



UNIVERSITY OF
CHEMISTRY AND TECHNOLOGY
PRAGUE

Institute of Thermomechanics, Czech Academy of Sciences

Department of Waves in Solids

Air flow in the urban area of Hsinchu city

REPORT FROM THE STUDENT INTERNSHIP

AUTHOR

PROGRAMME

YEAR

Ing. Tomáš Hlavatý

NARLabs student internship programme

2023

Summary

Abstract of the report will be there

Acknowledgements

Acknowledgements for supervisors and the NARLabs.

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 1 |
| 2 | Mathematical model | 2 |
| 2.1 | Model geometry generation and boundary conditions | 2 |
| 2.2 | Governing equations | 3 |
| 2.3 | Developed OpenFOAMCase python class documentation | 3 |
| 3 | Computational environment | 4 |
| 3.1 | Preparation of the singularity container | 4 |
| 3.2 | Preparation of Slurm control script and running the task | 5 |
| 4 | Numerical experiments | 6 |
| 5 | Conclusions | 7 |
| 6 | Nomenclature | 8 |
| | Bibliography | 10 |

1 Introduction

The present work arose as the part of the authors student internship in Taiwan National Applied Research Laboratories (NARLabs). The report was written out in National Center for High-performance Computing.

2 Mathematical model

In the present work, the air-flow in the inhabited area is investigated together with the transport of the pollution emerging on the main road. The chosen simulation approach is a segregated one: (i) a steady state velocity field is pre-calculated using Navier-Stokes equations, and (ii) it is used in scalar transport equation to simulate advancement of the pollution.

The present section describing the used mathematical model is structured as follows: (i) a model geometry and a computational mesh generation are presented, (ii) used boundary conditions are summarized and (iii) the governing equations are briefly outlined.

2.1 Model geometry generation and boundary conditions

As stated, the air flow and the spread of the pollution in the inhabited area is of an interest. A studied part of the city is $L_l = 500$ m long, $L_w = 500$ m wide, and the highest building is $L_h = 105$ m tall. The available *stl* file together with its dimensions and the orientation in the chosen Cartesian coordinate system is depicted in Figure 1.

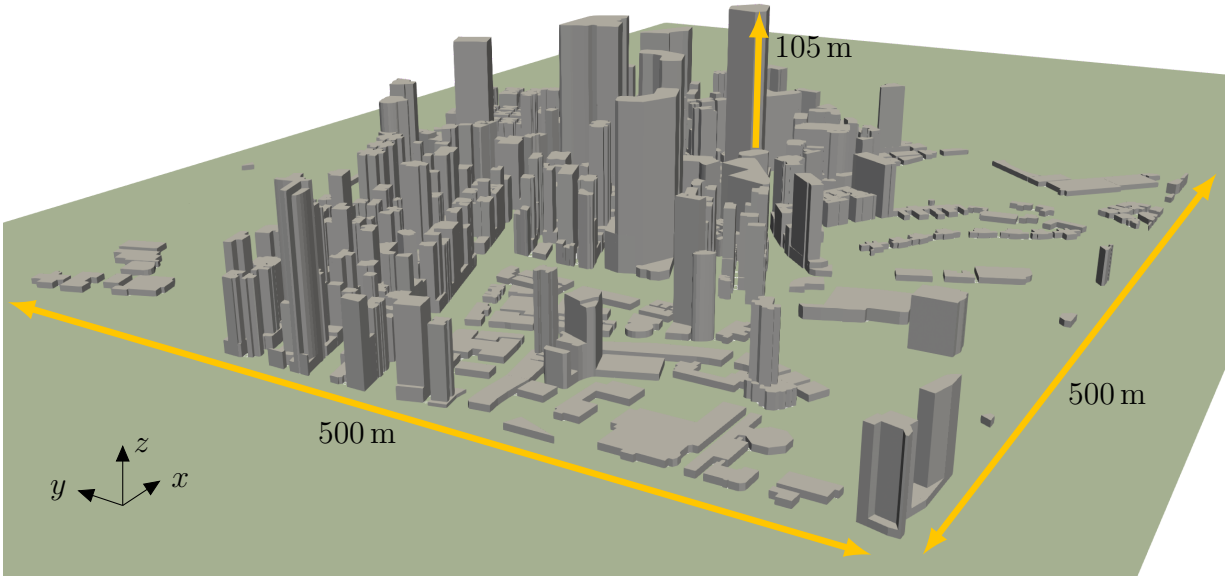


Figure 1: Studied city geometry (used *stl* file) with its dimensions and chosen Cartesian coordinate system.

