Report - CFD (fast methods)

PROJECT P3VENTI – HANDELINGSPERSPECTIEF GEBRUIK EN SYSTEEMCONFIGURATIE (D4.2)

(project phase 2025)

Date	Our reference	Version
September 11 2025	BE2025-2055380	1.0

Authors
Twan van Hooff
Marcel Loomans
Eugene Mamulova

EINDHOVEN
UNIVERSITY OF
TECHNOLOGY

Built Environment
Building Physics and Services

T +31 (0)40 247 2400
m.g.l.c.loomans@tue.nl

www.tue.nl

Contents

1. General introduction

1.1 Computational methods

Computational fluid dynamics (CFD) is currently the method used to simulate the airflow in a chamber (e.g., [1]). Although many CFD methods currently exist, Reynolds-averaged Navier-Stokes (RANS) is often used due to its relatively low computational costs compared to, for example, large-eddy simulation (LES) (e.g. [2]), combined with the ability to provide a sufficiently accurate prediction of the most common flow problems. In the further report, RANS is called conventional CFD (convCFD) because it is more commonly used for simulating indoor environments than the other computational methods. Despite its popularity, convCFD is still relatively computationally expensive (e.g. [3]) and prone to user input [4]. As a result, engineers may choose not to perform convCFD simulations in practice. It is likely that, regardless of the accuracy of the method, the computational costs and the way the user experiences the simulation process will influence its acceptance in (engineering) practice. It is also possible that methods with a lower accuracy are sufficient for certain technical applications, such as calculating average concentrations of aerosols in the user zone. As a result, research into the performance of so-called "fast computational methods" is important for both academic and practical applications. Within P3Venti, the calculation speed, accuracy and user experience of alternative computational methods were examined, and compared with convCFD. The goal is to design a framework for evaluating the speed (SP), accuracy (ACC), and user experience (UX) of different methods, and to demonstrate this using multiple benchmark cases.

1.2 Fast computational methods

This part of the study looked at methods that are faster than convCFD [5]. This subset of methods is called "fast computational methods" (FCMs). Previous studies (e.g. [6, 7]) have assessed FCMs of interest for modeling air flows and heat and mass transport in spaces. For an up-to-date overview of relevant FCMs, the reviews can be supplemented with recent machine learning studies (e.g. [8, 9]). FCMs applied to indoor airflows are listed in Table 1 and divided based on whether they assume total or partial mixing (non-CFD methods), originate from convCFD (RANS-CFD methods), are based on other transport equations (non-RANS-CFD methods), or are based on machine learning techniques (ML methods). Non-RANS-CFD methods are currently excluded from this study, despite promising results using Lattice Boltzmann methods (e.g. [10]) and fast fluid dynamics (e.g. [11]) for airflows in the built environment. Finally, despite interesting work done by the machine learning community in areas such as data-driven surrogate modeling (e.g. [12]), reduced-order modeling (e.g. [13]) and on physics-based learning (e.g. [14]), the assessment framework presented in this study does not currently allow for the evaluation of methods that require a (machine) learning phase. In the future, the framework should be expanded to allow for non-RANS CFD and ML methods.

Table 1: Non-exhaustive list of FCMs for indoor modeling, based on [5-8].

Non-CFD	RANS-CFD	Non-RANS-CFD	ML
Mixing models	Coarse-grid CFD	Lattice Boltzmann methods	Reduced-order models
Nodal models	Zero-equation CFD	Fast fluid dynamics	Surrogate models
Zonal models	One-equation CFD		Hybrid ML-CFD models
	GPU-based CFD		Physics-based models

This particular study focuses exclusively on RANS-CFD methods, namely coarse grid CFD (cCFD), zero-equation modeling (zCFD), one-equation modeling (oCFD), and GPU-based CFD (gpuCFD), using the advantages and disadvantages of grid manipulation, simplified turbulence modeling, and parallel computing.

1.3 Literature study

An extensive but non-exhaustive literature review of the chosen FCMs was conducted, with the aim of determining (a) how broadly each FCM has been studied, and (b) how, if at all, existing studies relate to the speed, accuracy, and usefulness of these methods. The literature search was conducted using Scopus and used mandatory keywords such as "indoor" and "RANS", as well as the name of the respective method (e.g., "one-equation" and "Spalart-Allmaras" for oCFD). The goal of the query was to find cases where one or more FCMs were applied in the context of indoor environment simulations. All other studies, such as non-CFD, non-RANS-CFD, ML and studies for an outdoor environment, were excluded. Table 2 provides an overview of the 20 studies selected with regard to the performance of the method (SP, ACC and UX).

The literature review shows that some methods have been studied more broadly than others. zCFD was found to be the most common in the literature (11 studies) [24-34], followed by cCFD [16-19, 29, 31, 34] and oCFD [20-23, 30, 32, 33], with only one study on gpuCFD [15]. It should be noted that the search may have affected the search results. GPU programming poses a significant challenge, which would explain why gpuCFD literature is scarce and more fundamental in nature. Second, the study shows that FCMs are often compared to convCFD [15, 17-26, 29-34]. Interestingly, only 6 studies include more than one FCM [29-34] and none examine more than two FCMs.

Table 2: Overview of fast computational methods, resulting from the literature review.

		convCFD	cCFD	oCFD	zCFD	gpuCFD	Remarks
[15]	SP	x				x	gpuCFD offers a 588-850x speed-up compared to convCFD (5 CPU cores) with good accuracy (max. velocity error < 1.3%).
	ACC					x	
	UX						
[16]	SP						The difference in max. velocity between cCFD and convCFD is 28.5% for a simple case.
	ACC		x				
	UX						
[17]	SP						convCFD and cCFD yield similar velocity and temperature profiles for a simple 2D case. UX: discussion on convCFD and cCFD meshing differences.
	ACC	x	x				
	UX	x	x				
[18]	SP	x	x				cCFD is 16-227x faster than convCFD for a 1-3x decrease in accuracy for 2D and 3D cases.
	ACC	x	x				
	UX						
[19]	SP	x	x				cCFD is two orders of magnitude faster at comparable accuracy based on 2D and 3D cases. UX: meshing workflow is provided.
	ACC	x	x				
	UX	x	x				
[20]	SP						oCFD (error = 5%) performs similarly to convCFD (error = 3.38% - 10.3%) for a simple 3D case.
	ACC	x		x			
	UX						
[21]	SP						oCFD more accurate (error = 7%) than convCFD (error = 13.3% - 26.5%) for a 3D moving grid.
	ACC	x		x			
	UX						
[22]	SP						oCFD and convCFD predict jet behavior with similar accuracy (error = 15%).
	ACC	x		x			
	UX						
[23]	SP						convCFD yields more accurate temperature predictions (error = 3.8% - 9.3%) than oCFD (error = 0.2% - 10.7%) for 3D case.
	ACC	x		x			
	UX						
	SP	x			x		

[24]	ACC	x			x	zCFD is 2-3 orders of magnitude faster than convCFD, yielding similar temperature profiles for a complex 3D case. UX: discussion on software differences.
	UX	x			x	
[25]	SP					zCFD and convCFD velocity and SF_6 concentration profiles follow the experimental trend (ACC not quantified) for a complex 3D case.
	ACC	x			x	
	UX					
[26]	SP					zCFD (error = 0.17, 0.72 %) is less accurate than convCFD (error = 0.17, 0.06 %) at predicting heat transfer for a complex case.
	ACC	x			x	
	UX					
[27]	SP					zCFD can predict velocity (error < 0.07 m/s) and temperature (error < 1°C) for a complex case.
	ACC				x	
	UX					
[28]	SP					zCFD can predict the flow field for a simple 2D case (error ≤ 0.08 m/s and ≤ 0.8°C) and 3D case (error ≤ 0.05 m/s and ≤ 1.5°C).
	ACC				x	
	UX					
[29]	SP	x	x		x	zCFD (combined with cCFD) is 6-8x faster than convCFD for a complex 3D case. Good velocity and temperature agreement. UX: discussion on software limits.
	ACC				x	
	UX	x				
[30]	SP	x		x	x	zCFD, oCFD and convCFD perform similarly in terms of speed (t = 2.5-3.5 s/it) and accuracy (RMSE = 0.2-3.1 m/s) for a complex 3D case.
	ACC	x		x	x	
	UX					
[31]	SP	x	x		x	zCFD is 10x faster than convCFD and over 1000x faster when combined with cCFD. Accuracy depends on case (not quantified).
	ACC	x	x		x	
	UX					
[32]	SP					convCFD and oCFD contours resemble measurements (ACC not quantified) for a simple case, while zCFD underperforms. UX: discussion on meshing workflow.
	ACC	x		x	x	
	UX	x				
[33]	SP					convCFD and oCFD approach measurements (ACC not quantified) for a simple case, while zCFD underperforms.
	ACC	x		x	x	
	UX					
[34]	SP	x	x		x	convCFD, cCFD and zCFD can predict flow characteristics (error below 15%) in real-time (time ratio = 1). Very coarse grids underperform (ACC).
	ACC	x	x		x	
	UX					

Table 2 illustrates the speed, accuracy and user experience of different methods as reported in the different studies. First, it is clear that UX is not systematically evaluated in the literature and that UX assessment is currently limited to short discussions. Second, the literature shows that ACC and SP are quantified in different ways, making it difficult to compare the performance of the methods in different studies. A general observation could be made that FCMs are able to speed up the simulations by 1-3 orders of magnitude [15, 18, 19, 24, 29, 30, 31, 34], but it is not clear how the methods perform relative to each other. Finally, the overall package of UX-ACC-SP has not been well studied (3 studies). It is clear that an assessment framework addressing all three aspects is lacking, although Phan et al. [30] raised the need for such a framework through their SP-ACC assessment.

More information on the literature studied can be found in:

Mamulova, E., Loomans, M., & van Hooff, T. (2025).RANS-based fast computational methods for indoor flows: a framework-driven performance assessment for a simple benchmark. Developments in the Built Environment,23, Article 100716.[*https://doi.org/10.1016/j.dibe.2025.100716*](https://doi.org/10.1016/j.dibe.2025.100716)

2 Methods

2.1 Fast methods

2.1.1. Conventional CFD (convCFD)

In this study, conventional CFD refers to the discretized, stationary (steady) Reynolds-averaged Navier-Stokes (RANS) equations. In practice, this is a commonly used form of CFD [35], which can be used as long as stationary conditions are considered representative. The convCFD software used in this study is ANSYS Fluent 2021 R1 [36].

2.1.2. Coarse-grid CFD (cCFD)

Coarse-grid CFD is based on the same RANS formulation as convCFD, with the only obvious difference being the resolution of the computational grid. The method is based on the hypothesis that numerical accuracy is not a global feature and that similar results can be obtained at low grid resolutions, as demonstrated in [37]. In other words, cCFD has the potential to produce results similar to those obtained with convCFD for a fraction of the computational cost, due to its reduced number of computational cells. An increase in discretization/rounding errors and numerical (or false) diffusion will be present for very coarse grids, which may negatively affect the accuracy of the simulation. It should be noted that numerical (false) diffusion is present when the flow is not aligned with the grid, as is often the case with indoor airflows, and can be mitigated using higher grid resolutions and higher-order discretization schemes. Currently, there is no single go-to grid procedure for cCFD, as shown by the literature review. In the interest of studying the impact of grid resolution, the convCFD grid is coarsened in this study while maintaining the same grid refinement ratios in all directions.

2.1.3. One-equation modeling (oCFD)

One-equation modeling is also a RANS method, but the difference between convCFD and oCFD lies in the turbulence model: namely the number of transport equations. In this study, the one-equation model refers to the Spalart–Allmaras formulation, which uses a single transport equation to model turbulence in the form of the eddy viscosity (ν_t) [38]. Other one-equation models are the one-equation model of Prandtl [39]. This simplification in turbulence modeling eases the computational burden of solving two transport equations, but is also expected to have a negative effect on the accuracy of the simulation. In this study, the oCFD simulations are performed using the convCFD grid.

2.1.4. Zero-equation modeling (zCFD)

Zero-equation modeling is also known as the mixing-length model. zCFD is also a RANS method, but does not use transport equations for modeling turbulence [40]. Instead, the method models the eddy viscosity (ν_t) as a function of a given mixing length (l_m). This method is computationally advantageous, but is not suitable for flows with a velocity gradient in multiple directions (e.g. recirculation flows) due to the mixture length assumption. Nevertheless, zCFD can still be suitable for simple cases [41], making it worthwhile to include this method in the research. In this study, the zCFD simulations are also performed using the convCFD grid.

