# Towards Real-Time Urban Physics Simulations with Digital Twins

Jacopo Bonari,* Lisa Kühn,* Max von Danwitz,* Alexander Popp,*†

*German Aerospace Center, Institute for the Protection of Terrestrial Infrastructures
Rathausallee 12, 53757 Sankt Augustin, Germany
†University of the Bundeswehr Munich, Institute for Mathematics and Computer-Based Simulation
Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany

*Abstract*—**Urban populations continue to grow, highlighting the critical need to safeguard civilians against potential disruptions, such as dangerous gas contaminant dispersion. The digital twin (DT) framework offers promise in analyzing and predicting such events. This study presents a computational framework for modelling airborne contaminant dispersion in built environments. Leveraging automatic generation of computational domains and solution processes, the proposed framework solves the underlying physical model equations with the finite element method (FEM) for numerical solutions. Model order reduction (MOR) methods are investigated to enhance computational efficiency without compromising accuracy. The study outlines the automatic model generation process, the details of the employed model, and the future perspectives for the realization of a DT. Throughout this research, the aim is to develop a reliable predictive model combining physics and data in a hybrid DT to provide informed real-time support within evacuation scenarios.**

*Index Terms*—**Digital twins, automatic model generation, computational fluid dynamics, model order reduction.**

## I. INTRODUCTION

Critical infrastructures constitute the backbone of modern societies and, as cities expand and the number of urban inhabitants grows, protecting civil population against disruptions of potentially dangerous systems is a matter of primary importance. Among several different threats posed to the community by possible incidents, the protection against dangerous gas contaminant dispersion emerges as a focal point, given its potential for catastrophic consequences on public health, environment, and economic activity. At the same time, the conceptual and practical development of the digital twin (DT) framework has allowed for it to emerge as a prominent tool of analysis and prediction. Even though no unique definition of a DT is acknowledged by researchers, the thorough review by Boyes and Watson [1] finds in [2] a comprehensive definition as "*A live digital coupling of the state of a physical asset or process to a virtual representation with a functional output*".

In [3], the DT concept is customized for the infrastructure domain. Here, a requirements analysis is performed with the target of improving possible crisis management and defining measures to increase the resilience of infrastructure components. The authors conclude that the DT paradigm is appropriate for the scope. Concurrently, many efforts have been undertaken by researchers to apply this concept to the analysis of dangerous gas or generic contaminant dispersion,

with particular focus on risks directly connected to oil and gas industry [4] and contaminant spread in small scale closed environments [5]. On the other hand, to the best of the authors' knowledge, less attention has been devoted to the analysis of contaminant dispersion in urban or, in general, built environments. This is in spite of the existence of a well-established methodology in both computational fluid dynamics (CFD) applied to urban physics [6] and research on DT for cities, both on the concept level and applied to specific case studies [7], [8].

In this study, we propose a computational framework for the analysis of airborne contaminant dispersion in a built environment to enhance situational awareness in a specific risk scenario. The framework strongly relies on an automatic generation of the computational domain and solution process. Information about the position and the geometry of buildings influencing the gas diffusion process can be either collected automatically according to location based queries, or prompted by a user in a structured file, while simulation parameters are prompted once at the beginning of the workflow. The objective is to develop a reliable predictive model for a DT, aiming to offer informed real-time support within, for example, evacuation scenarios.

The core part of the workflow foresees the solution of two sets of partial differential equations (PDEs). The first simulation step deals with the evaluation of a wind vector field, coincident with the solution of the incompressible Navier-Stokes (INS) equations; the second step deals with the atmospheric diffusion of the contaminant agent based on an advection-diffusion (AD) process, governed by the previously evaluated wind field.

The finite element method (FEM) is employed for the numerical solution of the PDEs. Thanks to its versatility and to the increase of available computational resources, numerical methods have proven themselves as reliable instruments in the context under examination.

During the last three decades, simulation of wind distribution in urban environments has strongly relied on computational fluid dynamics (CFD). From the first seminal analyses of wind flows over simple buildings [9], the attention evolved towards specific studies of pedestrian comfort in windy environments [10] and to cases tailored for the analysis of wind

interaction with real structures [11].

Since a DT model requires by definition a continuous and bidirectional stream of data and information, and since it strives to virtually reproduce a real phenomenon, the capability of delivering real-time results is of paramount importance. To obtain this goal, model order reduction (MOR) methods are investigated in the current study to deliver fast yet accurate results in terms of wind field evaluation, which is the most computationally intense part of the workflow. To achieve this, we apply proper orthogonal decomposition (POD) in combination with an approximation of the nonlinearities of the INS equations following the works in [12], [13]. For a general overview of MOR for CFD in general we refer to [14].

The reminder of the article is structured as follows: in Sec. II a brief overview of the complete workflow is given, while Sec. III, Sec. IV, and Sec. V present the methodological details of data collection, preparing a virtual replica and performing urban physics simulations, respectively. Section VI showcases exemplary solutions for two different test cases characterized by increasing levels of complexity. Finally, Sec. VII presents future perspectives and possible model extensions, together with the steps required for a future migration from a numerical model to a fully bidirectional DT.

## II. WORKFLOW OVERVIEW

To enable urban physics simulations for general built environments, we develop a highly automatized workflow, see Fig. 1. The complete process is governed by a Python class. The process starts from geo-referenced building data in a database and delivers at the end of each simulation run results in a format compliant with standard geographic information systems (GIS). Details of core ingredients are outlined in the following sections.