Computational mesh Computational mesh was prepared using the OpenFOAM `blockMesh` and `snappyHexMesh` utilities. The computational domain dimensions were determined based on the review by Pantusheva et al. [1]. In particular, we introduce both upstream, and downstream buffers in front of, and behind the city, which are in order $L_{CFDu} = 5 L_h$, and $L_{CFDd} = 10 L_h$ long. Furthermore, the computational domain is $L_{CFDh} = 6 L_h$ high. The y -normal slices cut in the middle of the used computational mesh are shown in Figure 2. Note the grading of the computational mesh in the z -direction, and its refinement in the vicinity of the city.

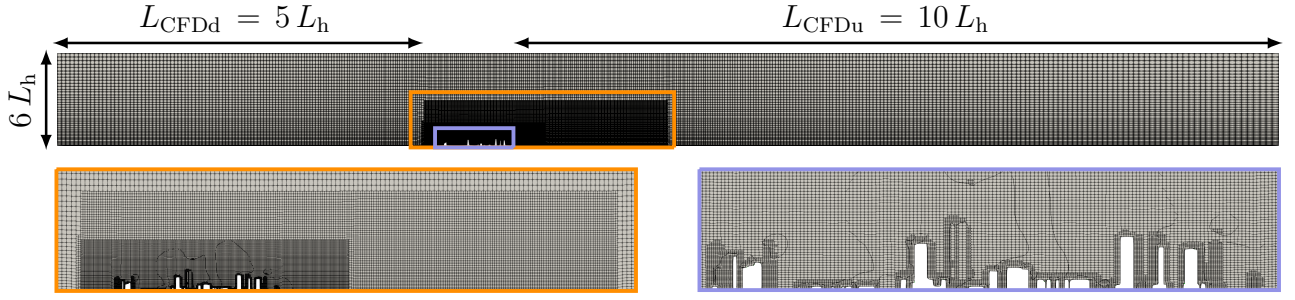


Figure 2: y -normal slices through the half of the computational mesh.

Boundary conditions The model behavior in the computational domain (Ω) is described by partial differential equations outlined in the following subsection 2.2 (Governing equations). However, these need to be supplied with the consistent boundary conditions on the Ω boundary, $\partial\Omega$.

As depicted in Figure 3, the computational domain boundary is split into (i) *inlet* ($\partial\Omega_i$), (ii) *outlet* ($\partial\Omega_o$), (iii) *ground* ($\partial\Omega_g$), (iv) *sides* ($\partial\Omega_s$), and (v) *top* ($\partial\Omega_t$).

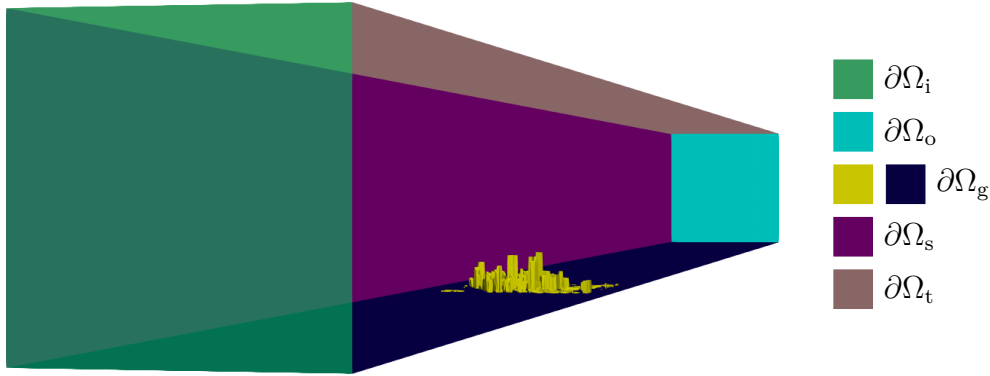


Figure 3: Boundaries of the computational domain.

2.2 Governing equations

The description of the chosen governing equations can be split into two parts, (i) air-flow governing equations, and (ii) the pollution

2.3 Developed OpenFOAMCase python class documentation

3 Computational environment

NCHC servers use Singularity container platform [2] and Slurm workload manager [3] to run user-defined tasks. Thus,

- (i) custom singularity container, which includes necessary packages installed and compiled inside, is prepared, and
- (ii) Slurm task (which runs within the container) is prepared and run at the computational server.

3.1 Preparation of the singularity container

Assuming you have super-user permission and singularity installed on local (see step-by-step guide in [4]), a preparation of the singularity container image from docker ubuntu:latest release can be done as follows:

1. Navigate outside the home directory and work here, e.g.:

```
mkdir /tmp/  
mkdir /tmp/test  
cd /tmp/test/
```

2. New container (./ubuntu) can be built from ubuntu docker repository using:

```
sudo singularity build --sandbox ./ubuntu docker://ubuntu:latest
```

NOTE: --sandbox flag allows to write into container later

Shell inside container can be opened using:

```
sudo singularity shell ./ubuntu --writable
```

where --writable flag again allows to write into container and install packages here.