2.1.5. GPU-based CFD (gpuCFD)

In this study, gpuCFD refers to the GPU implementation of convCFD. As the literature review shows, the GPU is better suited to perform a very large number of computational tasks in parallel than the central processing unit (CPU). This particular GPU implementation benefits from the use of a structured computational grid, i.e., an auto-generated, voxel-based computational grid with "immersed boundaries," similar to [42]. The number of voxels generated is determined by the amount of dedicated memory (VRAM) available.

The gpuCFD method used in this study was performed using ANSYS Discovery 2022 R2 [43]. As shown by the literature review, gpuCFD is still challenging to implement, resulting in a highly competitive market. Due to the proprietary aspect of the software, certain software details are not disclosed. Unlike other CFD software, it is designed specifically for technical purposes, making it interesting to examine its speed, accuracy, and user experience.

2.2 Assessment framework

The assessment framework proposed in this study is summarized in Figure 1. The workflow starts with a validation phase in which the simulation results are compared to benchmark data. In the event that benchmark data is not available, the user may want to consider using high-fidelity CFD data instead. The next step is to select performance metrics for ACC, SP, UX. During this phase, performance thresholds are defined based on how accurate, fast, and user friendly the methods are expected to be for a given application. In the final phase, an SP-ACC-UX evaluation is carried out on the basis of the results from phase 2. The final assessment provides information on the trade-off between SP-ACC-UX that can be used to compare different methods. An implementation of this assessment framework is shown below.

More information on the assessment framework can be found in:

Mamulova, E., Loomans, M., & van Hooff, T. (2025).RANS-based fast computational methods for indoor flows: a framework-driven performance assessment for a simple benchmark. Developments in the Built Environment,23, Article 100716.[*https://doi.org/10.1016/j.dibe.2025.100716*](https://doi.org/10.1016/j.dibe.2025.100716)

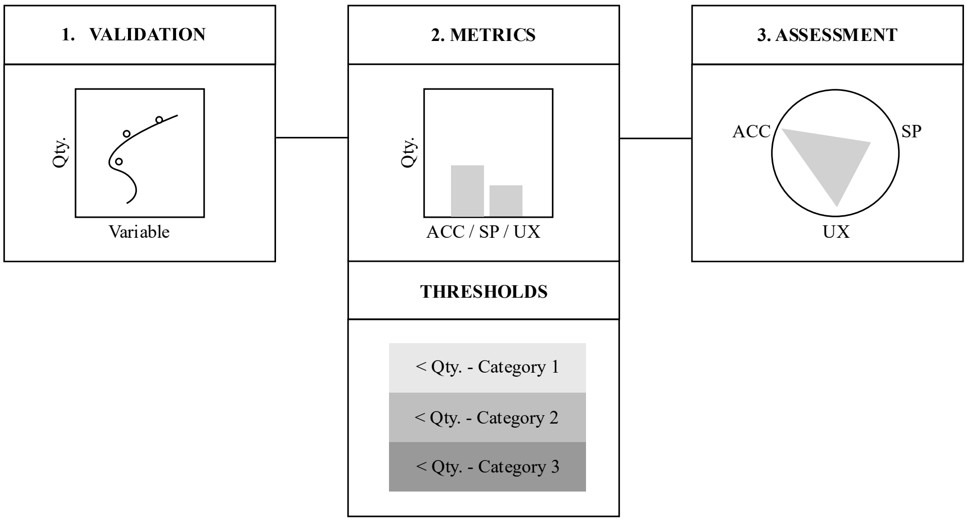

Figure 1: Schematic representation of the proposed SP-ACC-UX assessment framework.

2.3 Case studies

2.3.1. Case study 1

This research uses a simple 3D case as a benchmark (Fig. 2). The case is based on measurements from the International Energy Agency (IEA) annex on airflow patterns in buildings [44]. The measurement setup consists of a generic space, with dimensions equal to 9 m (*L*), 3 m (*W*) and 3 m (*H*). The ventilation supply is located near the ceiling and the ventilation outlet is located near the floor on the opposite wall, both over the full width of the room (l = 3 m). The air is supplied (U_0) at 0.455 m/s, perpendicular to the supply surface. The turbulence intensity at the inlet (*TI0*) is 4%. The airflow pattern consists of a supply jet and subsequent recirculation in the room (mixed ventilation). The measurements were made for a Reynolds number (*Re*) equal to 5,000, based on the height of the supply opening (h) equal to 0.168 m and a kinematic viscosity (v) of 15.3 $\times 10^{-6}$ m^2/s, at an air temperature (T_a) of 20 °C. The outlet height (t) is equal to 0.480 m. The measurement data used for validation will include the mean x-velocity component (U_x) and the turbulence intensity, defined in [48] as the square root of the x-velocity fluctuations ($U_{x,RMS}$), divided by the inlet velocity ($TI = U_{x,RMS}/U_x$). The data consists of 25 measurement points at x = H and 25 measurement points at x = 2H, resulting in a total of 50 values for U_x and 50 values for *TI.*

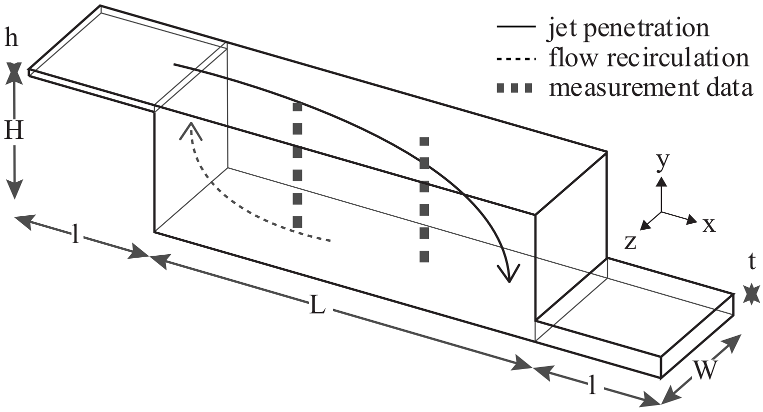

Figure 2: Geometry of the space with the two lines where the evaluation was performed (z = 0.5W).

2.3.2. Case study 2

Case study 2 is based on research by Li et al. [45], who carried out a series of temperature and velocity measurements in a room ventilated by displacement ventilation (3.6 m × 2.75 m × 4.2 m (X × Y × Z)). The geometry consists of a space with one inlet, one outlet and a heat source in the room, as shown in Figure 3. Additional information on the dimensions is provided by Li et al. [45].

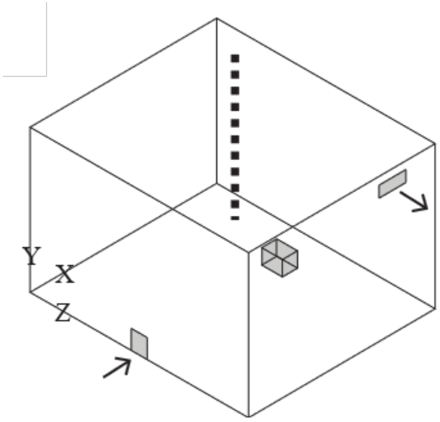

Figure 3: Geometry of the area under investigation. The marked areas indicate the surface of the inlet and outlet openings.

Configuration B1 is chosen from several configurations available, as it is reported in the most detail by Li et al. [45]. The measurements are carried out for an Archimedes number $Ar = 21$, specific flow rate $n = 1$ h^{-1} and a heat source of $Q = 300$ W. A total of 30 temperature measurements are available at $x = 3.2$ m and $z = 0.75$ m. The position of the measuring pole is shown in Figure 3.

2.3.3. Case study 3

Case study 3 covers a more realistic space and more realistic conditions. This room is identical to the test facility as built at TU/e (unit BPS) and is shown in Figure 4. In this room, TNO has carried out measurements of the air velocities, air temperatures, and particle concentrations. There are two separate experimental datasets available for an ACH = 3 h^{-1}. The first dataset consists of velocity and temperature measurements (VT) and the second dataset consists of particle concentration data (PE) for six particle sizes (0.3 µm, 0.5 µm, 0.7 µm, 1 µm, 2 µm and 5 µm). There are 9 heat sources in the room. Supply is through two openings in the ceiling (yellow in Figure 4) and exhaust is through two openings in the ceiling, x m away from the supply openings. This case therefore involves a typical mixed ventilation situation, with thermal effects as a result of the 9 heat sources. Detailed information is available in Peng et al. [46].

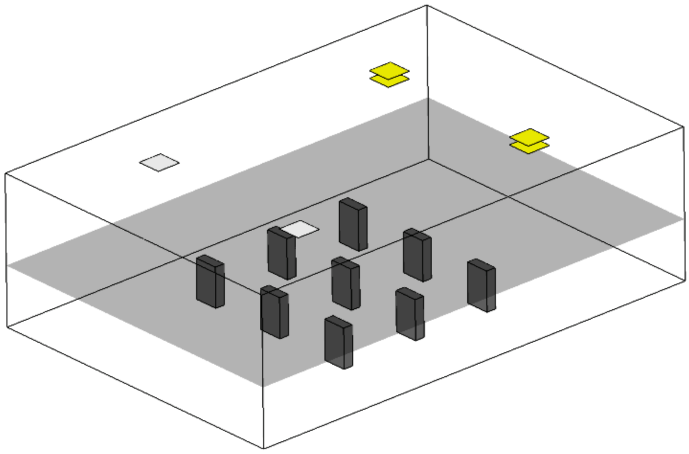

Figure 4: Geometry of the test facility room as present in the laboratory of the unit BPS of TU/e.

3 Case study 1

3.1 Description of the simulations

3.1.1. Conventional CFD (convCFD)

The computational domain is discretized in hexahedral computing cells. Table 3 provides an overview of the characteristics of the coarse, basic and fine computational grids. The coarse grid is the coarsest possible grid that still satisfies the $y^*_{mean} < 5$ condition for low-Reynolds number modeling, as shown in Table 3.

Table 3: Overview of the computational grids.

Grid	Cell count	y^*_{max}	y^*_{mean}
Coarse	265,650	6	4
Basic	766,250	4	3
Fine	2,211,863	3	2

The inlet speed (U_0) is set to 0.455 m/s and the static overpressure at the outlet (P_{out}) is set to 0 Pa. The boundary conditions on the walls are no-slip, i.e. 0 m/s on the surface. The RANS equations are closed via the RNG k-ε two-equation turbulence model [47] and low-Reynolds number modeling [48] is used as wall treatment. In other words, convCFD models turbulence using two transport equations, in this case the turbulent kinetic energy (k) and the turbulent dissipation rate (ε). At the inlet, the turbulent kinetic energy (k_0) and the turbulent dissipation velocity (ε_0) are set to 5.0 10^{-4} m^2/s^2 and 6.6 10^{-4} m^2/s^3 respectively, based on Eq. (1) and Eq. (2), where the turbulent length scale (l_0) is taken as 10% of the opening height [49].

$$k_0 = 1.5(0.04 \cdot U_0)^2 \tag{1}$$

$$\varepsilon_0 = \frac{k_0^{1.5}}{l_0} \tag{2}$$

Pressure and velocity are coupled using the SIMPLE algorithm and pressure is interpolated using second-order discretization. The remaining terms are computed using the second-order upwind discretization scheme. The under-relaxation factors for momentum, pressure, density, body forces, turbulent kinetic energy, turbulent dissipation rate and turbulent viscosity are lowered to 0.3, 0.3, 0.6, 0.6, 0.5, 0.5 and 0.6, respectively. Default values of 0.3, 0.7, 1, 1, 0.8, 0.8 and 1 were insufficient for numerical stability. Although the under-relaxation factors should not affect the simulation results, they do affect the convergence rate and are therefore reported.

A grid-sensitivity analysis was performed to select a computational grid that is sufficient for an almost grid-independent result. The three calculation computational grids consist of 265,650 cells (coarse grid; see Fig. 5), 766,250 cells (base grid) and 2,211,863 cells (fine grid). The analysis focuses on the speed along vertical lines at x = H and x = 2H. The results are shown in Figure 6. The grid-convergence index (GCI) is calculated using Eq. (3), where the linear grid refinement factor r is equal to 2, the formal order of accuracy p is equal to 2, and the safety factor F_s is equal to 1.25 for a sensitivity analysis involving three or more grids [54].

$$GCI_{coarse} = F_s \frac{r^p[(U_{x,coarse} - U_{x,basic})/U_0]}{1-r^p} \tag{3}$$

The maximum and mean GCI_{basic} values at x = H are 0.05 and 0.08, and the maximum and mean GCI_{basic} values at X = 2H are 0.02 and 0.06, respectively. Meanwhile, the maximum and mean GCI_{coarse} values at x = H are equal to 0.02 and 0.07, and the maximum and mean GCI_{coarse} values at x = 2H are 0.01 and 0.07, respectively (Fig. 6). The results show that the coarse grid produces almost grid-independent results, so it is ultimately chosen. The coarse grid is also used for oCFD and zCFD.

