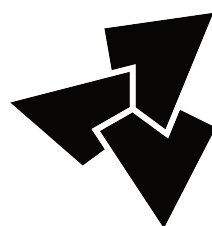


# Automatic reconstruction of 3D city models tailored to urban flow simulations

PhD Proposal  
23rd July 2021

**Ivan Paden**  
PhD Candidate, 2020 - 2024



**3D geoinformation**

Department of Urbanism  
Faculty of Architecture and the Built Environment  
Delft University of Technology

# Contents

<b>List of Abbreviations</b>	<b>vi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Research Objective . . . . .	2
1.3 Scope of the Research . . . . .	2
1.4 Structure of this Report . . . . .	3
<b>2 Related Work and Background Theory</b>	<b>4</b>
2.1 Urban Flows . . . . .	4
2.2 3D City Modelling . . . . .	7
2.2.1 CityJSON and CityGML . . . . .	7
2.2.2 Levels-of-Detail . . . . .	8
2.2.3 Validity . . . . .	8
2.2.4 Automatic reconstruction of semantic 3D city models . . . . .	10
2.2.5 Generalisation . . . . .	11
2.3 3D City Models in CFD . . . . .	11
2.3.1 Creation of CFD-Ready 3D Urban Environment Models . . . . .	12
2.3.2 Simplification . . . . .	14
2.4 Vegetation Modelling . . . . .	19
2.5 Uncertainty Quantification . . . . .	21
<b>3 Proposed PhD Research</b>	<b>24</b>
3.1 Problem Statement . . . . .	24
3.2 Research Objective and Questions . . . . .	25
3.3 Methodology . . . . .	26
3.3.1 Input Data and Preparation . . . . .	27
3.3.2 CFD-Ready Building Reconstruction . . . . .	27
3.3.3 Digital Terrain Model . . . . .	30
3.3.4 Outside the Influence Region . . . . .	30
3.3.5 Automatic Vegetation Modelling . . . . .	31
3.3.6 Data Parsing From 3D Models to CFD Software . . . . .	32
3.3.7 Computational Fluid Dynamics . . . . .	33
3.3.8 Uncertainty Quantification . . . . .	34
3.3.9 Validation . . . . .	35
<b>4 Preliminary Results</b>	<b>36</b>
4.1 CFD-Ready Reconstruction Framework . . . . .	36
4.1.1 Digital Terrain Model . . . . .	36
4.1.2 LoD1.2 Building Reconstruction . . . . .	36
4.1.3 CFD Domain and Meshing . . . . .	37

<b>5</b>	<b>Planning and Practical Aspects</b>	<b>40</b>
5.1	Timetable . . . . .	40
5.1.1	First Year Report . . . . .	40
5.2	Tools and Technical Aspects . . . . .	40
5.3	Graduate School Obligations . . . . .	40
5.4	Data Management Plan . . . . .	42
5.5	Publications . . . . .	42
5.5.1	Publication Plan . . . . .	42
5.5.2	Previous Publications . . . . .	43
5.6	Acknowledgements . . . . .	43
	<b>Bibliography</b>	<b>44</b>

# List of Figures

2.1	Atmospheric tempo-spatial scales, figure a combination of Orlanski (1975); Randerson (1976); Moussiopoulos et al. (2003), from Blocken (2014). . . . .	5
2.2	Schematics of a typical daytime boundary layer, from Oke et al. (2017). . . . .	5
2.3	Generic city geometry (up) and its corresponding finite volume mesh (below). . . . .	6
2.4	LoD definitions for buildings, from Biljecki et al. (2016b). . . . .	8
2.5	ISO 19017 primitives relevant for the modelling of the built environment, from Arroyo Ohori et al. (2021). . . . .	9
2.6	LoD2.2 reconstruction algorithm by Ravi Peters, from Arroyo Ohori et al. (2021). . . . .	10
2.7	Five domain areas in a typical urban flow simulation, from Blocken (2015). . . . .	12
2.8	CFD-ready geometry from different acquisition methods, from Hågbo et al. (2020). . . . .	13
2.9	Geometry for a CFD case with different LoDs for buildings, from van Hooff and Blocken (2010a). . . . .	15
2.10	The results of a sweep-plane simplification algorithm, from Piepereit et al. (2018). . . . .	16
2.11	Geometry simplification process proposed by Park et al., from Park et al. (2020). . . . .	16
2.12	Detailed 3D city model (left) and parameterised model with roughness patches (right) between the inlet (star) and the zone of interest (in square), from Liu et al. (2018). . . . .	17
2.13	The definition of the influence zone by Tominaga et al. (2008) (left), Tong et al. (2016) (middle) and Liu et al. (2018) (right), from Liu et al. (2018). . . . .	18
2.14	Different level of details for tree reconstruction from LiDAR data. . . . .	19
2.15	Vegetation modelling in CFD: (a) street canyon with trees; (b) basic approach omitting the vegetation; (c) increased roughness; (d) porous zones, from Hefny Salim et al. (2015). . . . .	19
2.16	Vertical profiles of LAD for two values of LAI, from Shaw and Schumann (1992). . . . .	21
2.17	Schematic representation of the two approaches to UQ, from Xiao and Cinnella (2019). QoI stands for the quantity of interest. . . . .	23
3.1	Geometry preparation workflow. . . . .	26
3.2	Schematic representation of the domain. . . . .	28
3.3	Building reconstruction workflow. . . . .	28
3.4	LoD1 trees reconstruction, from de Groot (2020). . . . .	32
4.1	Digital terrain model, detail. . . . .	37
4.2	Combination of the DTM and LoD1.2 buildings. . . . .	37
4.3	Domain used for finite volume mesh creation. . . . .	38
4.4	Computational domain made with <i>cfMesh</i> . . . . .	39
4.5	Problematic locations for mesh generation. . . . .	39
5.1	Planning overview . . . . .	41

# List of Tables

- 2.1 Validity issues and problems it create in CFD . . . . . 9
- 5.1 Tools and Technical aspects . . . . . 41
- 5.2 Graduate School Progress and Plan . . . . . 42

# List of Abbreviations

<b>ABL</b>	.....	Atmospheric Boundary Layer
<b>CAD</b>	.....	Computer-Aided Design
<b>CFD</b>	.....	Computational Fluid Dynamics
<b>CT</b>	.....	Constrained Triangulation
<b>CityGML</b>	.....	City Geography Markup Language
<b>CityJSON</b>	.....	City JavaScript Object Notation
<b>CWE</b>	.....	Computational Wind Engineering
<b>DTM</b>	.....	Digital Terrain Model
<b>FME</b>	.....	Feature Manipulation Engine
<b>FVM</b>	.....	Finite Volume Method
<b>GIS</b>	.....	Geographic Information System
<b>GML</b>	.....	Geography Markup Language
<b>ISO</b>	.....	International Organisation for Standardisation
<b>LAI</b>	.....	Leaf Area Index
<b>LAD</b>	.....	Leaf Area Density
<b>LES</b>	.....	Large Eddy Simulations
<b>LiDAR</b>	.....	Light Detection and Raging
<b>LOD</b>	.....	Level of Detail
<b>OGC</b>	.....	Open Geospatial Consortium
<b>PDF</b>	.....	Probability Density Function
<b>RANS</b>	.....	Reynolds Averaged Navier Stokes
<b>RSL</b>	.....	Roughness Sublayer
<b>SOSI</b>	.....	Systematic Organization of Spatial Information
<b>TIN</b>	.....	Triangulated Irregular Network
<b>UCL</b>	.....	Urban Canopy Layer
<b>UQ</b>	.....	Uncertainty Quantification
<b>XML</b>	.....	eXtensible Markup Language

# Chapter 1

## Introduction

### 1.1 Motivation

At present, 55% of world's population live in urban areas (Ritchie and Roser, 2018), and the UN World Population Prospects estimates this number will increase to 68% by 2050. While living in or near cities brings many benefits and, in general, higher living standard to individuals, there are rather strong concerns related to it as well. Of all the population living in urban areas, 90% are exposed to pollutant concentrations exceeding the guidelines prescribed by the World Health Organization (Pruss-Ustun and Corvalan, 2016). Cities tend to produce and retain more heat, an effect that is known as the urban heat island (UHI); the effect exposes citizens to discomfort and health-related problems, and also leads to higher energy demands for cooling (Moonen et al., 2012). Furthermore, pedestrian-level wind is exaggerated in groups of high-rise buildings which causes discomfort to pedestrians or can even result in casualties (van Druenen et al., 2019).

Computational wind engineering (CWE) has become a valuable tool in quantifying those issues and in developing mitigation strategies that can make cities a more pleasant, healthier, and safer place to live in. Recent CWE investigations have shown that minor alterations to urban features, such as adding canopies to buildings, can reduce strong winds up to 39% at the pedestrian level around high-rise buildings (van Druenen et al., 2019). Research by Jia and Wang (2021) targeting thermal comfort has found that peak temperatures in a city can be reduced by over 4 °C with tree plantation. McNabola et al. (2009) investigated the placement of low boundary walls (1 - 2 m) in street canyons; the results indicated possible reductions in pollutant concentration by up to 75%, depending on the wind direction. Many of the mitigation strategies that involved CFD as a tool are covered in Li et al. (2021). These examples show that investing resources in the development of tools used in CWE contributes towards very important goals.

In 2014, NASA published the CFD Vision 2030 Study (Slotnick et al., 2014) with the state-of-art overview and the research strategy to address the main issues in the whole of the CFD field. They acknowledged that CFD workflows spend the most of time on geometry pre-processing and grid generation. Geometry preparation was recognised as one of the main bottlenecks in present CFD simulations. The study also expressed the urge to achieve a much higher degree of automation in all analysis process steps, with geometry creation exhibited as the prime example. Years later, attempts to confront this issue have been made, albeit with little progress. Computational wind engineering, which deals with some of the largest temporal and spatial scales in the field, still requires a considerable amount of manual labour during the pre-processing step. Blocken (2021) recently confirmed that geometry preparation and meshing still occupies roughly 80% of human hours. Moreover, geometries can be modelled at different levels of detail (as will be illustrated in Section 2.2); the effect of levels of detail on the predictability of CFD simulations has not been studied yet. Lastly, guidelines on geometry modelling are practically non-existent.

On the other hand, the geomatics field has established standards and has undergone noticeable achievements in 3D city modelling. Researchers have developed algorithms and procedures that can automatically reconstruct entire countries (Airaksinen et al., 2019; Dukai et al., 2021; Ledoux et al., 2021). Initially used for visualisation purposes, semantic 3D city models have grown to be a part of many applications (Biljecki et al., 2015). Goals and methodologies in geoinformation tend to be comprehensive and applicable to as many different fields as possible. However, this means that application-specific requirements need to be addressed.

In CWE, the level of detail in geometry depends on the meshing process which is lead by the available computational power and numerical method requirements for the subsequent simulation, as well as the limitations of the meshing software. Consequently, there is a need for a certain amount of flexibility with the geometry creation; above all, it is necessary to provide it in an efficient and fully automated way. Additionally, semantic 3D models are only used in CWE for their geometry; notwithstanding the potential benefits, nobody has tried to capture semantic data useful for CWE, such as roughness, and store it as a part of the city model.

All the mentioned issues provide strong motivation for a research topic — one that is interdisciplinary and addresses bottlenecks in computational wind engineering using expertise from 3D city model reconstruction.

## 1.2 Research Objective

The goal of my research is to: *develop a framework that automatically reconstructs semantic three-dimensional urban environment models optimised for CFD at different levels-of-detail.*

The research questions are refined in Chapter 3.

## 1.3 Scope of the Research

- Main bottlenecks in the simulation of urban flows (e.g. wind flow and pollutant dispersion), particularly related to largely manual geometry preparation, will be pinpointed.
- The framework will provide a set of algorithms and rules that enable the automatic creation of urban environment geometry acceptable for *Finite Volume Method* (FVM) meshers.
- The input for the framework will be the combination of the data readily available in many countries, such as 2D GIS and point cloud elevation data, or 2D GIS with height data for individual buildings.
- This work will consider buildings, terrain and vegetation.
- This work will focus on two distinct areas:
  - Zone of influence where buildings are explicitly constructed at different levels of details (LoDs), and
  - The rest of the domain where buildings are implicitly modelled by either surface roughness or additional terms in transport equations.
- This work does not aim to develop algorithms for the FVM meshing process, but rather focuses on identifying important parameters for open-source meshing algorithms that could greatly aid potential users.
- The response of urban flows to different LoDs and simplifications will be systematically quantified using an uncertainty framework.



- Verification and validation of the framework will be conducted by comparing simulation outputs to available wind tunnel and field measurement data.

## 1.4 Structure of this Report

This proposal is organised in 5 chapters:

- The current chapter introduces the motivation guiding the research, the research objectives and the scope of the research.
- Chapter 2 summarises related work as well as the relevant background knowledge and theory necessary for conducting the research.
- Chapter 3 explains the proposed research strategy including the research questions and methodology.
- Chapter 4 discusses my initial results.
- Chapter 5 provides an overarching view of the research plan and the practical aspects of completing the degree requirements including the timeline and graduate school obligations.

# Chapter 2

## Related Work and Background Theory

This section gives a general overview of the theory directly related to the research proposal and an overview of the related work, with a particular emphasis on recent developments. As my topic is interdisciplinary, laying on the border of two different fields, there might be some collisions in the terminology. My aim is to address them accordingly.

### 2.1 Urban Flows

Atmospheric flows contain large temporal and spatial scales, ranging from seconds to years and from millimetres to thousands of kilometres, respectively (Figure 2.1). They can be divided into three distinct categories based on horizontal scales (Orlanski, 1975):

- *Macroscale* or *synoptic* scale: large scale atmospheric motions ranging from hundreds to thousands of kilometres, such as cyclones or even global climate.
- *Mesoscale*: from few to several hundreds of kilometres; includes phenomena such as cloud formation, thunderstorms, precipitation.
- *Microscale*: ranges from few hundreds of meter to few kilometres; focuses on phenomena caused by objects in the vicinity of the ground, e.g. corner vortices, canyon circulation, building wakes, urban microclimate (Oke et al., 2017).

The atmospheric boundary layer (ABL), shown in Figure 2.2, is the bottom layer of the troposphere, in direct contact with the Earth's surface (AMS). Local small-scale phenomena happening in the urban canopy layer (UCL) are not of interest in the two larger-scale models. Thus, they are not modelled explicitly; they are approximated, for example, with aerodynamic surface roughness length. This way, the whole city can be replaced with one value and an equation that models the influence of the city on the ABL.

Microscale models solve the transport of quantities of interest around geometrical features, i.e. explicitly modelled obstacles. They are generally referred to as CFD models (Blocken, 2015). The computational domain, space where the fluid resides, is divided into finite volumes. In urban flows, this is the area between buildings and other urban features where the wind flows (Figure 2.3). A set of partial differential equations, called transport equations, is solved for every finite volume, whose general formulation in the differential form is

$$\frac{\partial}{\partial t} (\rho\phi) + \nabla \cdot (\rho\mathbf{v}\phi) = \nabla \cdot (\Gamma\phi\nabla\phi) + Q^\phi, \quad (2.1)$$

where the first term on the left hand side is the unsteady term, the second term is the convection term, the first term on the right hand side is the diffusion term, and the last term is the source/sink

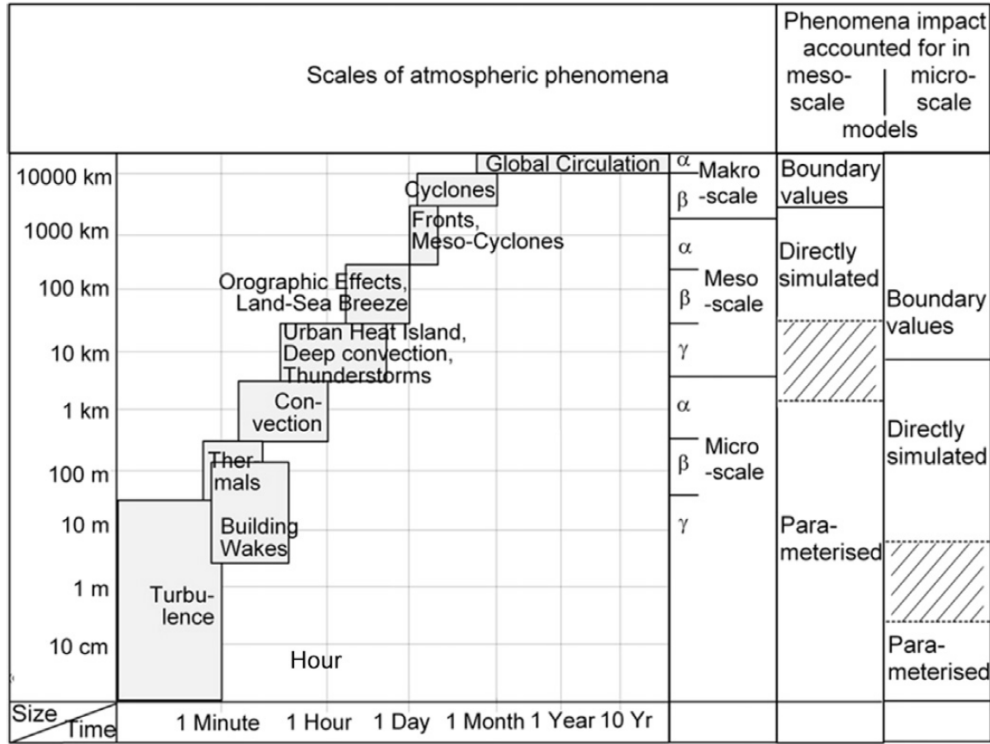


Figure 2.1: Atmospheric tempo-spatial scales, figure a combination of Orlanski (1975); Randerson (1976); Moussiopoulos et al. (2003), from Blocken (2014).

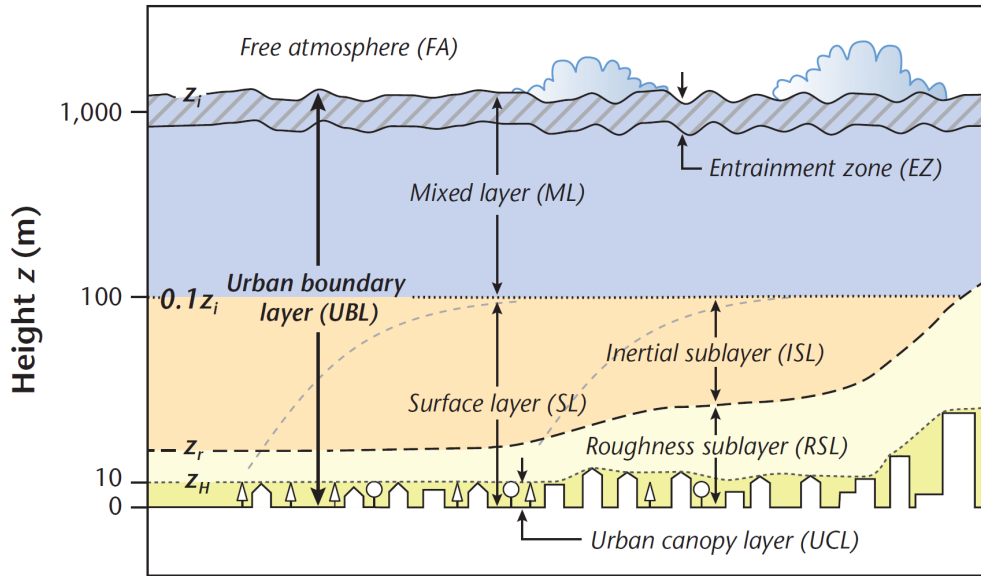


Figure 2.2: Schematics of a typical daytime boundary layer, from Oke et al. (2017).

term;  $\rho$  denotes the density,  $\phi$  is the quantity of interest,  $\mathbf{v}$  is the velocity vector and  $\Gamma$  is the diffusion coefficient (Moukalled et al., 2016). By assigning correct values to  $\phi$ ,  $\Gamma^\phi$  and  $Q^\phi$ , one can obtain the differential form of any conservation equation, for example, continuity, linear momentum, energy or species transport. There are other methods to model fluid flow, but the finite volume method (FVM) is among the most popular (Rapp, 2017).

Geometrical features such as buildings represent an obstacle, or a boundary of the domain. This

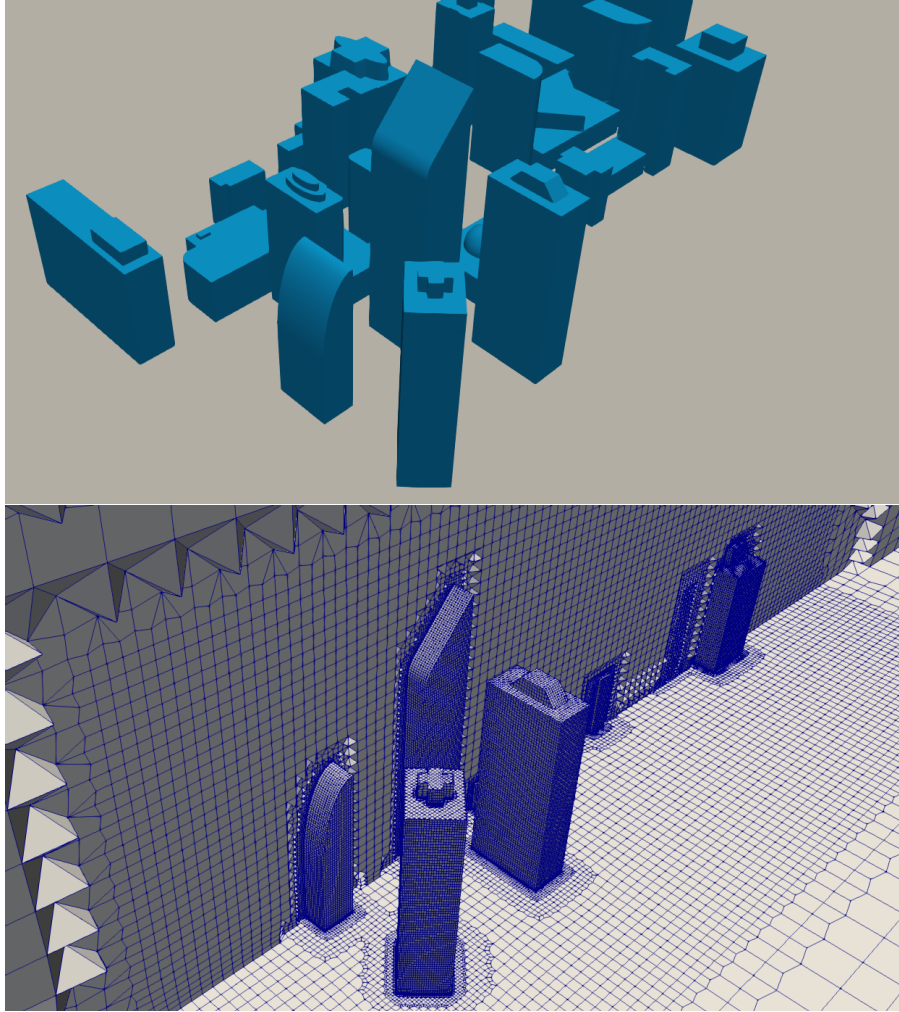


Figure 2.3: Generic city geometry (up) and its corresponding finite volume mesh (below).

boundary can be modelled either as a body-fitted mesh or approximated with a group of methods known as the immersed boundary (Constant et al., 2017). The number of control volumes directly affects the turnaround time of the simulation. Buildings can be (re)constructed at different levels of detail (LoDs) depending on the quality of input data, computational power, zone of interest, and capabilities of CFD pre-processing tools; the concept of LoD and its application in CFD is explained in more detail in Sections 2.2 and 2.3. Explicitly reconstructing the whole domain of simulation upstream and downstream of the area of interest is prohibitively demanding for the most present-day computational resources. To make simulations feasible, mathematical representation in the form of porosity or drag force substitutes the geometry. One of few notable exceptions to this is Ashie and Kono (2011), where the authors simulated 23 wards of Tokyo, a 32x32 km area. They used 5 billion control volumes, a number that is demanding even for modern high-performance computing clusters.