## III. DOMAIN DEFINITION AND DATA COLLECTION

To facilitate a standardized data collection procedure, we start with a precise definition of the computational domain that is considered in the urban physics simulations.

### A. Description of the simulation domain

A computational domain $Q_T = (0, t_f) \times \Omega$ is defined with $(0, t_f)$ an analysis time window and $\Omega$ a rectangular, connected subset of $\mathbb{R}^2$. Since the focus of the study hinges on the distribution of the contaminant in the open air, buildings are characterized by their outer perimeter only, which represents an obstacle for free air circulation. Wind is supposed to enter the domain at constant speed and perpendicular to the bottom side of the model, while wind velocity is assumed to be null in correspondence of the lateral sides of the domain and by the buildings sides, the latter being the only physical boundaries of the domain. Finally, the top is supposed to be the outflow side, for which free-flow conditions are foreseen. In symbols: $\partial \Omega = \Gamma_i \cap \Gamma_o \cap \Gamma_n$, where $i$, $o$, and $n$ are the subscripts for the *inflow* side, the *outflow* side, and the *no slip* sides. As a last step, a source for the contaminant leak is supposed to be
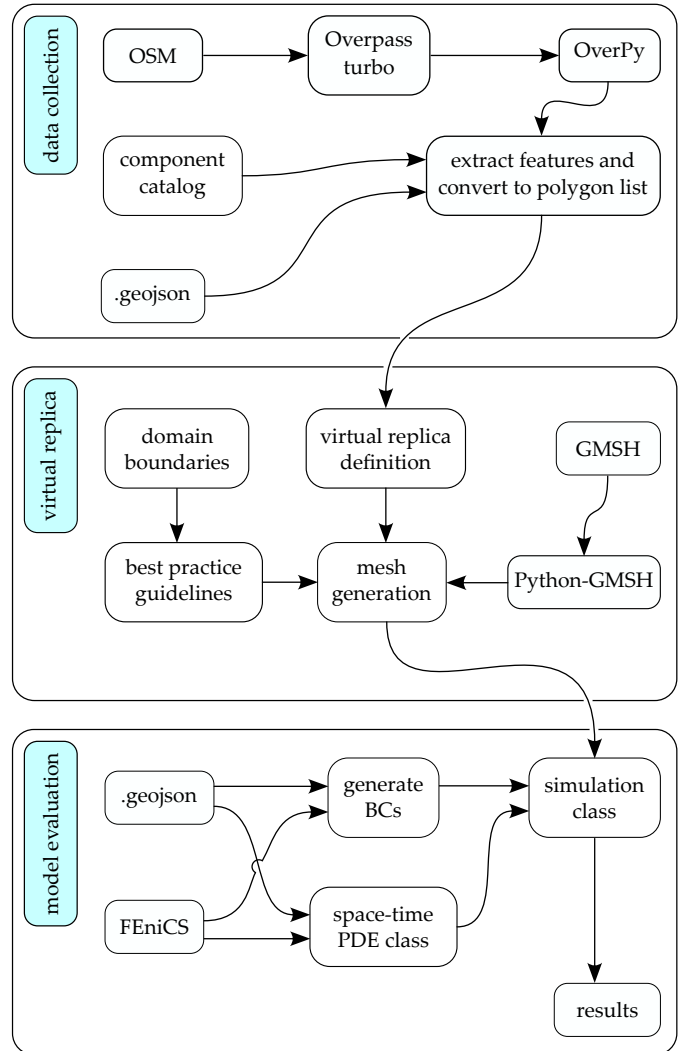


Fig. 1: Urban physics simulation workflow. Starting from geo-referenced building data in a database, here OpenStreetMap (OSM), the highly automatized workflow generates a FEM-mesh using GMSH [15] and performs simulations based on the PDE solver FEniCS [16].

located in the surrounding of a specific point $x_c \in \Omega$ in the form of a scalar field $c_0(x)$. A sketch of the aforementioned environment is depicted in Fig. 2.

### B. Geo-referenced data collection

An automatic domain construction is then performed exploiting geo-referenced data according to three different procedures.

- Information on building geometries can be recollected using OpenStreetMap (OSM) [17], which constitutes a free and openly accessible geographic repository, continuously updated and managed by volunteers through open cooperation. Within OSM, data is collected through public surveys, delineated from aerial visuals, or inte-
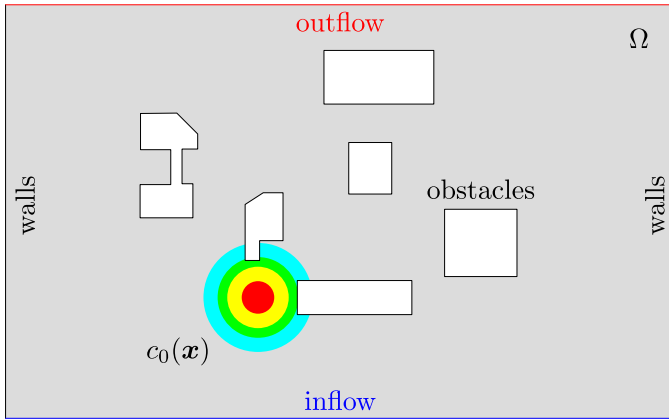
Fig. 2: Computational domain with highlighted inflow and outflow boundaries and location of initial gas source.