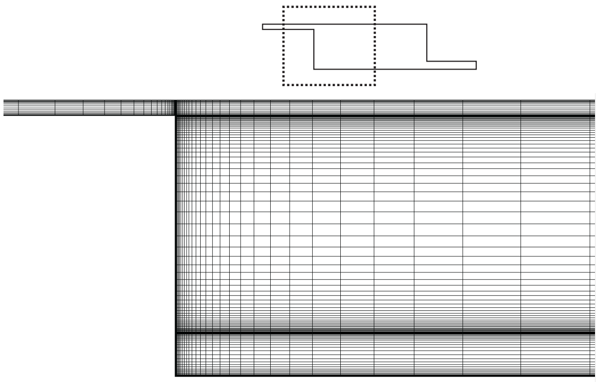

Figure 5: Two-dimensional, close-up view of the convCFD computational grid. The grid consists of 265,650 cells (coarse grid).

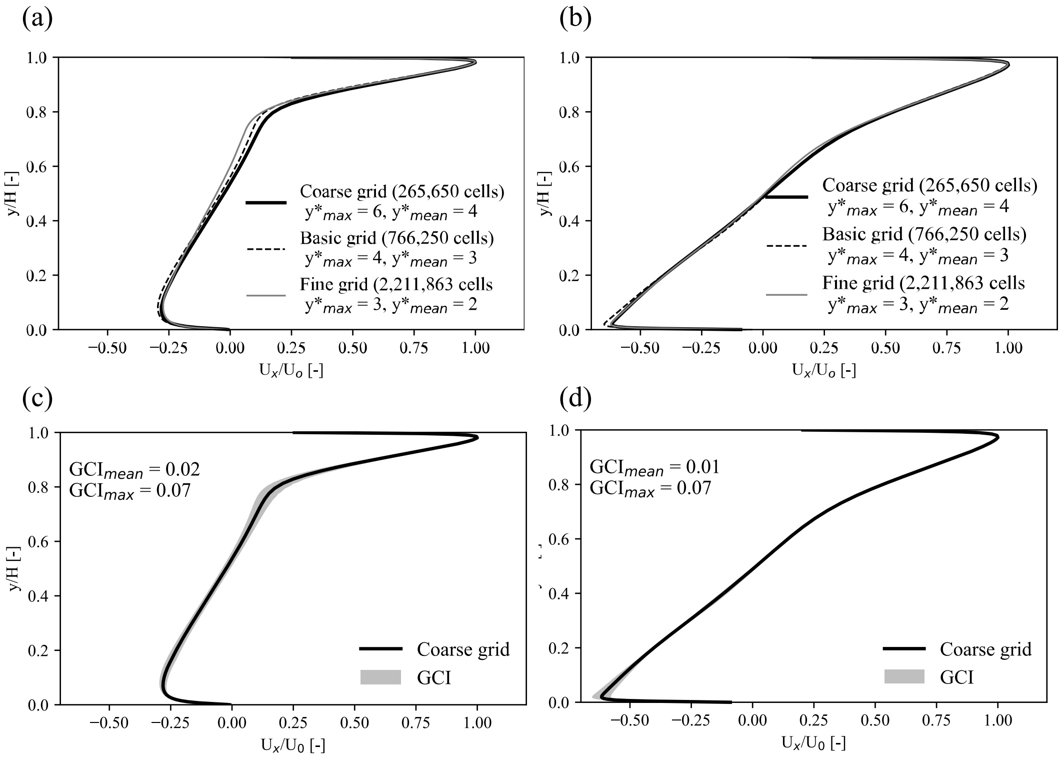

Figure 6: Grid-sensitivity analysis. (a) *U_x/U_0* at x = H. (b) *U_x/U_0* at x = 2H. (c) *GCI_{coarse}* at x = H. (d) *GCI_{coarse}* at x = 2H.

3.1.2. Coarse-grid CFD (cCFD)

The convCFD grid has been robbed from 265,650 computational cells to 4,410 computational cells, while maintaining the same grid refinement ratios in all directions. The grid for cCFD is shown in Figure 7.

The cCFD simulations in this study are also performed in ANSYS Fluent 2021 R1. The boundary conditions used in this simulation are equivalent to those of the convCFD model (see section 3.1.1). The turbulent boundary conditions are also the same, with the exception of the wall treatment. Standard wall features are used instead of low-Reynolds-number modeling, because in this case, the condition $y^* < 5$ is not met. Pressure and velocity are coupled using the Coupled scheme and pressure is interpolated via a second-order discretisation scheme. The remaining conditions are calculated using the second-order upwind scheme.

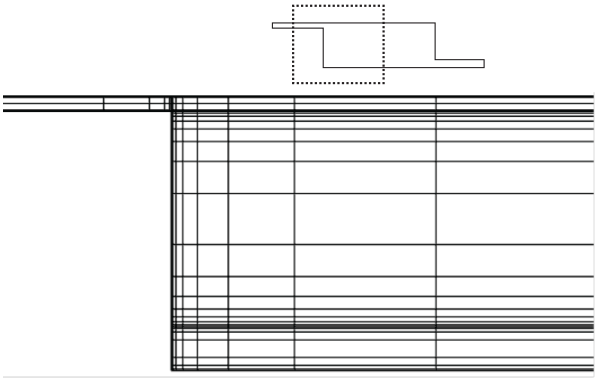

Figure 7: Two-dimensional, close-up view of the cCFD computational grid.

3.1.3. One-equation modeling (oCFD)

The oCFD software used in this study is ANSYS Fluent 2021 R1. The boundary conditions used for oCFD are therefore similar to those of convCFD, but they differ in that the turbulent boundary conditions at the inlet and outlet are determined by the turbulence intensity (*TI)* and hydraulic diameter (D_h), the former being calculated using Eq. (4) and equals 6% for *Re* = 5,000 [58]. The hydraulic diameter D_h is calculated using Eq. (5), where *W* and *H* correspond to the width and height of the rectangular opening. $D_{h,0}$ and $D_{h,out}$ are equal to 0.32 m and 0.83 m, respectively.

$$TI = 0.16 \times Re^{(-0.125)} \tag{4}$$

$$D_h = \frac{2WH}{(W+H)} \tag{5}$$

The pressure-velocity coupling scheme is SIMPLE and all discretization schemes are set to the second-order upwind.

3.1.4. Zero-equation modeling (zCFD)

The zCFD software used in this study is ANSYS Fluent 2021 R1. The boundary conditions are equivalent to those of the convCFD model, while some model parameters are determined by the mixing length model (e.g., near-wall treatment). No turbulent boundary conditions are used. The pressure-velocity coupling scheme is SIMPLE and all discretization schemes are set to the second order. Steady statistics are turned on once the residuals are stable and convergence is determined based on the percentage deviation between two iterations, 1,000 iterations apart.

3.1.5. GPU-based CFD (gpuCFD)

The gpuCFD method used in this study was performed using ANSYS Discovery 2022 R2 [62]. As shown by the literature review, gpuCFD is still challenging to implement, resulting in a highly competitive market. Due to the proprietary aspect of the software, certain software details are not disclosed. Unlike other CFD software, it is designed specifically for engineering purposes, making it interesting to examine its speed, accuracy, and user experience. The boundary conditions are limited to U_0 = 0.455 m/s and P_{out} = 0 Pa. There is no possibility to manually enter turbulent boundary conditions into the software. The RANS equations are closed via the standard k-ε model with two equations [63]. The wall treatment cannot be chosen and is not known. The pressure-velocity coupling scheme is SIMPLE. In the software used, there is currently no possibility to enter convergence criteria. The simulation stops based on the percent change in the solution variables, where the default tolerance for convergence is 1% [64]. The number of iterations running at that time isn't listed. The computational grid consists of 1,338,095 voxels and is shown in Figure 8.

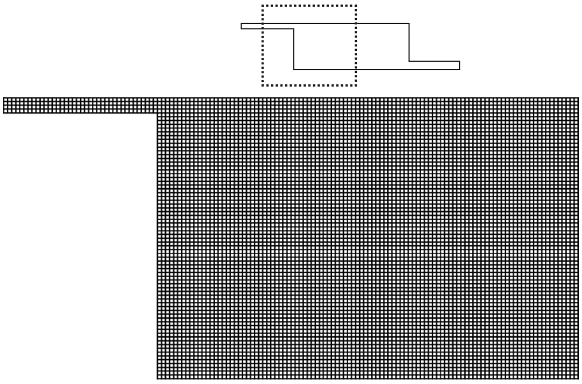

Figure 8: Two-dimensional, close-up view of the gpuCFD computational grid.

3.2 Results

3.2.1. Speed, accuracy and user experience framework

The SP-ACC-UX framework consists of indicators that are used to evaluate the performance of each computational method in a robust manner. The SP indicators used in this study are, first, the time in minutes it takes for a method to converge (obtaining the final solution) and second, the number of seconds it takes for a method to perform a single iteration. The indicators are further divided into four categories. Ranging from the slowest category to the fastest, the time to come together is broken down into (1) $t > 10^3$ min, (2) 10^2 min $< t \leq 10^3$ min, (3) 10 min $< t \leq 10^2$ min, and (4) $t \leq 10$ min. Similarly, the categories for the number of seconds per iteration are split into (1) $t > 1$ s, (2) 0.1 s $< t \leq 1$ s, (3) 0.01 s $< t \leq 0.1$ s and (4) $t \leq 0.01$ s. These values can be adjusted for future studies.

The ACC indicators used in this study are the root mean squared error (RMSE) and the factor x of observations (FACx). The two indicators are defined in Eq. (7) and Eq. (8). In addition to FAC1.1, FAC1.3 and FAC1.5 are also used. The RMSE and FACx are also split into four categories that range from the least accurate to the most accurate. These categories are defined based on the variables in question, regardless of the results obtained for this simulation case. The ACC of each FCM shall be assessed on the basis of (1) RMSE > 0.1 m/s, (2) 0.05 m/s < RMSE ≤ 0.1 m/s, (3) 0.01 m/s < RMSE ≤ 0.05 m/s, and (4) RMSE ≤ 0.01 m/s for x velocity, (1) RMSE > 10 %, (2) 5 % < RMSE ≤ 10 %, (3) 1 % < RMSE ≤ 5 %, and (4) RMSE ≤ 1 % for draught rate, as well as (1) FACx < 0.2, (2) 0.2 ≤ FACx < 0.5, (3) 0.5 ≤ FACx < 0.8, and (4) FACx ≥ 0.8 for both x velocity and draught.

$$RMSE = \sqrt{\left(\sum_{n=1}^{N} \frac{(Ux_{CFD} - Ux_{EXPERIMENT})^2}{N}\right)} \quad (7)$$

$$\mathrm{FAC}x = \frac{1}{N}\sum_{n=i}^{N} n_i, \;\; n_i = \begin{cases} 1 \; for \; \frac{1}{x} \leq \frac{Ux_{CFD_i}}{Ux_{EXP_i}} \leq x \\ \quad 0 \; else \end{cases} \quad (8)$$

The UX indicators used in this study relate to (a) geometry generation, (b) meshing (grid generation), (c) implementation of boundary conditions (BCs), (d) setting the physics involved, (e) data export, and (f) visualization. The indicators are selected based on six essential steps in a CFD research. Each indicator is rated based on how user-friendly each step is using a four-point categorical scale: (1) Programming, (2) Text User Interface (TUI), (3) Graphical User Interface (GUI), (4) Automatic. Based on this definition, a fully automated method would be better than one that requires extensive programming, in terms of UX. While that's not entirely true, this study uses this UX scale based on the assumption that new users are more likely to use a method that is easier to use rather than one that offers more flexibility.

3.2.2 Validation

The simulated U_x and $U_{x,RMS}$ values at x = H and x = 2H are depicted in Figure 9. As shown in Figure 9a and Figure 9b, the general agreement for U_x between the FCMs and the experimental data is generally satisfactory, as each method shows the inlet jet and the recirculation patterns as measured. This agreement is supported by the contour graphs of U_x in Figure 10. However, on closer examination of Figure 9, some methods capture certain flow characteristics better than others. At x = H, zCFD slightly underestimates the velocity in the jet and does not correctly predict the velocity gradient over the height. Meanwhile, U_x in the recirculation zone is overpredicted. These discrepancies are likely due to the incorrect prediction of local values, which are influenced by the wall boundary and the two-dimensional characteristics of the flow. The zero-equation model has no transport equations for turbulence, which limits its accuracy compared to models with one and two transport equations (oCFD, convCFD). The velocity in the recirculation zone (y < 0.3H) at x = H is overestimated by convCFD and oCFD. cCFD predicts the velocity in the recirculation zone most accurately. Figure 9a shows that U_x in the recirculation zone (y < 0.3H) at x = H is fairly constant, which could be due to the rough computational grid for this method. The values at x = 2H largely correspond to the experimental data. The only visible discrepancy is in the jet region, where zCFD underestimates U_x, while cCFD overestimates it.

