

## Development of a fast fluid dynamics model based on PISO algorithm for simulating indoor airflow

Li, Sib0; Qiao, Hongtao

TR2021-120 October 19, 2021

### Abstract

Real-time or faster-than-real-time flow simulation is crucial for studying airflow and heat transfer in buildings, such as building design, building emergency management and building energy performance evaluation. Computational Fluid Dynamics (CFD) with PISO or SIMPLE algorithm is accurate but requires great computational resources. Fast Fluid Dynamics (FFD) can reduce the computational effort but generally lack prediction accuracy due to simplification. This study developed a fast computational method based on FFD in combination with the PISO algorithm. Boussinesq approximation is adopted for simulating buoyancy effect. The proposed solver is tested in a two-dimensional case and a three-dimensional case with experimental data. The predicted results have good agreement with the experimental results and at the same time, the proposed method can reduce computational cost greatly compared to CFD.

*ASME - Summer Heat Transfer Conference 2021*

© 2021 MERL. This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.



HT2021-63909

## DEVELOPMENT OF A FAST FLUID DYNAMICS MODEL BASED ON PISO ALGORITHM FOR SIMULATING INDOOR AIRFLOW

**Sibo Li**

Department of Mechanical and Industrial Engineering  
University of Illinois at Chicago  
Chicago, Illinois 60607  
Email: sli218@uic.edu

**Hongtao Qiao**

Mitsubishi Electric Research Laboratories  
Cambridge, Massachusetts, 02139  
Email: qiao@merl.com

### ABSTRACT

*Real-time or faster-than-real-time flow simulation is crucial for studying airflow and heat transfer in buildings, such as building design, building emergency management and building energy performance evaluation. Computational Fluid Dynamics (CFD) with Pressure Implicit with Splitting of Operator (PISO) or Semi-Implicit Method for Pressure Linked Equations (SIMPLE) algorithm is accurate but requires great computational resources. Fast Fluid Dynamics (FFD) can reduce the computational effort but generally lack prediction accuracy due to simplification. This study developed a fast computational method based on FFD in combination with the PISO algorithm. Boussinesq approximation is adopted for simulating buoyancy effect. The proposed solver is tested in a two-dimensional case and a three-dimensional case with experimental data. The predicted results have good agreement with the experimental results. In the two test cases, the proposed solver generates lower Root Mean Square Error (RMSE) compared to the FFD and at the same time, the proposed method reduces computational cost by a factor of 10 and 13 in the two cases compared to CFD.*

Keywords: Heat transfer; Mixed convection; Fast Fluid Dynamics; PISO; Semi-Lagrangian advection

### 1 INTRODUCTION

With the continuous development of computing technology, numerical simulations of indoor airflow is becoming increasingly more and more important in design process. CFD simulation is one of the most important approaches because it provides de-

tailed information about air velocity and temperature distribution [1]. However, CFD simulation usually requires great computational resources, especially on fine mesh for a large or complex indoor space [2,3]. In recent years, there have been growing interest in applying FFD as an alternative to CFD for simulating indoor airflow. FFD solves the Navier Stokes equations with a three-step time-advancement scheme and a semi-Lagrangian scheme [4,5]. Although FFD is not as accurate as a CFD model, it can capture main flow features of indoor air flows. Zuo and Chen [6] applied FFD to fast two-dimensional (2D) indoor environment modeling and found that FFD significantly accelerates the computation. Liu and Chen [5] implemented FFD in OpenFOAM [7] with unstructured mesh, enabling the practical application of the algorithm. Jin et al. [8] applied FFD to different types of natural ventilation simulation. Those studies have shown that the FFD is an attractive alternative to CFD for indoor airflow simulation. However, although FFD simulations achieve the improvement in computational speed, the accuracy is far from satisfaction [9].

The improvement of accuracy in FFD has been a concern in many recent studies. Zuo et al. [9] improved the accuracy of FFD by adopting the finite volume method and mass conservation correction. A hybrid scheme of a linear and a third-order interpolation [10] is also applied to reduce the numerical diffusion in low order scheme and the numerical dispersion in high order scheme. Xue et al. [11] combine the semi-Lagrangian (SL) scheme with Pressure Implicit with Splitting of Operator (PISO) [12] to increase accuracy. Molemaker et al. [13] tried to eliminate the numerical diffusion by applying the Quadratic Upstream Interpolat-

tion for Convective Kinematics (QUICK) [14] scheme, however, it was found that the FFD model with QUICK scheme introduces significant numerical dispersion [10].

In the current study, we propose a method by combining FFD with a PISO algorithm. The Boussinesq approximation [15] is adopted for simulating buoyancy effect. Temperature is obtained by solving the temperature transport equation implicitly to ensure stability for large time step size. The solver is implemented in open-source CFD software OpenFOAM. To ensure the validity of the implementation, the proposed method has been tested for mixed convection in a two-dimensional cavity and a three-dimensional room with an obstacle. The results show that the proposed method can achieve similar results with CFD but with much less computational time.

## 2 Methodologies

### 2.1 Semi-Lagrangian advection

The Lagrangian method considers the flow field as a discrete particle system and the SL scheme traces the flow particle trajectory as shown in Fig. 1. To predict the velocity of arrival point A of the next time step ( $t + \Delta t$ ), the SL scheme firstly finds its upstream position D via  $X_D = X_A - \Delta t U(X_A, t)$  where  $X$  represents the location. Since the obtained upstream position D will not necessarily match the center location of the cell, the values in the surrounding cells can be used to interpolate the values at position D. The cellPoint interpolation methods recommended by Liu [5] is adopted in the current study. Since the Lagrangian approach labels each point of the fluid as a separate particle, the velocity of a particle will remain the same with the lapse of time. Therefore, the  $U(X_D, t)$  estimated through interpolation is assigned to position A of the next time step, which is denoted as  $U(X_A, t + \Delta t)$ .