Fig. 3: Analysis-ready FEM mesh with automatic local refinement in the narrow crevices between the buildings.

grated from other freely licensed geographic data outlets. Web-based data mining tools for OSM, e.g. Overpass turbo [18], can be used to extract information in a format suitable for a seamless conversion to CAD software kernels or, in general, geometry libraries. Furthermore, the Python wrapper OverPy is employed to access Overpass turbo, making the whole geometry acquisition step self-contained within a Python environment workflow.

- Building locations can also be accessed through a *component catalog* [19], consisting of a database containing datasheets, components, symbols, and code in a graph-shaped structure that facilitates the integration of components into the proposed simulation environment.
- Finally, data can also be prompted providing a custom `.geojson` file, in which every feature describes a building in terms of type and geometry, the latter being limited for now to the list of its corners according to an established geodesic reference system.

## IV. VIRTUAL REPLICA

With the polygons extracted, an overall rectangular domain border can be defined, where the clearance of each side from the buildings cluster can be defined according to different strategies. These borders do not represent a physical boundary, and, to avoid the introduction of spurious effects during the simulation, they must be located far enough from the represented physical objects. Here, the concept of *blockage ratio* ($BR$) [6] is applied, which states that artificial accelerations are avoided when the ratio between the projected length of the buildings' sides on the domain border perpendicular to wind direction, and the border itself, is less than $17\%$.

### A. Mesh generation

After the two-dimensional domain is defined as a surface in terms of bounding polygons, the geometry is discretized into a FEM mesh by the GMSH [15] Python API Python-GMSH, with proper physical tags assigned to discriminate among portions of the boundary representing inflow and outflow portions,
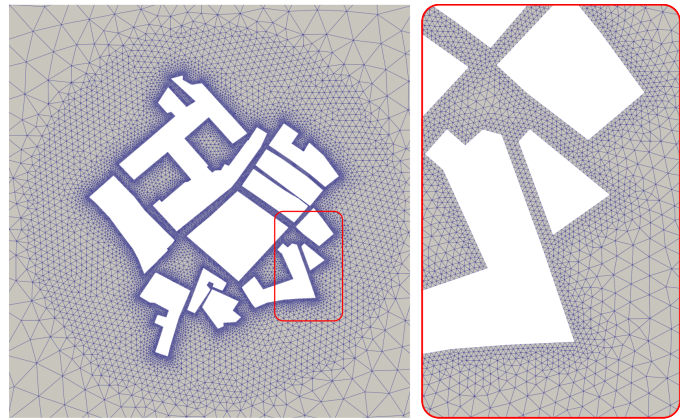
and the no-slip sides, where the wind velocity is supposed to be zero. The whole mesh generation procedure is carried out automatically provided that characteristic mesh sizes $l_c$ are defined during the initialization step. This is necessary in order to satisfy the convergence criteria of the numerical solution procedure of the PDEs describing the problem. In the specific example, three different values of $l_c$ have been employed, each resulting in a increasing level of refinement, Fig. 3. A very fine mesh is defined on the sides of the buildings, a requirement demanded by an accurate wind field evaluation; a mesh with slightly larger elements is then defined between the buildings, small enough to guarantee the stability of the numerical evaluation of the contaminant dispersion; finally, a coarse mesh is constructed in the area far from the buildings, since this area only represents a buffer zone necessary to respect the requirement on the $BR$ parameter. The triangular mesh is generated with a frontal Delaunay algorithm allowing for automatic local mesh refinement [20].

### B. Boundary and initial conditions

The last input requirement consists in the definition of boundary and initial conditions required by the two sets of PDEs. Concerning the INS problem, the required input data are *wind intensity* and *wind direction*; regarding the AD problem they consist in an initial distribution of the contaminant under examination. Since the final goal of our research foresees the integration of the workflow into a DT framework, this data is supposed to be continuously collected through sensors, e.g. metereological stations and detection control units. At the current preliminary stage, a structured `.geojson` file is hard-coded at the beginning of the process and prompts to the required pieces of information.

## V. PHYSICS-BASED SIMULATION

The core section of the simulation run consists in the subsequent numerical solution of the INS and AD systems of equations, which are both performed using FEniCS [16], an open-source computing platform for the solution of PDEs.

The two sets of equations are coupled through the wind field, which is first obtained as solution of the INS problem, and then considered as a background field during the AD analysis. Given the highly nonlinear nature of the INS equations, this is the most computationally intensive step. Since it can be reasonably assumed that the variation of the underlying meteorological conditions happens on a time scale higher than the one related to the diffusion process, the wind field is considered to be steady and therefore evaluated only once at the beginning of the procedure. Furthermore, MOR techniques are applied to further speed up the wind field evaluation, striving to achieve real-time simulation runs in the medium and long term. On the other hand, the AD process can be described by a linear system of equations, thus making its solution much less computationally expensive.