Figure 9c and Figure 9d show the profiles for $U_{x,RMS}$ for convCFD and cCFD. convCFD gives a more accurate prediction than cCFD for both profiles. Due to software limitations, *k* cannot be obtained for the vCFD method. Moreover, oCFD and zCFD do not model *k*, nor can they be extracted.

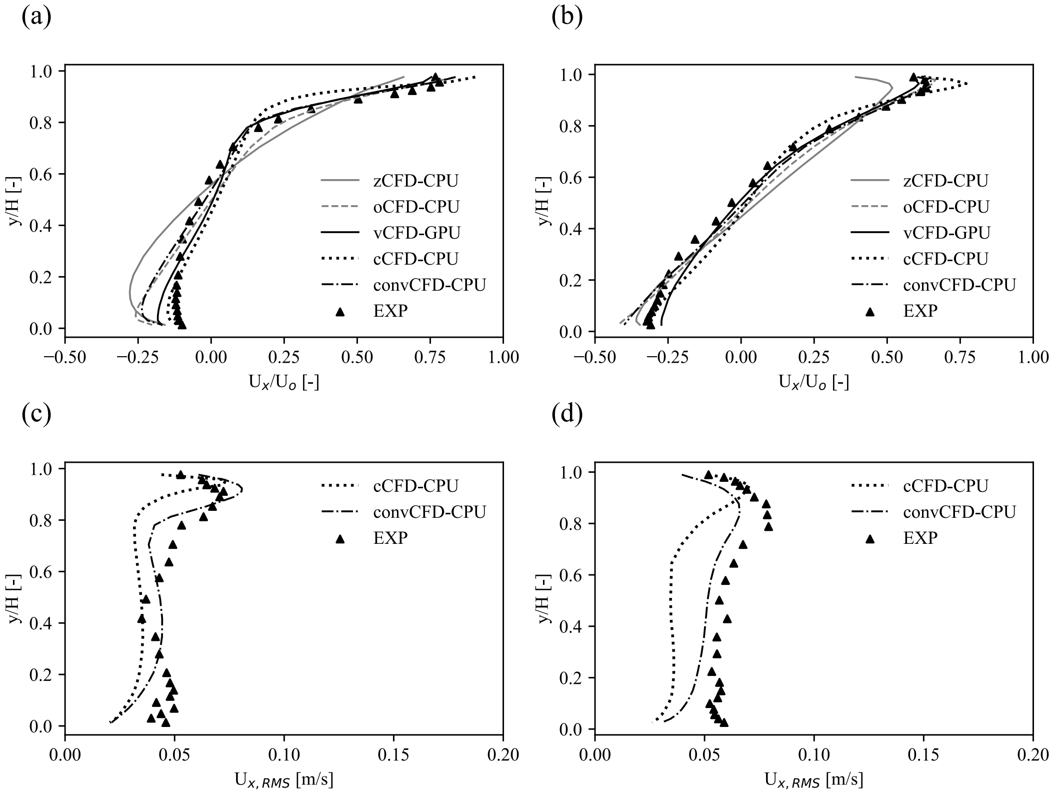

Figure 9: Simulation results for zCFD, oCFD, cCFD, convCFD and vCFD. (a) U_x at x = H. (b) U_x at x = 2H. (c) $U_{x,RMS}$ at x = H. (d) $U_{x,RMS}$ at x = 2H.

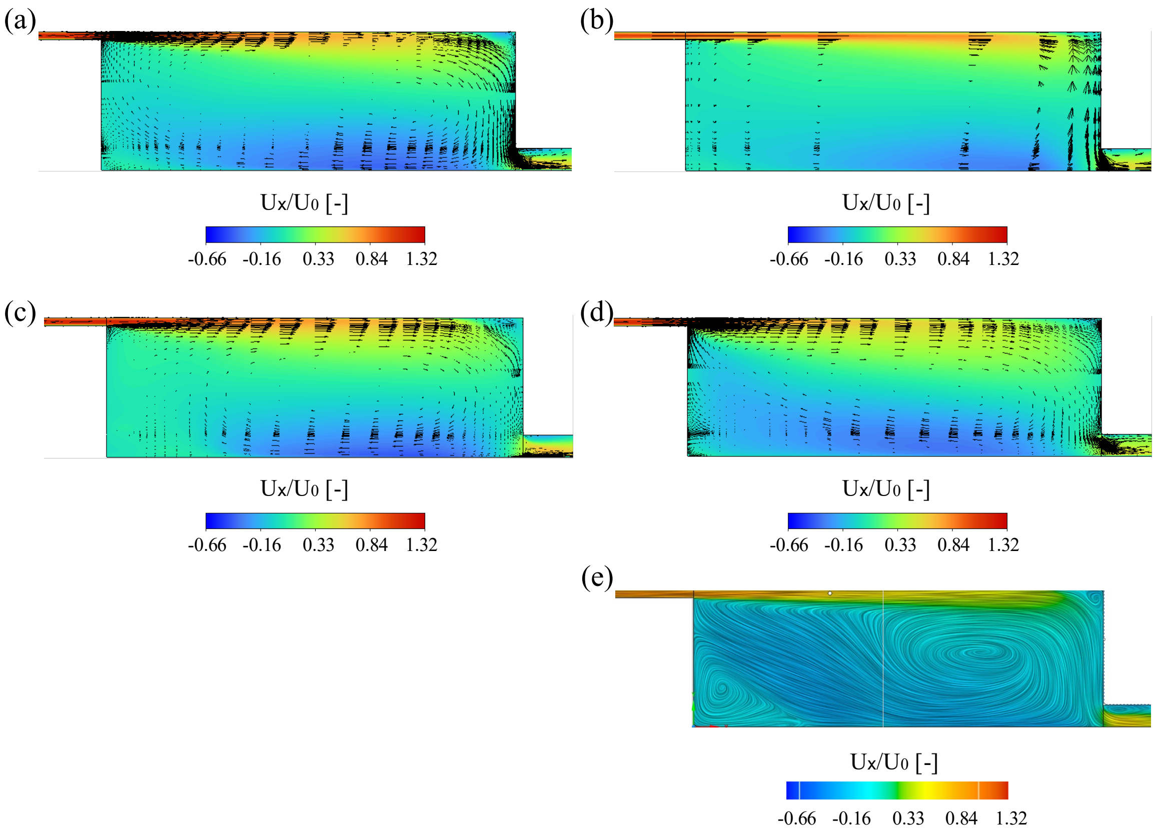

Figure 10: U_x/U_0 contour plots. (a) convCFD (b) cCFD (c) oCFD (d) zCFD (e) vCFD.

3.2.3. Accuracy, Speed, and User Experience

Figure 11 shows the ACC evaluation of the different FCMs at x = H and x = 2H. Figure 11a shows the RMSE for U_x. The highest error is observed for zCFD (RMSE = 0.047 m/s) and the lowest error is observed for vCFD (RMSE = 0.021 m/s). Figure 11b shows the corresponding FAC1.1 and FAC1.3 values. The FAC1.1 values range between 0.10 (zCFD) and 0.30 (oCFD), with oCFD ranking the highest. The FAC1.3 values range between 0.44 (zCFD) and 0.62 (convCFD and vCFD).

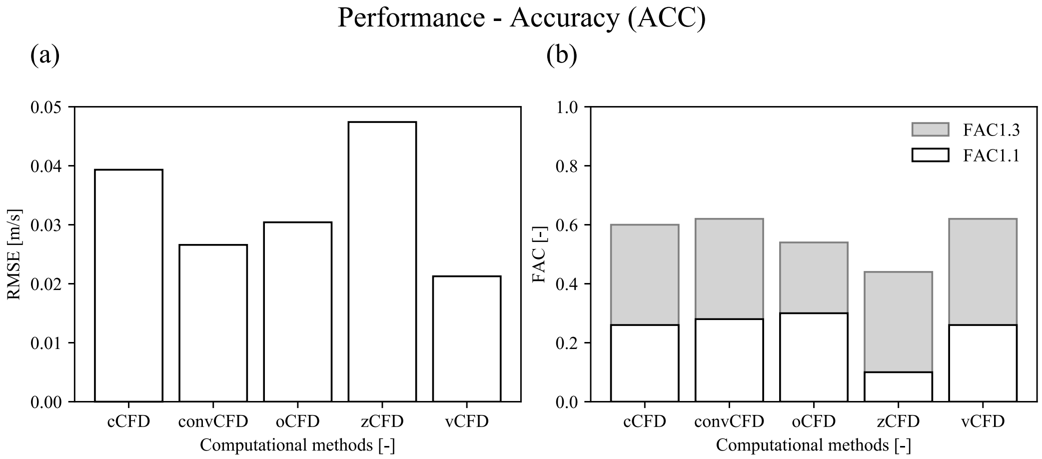

Figure 11: ACC results for zCFD, oCFD, cCFD, convCFD and vCFD, using TI = 4% approximation where necessary. (a) RMSE U_x at x = H and x = 2H. (b) FACx U_x at x = H and x = 2H.

Figure 12 shows the SP rating for the different FCMs. Figure 12a indicates that cCFD is much faster than the other methods, with a speed of about 0.3 s/iteration. Meanwhile, convCFD, oCFD, and zCFD take up about 1.30, 1.25, and 1 s/iteration. Unfortunately, the iteration rate for vCFD is not known. Figure 12b shows the convergence rate of the different FCMs. The results indicate that vCFD and cCFD are several orders of magnitude faster than the other FCMs. The convergence speeds are 11 min for cCFD, 583 min for convCFD, 378 min for oCFD, 127 min for zCFD, and 5 min for vCFD.

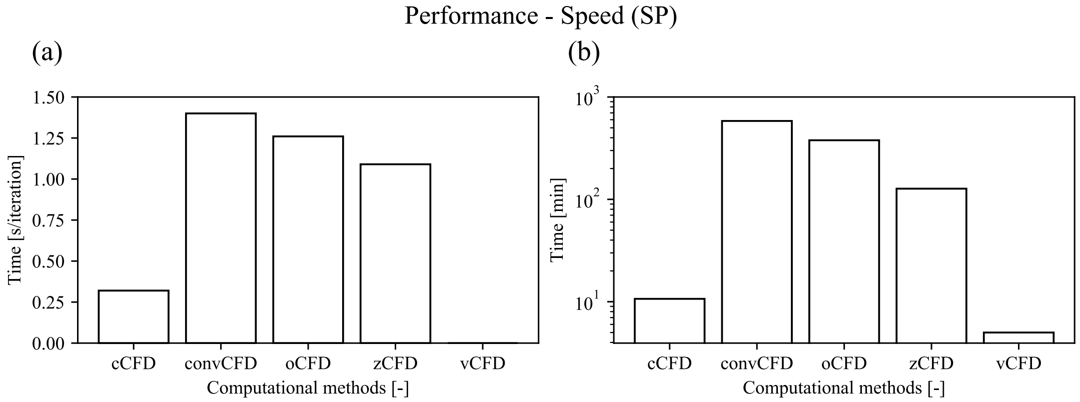

Figure 12: SP results for zCFD, oCFD, cCFD, convCFD and vCFD. (a) Iteration speed. (b) Convergence speed.

The UX assessment for the FCMs is shown in Table 4. First, the geometry generation process is reviewed, and all methods are given a 3 because they provide a GUI. For the meshing process, convCFD, cCFD, zCFD, and oCFD get a 3, while vCFD gets a 4 because of its automatic meshing capabilities. For the implementation of BCs, all FCMs are again given a 3. When it comes to the computational settings (physics), vCFD scores 4 because it automatically inputs most of the simulation parameters, while the other methods score a 3. When it comes to exporting, convCFD, cCFD, zCFD, and oCFD score a 3 because of the ability to use the GUI, while vCFD scores a 2 because of the TUI implementation. The visualization process gets a 4 for vCFD because of its real-time visualization capabilities and a 3 for the remaining FCMs. The UX parameters shown in Table 4 are averaged, resulting in an overall UX rating of 3 for vCFD and a rating of 3.0 for all other methods. Table 4 also shows the ACC and SP ratings for each FCM.

Table 4: Rating of each FCM based on speed, accuracy, and user experience.

