W. Tian, J.W. VanGilder, X. Han, C.M. Healey, M.B. Condor, W. Zuo 2019. "A New Fast Fluid Dynamics Model for Data-Center Floor Plenums." ASHRAE Transactions, 125, pp. 141-148.

A New Fast Fluid Dynamics Model for Data-Center Floor Plenums

Wei Tian, PhD

James W. VanGilder, PE
Member ASHRAE

Xu Han
Student Member ASHRAE

Christopher M. Healey, PhD

Michael B. Condor

Wangda Zuo, PhD
Member ASHRAE

ABSTRACT

A raised-floor plenum is often used to distribute cooling airflow in data centers. Traditional Computational Fluid Dynamics (CFD) can accurately model airflow distribution in plenums, but it is expensive (computationally and financially) and is, therefore, often not accessible to the data center designer or operator. This paper presents an alternative CFD methodology based on Fast Fluid Dynamics (FFD), which is faster, more parallelizable, and simpler to program. The improvement in computational speed afforded by FFD is particularly important for multiple-iteration optimization, transient simulations, or applications for which near-real-time performance is required. The potential speed improvements of FFD have been widely documented; however, it has been generally accepted that this benefit is accompanied by a modest loss of accuracy relative to traditional CFD. We show here that this is not necessarily true. Plenum airflow predictions by our FFD implementation, which utilizes a first-order upwind rather than Semi-Lagrangian advection scheme of previous research, are extremely close to those predicted by traditional CFD - provided the same turbulence model and computational grid are used consistently.

INTRODUCTION

A raised-floor plenum distributes cooling air from computer room air handlers (CRAHs) to the data-center room through perforated tiles to maintain required IT inlet temperatures. Numerical simulation (of just the plenum in isolation, in many cases) can help evaluate the efficacy of the data center cooling system (VanGilder & Schmidt, 2005). By predicting the airflow rate through each perforated tile, the data center designer or operator can explore, for example, different perforated-tile types and locations to better match the cooling airflow delivery to the IT equipment's needs.

Kang et al. (2000) proposed a Flow Network Model (FNM) for predicting tile airflow rates. While FNM is efficient from the perspective of implementation and computation, the assumption of a uniformly pressurized plenum

Wei Tian is a research engineer, **James W. VanGilder** is the director, **Christopher M. Healey** is a senior research engineer, and **Michael B. Condor** is a senior engineer in Thermal Analytics at Schneider Electric, Andover, Massachusetts. **Xu Han** is a graduate student, and **Wangda Zuo** is an associate professor in the department of Civil, Environmental and Architectural Engineering at University of Colorado Boulder, Boulder, Colorado.

may only apply to ideal scenarios, such as those featuring restrictive tiles and deep plenums.

Potential Flow Models (PFMs), which compute airflow everywhere in the plenum just like more sophisticated types of CFD, can be employed to account for more plenum-airflow physics than FNM (VanGilder et al., 2011). While PFM is simple, fast, and stable, it does not model jet-like and circulating airflow patterns, which may compromise tile-airflow-prediction accuracy.

Traditional CFD can model essentially all the relevant airflow physics by solving a simplified form of the full Navier-Stokes equations (Kang et al., 2000; Karki et al., 2003; VanGilder & Schmidt, 2005; Pardey et al., 2015). While versatile and accurate, traditional CFD is computationally expensive. For example, VanGilder et al. (2011) reported one plenum simulation requiring several minutes in traditional CFD which could be simulated in just a few seconds with PFM. Note that computational speed is particularly limiting, when traditional CFD is used for simulating transient dynamics or used for iterative design and optimization involving multiple simulation calls.

