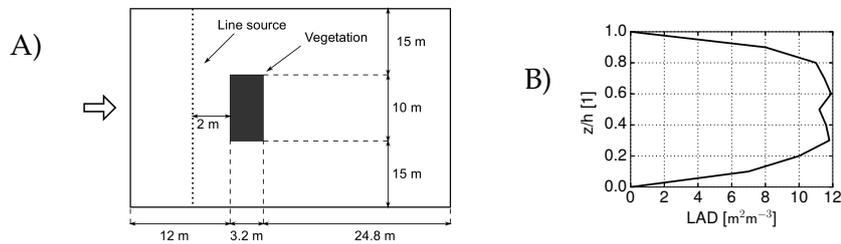
Figure 6.7: (A) Overhead view of the domain (not to scale). (B) LAD profile of the vegetation.

The computational mesh was created using the OpenFOAM *snappyHexMesh* generator. The domain consist of 376 000 cells, refined near the ground and around the hedge. The near-ground cells are 0.07 m high and the hedge itself is discretized using 54 x 20 x 22 cells.

The wind profile at the inlet is set as logarithmic with $u_{\mathrm{ref}} = 2.5\,\mathrm{m\,s^{-1}}$ at height 2.4 m and $z_0 = 0.1$ m. Boundary conditions for the sectional solver and moment method solvers are set similarly as in Sec. 6.3.1: Neumann boundary conditions are used at the ground, top, sides, and at the outlet. No resuspension of the particles fallen to the ground is allowed. Again, small amount of the particles given by the lognormal distribution with the parameters $N_T = 10^{-6}, d_{gn} = 0.86\,\mu\mathrm{m}, \sigma_g = 2.21$ is prescribed at the inlet.

Streamlines of the flow field calculated by the separate CFD solver are shown on Fig. 6.8. As in the 2D simulation in (Tiwary et al., 2005), recirculation zone is developed behind the dense hedge. Unlike the 2D case, here we can observe part the of the flow to be deflected to the sides.
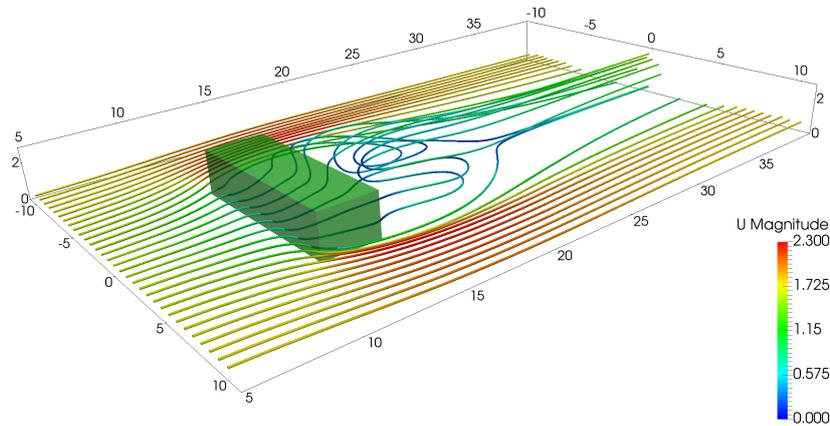
Figure 6.8: Streamlines of the flow around the hedgerow. Streamlines are released at height 0.5 m and are coloured by the velocity magnitude.

Third moment of the particle size distribution obtained by the moment method is shown on the left panels of Fig. 6.9 and Fig. 6.10. While a portion of the pollutant penetrates the barrier, part is deflected to the sides of the hedgerow, creating a zone with a reduced pollutant concentration behind it.

Relative difference between the solution obtained by the moment method and sectional approach is shown on the right panels of Fig. 6.9 and Fig. 6.10. As in the tree patch case, the moment method overpredicts the deposition and consequently underestimates the volume concentration behind the barrier. The difference is below 7%, and decreases away from the barrier.

Effects of the coarser mesh in the upper part of the computational domain are visible on Fig. 6.10. However, it does not negatively affect the difference between the two methods.

Volume concentration distribution at two points - inside the vegetation and downstream from the vegetation - is shown on Fig. 6.11. Due to the smaller size of the veg-
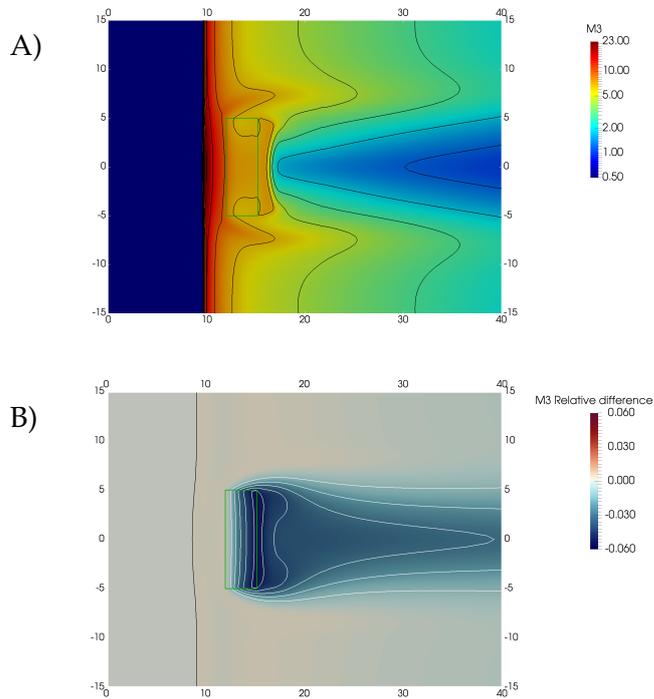
Figure 6.9: Results for the 3D hedgerow case. Horizontal cut at height $z = 0.5$ m. (A) Third moment of the size distribution calculated by the moment method (in $\mu m^3\, m^{-3}$). (B) Relative difference $(M_3^{mm} - M_3^{sec})/M_3^{mm}$ of the third moment calculated by the moment method and the sectional approach.
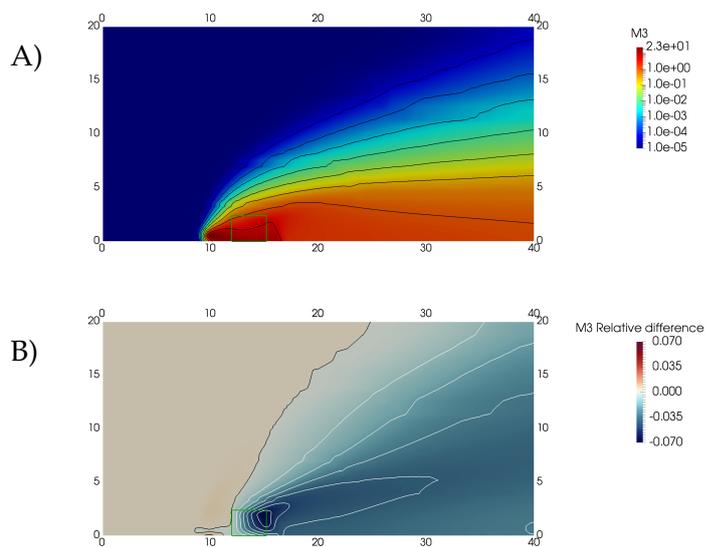
Figure 6.10: Results for the 3D hedgerow case. Vertical cut at $y = 0$ m. Quantities shown are as on Fig. 6.9.
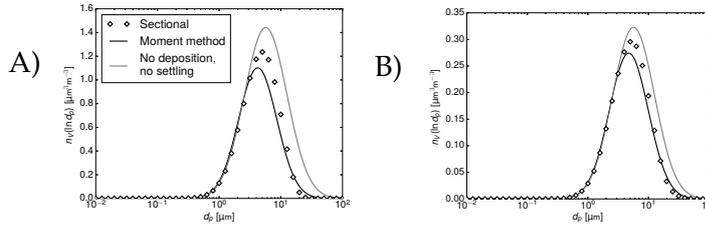
Figure 6.11: Results for the 3D hedgerow case. (A) Volume concentration at [15; 0; 2]. (B) Volume concentration at [30; 0; 2]. For reference, the distribution calculated without the size dependent deposition and gravitational settling terms is shown.

etation than in the 2D tree patch case, the effect of the vegetation is less pronounced. The moment method is able to reproduce the shape of the distribution well, but again produces a lower peak than the sectional method. Similarly as before, better fit can be observed further from the barrier due to the mixing with unfiltered air.

6.3.3   Computational performance

To compare the computational performance of the developed model, we measured the runtime of the sectional approach and the moment method approach for the 3D case described in section 6.3.2. Both solvers were run on a single core of an Intel Xeon X5365 processor.

The sectional model, comprised of 41 scalar PDEs, finished in 7015 seconds. The average runtime per each equation was thus 171 seconds. Moment model, comprised of 3 coupled PDEs, finished in 580 seconds. Time per one equation is therefore 193 seconds, only slightly above of the time needed for one equation of the sectional model. Overall, the use of the moment method give us more than tenfold acceleration compared to the sectional model using 41 bins. Even though the high number of bins used in this study might not be necessary to obtain sufficiently accurate results, the moment method still constitutes significant computational improvement for lower numbers of bins.