		Computational method rating per category				
		convCFD	vCFD	cCFD	zCFD	oCFD
SP	Convergence	2	4	3	2	2
	Iteration	1	-	1	1	1
	Total [1]	**2**	**4**	**3**	**2**	**2**
ACC	FAC1.3 - U_x	3	3	3	2	3
	FAC1.1 - U_x	2	2	2	1	2
	RMSE - U_x	3	3	3	3	3
	FAC1.5 – *DR* [2]	4	1	3	3	3
	FAC1.3 – *DR* [2]	4	1	3	1	2
	RMSE – *DR* [2]	3	2	3	3	3

	Total	**3**	**2**	**3**	**2**	**3**
UX [3]	Geometry	3	3	3	3	3
	Mesh	3	4	3	3	3
	BCs	3	3	3	3	3
	Physics	3	4	3	3	3
	Exporting	3	2	3	3	3
	Visualization	3	4	3	3	3
	Total	**3**	**3**	**3**	**3**	**3**

[1] Iteration speed is omitted from the calculation due to missing data. [2] *DR* is only assessed in the occupied zone as it is more relevant. [3] Rating for UX will vary depending on the software.

Figure 13 provides a visual summary of the FCM performance assessment. On the radar graphs, SP, ACC, and UX are assumed to carry equal weight, and all three use an equidistant scale of 1 to 4. Of the five methods, vCFD scores the highest overall (SP – 4, ACC – 2, UX – 3), together with cCFD (SP – 3, ACC – 3, UX – 3). zCFD scores the lowest (SP – 2, ACC – 2, UX – 3). convCFD and oCFD share the same score (SP – 2, ACC – 3, UX – 3).

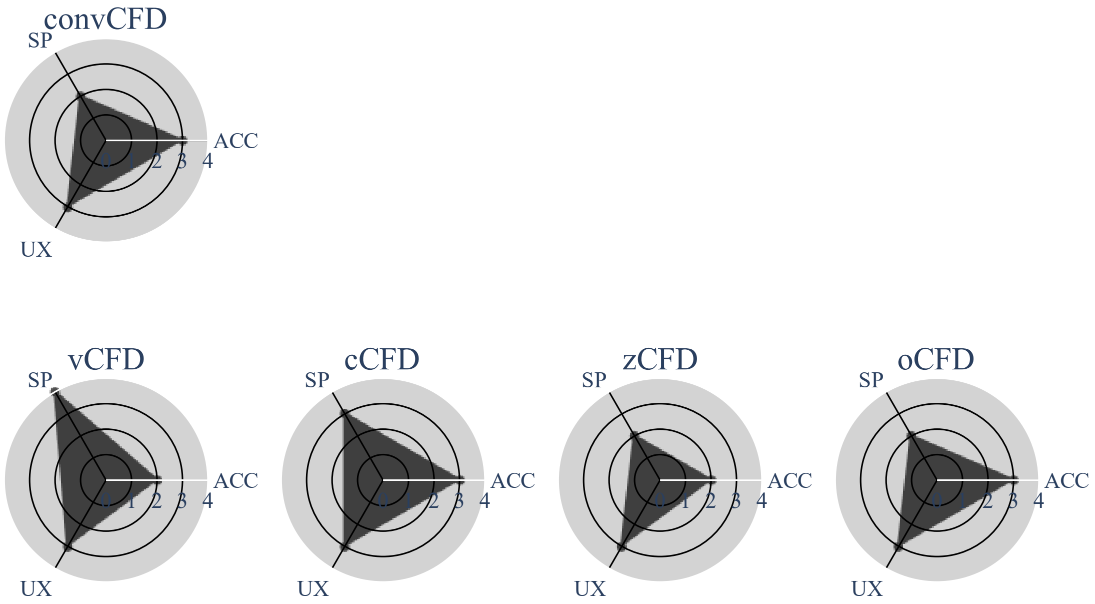

Figure 13: Overall performance of each FCM.

More information can be found in:

Mamulova, E., Loomans, M., & van Hooff, T. (2025).RANS-based fast computational methods for indoor flows: a framework-driven performance assessment for a simple benchmark. Developments in the Built Environment,23, Article 100716.https://doi.org/10.1016/j.dibe.2025.100716 .

4 Case study 2

4.1 Description of the simulations

4.1.1. Conventional CFD

The domain is discretized in hexahedral cells using ANSYS Meshing. A grid-sensitivity analysis is performed to establish a grid resolution that provides a nearly grid-independent result. The coarse, basic and fine grids consist of 1,634,616, 4,660,842 and 13,320,456 calculation cells respectively. The average dimensionless wall distance y^* does not exceed the limit $y^* < 5$ required for low Reynolds number modeling (LRNM) and the coarse grid in this case is the coarsest possible grid that meets the above criterion for LRNM.

The effective inlet area in the experiments is 0.1125 m^2 due to 50% perforation at the inlet. The mass method described by Li et al. [50] is applied and a reduced inlet velocity $U_{in} = 0.05$ m/s is imposed over an inlet area of $0.5 \times 0.45 = 0.225$ m^2, in addition to an inlet temperature of $T_{in} = 16$ °C. The static pressure at the outlet *= 0 Pa* and the backflow temperature $T_{out} = 24.3$ °C. The boundary conditions on the walls are no-slip (i.e. 0 m/s). Turbulence intensity *TI* is set to 10% at the inlet and outlet, based on a sensitivity analysis not shown here. The hydraulic diameter D_h is calculated using equation (3), where *W* and *H* correspond to the width and height of the rectangular opening. D_h at the inlet and outlet is equal to 0.47 m and 0.31 m, respectively.

$$D_h = \frac{2WH}{W+H} \tag{3}$$

A vertical temperature profile is imposed on the walls, based on five linearly interpolated measuring points. The five measuring points over the height (A-E) are shown in Table 5. $T_{floor} = 24.0$ °C and $T_{ceiling} = 24.2$ °C are estimated by Gilani et al. [51]. A heat flux φ_q of 454.44 W/m^2 is assigned to each plane of the heat source. The fluid properties used for the simulations are a dynamic viscosity of $\mu = 1.83 \times 10^{-5}$ kg/ms, a specific heat C_p = 1007 J/kgK and thermal conductivity $\lambda_{air} = 2.58 \times 10^{-2}$ W/mK. The incompressible ideal gas law is used to estimate the density of the air (ρ_{air}) as a function of temperature *T*.

Table 5: Wall surface temperature data obtained from Gilani et al. [51].

	Point A	Point B	Point C	Point D	Point E
y [m]	0.08	0.73	1.39	2.04	2.68
Wall surface temperature [°C]	22.4	23.4	24.0	24.5	24.4

The simulations are performed using the steady solver in ANSYS Fluent 2021 R1. The RANS equations are closed via the RNG k-ε model and LRNM is used to solve the turbulent flow close to the wall. Pressure-velocity coupling is done by the SIMPLEC algorithm and pressure is interpolated via the second-order discretization scheme. The remaining equations are solved using the second-order upwind scheme.

The grid-sensitivity analysis focuses on the temperature profiles, as shown in Figure 3a. The grid convergence index (GCI) is calculated using equation (4), where *r* is the linear grid refinement factor, equal to 2, *p* is the formal order of accuracy, equal to 2, and F_s the safety factor, which has a value of 1.25 for a sensitivity analysis involving three or more grids. The maximum and mean GCI_{basic} values are 0.08 and 0.47, respectively. The values indicate that the base grid produces almost grid-independent results.

$$GCI_{basic} = F_s \frac{r^p[(U_{basic} - U_{fine})/U_{in}]}{1 - r^p} \tag{4}$$

All simulations are run on an HP Zbook Studio G5 laptop, equipped with an Intel® Core™ i7-9750H central processing unit (CPU) and 16 GB of RAM. The laptop is also equipped with an NVIDIA Quadro P2000 graphics processing unit (GPU) with 4GB of video memory (VRAM). The CFD simulation is performed using 1 CPU core.

4.1.2. cCFD Method

The cCFD grid is made by making the conventional CFD grid coarser, while keeping the cell size smaller closer to the wall. The number of cells decreases from 4,660,842 cells for conventional CFD to 50,174 cells for cCFD. The boundary conditions set for cCFD are similar to those for CFDs. The cCFD simulations are also performed using the steady solver in ANSYS Fluent 2021 R1. The RANS equations are closed via the RNG k-ε model and wall functions are used for wall treatment. The simulation is run using 1 CPU core and stopped after 2,000 iterations.

4.1.3. vCFD Method

The vCFD simulations are performed via the steady solver in ANSYS Discovery 2021 R2. The software scales the voxels-based grid based on the available video memory (VRAM). The highest confidence setting discretizes the domain using 31.58 mm voxels. The total number is equal to approximately 8,654,811 voxels. The software requires basic input in the form of boundary conditions. A mass flow rate $\dot{m}$ = 0.014091 kg/s is imposed at the inlet and the dimensions of the orifice (0.3178 m × 0.3540 m (W × H)) correspond to the effective area. The software does not currently allow the user to specify a temperature profile as a boundary requirement. For this reason, the wall surface is discretized into eleven layers, each with its own temperature, as reported in Table 6. No surface temperature is assigned to the top and bottom layers. The other boundary conditions are similar to those for CFD and cCFD. There is no possibility to specify turbulent boundary conditions. The solver settings cannot be changed and various proprietary aspects of the software remain secret. Turbulence modeling is performed using the standard k-ε model. Pressure-velocity coupling is taken care of by the SIMPLE algorithm. The vCFD simulation is run on GPU and the simulation convergence is determined automatically.

Table 6: Discrete surface temperature values imposed on the walls for vCFD.

	1	2	3	4	5	6	7	8	9	10	11
y_{bot} [m]	0.0000	0.0800	0.2425	0.5675	0.8950	1.2250	1.5525	1.8775	2.2000	2.5200	2.6350
y_{top} [m]	0.0800	0.2425	0.5675	0.8950	1.2250	1.5525	1.8775	2.2000	2.5200	2.6350	2.7500
Wall [°C]	-	22.40	22.90	23.40	23.70	24.00	24.25	24.50	24.45	24.40	-

4.2 Results

Figure 14 shows the temperature profiles obtained with CFD (convCFD), cCFD and vCFD. Figures 14a-b compare the simulated values to the values measured during the experiment (EXP). The results show that all three methods capture a form of thermal stratification expected in a displacement ventilation scenario. Obviously, CFD (conventional) is the most successful in capturing the stratification, followed by cCFD and vCFD. cCFD provides the most accurate results at $y/H > 0.5$, but fails to capture the influence of the recirculation flow. Meanwhile, vCFD and CFD systematically overestimate the temperature. CFD slightly overestimates the temperature in the recirculation area, while vCFD provides the most accurate estimate of this.

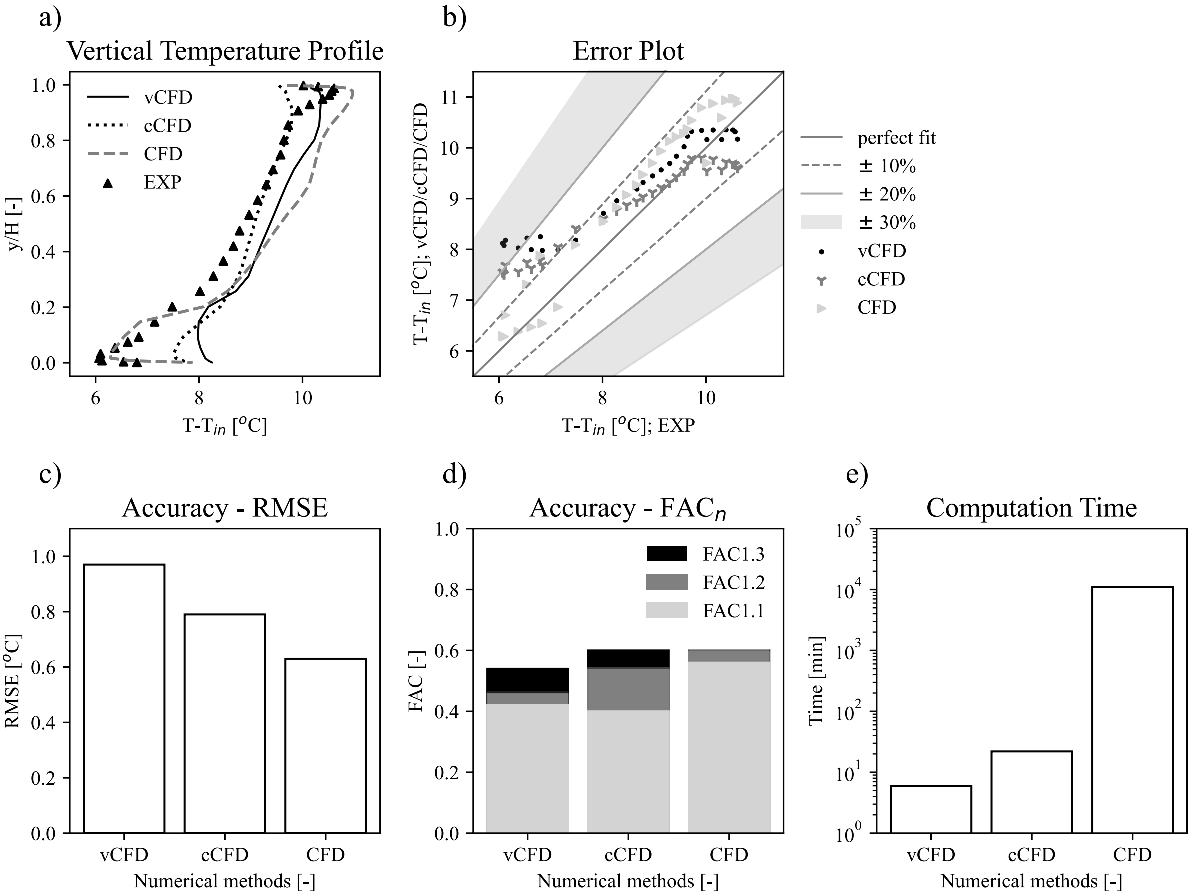

Figure 14: Results at x = 3.2 m and z = 0.75 m. a) Temperature profile, b) error plot, c) RMSE, d) FAC1.1, FAC1.2 and FAC1.3, and e) computation time for vCFD, cCFD and CFD.

