Zuo, W., Jin, M., and Chen, Q. 2012. “Reduction of numerical diffusion in the FFD model,” Engineering Applications of Computational Fluid Mechanics, 6(2), 234–247.

REDUCTION OF NUMERICAL DIFFUSION IN THE FFD MODEL

Wangda Zuo[1,2*], Mingang Jin[3], and Qingyan Chen[3]

[1]Environmental Energy Technologies Division, Lawrence Berkeley National Laboratory, Berkeley, California, USA

[2]Key Laboratory of Three Gorges Reservoir Region’ s Eco-Environments under Ministry of Education, Chongqing University, Chongqing, P.R.China

[3]School of Mechanical Engineering, Purdue University, West Lafayette, Indiana, USA

*Corresponding Author: wzuo@lbl.gov, Phone:+1-510-492-2338, Fax:+1-510-486-4089, 1 Cyclotron Road, MS 90-3147, Berkeley, CA 94720, USA

ABSTRACT

Fast flow simulations are needed for some applications in building industry, such as the conceptual design of indoor environment or education of Heating Ventilation and Air Conditioning (HVAC) system design in classroom. Instead of pursuing high accuracy, those applications require only conceptual distributions of the flow but within a short computing time. To meet these special needs, a Fast Fluid Dynamics (FFD) method was proposed to provide fast airflow simulation with some compromise in accuracy. This study is to further improve the FFD method by reducing the numerical viscosity that is caused by a linear interpolation in its semi-Lagrangian solver. We propose a hybrid scheme of a linear and a third-order interpolation to reduce the numerical diffusion in low order scheme and the numerical dispersion in high order scheme. The FFD with both linear and hybrid

interpolations are evaluated by simulating four different indoor flows. The results show that the hybrid interpolation can significantly improve the accuracy of the FFD with a small amount of extra computing time.

KEYWORDS

Fast Fluid Dynamics, Indoor Airflow, Numerical Diffusion, Semi-Lagrangian Solver, Hybrid Interpolation

INTRODUCTION

Fast flow simulations are useful for the conceptual design of indoor environment, the development of education software for HVAC system design, the building emergency management in fire, and the real time control of indoor environment. Instead of pursuing high accuracy, those applications require only conceptual distribution of flow in time and space, but within a short computing time. However, none of current models for indoor airflow simulation can satisfy this requirement. Assuming the air in a zone, such as a room, is well-mixed, multizone network simulations (Axley, 2007) are fast, but cannot provide the spatial distribution of airflow inside the zone. The computational fluid dynamics (CFD) (Chen, 2009) can provide the information that are more accurate than the requirements, but its computing speed is too slow. As an intermediate method between the multizone models and the CFD, the FFD method (Zuo, 2009) was proposed to fill the gap by providing more detailed flow information than the multizone models at a speed much faster than the CFD. On the other side, the FFD is slower than the multizone models and less accurate than the CFD.

The FFD was originally developed by Stam (1999) for computer visualization and computer games. Since his goal was to generate plausible effects of flow motion in real time, Stam mainly adopted low order numerical schemes due to their simplicity. By scarifying some accuracy, the FFD can solve the Navier-Stokes equations at a speed of 50 times faster than the CFD (Zuo, 2009). It can provide detailed flow information, such as flow velocity, air temperature and contaminant concentration with sufficient accuracy for building conceptual design and emergency management. Thus, the FFD model has a great potential for fast and informative flow simulation in buildings.

To apply the FFD for industrial practices, the accuracy and computing speed of the FFD need to be further improved. By studying the impacts of different numerical schemes, we have significantly improved the accuracy of the FFD model and reduced its computing time by half (Zuo, 2010b). However, this work did not solve the problem in numerical viscosity, which was caused by the linear interpolation in the semi-Lagrangian solver for the advection equation (Fedkiw, 2001, Song, 2005, Molemaker, 2008).

Previous studies for reducing the numerical viscosity in the FFD model can be divided into in two categories. One is to apply high order interpolation schemes in the semi-Lagrangian solver. For instance, Fedkiw et al (2001) proposed a monotonic cubic interpolation, while Song et al (2005) adopted a constrained interpolation profile (CIP). However, the CIP method is too computationally expensive for fast flow simulation since it introduces more than ten extra equations for each interpolated variable. We implemented the monotonic

cubic interpolation and compared its performance with the linear interpolation. Unfortunately, the monotonic cubic interpolation did not provide better results than the linear interpolation for the studied flows.

The other approach is to replace the semi-Lagrangian solver with convectional CFD solvers. For instance, Molemaker et al (2008) tried to eliminate the numerical diffusion by applying the QUICK (Leonard, 1979) scheme. We found that the FFD with QUICK scheme could reduce the numerical diffusion, but it introduced significant numerical dispersion. In addition, to achieve a stable simulation, the time step size of the FFD simulation with QUICK scheme cannot be as large as the one used by the FFD with semi-Lagrangian approach.