Both the size and the quality of the computational mesh are the critical points in reaching a stable and accurate numerical solution. The quality of the finite volume mesh is a broad topic, but generally speaking, the highest quality can be reached with a structured hexahedral mesh (Baker, 2005). Structured mesh is made manually, so a city-scale geometry takes a long time to finish. Automatic unstructured mesh generators can produce a good quality mesh, but this depends on the preparedness of the computer-aided design (CAD) model<sup>1</sup>. Small features such as chimneys and windows can severely degrade the mesh quality as they force the mesh to deviate from a regular shape. Geometric

---

<sup>1</sup><https://www.pointwise.com/case-studies/5fa46c815787332e68115137>

simplification aims to alleviate this problem by removing small geometrical features, lowering the LoD of buildings and/or parameterising the whole regions of the domain. Section 2.3.2 reviews the topic of simplification in more detail. These simplifications can affect the results and uncertainties attached to them, but still may offer acceptable accuracy of the larger scale motions over the whole city. It is a balancing act and one of the main outstanding issues in the CFD community (Mirzaei, 2021). It is also one of the key scientific goals of this PhD research.

Initially, there used to be a distinct separation between the different tempo-spatial scales; in recent years, researchers are focusing on the coupling between the mesoscale and the microscale simulations, where the larger scale is used as a boundary condition for the smaller scale (Muñoz-Esparza et al. (2014); García-Sánchez and Gorlé (2018); Temel et al. (2018); Piroozmand et al. (2020) to name a few).

## 2.2 3D City Modelling

The 3D modelling of the built environment involves the creation, manipulation and use of 3D digital representations of real-world objects, including buildings, terrains and infrastructure (Arroyo Otori et al., 2021). These representations usually include a mix of geometric, topological and semantic information (Arroyo Otori et al., 2021), some of which are shared among different applications, and others that are application-specific. Initially used for visualisation, the applications of 3D city models rose rapidly in the last decade. Biljecki et al. (2015) made a comprehensive review of 29 use cases, one of them being CFD. Nevertheless, application-specific requirements have not yet been defined for CFD uses; 3D city models are used exclusively to extract the geometry and place it through a largely manual and tedious *geometry preparation* step, explained in Section 2.3.

### 2.2.1 CityJSON and CityGML

Many formats can be used to represent and store 3D urban objects. The list includes Wavefront OBJ, PLY, gITF, etc., with most of them primarily used for visualisation. What is common to most of those models is that they lack support for semantics and attributes. Arroyo Otori et al. (2021) explained that a semantic 3D city model is a data model which “is decomposed into classes that we deem relevant for certain applications, for instance the city is decomposed into the classes ‘building’, ‘road’, ‘tree’, ‘lamppost’, etc. and each of the objects has its own 3D geometry and potentially (thematic) attributes (e.g. the owner of a building, the name of street, the city identifier for a lamppost, etc.)”. To illustrate the applicability of semantics in CFD: a building is composed of walls and a roof, where both of those classes can have different values of the roughness attribute attached to them. See Section 2.3.2 for the roughness concept. Open Geospatial Consortium (OGC) adopted the CityGML data model as the international standard that represents semantic 3D models of cities and landscape (Open Geospatial Consortium, 2012). It contains classes commonly found in an urban context, such as buildings, bridges, vegetation, and bodies of water to name the few; it also incorporates hierarchical relationships between them, e.g. a building is composed of parts, which are formed of walls, which have windows (Ledoux et al., 2019).

The most significant issue of CityGML is its XML-based encoding. XML requires special libraries to handle the data, and GML has several different ways to store the same geometry (Ledoux et al., 2019). All that makes CityGML very hard to parse or extract information from (Arroyo Otori et al., 2021). The sheer complexity of CityGML hardly motivates its application in other fields. If we take a glance at the review of CityGML’s application domain extensions (ADE - application-specific augmentations to the data model) by Biljecki et al. (2018b), we will notice there is not a single one related to CFD.

Ledoux et al. (2019) addressed the main issues of CityGML encoding; they created a new JSON-based encoding, CityJSON, that tackles most of the disadvantages CityGML has. Not only that,

CityJSON files are on average six times more compact in size (Ledoux et al., 2019).

## 2.2.2 Levels-of-Detail

Biljecki et al. (2014) formalised the definition of the level of detail (LoD) in 3D city modelling as the degree of correspondence between the model and the real-world object, being driven by the geometry, appearance, semantics and other related metrics which can be quantified, with separated exterior and interior concept. CityGML prescribed five standard LoDs for exterior representation (Figure 2.4a). Biljecki et al. (2016b) extended the CityGML classification with a set of 16 LoDs to remedy ambiguities that had arisen from the original classification. These are now referred to as “TUDelft LoDs”.

To the best of my knowledge, there have not been attempts to adopt any of the LoD for buildings in urban flows. From the CWE standpoint, the TUDelft classification is more useful than the one from CityGML. My statement is supported by Mirzaei (2021) who proposed the introduction of TUDelft LoDs in CFD applications. For example, Ricci et al. (2017) compared results of wind flow field between what would be considered as LoD1.1 and LoD1.3; they found considerable differences in the flow field between the two classifications. I will review this work in more detail in Section 2.3.2.

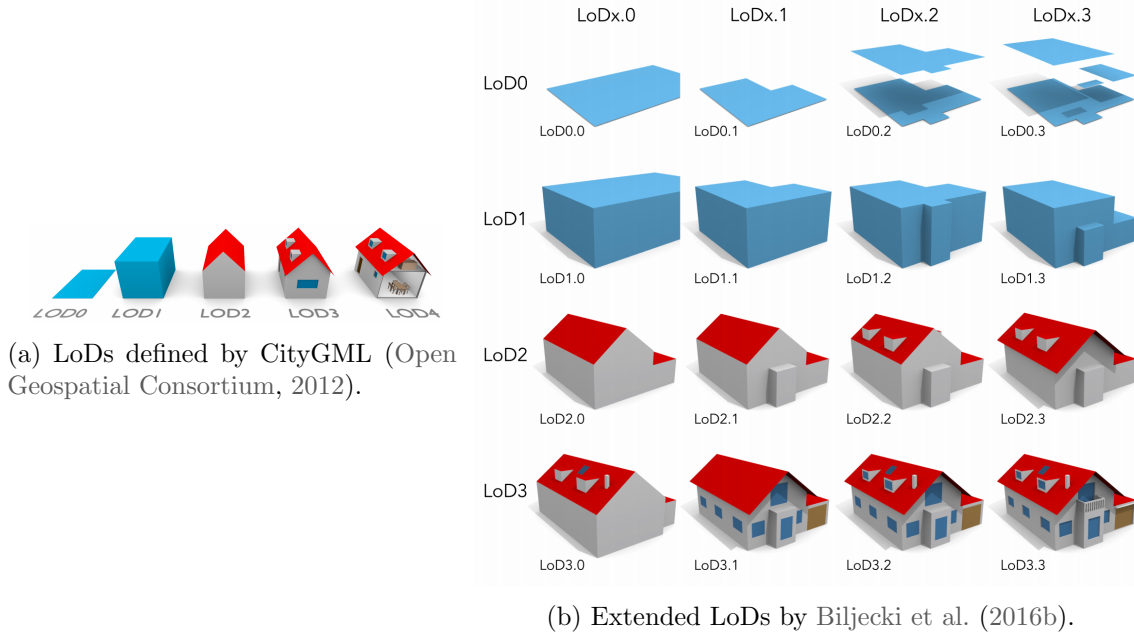


Figure 2.4: LoD definitions for buildings, from Biljecki et al. (2016b).

## 2.2.3 Validity

CityGML data model prescribes the subset of ISO 19107 (ISO, 2003) standard to represent the geometry of its 3D objects. Figure 2.5 shows primitives relevant for the modelling in the built environment. In short, a 3D object should be 2-manifold, should have consistently oriented faces, should not have duplicate vertices nor self-intersecting geometries (Arroyo Otori et al., 2021). The two restrictions that the CityGML imposes are:

1. GM\_Curves can only be linear (thus only LineStrings and LinearRings are used);
2. GM\_Surfaces can only be planar (thus Polygons are used).

Typical requirements for a geometry later used for finite volume mesh generation are: clean, free of defects, watertight (Ho, 2019). Beall et al. (2004) marked gaps and overlaps as the most common

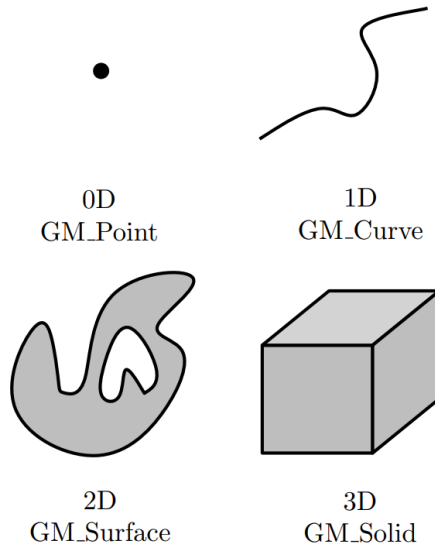


Figure 2.5: ISO 19017 primitives relevant for the modelling of the built environment, from Arroyo Ohori et al. (2021).

defects in geometry preparation. The authors also acknowledged dangling faces as one of the common reasons automatic mesh generation algorithms can fail. Some automatic finite volume mesh generators manage to work around a few of the problems, but it is rarely without issues that require manual repair.

Looking at CityGML specifications and then back at CFD requirements, it is safe to assume that a valid ISO 19107 geometry is one of the foundations for a CFD-ready geometry. To investigate to which extent 3D datasets are truly valid, Ledoux (2013) proposed a methodology which resulted in an open-source software *val3dity* (Ledoux, 2018). Unfortunately, Biljecki et al. (2016a) observed that perfectly valid 3D datasets are scarce. The good news is that in the recent years the geomatics community recognised the importance of valid datasets, and the latest geometry reconstruction algorithms aim to avoid as many errors as possible. Table 2.1 shows some of the validity issues and the importance of avoiding them in CFD according to my own experience and the experience of others<sup>2</sup> (Sadrehaghghi, 2018).

Table 2.1: Validity issues and problems it create in CFD

Issue	Severity in CFD
Non-manifold	Severe
Non-watertight	Severe
Intersections	Severe
Duplicated outer surfaces	Severe
Wrong orientation	Moderate
Duplicated vertices	Moderate
Duplicated/missing inner surfaces	None

Additional issues not covered by ISO19107 requirements for CFD include: small features, small edges, and small gaps between buildings. The first two issues are handled with simplification; see Section 2.3.2 for more details. The third one with the footprint generalisation (Section 2.2.5).

<sup>2</sup><https://knowledge.autodesk.com/support/cfd/learn-explore/caas/CloudHelp/cloudhelp/2019/ENU/SimCFD-UsersGuide/files/GUID-6758F06E-D52E-4191-B0FE-4A0A11EBC457-htm.html>

## 2.2.4 Automatic reconstruction of semantic 3D city models

Automatic building reconstruction is a balance between the data-driven and model-driven approaches; it combines the input data and modelling assumptions on the building shape to reach the main goals: low complexity, high accuracy and geometric validity (Arroyo Ohori et al., 2021). The importance of respective goals varies from one application to another, so some degree of flexibility in algorithms is necessary. The 3D geoinformation research group has expertise in the development of building reconstruction algorithms. Here, I will present a few of them.

Ledoux et al. (2021) developed an LoD1.2 footprint extrusion framework called *3dfier*. The framework uses 2D geographical datasets and “lifts” them to a height calculated from elevation data such as LiDAR. It also defines a set of rules on feature reconstruction based on the semantic data from the 2D dataset. For example, water polygons are horizontal, buildings are prismatic, and roads are smooth. *3dfier* was used by Deininger et al. (2020) as a basis for geometry preparation in their semi-automatic workflow for urban flows (see Section 2.3.1).

Ravi Peters developed an LoD2.2 automatic reconstruction algorithm whose overview is presented in Arroyo Ohori et al. (2021). The algorithm, just like *3dfier*, uses 2D geographical data and a point cloud, more specifically the Dutch Register of Buildings and Addresses (BAG) and the Dutch National Height Model (AHN). The algorithm was used as a part of the 3D BAG project (Dukai et al., 2021) that successfully reconstructed 10 million buildings in the Netherlands. The algorithm, schematically shown in Figure 2.6, detects roof planes (step 2 in the figure) and lines (step 3) from the point cloud, regularises and clusters them (step 4) to create the so-called *roof partitions* — a planar partition of the footprint where each face corresponds to a planar piece of the roof and is labeled with a roof plane (step 5). In the end, roof partitions are lifted to their respective heights, and with vertical walls they form a final building model (step 6). Geometric validity was one of the main goals during the development of the algorithm; with that, the 3D BAG database contains around 85% valid buildings. Limitations to this method is that it constructs 2.5D geometry and it is piecewise planar.

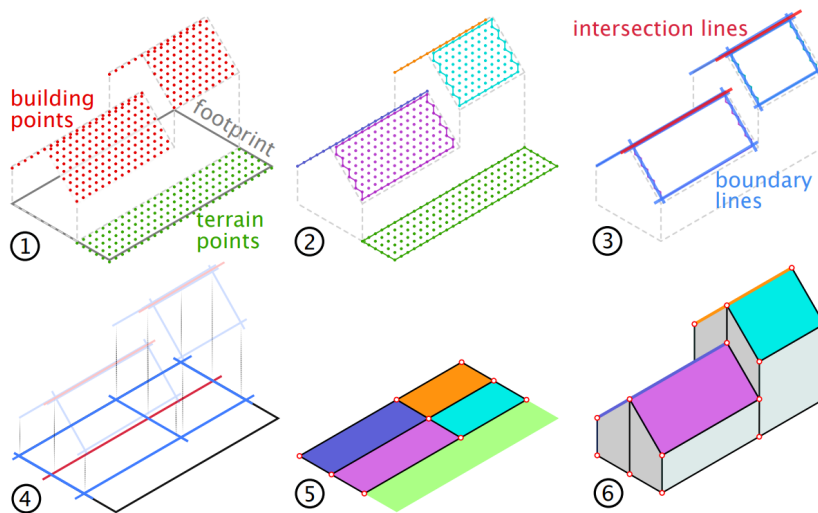


Figure 2.6: LoD2.2 reconstruction algorithm by Ravi Peters, from Arroyo Ohori et al. (2021).

Nan and Wonka (2017) developed a reconstruction method that intersects primitives (planes) and seeks an appropriate combination of them to obtain a manifold and watertight polygonal surface mode. The method assumes that all surfaces are detectable from a point cloud, but this is never the case in the airborne LiDAR — some surfaces do not get captured by the laser beam due to having obstacles (other surfaces) in their way, i.e. the *occlusion* effect (Ledoux et al., 2020). Additionally, the method is not optimised for large datasets. However, with a quality point cloud, the method can produce 3D



geometry whose complexity (for instance small features like chimneys) can be regulated with a model complexity term. Huang (2020) is currently working on a PhD topic that uses a similar method as Nan and Wonka (2017), but detects and adds missing planes for the algorithm and enables city-scale applicability, making the potential outcomes beneficial for my topic.

### 2.2.5 Generalisation

Generalisation provides the opportunity to rectify inconsistencies in detailed models and to simplify data for application-specific usage (Labetski et al., 2017). Even though *simplification* is just one aspect of generalisation, the CFD community generally uses that term for any type of generalisation and implicit modelling of geometrical features with mathematical formulae. For 3D city models, 2D footprint generalisation is fundamental as it can efficiently solve many problems, especially for buildings obtained from extrusion. Commandeur (2012) proposed a generalisation algorithm tailored specifically for building footprints, based on the 3D generalisation algorithm by Kada (2008). The algorithm uses distance and angle threshold as a parameter, potentially very useful in CFD applications.

There have been attempts to make 3D generalisation specifically for CWE, but only with limited success. Section 2.3.2 presents those methods as a part of a larger topic of simplification in CFD. For LoD2 buildings, the reconstruction algorithm should offer some flexibility in the extent of details that are being reconstructed. This way, the whole workflow can be less complicated and presumably more time-efficient than a generalisation step after reconstruction. The automatic reconstruction algorithm used in Dukai et al. (2021) does precisely this; it conducts graph-cut optimisation of planar roof partitions, which directly affects the geometric complexity of the output.

## 2.3 3D City Models in CFD

Unlike 3D city models where each feature (building, road, tree, etc.) is represented with geometry and attributes, 3D city models in CFD consist of different areas, some explicitly modelled with geometries and the others that are implicitly modelled. Blocken (2015) made a distinction of five areas of the computational domain widely used for urban flow simulations (Figure 2.7):

- Area 1: The area upstream of the inlet which is taken into account with an inflow boundary condition and an aerodynamic roughness length  $z_{0,1}$  defined with the updated roughness classification of Davenport (Wieringa, 1992).
- Area 2: The area between the inlet and the explicitly modelled buildings. Buildings and other obstacles here are simplified, i.e. implicitly modelled with aerodynamic roughness length  $z_{0,2}$ , also defined by Wieringa (1992), as a porous media or additional drag terms. Modelling of that area is further discussed in Section 2.3.2.
- Area 3: The area amongst buildings in the *influence region* or the *area of interest*. It includes implicitly modelled features such as sidewalks, poles, trees, etc. This area can be modelled the same way as Area 2 - with aerodynamic roughness length and porosities/drag terms.
- Area 4: The explicitly modelled buildings in the *influence region* or the *area of interest* that can be created in different LoDs. The rough surfaces of those buildings (facades, windows, balconies) can be implicitly modelled with a standard wall function (Launder and Spalding, 1974) with a sand-grain roughness height  $k_s$  modification (Cebeci and Bradshaw, 1977).
- Area 5: The area between the outlet patch and the influence region. This area can once again be implicitly modelled with the aerodynamic roughness length  $z_{0,5}$ .

It is worth noting that specific inflow boundary conditions (i.e. changing the inflow profile) cannot account for perturbations caused by the city upstream of the zone of interest (Area 2 in Figure 2.7). Liu et al. (2017) compared the full city model against a district of interest with enhanced roughness

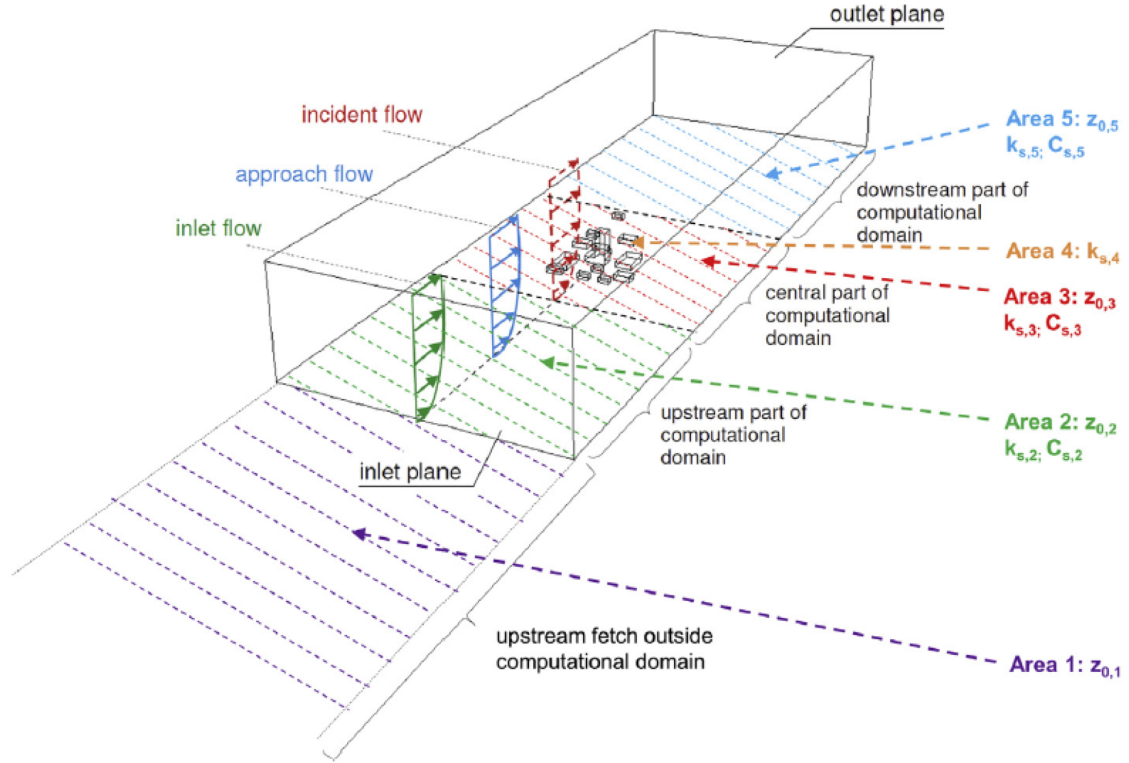


Figure 2.7: Five domain areas in a typical urban flow simulation, from Blocken (2015).

length for the inflow condition. They showed that the modified inflow boundary does not capture the wind decay through the city sufficiently, leading to a gross overestimation, by a factor of two, of wind velocity at the zone of interest.

When talking about explicitly modelled buildings (Area 4 in Figure 2.7), I have to add few important remarks. First, the concept of LoD has not found its way to CFD. Mirzaei (2021), in one of the latest review papers on urban flows, recognised the necessity for LoDs and proposed the introduction, using Biljecki et al. (2016b) as an example. Even more, there is no established standard on storing geometry semantic data for CFD applications, nor research papers outline details on how they handled semantics. Lastly, as already noted in the previous section, there are no guidelines nor application-specific requirements on 3D city models for CWE. CFD practitioners solely use 3D city models to extract geometry in one of the formats accepted by most CFD software (STL, STEP, IGES, Wavefront OBJ); what follows is the *geometry preparation* step explained in the next section.