### 2.2 Fast fluid dynamics with PISO algorithm

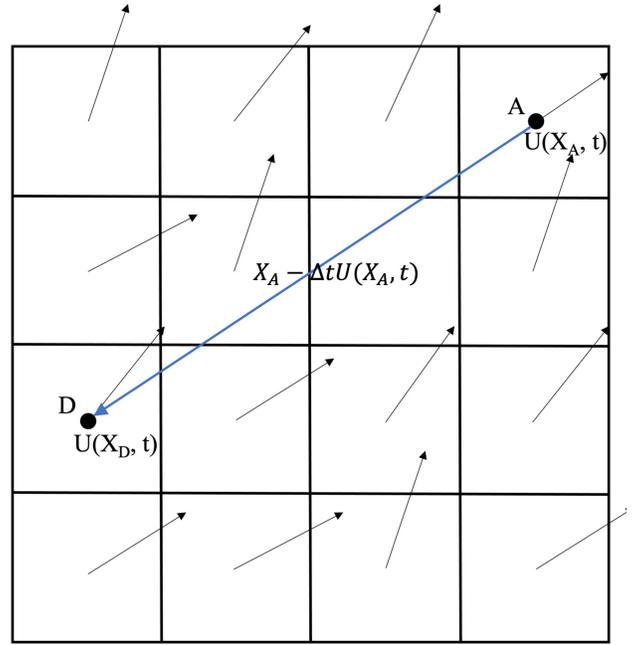
Due to the low Mach number, the indoor airflow is considered to be incompressible and viscous, FFD solves the following Navier-Stokes equations:

$$\frac{\partial U_i}{\partial x_i} = 0 \quad (1)$$

$$\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 U_i}{\partial x_j \partial x_j} + \frac{1}{\rho} F_i \quad (2)$$

where  $U_i$  is the  $i$ -th component of the velocity vector,  $p$  is the pressure,  $\rho$  is the density,  $F_i$  is the  $i$ -th component of the body forces and  $\nu$  is the effective viscosity.

FFD applies a three-step time-advancement scheme [16] that splits the momentum equation (Eqn.2) into three discretized



**FIGURE 1.** ILLUSTRATION OF THE SEMI-LAGRANGIAN SCHEME. POINT A DENOTES THE ARRIVAL POINT AND POINT D DENOTES THE DEPARTURE POINT.

equations. Firstly, the semi-lagrangian advection [4] (Eqn.3) is adopted to obtain the first intermediate velocity field  $U^*$ .  $U^n$  represents the initial velocity condition.

$$\frac{U^* - U^n}{\Delta t} = -U^n (\nabla \cdot U^n) \quad (3)$$

Then, the obtained  $U^*$  is used to solve Eqn.4 to generate the second intermediate velocity field  $U^{**}$ . The density will be incorporated into pressure term in the incompressible solver implementation. It should be noted that the FFD algorithm in the current study neglects the influence of pressure obtained from the previous time step and the obtained  $U^{**}$  does not satisfy the continuity equation so it's often called predicted velocity.

$$\frac{U^{**} - U^*}{\Delta t} = \nu \nabla^2 U^{**} + \frac{1}{\rho} F_i \quad (4)$$

In order to obtain the velocity which satisfy the continuity equation, we need to correct the intermediate velocity field  $U^{**}$  using a new pressure field  $p^{***}$ . The new pressure field  $p^{***}$  and corresponding velocity field  $U^{***}$  are sought to satisfy the following continuity and momentum equations:

$$\nabla U^{***} = 0 \quad (5)$$

$$\frac{U^{***} - U^*}{\Delta t} = -\frac{1}{\rho} \nabla p^{***} + \nu \nabla^2 U^{**} + \frac{1}{\rho} F_i \quad (6)$$

Eqn.6 subtracting Eqn.4 yields:

$$\frac{U^{***} - U^{**}}{\Delta t} = -\frac{1}{\rho} \nabla p^{***} \quad (7)$$

The FFD algorithm assumes pressure is purely based on the velocity field under the continuity restriction. To obtain the Poisson equation (Eqn.8), we take divergence for both sides of Eqn.7 and combining with Eqn.5. The new pressure field  $p^{***}$  can be obtained by solving the Poisson equation.

$$\nabla^2 p^{***} = \frac{\rho}{\Delta t} \nabla \cdot U^{**} \quad (8)$$

The new pressure field  $p^{***}$  is then substituted into Eqn.7 to yield the corrected velocity field  $U^{***}$ . However, with the corrected velocity  $U^{***}$ , the pressure  $p^{***}$  no longer satisfies the Poisson equation (Eqn.8). This is the so called pressure-velocity coupling system. For CFD, OpenFOAM mainly uses two algorithms to solve this system, PISO and SIMPLE [17]. Although both of SIMPLE and PISO algorithm can be applied for transient simulation, PISO algorithm is more computational efficient. The standard SIMPLE algorithm in OpenFOAM does not have the time derivative term in the momentum equation. Therefore, in the current study, due to the transient nature of the indoor airflow, we implement the PISO algorithm into the FFD solver as an inner loop to solve Eqn.8 and Eqn.7 repeatedly to better handle the pressure-velocity coupling system.