Two points regarding the computational performance can be made in favor of the sectional method though. First, the solution process of every equation is independent on the other equations, therefore the approach offers effortless parallelization for the number of cores up to the numbers of bins used. This is not especially advantageous in our implementation, as the OpenFOAM solvers are already parallelizable, but it could be an important factor for other implementations. Secondly, the relaxation factor 0.95 used for all simulations in the sectional approach was needed only for the bins representing the larger particles. Using different values of this parameter for different bins can provide some reduction of the runtime.

6.4   Conclusions

In this study, we introduced a formulation of a dry deposition model suitable for implementation in a moment method. As the original model by Petroff et al. (2008b), our

approximation includes five main processes of the dry deposition: Brownian diffusion, interception, impaction, turbulent impaction, and sedimentation.

The developed deposition velocity model was implemented in a microscale finite volume solver based on the OpenFOAM platform. The solver employs the moment method to calculate the particle size distribution in the domain. The deposition model was tested on two example problems of microscale pollutant dispersion. Comparison with the sectional method using the original dry deposition model revealed that the moment method is able to reproduce the shape of the particle size distribution well. The relative differences between the sectional and the moment method in terms of the third moment of the distribution were below 7%. The peak of the distribution as calculated by the moment method was always reduced compared to the peak calculated by the sectional method. We assume that this was because the particle size distribution is not guaranteed to stay lognormal under the influence of the dry deposition process, and the assumptions of the moment method are thus not satisfied. Nevertheless, the agreement with the sectional method is good despite this obstacle.

The moment method, described by three coupled PDEs, proved to be more computationally efficient than the sectional model using 41 bins. Above tenfold speedup was achieved, and the solution of each of the moment equations was only marginally more computationally demanding that the solution of one sectional equation.

This performance improvement together with the reliable results shows that the moment methods, often used in large scale atmospheric models, can be useful also for the microscale problems of pollutant dispersion in the urban environment.

The developed method as formulated here is applicable only when the particle size distribution can be approximated as a lognormal distribution. Here we used only unimodal distribution, but the usage of multimodal distribution would be also possible by superposition of several unimodal ones. Furthermore, the method could be reformulated for other distributions, provided that algebraic relations between the moments and distribution parameters are known.

Chapter 7

Conclusions

7.1 Achieved results

This work dealt with the numerical simulations of the atmospheric flows, and its main aim was to develop a method for the solution of the air flow and pollution dispersion problems in the urban areas with vegetation. Let us now summarize what was actually achieved.

- We developed a finite volume CFD solver based on the RANS turbulence modelling approach. The solver includes two fluid flow models: compressible flow and atmospheric boundary layer flow model based on the Boussinesq approximation. Fully implicit temporal integration scheme utilizing the Jacobian-free Newton-Krylov method provides an efficient way to achieve large time steps of the solver. The CFD solver features k-ϵ turbulence model, and the implemented passive scalar equation solver allows the solution of pollutant dispersion problems.

- Via the use of the AUSM$^+$-up numerical flux, suitable for low speed flows, and the low Mach preconditioning technique, the solver based on the compressible flow equations achieved a comparable accuracy and performance as the solver dedicated to ABL flows. This constitutes a promising result, as it shows that a generic compressible flow solver can be applied to ABL flow with only small changes, thus limiting the need for a dedicated solver. However, the comparison of the models was performed only for limited set of test cases due to the time constraint, and further comparison on different problems is advisable.

- The chosen approach of modelling the vegetation as a porous zone via the additional terms in momentum and turbulence equations is conceptually simple, and the selected k-ϵ vegetation model is shown to reproduce the flow inside and around the vegetation well in the simulated problems. Furthermore, the approach does not introduce significant computational overhead compared to the vegetation-free model, except for the need of finer mesh around the vegetation to capture the flow in necessary detail.

- We adopted a detailed, physically based dry deposition model not previously used in the microscale CFD simulations. The model proved to be capable of reproducing

97

the experimentally observed filtration effects, however, some fine-tuning of the vegetation parameters is needed due to the uncertainty of the vegetation parameters.

- The developed solver was used to investigate the optimal parameters of a near-road vegetation barrier when considering its effects on the pollution levels behind the barrier. The solver was used as the crucial component of the optimization loop which sought for the optimal position and density of the barrier. The optimized variant relied mainly on the stirring of the air by the sparse vegetation, which resulted in the high levels of turbulence and faster redistribution of the pollutant to the upper layers of the atmosphere. The optimization method proved to be reliable, and can be used for other environmental problems.

- An alternative approach for dust dispersion based on the moment method was developed, and the deposition velocity model was adapted to fit in the moment method framework. The numerical experiments showed that the moment method may be more computationally efficient that the sectional approach based on the solution of passive scalar equations. Taking into account the introduced complexity of the method and the dry deposition model, the method is more suited for the problems where the performance is of utmost concern. The development of the moment method was not among the original objectives of the work, but it is very much relevant to the problems discussed here and thus belongs in this thesis.

Not every initial goal was however fulfilled to its fullest, mostly due to the time constraints. As noted above, performance comparison of the compressible and ABL solver would benefit from being extended by other test cases. Furthermore, as discussed in Sec. 2.3.4, the employed k-ϵ model does not include the buoyancy effects, possibly limiting its accuracy in stratified flows. Consequently, not much place was given to the investigation of the stratification effects in urban flows, as we have dealt mostly with neutrally stratified atmosphere.

7.2 Further applications of the developed methods

The developed methods were applied by the author and his colleagues to some problems related to the urban vegetation which were not discussed in this thesis. In the framework of the grant TD020357 of the Technology Agency of the Czech Republic we investigated the influence of the near-road barrier on the pollutant emitted from the highway in a cut (Šíp et al., 2015). The air flow and the pollutant dispersion were calculated for more than fifty geometrical variants of the vegetation block in order to help with the choice of the optimal barrier in real world situations. The developed solver was a crucial tool in this endeavor.

The moment method was applied to the investigation of the effects of vegetation inside the street canyons in (Šíp and Beneš, 2016a). We investigated the dispersion of the pollution from the road traffic located in the 3D street canyon under varying wind conditions and with or without vegetation present. Behaviour of particles of different sizes was under scrutiny in the work. In line with the previous experimental and numerical studies we have shown that the vegetation reduces circulation inside the street canyon,

and thus increases the pollutant levels at the pedestrian level. The dry deposition, often neglected in similar studies, was shown to play a role only for coarse mode particles. However, even for these, their reduction by the dry deposition was negligible compared to the increase caused by the reduced circulation.

7.3   Future directions

The work has opened many additional question and possible further research directions. We point out the following few, ranging from implementation issues to possible applications of the solver.

- **Distributed-memory system parallelization.** In the present implementation, the solver is parallelized for shared-memory systems using the OpenMP multithreading model. Such approach introduces a limit on the size of the solvable problems (or, more precisely, size of the employed computational meshes) due to the memory constraints, and currently leaves the large three dimensional problems with complex geometries out of reach. Distributed-memory parallelization using the MPI (Message Passing Interface) platform would remove this barrier.

- **Turbulence modelling.** Accuracy of the results is to a large extent dependent on the choice of the turbulent model. As discussed in above, we did not consider the stratification effects on the turbulence in the employed k-ϵ model. Also, the limit on the size of the turbulence eddies in the ABL was not taken into account, leaving the domain of applicability of the developed solver to restricted to small-scale problems.

  Even when small-scale problems are of concern, one might find a better model than our employed standard k-ϵ model. Koutsourakis et al. (2012) showed that the two-equation renormalization group (RNG) k-ϵ model performs better for street canyon flows. Further improvements may be obtained by using an anisotropic Reynolds-stress model, however, this options is more computationally demanding due to the higher number of equations and in some cases worse convergence properties (Crasto, 2007; Koutsourakis et al., 2012).

  Related problem is the choice of the turbulent Schmidt number in dispersion studies, as the proper choice is generally case dependent (Tominaga and Stathopoulos, 2007), and can significantly affect the results. This problem is reduced, although not fully eliminated, when large eddy simulation is employed. In LES the Schmidt number is needed only for the subgrid turbulence fluxes, and its choice therefore has smaller influence.

- **Uncertainty in the vegetation parameters.** The effects of the vegetation on the air flow and pollutant dispersion are highly dependent on a number of parameters that are difficult to estimate or measure: leaf area density profile, drag coefficient, or size of the leaves and needles. In computational studies, this uncertainty should be acknowledged and quantified using an appropriate method. Among such methods could be the Monte Carlo sampling, or polynomial chaos expansion.

- **Urban environment assessment.** Further improvements of the model would be needed in order to provide a tool for comprehensive urban environment assessment. For example, thermal effects of the tree canopy, such as shading effects or heat transfer, were not considered here. A possible approach was presented in (Mochida and Lun, 2008), where the authors discussed a CFD model including a moisture transport equation coupled with the radiative heat transfer model. Another option would be to incorporate a model of "green walls", i.e. walls covered by vegetation. It was indicated that green walls in street canyons can reduce street level concentration of PM10 by up to 60% (Pugh et al., 2012), even though this seems overly optimistic to the author in light of the results obtained in (Šíp and Beneš, 2016a).