Solving the same simplified Navier-Stokes equations as traditional CFD, Fast Fluid Dynamics (FFD) may run up to 50 times faster than traditional CFD (Zuo & Chen, 2009). This speed-up comes primarily from the time-splitting approach employed to solve velocity-pressure coupling that is simpler than that of traditional CFD – as discussed further below. Additional speedup can be achieved by parallelizing FFD and running it on a Graphics Processing Unit (GPU) (Zuo & Chen, 2010). Zuo and Chen reported that FFD is generally less accurate than CFD; however, we note that their FFD and CFD simulations were not compared under identical conditions. For example, in Zuo & Chen (2009), it was stated that the grid distributions in CFD and FFD were different though the number of cells was the same. Moreover, their implementation used a Semi-Lagrangian advection model. Healey et al. (2015) modeled a two-dimensional plenum using FFD and found that, in most scenarios presented, the accuracy of FFD was only marginally better than PFM and less than CFD. The discrepancy between FFD and CFD was believed to be due to the inherent lower accuracy of FFD; however, again, their FFD implementation used a Semi-Lagrangian advection scheme. Further, no turbulence model was employed in the FFD implementation (i.e., laminar flow was assumed) while the k-ϵ model was utilized in the CFD benchmark. Finally, the two-dimensional modeling of boundary conditions (e.g., outflow through the tiles in the "third" dimension) made it difficult to ensure that FFD and CFD models were rigorously consistent.

In this paper, we find that our FFD implementation with a first-order upwind advection scheme, can achieve comparable accuracy to CFD while maintaining its speed advantage for data center plenum applications. We determine this primarily from a hypothetical plenum example in which computational grid, turbulence model, and discretization order are specified identically in both FFD and CFD. We also assess the accuracy of our FFD implementation relative to traditional CFD and experimental data for a real 7,400 ft^2 (687 m^2) data-center floor-plenum. Despite the fact that our current FFD implementation uses a zero-equation turbulence model (Dhoot et al., 2017), it, nonetheless, delivers a similar level of accuracy as traditional CFD utilizing a k-ϵ model in predicting perforated-tile flow rates while solving significantly faster.

PLENUM AIRFLOW DYNAMICS

In this section, we discuss aspects of plenum modeling common to traditional CFD and FFD including the governing equations and relevant boundary conditions. We focus here on modeling only the raised-floor plenum space with simple boundary conditions to represent perforated-tile and inlet-airflow models.

Governing Equations

The simplified Navier-Stokes momentum equations solved by traditional CFD and FFD alike are:

$$\frac{\partial \boldsymbol{U}_i}{\partial t} = -\boldsymbol{U}_j \frac{\partial \boldsymbol{U}_i}{\partial \boldsymbol{x}_j} + \nu \frac{\partial^2 \boldsymbol{U}_i}{\partial \boldsymbol{x}_j \partial \boldsymbol{x}_j} - \frac{1}{\rho}\frac{\partial P}{\partial \boldsymbol{x}_i} + \boldsymbol{F}_i, \tag{1}$$

where $\boldsymbol{U}$ is the velocity vector, t is the time, $\boldsymbol{x}$ is the spatial coordinate, ν is the kinematic viscosity, ρ is the air density, P is the pressure, and $\boldsymbol{F}$ is the source term.

Boundary Conditions

The plenum walls are modeled as "no-slip" and "slip" boundaries for physical-wall and symmetry boundaries respectively. All inlet airflow, whether from CRAHs or vertical inlets, is modeled as uniformly distributed over the entire applicable area.

Perforated tiles are modeled as flow resistances:

$$\Delta P = \frac{1}{2}\rho f V^2, \tag{2}$$

where ΔP is the pressure drop across the tile, V is the velocity approaching the tile, and f is the dimensionless loss coefficient, which can be estimated from the manufacture data or by an empirical formula, for example, Fried and Idelchik (1989) :

$$f = \frac{1}{\beta^2}[1 + 0.5(1-\beta)^{0.75} + 1.414(1-\beta)^{0.375}], \tag{3}$$

where β is the open-area-ratio of the perforated tile.