The temperature distribution must also be predicted for an indoor airflow simulation. In the current study, the temperature is obtained by solving the transport equation.

$$\frac{\partial T}{\partial t} + U_j \frac{\partial T}{\partial x_j} = \lambda \frac{\partial^2 T}{\partial x_j \partial x_j} + S_T \quad (9)$$

where  $\lambda$  is the effective thermal conductivity and  $S_T$  is the energy source. The implicit scheme is used to solve Eqn.9 to ensure stability when the time step size is large. The Boussinesq

approximation is applied to simulate the buoyancy effect. OpenFOAM incompressible solvers solve the temperature field separately from the velocity and pressure field. However, in the current study, the coupling between the temperature and velocity has to be considered due to the Boussinesq approximation. Therefore, the solved temperature field will be substituted into Eqn.4 to update the velocity.

The computational sequence for the proposed method is summarized in Fig. 2. By implementing the PISO algorithm, the proposed method is expected to improve the accuracy of FFD. At the same time, by maintaining the FFD structure and SL scheme, the proposed method won't sacrifice much computing speed.

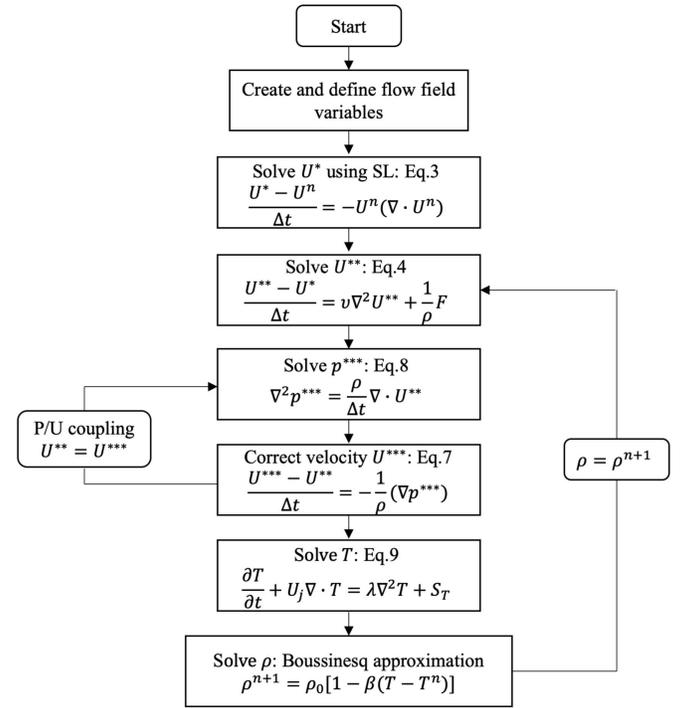


FIGURE 2. WORKFLOW OF THE PROPOSED APPROACHES

### 3 Results and Discussion

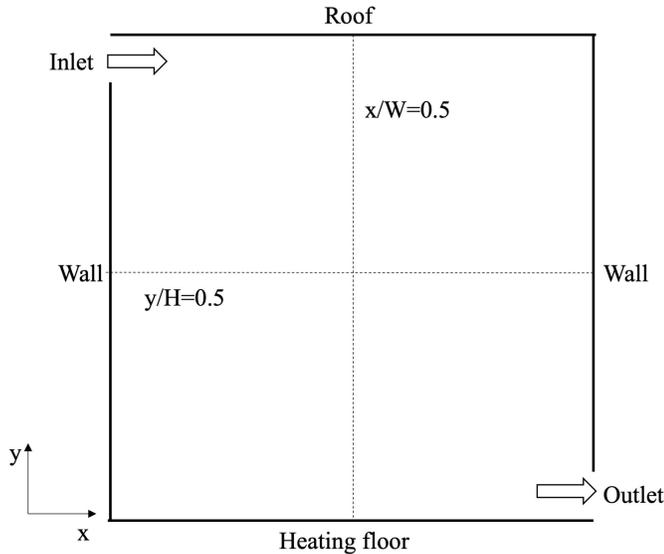
We hereafter present two test cases: a two-dimensional cavity and a three-dimensional room with an obstacle, with the goal of testing the accuracy and computational cost of the proposed solver, here referred to as ffdTPisoFoam based on OpenFOAM naming convention. The performance of ffdTPisoFoam is evaluated by comparing with the standard OpenFOAM solver buoyantBoussinesqPimpleFoam [18] and FFD.

**TABLE 1.** INITIAL AND BOUNDARY CONDITIONS FOR THE 2D CAVITY CASE

$U_x(m/s)$	$U_y(m/s)$	$T_{in}(^{\circ}C)$	$T_f(^{\circ}C)$	$T_w(^{\circ}C)$
0.57	0	15	35	15