Appendix A

Mathematical notation

Coordinates. Coordinate vector is denoted by $\boldsymbol{x}$. Its components in three dimensions are $\boldsymbol{x} = (x_1, x_2, x_3)$. In some contexts, an alternative notation $\boldsymbol{x} = (x, y, z)$ is established and is also used in this work. Similarly, vector components are written using the index notation, $\boldsymbol{a} = (a_1, a_2, a_3)$, or using the alternative notation, $\boldsymbol{a} = (a_x, a_y, a_z)$, depending on the context.

In the following we assume that all vectors are from the space $\mathbb{R}^3$ and all tensors are from the space $\mathbb{R}^{3 \times 3}$.

Vectors and matrices. Dot product of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$:

$$\boldsymbol{a} \cdot \boldsymbol{b} = a_1 b_1 + a_2 b_2 + a_3 b_3 \tag{A.1}$$

Tensor product of two vectors:

$$\boldsymbol{a} \otimes \boldsymbol{b} = \begin{pmatrix} a_1 b_1 & a_1 b_2 & a_1 b_3 \\ a_2 b_1 & a_2 b_2 & a_2 b_3 \\ a_3 b_1 & a_3 b_2 & a_3 b_3 \end{pmatrix} \tag{A.2}$$

Vector product of two vectors:

$$\boldsymbol{a} \times \boldsymbol{b} = (a_2 b_3 - a_3 b_2, a_3 b_1 - a_1 b_3, a_1 b_2 - a_2 b_1) \tag{A.3}$$

Differential operators. Gradient of a scalar function $f(\boldsymbol{x})$:

$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial f}{\partial x_3} \right)^T \tag{A.4}$$

Jacobian of a vector function $\boldsymbol{a}(\boldsymbol{x}) = (a_1(\boldsymbol{x}), a_2(\boldsymbol{x}), a_3(\boldsymbol{x}))^T$:

$$\nabla \boldsymbol{a} = \begin{pmatrix} \frac{\partial a_1}{\partial x_1} & \frac{\partial a_1}{\partial x_2} & \frac{\partial a_1}{\partial x_3} \\ \frac{\partial a_2}{\partial x_1} & \frac{\partial a_2}{\partial x_2} & \frac{\partial a_2}{\partial x_3} \\ \frac{\partial a_3}{\partial x_1} & \frac{\partial a_3}{\partial x_2} & \frac{\partial a_3}{\partial x_3} \end{pmatrix} \tag{A.5}$$

Divergence of a vector function $\boldsymbol{a}(\boldsymbol{x}) = (a_1(\boldsymbol{x}), a_2(\boldsymbol{x}), a_3(\boldsymbol{x}))^T$:

$$\text{div } \boldsymbol{a} = \frac{\partial a_1}{\partial x_1} + \frac{\partial a_2}{\partial x_2} + \frac{\partial a_3}{\partial x_3} \tag{A.6}$$

Divergence of a tensor $\boldsymbol{A}(\boldsymbol{x}) = (A_{ij}(\boldsymbol{x}))_{i,j=1}^3$:

$$\text{div } \boldsymbol{A} = \left(\sum_{j=1}^{3} \frac{\partial A_{j1}}{\partial x_j}, \sum_{j=1}^{3} \frac{\partial A_{j2}}{\partial x_j}, \sum_{j=1}^{3} \frac{\partial A_{j3}}{\partial x_j} \right)^T \tag{A.7}$$

Einstein notation. Occasionally the Einstein summation notation is used. Under this notation, if an index appears twice in a term, a summation over this index is implied. For example,

$$\frac{\partial u_j}{\partial x_j} = \sum_{j=1}^{3} \frac{\partial u_j}{\partial x_j}. \tag{A.8}$$

Appendix B

Vector form of the equations

Systems of equations implemented in the solver can be written in the form

$$\mathbf{\Gamma}\frac{\partial \boldsymbol{W}}{\partial t} + \sum_{j=1}^{3}\frac{\partial \boldsymbol{F}_j(\boldsymbol{W})}{\partial x_j} = \sum_{j=1}^{3}\frac{\partial \boldsymbol{R}_j(\boldsymbol{W}, \nabla \boldsymbol{W})}{\partial x_j} + \boldsymbol{Q}(\boldsymbol{W}),\qquad\text{(B.1)}$$

where $\mathbf{\Gamma}$ is the local preconditioning matrix, $\boldsymbol{W}$ is the state vector, $\boldsymbol{F}_j(\boldsymbol{W})$ are the inviscid fluxes, $\boldsymbol{R}_j(\boldsymbol{W}, \nabla \boldsymbol{W})$ are the viscous fluxes, and $\boldsymbol{Q}(\boldsymbol{W})$ are the sources and sinks. Its forms for all implemented systems are given below.

When temporal evolution of the system is of interest, and the preconditioning is used, dual time stepping in time τ (see Sec. 3.1) has to be employed:

$$\mathbf{\Gamma}\frac{\partial \boldsymbol{W}}{\partial \tau} + \boldsymbol{A}\frac{\partial \boldsymbol{W}}{\partial t} + \sum_{j=1}^{3}\frac{\partial \boldsymbol{F}_j(\boldsymbol{W})}{\partial x_j} = \sum_{j=1}^{3}\frac{\partial \boldsymbol{R}_j(\boldsymbol{W}, \nabla \boldsymbol{W})}{\partial x_j} + \boldsymbol{Q}(\boldsymbol{W}).\qquad\text{(B.2)}$$

For details on the equations and the notation used see the referred sections.

B.1 Compressible flow equations

The compressible flow equations were described in Sections 2.1.1 and 2.3.

$$\boldsymbol{W} = (\rho, \rho u_1, \rho u_2, \rho u_3, \rho E)^T \qquad\text{(B.3)}$$

$$\boldsymbol{F}_j = (\rho u_j, \rho u_j u_1 + \delta_{1j}p, \rho u_j u_2 + \delta_{2j}p, \rho u_j u_3 + \delta_{3j}p, \rho u_j(E + p/\rho))^T \qquad\text{(B.4)}$$

$$\boldsymbol{R}_j = (0, \tau_{1j}^E, \tau_{2j}^E, \tau_{3j}^E, \tau_{j1}^E u_1 + \tau_{j2}^E u_2 + \tau_{j3}^E u_3 - pu_j + (k_L + \mu_T c_p/Pr_T)\,(\partial T/\partial x_j))^T \qquad\text{(B.5)}$$

$$\boldsymbol{Q} = (0, \rho f_1, \rho f_2, \rho f_3, \rho q + \rho \boldsymbol{f}\cdot\boldsymbol{u})^T \qquad\text{(B.6)}$$

$$\mathbf{\Gamma} = \boldsymbol{I} \qquad \text{(Non-preconditioned equations)} \qquad\text{(B.7)}$$

Weiss-Smith preconditioner (Sec. 3.2)

$$\boldsymbol{\Gamma}^{-1} = \boldsymbol{I} - \frac{(1-\epsilon)(\gamma-1)}{c^2} \begin{pmatrix} \frac{U^2}{2} & -u_1 & -u_2 & -u_3 & 1 \\ u_1\frac{U^2}{2} & -u_1^2 & -u_1u_2 & -u_1u_3 & u_1 \\ u_2\frac{U^2}{2} & -u_1u_2 & -u_2^2 & -u_2u_3 & u_2 \\ u_3\frac{U^2}{2} & -u_1u_3 & -u_2u_3 & -u_3^2 & u_3 \\ h\frac{U^2}{2} & -u_1h & -u_2h & -u_3h & h \end{pmatrix} \tag{B.8}$$

$$\boldsymbol{A} = \boldsymbol{I} \tag{B.9}$$

B.2 ABL flow equations

The ABL flow equations were described in Sections 2.1.3 and 2.3, and the artificial compressibility approach was elaborated on in Sec. 3.1.

$$\boldsymbol{W} = (p^*, u_1, u_2, u_3, \theta)^T \tag{B.10}$$

$$\boldsymbol{F}_j = (u_j, u_ju_1 + \delta_{1j}p^*/\rho_{\text{ref}}, u_ju_2 + \delta_{2j}p^*/\rho_{\text{ref}}, u_ju_3 + \delta_{3j}p^*/\rho_{\text{ref}}, u_j\theta)^T \tag{B.11}$$

$$\boldsymbol{R}_j = (0, \nu_E(\partial u_1/\partial x_j), \nu_E(\partial u_2/\partial x_j), \nu_E(\partial u_3/\partial x_j), (k_L/\rho_{\text{ref}}c_p + \nu_T/Pr_T)(\partial\theta/\partial x_j))^T \tag{B.12}$$

$$\boldsymbol{Q} = (0, f_1^B, f_2^B, f_3^B, q/c_p)^T \tag{B.13}$$

$$\boldsymbol{A} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{B.14}$$

Classical artificial compressibility

$$\boldsymbol{\Gamma} = \begin{pmatrix} 1/\beta & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{B.15}$$