### A. Simulation of Wind Flow Field

*1) Full-Order Model:* We consider the steady-state incompressible Navier-Stokes equation system [21]. Written in terms of the kinematic viscosity $\nu$, it reads:

$$
\begin{aligned}
-\nu\nabla^2\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla p &= \mathbf{0} & \text{in} \quad \Omega, \\
\nabla\cdot\mathbf{u} &= 0 & \text{in} \quad \Omega, \\
\mathbf{u} &= \mathbf{g}(\mu) & \text{on} \quad \Gamma_D, \\
\nu\frac{\partial\mathbf{u}}{\partial\hat{\mathbf{n}}} - p\hat{\mathbf{n}} &= \mathbf{0} & \text{on} \quad \Gamma_N,
\end{aligned}
\tag{1}
$$

where $\mathbf{u}$ denotes the wind velocity and $p$ the pressure. The boundary conditions are moreover defined on the Dirichlet boundary $\Gamma_D$ and the Neumann boundary $\Gamma_N$. To account for different inflow velocities, a multiplicative factor $\mu \in \mathbb{R}$ is considered in the inhomogeneous Dirichlet boundary condition, so that $(\mathbf{u}, p) = \big(\mathbf{u}(\mu), p(\mu)\big)$ [12]. Moreover, we refer to the maximum Reynolds number as $Re = \|\mathbf{u}_{\max}\| l/\nu$, where $\|\mathbf{u}_{\max}\|$ is the maximum value of the velocity magnitude and $l$ is the characteristic length of the domain.

To facilitate a numerical solution of the wind field, the strong form of the Navier-Stokes equation system described by Eq. (1) is transferred into a weak form and discretized using Taylor-Hood finite elements [21]. The parametric, inhomogeneous Dirichlet boundary condition accounting for varying wind speeds $\mathbf{g}(\mu)$ is treated with a so-called lifting function [22]. For the sake of brevity, we only state a short version of the resulting discretized weak form and refer to the literature for a detailed derivation and reference implementation [12], [23]:

$$
a(\mathbf{u}^h, \mathbf{v}^h; \mu) + b(\mathbf{v}^h, p^h; \mu) + c(\mathbf{u}^h, \mathbf{u}^h, \mathbf{v}^h; \mu) = 0
\tag{2}
$$
$$
b(\mathbf{u}^h, q^h; \mu) = 0.
$$

The linearized form of the problem is finally solved with the Newton algorithm implemented within the FEniCS framework [24].

### B. Simulation of Contaminant Transport

The airborne contaminant transport is modeled with the following time-dependent advection-diffusion equation:

$$
\mathcal{R}(c) := \frac{\partial c}{\partial t} + \mathbf{u}\cdot\nabla c - k\,\Delta c = 0.
\tag{3}
$$

Therein, the scalar unknown $c(\boldsymbol{x}, t)$ represents the contaminant concentration as a function of the spatial coordinates $\boldsymbol{x}$ and time $t$. The advection velocity is a given vector field $\mathbf{u}(\boldsymbol{x})$, obtained as solution of Eq. (1) and in practice approximated with $\mathbf{u}^h(\mu) \approx \mathbf{V}\hat{\mathbf{u}}(\mu)$. Furthermore, the constant diffusion coefficient is denoted by $k$ and the Laplacian operator can be expressed as $\Delta(\cdot) = \nabla\cdot\nabla(\cdot)$, based on the spatial gradient operator $\nabla(\cdot)$.

The resulting initial boundary value problem (IBVP) is defined on the already familiar domain $\Omega$ shown in Fig. 2. Moreover, we denote the derivative of $c$ in the direction of the outward-facing boundary normal as $\partial c/\partial\hat{\mathbf{n}}$. The IBVP states that we require Eq. (3) to hold on $\Omega$, along with a known initial concentration distribution $c_0(\boldsymbol{x})$ and given Dirichlet boundary condition $g$. The complete IBVP reads:

$$
\text{IBVP} \quad
\begin{cases}
\mathcal{R}\big(c(\boldsymbol{x}, t)\big) = 0, & \text{on } \Omega, \\
c(\boldsymbol{x}, t) = c_0(\boldsymbol{x}), & \text{at } t = 0, \\
c(\boldsymbol{x}, t) = g(\boldsymbol{x}, t), & \text{on } \Gamma^D, \\
\partial c/\partial\hat{\mathbf{n}} = 0, & \text{on } \Gamma^N.
\end{cases}
\tag{4}
$$

With suitable finite element function spaces, the well established SUPG-stabilized weak form of the boundary value problem can be derived following [25], [26], [27]. The transient nature of the problem is treated with an implicit Euler time stepping algorithm [21, Equation (10.25)], which results in the following discretized weak form

$$
\begin{aligned}
&\int w^h \cdot c^{n+1}\, d\Omega \\
&+ \Delta t \int w^h \cdot \big(\mathbf{u}\cdot\nabla c^{n+1}\big)\, d\Omega \\
&+ \Delta t \int \nabla w^h \cdot \big(k\nabla c^{n+1}\big)\, d\Omega \\
&+ \int \mathbf{a}\cdot\nabla w^h \cdot \tau \cdot c^{n+1}\, d\Omega \\
&+ \Delta t \int \mathbf{u}\cdot\nabla w^h \cdot \tau \cdot \big(\mathbf{u}\cdot\nabla c^{n+1} - \nabla\cdot k\nabla c^{n+1}\big)\, d\Omega \\
&= \int w^h \cdot c^n\, d\Omega + \int \mathbf{u}\cdot\nabla w^h \cdot \tau \cdot c^n\, d\Omega.
\end{aligned}
\tag{5}
$$

After an initial assembly of the linear system of equation, time-stepping only requires an update of the right hand side and the solution of the resulting system.