### 2.3.1 Creation of CFD-Ready 3D Urban Environment Models

The initial step of acquiring the unprepared geometry depends on the already available dataset or methods introduced in Section 2.2; here, I will focus on the specifics and examples related to CFD. These geometries are not ready for the finite volume mesh generation step, but typically undergo the *geometry preparation* step consisted of *geometry cleanup* (Simões and Estanqueiro, 2016) and *simplification* (Section 2.3.2). The geometry preparation refers to the repair of issues with the geometry: non-manifold edges, filling holes, resolving conflicting surfaces and other degenerate inputs, as well as removal features that are smaller than the targeted finite volume mesh size. This process has been automated to a certain extent. Lu et al. (2011) developed a framework that automatically cleans extruded polygons in order to create an error-free, CFD-ready 3D city model. The main idea was to generate polygon layers, repair those layers and then stitch them to get the final model. The main methods used were the  $k$ -way boolean and the Minkowski sum operations. Even though the framework showed to be computationally efficient and robust, the output mesh showed small, sharp features and

sliver triangles, meaning not all issues could have been avoided. There are many open-source and commercial software packages that offer geometry repair to some degree. Saeedraashed and Benim (2019) made a review of seven of them, specifically checking their usability for urban wind simulations. They concluded that no software could offer automatic geometry repair at the time to the point that it can be considered simulation-ready.

Buildings in CFD simulations are commonly created in LoD1, be that as LoD1.1, LoD1.2 or LoD1.3. As I already mentioned, the concept of LoD is still not recognised in the CWE community, but I took the liberty to categorise other works myself. Zhang et al. (2021) used GIS data with storey information to automatically reconstruct an LoD1.2 geometry of a city. They used 3 m as a storey height approximation. Gao et al. (2018) had previously employed the same approach of extruding polygons. Very few authors used LoD2 geometries on a larger scale, i.e. a city district or more. An example of that is Toja-Silva et al. (2018) and Toja-Silva et al. (2017), where the authors used the LoD2.1 model of a district in Munich. They noted that it took them a long time and a lot of manual effort to get a geometry that is adequate for a numerical simulation.

Hågbo et al. (2020) made a comparison of four different geometry acquisition methods: basic footprint extrusion, the Norwegian national feature catalogue FKB (Felles KartdataBase) stored in the geographical information format SOSI (Systematic Organization of Spatial Information), photogrammetry and LiDAR point clouds. They manually processed, i.e. cleaned all of the investigated models before running any simulations, with the results shown in Figure 2.8. Their simulation on pedestrian wind comfort showed similar results between the second, third and fourth acquisition methods, indicating that LoD1.2 footprint extrusion might not be sufficient. The authors concluded that the FKB model is the most promising for further applications, as it is easier to use than the two point cloud-based models. Furthermore, they referred to the point cloud processing step as “time-consuming”. Finally, they did not work on automation of the workflow, nor tried to combine different acquisition methods to exploit their advantages, as done in 3D BAG (Dukai et al., 2021).

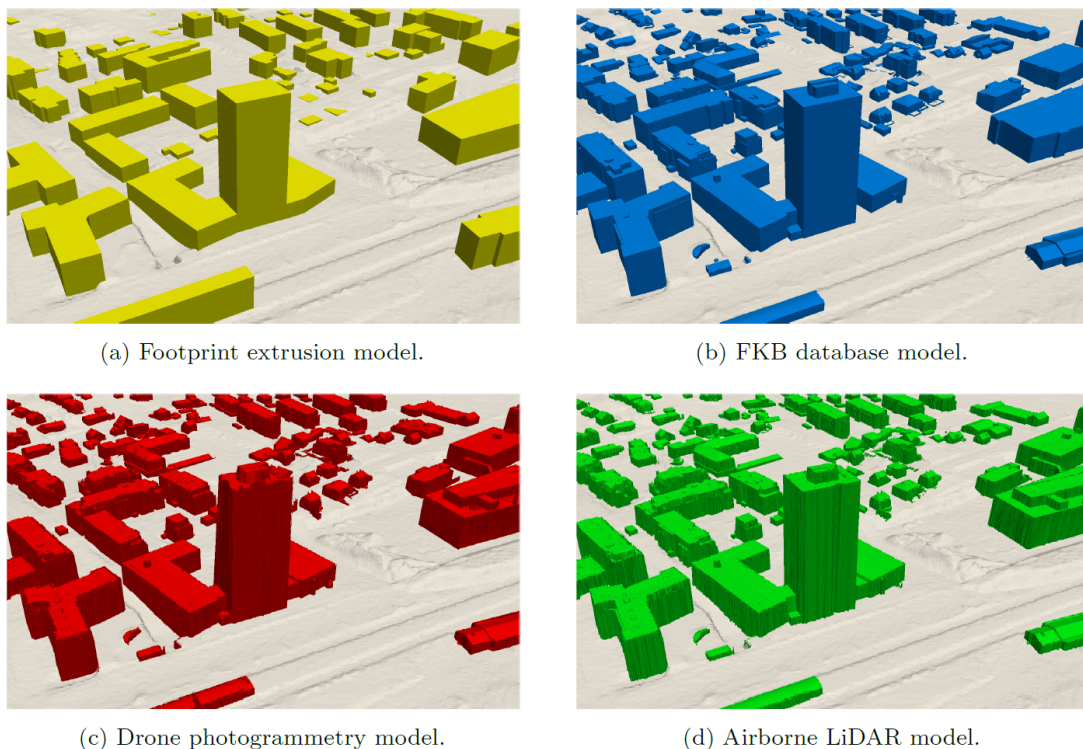


Figure 2.8: CFD-ready geometry from different acquisition methods, from Hågbo et al. (2020).

The example of a largely automated workflow can be found in Deininger et al. (2020). The authors

created a workflow that combined FME<sup>3</sup> for point cloud processing, 3dfier (Ledoux et al., 2021) for 3D city modelling and workflow by Piepereit et al. (2019) for geometry repair and simplification. The process is largely automated; however, there is still a need for some manual geometry repairs that authors conducted in ANSYS SpaceClaim. The following section will give more information on the simplification algorithm. Lastly, even the latest review papers on urban flows (Mirzaei, 2021) express the need for automated algorithms that can reconstruct urban areas on a larger scale.

### 2.3.2 Simplification

Some features are too complex (i.e. they create too many finite volume cells or bad quality mesh) to model them explicitly. The examples include short edges and gaps between buildings, balconies and windows, vegetation, sidewalks, lamp posts, or even whole city regions away from the area of interest (Blocken et al., 2012). This section gives an overview of simplifications typically used to make urban flow simulations feasible. By simplification, I am referring to a set of algorithms, models, and rules that replace the high-detail explicitly modelled urban features with computationally more efficient approximations, mainly for the momentum exchange. It means that even there is some overlap, *simplification in CFD is not the same as a simplification in geoinformatics*. I will also mention the usage of different LoDs as a part of the simplification process.

The following section is divided into two distinct categories: simplifications done in the influence region and outside of it. As mentioned earlier in this section, the influence region is the part of the domain where the buildings are modelled explicitly. Outside this region buildings are parametrised. Please note that vegetation modelling and its simplification are covered separately in Section 2.4.

#### Simplification within the influence region

The best practice guidelines (Blocken, 2015; Franke et al., 2007) ask for the highest detail of buildings depending on available resources, without any other specific requirements. The accepted practice in CFD is to remove features that are smaller than the targeted mesh size (Beall et al., 2004). The widespread use in the literature shows this is also very much represented in CWE (Porter, 2020). The combination of different LoDs within the influence region seems to be a prevalent option among researchers, as done in Yoshie et al. (2007); van Hooff and Blocken (2010a); Montazeri et al. (2013); Ricci et al. (2020a).

To give an example, van Hooff and Blocken (2010a) validated a natural ventilation case where the building of interest was reconstructed in a high LoD, whereas the surrounding buildings in the influence region had both the lower LoD and less detailed computational mesh (Figure 2.9). The works mentioned above do not give much information on the geometry modelling nor simplifications, but by visually observing, I can conclude they mostly resorted to LoD1.1 and LoD1.2 specifications according to classification by Biljecki et al. (2016b). Moreover, none of the works investigated the effect of those simplifications.

Ricci et al. (2017) published one of the first works that deal with the consequences of geometry simplifications on the wind flow field. They compared geometries of a city block reconstructed in what I would classify as LoD2.1, LoD1.3, and LoD1.1. They found substantial differences between LoD1.3 and LoD1.1, but also a satisfactory agreement between LoD2.1 and LoD1.3. The authors also addressed that the time required to create LoD2.1 geometry and the subsequent simulation took five times longer than LoD1.3. Even though lacking the uncertainty quantification, this research has shown that LoD1.1 might not be enough to capture the flow field in the influence region accurately. The comparison of different acquisition methods by Hågbo et al. (2020) also showed that there is a notable difference between LoD1.2 and LoD1.3 for pedestrian wind comfort simulations.

---

<sup>3</sup><https://www.safe.com/fme/>

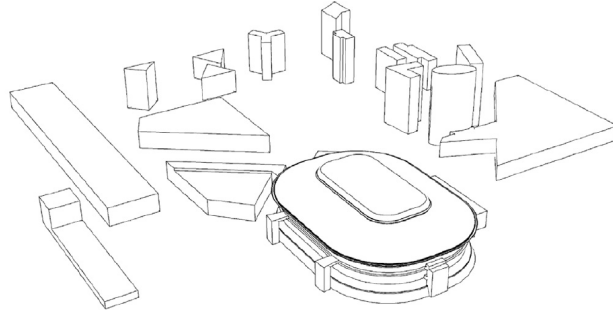


Figure 2.9: Geometry for a CFD case with different LoDs for buildings, from van Hooff and Blocken (2010a).

Certain works aimed to simplify geometry as least as possible, only in places where it could potentially cause problems for finite volume mesh generators. The idea here is to reduce the number of finite volume cells and increase the quality of the resulting finite volume mesh without a hefty penalty on simulation results. This on par with the generalisation concept in 3D city modelling explained in Section 2.2.5. Piepereit et al. (2019) proposed a simplification framework that incorporates the merging of edges and faces, the combination of CGAL’s union operator (Hachenberger and Kettner, 2021) and Minkowski sum (Hachenberger, 2021), sweep-plane algorithm (Piepereit et al., 2018) and coons algorithm (Piepereit et al., 2016). The core of the simplification is the sweep-plane algorithm that iteratively eliminates edges that are shorter than a given threshold by sweeping nearby faces (Figure 2.10). Deininger et al. (2020) employed the framework with a distance threshold of 2 m. The outcome of the framework was that 2172 edges shorter than 2 m were practically removed (8 left), the number of faces was reduced nearly 2.5 times, while the total volume of the buildings was around 2% smaller than the non-simplified one. However, the approach still requires some manual work after the simplification as it can create edges with sharp angles that end up being problematic for mesh generation algorithms.

Park et al. (2020) developed a 5-step simplification algorithm that also incorporates the geometry repair. The basic steps of the algorithm are shown in Figure 2.11. The algorithm works by defining each face of a solid as major and minor according to their area, distance and angle to adjacent faces, and a user-defined threshold. The authors used cfMesh<sup>4</sup> for automatic mesh generation and to compare the change in finite volume mesh quality as a result of the simplification. Observed parameters were the decrease in maximum non-orthogonality, decrease in maximum skewness and the increase in the ratio of hexahedron cells compared to other types. The conclusion was that the simplification algorithm results in a better quality mesh, especially with skewness, as its maximum value decreased exponentially. The disadvantage of the algorithm is that the average execution of the algorithm was 137 s per building on a state-of-the-art workstation. This could possibly be accelerated if a part of the simplification is initially made on floor plans of buildings.

Building walls are generally not smooth; they contain windows, doors, balconies, etc. Blocken et al. (2012) used sand-grain roughness height in the wall function to account for those features, one value for all explicitly modelled buildings. They also imposed aerodynamic roughness length on the terrain, differentiating between streets and green surfaces. Liu et al. (2018); Toparlak et al. (2018) used the same approach in their works.

In the end, the critical remark is that none of the reviewed papers conducted an investigation to ascertain the influence of proposed simplifications on simulation results, i.e. flow field, heat transfer, or pollutant dispersion.

---

<sup>4</sup><https://cfmesh.com/>

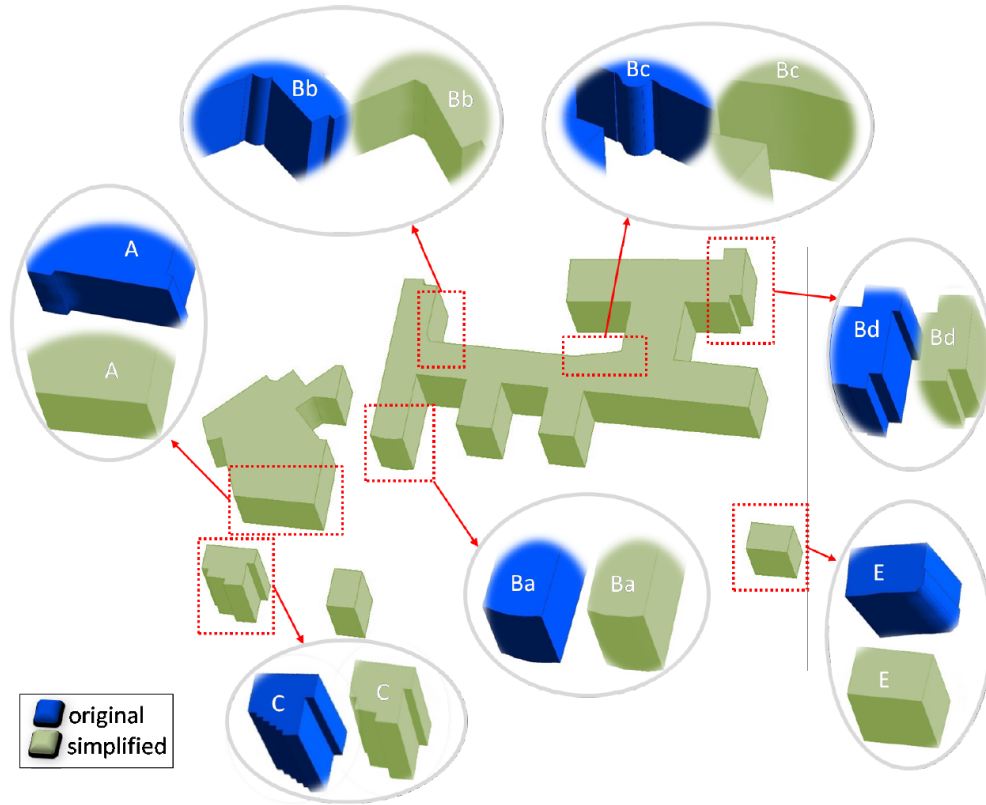


Figure 2.10: The results of a sweep-plane simplification algorithm, from Pieperit et al. (2018).

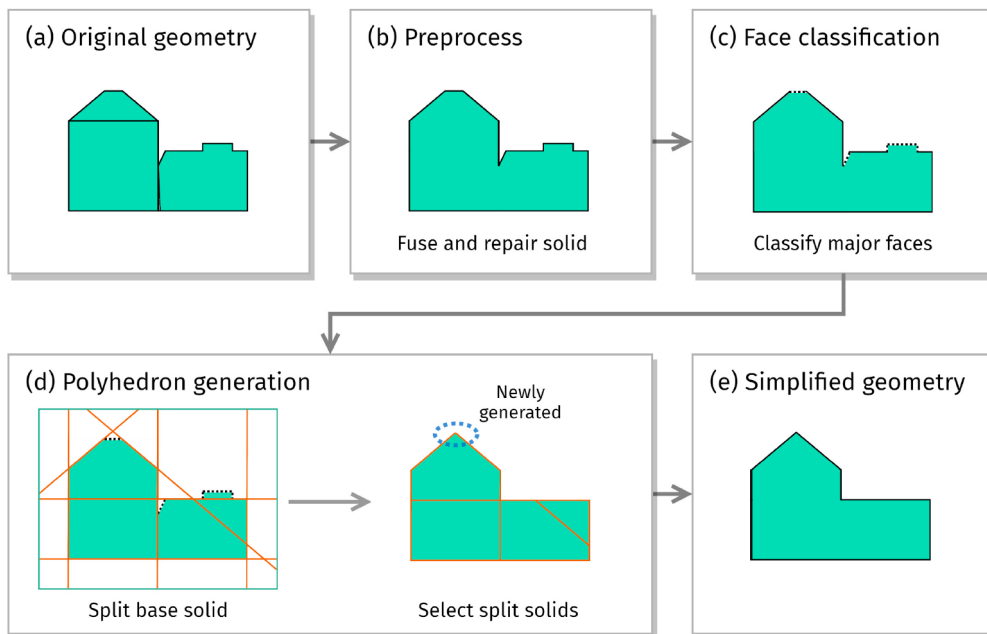


Figure 2.11: Geometry simplification process proposed by Park et al., from Park et al. (2020).

### Simplification outside the influence region

Simplification outside the influence region can generally be divided into three approaches (Liu et al., 2017):

1. Setting the roughness length patches,

2. Treating the attributes as porous media, and
3. Parameterising the drag force created by buildings and inducing it as a source term in the momentum equation.

The first approach seems to be prevalent in the urban flows community. The aerodynamic roughness length can roughly be explained as a height at which the wind velocity is zero (Stull, 1988). Liu et al. (2018) employed areas of different roughnesses between the weather station and investigated district within a city, shown in Figure 2.12. They found a 5.5% difference in the mean wind velocity between the simplified and detailed model, while the total number of finite volumes was 30% less. Different roughnesses account for varying densities of the built environment, as well as the roughnesses of the natural environment such as bodies of water, forests, fields and so on. These different rough patches can be further found in Toparlar et al. (2015); Blocken et al. (2016). The values for aerodynamic roughness lengths are mostly taken from a lookup table defined by Wieringa (1992), which contains eight classes (from the open sea to city centres) with values ranging from 0.0002 m to 2 m. Several other works proposed methods to calculate the aerodynamic roughness length using morphometrical data; Grimmond and Oke (1999) gave the overview of those methods. Typical variables used in those methods are the plan areas of roughness elements (buildings, vegetation and other surface features) and their ratio to the total surface area, the height of roughness elements, the frontal area and density of roughness elements. Since then, researchers have developed frameworks to estimate the aerodynamic roughness length from LiDAR data; they aimed to extract parameters required for the estimation methods mentioned above. Examples of such research can be found in De Vries et al. (2003); Holland et al. (2008); Colin and Faivre (2010); Faivre et al. (2017). I could only find one application of those methods in CFD simulations in the work of Lukač et al. (2017). In this publication, the authors did not use the aerodynamic roughness length estimation to model the area outside the influence zone but rather to obtain the logarithmic wind profile at the domain inlet.

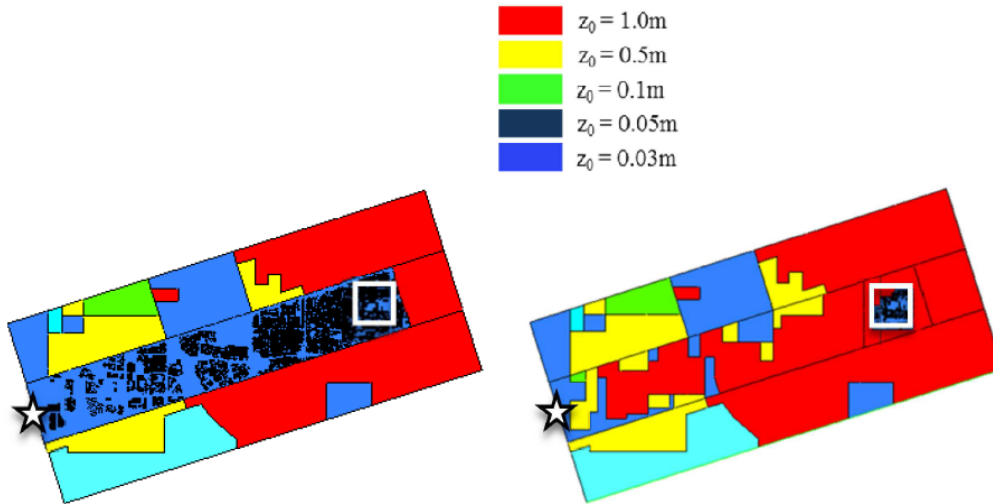


Figure 2.12: Detailed 3D city model (left) and parameterised model with roughness patches (right) between the inlet (star) and the zone of interest (in square), from Liu et al. (2018).

However, there is a concern on the usage of roughness patches if the height of the simplified part of the domain and the explicitly modelled area adjacent to it is similar. The roughness length approach was derived from the Monin-Obukhov similarity theory, which means its applicability encompasses the inertial sublayer above the city-induced roughness sublayer (Basu and Lacser, 2017). One way to alleviate this issue would be to use the roughness sublayer correction (Basu and Lacser, 2017). Still, there should be some average height difference between the explicitly modelled and parameterised buildings. The issue can be avoided by using one of the other two approaches.

The second approach is the porosity parametrisation. Hang and Li (2010) developed a porosity model for buildings which was validated with wind tunnel data of an idealised cluster of buildings. The model was applied by Wang et al. (2021a) on a city block and compared to field measurements. The results showed that the model was able to capture primary ventilation corridors and open spaces of the residential area within the block. However, the formulation is quite complex.

The third approach is the drag force parametrisation. Chan and Leach (2007) developed a drag force parametrisation method they called the *virtual building approach*. In its formulation, this method is similar to the vegetation porosity model explained in the next section (which differs from the porosity approach mentioned above). However, the model coefficients are formulated differently. The model acts as a drag term in the momentum equation with its magnitude so strong it effectively stops the fluid flow through the medium, i.e. the building. Muñoz-Esparza et al. (2020) extended this method for arbitrary grid size and thermal effects; they implemented it into their GPU-LES solver and called it the *immersed body force method*. The method was validated on a single building in Shin et al. (2021) and the Oklahoma City Joint Urban field campaign (Allwine and Flaherty, 2006) in Muñoz-Esparza et al. (2020). The authors reported that the method could capture flow patterns similar to explicitly modelled buildings in both the wind tunnel and field measurements.

### Size of the influence region

It is essential for an accurate simulation to define the size of the explicitly reconstructed region, i.e. the influence region. Van Hooff and Blocken (2010b) explored the most extreme case — they investigated the passageways into a building of interest with the building alone, and including the surrounding buildings; they concluded there are notable differences in wind magnitude whether surrounding buildings are explicitly modelled or not, approximately by a factor of two. Guidelines to solve this issue were presented in Tominaga et al. (2008); Tong et al. (2016); Liu et al. (2018) and are based on one target building. The guideline from Tominaga et al. (2008) is based on the height of the target building. The explicitly modelled region includes two times the height and one additional block of buildings. Tong et al. (2016) suggested including at least three layers of buildings around the target building for regular street canyons. The problem, however, is the definition of layers when surrounding buildings are unequally spaced. The last suggestion by Liu et al. (2018) proposes including three times the largest dimension of the building. Figure 2.13 shows the comparison of these guidelines on a case study.