Generalized artificial compressibility

$$\mathbf{\Gamma} = \begin{pmatrix} 1/\beta & 0 & 0 & 0 & 0 \\ u_1/\beta & 1 & 0 & 0 & 0 \\ u_2/\beta & 0 & 1 & 0 & 0 \\ u_3/\beta & 0 & 0 & 1 & 0 \\ \theta/\beta & 0 & 0 & 0 & 1 \end{pmatrix} \tag{B.16}$$

B.3 Passive scalar equation

The passive scalar equation was described in Sections 2.2 and 2.3.3.

$$\boldsymbol{W} = (c)^T \tag{B.17}$$
$$\boldsymbol{F}_j = ((u_j - u_s \delta_{j3})c)^T \tag{B.18}$$
$$\boldsymbol{R}_j = (\nu_T/Sc_T \ \partial c/\partial x_j)^T \tag{B.19}$$
$$\boldsymbol{Q} = (S_c)^T \tag{B.20}$$
$$\mathbf{\Gamma} = \boldsymbol{I} \tag{B.21}$$

B.4 Standard k-ϵ turbulence model

The standard k-ϵ turbulence model was described in Sec. 2.3.4.

$$\boldsymbol{W} = (\rho k, \rho \epsilon)^T \tag{B.22}$$
$$\boldsymbol{F}_j = (\rho u_j k, \rho u_j \epsilon)^T \tag{B.23}$$
$$\boldsymbol{R}_j = ((\mu_L + \mu_T/\sigma_k)(\partial k/\partial x_j), (\mu_L + \mu_T/\sigma_\epsilon)(\partial \epsilon/\partial x_j))^T \tag{B.24}$$
$$\boldsymbol{Q} = (P_k - \rho \epsilon, C_{\epsilon_1}(\epsilon/k)P_k - C_{\epsilon_2}\rho \epsilon^2/k)^T \tag{B.25}$$
$$\mathbf{\Gamma} = \boldsymbol{I} \tag{B.26}$$

Appendix C

Dry deposition model for the moment method

When adapting the dry deposition model for the moment method, we are looking the expression of the deposition velocity u'_d approximating the original model, $u'_d \approx u_d$, while allowing the analytical evaluation of the integral in the sink term of the moment equations

$$S_{dep,k} = -\text{LAD} \int_0^\infty d_p^k u'_d(d_p) n_N(d_p) \mathrm{d}d_p. \tag{C.1}$$

The model proposed in (Petroff et al., 2008b) and (Petroff et al., 2009) expresses the deposition velocity as a sum of the deposition velocities associated with the underlying processes: Brownian diffusion, interception, inertial impaction, turbulent impaction, and sedimentation,

$$u_d = 2(u_{BD} + u_{IN} + u_{IM} + u_{TI} + u_{SE}) = \sum_{proc} 2u_{proc}, \tag{C.2}$$

where we sum over the above mentioned physical processes.

In the following sections we will look at the expressions for the deposition velocities u_{proc} to consider if the integral associated with each process

$$\int_0^\infty d_p^k u_{proc}(d_p) n_N(d_p) \mathrm{d}d_p \tag{C.3}$$

can be evaluated analytically, and if not, we will propose an approximate form of the deposition velocity $u'_{proc} \approx u_{proc}$. With this approximation, we will then evaluate the integral to obtain the formulation of the sink term to be used in the moment equations,

$$S_{dep,k} = \sum_{proc} S_{proc,k} = \sum_{proc} -2\,\text{LAD} \int_0^\infty d_p^k u'_{proc}(d_p) n_N(d_p) \mathrm{d}d_p. \tag{C.4}$$

Brownian diffusion

Brownian diffusion is the dominant process driving the deposition of the particles smaller then 0.1 μm. Original model formulates the contribution to the deposition velocity due

to the Brownian diffusion as

$$u_{BD} = UC_B Sc^{-2/3} Re^{n_B - 1} \tag{C.5}$$

where U is the magnitude of the wind velocity, $Sc = \nu_a / D_B$ is the Schmidt number (with ν_a being the kinematic viscosity of air and D_B the Brownian diffusion coefficient, $D_B = (C_C k_b T_a)/(3\pi \mu_a d_p)$), $Re = U d_e / \nu_a$ is the Reynolds number, and d_e is the needle diameter.

Cunningham correction factor $C_C^A = 1 + 2\lambda/d_p(1.257 + 0.4\exp(-1.1d_p/2\lambda))$ is used in the original model (Petroff et al., 2008b), where λ is the mean free path of the particle in the air. In this chapter we use simpler approximation $C_C^B = 1 + 3.34\lambda/d_p$ (Bae et al., 2009). Comparison of these expressions is on Fig. C.1, where it can be seen that their difference peaks to 12% for particle diameter around 0.2 µm.
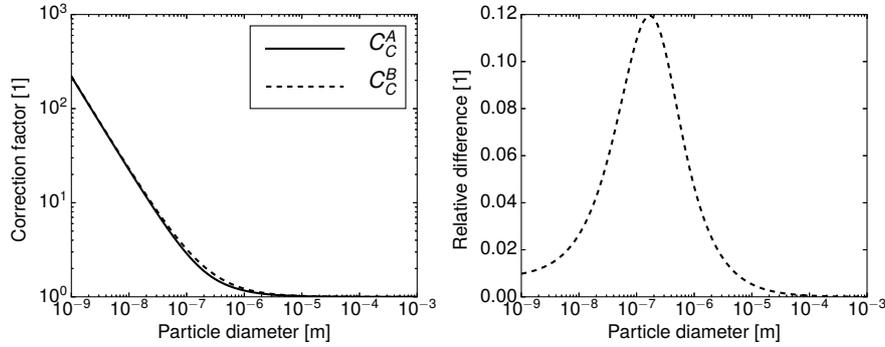
Figure C.1: (Left) Two expressions for the Cunningham correction factor. (Right) Relative difference $(C_C^B - C_C^A)/C_C^A$.

Furthermore, for the Brownian diffusion we take into account only the size-dependent part of the correction factor, dominant in the particle size range where the diffusion is significant, $C_C \approx 3.34\frac{\lambda}{d_p}$. Putting the expressions above into Eq. (C.5), we obtain

$$u_{BD} \approx u'_{BD} = U^{n_B}\gamma_1\gamma_2^{2/3}d_p^{-4/3}, \tag{C.6}$$

where $\gamma_1 = C_B\left(d_e/\nu_a\right)^{n_B-1}\left(3\pi\nu_a^2\rho_a/(k_b T_a)\right)^{-2/3}$ and $\gamma_2 = 3.34\lambda$.

Using this formula, the contribution to the moment equation can be written as

$$S_{BD,k} = -2\,\mathrm{LAD}\,U^{n_B}\gamma_1\gamma_2^{2/3}M_{k-4/3}. \tag{C.7}$$

Interception

Interception denotes the process where the particle follows the streamline, but too close to the obstacle so that it is captured on the surface. The original model parameterizes the contribution of the interception process to the deposition velocity differently for needles and broadleaves. For needles, it reads

$$u_{IN} = 2U k_x \frac{d_p}{d_e}, \tag{C.8}$$

107

and for broadleaves

$$u_{IN} = \frac{1}{2}Uk_x\frac{d_p}{d_e}\left(2 + \ln\left(\frac{4d_e}{d_p}\right)\right) = \frac{1}{2}Uk_x\frac{d_p}{d_e}\left(2 + \ln(4d_e) - \ln(d_p)\right),\qquad\text{(C.9)}$$

where k_x is the ratio of the leaf surface projected on the plane perpendicular to the flow to the total leaf surface.

The expression for needles can be integrated easily, since there is a linear dependence on the particle diameter. That is not the case for the broadleaves due to the logarithm in the expression. We approximated the last term in the bracket by the power law function,

$$-\ln(d_p) = a_{in}\, d_p^{b_{in}},\qquad\text{(C.10)}$$

where the coefficients $a_{in} = 4.57$ and $b_{in} = -0.078$ were obtained by the numerical minimization of the quadratic error of the original and approximated formulation on the diameter interval $d_p \in [10^{-8}, 10^{-3}]$m. Comparison of the original and approximated term is shown on Fig. C.2.
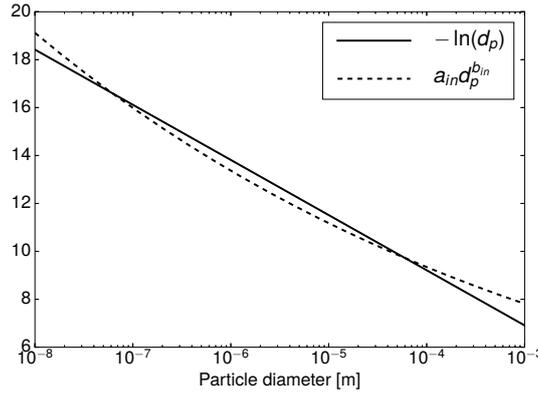
Figure C.2: Original and approximated expressions for $-\ln(d_p)$ in the interception term.