### 3.1 Mixed convection flow in a 2D cavity

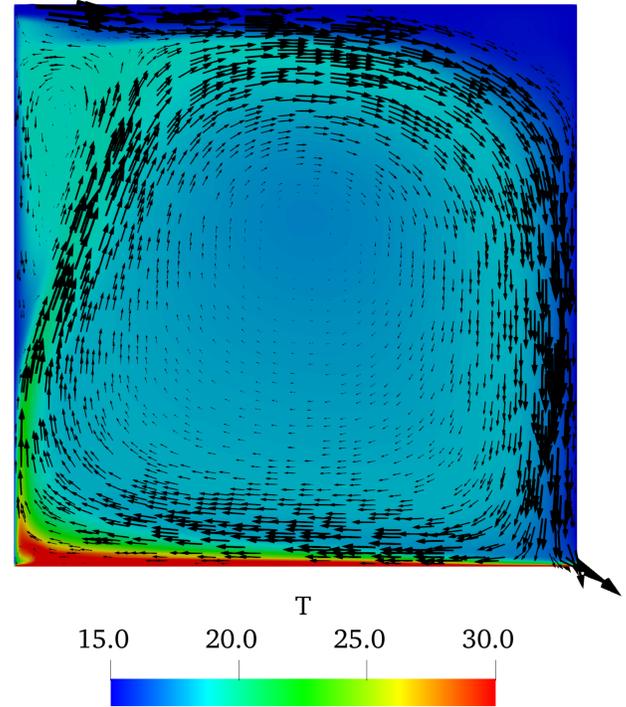
This study tested the `ffdTPisoFoam` for the case of mixed convection flow in a 2D cavity from Blay et al [19] as shown in Fig. 3. The inlet air conditions and boundary conditions are summarized in Table 1, where  $U_x$  and  $U_y$  are the inlet air velocity,  $T_{in}$  is the inlet air temperature,  $T_f$  and  $T_w$  represent the temperature of the floor and other three walls, respectively. Due to the temperature difference between the floor and the other three walls, the flow is driven by both inertia and buoyancy. The mesh size is  $80 \times 80$ , and the time step size is  $0.005s$ . The flow was calculated for a physical time of  $100s$ .



**FIGURE 3.** SKETCH OF THE 2D CAVITY

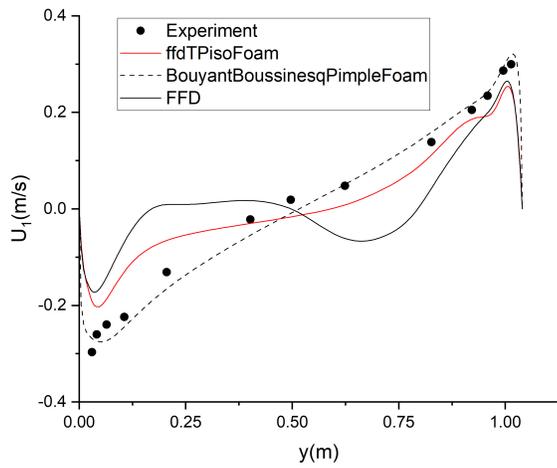
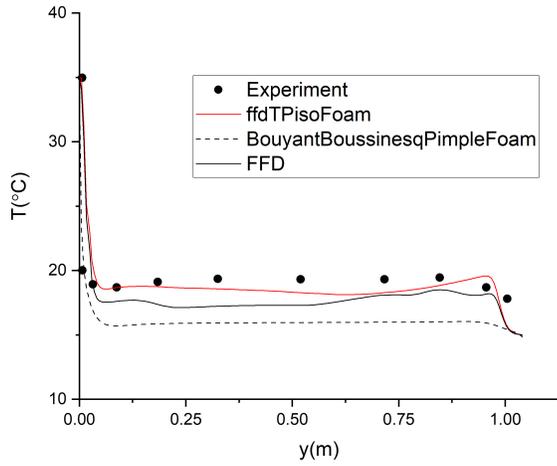
The airflow pattern and temperature distribution are shown in Fig. 4. It can be seen that the large circulation pattern near the cavity center and a secondary recirculation in the upper left corner are both predicted by the `ffdTPisoFoam`. `ffdTPisoFoam` is also able to capture the thermal plume caused by the warm floor. Fig. 5 and Fig. 6 compares the predicted results to the experimental data from Blay et al [19] and numerical results calculated by FFD and OpenFOAM baseline solver `buoyant-`

`BoussinesqPimpleFoam` in the  $x/W = 0.5$  and  $y/H = 0.5$  sections, where the BBPF denotes `buoyantBoussinesqPimpleFoam`. `buoyantBoussinesqPimpleFoam` is a transient solver for buoyant flow of incompressible fluids. It uses Boussinesq approximation for buoyancy effect and PIMPLE algorithm for velocity-pressure coupling. The so-called PIMPLE algorithm is essentially a combination of PISO and SIMPLE algorithms.

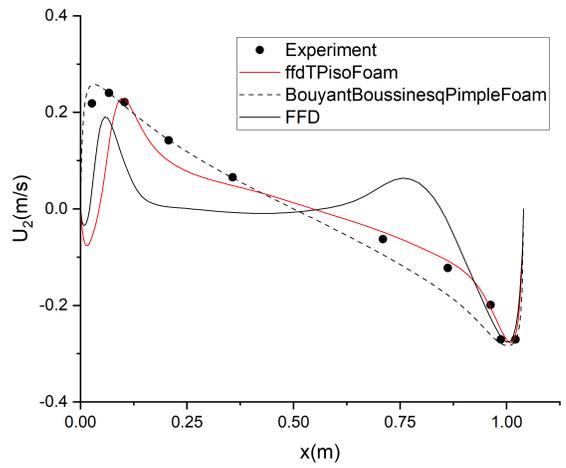
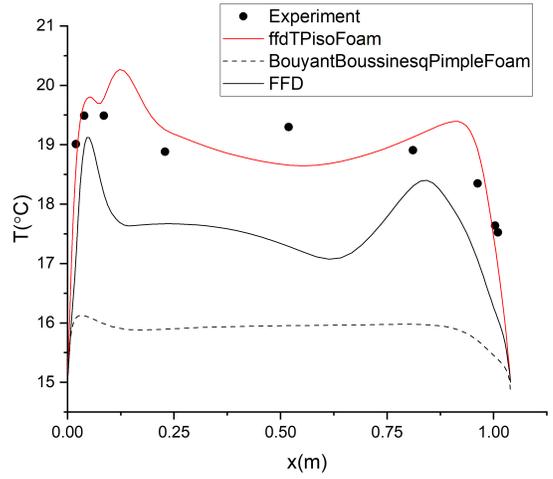