Figure 2.13: The definition of the influence zone by Tominaga et al. (2008) (left), Tong et al. (2016) (middle) and Liu et al. (2018) (right), from Liu et al. (2018).



## 2.4 Vegetation Modelling

Even though vegetation is often omitted from urban flows, its effect on the wind flow (Hefny Salim et al., 2015), pollutant dispersion (Amorim et al., 2013) and energy balance (Manickathan et al., 2018b) is shown to be meaningful. Away from CFD, there have been developments that offer automatic reconstruction of vegetation at different levels of details. For example, the works of Raunonen et al. (2013) and Du et al. (2019) offer accurate, highly-detailed reconstruction of trees from laser-scanned data (Figure 2.14a). Other works focused to provide a CityGML (Gobeawan et al., 2018) and CityJSON (de Groot, 2020) compliant reconstruction on a large scale (Figure 2.14b).

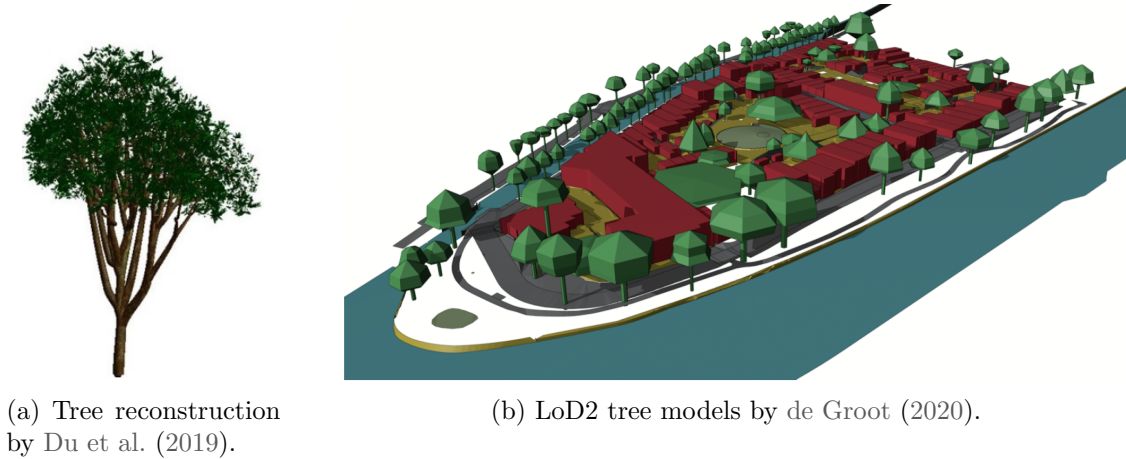


Figure 2.14: Different level of details for tree reconstruction from LiDAR data.

Vegetation modelling in CFD, depicted in Figure 2.15, does not focus on the geometric representation. Creating the finite volume mesh around vegetation would further complicate the already delicate process, and the total number of mesh cells would be hard to justify. A work by Wang et al. (2021b) illustrates this, with a finite volume mesh for one tree that contains millions of cells. For practical purposes, one can use three approaches to handle vegetation: the first is to omit it from the simulation (Figure 2.15b); the second is to use roughness length, the same that has been used for simplification of buildings (Figure 2.15c); the third is the porosity parametrisation (Figure 2.15d). Omitting vegetation adversely affects simulation results, but it is still used even today. Research has shown that the porosity approach is better than the roughness parametrisation (Hefny Salim et al., 2015), and it has also been widely accepted in the literature (Kang et al., 2020).

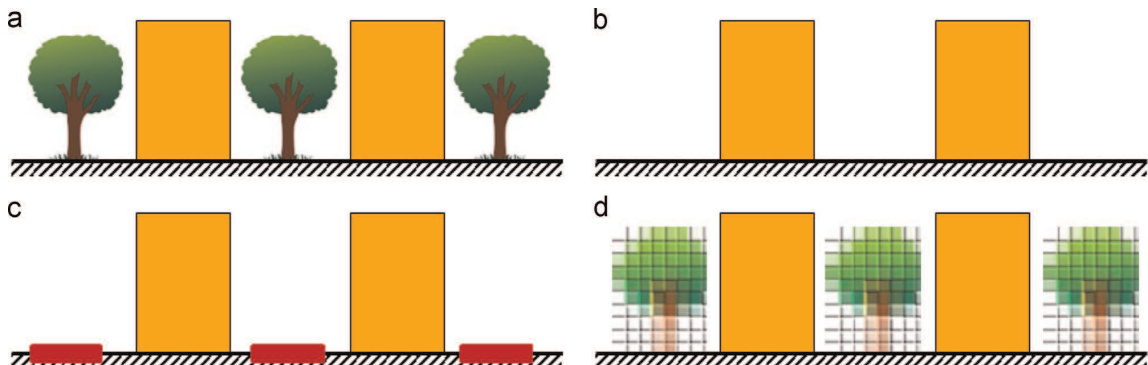


Figure 2.15: Vegetation modelling in CFD: (a) street canyon with trees; (b) basic approach omitting the vegetation; (c) increased roughness; (d) porous zones, from Hefny Salim et al. (2015).

The porosity approach does not aim to model the tree canopy but rather to mark finite volume cells that roughly account for trees. For example, Sousa and Gorlé (2019) used cylinders to define porosity regions. The effect of vegetation is defined as a source and/or sink term in the momentum

equation and equations of turbulence in control volumes that account for porous zones. The most used expressions in the literature, e.g. (Gromke and Blocken, 2015; Moradpour et al., 2017; Santiago et al., 2019), are:

$$\begin{aligned}
S_{u_i} &= -\rho C_d LAD U_i \mathbf{U} \left[ \frac{\text{N}}{\text{m}^3} \right] \\
S_k &= \rho C_d LAD U_i (\beta_p \mathbf{U}^3 - \beta_d \mathbf{U}k) \left[ \frac{\text{W}}{\text{m}^3} \right] \\
S_\epsilon &= \rho C_d LAD U_i \frac{\epsilon}{k} (C_{\epsilon 4} \beta_p \mathbf{U}^3 - C_{\epsilon 5} \beta_d \mathbf{U}k) \left[ \frac{\text{W}}{\text{m}^3} \right],
\end{aligned} \tag{2.2}$$

where the first equation is the sink term for the three momentum equations, the second equation is the source term for the turbulence kinetic energy equation, and the third equation is the source term for the turbulent dissipation rate equation. In the case of pollutant modelling, these can be further expanded with the expressions for deposition and resuspension (Hong et al., 2018). In the equation above,  $\rho$  is the air density,  $\mathbf{U}$  the velocity magnitude,  $k$  the turbulence kinetic energy,  $C_d$  the leaf drag coefficient and  $LAD$  is the leaf area density;  $\beta_p$  and  $\beta_d$  are parameters that define the fraction of the mean kinetic energy that is converted into wake turbulence kinetic energy, and short-circuiting of the energy cascade respectively;  $C_{\epsilon 4}$  and  $C_{\epsilon 5}$  are empirical constants. Overview of proposed calculations for  $\beta_p$ ,  $\beta_d$ ,  $C_{\epsilon 4}$  and  $C_{\epsilon 5}$  can be found in Buccolieri et al. (2018).  $C_d$  is experimentally obtained and dependent on tree species, albeit  $C_d = 0.2$  is a widely accepted approximation for an average value (Gromke and Blocken, 2015; Buccolieri et al., 2018).  $LAD$  ( $\text{m}^2\text{m}^{-3}$ ) is defined as a ratio of the one-sided leaf surface area in a given volume:

$$LAD = \frac{A_{\text{leaf}}}{V}. \tag{2.3}$$

Buccolieri et al. (2018) reports that  $LAD$  in CFD simulation is mostly defined as constant with height, with values ranging from 0.1 to 4  $\text{m}^2\text{m}^{-3}$ . Values of  $LAD$  depend on the type of vegetation and the season, and different values can be found in the literature (Buccolieri et al., 2018). Santiago et al. (2019) made a distinction between three different types of vegetation: hedgegrows and two types of trees based on their size. On the other hand, Simscale (2021) defines  $LAD$  as a constant value, but dependent on the tree height:

$$LAD = \frac{LAI}{h}, \tag{2.4}$$

where  $h$  is the height and  $LAI$  is the leaf area index, the ratio of the leaf area to the ground area. There have also been works that expressed  $LAD$  as a function of height. Liang et al. (2006) obtained vertical  $LAD$  distribution by conducting wind tunnel experiments. Kenjereš and Ter Kuile (2013) used the same distribution to make a case study on the TU Delft campus. Interestingly, their values for  $LAD$  range from 0.5 to 36  $\text{m}^2\text{m}^{-3}$ . Another method of ascertaining height-based  $LAD$  is presented in Shaw and Schumann (1992). The authors defined generalised canopy density curves that depended on cumulative forest  $LAI$  ( $\text{m}^2\text{m}^{-2}$ ), as shown in Figure 2.16. In this case, the  $LAD$ - $LAI$  relation is defined as

$$LAI = \int_z^h LAD(z) dz. \tag{2.5}$$

Von Der Grün et al. (2020) expanded the model by Shaw and Schumann (1992) with more  $LAI$  curves, applied and validated it on an urban case with different vegetation heights. This expanded model was utilized by Deininger et al. (2020) in their considerable effort to automate the vegetation

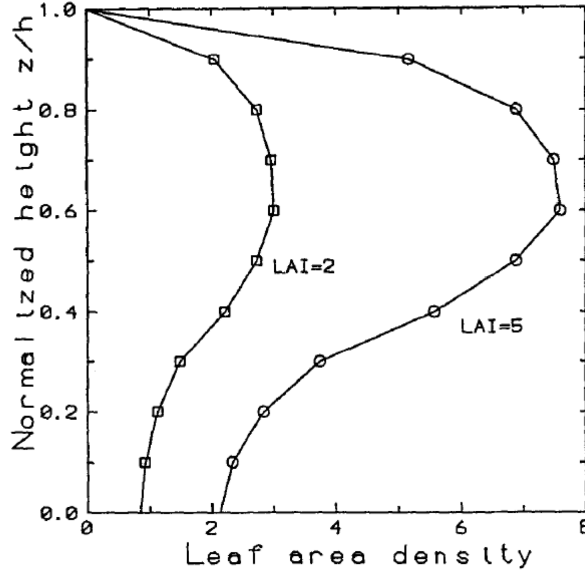


Figure 2.16: Vertical profiles of LAD for two values of LAI, from Shaw and Schumann (1992).

modelling in urban flows. They combined the point cloud with vegetation classification and the tree cadastre data of Stuttgart to create the FME workflow. To the best of my knowledge, this is the only publication that exploits the airborne LiDAR data to automatically model vegetation in numerical simulations. Furthermore, research has been done to quantify LAD and LAI from airborne LiDAR data (Oshio et al., 2015; Kamoske et al., 2019; Carrasco et al., 2019). Those are voxel-based methods that use the information on the number of returns in a specific voxel to reconstruct the LAD profile. The applicability of those methods in CFD simulations is yet to be investigated. Finally, I have not managed to come across any research that compares the effect of different LAD acquiring methods that I just wrote about on the results of numerical simulations.

Manickathan (2019) greatly expanded the modelling of vegetation, implementing an interaction between the soil, plant and atmosphere. However, a detailed model like that is still too demanding for city-scale applications (Manickathan, 2019). Thus, larger scales require simplifications. The same author expresses the need to account for the seasonality of vegetation; deciduous trees can be neglected during the winter season (Manickathan et al., 2018a).

Please bear in mind that some authors refer to the porosity approach as “explicit” modelling. I disagree with the term, as *explicit* would mean that the features of vegetation are geometrically modelled as objects.

## 2.5 Uncertainty Quantification

Even though numerical simulations are mainly presented as deterministic, many parameterisations and simplifications are necessary to make them feasible. This indicates that there should be a measure of confidence attached to them. The role of uncertainty quantification (UQ) is to stipulate the measure of confidence. The uncertainty can be divided into two types (Iaccarino et al., 2009):

- *Aleatory*: physical variability intrinsic to the modelled system,
- *Epistemic*: variability due to lack of knowledge and assumptions introduced in the derivation of the mathematical model.

As its names suggest, aleatory uncertainty, also known as stochastic uncertainty or irreducible uncertainty, can not be reduced. It is characterised by the nature of the observed problem, e.g. material

properties, operating conditions, manufacturing tolerances (Iaccarino et al., 2009). On the other hand, epistemic uncertainty can be reduced with models that offer better predictability, or for example, with better data. An obvious example of epistemic uncertainty reduction would be to replace RANS turbulence modelling with LES; however, that raises the question of whether the reduction in uncertainty justifies the added computational cost, especially for urban flows (García-Sánchez et al., 2014).

“In practical simulations of the ABL, the flow conditions are subjected to the large variability that characterizes the atmospheric environment. This renders the UQ approach particularly interesting for this type of applications, since there is a need for quantifying the uncertainty in the predictions if computational tools are to be used for design or regulatory decisions.” (Gorlé, 2021). From the preceding quote, we can see a strong argument to use UQ in urban flows. Gorlé et al. (2015) created a UQ framework for urban flows that combined previous works on UQ of inflow conditions (García-Sánchez et al., 2014) and RANS turbulence model (Gorlé and Iaccarino, 2013). The uncertain variables for the inflow conditions were determined to be the velocity magnitude and direction at a certain height and the atmospheric roughness height used to calculate the friction velocity. A polynomial chaos expansion response surfaces are generated for each of the uncertain variables by running a large number of simulations with different inflow conditions (729 simulations in García-Sánchez et al. (2014)) and subsequently sampled according to predefined probability density functions (PDF’s). The turbulence model uncertainty is investigated by conducting a baseline RANS simulation and then introducing perturbations to Reynolds stresses. Gorlé et al. (2015) performed 153 additional simulations to evaluate the turbulence model uncertainty. While such a large number of simulations makes the framework impractical for engineering purposes, it serves as a systematic and thorough way to evaluate new developments. This framework was applied for the uncertainty estimation of pollutant dispersion in García-Sánchez et al. (2017).

The framework mentioned above belongs to the *uncertainty propagation*, also known as the *forward analysis* or *data-free* approach to UQ modelling (Xiao and Cinnella, 2019). The disadvantage of the approach is the approximation of PDF’s describing the uncertain variables. Measurements used for inflow conditions are usually obtained from weather stations that are not always near the inlet of the domain. For the inflow conditions of ABL flows in Gorlé et al. (2015); García-Sánchez et al. (2017), this has been approximated with the *beta* distribution and the *beta* distribution with reduced variance by Vervecken et al. (2013). The authors recorded a dependency of the UQ results on the shape of the PDF, mainly the standard deviation; the mean values of the UQ largely remained similar. Solution to this issue could be to employ the second approach to UQ modelling, the *Bayesian inference* (*backwards analysis* or the *data-driven* approach) (Xiao and Cinnella, 2019). This approach can be used when data is available at the output, i.e. where the quantities of interest are. An example of that would be field data measurements from an anemometer. It uses that data to iteratively estimate the PDF’s for the inflow conditions utilizing the data assimilation, i.e. the inverse ensemble Kalman filter (Iglesias et al., 2013). The schematic comparison of the two approaches is shown in Figure 2.17. The data-driven approach has been utilized for urban flows by Sousa et al. (2018). The results were promising for the quantification of inflow conditions; however, the robustness of the data assimilation algorithm depended on the noise in experimental results and the location of sensors. The succeeding validation (Sousa and Gorlé, 2019) showed better predictions with this approach than the uncertainty propagation.

To recap, the developed framework has been used to quantify uncertainty in urban flows arising from inflow conditions and the turbulence model, and has been applied to model pollutant dispersion. To the best of my knowledge, nobody has conducted a systematic investigation on the effect of building LoD within the UQ framework. As reported by Hågbo et al. (2020), the review of urban microclimate studies by Toparlar et al. (2017) included 183 works, albeit not a single one evaluated the influence of geometry inputs on the wind flow. In the succeeding publications, Ricci et al. (2017) made a heuristic comparison between three different LoDs. However, this comparison was done for a limited number of cases and did not include uncertainties associated with the numerical model and inflow

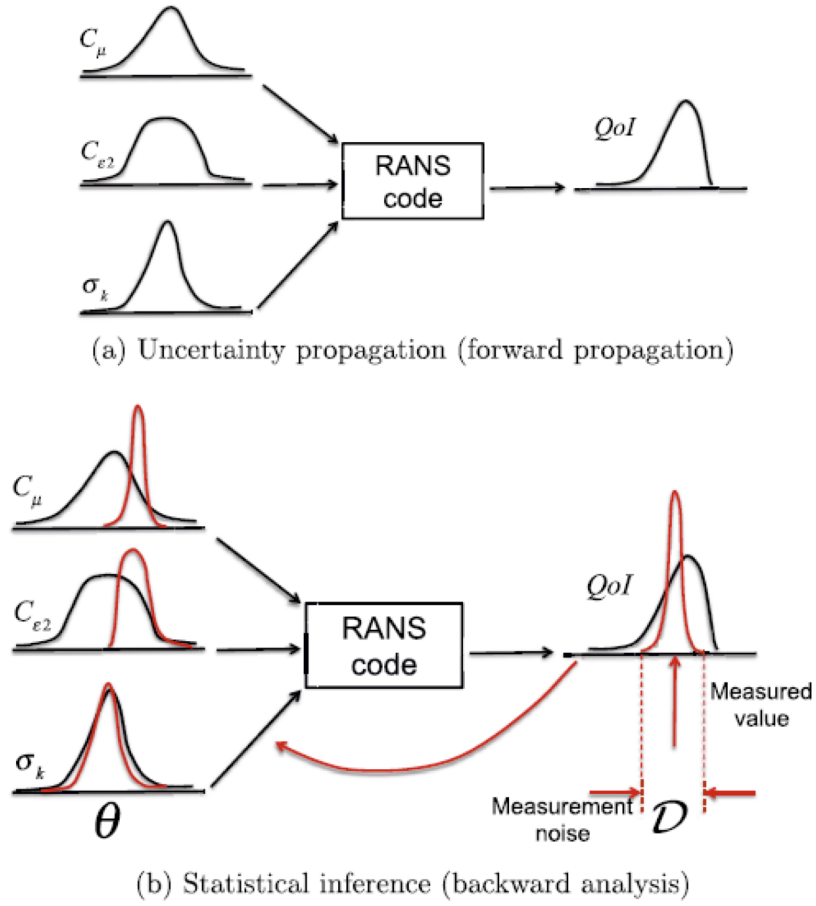


Figure 2.17: Schematic representation of the two approaches to UQ, from Xiao and Cinnella (2019). QoI stands for the quantity of interest.

conditions. That is as far as investigations of LoDs on urban flows have reached. Furthermore, the acquisition-based error manifested as the misalignment between building footprints and a point cloud is an investigated topic in spatial analysis (Biljecki et al., 2018a); however, to the best of my knowledge, the aleatory uncertainty in CFD arising from this error has not been investigated.

The situation with UQ of vegetation modelling is similar, if not worse. As presented in Section 2.4, the expressions used for vegetation modelling contain the drag coefficient and the leaf area density whose values are empirical, to say the least. Again, no efforts have been made to quantify the influence of those variables. The UQ of vegetation is something that should be investigated, and it was also pointed out by others (Šíp and Beneš, 2016).

# Chapter 3

## Proposed PhD Research

### 3.1 Problem Statement

Based on the literature review in Chapter 2, I can conclude there are gaps in research that require attention:

**Problem 1:** *There is still a considerable amount of manual labor involved in preparation of CFD-ready geometries.*

With the increase in computational power in the past decade, the computational wind engineering community has steered towards applying existing methodologies on a larger geometrical scale rather than employing more complex methodologies on the same scale (Blocken, 2018). The already tedious job of manual geometry preparation is becoming one of the main obstructions to the progress of the whole field. Some improvements have been made, and some research managed to attain some level of automation, but it has not nearly been enough. As recently reported by Blocken (2021), around 80% of human time is still spent on geometry preparation and mesh generation. Even in case of improvements, they combine different frameworks and software, some of which are expensive, closed source commercial packages. Geometry creation and geometry preparation are decoupled processes in practically all state-of-the-art applications. There is evidently a strong need to offer application-specific 3D city model reconstruction in CWE. On the other hand, the geomatics field provides open-source frameworks that can reconstruct buildings at a large scale in different LoDs. What they are missing are adjustments to make them entirely suitable for CWE.

**Problem 2:** *The CWE field has not utilised the full potential of semantic data in 3D city models.*

Semantic data plays an important role in 3D city models. The most meaningful semantic data for CWE would be roughness and porosity parameters. However, nobody proposed a data structure, issued guidelines, or gave detailed information on how they handled the semantics. By reviewing the up-to-date research, I concluded that many city details are left out because there are no standards or accepted practices when it comes to storing data for such large-scale simulations. That is why we often see one roughness value for all buildings or one porosity value for all trees in the domain. Problem 1 is another factor contributing to this problem, as automatic extraction and handling of semantic data is not an active research topic in CWE.

**Problem 3:** *Vegetation modelling in CWE is handled rather arbitrarily.*

CWE has universally accepted guidelines, most of them dealing with meshing requirements and domain size, choice of turbulence models and numerical schemes. However, vegetation modelling is still an uncharted territory. The porosity approach has gained the most support over the past years, but the coefficients used in porosity formulation are typically defined heuristically. This problem is also connected to problems 1 and 2. Information on vegetation is not stored as a part of the semantic 3D

model even though it contains parameters that are worth storing. Furthermore, automatic vegetation modelling is another problematic area in CWE. Research dealing with this issue is scarce, and nobody presented a universal solution thus far. It is, in my opinion, one of the main contributors to why vegetation omitting is an acceptable practice even today. On the other hand, geomatics and remote sensing fields contain solutions that have the potential to facilitate CWE efforts with some adjustments.

**Problem 4:** *There are no uncertainty quantification studies that deal with geometry uncertainties in urban areas.*

The literature review confirms that nobody has systematically investigated the influence of building LoDs on urban flow prediction uncertainties. It is unknown how different LoDs impact the wind flow or pollutant dispersion. There are no guidelines and no recommendations. Furthermore, the aleatory uncertainty arising from building footprints and point cloud misalignment is currently unknown. Also, there have not been any studies that quantified uncertainties related to vegetation parameters that are often chosen heuristically. It is essential to get an insight into uncertainty as the results could increase accuracy and lower computational demands. One could only speculate the reasons why nobody conducted a similar study. It could be that the previous three problems I emphasised are all agglomerated within the last problem. If obtaining the CFD-ready city geometry was not as big of an issue, comparing different representations of the same geometry would not be as problematic.