Formally, the approximated deposition velocity may be written as

$$u'_{IN} = \begin{cases} 2Uk_x\frac{d_p}{d_e} & \text{for needles,} \\ \frac{1}{2}Uk_x\frac{d_p}{d_e}\left(2 + \ln(4d_e) + a_{in}\, d_p^{b_{in}}\right) & \text{for broadleaves.} \end{cases}\qquad\text{(C.11)}$$

These expressions now can be integrated, and we obtain

$$S_{IN,k} = \begin{cases} -2\,\text{LAD}\gamma_3 U M_{k+1} & \text{for needles,} \\ -2\,\text{LAD}\left(\frac{\gamma_3}{4}U(\gamma_7 M_{k+1} + a_{in}M_{k+1+b_{in}})\right) & \text{for broadleaves,} \end{cases}\qquad\text{(C.12)}$$

where $\gamma_3 = 2k_x/d_e$ and $\gamma_7 = \ln(4d_e)$.

Inertial impaction

Inertial impaction occurs when particles do not follow the streamlines due to their inertia, resulting in the collision with the obstacle. The deposition velocity due to the inertial impaction is written as

$$u_{IM} = Uk_x E_{IM},$$ (C.13)

where $E_{IM} = \left(\frac{St}{St+\beta}\right)^2$ is the impaction efficiency, $St = \tau_p U/d_e$ is the Stokes number, $\tau_p = (\rho_p C_c d_p^2)(18\mu)$ is the particle relaxation time, and β is the model constant (0.6 for needles and 0.47 for broadleaves). To use this deposition velocity in the moment equations, we replace the impaction efficiency (considered as a function of the Stokes number) by its piecewise linear approximation,

$$E'_{IM} = \begin{cases} 0 & \text{if } St < s_0, \\ a_i St + b_i & \text{if } s_i \le St < s_{i+1}, \quad i = 0, \ldots, m-1, \\ 1 & \text{if } St \ge s_m, \end{cases}$$ (C.14)

where $\{s_i\}_{i=0}^m$ are selected interpolation points, a_i and b_i are coefficients calculated so that

$$E'_{IM}(s_i) = E_{IM}(s_i) \qquad \text{for} \quad i = 0, \ldots, m.$$ (C.15)

Here we have set $\{s_i\} = \{10^{-2}, 10^{-1}, 1, 10, 10^2\}$. Low number of interpolation points was chosen deliberately to limit the computational needs. The original function and the approximation are shown on Fig. C.3.
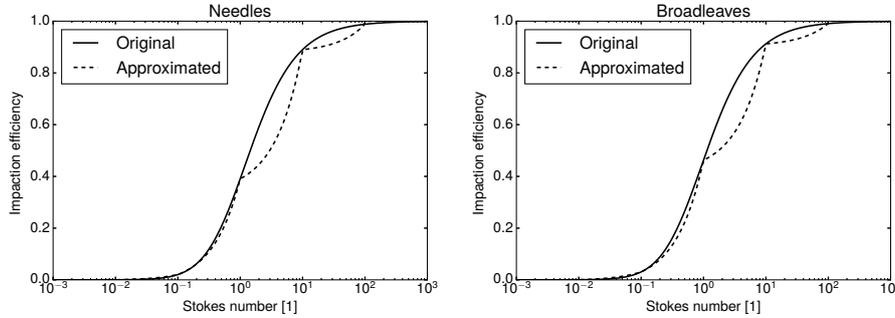
Figure C.3: Original and approximated expressions for the impaction efficiency for needles ($\beta = 0.6$, left) and broadleaves ($\beta = 0.47$, right).

This approximation introduces an underprediction of the impaction efficiency especially for Stokes numbers between 1 and 10. This was however deemed to be an acceptable compromise between the accuracy and the computational efficiency. A higher number of interpolation points would reduce this error.

Approximated deposition velocity can then be formally written as

$$u'_{IM} = Uk_x E'_{IM}.$$ (C.16)

The corresponding term in the moment equation is obtained by calculating the integrals in

$$S_{IM,k} = -2\,\mathrm{LAD} \int_0^\infty d_p^k U k_x E'_{IM}(d_p) n_N(d_p) \mathrm{d}d_p$$

$$= -2\,\mathrm{LAD}\, U k_x \left(\sum_{i=0}^{m-1} \int_{s_i}^{s_{i+1}} d_p^k (a_i St + b_i) n_N(d_p) \mathrm{d}d_p + \int_{s_m}^\infty d_p^k n_N(d_p) \mathrm{d}d_p \right). \quad (C.17)$$

After some algebraic manipulations, the term may be written using the incomplete moments $M_k^-(x) = \int_0^x d_p^k n_N(d_p) \mathrm{d}d_p$ and $M_k^+(x) = \int_x^\infty d_p^k n_N(d_p) \mathrm{d}d_p$ as

$$S_{IM,k} = -2\,\mathrm{LAD}\, U k_x \left[\gamma_8 \sum_{i=0}^{m-1} a_i (M_{k+2}^-(s_{i+1}) - M_{k+2}^-(s_i)) \right. \tag{C.18}$$

$$\left. + \sum_{i=0}^{m-1} b_i (M_k^-(s_{i+1}) - M_k^-(s_i)) + M_k^+(s_m) \right], \tag{C.19}$$

where $\gamma_4 = \rho_p U / (18 \mu d_e)$.

Turbulent impaction

Effect of the particle impaction due to the canopy turbulence is described by the deposition velocity

$$u_{TI} = \begin{cases} u_f K_{TI1} \tau_p^{+2} & \text{if } \tau_p^+ < 20, \\ u_f K_{TI2} & \text{if } \tau_p^+ \geq 20. \end{cases} \tag{C.20}$$

Here $\tau_p^+ = \tau_p u_f^2 / \nu_a$ is the dimensionless particle relaxation time, $\tau_p = \frac{\rho_p C_c d_p^2}{18 \mu_a}$ is the particle relaxation time, u_f is the local friction velocity, and K_{TI1} and K_{TI2} are the constants of the model. The contribution to the moment equation can be again expressed using the incomplete moments,

$$S_{TI,k} = -2\,\mathrm{LAD} \left(u_f^5 \gamma_5 M_{k+4}^-(d_p^{T2}) + u_f K_{TI2} M_k^+(d_p^{T2}) \right) \tag{C.21}$$

with the threshold $d_p^{T2} = \sqrt{\frac{360 \mu_a \nu_a}{\rho_p u_f^2}}$ and $\gamma_5 = \frac{K_{TI1} \rho_p^2 \rho_a^2}{(18 \mu_a^2)^2}$.

Sedimentation

Sedimentation plays the major role for the particles with the diameter above $10\,\mu\mathrm{m}$. The sedimentation contribution to the deposition velocity is expressed as

$$u_{SE} = \frac{k_z g \rho_p C_C d_p^2}{18 \mu_a}, \tag{C.22}$$

where k_z is the ratio of the leaf surface projected to the horizontal plane to the total leaf surface.

Substituting this expression to the integral in the moment equation, after some algebraic manipulations we obtain

$$S_{SE,k} = -2 \, \text{LAD} \gamma_6 (M_{k+2} + \gamma_2 M_{k+1}) \tag{C.23}$$

with $\gamma_6 = \frac{k_z g \rho_p}{18 \mu_a}$ and $\gamma_2 = 3.34\lambda$ as before.

Bibliography

Abdi, D. S. & Bitsuamlak, G. T. (2014). Wind flow simulations on idealized and real complex terrain using various turbulence models. *Adv. Eng. Software*, *75*, 30–41. doi:10.1016/j.advengsoft.2014.05.002

Apsley, D. D. & Castro, I. P. (1997a). A limited-length-scale k-ϵ model for the neutral and stably-stratified atmospheric boundary layer. *Boundary Layer Meteorol. 83*(1), 75–98. doi:10.1023/a:1000252210512

Apsley, D. D. & Castro, I. P. (1997b). Flow and dispersion over hills: comparison between numerical predictions and experimental data. *J. Wind Eng. Ind. Aerodyn. 67-68*, 375–386. doi:10.1016/s0167-6105(97)00087-1

Apsley, D. (1995). *Numerical modelling of neutral and stably stratified flow and dispersion in complex terrain* (Doctoral dissertation, University of Surrey).

Arya, S. P. (1999). *Air pollution meteorology and dispersion*. Oxford University Press New York.

Atkinson, B. (1995). Introduction to the fluid mechanics of meso-scale flow fields. In A. Gyr & F.-S. Rys (Eds.), *Diffusion and transport of pollutants in atmospheric mesoscale flow fields* (pp. 1–22). Springer.

Bae, S., Jung, C. & Kim, Y. (2009). Development of an aerosol dynamics model for dry deposition process using the moment method. *Aerosol Sci. Technol. 43*(6), 570–580. doi:10.1080/02786820902798476

Baines, P. G. (1995). *Topographic effects in stratified flows*. Cambridge University Press.