FFD IMPLEMENTATION

The primary difference between traditional CFD and FFD is the technique used to solve the governing equations. While traditional CFD commonly uses some variant of the Semi-Implicit Method for Pressure Linked Equations (SIMPLE) (Patankar, 1980), FFD uses a time-splitting method. While there are multiple valid ways to split the equations, our implementation of FFD breaks the momentum equations into three pieces:

$$\frac{\partial \boldsymbol{U}_i}{\partial t} = -\boldsymbol{U}_j \frac{\partial \boldsymbol{U}_i}{\partial \boldsymbol{x}_j} \tag{4}$$

$$\frac{\partial \boldsymbol{U}_i}{\partial t} = \nu \frac{\partial^2 \boldsymbol{U}_i}{\partial \boldsymbol{x}_j \partial \boldsymbol{x}_j} + \boldsymbol{F}_i \tag{5}$$

$$\frac{\partial \boldsymbol{U}_i}{\partial t} = -\frac{1}{\rho}\frac{\partial P}{\partial \boldsymbol{x}_i} \tag{6}$$

Equation (4), namely, advection, can be solved by either a Semi-Lagrangian (Courant et al., 1952) or a first-order upwind scheme. We use the latter method, as it ensures rigorous mass and energy conservation unlike the Semi-Lagrangian scheme. Equation (5), namely, diffusion, is similarly solved with a traditional first-order finite volume method. Finally, the pressure equation (6) is solved together with the continuity equation:

$$\frac{\partial \boldsymbol{U}_i}{\partial \boldsymbol{x}_i} = 0, \tag{7}$$

using a projection-correction method (Chorin, 1967). For a more detailed description of FFD, refer to (Zuo and Chen, 2009; Jin et al., 2012). Finally, note that the first-order upwind scheme increases the solution time by only a small percentage relative to the Semi-Lagrangian method. Note also that the accuracy of the Semi-Lagrangian method can be improved by utilizing a higher-order scheme; however, this also increases computation time.

There are several approaches to ensure that the computed airflow field and tile airflow rates are simultaneously consistent with the governing Navier-Stokes equations and (2). For example, the perforated tiles can be directly modeled as source terms in the momentum equations or some additional outer iterations can be applied to periodically adjust the tile airflow or pressure drop in a manner that drives the solution to "converged" tile airflow rates. We adopt the pressure correction technique proposed by VanGilder et al. (2011) which shifts the plenum pressure (as determined from solving Equations (6) and (7)) by a constant value periodically until a converged flow field consistent with Equation (1) for all perforated tiles is achieved. Additionally, it is noted that our FFD solver was parallelized in OpenCL to run on the GPU, largely consistent with Tian et al. (2017).

ACCURACY EVALUATION OF OUR FFD IMPLEMENTATION

The main purpose of this study is to assess the accuracy of our FFD implementation for data center plenum applications. We do this through two examples. First, we consider a hypothetical example in which airflow predictions are made with both FFD and traditional CFD using the same grid distribution, turbulence model, and discretization order. Next, we consider an actual data center for which we have experimentally-measured tile-airflow rates. We model only the floor plenum in this study; however, we note that, in other cases, such as when large-open-area tiles are utilized, it is better practice to model the plenum simultaneously with the whitespace above the raised floor to improve accuracy. While our hypothetical example does feature large-open-area (56%) tiles, here we are interested in only relative differences between FFD and traditional CFD and not absolute accuracy. Our real data-center application is dominated by 25%-open-area tiles; in this case, a plenum-only simulation is justified.