## 3.2 Research Objective and Questions

Based on the identified gap, and guided by the preliminary research, I have identified the main research objective to be the following:

*Develop a framework that automatically reconstructs semantic three-dimensional urban environment models optimised for CFD at different levels-of-detail.*

I plan to reach the research objective using the the following guide questions:

- Can the geometry preparation step of urban flow simulations be fully automated?
  - What are the main bottlenecks in the automatic generation of 3D city models for CFD applications?
  - Is it possible to create error-free geometries suitable for CFD applications?
- How can state-of-the-art 3D city models be modified and expanded to suit the requirements of CFD simulations better?
  - Which data useful for CFD is missing from the international standards for 3D city models? How can that data be stored and later used by CFD software?
- How can the vegetation data from point cloud be quantified and utilised in numerical simulations?
  - What is the acceptable way to store CFD-ready vegetation in a semantic 3D city model?
- What is the uncertainty introduced by geometry?
  - What is the epistemic uncertainty arising from different building LoDs?
  - What is the aleatory uncertainty caused by misalignment of footprints and point cloud data?
- What is the uncertainty caused by vegetation modelling?
- Can a single LoD specification be recommended for different types of urban flow simulations such as wind flow and pollutant dispersion?

### 3.3 Methodology

Methodology of this research can be split into two integral parts:

1. Automatic 3D city geometry preparation, and
2. Uncertainty quantification.

The first part focuses on algorithms and methods that are beneficial for creating error-free CFD-ready buildings and the digital terrain model (DTM). It also presents a plan to calculate and store semantic data used for CWE simulations. Figure 3.1 summarises the methodology related to the first part. The second part outlines methods that will be used for uncertainty quantification and the resulting verification and validation.

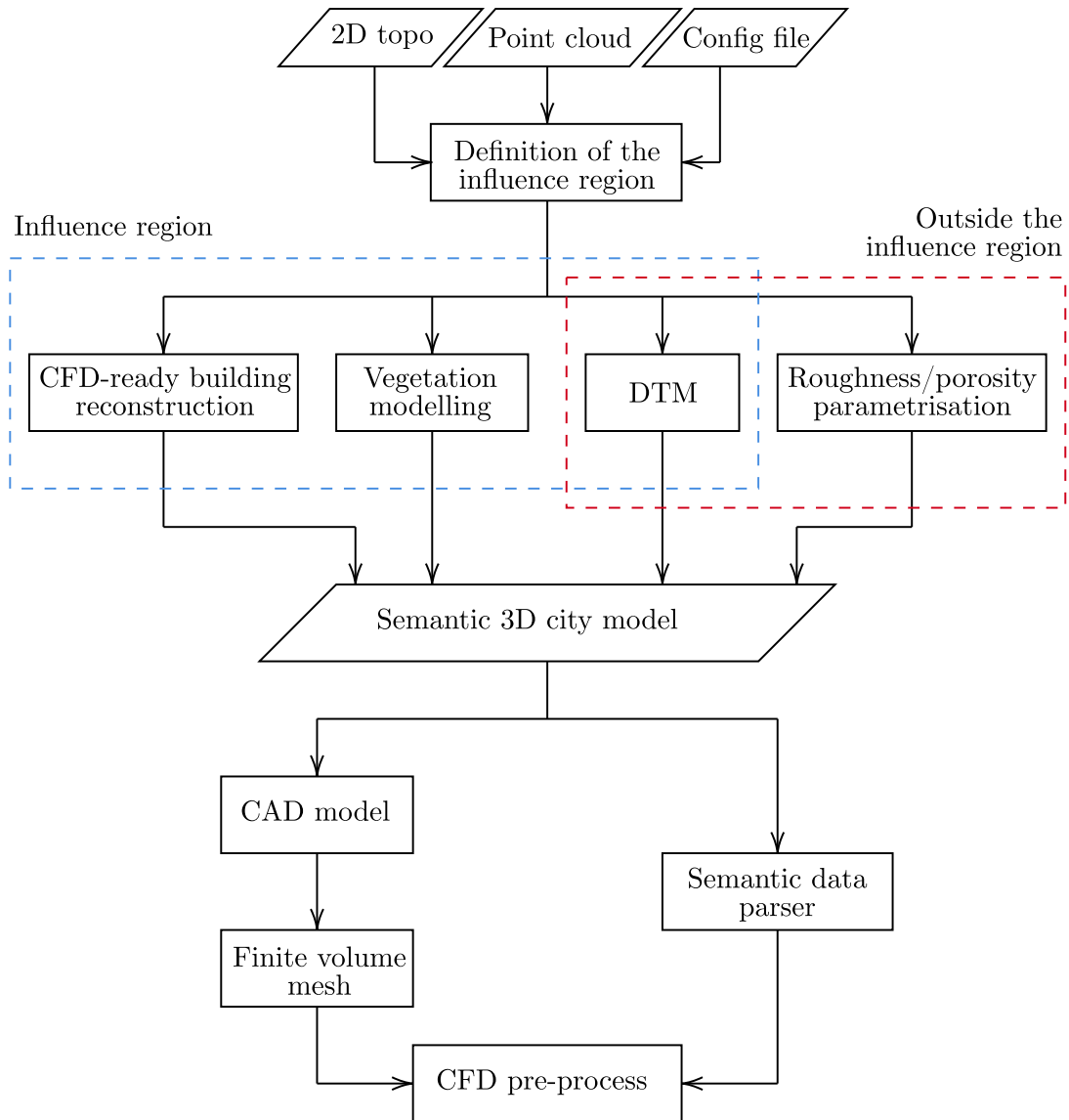


Figure 3.1: Geometry preparation workflow.

The workflow starts by defining the domain boundaries and the influence region. With the influence region defined, the following algorithms are being invoked: the creation of the digital terrain model (DTM) (Section 3.3.3), CFD-ready building reconstruction (Section 3.3.2), vegetation modelling (Section 3.3.5), and modelling of the area outside the influence region (Section 3.3.4). The



outputs of algorithms are all saved in a semantic 3D city model that uses CityJSON encoding as the format. The CAD model containing only explicit features is then extracted from the CityJSON model and provided to the finite volume mesh generator. Semantic data from the CityJSON model are then attached to the newly created mesh in the CFD software pre-processor (presented in Section 3.3.6). To make this step feasible, it is necessary to develop a CityJSON parser for an individual CFD software. I will provide the plan and goals regarding the respective parts in the continuation of this section.

### 3.3.1 Input Data and Preparation

The goal in the choice of the input data is to use the ones that are most commonly available. This is why the framework will accept two types of data:

- The combination of 2D topological data and a point cloud,
- 2D topologically consistent data with height information.

The second type of input data is just a simplified version of the first type; it alters the proposed workflow. First, the second type of data input can only provide footprint extrusion, meaning the buildings cannot be reconstructed in LoD higher than LoD1.2. Second, vegetation will be modelled using the information from a point cloud; therefore, it is not possible to model vegetation. Finally, DTM has to be handled differently in case there is no point cloud data to model it.

The configuration file will be in a JSON format and contain all necessary parameters for a setup.

The first task of the workflow will be to define the influence region or the area of interest, as the methods are split between it and the rest of the domain, with the creation of DTM encompassing both. The configuration file will contain the middle point of the influence region and the radius of the influence zone. All buildings within or on the border of the area covered with radius will be included. It is also possible to automatically determine the influence region in case one particular building is being investigated. In that event, the guideline on the influence region by Liu et al. (2018), explained in the literature review and shown in Figure 2.13, will be used. The algorithm will use the elevation data to calculate the height of the building closest to the coordinate and compare it to footprint edges. Three times the largest dimension will be defined as the radius.

Domain boundaries will be circular with dimensions adhering to the best practice guidelines (Franke et al., 2007; Tominaga et al., 2008; Blocken, 2015). Figure 3.2 schematically shows a round domain that will be used, excluding the DTM and vegetation. The green building is the building of interest; three times the largest dimension (height) is defined as the influence region, with blue buildings being within or on the border of the influence region and thus explicitly reconstructed. Buildings outside, coloured red, are modelled with the drag force parametrisation approach, presented in Section 3.3.4. The  $15H_{\max}$  distance is suggested in the best practice guidelines. These types of domain where one boundary is simultaneously used for inflow and outflow, depending on the wind direction, have already been used for urban flows in Juretić et al. (2017); Hågbo et al. (2020).

### 3.3.2 CFD-Ready Building Reconstruction

Generating error-free CFD-ready buildings is the greatest challenge of this proposal. For CFD, it is important that buildings are detailed but without “small features” such as chimneys and short edges. It is also important that geometries are valid, meaning 2-manifold, watertight and without self-intersections. I present here the methodology that is aimed to accomplish that. Figure 3.3 shows a reconstruction workflow for a single building.

The first part is the footprint preparation, where invalid footprint polygons are repaired and the shortest edges are removed. Then the workflow proceeds to conduct the building reconstruction at the specified LoD, be that LoD2.2, LoD1.3 or LoD1.2. In case the reconstruction algorithm is unable to provide a valid geometry, the workflow proceeds with the reconstruction at a lower LoD, noting down

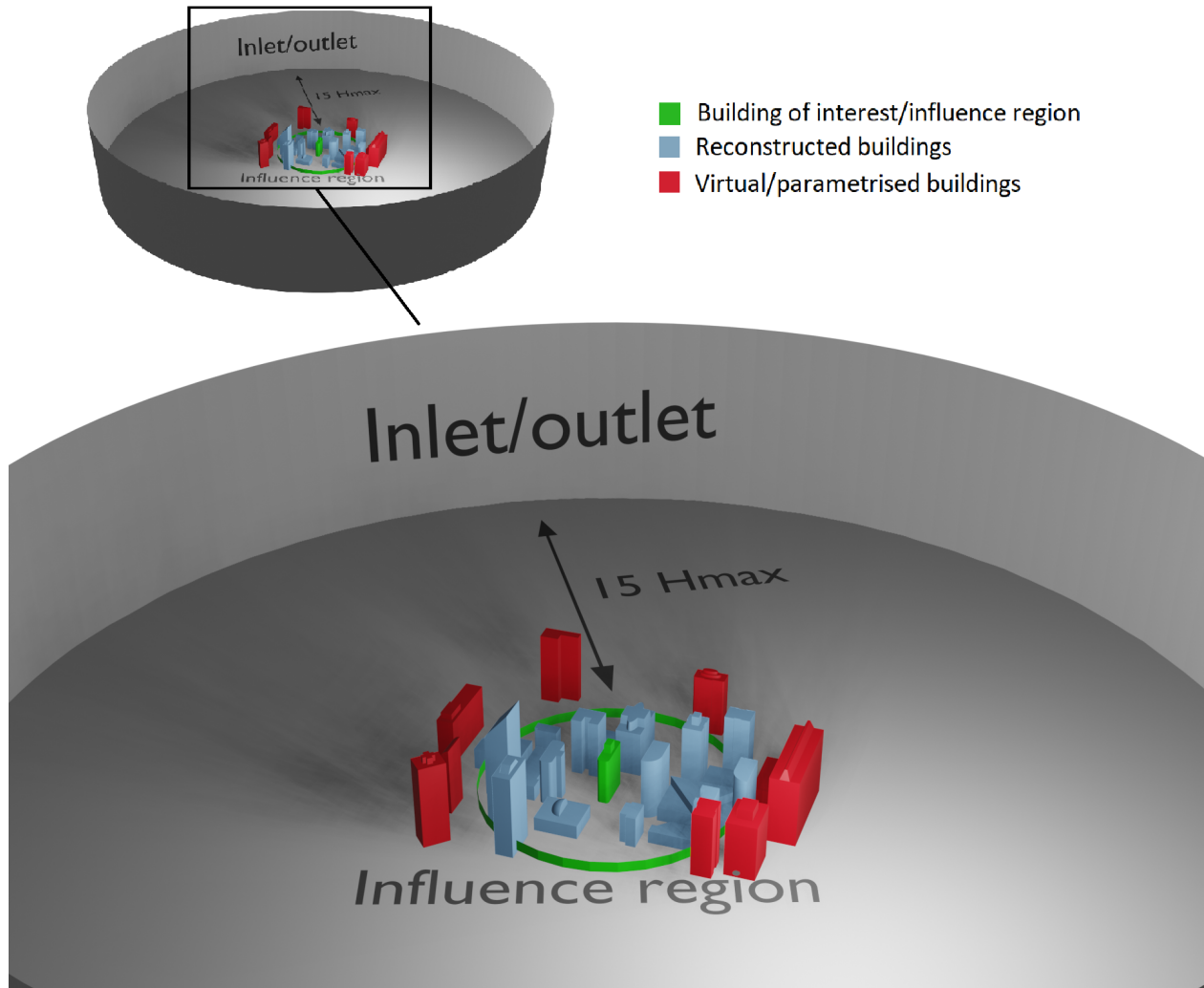


Figure 3.2: Schematic representation of the domain.

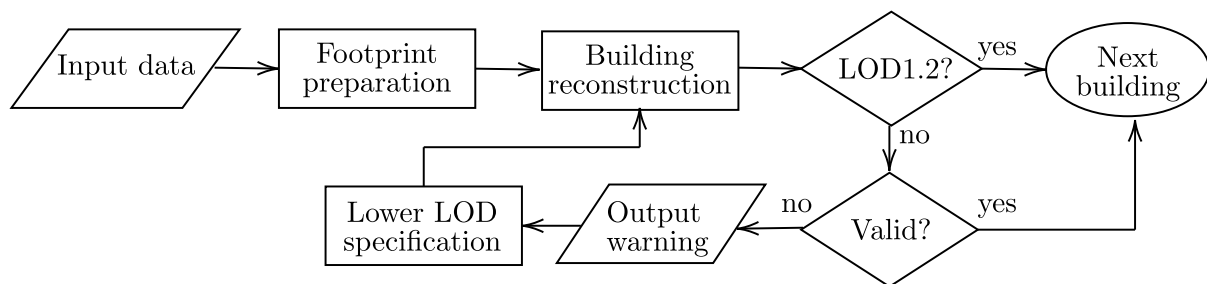


Figure 3.3: Building reconstruction workflow.

the buildings that failed the validity check. In case the higher LoD decomposition algorithm cannot produce a valid geometry, the final reconstruction is done with a simple extrusion. As Labetski et al. (2017) stated, it is better to have an error-free model at a lower LoD, rather than a highly detailed model of questionable quality.

## Footprint preparation

Prepared 2D polygons are a precondition for successful CFD-ready building reconstruction. Preparation includes polygon generalisation and subsequent validation and repair. The work of Commandeur (2012) resulted in a 2D footprint generalisation algorithm. It is an adaptation of the 3D generalisation method developed by Kada (2008). The method decomposes the space along the major planes of the buildings and is controlled by the minimum size of the building elements generated. The intuitive distance threshold value controls which vertices are kept and which are removed. Several other methods aim to do the same. I plan to investigate the suitability of generalisation algorithms. It is a balance between the fidelity of geometry, the resulting number of finite volume cells and the attainable quality of the finite volume mesh.

Validation and repair refer to obtaining valid polygons according to ISO 19107 (ISO, 2003). It benefits the reconstruction by handling ambiguities, such as open polygon rings and self-intersections, that could cause problems in succeeding steps. 2D polygon repair is considered a solvable problem in geomatics, and multiple methods proposed solutions for it. Not to mention that it is both easier and faster to work on 2D polygons rather than 3D geometry. My plan is to use the constrained triangulation (CT) based polygon repair method by Arroyo Ohori et al. (2012). The method preserves topology and guarantees valid polygons after the repair.

## Building reconstruction

As already stated in the literature review, the 3D geoinformation research group has expertise in the development of building reconstruction algorithms. I plan to assess the applicability of those algorithms for CFD simulations and implement one of them as a part of the workflow. Some of the algorithms offer the removal of small objects as part of the optimisation process. The small objects tend to ruin mesh quality or drastically increase the mesh size without noticeable benefits. This small object removal can be considered as the 3D simplification step. Both the methods used in 3D BAG (Dukai et al., 2021) and (Nan and Wonka, 2017) offer this feature. The former method will be the first whose applicability in numerical simulations I will test due to proven scalability and ease of use. The method was introduced in Section 2.2.4 and available in more detail in (Arroyo Ohori et al., 2021). By manipulating the smoothness operator, the graph-cut optimisation algorithm, implemented in the method, can remove small features such as chimneys, potentially aiding the mesh quality.

LoD1.2 is the lowest LoD of the workflow and relies on the preceding footprint generalisation with eventual 2D repairs. With valid 2D polygons, the simple footprint extrusion will result in a valid dataset. It is the only possible way to generate a 3D city model in case of 2D polygons with elevation data and without a point cloud. With a point cloud, it can be easily calculated by conducting operations on the elevation data that falls within the limits of polygons.

## Validity check

Three basic requirements are a prerequisite for a good geometry: 2-manifold, no self-intersections and watertight. All three of them are part of ISO 19107 specifications. While some automatic mesh generators are able to work around the requirements, it is almost certain that the invalid geometry will make the mesh quality lower. Therefore, it is necessary for an automated workflow to create a valid geometry. I plan to use *val3dity* (Ledoux, 2018) to check whether building reconstruction has been done correctly. The reconstruction method of 3D BAG integrates *val3dity* into the algorithm. Even though this algorithm aims for valid geometries, there are still around 15% invalid buildings measured over the whole territory of the Netherlands. This renders the validation step necessary, as well as the feedback loop that lowers the LoD of invalid buildings.

## Contingency plan

In case of repeating invalid reconstruction or unsatisfactory reconstruction quality, the contingency plan is to refer to other footprint repair and CFD-specific simplification methods mentioned in the literature review. I will also closely follow the ongoing research in the group. Labetski (2017) is working on error-free generalisation and Huang (2020) is improving an LoD2 automatic reconstruction algorithm of Nan and Wonka (2017). Worst case scenario, I can fall back on LoD1.2 reconstruction, i.e. the footprint extrusion.

### 3.3.3 Digital Terrain Model

Terrain representation and storing is a well-covered topic in geomatics (Ledoux et al., 2020). DTM is the only feature in the workflow that is modelled the same in the whole domain. For CWE, the terrain should be smooth, as abrupt jumps on the surface are prone to cause a bad quality finite volume mesh. CGAL library<sup>1</sup> offers a robust implementation of most required algorithms for terrain modelling, including triangulated irregular network (TIN) for the initial terrain and Gaussian filters for smoothing of the TIN (Giraudot, 2021). To ensure that the terrain is well connected to building footprints, I will use the constrained Delaunay triangulation with building edges as constraints. The average height of the terrain at the location of constrained points serves as the ground height of buildings.

Terrain surfaces, such as roads and lawns, can have roughness values attached to them; instead of modelling those entities as respective city objects in the CityJSON model, it is enough to semantically enrich parts of the terrain, denoting the entities with one value.

### 3.3.4 Outside the Influence Region

The area outside the influence region (or the area of interest) will have two ways of parametrisation. The first way is to use the virtual building or the immersed body force approach by Chan and Leach (2007), which was introduced in Section 2.3.2. The method uses the drag force that effectively stops the flow in cells that would otherwise be occupied with a building geometry. The expression used by this approach is

$$F_i = -C_d \rho U_i \mathbf{U}, \quad (3.1)$$

where  $C_d$  is the drag coefficient ( $\text{m}^{-1}$ ). Chan and Leach (2007) found that the magnitude of  $C_d$  affects the flow field near buildings and concluded that the fixed value of  $100 \text{ m}^{-1}$  shows best performance. The problem with the fixed value is that Equation 3.1 was derived from Equation 2.2 for canopy flow, so the term  $C_d$  is actually

$$C_d = C'_d A_p, \quad (3.2)$$

with  $C'_d$  (-) denoting the canopy drag and  $A_p$  the leaf area density (LAI) (unit  $\text{m}^{-1}$ ) in case of vegetation modelling. Muñoz-Esparza et al. (2020) observed dependence of fixed  $C_d$  on different grid sizes and proposed a scale-independent formulation:

$$C_d = C_1 \alpha_s \Delta^{-1}, \quad (3.3)$$

where  $C_1$  is the coefficient,  $\alpha_s$  is the volume fraction of cell occupied with the solid (1 for solid, 0 for fluid and  $0 < \alpha_s < 1$  for border cells), and  $\Delta$  is the nominal grid size ( $= \sqrt{\Delta x \Delta y \Delta z}$ ). The validation study showed  $C_1$  of  $10^3$  provides best results.

---

<sup>1</sup><https://www.cgal.org/>

The formulation by Muñoz-Esparza et al. (2020) has not been applied in conjunction with body-fitted mesh nor implemented for RANS simulations, which would make me the first one to utilise this approach in such applications. While the formulation is convenient to use and provides satisfactory results, it has to be implemented in CFD software, which requires advanced user knowledge and separate implementations for individual packages. I will nevertheless implement and test the approach due to its potential. From the geometry standpoint, explicitly reconstructed buildings are necessary; investigations need to be conducted whether LoD1.2 is enough for this simplification method.

The other feasible approach is to conduct the roughness parametrisation. Using information gathered from building footprints and height, one can estimate aerodynamic roughness length with expressions by Macdonald et al. (1998):

$$\begin{aligned} z_0 &= z \left(1 - \frac{z_d}{\bar{z}}\right) \exp \left[0.5 \frac{1.2}{\kappa^2} \left(1 - \frac{z_d}{\bar{z}}\right) \lambda_f\right]^{0.5} \quad [\text{m}] \\ z_d &= \bar{z} \left[1 + 4.43^{-\lambda_p} (\lambda_p - 1)\right] \quad [\text{m}], \end{aligned} \quad (3.4)$$

where  $\lambda_p = A_p/A_T$  denotes the ratio between the building footprint area  $A_p$  and the total ground area  $A_T$ ,  $\lambda_f = A_f/A_T$  is the ratio of the building frontal area  $A_f$  and  $A_T$ ,  $\kappa$  is the von Karman constant (typically 0.4) and  $\bar{z}$  is the average height of the whole area. The approach rasterises the domain and calculates  $z_0$  for every individual raster. More details on the implementation of this method for CFD is available in Lukač et al. (2017).

The roughness approach can be further simplified by comparing object heights and density to tabular values (Wieringa, 1992) and deriving a heuristic relation. Some CFD software use sand-grain roughness rather than aerodynamic roughness length. In that case, the two parameters can be correlated with the formulation by Blocken et al. (2007):

$$k_{\text{S,ABL}} = \frac{9.793 z_0}{C_S}, \quad (3.5)$$

where  $C_S$  is the roughness constant.

In any case, it is possible to conduct roughness parametrisation without modifying state-of-the-art CFD solvers.

### 3.3.5 Automatic Vegetation Modelling

It is evident from the literature review (Section 2.4) that modelling the trees as a porous medium (Equation 2.2) is an established approach. Since trees are a part of the CityJSON specification as the *SolitaryVegetationObject* or *PlantCover*, the existing city objects can be semantically enriched with the two parameters of the tree porosity model - LAD/LAI and the drag coefficient.

De Groot (2020) developed a workflow that automatically reconstructs trees from airborne LiDAR at different LoDs. The workflow classifies, segments, cleans and finally reconstructs the geometry of trees from a point cloud. The LoD1 (according to classification proposed by the author) reconstruction creates hexagonal shapes (Figure 3.4) that can be defined as porous zones in the CFD pre-processor. There is an active research that deals with vegetation classification and segmentation from point clouds. I will closely follow new developments, as the outcomes could be relevant for my research.