Balay, S., Abhyankar, S., Adams, M., Brown, J., Brune, P., Buschelman, K., ... Zhang, H. (2015). PETSc Web page. http://www.mcs.anl.gov/petsc.

Balogh, M., Parente, A. & Benocci, C. (2012). RANS simulation of ABL flow over complex terrains applying an enhanced k-ϵ model and wall function formulation: implementation and comparison for fluent and OpenFOAM. *J. Wind Eng. Ind. Aerodyn. 104-106*, 360–368. doi:10.1016/j.jweia.2012.02.023

Bäumer, D., Vogel, B. & Fiedler, F. (2005). A new parameterisation of motorway-induced turbulence and its application in a numerical model. *Atmos. Environ. 39*(31), 5750–5759. doi:http://dx.doi.org/10.1016/j.atmosenv.2004.10.046

Beckett, K. P., Freer-Smith, P. & Taylor, G. (2000). Particulate pollution capture by urban trees: effect of species and windspeed. *Global Change Biol. 6*(8), 995–1003. doi:10.1046/j.1365-2486.2000.00376.x

Beneš, L., Fürst, J. & Frauniė, P. (2011). Comparison of two numerical methods for the stratified flow. *Computers & Fluids*, *46*(1), 148–154. doi:10.1016/j.compfluid.2011.02.003

Binkowski, F. & Shankar, U. (1995). The regional particulate matter model: 1. model description and preliminary results. *J. Geophys. Res. 100*(D12), 26191–26209. doi:10.1029/95JD02093

Blackadar, A. (1962). The vertical distribution of wind and turbulent exchange in a neutral atmosphere. *J. Geophys. Res. 67*(8), 3095–3102. doi:10.1029/JZ067i008p03095

Blazek, J. (2001). *Computational fluid dynamics: principles and applications*. Elsevier.

Blocken, B., Janssen, W. & van Hooff, T. (2012). CFD simulation for pedestrian wind comfort and wind safety in urban areas: general decision framework and case study for the eindhoven university campus. *Environ. Modell. Software, 30*, 15–34. doi:10.1016/j.envsoft.2011.11.009

Blocken, B. (2015). Computational fluid dynamics for urban physics: importance, scales, possibilities, limitations and ten tips and tricks towards accurate and reliable simulations. *Build. Environ. 91*, 219–245. doi:10.1016/j.buildenv.2015.02.015

Blocken, B., Stathopoulos, T., Carmeliet, J. & Hensen, J. L. (2011). Application of computational fluid dynamics in building performance simulation for the outdoor environment: an overview. *J. Build. Perform. Simul. 4*(2), 157–184. doi:10.1080/19401493.2010.513740

Bodnár, T. (2003). *Numerical simulation of flows and pollution dispersion in atmospheric boundary layer* (Doctoral dissertation, Czech Technical University in Prague and Universite de Toulon et du Var).

Bodnár, T., Beneš, L., Fraunié, P. & Kozel, K. (2012). Application of compact finite-difference schemes to simulations of stably stratified fluid flows. *Appl. Math. Comput. 219*(7), 3336–3353. doi:10.1016/j.amc.2011.08.058

Bruse, M. (2007). Particle filtering capacity of urban vegetation: a microscale numerical approach. *Berliner Geographische Arbeiten, 109*, 61–70.

Buccolieri, R., Salim, S. M., Leo, L. S., Sabatino, S. D., Chan, A., Ielpo, P., … Gromke, C. (2011). Analysis of local scale tree–atmosphere interaction on pollutant concentration in idealized street canyons and application to a real urban junction. *Atmos. Environ. 45*(9), 1702–1713. doi:10.1016/j.atmosenv.2010.12.058

Castro, I. P. & Apsley, D. D. (1997). Flow and dispersion over topography: a comparison between numerical and laboratory data for two-dimensional flows. *Atmos. Environ. 31*(6), 839–850. doi:10.1016/s1352-2310(96)00248-8

Chan, T. & van der Vorst, H. (2001). Approximate and incomplete factorizations. *Parallel Numerical Algorithms, ICASE/LaRC Interdisciplinary Series in Science and Engeneering*, 167–202.

Choi, Y.-H. & Merkle, C. L. (1993). The application of preconditioning in viscous flows. *J. Comput. Phys. 105*(2), 207–223. doi:doi:10.1006/jcph.1993.1069

Chorin, A. J. (1967). A numerical method for solving incompressible viscous flow problems. *J. Comput. Phys. 2*(1), 12–26. doi:10.1006/jcph.1997.5716

Colin, Y., Deniau, H. & Boussuge, J.-F. (2011). A robust low speed preconditioning formulation for viscous flow computations. *Computers & Fluids, 47*(1), 1–15. doi:10.1016/j.compfluid.2011.01.015

Coudière, Y., Vila, J.-P. & Villedieu, P. (1999). Convergence rate of a finite volume scheme for a two dimensional convection-diffusion problem. *ESAIM: Mathematical Modelling and Numerical Analysis, 33*(03), 493–516. doi:10.1051/m2an:1999149

Crasto, G. (2007). *Numerical simulations of the atmospheric boundary layer* (Doctoral dissertation, Università degli Studi di Cagliari).

Defraeye, T., Derome, D., Verboven, P., Carmeliet, J. & Nicolai, B. (2014). Cross-scale modelling of transpiration from stomata via the leaf boundary layer. *Ann. Bot. 114*(4), 711–723. doi:10.1093/aob/mct313

dos Santos, A. A. C., Lima, L. d. S. M., de Paula, G. A. R., Moreira, G. A. A. & Valle, R. M. (2009). RANS turbulence model evaluation for neutral atmospheric boundary layer simulation over complex terrain. In *Proceedings of COBEM 2009*.

Duarte, M., Almgren, A. S., Balakrishnan, K., Bell, J. B. & Romps, D. M. (2014). A numerical study of methods for moist atmospheric flows: compressible equations. *Mon. Wea. Rev. 142*(11), 4269–4283. doi:10.1175/mwr-d-13-00368.1

Dupont, S., Bonnefond, J.-M., Irvine, M. R., Lamaud, E. & Brunet, Y. (2011). Long-distance edge effects in a pine forest with a deep and sparse trunk space: in situ and numerical experiments. *Agric. For. Meteorol. 151*(3), 328–344. doi:10.1016/j.agrformet.2010.11.007

Dupont, S. & Brunet, Y. (2008a). Edge flow and canopy structure: a large-eddy simulation study. *Boundary-Layer Meteorol. 126*(1), 51–71. doi:10.1007/s10546-007-9216-3

Dupont, S. & Brunet, Y. (2008b). Influence of foliar density profile on canopy flow: a large-eddy simulation study. *Agricultural and Forest Meteorology*, *148*(6-7), 976–990. doi:10.1016/j.agrformet.2008.01.014

Eckert, S., Baaser, H., Gross, D. & Scherf, O. (2004). A BDF2 integration method with step size control for elasto-plasticity. *Comput. Mech. 34*(5), 377–386. doi:10.1007/s00466-004-0581-1

Eidsvik, K. & Utnes, T. (1997). Flow separation and hydraulic transitions over hills modelled by the Reynolds equations. *J. Wind Eng. Ind. Aerodyn. 67-68*, 403–413. doi:10.1016/s0167-6105(97)00089-5

Endalew, A. M., Hertog, M., Delele, M., Baetens, K., Persoons, T., Baelmans, M., . . . Verboven, P. (2009). CFD modelling and wind tunnel validation of airflow through plant canopies using 3D canopy architecture. *Int. J. Heat Fluid Flow*, *30*(2), 356–368. doi:10.1016/j.ijheatfluidflow.2008.12.007

ERCOFTAC. (2004). ERCOFTAC QNET-CFD Wiki, application challenge 5-05. http://qnet-ercoftac.cfms.org.uk/w/index.php/AC_5-05. [Online; accessed 22-August-2016].

Estoque, M. A. & Bhumralkar, C. M. (1969). Flow over a localized heat source. *Mon. Weather Rev. 97*(12), 850–859. doi:10.1175/1520-0493(1969)097<0850:FOALHS>2.3.CO;2

Feistauer, M., Felcman, J. & Straškraba, I. (2003). *Mathematical and computational methods for compressible flow*. Oxford University Press.

Fuka, V. & Brechler, J. (2012). Large eddy simulation modelling of the dispersion of radioactive particulate matter. *International Journal of Environment and Pollution*, *48*(1/2/3/4), 156. doi:10.1504/ijep.2012.049662

Garratt, J. (1992). *The atmospheric boundary layer*. Cambridge atmospheric and space science series. Cambridge University Press.