Apples-to-Apples Comparison of FFD and CFD (and Impact of Turbulence Model)

This hypothetical case is derived from (VanGilder and Zhang, 2015). As shown in Figure 1, the "symmetric slice" of data-center is 14 ft (4.3 m) by 18 ft (5.5 m) with a plenum depth of 1 ft (0.3 m). Two identical (half) CRAHs are located diagonally opposite one another; each delivers 3,000 cfm (1.4 m^3/s) of airflow. The inflow at each CRAH is angled at 45° relative to the horizontal (and directed towards and parallel to the rows of perforated tiles), to create a stronger airflow recirculation which leads to interesting tile-to-tile airflow variations. Six 56%-open-area perforated tiles are arranged into two rows as shown.

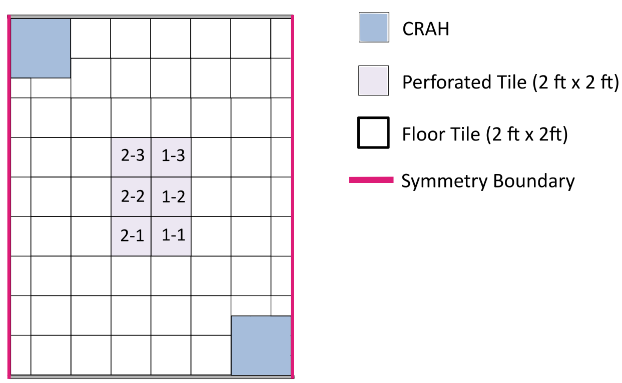


Figure 1 - Hypothetical Example Layout

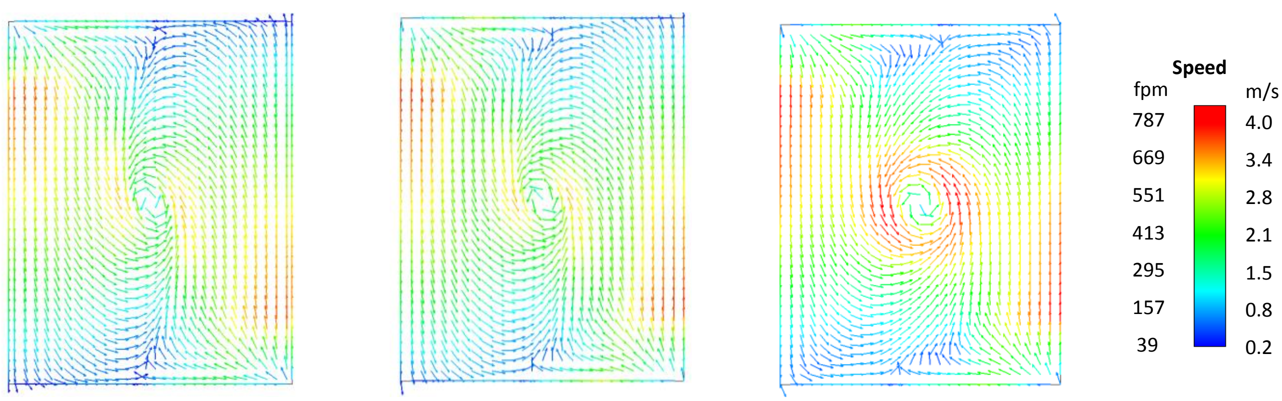
To facilitate an apples-to-apples comparison of FFD and CFD, we carefully controlled several parameters such as the computational grid, turbulence model, and numerical discretization order. Both FFD and CFD used uniform 6 in (0.15 m) cells in the horizontal directions and uniform 2 in (0.05 m) cells in the vertical direction, which produces a grid of 28 by 36 by 6 cells. Since our FFD implementation does not yet have a k-ϵ turbulence model option, we used a simple zero-equation model in which the turbulent viscosity is fixed at 100 times the molecular viscosity ($\nu_t = 100\,\nu$). Again, we note that, this may not be the best practice for plenum modeling in general; however, absolute accuracy is not our primary goal here. A first-order discretization model was employed in both FFD and CFD utilizing the upwind scheme where appropriate. The inherently-transient FFD model was solved for 100 seconds (with a time step size of 0.05 s) in order to conservatively reach steady state, while the CFD simulation was inherently steady state. Due to the diagonal symmetry of the example, we show only the airflow rates though the perforated tiles of Row 1 in Table 1. With consistent modeling choices, FFD predictions agree well with those by CFD. The maximum relative difference between tile-airflow predictions is 3.6%.