While the drag coefficient is almost exclusively defined as a constant value in the literature, the LAD is usually taken as a tabular value or integrated from an empirical profile. I will investigate the prospect of using open-source LAD estimation methods from point clouds, mentioned in Section 2.4. Worst-case scenario, the LAD can be defined through the configuration file.

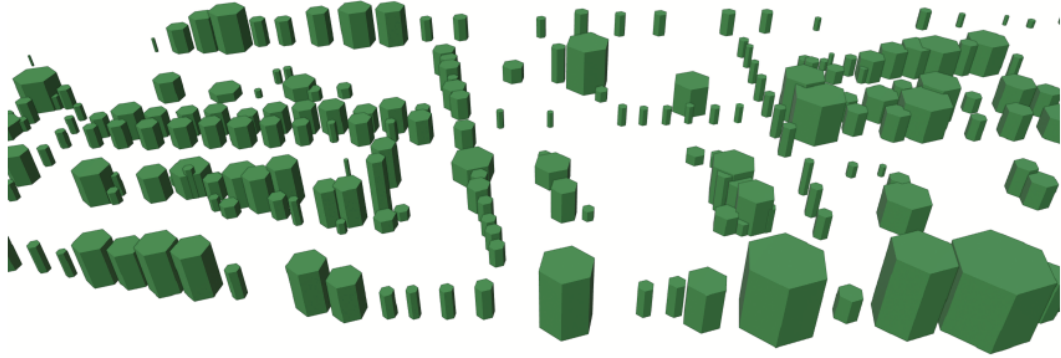


Figure 3.4: LoD1 trees reconstruction, from de Groot (2020).

Lower vegetation, such as hedges and lawns, are represented with the aerodynamic roughness length attached as a semantic information to the DTM, as explained in the *Digital Terrain Model* section above.

### 3.3.6 Data Parsing From 3D Models to CFD Software

To answer the question of why a parser is a necessity, analysing steps that follow after the creation of a semantic 3D city model is essential. First, created geometry undergoes the meshing process that results in a finite volume mesh. Meshing software typically accept few of the many “standard” CAD formats<sup>2</sup>, but Steleolitography (STL) and Wavefront OBJ seem to be universally accepted, as indicated by frequently used mesh generators (snappyHexMesh<sup>3</sup>, Pointwise<sup>4</sup>, cfMesh, etc.); obviously, the input 3D city geometry also has to be in one of those formats. The resulting finite volume mesh is generally exported in a proprietary format of CFD software. Most commercial packages have integrated mesh generators; for instance, ANSYS<sup>5</sup>, STAR-CCM+<sup>6</sup>, and AVL FIRE<sup>7</sup> have their own meshing utilities, whereas Pointwise is an independent mesh generator.

The issue that has to be tackled is introducing semantic data from a 3D city model to the simulation setup. Formats mentioned in this section are semantically poor; in the case of OBJ, storing semantic data is possible but limited in scope (Biljecki and Arroyo Ohori, 2015). Finite volume mesh cannot store semantic data, only topological information and designation of boundary patches — it is only the simulation preparation step *in the CFD software* where attributes are added to the mesh. This is why, as shown in Figure 3.1, the semantic 3D city model has to be segregated into a CAD model that undergoes the meshing process and the parsing part that attaches semantic information to the finite volume mesh in the CFD software.

While the semantic data used for CWE (roughnesses, porosity) are mostly universal for different CFD software, their implementations in individual CFD software are not; consequently, a different parser is required for every CFD solver. I will be using OpenFOAM (Weller et al., 1998), an open-source general-purpose CFD library, to conduct all of my simulations. Therefore, I will make a CityJSON-OpenFOAM parser and thoroughly document all the steps to make potential parsing to other software easier.

<sup>2</sup><https://transmagic.com/cad-formats/>

<sup>3</sup><https://cfd.direct/openfoam/user-guide/v6-snappyhexmesh/>

<sup>4</sup><https://www.pointwise.com/>

<sup>5</sup><https://www.ansys.com/products/meshing>

<sup>6</sup><https://www.plm.automation.siemens.com/global/en/products/simcenter/STAR-CCM.html>

<sup>7</sup><https://www.avl.com/fire-m>

### 3.3.7 Computational Fluid Dynamics

I will use incompressible, steady-state Reynolds-averaged Navier-Stokes equations with neutrally stratified ABL flow for my simulations. The mass (continuity) and linear momentum conservation equations are:

$$\frac{\partial \bar{u}_j}{\partial x_j} = 0, \quad (3.6)$$

$$\bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} - \frac{\partial \overline{u'_i u'_j}}{\partial x_j} + F_i, \quad (3.7)$$

where  $\bar{u}_i$  denotes the time-averaged velocity components,  $\rho$  the density,  $\bar{p}$  the pressure,  $\nu$  the kinematic viscosity,  $\overline{u'_i u'_j}$  the specific Reynolds stress tensor, and  $F_i$  the source or sink term. The source/sink term is equal to Equation 2.2 in case of tree modelling, and to Equation 3.1 in case of the virtual building/immersed body force approach; otherwise, it is zero.

For turbulence closure, I plan to use the  $k$ - $\epsilon$  model; it is well-established turbulence model in urban flows (Blocken, 2018; Ricci et al., 2020b). The model is based on the linear eddy viscosity hypothesis:

$$\overline{u'_i u'_j} = \frac{2}{3} k \delta_{ij} - 2\nu_t S_{ij}, \quad (3.8)$$

where  $k$  is the turbulence kinetic energy,  $\nu_t$  the turbulent viscosity and  $S_{ij}$  the time-averaged shear stress tensor. The turbulent viscosity is a property of a flow and defined as

$$\nu_t = C_\mu \frac{k^2}{\epsilon}, \quad (3.9)$$

with  $\epsilon$  being the turbulence dissipation rate and  $C_\mu$  a model constant equal to 0.09. The turbulence kinetic energy and dissipation rate are calculated from their transport equations:

$$u_j \frac{\partial k}{\partial x_j} = \frac{\partial}{\partial x_j} \left[ \left( \nu + \frac{\nu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right] + P_k - \epsilon, \quad (3.10)$$

$$u_j \frac{\partial \epsilon}{\partial x_j} = \frac{\partial}{\partial x_j} \left[ \left( \nu + \frac{\nu_t}{\sigma_\epsilon} \right) \frac{\partial \epsilon}{\partial x_j} \right] + C_{\epsilon 1} \frac{\epsilon}{k} P_k - C_{\epsilon 2} \frac{\epsilon^2}{k}, \quad (3.11)$$

where  $P_k$  is the turbulent production term and  $\sigma_k$ ,  $\sigma_\epsilon$ ,  $C_{\epsilon 1}$ , and  $C_{\epsilon 2}$  model constants, with values of 1.0, 1.3, 1.44, and 1.92, respectively. I will use UQ (see the next section) to account for the uncertainties arising from the use of a RANS model.

The inflow boundary condition will be modelled as a fully-developed neutral boundary condition with the following profiles for the velocity, turbulence kinetic energy, and dissipation (Richards and Hoxey, 1993):

$$u = \frac{u_*}{\kappa} \ln \left( \frac{z + z_0}{z_0} \right), \quad (3.12)$$

$$k = \frac{u_*^2}{\sqrt{C_\mu}}, \quad (3.13)$$

$$\epsilon = \frac{u_*^3}{\kappa (z + z_0)}, \quad (3.14)$$

with  $u_*$  being the friction velocity,  $z_0$  the aerodynamic roughness length and  $\kappa$  the von Karman constant with a value of 0.41.

### 3.3.8 Uncertainty Quantification

Geometry carries both epistemic and aleatory uncertainties; epistemic uncertainties come from the choice of LoD, whereas aleatory uncertainties come from errors in datasets. On the other hand, vegetation modelled with the porosity approach contains uncertainties due to the choice of porosity parameters. The answer on how these uncertain variables affect simulation results gives an insightful look into the reliability of employed methods. I plan to use the UQ framework by Gorlé et al. (2015) and RANS approach to carry out my research. Their UQ framework couples OpenFOAM with Dakota<sup>8</sup> to conduct necessary investigations.

It is hard at this moment to pinpoint the exact uncertainty propagation algorithms that I will use. What I can do is define uncertain variables and their intervals. This is presented in the following two sections for geometry and vegetation, respectively. Those uncertain variables will be propagated along with already established UQ methods for inflow conditions and turbulence model (Gorlé et al., 2015) to get the total uncertainty estimation, but also to investigate the influence of individual variables using Sobol indices (García-Sánchez et al., 2017). The results should give a final answer on the effect of LoD on simulation results, as well as help to devise guidelines regarding geometry and vegetation modelling.

#### Geometry uncertainty for Buildings

Biljecki et al. (2018a) explored LoD and acquisition-based errors for the error analysis of: 1) area of the building envelope, 2) gross volume of a building, and 3) solar irradiation of rooftops. The LoD-based error arises from the choice of reconstruction algorithm, whereas the acquisition-based error manifests as the magnitude of a building positional error. My goal is to conduct UQ in CFD with the same uncertain parameters.

Uncertainty arising from errors inherited in point clouds or building footprints is aleatory, and it cannot be improved from the standpoint of CFD. Nevertheless, the influence of acquisition-based errors on simulation results is unknown. Biljecki et al. (2018a) used normal distribution to conduct perturbations of building vertices while keeping the same angles; the same approach can be used as an input for CFD UQ. Alternatively, errors between building footprints and the point cloud can be measured on an individual dataset, and a distribution function can be derived from captured data. Distributions are then used to conduct deterministic simulations and calculate the influence of uncertain variables on quantities of interest, such as wind magnitude and direction, or pollutant concentration.

The second part of geometry UQ is related to the representation-induced error, i.e. the choice of LoD. I plan to add three representations — LoD1.2, LoD1.3, and LoD2.2 to the geometry UQ variables proposed above. UQ findings can serve as a guide and decision-maker in automatic geometry preparation, presented in Section 3.3.2. It will also help to determine which LoDs are recommended to use for wind flow and pollutant dispersion, respectively.

#### Vegetation uncertainty

The vegetation model proposed in Section 3.3.5 contains two parameters — drag coefficient  $C_d$  and LAD.  $C_d$  is determined rather heuristically as a modelling assumption, with the exception of few case-based calibrations such as Manickathan (2019). Buccolieri et al. (2018) reported typical drag values from 0.1 to 0.3  $\text{m}^{-1}$ , with most studies using values equal or close to 0.2  $\text{m}^{-1}$ . Wilson and Shaw (1977) and Vogel (1989) defined this interval to be  $C_d \in [0.2, 0.5]$ , while Manickathan (2019) observed from his sensitivity analysis that even  $C_d = 0.6 \text{ m}^{-1}$  can produce best result. Evidently, there is an interval  $C_d \in [0.1, 0.6]$  where drag coefficients can be chosen from. Having a clear interval but no known value to use makes the choice of  $C_d$  an epistemic uncertainty.

---

<sup>8</sup><https://dakota.sandia.gov/>



LAD depends on the tree type, canopy size and the season for deciduous trees. They are either empirically determined and available as a tabular value or a profile or measured, for example, using a point cloud. Empirical profiles change depending on LAI value (see Figure 2.16), which is another epistemic interval that can be investigated. On the other hand, the aleatory uncertainty is incorporated in the point cloud when using it for LAD calculation. Depending on the method that ends up being used for LAD modelling, I will combine the uncertainty with  $C_d$  and aim to narrow down the means of modelling those parameters.

### 3.3.9 Validation

This research is purely numerical, meaning no experimental studies will be conducted. I plan to rely on publicly available wind tunnel (AIJ; EWTL) and field measurement data (Allwine and Flaherty, 2006) to perform validation studies, as well as potential collaborations that might develop during the course of the research. I also plan to contact researchers who published studies that included field measurements for validation studies, such as Kenjereš and Ter Kuile (2013); Sousa and Gorlé (2019).

Wind tunnel studies offer greater control of parameters and potentially more straightforward answer on which approach performs better. For example, a wind tunnel experiment will answer which LAD/LAI modelling approach yields the best results. Nevertheless, field investigations are necessary to answer how investigated uncertainties compare to the overall uncertainties of the system. Following the previous example, a more complex way of modelling vegetation parameters might yield better results than the less complex one; however, if there is a negligible influence on the overall uncertainty on a field case, added complexity and more computational effort do not improve the methodology. In words of García-Sánchez et al. “Reducing the epistemic uncertainty in a system with significant aleatory uncertainty might only result in a small improvement of the predictive capabilities of the simulations, while coming at a significantly higher computational cost”.

# Chapter 4

## Preliminary Results

### 4.1 CFD-Ready Reconstruction Framework

The initial plan for the research was to use *3dfier* as the base and adapt it for CFD applications. During the initial work, in consultation with promoters, I came to a conclusion that *3dfier* might not be suitable after all due to strict rules on connecting different classes — a step that authors refer to as “stitching” (Ledoux et al., 2021). For instance, the height of a road is calculated differently from water, and its connection to the water surface is handled differently than a connection to terrain. These modelling-based decisions resulted from a cooperation with the Dutch Kadastre. Notice that CFD favours smooth surfaces (where smooth surfaces can be achieved), so water surfaces, terrain and roads can be incorporated in the DTM with a roughness value accounting for surface specifics. These modelling-based decisions complicate the code without providing benefits to CFD; even worse, they can complicate the meshing process.

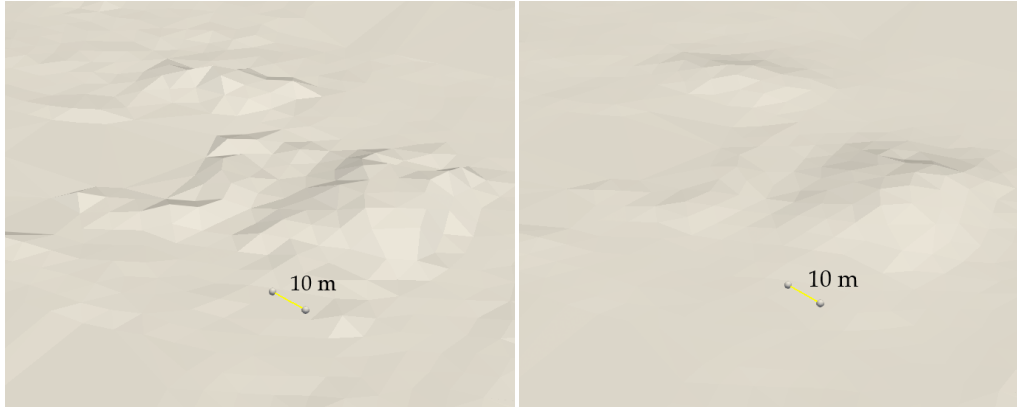
Following the previous decision, I have begun developing my own framework in C++ using the CGAL library. The framework, so far, reconstructs DTM (Section 4.1.1) and buildings at LoD1.2 (Section 4.1.2), and prepares the geometry for CFD by creating a closed round domain with side and top sections (Section 4.1.3). The particular example of the TUD campus I am about to show was made using the BAG dataset for building footprints and the AHN3 database for the point cloud.

#### 4.1.1 Digital Terrain Model

I made the DTM using the constrained Delaunay triangulation of the ground points from the AHN3, with BAG polygon edges specified as constraints. First, I defined the point of interest that serves as the centre of the domain and a radius that acts as the edge of the round domain. Next, following the proposed methodology in Section 3.3.3, I applied the Gaussian filter to smooth the terrain. I defined one height per polygon (the constrained edges) by averaging the elevation of 10 nearest neighbours of the first polygon vertex. Figure 4.1 shows the detail in resulting DTM, with (right) and without the Gaussian filter (left). Notice that smoothing handles local jumps in the terrain but preserves the overall shape. This is crucial to achieve, so the boundary cells near the ground are free from issues but without penalties on the final wind profile. Bear in mind that the tallest point in the figure is around 5 meters high.

#### 4.1.2 LoD1.2 Building Reconstruction

The reconstruction of each building was done by calculating the average height of all points falling within polygon boundaries and extruding them to height. I reconstructed all buildings that are within 60 per cent of the radius used for terrain; influence zone definition from guidelines (Blocken, 2015) will follow. Note that I left out the footprint preparation step, which is the next thing I’m planning



(a) Without smoothing.

(b) With smoothing.

Figure 4.1: Digital terrain model, detail.

to implement. Furthermore, I plan to investigate the 75-99 percentile height as geometric reference in the following iterations, as proposed by Dukai et al. (2019). To determine which geometric reference is the best for CFD, I will have to conduct a sensitivity analysis. The resulting combination of buildings and terrain are shown in Figure 4.2. For the purpose of easier mesh generation, I extruded footprints whose average height is over 5 m.

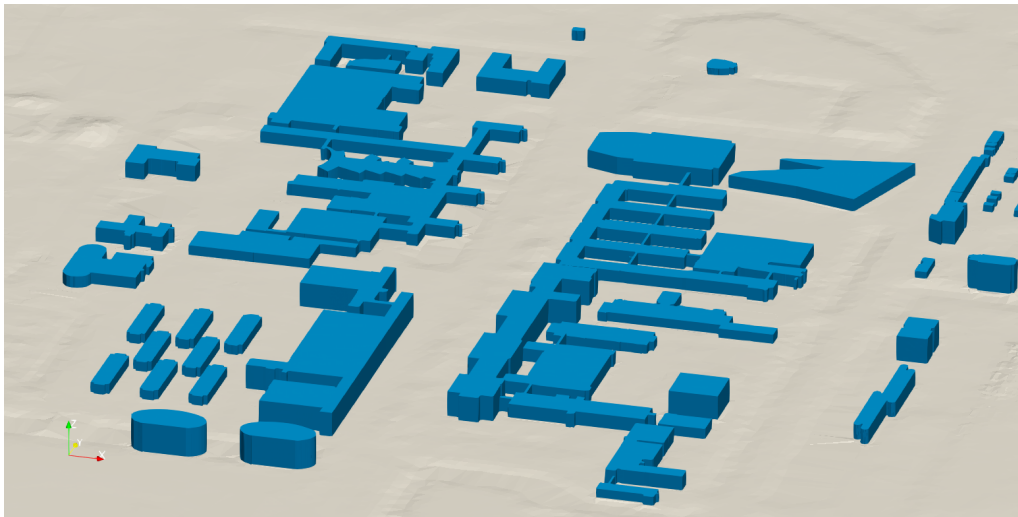


Figure 4.2: Combination of the DTM and LoD1.2 buildings.

### 4.1.3 CFD Domain and Meshing

The next step was to create the domain suitable for CFD and test how the produced geometry copes with an automatic finite volume mesh generator. The domain creation included the following steps:

1. Extruding the terrain outer rim to get round and flat area close to domain boundaries. This newly created outer rim was set to be 20 per cent larger than the terrain radius in this initial testing phase. The rim extrusion is done to avoid potential convergence issues by having perturbations close to inlet and outlet boundaries (Hågbo et al., 2020).
2. Constructing the inlet/outlet boundary as vertical walls that extend to the prescribed domain height,
3. Closing the domain with a surface on the top.

Figure 4.3 shows the final domain with numbers denoting individual steps from the list above. Dimensions in steps 1 and 2 were purposely made smaller than indicated by the best practice guidelines to accelerate the mesh generation and facilitate mesh handling. This closed domain now represents the space that is discretised with finite volumes.

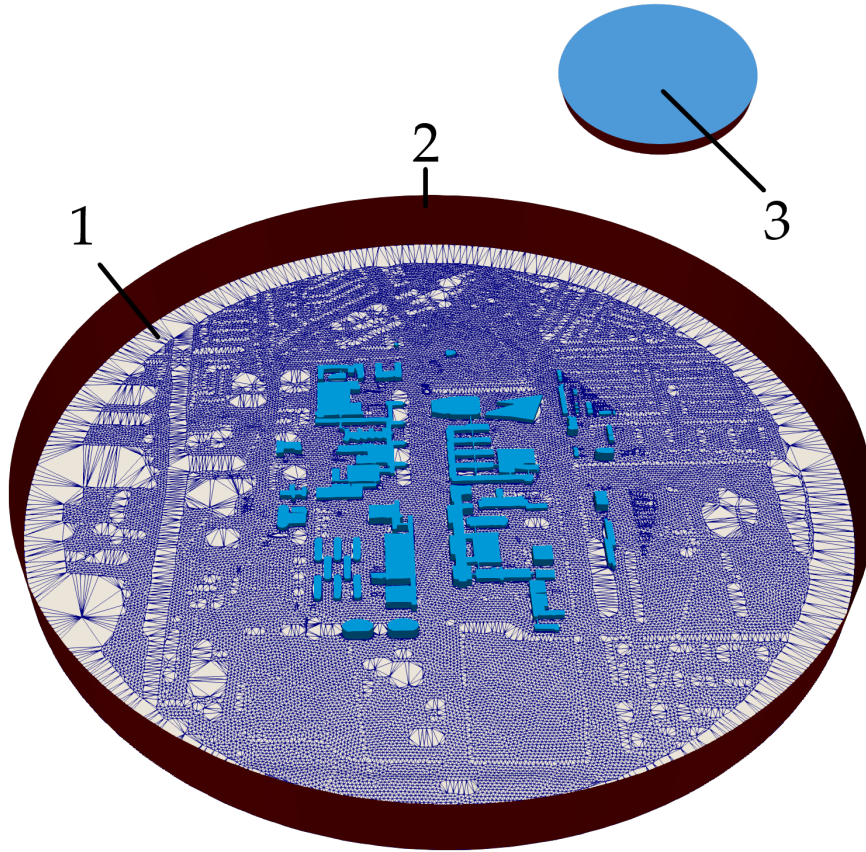


Figure 4.3: Domain used for finite volume mesh creation.

The first meshing tests were done with *cfMesh*, with the goal of pinpointing problematic areas for the meshing software. The first results are shown in Figure 4.4. Please bear in mind that a better quality mesh could have been achieved by adjusting the meshing parameters. But the goal of this research is to provide a geometry that does not require a lot of adjustments of the meshing parameters to obtain a simulation-worthy computational mesh.

With that said, the first problem I have encountered is the worsening of mesh quality close to the connection of the terrain and buildings. Notice in Figure 4.5a that some terrain polygons have sharp angles between them. This is caused by the choice of one average height for the entire building footprint. If the average height is noticeably different from the terrain height at a particular point, the constrained triangulation will result in issues denoted with number “1” in Figure 4.5b. My next course of action will be to set building height for each footprint point individually.

The second issue is the problematic meshing of short edges whose length is relatively small compared to the mesh size (locations “2” in Figure 4.5b). The necessity for footprint generalisation is evident in this case, and my next steps will be to implement the footprint generalisation presented in the methodology (see Section 3.3.2) and observe if there are any improvements.