## VI. APPLICATION EXAMPLES: MUNICH AND DÜSSELDORF

To showcase the capabilities of our model, two different arrangements of real built environments with different levels of complexity are chosen as numerical examples. The two geometries are both obtained from OSM via an OverPy query, asking for all the buildings located within a rectangle bounded by minimum and maximum values of latitude and longitude. The first setup, referred hereinafter as *Geometry 1*, reproduces a portion of the campus of the University of the Bundeswehr Munich, Germany, and contains a moderate number of buildings; the second set, *Geometry 2*, entails a higher number of buildings and coincides with a portion of a chemical plant located in Düsseldorf, Germany.
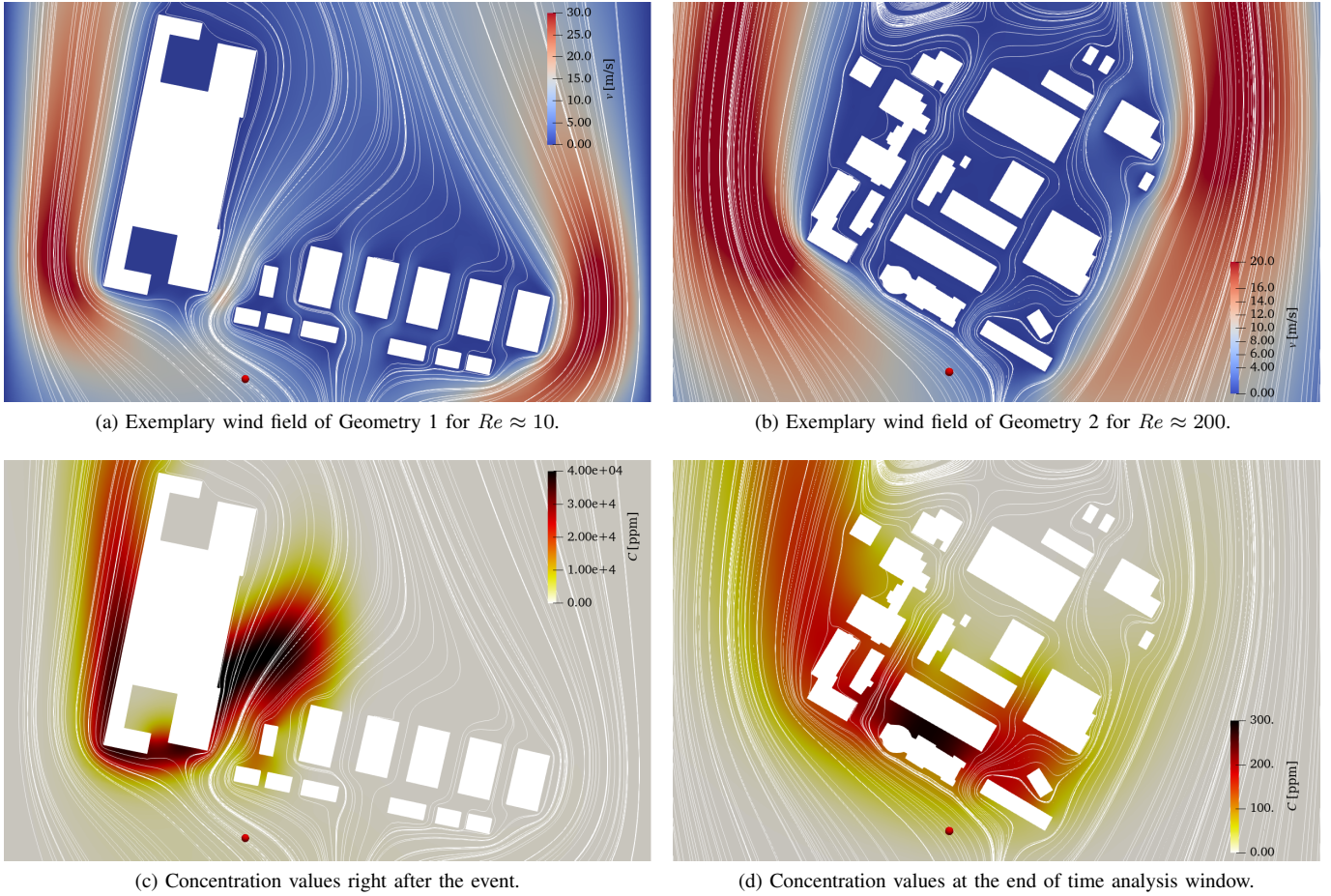
(a) Exemplary wind field of Geometry 1 for $Re \approx 10$.



(b) Exemplary wind field of Geometry 2 for $Re \approx 200$.



(c) Concentration values right after the event.



(d) Concentration values at the end of time analysis window.

Fig. 4: Qualitative results for the two use-cases in terms of wind vector field and concentration distribution in [ppm].

### A. Analysis of computed wind and concentration fields

The building arrangements for both setups can be observed in Fig. 4. In both examples a Neumann boundary condition is applied on the outflow boundary. Moreover, all sides except for inflow and outflow sides are considered no-slip walls with imposed zero velocity. To account for different incoming wind conditions, a parametrized inhomogeneous Dirichlet boundary condition is imposed on the inflow side, resulting in different values of the Reynolds number $Re$ as shown in Tab. I. Figures 4a and 4b illustrate a qualitative representation of the numerical solution of the wind field for the two examples with $Re \approx 10$ and $Re \approx 200$, respectively. The vector field is represented by means of streamlines, while a background contour plot defines the wind magnitude. A qualitative analysis of the streamline pattern allows to verify the plausibility of the evolution of the gas contaminant distribution.