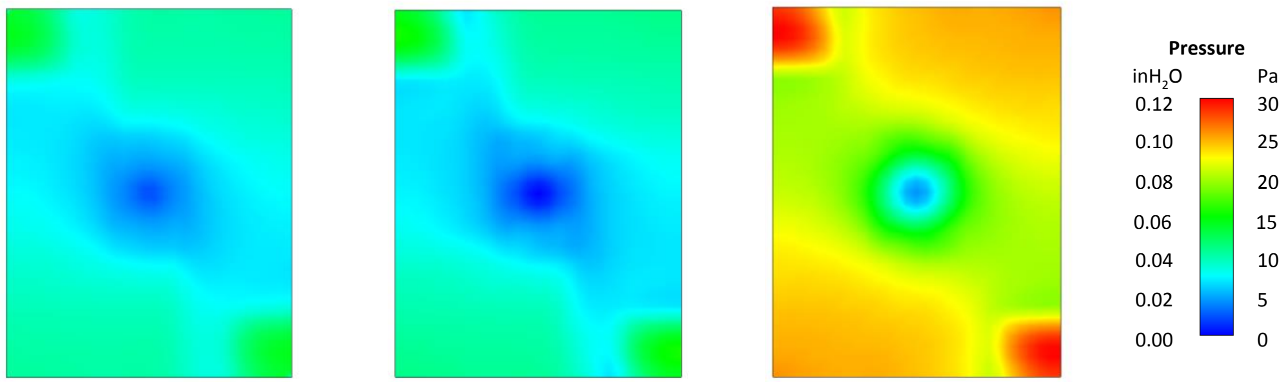
Table 1. Predicted Perforated-Tile Airflow Rates

Tile #	FFD with Constant Turbulence Model cfm (m^3/hr)	CFD with Constant Turbulence Model cfm (m^3/hr)	Relative Difference	CFD with k-ε Turbulence Model cfm (m^3/hr)
1-1	1,033 (1,755)	1,068 (1,815)	3.3%	1,138 (1,934)
1-2	912 (1,550)	913 (1,551)	0.1%	729 (1,239)
1-3	1,056 (1,794)	1,019 (1,731)	3.6%	1,133 (1,925)

For reference, Table 1 also shows the tile airflow predictions obtained by traditional CFD when the popular k-ϵ turbulence model is utilized. We see that turbulence model choice can significantly affect tile airflow predictions. This may help to explain the results of Healey et al. (2015) who did not achieve very good comparisons between "laminar FFD" and "k- ϵ CFD".

Figure 2 shows predicted velocity vectors and pressure contours at the mid-height of the plenum. Again, with consistent modeling choices, we find that FFD and CFD predictions are very similar. Both capture the recirculation of the airflow and the low-pressure region at the center of the plenum while only very small differences can be observed in the pressure field. Considering these results together with the tile-airflow predictions, we conclude that the time-splitting method in our FFD implementation delivers essentially the same accuracy as the SIMPLE scheme in traditional CFD, at least, in this application.





(a) FFD with Zero-Equation Turbulence Model　(b) CFD with Zero-Equation Turbulence Model　(c) CFD with k- ϵ Turbulence Model

Figure 2 - Velocity Vectors and Pressure Contours at Mid-height

Again, for reference, we include corresponding predictions using traditional CFD with the k- ϵ turbulence model in Figure 2. The recirculation predicted in this case is considerably stronger than predicted with the constant turbulence model and the pressure field is significantly altered. This highlights the importance of the turbulence model for practical plenum modeling and how it must be specified consistently to meaningfully compare one prediction to another. Additional study of the effect of turbulence model for plenum applications is left as future research.