**FIGURE 4.** AIRFLOW PATTERN (ARROW) AND TEMPERATURE DISTRIBUTION PREDICTED BY FFDTPISOFOAM

From Fig. 5 and Fig. 6, it can be seen that the temperature predicted by the `ffdTPisoFoam` agrees well with the experimental data. The `buoyantBoussinesqPimpleFoam` and FFD tend to underpredict the temperature. In terms of the velocity, `buoyantBoussinesqPimpleFoam` has the best performance, however it requires great computational resources, which will be discussed in section 3.3. The velocity predicted by `ffdTPisoFoam` has slight deviation from the experimental data but still much better than the FFD. The FFD underpredicts the peak value of velocity in both sections. Also, the FFD is not able to capture the trend in the middle of the domain, which means the large circulation in the cavity is wrongly predicted.



**FIGURE 5.** COMPARISON OF THE PREDICTED RESULTS TO FFD, BBPF AND EXPERIMENTAL DATA IN  $x/W = 0.5$  SECTION. TOP PANEL: TEMPERATURE; BOTTOM PANEL: VELOCITY



**FIGURE 6.** COMPARISON OF THE PREDICTED RESULTS TO FFD, BBPF AND EXPERIMENTAL DATA IN  $y/H = 0.5$  SECTION. TOP PANEL: TEMPERATURE; BOTTOM PANEL: VELOCITY

**TABLE 2.** RMSE RESULTS FOR THE 2D CAVITY CASE

Solvers	$T - y$	$U_1 - y$	$T - x$	$U_2 - x$
ffdTPisoFoam	2.53	0.06	0.29	0.05
BBPF	2.77	0.02	2.94	0.03
FFD	3.62	0.09	1.27	0.09

To determine the accuracy of the two solvers quantitatively, the current study employs the root mean square error (RMSE) which presents an estimate of by how much the simulation results will deviate from the experimental data.

$$RMSE = \sqrt{\frac{1}{N_{data}} \sum_{m=1}^{N_{data}} (Y_{m,simulation} - Y_{m,experiment})^2} \quad (10)$$

where  $N_{data}$  represents the number of experimental observations,  $Y_{simulation}$  and  $Y_{experiment}$  represent the simulation result and experimental result, respectively. The RMSE results in predicting temperature and velocity for the three solvers in the 2D cavity case are summarized in Table 2 where BBPF represents the buoyantBoussinesqPimpleFoam solver,  $T - y$ ,  $U_1 - y$ ,  $T - x$  and  $U_2 - x$  represent the predicted results as shown in Fig. 5 and Fig. 6. It can be seen that ffdTPisoFoam outperform FFD in all cases, on average, the RMSE for ffdTPisoFoam is 46% smaller than FFD.

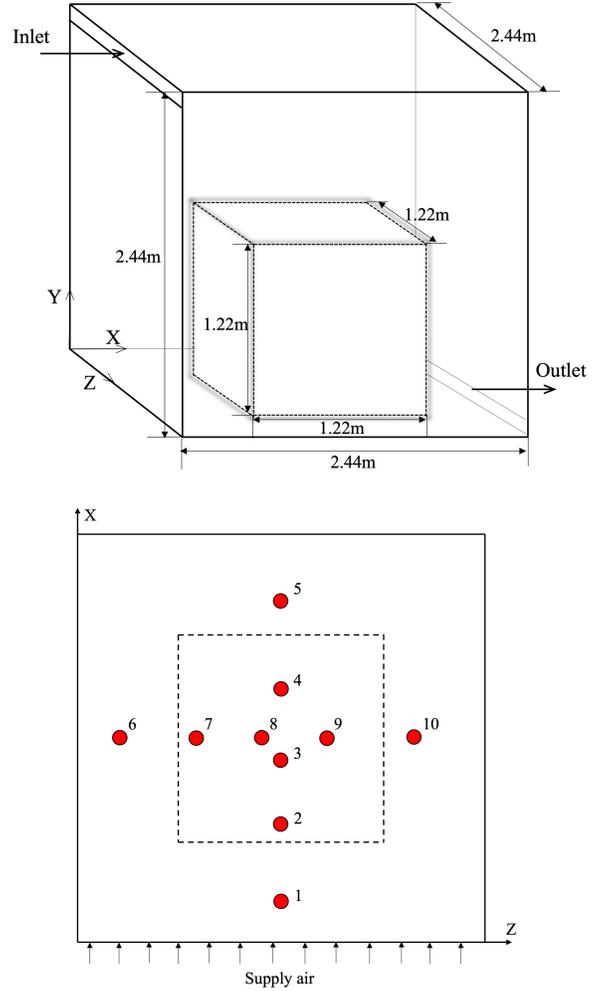
### 3.2 Mixed convection flow in a 3D room with an obstacle