Figures 14c-14d show the accuracy of the three methods for this study. CFD ranks highest on all validation metrics, with an RMSE equal to 0.63 and FAC1.3, FAC1.2, and FAC1.1 values equal to 0.60, 0.60, and 0.56, respectively. cCFD outperforms vCFD on most validation metrics. The former scores 0.79, 0.60, 0.54, and 0.40 for RMSE, FAC1.3, FAC1.2, and FAC1.1 values, respectively, while the latter scores 0.97, 0.54, 0.46, and 0.42. In addition, cCFD provides more accurate results than vCFD for the low-temperature area, which is an important aspect of displacement ventilation. Figure 14e shows the computation time for the CFD, cCFD and vCFD simulations. Both FCMs provide a significant reduction in computing time compared to CFD. vCFD requires the least computational time and is four orders of magnitude faster than conventional CFD. cCFD is three orders of magnitude faster than conventional CFD.

5 Case study 3

5.1 Description of the simulations

For this more complex geometry, three different types of FCM were tested; coarse-grid CFD, one-equation modeling, zero-equation modeling. The results were compared with both experimental data and the results of conventional RANS CFD simulations.

5.1.1. Conventional CFD

The domain is a replica of the measurement setup used (Fig. 15). The grid is based on a grid-sensitivity analysis and is fine enough to allow low-Reynolds number modeling (i.e. $y^* < 5$).

A supply velocity of 0.333 m/s was imposed at the two supply openings, which corresponds to a total ACH of 3 h^{-1}. The inlet turbulence intensity was set to 5 %, and the hydraulic diameter was 0.5 m. Zero-gauge pressure was imposed at the exhaust grilles. A surface heat flux of 65.6 W/m² was applied to the cardboard box surfaces (uniformly distributed) to maintain a total power input of 80 W per box. At the ceiling, floor, and side walls no-slip wall boundary conditions and a constant temperature were imposed.

3D steady Reynolds-averaged Navier-Stokes (RANS) simulations were performed with the RNG *k-ε* turbulence model to provide closure to the governing equations. Low-Reynolds number modeling was used to solve the flow down to the laminar sublayer. Radiative heat transfer was included using the surface-to-surface (S2S) radiation model. The drift-flux model was used to model aerosol concentrations, for which one user-defined scalar (UDS) was assigned for each particle size (e.g. dp0.3).
Pressure-velocity coupling was done using the pseudo-transient under-relaxing algorithm, pressure interpolation was solved with a staggered scheme using PRESTO!, and second-order upwind discretization schemes were used for both the convection and viscous terms of the governing equations. The incompressible ideal gas law was used to account for the buoyancy effects. Convergence of the simulation was assumed to be obtained when the mean velocities/concentrations that were monitored show only very negligible changes (< 1 %). The scaled residuals decreased to about 10^{-4} for continuity and 10^{-5}–10^{-7} for momentum, turbulent kinetic energy, turbulent dissipation rate and UDSs. All the simulations were performed with ANSYS Fluent 2023R1.

Detailed information on both the experiments and the CFD simulations can be found in Peng et al. [46].

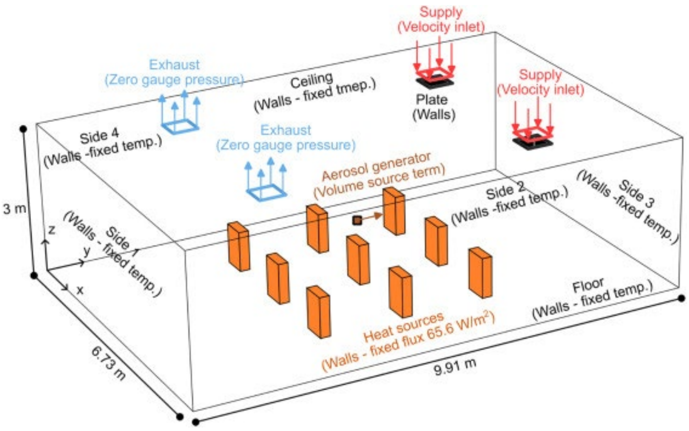

Figure 15: Computational domain with the indication of the boundary conditions [46].

5.1.2. coarse-grid CFD (cCFD)

The grid has been coarsened significantly. The settings and boundary conditions are all similar to those reported in Section 5.1.1 above.

5.1.3. One-equation modeling (oCFD)

The boundary conditions used for oCFD are similar to those of convCFD, but they differ in that the turbulent boundary conditions at the inlet and outlet are determined by the turbulence intensity (*TI)* and hydraulic diameter (D_h). Other settings are similar as well.

5.1.3. Zero-equation modeling (zCFD)

The boundary conditions are equivalent to those of the convCFD model, while some model parameters are determined by the mixing length model (e.g., near-wall treatment). No turbulent boundary conditions are used.

5.2 Results

Figure 16 shows a comparison of the velocity profiles at the nine measurement locations as obtained from the experiments and the different simulations. Overall, there is a fair to good agreement between the different computational methods and the experimental data. The results from the zero-equation model seem to deviate most from the experimental data, whereas the performance of the conventional CFD method (RANS), the coarse-grid CFD and one-equation model are in the same range.

Figure 17 shows the profiles of the air temperature. A different trend can be observed here; the largest differences with the experimental values occur with the coarse-grid CFD modeling. This can be explained by the reduced accuracy in the prediction of convective heat transfer from the heated manikins due to the too low resolution of the computational grid, necessitating the use of wall functions and subsequently resulting in lower air temperatures.

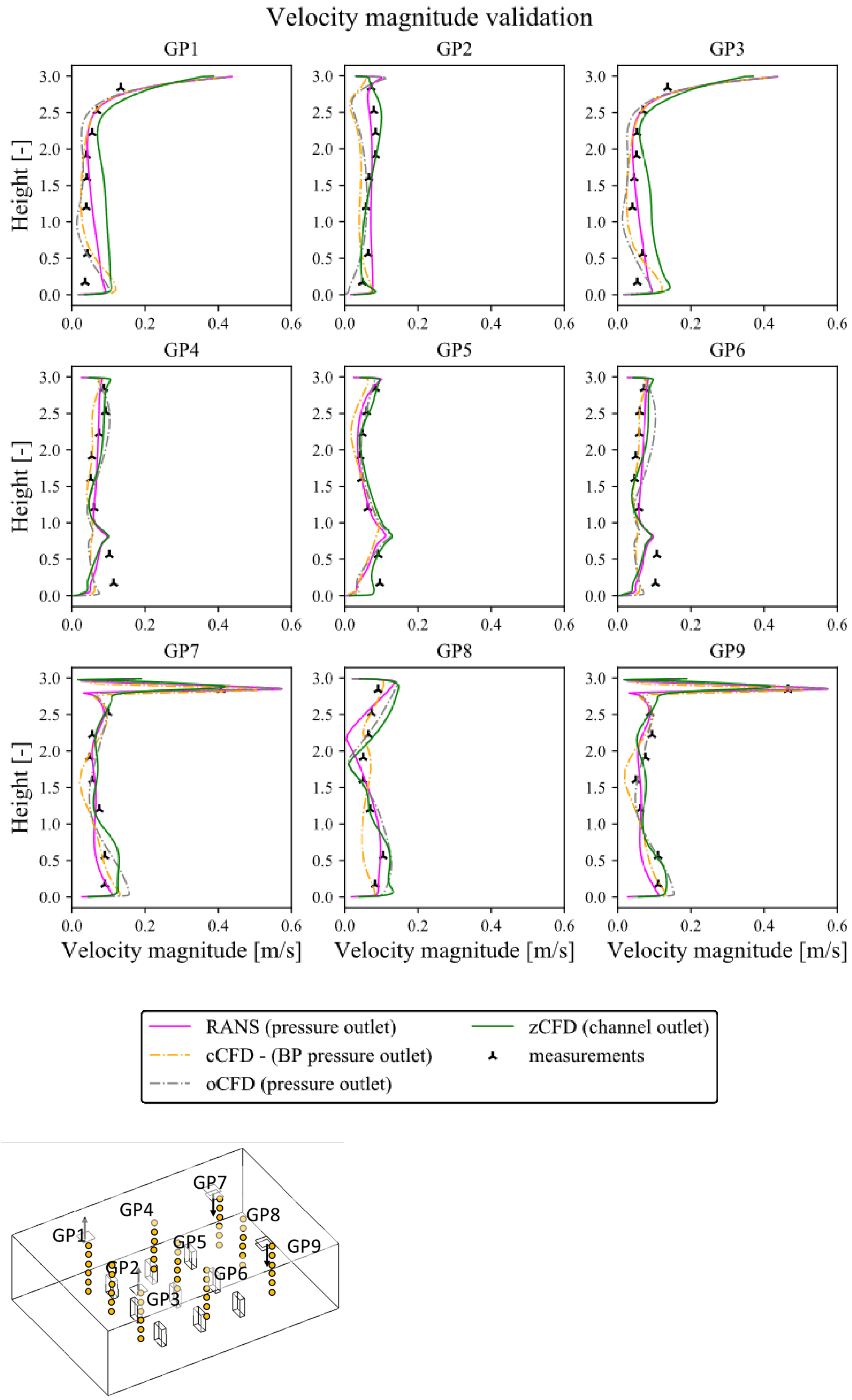

Figure 16: Comparison of velocity magnitude along 9 vertical lines in the domain, obtained from the experiments, regular RANS (convCFD), cCFD, oCFD, zCFD.

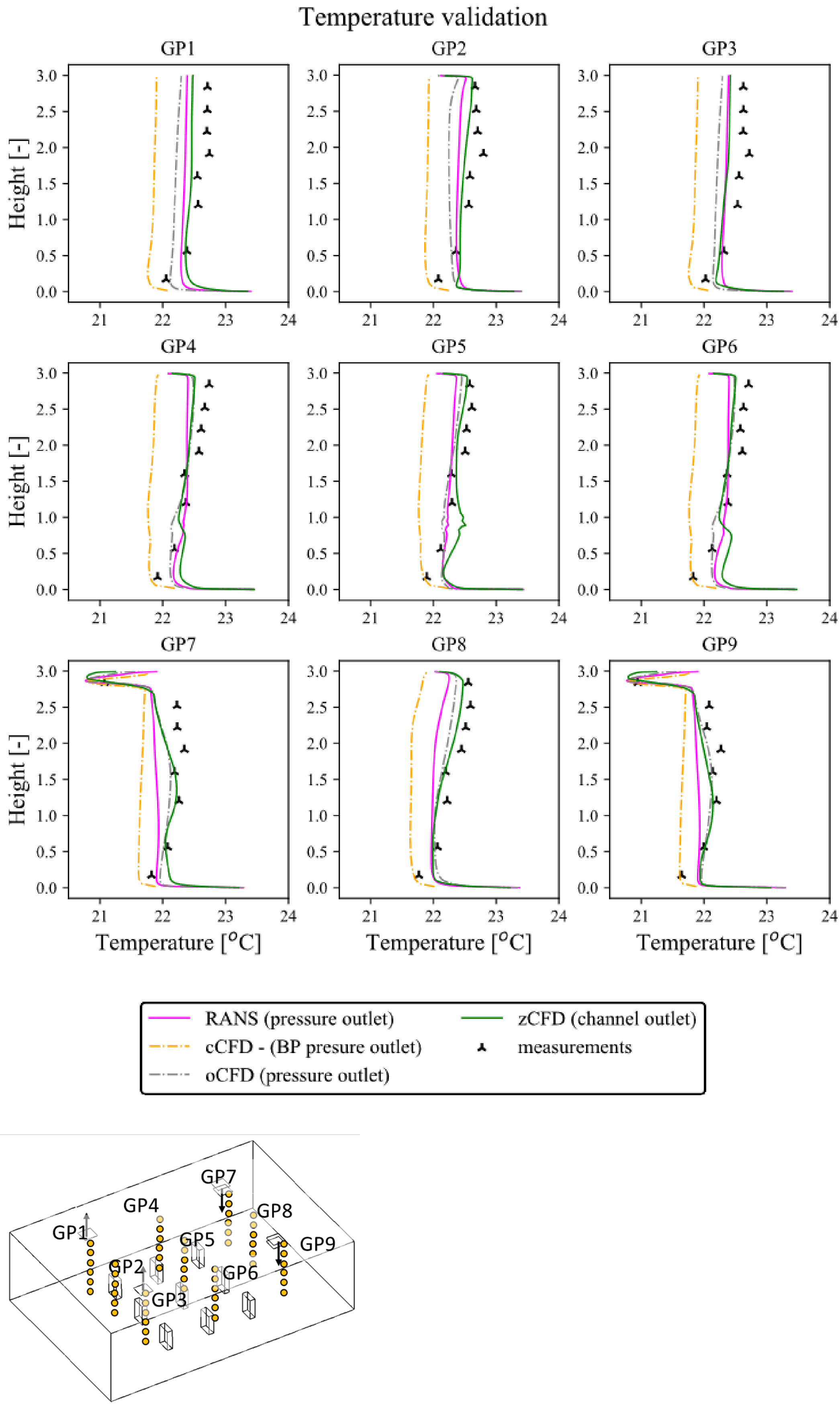

Figure 17: Comparison of air temperature along 9 vertical lines in the domain, obtained from the experiments, regular RANS (convCFD), cCFD, oCFD, zCFD.