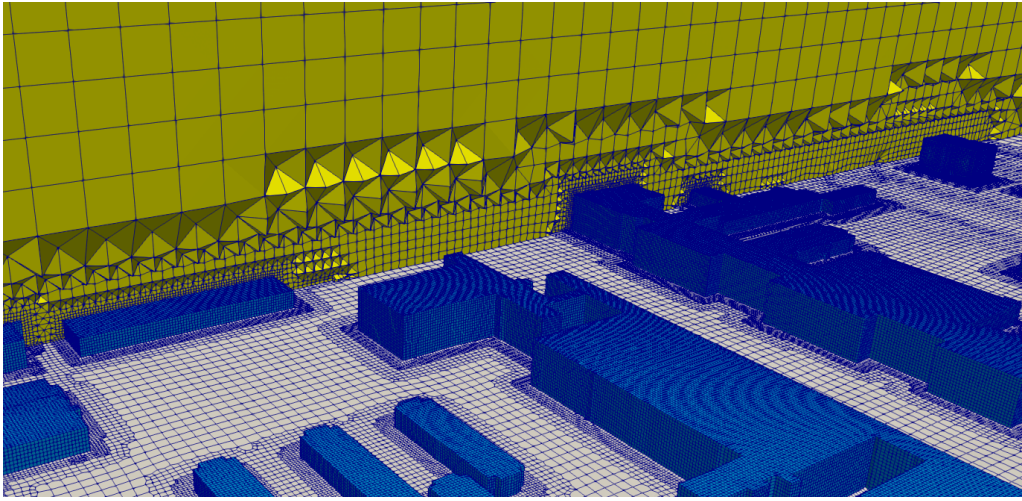
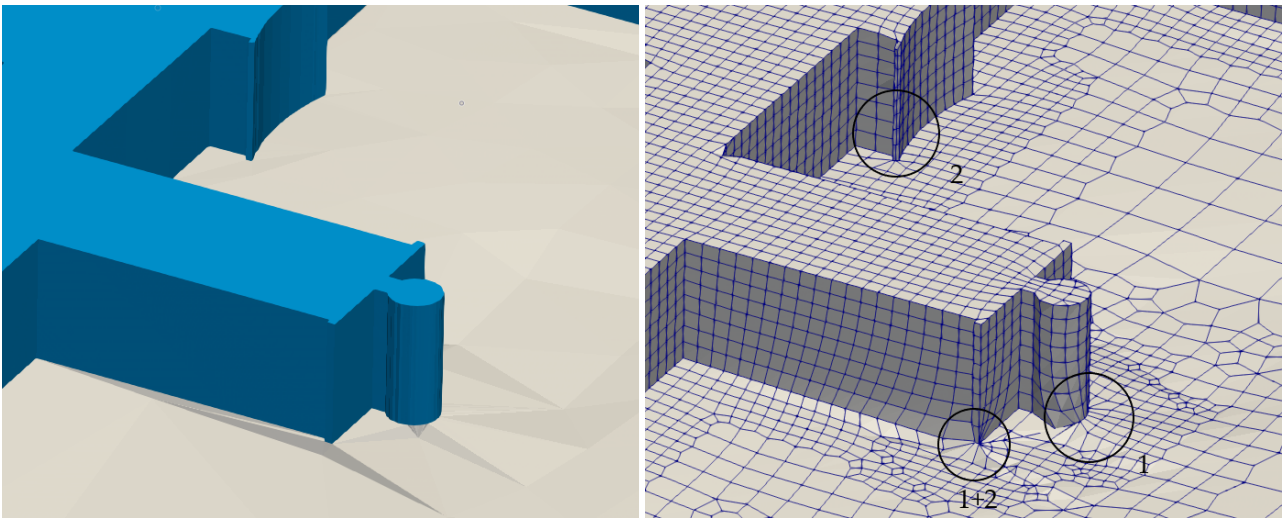


Figure 4.4: Computational domain made with *cfMesh*



(a) Geometry.

(b) Finite volume mesh at the wall boundary.

Figure 4.5: Problematic locations for mesh generation.

# Chapter 5

## Planning and Practical Aspects

### 5.1 Timetable

Figure 5.1 gives an overview of my planned activities over the course of the next three years. The main idea is to work on the geomatics part (automatic geometry preparation) and CFD simulations (uncertainty quantification) simultaneously and use UQ findings to steer decisions related to automatic geometry preparation. It will also serve to establish guidelines related to LoD in CFD. Along with the core research, I plan to conduct other activities that complement the PhD process:

- Publications in scientific journals,
- Assisting with courses and supervising master's thesis,
- Participating in conferences and workshops,
- Research visits and collaborations with other research groups,
- Fulfilling Graduate School obligations.

#### 5.1.1 First Year Report

Along with the research presented in Chapter 4, during the first year, I followed four master courses to gain education in PhD-related topics such as digital terrain modelling, modelling of the built environment, atmospheric turbulence and C++ programming. I supervised one master's thesis and served as a substitute supervisor for another.

### 5.2 Tools and Technical Aspects

Table 5.1 provides a list of software, programming languages and libraries used or planned to be used for this research. All the tools developed during the course of this research will be published as open-source and available through git repositories of the 3D geoinformation group<sup>1</sup> or my personal git<sup>2</sup>. Also, I will use the group's computational resources, as well as a high-performance computing (HPC) cluster to conduct my simulations.

### 5.3 Graduate School Obligations

In conjunction with research, PhD candidates at TU Delft are required to complete the Doctoral Education (DE) programme at the university's Graduate School (GS). There is a requirement to

---

<sup>1</sup><https://github.com/tudelft3d>

<sup>2</sup><https://github.com/ipadjen>

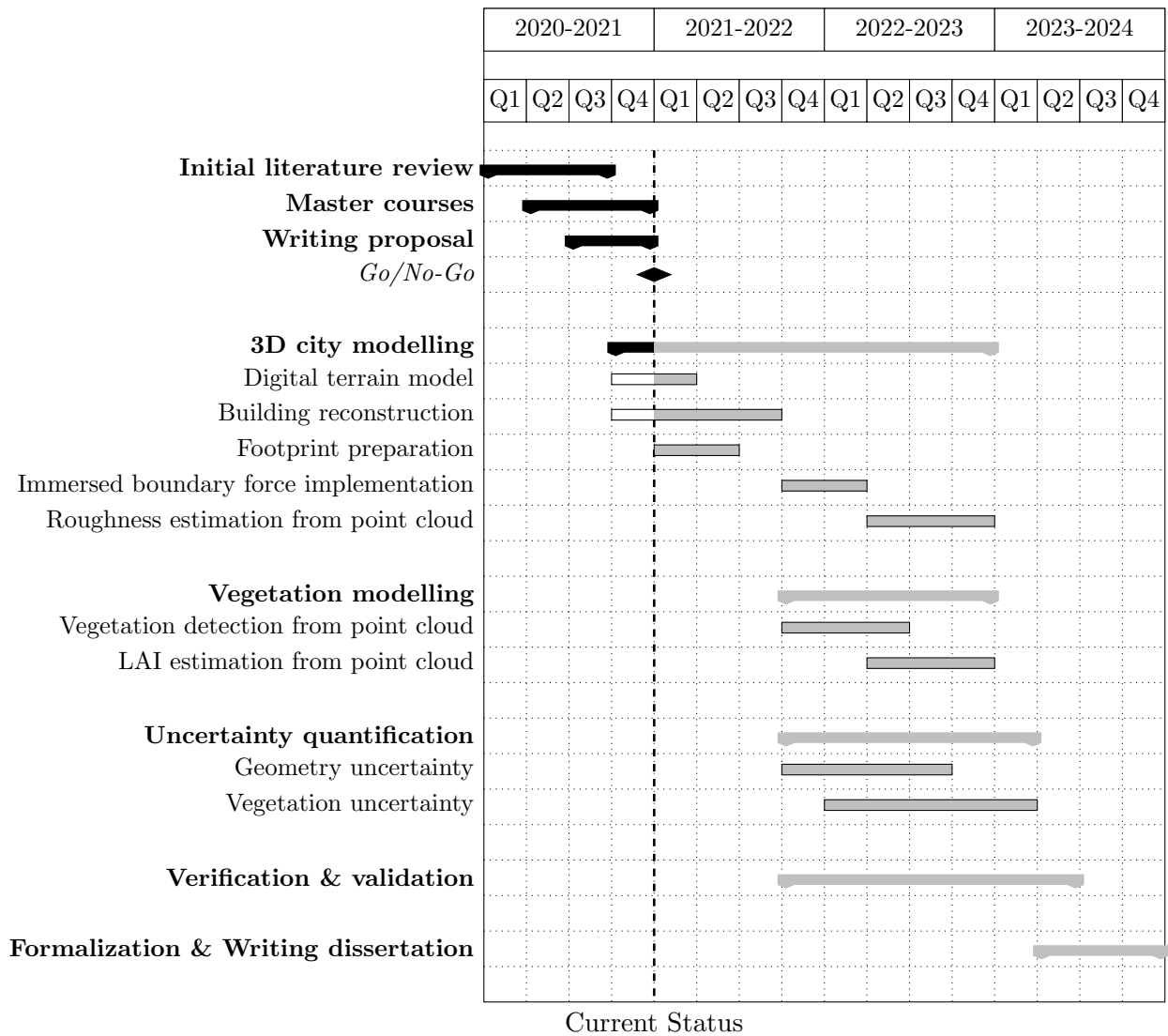


Figure 5.1: Planning overview

Table 5.1: Tools and Technical aspects

Technology	Purpose
Software	
OpenFOAM	CFD solver
Dakota	Uncertainty quantification
CLion	Coding & debugging
Meshlab	Visualisation
Paraview	Post-processing & visualisation
Programming Languages	
C++	CFD geometry preparation, OpenFOAM implementations
Python	Scripting
Libraries	
CGAL	Geometry processing
LAStools	LiDAR procesing
nlohmann/json	JSON file processing in C++

earn a total of 45 GS credits where 1 GS credit is equal to 8 hours of coursework and 4 hours of preparation/assignment. The DE programme is divided amongst three skill categories:

- Discipline-related skills (15 GS credits of which 5 - 15 can be Learning on-the-Job)
- Research skills (15 GS credits)
- Transferable skills (15 GS credits)

Table 5.2 summarises the plan to complete the DE programme including completed GS credits. In total, I have achieved 20/45 GS credits.

Table 5.2: Graduate School Progress and Plan

Category	Name	GS Credits	Status
Discipline Related	MS Course: Atmospheric Turbulence	4	Completed
	MS Course: Digital Terrain Modelling	5	Completed
	MS Course: OO Scientific Programming with C++	3	Completed
	OpenFOAM workshop	3	Planned
Research	Supervising MSc students	6	In Progress
	Work consultation with research partners	1	Planned
	Writing the first conference paper	1	Planned
	Writing an international, peer-reviewed journal article	4	Planned
	TA: assisting in laboratory course	3	Planned
Transferable	Brain Management	2	Completed
	PhD Start-Up	2	Completed
	Scientific Text Processing with L <sup>A</sup> T <sub>E</sub> X	1.5	Completed
	Teamwork, Leadership and Group Dynamics	1.5	Completed
	Time Management I - Foundation	1	Completed
	Personal and Career Development	1	Planned
	Writing a Scientific Article in English	3	Planned
Writing a Dissertation	3	Planned	

## 5.4 Data Management Plan

The data management plan has been filled and submitted through the online form of the TUD. In short, my research will be open-data, publicly available (GitHub) free and open-source software (FOSS) with a license that depends on terms and conditions of other open-source code I use as a part of my workflow.

## 5.5 Publications

### 5.5.1 Publication Plan

My plan with publications, both journals and conferences, is to establish a balance between the two communities. The goal is to bring two fields closer, so publications should reflect that with publishing in journals both dedicated to geomatics and fluid dynamics. Examples of journals include but are not limited to:

- For CFD: Computers & Fluids, Building and Environment, Journal of Wind Engineering & Industrial Aerodynamics,



- For geomatics: Computers, Environment and Urban Systems, International Journal of Geographical Information Science,
- For the final automated workflow: Journal of Open Source Software.

I am not able to devise a plan on the number of publications, as this largely depends on the research progress.

### 5.5.2 Previous Publications

- Pađen, I., Petranović, Z., Edelbauer, W., and Vujanović, M. (2021). Numerical modeling of spray secondary atomization with the Euler-Eulerian multi-fluid approach. *Computers and Fluids*, 222

## 5.6 Acknowledgements

I want to thank my colleague Anna Labetski for providing me with this L<sup>A</sup>T<sub>E</sub>Xtemplate.

# Bibliography

- AIJ (2021). Architectural Institute of Japan, Validation Benchmark Tests. [https://www.aij.or.jp/jpn/publish/cfdguide/index\\_e.htm](https://www.aij.or.jp/jpn/publish/cfdguide/index_e.htm). (Accessed July 2021). 35
- Airaksinen, E., Bergström, M., Heinonen, H., Kaisla, K., Lahti, K., and Suomisto, J. (2019). The Kalasatama digital twins project—The final report of the KIRA-digi pilot project. Technical report, City of Helsinki. 2
- Allwine, K. J. and Flaherty, J. E. (2006). Joint urban 2003: Study overview and instrument locations. Technical report, Pacific Northwest National Lab. (PNNL), Richland, WA (United States). 18, 35
- Amorim, J. H., Rodrigues, V., Tavares, R., Valente, J., and Borrego, C. (2013). CFD modelling of the aerodynamic effect of trees on urban air pollution dispersion. *Science of the Total Environment*, 461-462:541–551. 19
- AMS (2021). Glossary of Meteorology: Atmospheric Boundary Layer. [https://glossary.ametsoc.org/wiki/Atmospheric\\_boundary\\_layer](https://glossary.ametsoc.org/wiki/Atmospheric_boundary_layer). (Accessed June 2021). 4
- Arroyo Ohori, K., Ledoux, H., and Meijers, M. (2012). Validation and automatic repair of planar partitions using a constrained triangulation. *Photogrammetrie, Fernerkundung, Geoinformation*, 5:613–630. ISSN: 1432–8364. 29
- Arroyo Ohori, K., Ledoux, H., and Peters, R. (2021). 3D modelling of the built environment. Course materials at the Delft University of Technology. iv, 7, 8, 9, 10, 29
- Ashie, Y. and Kono, T. (2011). Urban-scale CFD analysis in support of a climate-sensitive design for the Tokyo Bay area. *International Journal of Climatology*, 31(2):174–188. 6
- Baker, T. J. (2005). Mesh generation: Art or science? *Progress in Aerospace Sciences*, 41(1):29–63. 6
- Basu, S. and Lacser, A. (2017). A Cautionary Note on the Use of Monin–Obukhov Similarity Theory in Very High-Resolution Large-Eddy Simulations. *Boundary-Layer Meteorology*, 163(2):351–355. 17
- Beall, M. W., Walsh, J., and Shephard, M. S. (2004). A comparison of techniques for geometry access related to mesh generation. *Engineering with Computers*, 20(3):210–221. 8, 14
- Biljecki, F. and Arroyo Ohori, K. (2015). Automatic Semantic-preserving Conversion Between OBJ and CityGML. In Biljecki, F. and Turre, V., editors, *Eurographics Workshop on Urban Data Modelling and Visualisation*. The Eurographics Association. 32
- Biljecki, F., Heuvelink, G. B., Ledoux, H., and Stoter, J. (2018a). The effect of acquisition error and level of detail on the accuracy of spatial analyses. *Cartography and Geographic Information Science*, 45(2):156–176. 23, 34
- Biljecki, F., Kumar, K., and Nagel, C. (2018b). CityGML Application Domain Extension (ADE): overview of developments. *Open Geospatial Data, Software and Standards*, 3(1):1–17. 7
- Biljecki, F., Ledoux, H., Du, X., Stoter, J., Soon, K. H., and Khoo, V. H. S. (2016a). The most

- common geometric and semantic errors in CityGML datasets. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-2/W1:13–22. 9
- Biljecki, F., Ledoux, H., and Stoter, J. (2016b). An improved LOD specification for 3D building models. *Computers, Environment and Urban Systems*, 59:25–37. iv, 8, 12, 14
- Biljecki, F., Ledoux, H., Stoter, J., and Zhao, J. (2014). Formalisation of the level of detail in 3D city modelling. *Computers, Environment and Urban Systems*, 48:1–15. 8
- Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., and Çöltekin, A. (2015). Applications of 3D city models: State of the art review. *ISPRS International Journal of Geo-Information*, 4(4):2842–2889. 2, 7
- Blocken, B. (2014). 50 years of Computational Wind Engineering: Past, present and future. *Journal of Wind Engineering and Industrial Aerodynamics*, 129:69–102. iv, 5
- Blocken, B. (2015). Computational Fluid Dynamics for urban physics: Importance, scales, possibilities, limitations and ten tips and tricks towards accurate and reliable simulations. *Building and Environment*. iv, 4, 11, 12, 14, 27, 36
- Blocken, B. (2018). *LES over RANS in building simulation for outdoor and indoor applications: A foregone conclusion?*, volume 11. Springer. 24, 33
- Blocken, B. (2021). Introduction to the simulation of atmospheric flows. In *CFD for atmospheric flows and wind engineering*. von Karman Institute Lecture Series. 1, 24
- Blocken, B., Janssen, W. D., and 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. *Environmental Modelling and Software*, 30:15–34. 14, 15
- Blocken, B., Stathopoulos, T., and Carmeliet, J. (2007). CFD simulation of the atmospheric boundary layer: wall function problems. *Atmospheric Environment*, 41(2):238–252. 31
- Blocken, B., Vervoort, R., and van Hooff, T. (2016). Reduction of outdoor particulate matter concentrations by local removal in semi-enclosed parking garages: A preliminary case study for Eindhoven city center. *Journal of Wind Engineering and Industrial Aerodynamics*, 159(October):80–98. 17
- Buccolieri, R., Santiago, J. L., Rivas, E., and Sanchez, B. (2018). Review on urban tree modelling in CFD simulations: Aerodynamic, deposition and thermal effects. *Urban Forestry and Urban Greening*, 31(July 2017):212–220. 20, 34
- Carrasco, L., Giam, X., Papes, M., and Sheldon, K. S. (2019). Metrics of lidar-derived 3D vegetation structure reveal contrasting effects of horizontal and vertical forest heterogeneity on bird species richness. *Remote Sensing*, 11(7):1–19. 21
- Cebeci, T. and Bradshaw, P. (1977). Momentum transfer in boundary layers. 11
- Chan, S. T. and Leach, M. J. (2007). A validation of FEM3MP with Joint Urban 2003 data. *Journal of Applied Meteorology and Climatology*, 46(12):2127–2146. 18, 30
- Colin, J. and Faivre, R. (2010). Aerodynamic roughness length estimation from very high-resolution imaging LIDAR observations over the Heihe basin in China. *Hydrology and Earth System Sciences*, 14(12):2661–2669. 17
- Commandeur, T. J. F. (2012). Footprint decomposition combined with point cloud segmentation for producing valid 3D models. Master’s thesis, Delft University of Technology. 11, 29
- Constant, E., Favier, J., Meldi, M., Meliga, P., and Serre, E. (2017). An immersed boundary method in OpenFOAM: Verification and validation. *Computers and Fluids*, 157:55–72. 6

- de Groot, R. (2020). *Automatic construction of 3D tree models in multiple levels of detail from airborne LiDAR data*. Master’s thesis, Delft University of Technology. iv, 19, 31, 32
- De Vries, A. C., Kustas, W. P., Ritchie, J. C., Klaassen, W., Menenti, M., Rango, A., and Prueger, J. H. (2003). Effective aerodynamic roughness estimated from airborne laser altimeter measurements of surface features. *International Journal of Remote Sensing*, 24(7):1545–1558. 17
- Deiningner, M. E., von der Grün, M., Pieperreit, R., Schneider, S., Santhanavanich, T., Coors, V., and Voß, U. (2020). A Continuous, Semi-Automated Workflow: From 3D City Models with Geometric Optimization and CFD Simulations to Visualization of Wind in an Urban Environment. *ISPRS International Journal of Geo-Information*, 9(11):657. 10, 13, 15, 20
- Du, S., Lindenbergh, R., Ledoux, H., Stoter, J., and Nan, L. (2019). AdTree: Accurate, detailed, and automatic modelling of laser-scanned trees. *Remote Sensing*, 11(18):1–19. 19
- Dukai, B., Ledoux, H., and Stoter, J. E. (2019). A multi-height LOD1 model of all buildings in the Netherlands. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume 4. 37
- Dukai, B., van Liempt, J., Peters, R., Stoter, J., Vitalis, S., and Wu, T. (2021). 3D BAG. <https://3dbag.nl/>. (Accessed June 2021). 2, 10, 11, 13, 29
- EWTL (2021). University of Hamburg, Compilation of Experimental Data for Validation of Microscale Dispersion Models. <https://mi-pub.cen.uni-hamburg.de/index.php?id=433>. (Accessed July 2021). 35
- Faivre, R., Colin, J., and Menenti, M. (2017). Evaluation of methods for aerodynamic roughness length retrieval from very high-resolution imaging LIDAR observations over the heihe basin in China. *Remote Sensing*, 9(1):1–25. 17
- Franke, J., Hellsten, A., Schlünzen, H., and Carissimo, B. (2007). *Best practice guideline for the CFD simulation of flows in the urban environment*, volume 44. 14, 27
- Gao, Z., Bresson, R., Qu, Y., Milliez, M., de Munck, C., and Carissimo, B. (2018). High resolution unsteady RANS simulation of wind, thermal effects and pollution dispersion for studying urban renewal scenarios in a neighborhood of Toulouse. *Urban Climate*, 23:114–130. 13
- García-Sánchez, C. and Górlé, C. (2018). Uncertainty quantification for microscale CFD simulations based on input from mesoscale codes. *Journal of Wind Engineering and Industrial Aerodynamics*, 176(February 2018):87–97. 7
- García-Sánchez, C., Philips, D. A., and Górlé, C. (2014). Quantifying inflow uncertainties for CFD simulations of the flow in downtown Oklahoma City. *Building and Environment*, 78(2014):118–129. 22, 35
- García-Sánchez, C., Van Tendeloo, G., and Górlé, C. (2017). Quantifying inflow uncertainties in RANS simulations of urban pollutant dispersion. *Atmospheric Environment*, 161(2017):263–273. 22, 34
- Giraudot, S. (2021). CGAL 5.3 - GIS (Geographic Information System). [https://doc.cgal.org/latest/Manual/tuto\\_gis.html](https://doc.cgal.org/latest/Manual/tuto_gis.html). (Accessed June 2021). 30
- Gobeawan, L., Lin, E. S., Tandon, A., Yee, A. T., Khoo, V. H., Teo, S. N., Yi, S., Lim, C. W., Wong, S. T., Wise, D. J., Cheng, P., Liew, S. C., Huang, X., Li, Q. H., Teo, L. S., Fekete, G. S., and Poto, M. T. (2018). Modeling trees for virtual Singapore: From data acquisition to CityGML models. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 42(4/W10):55–62. 19