The ffdTPisoFoam solver is also tested for the case of a three-dimensional simplified room [20]. The top panel in Fig. 7 shows a  $2.44m \times 2.44m \times 2.44m$  room with a plane jet from the upper left corner. The inlet height and outlet height are  $0.03m$  and  $0.08m$ , respectively. A heated box in the middle of the room on the floor, with dimensions of  $1.22m \times 1.22m \times 1.22m$ , is used to represent an electric appliance. The inlet air conditions and boundary temperatures are summarized in Table 3.  $U_{air}$ ,  $Re$  and  $T_{air}$  represent the inlet air velocity, Reynolds number and temperature, respectively.  $T_s$ ,  $T_c$ ,  $T_{sw}$  and  $T_f$  represent the temperatures of the box surface, ceiling, surrounding walls, and floor, respectively. The type of airflow in the room is mixed convection. Simulations are performed with the three solvers for a physical time of  $100s$  with a time step size of  $0.05s$ . In accordance with the grid independence test conducted by Wang and Chen [20], a structured mesh with  $44 \times 44 \times 44$  is adopted in the simulations.

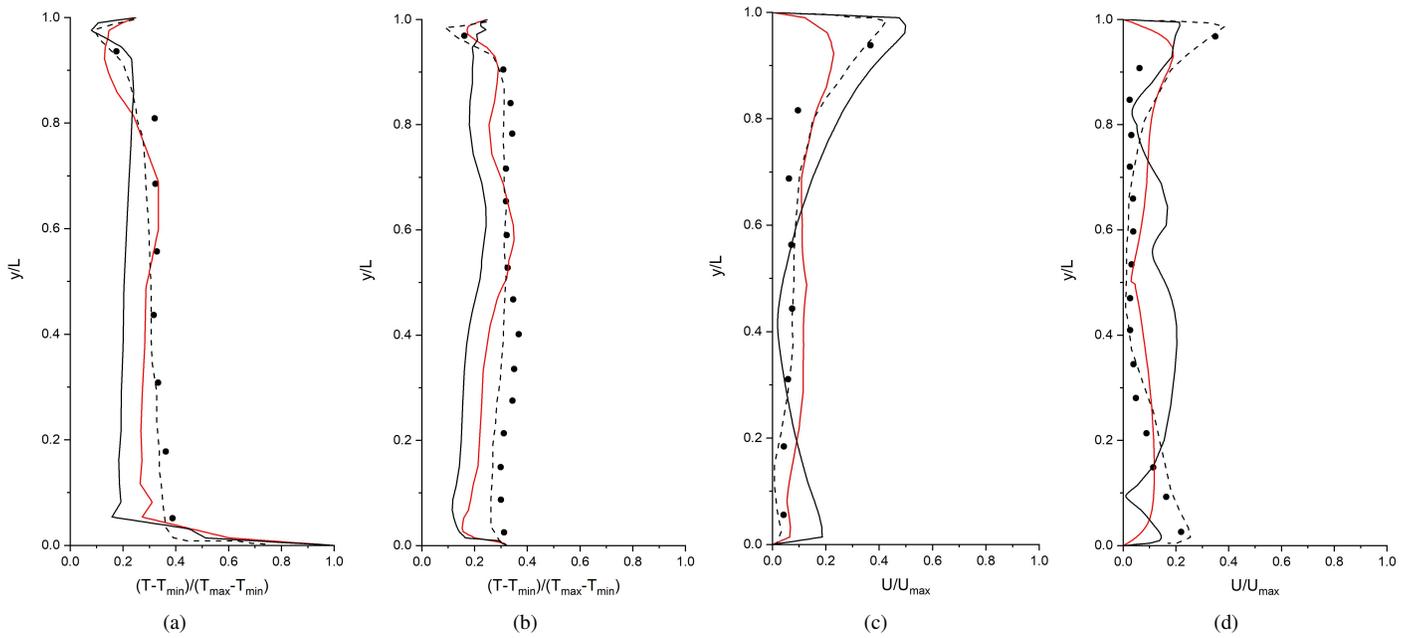
The bottom panel in Fig. 7 shows ten positions which contain most important information of the flow. The air velocity and temperature profiles at positions 3 ( $X = 1.14m, Z = 1.22m$ ) and