Gebremedhin, A. H., Manne, F. & Pothen, A. (2005). What color is your Jacobian? Graph coloring for computing derivatives. *SIAM review*, *47*(4), 629–705. doi:10.1137/S0036144504444711

Giraldo, F. X. & Restelli, M. (2008). A study of spectral element and discontinuous Galerkin methods for the Navier–Stokes equations in nonhydrostatic mesoscale atmospheric modeling: Equation sets and test cases. *J. Comput. Phys. 227*(8), 3849–3877. doi:10.1016/j.jcp.2007.12.009

Gorlé, C., van Beeck, J., Rambaud, P. & Tendeloo, G. V. (2009). CFD modelling of small particle dispersion: the influence of the turbulence kinetic energy in the atmospheric boundary layer. *Atmos. Environ. 43*(3), 673–681. doi:10.1016/j.atmosenv.2008.09.060

Green, S. (1992). Modelling turbulent air flow in a stand of widely-spaced trees. *Phoenics J. 5*(3), 294–312.

Greenshields, C. J. (2015). *OpenFOAM - The Open Source CFD Toolbox - User's Guide. version 3.0.0*. CFD Direct Ltd.

Gromke, C. & Blocken, B. (2015). Influence of avenue-trees on air quality at the urban neighborhood scale. Part I: quality assurance studies and turbulent schmidt number analysis for RANS CFD simulations. *Environ. Pollut. 196*, 214–223. doi:10.1016/j.envpol.2014.10.016

Gromke, C. & Ruck, B. (2012). Pollutant concentrations in street canyons of different aspect ratio with avenues of trees for various wind directions. *Boundary-Layer Meteorol. 144*(1), 41–64. doi:10.1007/s10546-012-9703-z

Gunzburger, M. (2003). *Perspectives in flow control and optimization*. Society for Industrial and Applied Mathematics. doi:10.1137/1.9780898718720. eprint: http://epubs.siam.org/doi/pdf/10.1137/1.9780898718720

Guo, L. & Maghirang, R. G. (2012). Numerical simulation of airflow and particle collection by vegetative barriers. *Engineering Applications of Computational Fluid Mechanics*, *6*(1), 110–122. doi:10.1080/19942060.2012.11015407

Hargreaves, D. & Wright, N. (2007). On the use of the k– model in commercial CFD software to model the neutral atmospheric boundary layer. *J. Wind Eng. Ind. Aerodyn. 95*(5), 355–369. doi:10.1016/j.jweia.2006.08.002

Hinds, W. (1999). *Aerosol technology: properties, behavior, and measurement of airborne particles* (2nd). Wiley.

Holton, J. (2004). *An introduction to dynamic meteorology* (4.). Elsevier Academic Press.

Jacobson, M. Z. (2005). *Fundamentals of atmospheric modeling*. Cambridge university press.

Jafari, S. (2014). *Efficient simulation of wind and wake flows in wind farms using a preconditioned multistage solver* (Doctoral dissertation, ETH Zürich).

Janhäll, S. (2015). Review on urban vegetation and particle air pollution - deposition and dispersion. *Atmos. Environ. 105*, 130–137. doi:10.1016/j.atmosenv.2015.01.052

Johnson, S. (2015). The NLopt nonlinear-optimization package. http://ab-initio.mit.edu/wiki/index.php/NLopt.

Jung, C., Kim, Y. & Lee, K. (2003). A moment model for simulating raindrop scavenging of aerosols. *J. Aerosol Sci. 34*(9), 1217–1233. doi:10.1016/S0021-8502(03)00098-3

Karel, J. (2014). *Numerical simulation of streamer propagation on unstructured dynamically adapted grids* (Doctoral dissertation, Czech Technical University in Prague and Université Paris 13).

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

Kenjereš, S. & ter Kuile, B. (2013). Modelling and simulations of turbulent flows in urban areas with vegetation. *J. Wind Eng. Ind. Aerodyn. 123*, 43–55. doi:10.1016/j.jweia.2013.09.007

Khurshudyan, L. H., Snyder, W. H. & Nekrasov, I. V. (1981). *Flow and dispersion of pollutants over two-dimensional hills: summary report on joint soviet-american study* (tech. rep. No. EPA-600/4-81-067). U.S. Environmental Protection Agency. US Environmental Protection Agency, Environmental Sciences Research Laboratory.

Knoll, D. & Keyes, D. (2004). Jacobian-free Newton–Krylov methods: a survey of approaches and applications. *J. Comput. Phys. 193*(2), 357–397. doi:10.1016/j.jcp.2003.08.010

Koutsourakis, N., Bartzis, J. G. & Markatos, N. C. (2012). Evaluation of Reynolds stress, k-ϵ and RNG k-ϵ turbulence models in street canyon flows using various experimental datasets. *Environ. Fluid Mech. 12*(4), 379–403. doi:10.1007/s10652-012-9240-9

Koziol, A. & Leighton, H. (2007). The moments method for multi-modal multi-component aerosols as applied to the coagulation-type equation. *Q. J. R. Meteorol. Soc. 133*(625), 1057–1070. doi:10.1002/qj.51

Lalic, B. & Mihailovic, D. (2004). An empirical relation describing leaf-area density inside the forest for environmental modeling. *J. Appl. Meteor. 43*, 641–645. doi:10.1175/1520-0450(2004)043<0641:AERDLD>2.0.CO;2

Launder, B. E. & Spalding, D. (1974). The numerical computation of turbulent flows. *Comput.Methods in Appl.Mech.Eng. 3*(2), 269–289. doi:10.1016/0045-7825(74)90029-2

LeVeque, R. J. (2004). *Finite volume methods for hyperbolic problems*. Cambridge university press.

Liou, M.-S. (1996). A sequel to AUSM: AUSM+. *J. Comput. Phys. 129*(2), 364–382. doi:10.1006/jcph.1996.0256

Liou, M.-S. & Steffen, C. J. (1993). A new flux splitting scheme. *J. Comput. Phys. 107*(1), 23–39. doi:10.1006/jcph.1993.1122

Liou, M.-S. (2006). A sequel to AUSM, part II: AUSM+-up for all speeds. *J. Comput. Phys. 214*, 137–170. doi:10.1016/j.jcp.2005.09.020

Litschke, T. & Kuttler, W. (2008). On the reduction of urban particle concentration by vegetation - a review. *Meteorol. Z. 17*, 229–240. doi:10.1127/0941-2948/2008/0284

Liu, J., Chen, J., Black, T. & Novak, M. (1996). E-ε modelling of turbulent air flow downwind of a model forest edge. *Boundary Layer Meteorol. 77*(1), 21–44. doi:10.1007/BF00121857

Mehta, D., van Zuijlen, A., Koren, B., Holierhoek, J. & Bijl, H. (2014). Large eddy simulation of wind farm aerodynamics: a review. *J. Wind Eng. Ind. Aerodyn. 133*, 1–17. doi:10.1016/j.jweia.2014.07.002

Mochida, A. & Lun, I. Y. (2008). Prediction of wind environment and thermal comfort at pedestrian level in urban area. *J. Wind Eng. Ind. Aerodyn. 96*(10–11), 1498–1527. 4th

International Symposium on Computational Wind Engineering (CWE2006). doi:10 .1016/j.jweia.2008.02.033

Mueller, E., Mell, W. & Simeoni, A. (2014). Large eddy simulation of forest canopy flow for wildland fire modeling. *Can. J. For. Res. 44*(12), 1534–1544. doi:10.1139/cjfr-2014 -0184

Muldoon, F. & Acharya, S. (2007). A modification of the artificial compressibility algorithm with improved convergence characteristics. *Int. J. Numer. Meth. Fl. 55*(4), 307–345. doi:10.1002/fld.1435

Parente, A., Gorlé, C., van Beeck, J. & Benocci, C. (2011). Improved k–ϵ model and wall function formulation for the RANS simulation of ABL flows. *J. Wind Eng. Ind. Aerodyn. 99*(4), 267–278. doi:10.1016/j.jweia.2010.12.017

Patankar, S. V. & Spalding, D. B. (1972). A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *Int. J. Heat Mass Transfer*, *15*(10), 1787–1806.