Comparison of FFD and CFD for an Actual Data Center Plenum

The raised-floor plenum considered in this section corresponds to a medium-sized data center which houses 138 IT racks. The white space of the data center is 100 ft (30.5 m) by 74 ft (22.6 m). The depth of the plenum is 2 ft. As shown in Figure 3, there are 192 perforated tiles, among which 6 are 56%, 1 is 68%, and the rest are 25% open area. The plenum is completely blocked-off from the slab to the raised floor under the Power Distribution Units (PDUs). The cooling air is supplied from vertical bays along the two short sides with a total cooling airflow rate of about 102,500 cfm (48.4 m^3/s). Refer to Pardey et al. (2015) for a more complete description of the data center.

The stanchions which support the raised floor are modeled in a compact manner as suggested by VanGilder et al. (2016) with the loss coefficient f' set to 0.080 1/m (0.024 1/ft) in both FFD and CFD. A zero-equation turbulence model (Dhoot et al., 2017) was utilized in our FFD implementation. The grid size was set to be 108 by 94 by 6; the time step size was 0.1 seconds, and the total simulated time was 100 seconds in order to reach steady state starting from an initial static velocity distribution. The CFD simulation was performed using a commercial CFD program (Mentor, 2016) with the k-ϵ turbulence model. The FFD and CFD simulations employed different grid distributions although the average cell size is similar. Grid independence studies were performed for both FFD and CFD simulations.

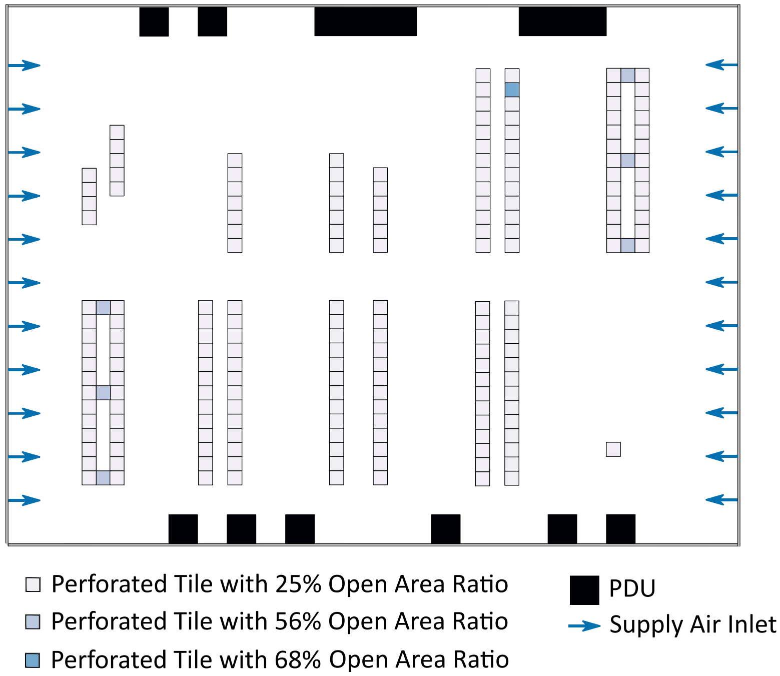
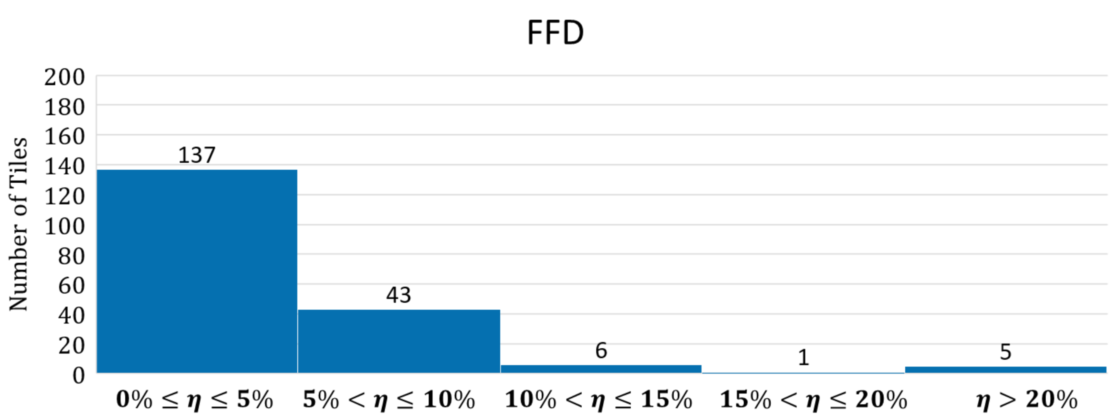