Unlike wind distribution, the AD phenomenon is inherently transient. To this end, an analysis time window is defined, lasting from $t_0 = 0\,\mathrm{s}$ to $t_\mathrm{f} = 5\,\mathrm{s}$ for Geometry 1 and to $t_\mathrm{f} = 50\,\mathrm{s}$ for Geometry 2, with constant time increments in both cases. We study a scenario in which accidents (e.g. explosions) just before $t_0$ lead to an initial distribution of gas

$c_0(\boldsymbol{x})$ in the form of a truncated Gaussian bell. The centers of the initial contaminant concentrations are located at the points highlighted by the coloured mark in Fig. 4. Figure 4c shows the concentration distribution right after the event, where very high values, in the range of $10^4$ ppm can be observed, before the contaminant is rapidly blown away. On the other hand, the snapshot of Fig. 4d is taken at $t = t_\mathrm{f}$, and despite that the final concentration is orders of magnitude lower, it stagnates at potentially critical levels in zones of the domain characterized by low air circulation, qualifying them as the most dangerous parts.

The depicted results confirm that the contaminant transport follows the streamlines of the wind field and respects the influence of the buildings.

### B. Quantitative performance analysis of the reduced-order model

To approach real-time applications, a good performance of the ROM procedure is essential and therefore analyzed in the following. The numerical parameters of the reduction and hyper-reduction are stated in Tab. I along with the Reynolds number ranges corresponding to the inflow velocity considered

TABLE I: Parameters of the presented numerical examples.

| Parameter | | Numerical Examples | |
|---|---|---|---|
| | | *Geometry 1* | *Geometry 2* |
| Reynolds number | | $Re \in [5.5, 515]$ | $Re \in [10, 380]$ |
| Wind direction | | South | |
| Subspace dimension | POD ($N_r$) | 10 | |
| | DEIM ($N_m$) | 20 | |
| Snapshots (POD, DEIM) | | 50 | |
| Test set size (POD, DEIM) | | 20 | |

as parameter in the ROM. Both the speed up and the error are evaluated over an independent test set throughout the parameter space.

Figure 5 collects the ROM performance results of both test cases. The quick decay of the eigenvalues in the snapshot data correlation matrix shows that about 20 RB functions are sufficient to cover the entire system characteristics for both applications. Further evidence comes from the examination of the maximum relative error over the entire computational domain and the entire parameter space, which is shown in Fig. 5b as a function of the RB dimension. The error is acceptable for small dimensions of the RB ($N_r < 10$). When aiming for a maximum deviation from the FOM results of less than 1%, a RB dimension of $N_r = 6$ is already enough.

Figure 5d shows the speed up of the ROM evaluation in comparison to the FOM. As an overall trend, the speed up decreases as the model complexity increases, as can be expected. However, by choosing $N_r = 6$ as explained above, a minimum, average, and maximum speed up of approximately 30, 50, and 110 is achieved, on Geometry 1, for the entire parameter space. Also for the more complex Geometry 2 a significant average speed up of 20 is obtained for $N_r = 6$. In summary, the POD-based ROM procedure has the potential to reduce the computational cost of urban physics simulations and represents a first step towards real-time simulations.

## VII. CONCLUSION AND OUTLOOK

As urban populations grow and critical infrastructures become increasingly threatened, safeguarding against potential disruptions becomes more significant than ever before. Among the possible threats, dangerous gas contaminants dispersion stands out due to its potentially catastrophic impact. The emergence of the DT framework offers promise in analyzing and predicting such events, fostering situational awareness and crisis management. This study proposes a computational framework for analyzing airborne contaminant dispersion, leveraging on the automatic generation of computational domains and solution processes. Employing FEM-based CFD to obtain the numerical solutions of the systems of equations governing the phenomena under examination, this approach appears suitable in improving situational awareness and real-time decision support within, e.g., evacuation scenarios.

The framework was tested with wind field and concentration computations on two real world geometries. In both examples, an explosion-like accident with an instant leakage was studied.

However, with little additional effort a continuous source can be considered as well within the same framework. Furthermore, the developed workflow is easily extended to further urban physics problem classes, such as simulating the temperature field in built environments. Intrusive MOR methods were explored to enhance computational efficiency and obtain a quantitative assessment on the achievable accuracy levels. While the solution accuracy was found to be sufficient, the speed up needs further improvement.

The development and extension of the current virtual replica involves further necessary steps to add to the model necessary features. Amongst them, the most important is the substitution of hard-coded inputs with real-time data coming from sensors and detection control units. The increased simulation time associated to a continuous data stream will be also carefully investigated and optimized, being this a potential bottleneck. Given the proposed scope of performing accurate real-time simulations, MOR will be extended to the transient part of the solution step as well, and the overall time required by a single time step will be analyzed to guarantee the absence of over-run and lags. These topics will be thoroughly investigated in future research and their features added to the current model, aiming at the declared final goal of instantiating a functional DT.