Figure 18 shows a comparison of the aerosol concentrations at the six measurement locations (dp0p3; i.e. diameter of 0.3 μm) as obtained from the experiments and the different simulations. In general, the agreement between experiments and the different computational methods is quite good, with differences within roughly 10%.

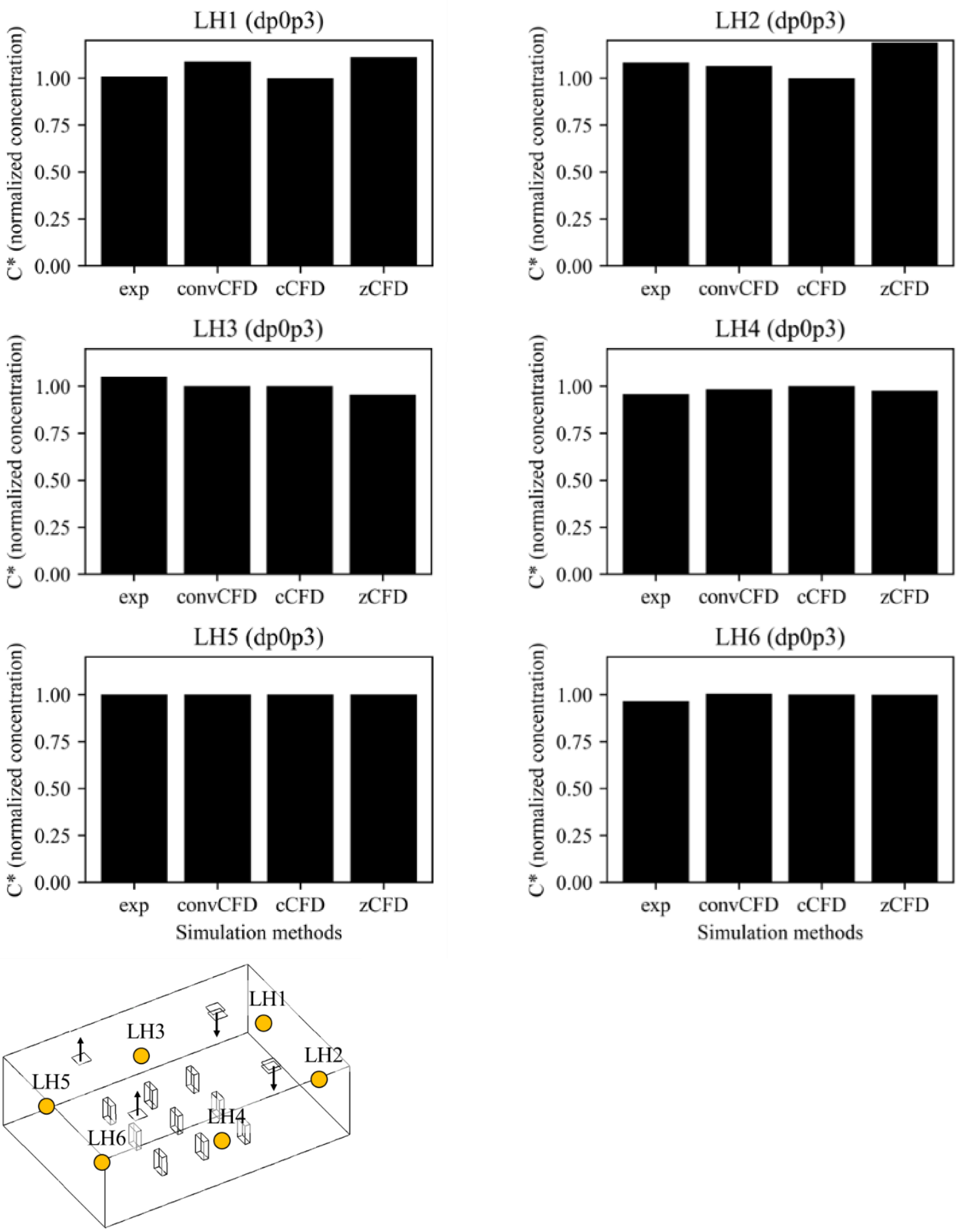

Figure 18: Comparison of aerosol concentration for (dp0p3) at 6 locations in the domain, obtained from the experiments, regular RANS (convCFD), cCFD, oCFD, zCFD.

Figures 19 and 20 show a comparison in the horizontal plane (breathing height) of the dimensionless velocity and dimensionless aerosol concentrations, respectively. The contours of the velocity in Figure 19 show that a similar flow field is present with conventional CFD (convCFD) and coarse-grid CFD (cCFD), while clear differences are present when comparing conventional CFD with one-equation modeling (oCFD) and zero-equation modeling (zCFD). The velocity field is thus largely influenced by the turbulence modeling approach, and in this particular case less by the grid resolution.

The dimensionless aerosol concentrations as depicted in Figure 20 show similar results; the concentrations obtained with coarse-grid CFD are quite similar to those obtained using conventional CFD, while the agreement is less when one-equation modeling (oCFD) and zero-equation modeling (zCFD) are used. This observation is present for both source location bottom-left and bottom-center.

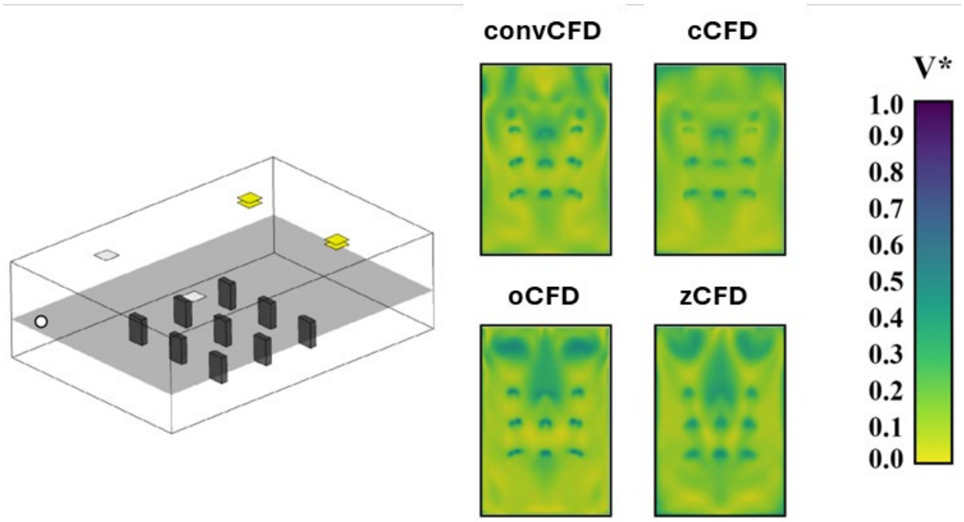

Figure 19: Comparison of dimensionless velocity in a horizontal plane at breathing height between regular RANS (convCFD), cCFD, oCFD, zCFD.

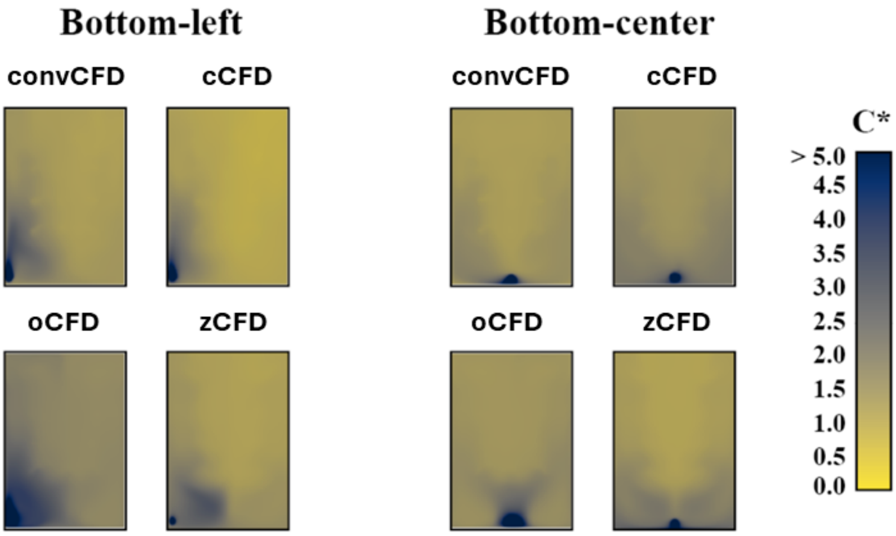

Figure 20: Comparison of dimensionless concentration in a horizontal plane at breathing height between regular RANS (convCFD), cCFD, oCFD, zCFD. Results for two different source locations; bottom-left (left figures); bottom-center (right figures).

6 Conclusions

Computational fluid dynamics (CFD) is currently the method used to simulate the airflow in a room. The time-averaged approach as applied within the RANS method limits the computational time compared to alternatives such as LES, while the predictions that can be made with it are normally sufficiently accurate. Nevertheless, a CFD calculation with RANS still requires a lot of calculation time, and there are many parts of the preparation of a CFD model in which the user can (unintentionally) influence the accuracy of the obtained results. Consequently, the use of RANS CFD, and certainly LES or DNS, in engineering practice will be limited to specialists in this field for the time being.

However, a parallel development is visible, whereby with simpler or alternative techniques, the airflow in a room can be calculated faster and/or easier, without sacrificing too much accuracy. The research within p3Venti shows that the development of these techniques goes rapidly.

However, the use of simplified turbulence models (zero equation, one-equation models) to shorten the simulation time is not recommended. The time saved does not outweigh the reduction in the quality of the results. The use of a coarse grid CFD, and the application of models that use GPUs (GPU-based CFD) do provide potentially interesting added value. In particular, the GPU-based CFD models provide significant time savings in calculating the flow problem, while the quality of the result is reduced to a limited extent. For the software studied, the simpler user interface and grid generation also provide added value. However, this is partly at the expense of full visibility and control over the created CFD model. In order to be able to estimate the effect of this, some expertise is required. In addition, the GPU-based method analyzed cannot provide concentration distributions, only velocity and temperature fields.

At the moment, it is recommended to only use these kinds of faster CFD methods (coarse grid and GPU-based CFD) in the design phase to make design decisions faster. However, for a final assessment, the use of conventional RANS models with a sufficiently fine computational grid is still recommended.

References

[1] P. V. Nielsen, “Fifty years of CFD for room air distribution,” *Build. Environ.*, vol. 91, pp. 78–90, 2015. doi: 10.1016/j.buildenv.2015.02.035.

[2] T. van Hooff, B. Blocken, and Y. Tominaga, “On the accuracy of CFD simulations of cross-ventilation flows for a generic isolated building: Comparison of RANS, LES and experiments,” *Build. Environ.*, vol. 114, pp. 148–165, 2017. doi: 10.1016/j.buildenv.2016.12.019.

[3] Q. Chen, Z. Zhang, and W. Zuo, “Computational fluid dynamics for indoor environment modeling: Past, present, and future,” in *Proc. 6th Int. Conf. Indoor Air Quality, Ventilation & Energy Conservation in Buildings (IAQVEC)*, Sendai, Japan, 2007, pp. 1–9.