Petroff, A., Mailliat, A., Amielh, M. & Anselmet, F. (2008a). Aerosol dry deposition on vegetative canopies. Part I: Review of present knowledge. *Atmos. Environ. 42*, 3625–3653. doi:10.1016/j.atmosenv.2007.09.043

Petroff, A., Mailliat, A., Amielh, M. & Anselmet, F. (2008b). Aerosol dry deposition on vegetative canopies. Part II: a new modelling approach and applications. *Atmos. Environ. 42*(16), 3654–3683. doi:10.1016/j.atmosenv.2007.12.060

Petroff, A. & Zhang, L. (2010). Development and validation of a size-resolved particle dry deposition scheme for application in aerosol transport models. *Geoci. Model Dev. 3*, 753–769. doi:10.5194/gmd-3-753-2010

Petroff, A., Zhang, L., Pryor, S. & Belot, Y. (2009). An extended dry deposition model for aerosols onto broadleaf canopies. *J. Aerosol Sci. 40*(3), 218–240. doi:10.1016/j. jaerosci.2008.11.006

Pirjola, L., Kulmala, M., Wilck, M., Bischoff, A., Stratmann, F. & Otto, E. (1999). Formation of sulphuric acid aerosols and cloud condensation nuclei: an expression for significant nucleation and model comparison. *J. Aerosol Sci. 30*(8), 1079–1094. doi:10.1016 /S0021-8502(98)00776-9

Pugh, T. A. M., MacKenzie, A. R., Whyatt, J. D. & Hewitt, C. N. (2012). Effectiveness of green infrastructure for improvement of air quality in urban street canyons. *Environ. Sci. Technol. 46*(14), 7692–7699. doi:10.1021/es300826w

Raupach, M., Briggs, P., Ford, P., Leys, J., Muschal, M., Cooper, B. & Edge, V. (2001). Endosulfan transport ii. modeling airborne dispersal and deposition by spray and vapor. *J. Environ. Qual. 30*(3), 729–740. doi:10.2134/jeq2001.303714x

Richards, P. & Hoxey, R. (1993). Appropriate boundary conditions for computational wind engineering models using the k-ϵ turbulence model. *J. Wind Eng. Ind. Aerodyn. 46 & 47*, 145–153. doi:10.1016/0167-6105(93)90124-7

Rizwan, A. M., Dennis, L. Y. & Liu, C. (2008). A review on the generation, determination and mitigation of urban heat island. *J. Environ. Sci. 20*(1), 120–128. doi:10.1016/s10 01-0742(08)60019-4

Robert, A. (1993). Bubble convection experiments with a semi-implicit formulation of the euler equations. *J. Atmos. Sci. 50*(13), 1865–1873. doi:10.1175/1520-0469(1993)050<1 865:BCEWAS>2.0.CO;2

Saad, Y. & Schultz, M. H. (1986). GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, *7*(3), 856–869. doi:10.1137/0907058

Seinfeld, J. & Pandis, S. (2006). *Atmospheric chemistry and physics: from air pollution to climate change* (2nd). A Wiley-Interscience publication. Wiley.

Šíp, V. & Beneš, L. (2016a). Investigating the street canyon vegetation effects using the moment method. In S. Sauvage, J. Sánchez-Pérez & A. Rizzoli (Eds.), *Proceedings of the 8th International Congress on Environmental Modelling and Software, july 10-14, Toulouse, France* (pp. 83–90).

Šíp, V. & Beneš, L. (2016b). CFD optimization of a vegetation barrier. In B. Karasözen, M. Manguoglu, M. Tezer-Sezgin, S. Göktepe & Ö. Ugur (Eds.), *Numerical mathematics and advanced applications - ENUMATH 2015*. Cham: Springer International Publishing.

Šíp, V. & Beneš, L. (2016c). Dry deposition model for a microscale aerosol dispersion solver based on the moment method. *arXiv e-print*. https://arxiv.org/abs/1605.033 97 (Submitted).

Šíp, V. & Beneš, L. (2016d). RANS solver for microscale pollution dispersion problems in areas with vegetation: Development and validation. *arXiv e-print*. https://arxiv. org/abs/1609.03427 (Submitted).

Šíp, V., Řezníček, H. & Beneš, L. (2015). *Šíření pevných částic ze silniční komunikace - CFD modelování*. Czech Technical University in Prague.

Slinn, W. (1982). Predictions for particle deposition to vegetative canopies. *Atmos. Environ. 16*(7), 1785–1794. doi:10.1016/0004-6981(82)90271-2

Sogachev, A., Kelly, M. & Leclerc, M. Y. (2012). Consistent two-equation closure modelling for atmospheric research: buoyancy and vegetation implementations. *Boundary Layer Meteorol. 145*(2), 307–327. doi:10.1007/s10546-012-9726-5

Steffens, J., Wang, Y. & Zhang, K. (2012). Exploration of effects of a vegetation barrier on particle size distributions in a near-road environment. *Atmos. Environ. 50*, 120–128. doi:10.1016/j.atmosenv.2011.12.051

Su, H.-B., Shaw, R. H., Paw, K. T., Moeng, C.-H. & Sullivan, P. P. (1998). Turbulent statistics of neutrally stratified flow within and above a sparse forest from large-eddy simulation and field observations. *Boundary Layer Meteorol. 88*(3), 363–397. doi:10.1 023/a:1001108411184

Sun, J., Nappo, C. J., Mahrt, L., Belušić, D., Grisogono, B., Stauffer, D. R., . . . Neff, W. D. (2015). Review of wave-turbulence interactions in the stable atmospheric boundary layer. *Rev. Geophys. 53*(3), 956–993. doi:10.1002/2015rg000487

Svanberg, K. (2002). A class of globally convergent optimization methods based on conservative convex separable approximations. *SIAM J. Optim. 12*(2), 555–573. doi:10.1 137/S1052623499362822

Svensson, U. & Häggkvist, K. (1990). A two-equation turbulence model for canopy flows. *J. Wind Eng. Ind. Aerodyn. 35*, 201–211. doi:10.1016/0167-6105(90)90216-Y

Tiwary, A., Morvanb, H. & Colls, J. (2005). Modelling the size-dependent collection efficiency of hedgerows for ambient aerosols. *J. Aerosol Sci. 37*, 990–1015. doi:10.1016 /j.jaerosci.2005.07.004

Tomas, J. M., Pourquie, M. J. B. M. & Jonker, H. J. J. (2016). Stable stratification effects on flow and pollutant dispersion in boundary layers entering a generic urban environment. *Boundary-Layer Meteorol*, *159*(2), 221–239. doi:10.1007/s10546-015-0124-7

Tominaga, Y. & Stathopoulos, T. (2007). Turbulent Schmidt numbers for CFD analysis with various types of flowfield. *Atmos. Environ. 41*, 8091–8099. doi:10 . 1016 /j.atmosenv.2007.06.054

Tominaga, Y. & Stathopoulos, T. (2013). CFD simulation of near-field pollutant dispersion in the urban environment: a review of current modeling techniques. *Atmos. Environ. 79*, 716–730. doi:10.1016/j.atmosenv.2013.07.028

Turkel, E. (1985). Algorithms for the Euler and Navier-Stokes equations for supercomputers. In *Progress and supercomputing in computational fluid dynamics* (pp. 155–172). Springer.

Turkel, E. (1987). Preconditioned methods for solving the incompressible and low speed compressible equations. *J. Comput. Phys. 72*(2), 277–298. doi:10.1016/0021-9991(87)90084-2

Turkel, E. (1999). Preconditioning techniques in computational fluid dynamics. *Annu. Rev. Fluid Mech. 31*(1), 385–416. doi:10.1146/annurev.fluid.31.1.385

Uehara, K., Murakami, S., Oikawa, S. & Wakamatsu, S. (2000). Wind tunnel experiments on how thermal stratification affects flow in and above urban street canyons. *Atmospheric Environment*, *34*(10), 1553–1562. doi:10.1016/s1352-2310(99)00410-0

Van Leer, B., Lee, W.-T. & Roe, P. L. (1991). Characteristic time-stepping or local preconditioning of the Euler equations. In *10th computational fluid dynamics conference* (Vol. 1, pp. 260–282).

Venkatakrishnan, V. (1995). Convergence to steady state solutions of the Euler equations on unstructured grids with limiters. *J. Comput. Phys. 118*, 120–130. doi:10.1006/jcph. 1995.1084

Vranckx, S., Vos, P., Maiheu, B. & Janssen, S. (2015). Impact of trees on pollutant dispersion in street canyons: a numerical study of the annual average effects in Antwerp, Belgium. *Sci. Total Environ. 532*, 474–483. doi:10.1016/j.scitotenv.2015.06.032

Weiss, J. M. & Smith, W. A. (1995). Preconditioning applied to variable and constant density flows. *AIAA Journal*, *33*(11), 2050–2057. doi:10.2514/3.12946

Whitby, E. & McMurry, P. (1997). Modal aerosol dynamics modeling. *Aerosol Sci. Technol. 27*(6), 673–688. doi:10.1080/02786829708965504

Whitby, E., McMurry, P., Shankar, U. & Binkowski, F. (1991). *Modal aerosol dynamics modeling*. U.S. Environmental Protection Agency. Research Triangle Park, NC 27711.

Wilcox, C. (1993). *Turbulent modelling for cfd*. DCW Industries Inc., California.

Wilson, N. R. & Shaw, R. H. (1977). A higher order closure model for canopy flow. *J. Appl. Meteorol. 16*(11), 1197–1205. doi:10.1175/1520-0450(1977)016<1197:AHOCMF>2.0 .CO;2

Yang, C. & Cai, X.-C. (2014). A scalable fully implicit compressible Euler solver for mesoscale nonhydrostatic simulation of atmospheric flows. *SIAM J. Sci. Comput. 36*(5), S23–S47. doi:10.1137/130919167

Zeytounian, R. K. (2003). Joseph Boussinesq and his approximation: a contemporary view. *Comptes Rendus Mécanique*, *331*(8), 575–586. doi:10.1016/s1631-0721(03)001 20-7

Zhang, L., Gong, S., Padro, J. & Barrie, L. (2001). A size-segregated particle dry deposition scheme for an atmospheric aerosol module. *Atmos. Environ. 35*(3), 549–560. doi:10.1016/S1352-2310(00)00326-5