## REFERENCES

[1] H. Boyes and T. Watson, "Digital twins: An analysis framework and open issues," *Computers in Industry*, vol. 143, p. 103763, Dec. 2022. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0166361522001609

[2] J. Eyre, S. Hyde, D. Walker, S. Ojo, O. Hayes, R. Hartley, R. Scott, and J. Bray, "Untangling the requirements of a Digital Twin," 2020. [Online]. Available: https://rgdoi.net/10.13140/RG.2.2.28776.96001

[3] E. Brucherseifer, H. Winter, A. Mentges, M. Mühlhäuser, and M. Hellmann, "Digital Twin conceptual framework for improving critical infrastructure resilience," *at - Automatisierungstechnik*, vol. 69, no. 12, pp. 1062–1080, Dec. 2021, publisher: De Gruyter (O). [Online]. Available: https://www.degruyter.com/document/doi/10.1515/auto-2021-0104/html

[4] K. N. Abdrakhmanova, A. V. Fedosov, K. R. Idrisova, N. K. Abdrakhmanov, and R. R. Valeeva, "Review of modern software complexes and digital twin concept for forecasting emergency situations in oil and gas industry," *IOP Conference Series: Materials Science and Engineering*, vol. 862, no. 3, p. 032078, May 2020. [Online]. Available: https://iopscience.iop.org/article/10.1088/1757-899X/862/3/032078

[5] V. Nagarajan, N. Fougere, E. M. Schechter-Perkins, W. E. Baker, A. Mann, J. Jilesen, and Z. Altawil, "Predicting Contamination Spread Inside a Hospital Breakroom with Multiple Occupants Using High Fidelity Computational Fluid Dynamics Simulation on a Virtual Twin," *Sustainability*, vol. 15, no. 15, p. 11804, Aug. 2023. [Online]. Available: https://www.mdpi.com/2071-1050/15/15/11804

[6] B. Blocken, "Computational Fluid Dynamics for urban physics: Importance, scales, possibilities, limitations and ten tips and tricks towards accurate and reliable simulations," *Building and Environment*, vol. 91, pp. 219–245, Sep. 2015. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0360132315000724
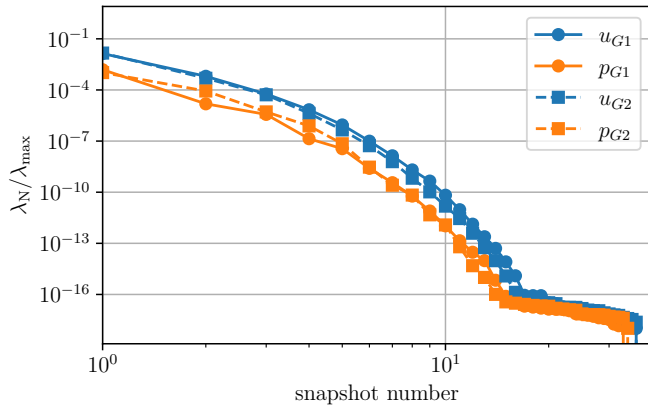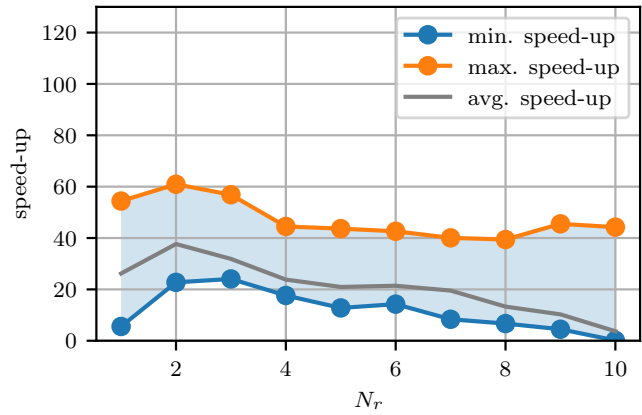
(a) Normalized eigenvalues of snapshot correlation matrix of POD procedure.
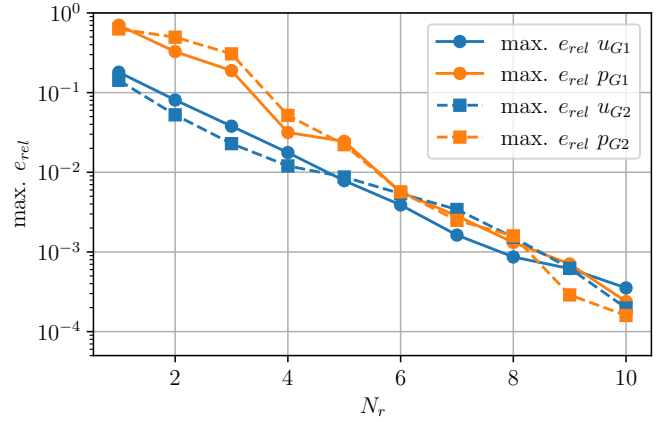


(b) Maximum relative error between ROM and FOM solutions.
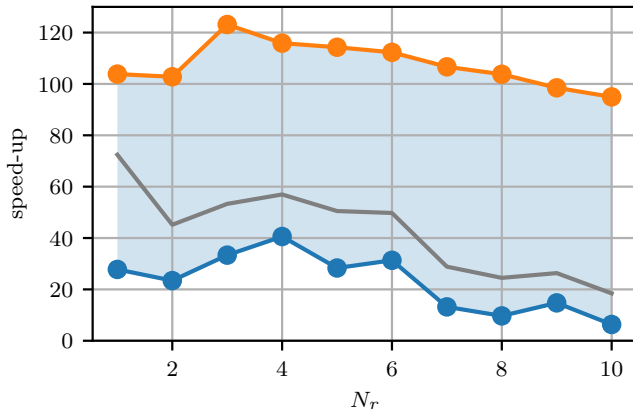


(c) Speed up for Geometry 1 as function of the RB dimensions $N_r$.