[4] R. Ramponi and B. Blocken, “CFD simulation of cross-ventilation for a generic isolated building: Impact of computational parameters,” *Build. Environ.*, vol. 53, pp. 34–48, 2012. doi: 10.1016/j.buildenv.2012.01.004.

[5] M. L. Hosain and R. B. Fdhila, “Literature review of accelerated CFD simulation methods towards online application,” *Energy Procedia*, vol. 75, pp. 3307–3314, 2015. doi: 10.1016/j.egypro.2015.07.714.

[6] Z. Feng, C. W. Yu, and S. J. Cao, “Fast prediction for indoor environment: Models assessment,” *Indoor Built Environ.*, vol. 28, no. 6, pp. 727–730, 2019. doi: 10.1177/1420326X19852450.

[7] Z. Zhai and H. Wang, “Ventilation system design: Fast prediction,” in *Handbook of Ventilation Technology for the Built Environment: Design, Control and Testing*, M. Sandberg and D. Etheridge, Eds. London, U.K.: IET, 2022, pp. 307–346. doi: 10.1049/PBBE001E_ch11.

[8] G. Calzolari and W. Liu, “Deep learning to develop zero-equation based turbulence model for CFD simulations of the built environment,” *Build. Simul.*, vol. 17, no. 3, pp. 399–414, 2024. doi: 10.1007/s12273-023-1083-4.

[9] C. Caron, P. Lauret, and A. Bastide, “Machine learning to speed up computational fluid dynamics engineering simulations for built environments: A review,” *Build. Environ.*, vol. 267, p. 112229, 2025. doi: 10.1016/j.buildenv.2024.112229.

[10] M. Hu, Z. Zhang, and M. Liu, “Transient particle transport prediction based on lattice Boltzmann method-based large eddy simulation and Markov chain model,” *Build. Simul.*, vol. 16, no. 7, pp. 1135–1148, 2023. doi: 10.1007/s12273-023-0995-3.

[11] C. Zhang, Q. Chen, Y. He, Z. Liu, and Y. Yang, “Accelerating flow simulations in the built environment by using the fast fluid dynamics initializer,” *Build. Environ.*, vol. 253, p. 111274, 2024. doi: 10.1016/j.buildenv.2024.111274.

[12] N. Morozova, F. X. Trias, V. Vanovskiy, C. Oliet, and E. Burnaev, “A CFD-based multi-fidelity surrogate model for predicting indoor airflow parameters using sensor readings,” *Build. Environ.*, vol. 270, p. 112533, 2025. doi: 10.1016/j.buildenv.2025.112533.

[13] R. Ooka, “Indoor airflow simulation via deep neural network on the basis of CFD database,” *Build. Environ.*, vol. 1, pp. 3–6, 2020.

[14] C. Wei and R. Ooka, “Indoor airflow field reconstruction using physics-informed neural Network,” *Build. Environ.*, vol. 242, p. 110563, 2023. doi: 10.1016/j.buildenv.2023.110563.

[15] D. Botezelli, E. dos Santos Magalhães, and D. Antônio dos Santos, “A GPU-accelerated algorithm for solving Navier-Stokes equations,” in *Proc. 18th Brazilian Congress of Thermal Sciences and Engineering (ENCIT)*, Foz do Iguaçu, Brazil, 2022. doi: 10.26678/abcm.encit2022.cit22-0610.

[16] S. Torregrosa, V. Champaney, A. Ammar, V. Herbert, and F. Chinesta, “Predicting high-fidelity data from coarse-mesh computational fluid dynamics corrected using hybrid twins based on optimal transport,” *Mech. Ind.*, vol. 25, 2024. doi: 10.1051/meca/2024023.

[17] L. Mora, A. J. Gadgil, and E. Wurtz, “Comparing zonal and CFD model predictions of isothermal indoor airflows to experimental data,” *Indoor Air*, vol. 13, no. 2, pp. 77–85, 2003. doi: 10.1034/j.1600-0668.2003.00160.x.

[18] H. Wang and Z. (John) Zhai, “Application of coarse-grid computational fluid dynamics on indoor environment modeling: Optimizing the trade-off between grid resolution and simulation accuracy,” *HVAC&R Res.*, vol. 18, no. 5, pp. 915–933, 2012. doi: 10.1080/10789669.2012.688012.

[19] Z. Zhai and H. Wang, “Optimizing the trade-off between grid resolution and simulation accuracy: Coarse grid CFD,” in *Proc. ASHRAE Annual Conference*, Montreal, QC, Canada, 2011.

[20] K. M. Almohammadi, “Assessment of reattachment length using turbulence models on backward-facing step (BFS) for turbulent flow with modified general Richardson method,” *Arab. J. Sci. Eng.*, vol. 45, no. 11, pp. 9293–9303, 2020. doi: 10.1007/s13369-020-04695-0.

[21] D. Casseer and C. Ranasinghe, “Assessment of Spallart-Almaras turbulence model for numerical evaluation of ceiling fan performance,” in *Proc. 5th Int. Multidiscip. Moratuwa Eng. Res. Conf. (MERCon)*, Moratuwa, Sri Lanka, 2019, pp. 577–582. doi: 10.1109/MERCon.2019.8818888.

[22] M. Hurnik, P. Ciuman, and Z. Popiolek, “Eddy–viscosity Reynolds-averaged Navier–Stokes modeling of air distribution in a sidewall jet supplied into a room,” *Energies*, vol. 17, no. 5, pp. 1–19, 2024. doi: 10.3390/en17051261.

[23] S. Hussain, P. H. Oosthuizen, and A. Kalendar, “Evaluation of various turbulence models for the prediction of the airflow and temperature distributions in atria,” *Energy Build.*, vol. 48, pp. 18–28, 2012. doi: 10.1016/j.enbuild.2012.01.004.

[24] F. Bai, X. M. Gong, H. W. Li, H. B. Guo, and W. Q. Tao, “An improved black box model and the details of its numerical treatments for rack in data center simulation,” *Int. Commun. Heat Mass Transf.*, vol. 158, p. 107916, 2024. doi: 10.1016/j.icheatmasstransfer.2024.107916.

[25] B. P. P. Barbosa and N. de Carvalho Lobo Brum, “Ventilation mode performance against airborne respiratory infections in small office spaces: Limits and rational improvements for COVID-19,” *J. Braz. Soc. Mech. Sci. Eng.*, vol. 43, no. 6, pp. 1–19, 2021. doi: 10.1007/s40430-021-03029-x.

[26] H. T. Chen, Y. C. Huang, S. Rashidi, B. L. Chen, and W. M. Yan, “Numerical and experimental studies on heat transfer characteristics and ventilation for indoor vertical farming system,” *Therm. Sci. Eng. Prog.*, vol. 52, p. 102667, 2024. doi: 10.1016/j.tsep.2024.102667.

[27] X. Shao, X. Ma, X. Li, and C. Liang, “Fast prediction of non-uniform temperature distribution: A concise expression and reliability analysis,” *Energy Build.*, vol. 141, pp. 295–307, 2017. doi: 10.1016/j.enbuild.2017.02.048.

[28] B. Zhao, X. Li, Y. Li, and Q. Yan, “Indoor and outdoor airflow simulation by a zero equation turbulence model,” in *Proc. Roomvent 2000*, pp. 449–454.

[29] B. Zhao, X. Li, and Q. Yan, “A simplified system for indoor airflow simulation,” *Build. Environ.*, vol. 38, no. 4, pp. 543–552, 2003. doi: 10.1016/S0360-1323(02)00182-8.

[30] L. Phan, B. Hu, and C. X. Lin, “An evaluation of turbulence and tile models at server rack level for data centers,” *Build. Environ.*, vol. 155, pp. 421–435, 2019. doi: 10.1016/j.buildenv.2019.03.060.

[31] Q. Chen and W. Xu, “A zero-equation turbulence model for indoor airflow simulation,” *Energy Build.*, vol. 28, no. 2, pp. 137–144, 1998. doi: 10.1016/S0378-7788(98)00020-6.

[32] J. Toraño, R. Rodríguez, and I. Diego, “Computational fluid dynamics (CFD) use in the simulation of the death end ventilation in tunnels and galleries,” *WIT Trans. Eng. Sci.*, vol. 52, pp. 113–121, 2006. doi: 10.2495/AFM06012.

[33] X. Li and J. Tu, “Evaluation of the eddy viscosity turbulence models for the simulation of convection–radiation coupled heat transfer in indoor environment,” *Energy Build.*, vol. 184, pp. 8–18, 2019. doi: 10.1016/j.enbuild.2018.11.043.

[34] N. Morozova, R. Capdevila, F. X. Trias, and A. Oliva, “Towards real-time CFD simulation of indoor environment,” in *Proc. 10th Int. Conf. Comput. Fluid Dyn. (ICCFD)*, Barcelona, Spain, 2018.

[35] B. Blocken, “LES over RANS in building simulation for outdoor and indoor applications: A foregone conclusion?” *Build. Simul.*, vol. 11, pp. 821–870, 2018. doi: 10.1007/s12273-018-0459-3.

[36] ANSYS Fluent User's Guide. Available: https://www.afs.enea.it/project/neptunius/docs/fluent/html/ug/main_pre.htm

[37] H. Wang, Z. J. Zhai, and X. Liu, "Feasibility of utilizing numerical viscosity from coarse grid CFD for fast turbulence modeling of indoor environments," *Build. Simul.*, vol. 7, no. 2, pp. 155–164, 2014. doi: 10.1007/s12273-013-0163-2.

[38] P. R. Spalart and S. R. Allmaras, "A one-equation turbulence model for aerodynamic flows," *La Recherche Aérospatiale*, no. 1, pp. 5–21, 1992.

[39] L. Prandtl, "Bericht über Untersuchungen zur ausgebildeten Turbulenz," *Z. Angew. Math. Mech.*, vol. 5, no. 2, pp. 136–139, 1925.

[40] K. Chan and S. Sabatino, "Validity tests of the mixing-length theory of deep convection," *Science*, vol. 235, no. 4787, pp. 465–467, 1987. doi: 10.1126/science.235.4787.465.

[41] Z. J. Zhai, Z. Zhang, W. Zhang, and Q. Y. Chen, "Evaluation of various turbulence models in predicting airflow and turbulence in enclosed environments by CFD: Part 1—Summary of prevalent turbulence models," *HVAC&R Res.*, vol. 13, no. 6, pp. 853–870, 2007. doi: 10.1080/10789669.2007.10391459.

[42] T. Ye, R. Mittal, H. S. Udaykumar, and W. Shyy, "An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries," *J. Comput. Phys.*, vol. 156, no. 2, pp. 209–240, 1999.

[43] ANSYS Discovery. Available: https://www.ansys.com/products/3d-design/ansys-discovery

[44] P. V. Nielsen, L. Rong, and I. Olmedo, "The IEA Annex 20 two-dimensional benchmark test for CFD predictions," in *Proc. 10th REHVA World Congr. CLIMA 2010*, Antalya, Turkey, 2010. Available: https://www.researchgate.net/publication/267837630_The_IEA_Annex_20_Two-Dimensional_Benchmark_Test_for_CFD_Predictions

[45] Y. Li. M. Sandberg, L. Fuchs, 1992. Vertical Temperature Profiles in Rooms Ventilated by Displacement: Full-Scale Measurement and Nodal Modelling. Indoor Air, 2(4), 225–243.

[46] Q. Peng, A. de Lange, T. van Hooff, R. Traversari, MGLC Loomans, 2025. Experimental and CFD analysis of particle distribution in a controlled test facility: Impact of exhalation flow velocity and direction. Building and Environment. Accepted.

[47] V. Yakhot and S. A. Orszag, "Renormalization-group analysis of turbulence," *Phys. Rev. Lett.*, vol. 57, no. 14, pp. 1722–1724, 1986.

[48] M. Wolfshtein, "The velocity and temperature distribution of one-dimensional flow with turbulence augmentation and pressure gradient," *Int. J. Heat Mass Transf.*, vol. 12, pp. 301–318, 1969.

[49] H. K. Versteeg and W. Malalasekera, "An Introduction to Computational Fluid Dynamics: The Finite Volume Method", Harlow, U.K.: Pearson Education, 1995.

[50] Y. Li, M. Sandberg, L. Fuchs, 1993. Effects of thermal radiation on airflow with displacement ventilation: an experimental investigation. Energy and Buildings, 19(4), 263–274.

[51] S. Gilani, H. Montazeri, B. Blocken, 2016. CFD simulation of stratified indoor environment in displacement ventilation: Validation and sensitivity analysis. Building and Environment, 95(2016), 299–313.