Openfoam.org/v10 and other used packages can be installed inside the container as:

```
apt update  
apt install python3 python3-pip wget vim software-properties-common  
python3-tk pip3 install matplotlib  
sh -c "wget -O - https://dl.openfoam.org/gpg.key >  
/etc/apt/trusted.gpg.d/openfoam.asc"  
add-apt-repository http://dl.openfoam.org/ubuntu  
apt update  
apt install openfoam10
```


Compilation of the custom solver inside container is done as follows:

1. Source OpenFOAM in the container shell:

```
. /opt/openfoam10/etc/bashrc
```

2. Navigate to solver folder, e.g.:

```
cd /tmp/pollutionFoam
```

3. Compile solver executing:

```
wmake.
```

When everything is installed, the `.sif` container file can be built from prepared `/tmp/test/ubuntu` directory using:

```
sudo singularity build /tmp/test/ubuntu.sif /tmp/test/ubuntu/
```

Following the above listed guideline, singularity container image `ubuntu.sif` is created. This can be uploaded to NCHC servers and used as described in following subsection.

3.2 Preparation of Slurm control script and running the task

4 Numerical experiments

Some nice results her.

5 Conclusions

And conclusion here

6 Nomenclature

| | | |
|--------------------------|-------|---|
| c_i | | Molar concentration of i -th molar specie |
| c_T | | Total molar concentration |
| Co | | Courant number |
| d | | Diameter |
| D_i | | Molar diffusivity of i -th molar specie |
| D_i^{eff} | | Effective molar diffusivity of i -th molar specie |
| \mathbf{d}_{PN} | | Vector connecting centroids of P and N |
| f | | Face of the cell |
| \mathbf{f}_b | | Body forces acting on cell |
| \mathbf{g} | | Gravitational acceleration |
| h | | Specific enthalpy |
| I | | Time interval |
| I^h | | Discretized time interval |
| m | | Number of discretized FV cells |
| M | | Molar mass |
| n | | Number of species |
| \mathbf{n} | | Outer normal vector |
| \mathbf{n}_f | | Outer normal vector of the face f |
| p | | Pressure |
| p_{ref} | | Reference pressure |
| \tilde{p} | | Kinematic pressure |
| Q | | Computational domain |
| r_i | | Reaction source of the i -th molar specie |
| R^g | | Universal gas constant |
| Re | | Reynolds number |
| s_ϕ | | Source of the ϕ |
| \mathbf{S}_f | | Face area vector |
| t | | Time |
| T | | Temperature |
| T_{ref} | | Reference temperature |
| $\mathbf{u} = (u, v, w)$ | | Velocity |
| y_i | | Molar fraction of the i -molar specie |
| α | | Heat transfer coefficient |
| ε | | Porosity |
| Γ_ϕ | | Diffusivity of ϕ |
| κ | | Permeability |
| λ | | Heat conductivity |
| μ | | Dynamic viscosity |
| ν | | Kinematic viscosity |
| Ω | | Domain |
| Ω^h | | Discretized domain |
| Ω_P^h | | Cell P |
| $\delta\Omega_i^h$ | | Volume of the cell |
| $\partial\Omega$ | | Domain boundary |
| ϕ | | Intensive tensorial quantity |
| ϕ_P | | Value of ϕ in the cell centroid of cell P |

| | | |
|---------------------------|-------|--|
| ϕ_f | | Value of ϕ in the face centroid of face f |
| Φ_ϕ | | Flux intensity of ϕ |
| $\Phi_{\phi,\text{conv}}$ | | Convective flux intensity of ϕ |
| $\Phi_{\phi,\text{diff}}$ | | Diffusive flux intensity of ϕ |
| ρ | | Fluid mass density |
| Σ | | Total stress tensor |
| τ | | Tortuosity |
| $\boldsymbol{\tau}$ | | Viscous stress tensor |
| ∇ | | Nabla differential operator |

References

- [1] Mariya Pantusheva et al. “Air Pollution Dispersion Modelling in Urban Environment Using CFD: A Systematic Review”. In: *Atmosphere* 13.10 (2022). DOI: 10.3390/atmos13101640.
- [2] SingularityCE Project Contributors. *SingularityCE User Guide*. <https://docs.sylabs.io/guides/main/user-guide.pdf>. Accessed: 20-2-2023.
- [3] Andy B. Yoo, Morris A. Jette and Mark Grondona. “SLURM: Simple Linux Utility for Resource Management”. In: *Job Scheduling Strategies for Parallel Processing*. 2003.
- [4] SingularityCE Project Contributors. *SingularityCE Installation*. <https://docs.sylabs.io/guides/3.0/user-guide/installation.html>. Accessed: 20-2-2023.

Potencial appendix here.