Figure 3 - Actual Data Center Layout

Figure 4 compares the perforated-tile flow rates predicted by FFD and CFD to the experimental measurements. Results are presented in terms of relative difference η, which is defined by:

$$\eta = \frac{|Q_{sim} - Q_{mea}|}{Q_{mea}}, \tag{8}$$

where Q_{sim} and Q_{mea} are the simulated and experimentally-measured perforated tile flow rates. The results associated with the 192 perforated tiles are grouped into five categories. The total number of predictions by FFD and CFD which fall into each relative-difference category are quite close. We observe that, even with a less-sophisticated turbulence model, our FFD implementation delivers comparable results to traditional CFD for this existing midsize data center. However, it should be noted that this case features a deep plenum and mostly restrictive perforated tiles so that here even a simple uniform-pressure assumption (e.g., hand calculation) yields reasonable estimates – 126, 50, 8, 6, and 2 for the five relative-difference categories of Figure 4 from left to right.



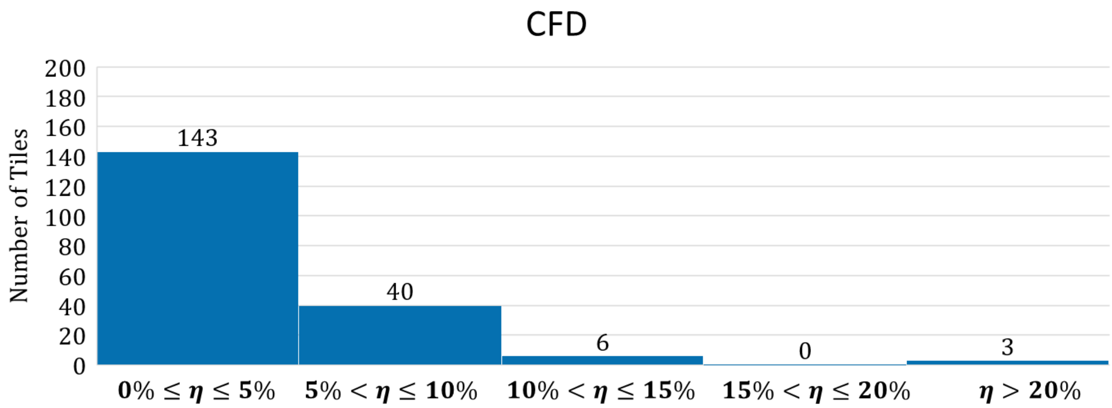


Figure 4 – Predictions of Perforated-Tile Airflow Categorized by Relative Difference from Experimentally-Measured Data

While not our primary focus here, the primary benefit of FFD over traditional CFD is its speed. With that in mind, we present our observed solution times for this real-plenum example. Our FFD implementation runs on a GPU with 20 multiprocessors, each of which has 32 cores. The CFD simulation for the actual-data-center plenum runs on 2 cores of a 4-core CPU. The FFD simulation requires approximately 66 seconds of simulation time while the CFD simulation requires 703 seconds. Note that an approximately 11-times speed advantage is achieved even though the inherently-transient FFD must solve until steady-state conditions are reached whereas the CFD simulation is inherently steady state. Consequently, for applications in which transient predictions are sought, the speed advantage is even more dramatic. Finally, we note that these observations are more anecdotal than rigorous as we are benchmarking our own FFD implementation against commercial CFD software.