(d) Speed up for Geometry 2 as function of the RB dimensions $N_r$.

Fig. 5: Performance analysis of the reduced-order model (ROM) for both geometries.

[7] E. Shahat, C. T. Hyun, and C. Yeom, "City Digital Twin Potentials: A Review and Research Agenda," *Sustainability*, vol. 13, no. 6, p. 3386, Mar. 2021. [Online]. Available: https://www.mdpi.com/2071-1050/13/6/3386

[8] G. Caprari, G. Castelli, M. Montuori, M. Camardelli, and R. Malvezzi, "Digital Twin for Urban Planning in the Green Deal Era: A State of the Art and Future Perspectives," *Sustainability*, vol. 14, no. 10, p. 6263, May 2022. [Online]. Available: https://www.mdpi.com/2071-1050/14/10/6263

[9] S. Murakami and A. Mochida, "Three-dimensional numerical simulation of turbulent flow around buildings using the $k - \varepsilon$ turbulence model," *Building and Environment*, vol. 24, no. 1, pp. 51–64, 1989. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0360132389900164

[10] Y. Tominaga, A. Mochida, R. Yoshie, H. Kataoka, T. Nozu, M. Yoshikawa, and T. Shirasawa, "AIJ guidelines for practical applications of CFD to pedestrian wind environment around buildings," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 96, no. 10-11, pp. 1749–1761, Oct. 2008. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0167610508000445

[11] T. Van Hooff and B. Blocken, "Coupled urban wind flow and indoor natural ventilation modelling on a high-resolution grid: A case study for the Amsterdam ArenA stadium," *Environmental Modelling & Software*, vol. 25, no. 1, pp. 51–65, Jan. 2010. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1364815209001790

[12] F. Ballarin, A. Manzoni, A. Quarteroni, and G. Rozza, "Supremizer stabilization of pod–galerkin approximation of parametrized steady incompressible navier–stokes equations," *International Journal for Numerical Methods in Engineering*, vol. 102, no. 5, pp. 1136–1161, 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.4772

[13] L. Cicci, S. Fresca, S. Pagani, A. Manzoni, and A. Quarteroni, "Projection-based reduced order models for parameterized nonlinear time-dependent problems arising in cardiac mechanics," *Mathematics in Engineering*, vol. 5, no. 2, pp. 1–38, 2022.

[14] G. Rozza, G. Stabile, and F. Ballarin, *Advanced reduced order methods and applications in computational fluid dynamics*, ser. Computational science and engineering. Philadelphia, PA: siam Society for Industrial and Applied Mathematics, 2023, vol. 27. [Online]. Available: https://zbmath.org/7647935

[15] C. Geuzaine and J.-F. Remacle, "Gmsh: A 3–d finite element mesh generator with built–in pre– and post–processing facilities," *International Journal for Numerical Methods in Engineering*, vol. 79, no. 11, pp. 1309–1331, 2009. [Online]. Available: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.2579

[16] M. S. Alnaes, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. N. Richardson, J. Ring, M. E. Rognes, and G. N. Wells, "The FEniCS project version 1.5," *Archive of Numerical Software*, vol. 3, 2015.

[17] OpenStreetMap contributors, "Planet dump retrieved from https://planet.osm.org ," https://www.openstreetmap.org, 2017.

[18] M. Raifer, "Overpass turbo," https://github.com/tyrasd/overpass-turbo, 2012.

[19] J. M. Stürmer, M. Graumann, and T. Koch, "Demonstrating automated generation of simulation models from engineering diagrams," in

*2023 International Conference on Machine Learning and Applications (ICMLA).* IEEE, Dec. 2023.

[20] S. Rebay, "Efficient unstructured mesh generation by means of delaunay triangulation and bowyer-watson algorithm," *Journal of Computational Physics*, vol. 106, no. 1, pp. 125–138, 1993.

[21] H. C. Elman, D. J. Silvester, and A. J. Wathen, *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*, 2nd ed., ser. Numerical mathematics and scientific computation. Oxford: Oxford University Press, 2014.

[22] F. Key, M. von Danwitz, F. Ballarin, and G. Rozza, "Model order reduction for deforming domain problems in a time–continuous space–time setting," *International Journal for Numerical Methods in Engineering*, vol. 124, no. 23, pp. 5125–5150, 2023.

[23] F. Ballarin, A. Sartori, and G. Rozza, "Rbnics – reduced order modelling in fenics," 2015. [Online]. Available: https://www.rbnicsproject.org/

[24] A. Logg, K.-A. Mardal, G. N. Wells *et al.*, *Automated Solution of Differential Equations by the Finite Element Method*, A. Logg, K.-A. Mardal, and G. N. Wells, Eds. Springer, 2012.

[25] A. N. Brooks and T. J. Hughes, "Streamline upwind/petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 32, no. 1-3, pp. 199–259, 1982. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0045782582900718

[26] M. Behr, "Simplex space–time meshes in finite element simulations," *International Journal for Numerical Methods in Fluids*, vol. 57, no. 9, pp. 1421–1434, 2008.

[27] K. Key, M. R. Abdelmalik, S. Elgeti, T. J. Hughes, and F. A. Baidoo, "Finite element and isogeometric stabilized methods for the advection-diffusion-reaction equation," *Computer Methods in Applied Mechanics and Engineering*, vol. 417, p. 116354, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045782523004784