- Gorlé, C. (2021). Uncertainty quantification for atmospheric boundary layer flows. In *CFD for atmospheric flows and wind engineering*. von Karman Institute Lecture Series. 22
- Gorlé, C., Garcia-Sanchez, C., and Iaccarino, G. (2015). Quantifying inflow and RANS turbulence model form uncertainties for wind engineering flows. *Journal of Wind Engineering and Industrial Aerodynamics*, 144(2015):202–212. 22, 34
- Gorlé, C. and Iaccarino, G. (2013). A framework for epistemic uncertainty quantification of turbulent scalar flux models for Reynolds-averaged Navier-Stokes simulations. *Physics of Fluids*, 25(5). 22
- Grimmond, C. S. and Oke, T. R. (1999). Aerodynamic properties of urban areas derived from analysis of surface form. *Journal of Applied Meteorology*, 38(9). 17
- Gromke, C. and 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. *Environmental Pollution*, 196(1):214–223. 20
- Hachenberger, P. (2021). CGAL 5.2.1 - 3D Minkowski Sum of Polyhedra. [https://doc.cgal.org/latest/Minkowski\\_sum\\_3/index.html](https://doc.cgal.org/latest/Minkowski_sum_3/index.html). (Accessed May 2021). 15
- Hachenberger, P. and Kettner, L. (2021). CGAL 5.3 - 3D Boolean Operations on Nef Polyhedra. [https://cgal.geometryfactory.com/CGAL/doc/master/Nef\\_3/index.html](https://cgal.geometryfactory.com/CGAL/doc/master/Nef_3/index.html). (Accessed May 2021). 15
- Hågbo, T.-O., Giljarhus, K. E. T., and Hjertager, B. H. (2020). Influence of Geometry Acquisition Method on Pedestrian Wind Simulations. Manuscript submitted for publication. iv, 13, 14, 22, 27, 37
- Hang, J. and Li, Y. (2010). Wind conditions in idealized building clusters: Macroscopic simulations using a porous turbulence model. *Boundary-Layer Meteorology*, 136(1):129–159. 18
- Hefny Salim, M., Heinke Schlünzen, K., and Grawe, D. (2015). Including trees in the numerical simulations of the wind flow in urban areas: Should we care? *Journal of Wind Engineering and Industrial Aerodynamics*, 144(August):84–95. iv, 19
- Ho, B. (2019). Simscales - Computational Wind Engineering with CFD in the Cloud. <https://www.simscales.com/blog/2019/03/computational-wind-engineering/>. (Accessed June 2021). 8
- Holland, D. E., Berglund, J. A., Spruce, J. P., and Mckellip, R. D. (2008). Derivation of effective aerodynamic surface roughness in urban areas from airborne lidar terrain data. *Journal of Applied Meteorology and Climatology*, 47(10):2614–2626. 17
- Hong, B., Qin, H., and Lin, B. (2018). Prediction of wind environment and indoor/outdoor relationships for PM2.5 in different building-tree grouping patterns. *Atmosphere*, 9(2). 20
- Huang, J. (2020). Structure-aware urban building reconstruction. PhD proposal, 3D Geoinformation Research Group, Delft University of Technology. 11, 30
- Iaccarino, G., Eldred, M., Doostan, A., and Ghattas, O. (2009). Introduction to uncertainty quantification. In *SIAM Conference on Computational Science and Engineering Report*. 21, 22
- Iglesias, M. A., Law, K. J., and Stuart, A. M. (2013). Ensemble kalman methods for inverse problems. *Inverse Problems*, 29(4):045001. 22
- ISO (2003). ISO 19107:2003 (Geographic Information: Spatial Schema). *International Organization for Standardization*, (01-101). 8, 29
- Jia, S. and Wang, Y. (2021). Effect of heat mitigation strategies on thermal environment, thermal comfort, and walkability: A case study in Hong Kong. *Building and Environment*, 201(May). 1

- Juretić, F., Barač, M., and Bosnar, D. (2017). Automatic Mesh Generation of a simplified 3D model of The White House for CFD Analysis with cfMeshPRO. Technical report, Creative Fields, Zagreb. 27
- Kada, M. (2008). Generalization of 3D building models for map-like presentations. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences: XXXVII.[S. l.]*: ISPRS, pages 399–404. 11, 29
- Kamoske, A. G., Dahlin, K. M., Stark, S. C., and Serbin, S. P. (2019). Leaf area density from airborne LiDAR: Comparing sensors and resolutions in a temperate broadleaf forest ecosystem. *Forest Ecology and Management*, 433(November 2018):364–375. 21
- Kang, G., Kim, J. J., and Choi, W. (2020). Computational fluid dynamics simulation of tree effects on pedestrian wind comfort in an urban area. *Sustainable Cities and Society*, 56(August 2019):102086. 19
- Kenjereš, S. and Ter Kuile, B. (2013). Modelling and simulations of turbulent flows in urban areas with vegetation. *Journal of Wind Engineering and Industrial Aerodynamics*, 123(PA):43–55. 20, 35
- Labetski, A. (2017). A framework for application-specific generalisation of buildings and roads in 3d city models. PhD proposal, 3D Geoinformation Research Group, Delft University of Technology. 30
- Labetski, A., Ledoux, H., and Stoter, J. (2017). Generalising 3D buildings from LoD2 to LoD1. In *GISRUK 2017 Conference Proceedings*. 11, 28
- Lauder, B. E. and Spalding, D. B. (1974). The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, 3(2). 11
- Ledoux, H. (2013). On the validation of solids represented with the international standards for geographic information. *Computer-Aided Civil and Infrastructure Engineering*, 28(9):693–706. 9
- Ledoux, H. (2018). val3dity: validation of 3D GIS primitives according to the international standards. *Open Geospatial Data, Software and Standards*, 3(1):1–12. 9, 29
- Ledoux, H., Arroyo Ohori, K., Kumar, K., Dukai, B., Labetski, A., and Vitalis, S. (2019). CityJSON: a compact and easy-to-use encoding of the CityGML data model. *Open Geospatial Data, Software and Standards*, 4(1). 7, 8
- Ledoux, H., Arroyo Ohori, K., and Peters, R. (2020). *Computational modelling of terrains*. 3D geoinformation, Delft University of Technology. 10, 30
- Ledoux, H., Biljecki, F., Dukai, B., Kumar, K., Peters, R., Stoter, J., and Commandeur, T. (2021). 3dfier: automatic reconstruction of 3D city models. *Journal of Open Source Software*, 6(57):2866. 2, 10, 14, 36
- Li, Z., Ming, T., Shi, T., Zhang, H., Wen, C. Y., Lu, X., Dong, X., Wu, Y., de Richter, R., Li, W., and Peng, C. (2021). Review on pollutant dispersion in urban areas-part B: Local mitigation strategies, optimization framework, and evaluation theory. *Building and Environment*, 198(February). 1
- Liang, L., Xiaofeng, L., Borong, L., and Yingxin, Z. (2006). Improved k- $\epsilon$  two-equation turbulence model for canopy flow. *Atmospheric Environment*, 40(4):762–770. 20
- Liu, S., Pan, W., Zhang, H., Cheng, X., Long, Z., and Chen, Q. (2017). CFD simulations of wind distribution in an urban community with a full-scale geometrical model. *Building and Environment*, 117:11–23. 11, 16
- Liu, S., Pan, W., Zhao, X., Zhang, H., Cheng, X., Long, Z., and Chen, Q. (2018). Influence of surrounding buildings on wind flow around a building predicted by CFD simulations. *Building and Environment*, 140:1–10. iv, 15, 17, 18, 27

- Lu, Y., Behar, E., Donnelly, S., Lien, J. M., Camelli, F., and Wong, D. (2011). Fast and robust generation of city-scale seamless 3D urban models. *CAD Computer Aided Design*, 43(11):1380–1390. 12
- Lukač, N., Štumberger, G., and Žalik, B. (2017). Wind resource assessment using airborne LiDAR data and smoothed particle hydrodynamics. *Environmental Modelling and Software*, 95:1–12. 17, 31
- Macdonald, R. W., Griffiths, R. F., and Hall, D. J. (1998). An improved method for the estimation of surface roughness of obstacle arrays. *Atmospheric Environment*, 32(11):1857–1864. 31
- Manickathan, L. (2019). *Impact of vegetation on urban microclimate*. PhD thesis, ETH Zürich. 21, 34
- Manickathan, L., Defraeye, T., Allegrini, J., Derome, D., and Carmeliet, J. (2018a). Comparative study of flow field and drag coefficient of model and small natural trees in a wind tunnel. *Urban Forestry and Urban Greening*, 35(December 2017):230–239. 21
- Manickathan, L., Defraeye, T., Allegrini, J., Derome, D., and Carmeliet, J. (2018b). Parametric study of the influence of environmental factors and tree properties on the transpirative cooling effect of trees. *Agricultural and Forest Meteorology*, 248(October 2017):259–274. 19
- McNabola, A., Broderick, B. M., and Gill, L. W. (2009). A numerical investigation of the impact of low boundary walls on pedestrian exposure to air pollutants in urban street canyons. *Science of the Total Environment*, 407(2):760–769. 1
- Mirzaei, P. A. (2021). CFD modeling of micro and urban climates: Problems to be solved in the new decade. *Sustainable Cities and Society*, 69(March). 7, 8, 12, 14
- Montazeri, H., Blocken, B., Janssen, W. D., and van Hooff, T. (2013). CFD evaluation of new second-skin facade concept for wind comfort on building balconies: Case study for the Park Tower in Antwerp. *Building and Environment*, 68:179–192. 14
- Moonen, P., Defraeye, T., Dorer, V., Blocken, B., and Carmeliet, J. (2012). Urban Physics: Effect of the micro-climate on comfort, health and energy demand. *Frontiers of Architectural Research*, 1(3):197–228. 1
- Moradpour, M., Afshin, H., and Farhanieh, B. (2017). A numerical investigation of reactive air pollutant dispersion in urban street canyons with tree planting. *Atmospheric Pollution Research*, 8(2):253–266. 20
- Moukalled, F., Mangani, L., and Darwish, M. (2016). *The finite volume method in computational fluid dynamics : An Advanced Introduction with OpenFOAM and Matlab*. Springer International Publishing. 5
- Moussiopoulos, N., Schlünzen, H., and Louka, P. (2003). *Modelling Urban Air Pollution*, pages 121–154. Springer Berlin Heidelberg, Berlin, Heidelberg. iv, 5
- Muñoz-Esparza, D., Kosović, B., Mirocha, J., and van Beeck, J. (2014). Bridging the Transition from Mesoscale to Microscale Turbulence in Numerical Weather Prediction Models. *Boundary-Layer Meteorology*, 153(3). 7
- Muñoz-Esparza, D., Sauer, J. A., Shin, H. H., Sharman, R., Kosović, B., Meech, S., García-Sánchez, C., Steiner, M., Kniewel, J., Pinto, J., and Swerdlin, S. (2020). Inclusion of Building-Resolving Capabilities Into the FastEddy® GPU-LES Model Using an Immersed Body Force Method. *Journal of Advances in Modeling Earth Systems*, 12(11):1–17. 18, 30, 31
- Nan, L. and Wonka, P. (2017). PolyFit: Polygonal Surface Reconstruction from Point Clouds. In *Proceedings of the IEEE International Conference on Computer Vision*. 10, 11, 29, 30

- Oke, T. R., Mills, G., Christen, A., and Voogt, J. A. (2017). *Urban Climates*. Cambridge University Press, Cambridge. iv, 4, 5
- Open Geospatial Consortium (2012). OGC City Geography Markup Language (CityGML) Encoding Standard 2.0.0. 7, 8
- Orlanski, I. (1975). A rational subdivision of scales for atmospheric processes. *Bulletin of the American Meteorological Society*, 56(5). iv, 4, 5
- Oshio, H., Asawa, T., Hoyano, A., and Miyasaka, S. (2015). Estimation of the leaf area density distribution of individual trees using high-resolution and multi-return airborne LiDAR data. *Remote Sensing of Environment*, 166:116–125. 21
- Park, G., Kim, C., Lee, M., and Choi, C. (2020). Building geometry simplification for improving mesh quality of numerical analysis model. *Applied Sciences (Switzerland)*, 10(16). iv, 15, 16
- Pađen, I., Petranović, Z., Edelbauer, W., and Vujanović, M. (2021). Numerical modeling of spray secondary atomization with the Euler-Eulerian multi-fluid approach. *Computers and Fluids*, 222.
- Pieperreit, R., Beuster, A., von der Gruen, M., Voß, U., Pries, M., and Wagner, U. (2019). Towards wind-simulation of virtual 3D city models in a collaborative VR environment. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*. 14, 15
- Pieperreit, R., Deininger, M., Kada, M., Pries, M., and Voß, U. (2018). A sweep-plane algorithm for the simplification of 3D building models in the application scenario of wind simulations. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*. iv, 15, 16
- Pieperreit, R., Schilling, A., Alam, N., Wewetzer, M., Pries, M., and Coors, V. (2016). Towards Automatic Processing of Virtual City Models for Simulations. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4(2W1):39–45. 15
- Piroozmand, P., Mussetti, G., Allegrini, J., Mohammadi, M. H., Akrami, E., and Carmeliet, J. (2020). Coupled CFD framework with mesoscale urban climate model: Application to microscale urban flows with weak synoptic forcing. *Journal of Wind Engineering and Industrial Aerodynamics*, 197(December 2019):104059. 7
- Porter, S. R. (2020). *Reducing Computational Fluid Dynamics Simulation Cost by Intelligently Reducing Geometric Complexity*. PhD thesis, Curtin University. 14
- Pruss-Ustun, A. and Corvalan, C. (2016). Preventing disease through healthy environment: A global assessment of the burden of disease from environmental risks. *World Health Organization*. 1
- Randerson, D. (1976). Overview of regional-scale numerical models. iv, 5
- Rapp, B. E. (2017). Finite Volume Method. *Microfluidics: Modelling, Mechanics and Mathematics*, pages 633–654. 5
- Raumonen, P., Kaasalainen, M., Markku, Å., Kaasalainen, S., Kaartinen, H., Vastaranta, M., Holopainen, M., Disney, M., and Lewis, P. (2013). Fast automatic precision tree models from terrestrial laser scanner data. *Remote Sensing*, 5(2):491–520. 19
- Ricci, A., Janssen, W. D., van Wijhe, H. J., and Blocken, B. (2020a). CFD simulation of wind forces on ships in ports: Case study for the Rotterdam Cruise Terminal. *Journal of Wind Engineering and Industrial Aerodynamics*, 205(July):104315. 14
- Ricci, A., Kalkman, I., Blocken, B., Burlando, M., Freda, A., and Repetto, M. P. (2017). Local-scale forcing effects on wind flows in an urban environment: Impact of geometrical simplifications.



- Journal of Wind Engineering and Industrial Aerodynamics*, 170(September 2016):238–255. 8, 14, 22
- Ricci, A., Kalkman, I., Blocken, B., Burlando, M., and Repetto, M. P. (2020b). Impact of turbulence models and roughness height in 3D steady RANS simulations of wind flow in an urban environment. *Building and Environment*, 171:106617. 33
- Richards, P. and Hoxey, R. P. (1993). Appropriate boundary conditions for computational wind engineering models using the k- $\epsilon$  turbulence model. *Journal of Wind Engineering and Industrial Aerodynamics*, 46-47. 33
- Ritchie, H. and Roser, M. (2018). Urbanization. *Our World in Data*. <https://ourworldindata.org/urbanization>. 1
- Sadrehaghghi, I. (2018). Mesh Generation in CFD. *CFD Open Series*, 1(July). 9
- Saeedraashed, Y. S. and Benim, A. C. (2019). Validation Methods of Geometric 3D-CityGML Data for Urban Wind Simulations. *E3S Web of Conferences*, 128. 13
- Santiago, J. L., Buccolieri, R., Rivas, E., Calvete-Sogo, H., Sanchez, B., Martilli, A., Alonso, R., Elustondo, D., Santamaría, J. M., and Martin, F. (2019). CFD modelling of vegetation barrier effects on the reduction of traffic-related pollutant concentration in an avenue of Pamplona, Spain. *Sustainable Cities and Society*, 48(February):101559. 20
- Shaw, R. H. and Schumann, U. (1992). Large-eddy simulation of turbulent flow above and within a forest. *Boundary-Layer Meteorology*, 61(1-2):47–64. iv, 20, 21
- Shin, H. H., Muñoz-Esparza, D., Sauer, J. A., and Steiner, M. (2021). Large-eddy simulations of stability-varying atmospheric boundary layer flow over isolated buildings. *Journal of the Atmospheric Sciences*, 78(5):1487–1501. 18
- Simões, T. and Estanqueiro, A. (2016). A new methodology for urban wind resource assessment. *Renewable Energy*, 89:598–605. 12
- Simscale (2021). Advanced Modelling PWC. <https://www.simscale.com/docs/analysis-types/pedestrian-wind-comfort-analysis/advanced-modelling/>. (Accessed June 2021). 20
- Šíp, V. and Beneš, L. (2016). CFD Optimization of a Vegetation Barrier. In *Numerical Mathematics and Advanced Applications ENUMATH 2015*, pages 471–479, Cham. Springer International Publishing. 23
- Slotnick, J., Khodadoust, A., Alonso, J., and Darmofal, D. (2014). CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences. *NNASA/CR-2014-218178*. 1
- Sousa, J., García-Sánchez, C., and Górlé, C. (2018). Improving urban flow predictions through data assimilation. *Building and Environment*, 132(January):282–290. 22
- Sousa, J. and Górlé, C. (2019). Computational urban flow predictions with Bayesian inference: Validation with field data. *Building and Environment*, 154(November 2018):13–22. 19, 22, 35
- Stull, R. B. (1988). *An introduction to boundary layer meteorology*. Springer, Dordrecht. 17
- Temel, O., Bricteux, L., and van Beeck, J. (2018). Coupled WRF-OpenFOAM study of wind flow over complex terrain. *Journal of Wind Engineering and Industrial Aerodynamics*, 174. 7
- Toja-Silva, F., Chen, J., Hachinger, S., and Hase, F. (2017). CFD simulation of CO2 dispersion from urban thermal power plant: Analysis of turbulent Schmidt number and comparison with Gaussian plume model and measurements. *Journal of Wind Engineering and Industrial Aerodynamics*, 169(August):177–193. 13

- Toja-Silva, F., Pregel-Hoderlein, C., and Chen, J. (2018). On the urban geometry generalization for CFD simulation of gas dispersion from chimneys: Comparison with Gaussian plume model. *Journal of Wind Engineering and Industrial Aerodynamics*, 177(December 2017):1–18. 13
- Tominaga, Y., Mochida, A., Yoshie, R., Kataoka, H., Nozu, T., Yoshikawa, M., and Shirasawa, T. (2008). AIJ guidelines for practical applications of CFD to pedestrian wind environment around buildings. *Journal of Wind Engineering and Industrial Aerodynamics*, 96(10-11). iv, 18, 27
- Tong, Z., Chen, Y., and Malkawi, A. (2016). Defining the Influence Region in neighborhood-scale CFD simulations for natural ventilation design. *Applied Energy*, 182. iv, 18
- Toparlar, Y., Blocken, B., Maiheu, B., and van Heijst, G. J. (2017). A review on the CFD analysis of urban microclimate. *Renewable and Sustainable Energy Reviews*, 80(May):1613–1640. 22
- Toparlar, Y., Blocken, B., Maiheu, B., and van Heijst, G. J. (2018). The effect of an urban park on the microclimate in its vicinity: a case study for Antwerp, Belgium. *International Journal of Climatology*, 38(December 2017):e303–e322. 15
- Toparlar, Y., Blocken, B., Vos, P., Van Heijst, G. J., Janssen, W. D., van Hooff, T., Montazeri, H., and Timmermans, H. J. (2015). CFD simulation and validation of urban microclimate: A case study for Bergpolder Zuid, Rotterdam. *Building and Environment*, 83:79–90. 17
- van Druenen, T., van Hooff, T., Montazeri, H., and Blocken, B. (2019). CFD evaluation of building geometry modifications to reduce pedestrian-level wind speed. *Building and Environment*, 163(April):106293. 1
- van Hooff, T. and Blocken, B. (2010a). Coupled urban wind flow and indoor natural ventilation modelling on a high-resolution grid: A case study for the Amsterdam ArenA stadium. *Environmental Modelling and Software*, 25(1):51–65. iv, 14, 15
- van Hooff, T. and Blocken, B. (2010b). On the effect of wind direction and urban surroundings on natural ventilation of a large semi-enclosed stadium. *Computers and Fluids*, 39(7):1146–1155. 18
- Vervecken, L., Camps, J., and Meyers, J. (2013). Accounting for wind-direction fluctuations in Reynolds-averaged simulation of near-range atmospheric dispersion. *Atmospheric Environment*, 72:142–150. 22
- Vogel, S. (1989). Drag and reconfiguration of broad leaves in high winds. *Journal of Experimental Botany*, 40(8):941–948. 34
- von Der Grün, M., Zamre, P., Chen, Y., Lutz, T., Voß, U., and Kramer, E. (2020). Numerical study and LiDAR based validation of the wind field in urban sites. *Journal of Physics: Conference Series*, 1618(4). 20
- Wang, H., Peng, C., Li, W., Ding, C., Ming, T., and Zhou, N. (2021a). Porous media: A faster numerical simulation method applicable to real urban communities. *Urban Climate*, 38(April):100865. 18
- Wang, L., Su, J., Gu, Z., and Tang, L. (2021b). Numerical study on flow field and pollutant dispersion in an ideal street canyon within a real tree model at different wind velocities. *Computers and Mathematics with Applications*, 81(xxxx):679–692. 19
- Weller, H. G., Tabor, G., Jasak, H., and Fureby, C. (1998). A tensorial approach to computational continuum mechanics using object-oriented techniques. *Computers in Physics*, 12(6). 32
- Wieringa, J. (1992). Updating the Davenport roughness classification. *Journal of Wind Engineering and Industrial Aerodynamics*, 41(1-3):357–368. 11, 17, 31

- Wilson, N. R. and Shaw, R. H. (1977). A higher order closure model for canopy flow. *Journal of Applied Meteorology*, 16(11 , Nov.1977). 34
- Xiao, H. and Cinnella, P. (2019). Quantification of model uncertainty in RANS simulations: A review. *Progress in Aerospace Sciences*, 108(April):1–31. iv, 22, 23
- Yoshie, R., Mochida, A., Tominaga, Y., Kataoka, H., Harimoto, K., Nozu, T., and Shirasawa, T. (2007). Cooperative project for CFD prediction of pedestrian wind environment in the Architectural Institute of Japan. *Journal of Wind Engineering and Industrial Aerodynamics*, 95(9-11):1551–1578. 14
- Zhang, S., Kwok, K. C., Liu, H., Jiang, Y., Dong, K., and Wang, B. (2021). A CFD study of wind assessment in urban topology with complex wind flow. *Sustainable Cities and Society*, 71(May):103006. 13