**TABLE 3.** INITIAL AND BOUNDARY CONDITIONS FOR THE 3D ROOM CASE

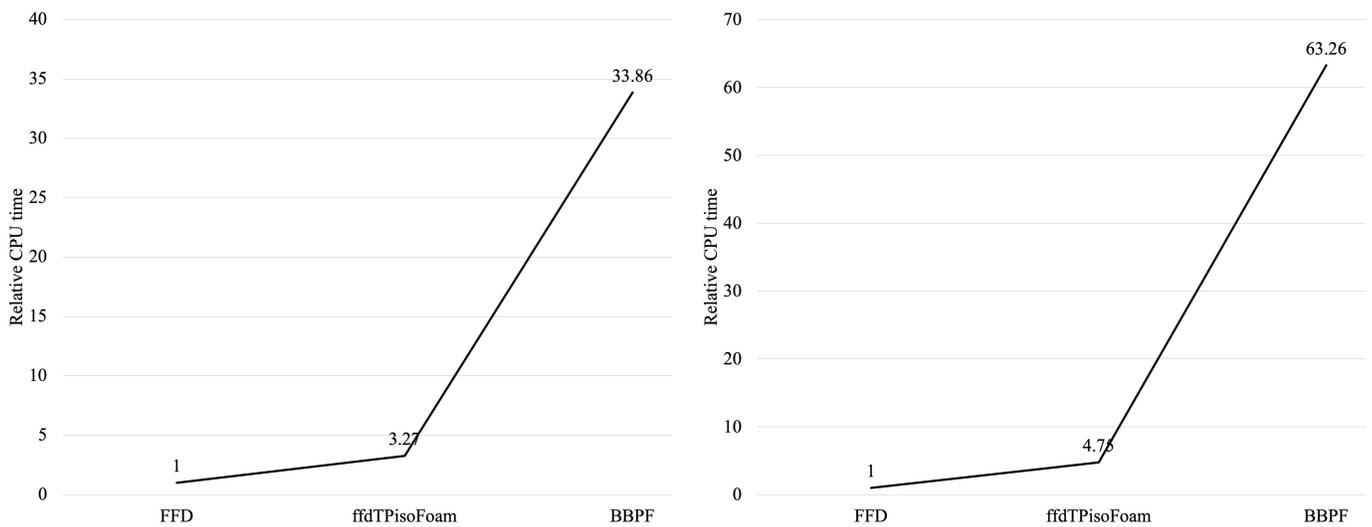
$U_{air}(m/s)$	$Re$	$T_{air}(^{\circ}C)$	$T_s(^{\circ}C)$	$T_c(^{\circ}C)$	$T_{sw}(^{\circ}C)$	$T_f(^{\circ}C)$
0.455	2000	22.2	36.7	25.8	27.4	26.9

**FIGURE 7.** SKETCH OF THE SIMPLIFIED ROOM WITH A HEATED BOX (TOP PANEL) AND THE MEASUREMENT POSITIONS FROM WANG AND CHEN [20] (BOTTOM PANEL)

position 6 ( $X = 1.22m, Z = 0.23m$ ) are selected to validate the predictions. The two positions corresponds to the jet downstream and a location close to the side wall.



**FIGURE 8.** AIR TEMPERATURE AND VELOCITY PROFILES PREDICTED BY ffdTPisoFoam (RED SOLID LINE), COMPARED WITH EXPERIMENTS BY WANG AND CHEN [20] (CIRCLES), BBPF (DASH LINE) and FFD (BLACK SOLID LINE). (a), (c): POSITION 3; (b), (d): POSITION 6



**FIGURE 9.** COMPUTATIONAL COST COMPARISON. LEFT PANEL: 2D CAVITY CASE; RIGHT PANEL: 3D ROOM CASE

**TABLE 4.** RMSE RESULTS FOR THE 3D ROOM CASE

Solvers	$T - p3$	$T - p6$	$U - p3$	$U - p6$
ffdTPisoFoam	0.0019	0.0018	0.0028	0.0032
BBPF	0.0011	0.0013	0.0017	0.0018
FFD	0.0029	0.0031	0.0039	0.0051

Figure 8 shows the air temperature and velocity profiles at the selected measurement positions, as predicted by ffdTPisoFoam, buoyantBoussinesqPimpleFoam and FFD. It can be seen that in predicting temperature, ffdTPisoFoam generates similar results with the buoyantBoussinesqPimpleFoam, they both agree well with the experimental data. FFD underpredicts the temperature, especially at the lower region of the room. The temperature in Fig. 7 is normalized by using  $T_{min} = 22.2^{\circ}\text{C}$  and  $T_{max} = 36.7^{\circ}\text{C}$ . In predicting velocity, ffdTPisoFoam performs better than FFD, but it failed to capture the maximum value. It might be because the first-order discretization scheme for time and linear interpolation method in SL scheme generate too much numerical dissipation.

The RMSE results in predicting temperature and velocity for the three solvers in the 3D room case are summarized in Table 4 where BBPF represents the buoyantBoussinesqPimpleFoam solver,  $T - p3$  and  $T - p6$  represent the temperature prediction at position 3 and position 6, respectively, and  $U - p3$  and  $U - p6$  represent the velocity prediction at position 3 and position 6, respectively. The temperature and velocity prediction at the two positions correspond to the simulation results in Fig. 8. It can be seen that ffdTPisoFoam outperforms FFD in all cases, on average, the RMSE for ffdTPisoFoam is 36% smaller than FFD.

### 3.3 Computational cost comparison

Real-time or faster-than-real-time flow simulation is crucial for studying airflow and heat transfer in buildings. Therefore, it is critical to evaluate the solver’s computational cost.

In the 2D cavity case, the comparison of the computational cost is shown in the left panel of Fig. 9 where the BBPF denotes buoyantBoussinesqPimpleFoam. The ffdTPisoFoam requires around three times larger computational time than FFD but still much faster than buoyantBoussinesqPimpleFoam. Fig. 9 right panel shows the comparison of the computational cost of the three solvers in the 3D room case. In this case, the ffdTPisoFoam is around 13.3 times faster than buoyantBoussinesqPimpleFoam while in the 2D cavity case, ffdTPisoFoam is 10.5 times faster than buoyantBoussinesqPimpleFoam. It is because that as the simulation grid size increases, the computational cost of SL scheme in ffdTPisoFoam increases linearly while the buoyantBoussinesqPimpleFoam shows exponential growth trend [11].

Therefore, the differences in computational cost between ffdTPisoFoam and buoyantBoussinesqPimpleFoam will be eventually even larger for larger-scale simulations. The advantage of ffdTPisoFoam is further enhanced with its ability to use a larger time step size than traditional CFD solvers [21].