CONCLUSION

Traditional CFD has been shown to be sufficiently accurate for modeling many applications including data center floor plenums. However, its utility is compromised by its long run times and high cost. FFD has been previously shown to be simpler and faster than traditional CFD - but also deemed less accurate. In contrast to this view, we show that our FFD implementation can deliver plenum-airflow predictions that are nearly indistinguishable from traditional CFD when consistent modeling settings are employed. Unlike previous studies which employed a Semi-Lagrangian scheme, our FFD implementation utilizes a first-order upwind scheme for advection. We also show that the choice of turbulence model can significantly affect plenum airflow predations which, at least, partially explains previously-reported poor comparisons between "laminar FFD" and "k-ϵ CFD". Finally, we note that our first-order upwind advection scheme adds very little to overall computational time and, therefore, preserves, the primary benefit of FFD.

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Award Number DE-0007688 and the National Science Foundation under Award No. IIS-1802017.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government.

REFERENCES

Chorin, A.J., 1967. A numerical method for solving incompressible viscous flow problems. Journal of computational physics 135, 118–125.

Courant, R., Isaacson, E., Rees, M., 1952. On the solution of nonlinear hyperbolic differential equations by finite differences. Communications on Pure and Applied Mathematics 5, 243–255.

Dhoot, P., VanGilder, J.W., Pardey, Z.M., Healey, C.M., 2017. Zero-Equation Turbulence Models for Large Electrical and Electronics Enclosure Applications.

Fried, E., Idelchik, I., 1989. Flow resistance, A Design Guide for Engineering, Hemisphere Publ.

Jin, M., Zuo, W., Chen, Q., 2012. Improvements of fast fluid dynamics for simulating air flow in buildings. Numerical Heat Transfer, Part B: Fundamentals 62, 419–438.

Mentor, 2016. FloVENT 11.3. www.mentor.com.

Pardey, M., VanGilder, J., Healey, C., Plamondon, D., 2015. Creating a calibrated CFD model of a midsize data center, in: Proceedings InterPACK, July 6-9.

Patankar, S., 1980. Numerical heat transfer and fluid flow. CRC press.

Tian, W., Sevilla, T.A., Zuo, W., 2017. A systematic evaluation of accelerating indoor airflow simulations using cross-platform parallel computing. Journal of Building Performance Simulation 10, 243–255.

VanGilder, J.W., Pardey, Z.M., Bemis, P., Plamondon, D.W., 2016. Compact modeling of data center raised-floor-plenum stanchions: Pressure drop through sparse tube bundles, in: Thermal Thermomechanical PhenomenaElectronicSystems(ITherm),201615thIEEEIntersocietyConference. IEEE, pp. 1148–1155.

VanGilder, J.W., Sheffer, Z.R., Zhang, X.S., Healey, C.M., 2011. Potential Flow Model for Predicting Perforated Tile Airflow in Data Centers. ASHRAE Transactions 117.

VanGilder, J.W., Zhang, X.S., 2015. Development of a Raised-Floor Plenum Design Tool, in: ASME 2015 International Technical Conference Exhibition Packaging IntegrationElectronic Photonic Microsystems collocated ASME 2015 13th International Conference Nanochannels, Microchannels,Minichannels. American Society of Mechanical Engineers, pp. V001T09A027–V001T09A027.

Zuo, W., Chen, Q., 2009. Real-time or faster-than-real-time simulation of airflow in buildings. Indoor air 19, 33–44.