## 4 Conclusions

In this study, we presented a new solver called ffdTPisoFoam for the integration of FFD in combination with PISO algorithm in the CFD package OpenFOAM. Boussinesq approximation is adopted for simulating buoyancy effects. Temperature distribution is obtained by solving the temperature transport equation. We assessed the performance of the proposed solver ffdTPisoFoam by applying it to predict the air distribution in a 2D cavity with mixed convection, and in a 3D room with a heated box. The results have led to the following conclusions. The developed ffdTPisoFoam achieves a higher accuracy than FFD in both temperature and velocity predictions. Although it is slightly slower than FFD, ffdTPisoFoam is still much faster than conventional CFD algorithms. For the two problems considered here, we obtained a reduction of CPU time by a factor of 10 and 13 when using ffdTPisoFoam compared to buoyantBoussinesqPimpleFoam. This factor will be eventually even larger for larger-scale simulations employing grids with millions of cells. The proposed solver is potentially a useful alternative to FFD for the simulation of indoor airflows requiring higher accuracy.

## REFERENCES

- [1] Chen, Q., 2009. “Ventilation performance prediction for buildings: A method overview and recent applications”. *Building and Environment*, **44**(4), pp. 848–858.
- [2] Lin, C. H., Horstman, R. H., Ahlers, M. F., Sedgwick, L. M., Dunn, K. H., Topmiller, J. L., Ben-nett, J. S., , and Wirogo, S., 2005. “Numerical simulation of airflow and airborne pathogen transport in aircraft cabins - part 1 : numerical simulation of the flow field”. *Ashrae 2005 Winter meeting technical and symposium papers*, **19**(1), pp. 755–763.
- [3] Li, S., Paoli, R., and D’Mello, M., 2020. “Scalability of openfoam density-based solver with runge–kutta temporal discretization scheme”. *Scientific Programming*, **2020**(9083620), p. 11.
- [4] Courant, R., Isaacson, E., and Rees, M., 1952. “On the solution of nonlinear hyperbolic differential equations by finite differences”. *Communications on Pure and Applied Mathematics*, **5**(3), pp. 243–255.
- [5] Liu, W., Jin, M., Chen, C., You, R., and Chen, Q., 2016. “Implementation of a fast fluid dynamics model in openfoam for simulating indoor airflow”. *Numerical Heat Transfer, Part A: Applications*, **69**(7), pp. 748–762.

- [6] Zuo, W., and Chen, Q., 2009. “Real time or faster-than-real-time simulation of airflow in buildings”. *International Journal of Indoor Environment and Health*, **111**(1), pp. 33–44.
- [7] Weller, H. G., Tabor, G., Jasak, H., and Fureby, C., 1998. “A tensorial approach to computational continuum mechanics using object-oriented techniques”. *Computers in Physics*, **12**(6), pp. 620–631.
- [8] Jin, M., Zuo, W., and Chen, Q., 2013. “Simulating natural ventilation in and around buildings by fast fluid dynamics”. *Numerical Heat Transfer, Part A*, **64**, pp. 273–289.
- [9] Zuo, W., Hu, J., and Chen, Q., 2010. “Improvements in ffd modeling by using different numerical schemes”. *Numerical Heat Transfer, Part B: Fundamentals*, **58**(1), pp. 1–16.
- [10] Zuo, W., Jin, M., and Chen, Q., 2012. “Reduction of numerical diffusion in ffd model”. *Engineering Applications of Computational Fluid Mechanics*, **6**(2), pp. 234–247.
- [11] Xue, Y., Liu, W., and Zhai, Z., 2016. “New semi-lagrangian-based piso method for fast and accurate indoor environment modeling”. *Building and Environment*, **105**, pp. 236–244.
- [12] Issa, R., 1986. “Solution of the implicitly discretised fluid flow equations by operator-splitting”. *Journal of Computational Physics*, **62**(1), pp. 40–65.
- [13] Molemaker, J., Cohen, J. M., Patel, S., and Noh, J., 2008. “Low viscosity flow simulations for animation”. In Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA '08, Eurographics Association, p. 9–18.
- [14] Leonard, B. P., 1979. “Stable and accurate convective modelling procedure based on quadratic upstream interpolation”. *Computer Methods in Applied Mechanics and Engineering*, **19**(1), pp. 59–98.
- [15] Gray, D. D., and Giorgini, A., 1976. “The validity of the boussinesq approximation for liquids and gases”. *International Journal of Heat and Mass Transfer*, **19**(5), pp. 545–551.
- [16] Ferziger, J. H., and Perić, M., 2002. *Computational Methods for Fluid Dynamics*. Springer.
- [17] Patankar, S., and Spalding, D. B., 1983. “A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows”. *Numerical Prediction of Flow, Heat Transfer, Turbulence and Combustion*, pp. 54–73.
- [18] Venkatesh, B. V., 2016. “Tutorial of convective heat transfer in a vertical slot.”. In CFD with OpenSource Software.
- [19] Blay, D., Mergui, S., and Niculae, C., 1992. “Confined turbulent mixed convection in the presence of horizontal buoyant wall jet”. *Fundamentals of mixed convection, ASME HTD*, **213**, pp. 65–72.
- [20] Wang, M., and Chen, Q., 2009. “Assessment of various turbulence models for transitional flows in an enclosed environment (rp-1271)”. *HVAC&R Research*, **15**(6), pp. 1099–1119.
- [21] Liu, W., and Chen, Q., 2018. “Development of adaptive coarse grid generation methods for fast fluid dynamics in simulating indoor airflow”. *Journal of Building Performance Simulation*, **11**(4), pp. 470–484.