The literature review and preliminary studies show that current solutions cannot solve the numerical diffusion problem in the FFD. Thus, it is necessary to seek an alternative solution. Considering the semi-Lagrangian solver is stable with large time step size that is important for fast flow simulation, we focus our efforts on developing a better interpolation scheme for it. In the following sections, we will first briefly introduce the FFD model. Then we will conduct a mathematical analysis for the numerical diffusion in the semi-Lagrangian solver. To reduce the numerical diffusion, different interpolations will be proposed and evaluated using ideal flows. After that, the FFD with the most promising approach will be further studied by comparing its results with the literature data and the results from the FFD

with linear interpolation for four typical indoor airflows. Finally, we will discuss the future work and make a conclusion based on the numerical studies.

FAST FLUID DYNAMICS MODEL

Using a first-order time splitting method (Ferziger, 2002), the FFD splits the momentum equation

$$\frac{\partial U_i}{\partial t} = -U_j \frac{\partial U_i}{\partial x_j} + \upsilon \frac{\partial^2 U_i}{\partial x_j^2} - \frac{1}{\rho}\frac{\partial P}{\partial x_i} + \frac{1}{\rho} S_{F,i}, \tag{1}$$

into three equations:

$$\frac{\partial U_i}{\partial t} = -U_j \frac{\partial U_i}{\partial x_j}, \tag{2}$$

$$\frac{\partial U_i}{\partial t} = \upsilon \frac{\partial^2 U_i}{\partial x_j^2} + \frac{1}{\rho} S_{F,i}, \tag{3}$$

$$\frac{\partial U_i}{\partial t} = -\frac{1}{\rho}\frac{\partial P}{\partial x_i}, \tag{4}$$

where $i, j = 1,2,3$, U_i is the ith component of the velocity vector, P the static pressure of a flow field, $S_{F,i}$ the ith component of the source, such as buoyancy force and other external forces, υ the kinematic viscosity, and ρ the density of fluid.

Then the FFD solves the advection equation (2) by using a first-order semi-Lagrangian method (Robert, 1981). To understand the principle of semi-Lagrangian method, we can write (2) in a material difference form

$$\frac{DU_i}{Dt} = 0. \tag{5}$$

It means that if an observer is standing on a particle moving with the flow, he will see that the U_i on the particle is constant in time. In other words, $U_i(t) = U_i(t-\Delta t)$. Therefore, we will know $U_i(t)$ if we find $U_i(t-\Delta t)$. To calculate $U_i(t-\Delta t)$, we can trace the particle back to its previous position at $t-\Delta t$. Since $U_i(t-\Delta t)$ at all grid points are known, we can get $U_i(t-\Delta t)$ at any other positions by interpolations.

Thereafter the FFD solves the diffusion and source equation (3) with a first-order implicit scheme. Finally, it ensures the mass conservation by solving the pressure equation (4) and continuity equation

$$\frac{\partial U_i}{\partial x_i} = 0, \tag{6}$$

together using a pressure-correction method (Chorin, 1967).

Similarly, the FFD divides the governing equation for scalar variable

$$\frac{\partial C}{\partial t} = -U_j \frac{\partial C}{\partial x_j} + \upsilon_C \frac{\partial^2 C}{\partial x_j^2} + S_{C,i}, \tag{7}$$

into two equations:

$$\frac{\partial C}{\partial t} = -U_j \frac{\partial C}{\partial x_j}, \tag{8}$$

$$\frac{\partial C}{\partial t} = \upsilon_C \frac{\partial^2 C}{\partial x_j^2} + S_{C,i}, \tag{9}$$

where C represents scalar variables, such as temperature and species concentration, and υ_C denotes the diffusivity of C. The (8) and (9) are solved in a way similar to (2) and (3).

ANALYSIS OF NUMERICICAL DIFFUSION IN THE FFD MODEL

Since (2) for velocity and (8) for scalar variables are analogous, we can write them in a general form:

$$\frac{\partial \phi}{\partial t} + U_i \frac{\partial \phi}{\partial x_i} = \frac{D\phi}{Dt} = 0 \tag{10}$$

where ϕ can be U_i or C. To analyze the numerical diffusion, a one-dimensional flow is applied for simplicity and similar conclusions can also be made by using two or three dimensional flows. We also assume that the grid is uniform and the particle cannot move more than one cell in each time step Δt:

$$\Delta x = x_{i+1} - x_i; C = \frac{|U|\Delta t}{\Delta x} < 1, \tag{11}$$

where x_{i+1} and x_i are the coordinates of grid points i+1 and i. The C is the CFL number (Courant, 1928).

Let $x^{n+1} = x_i$ and $\phi^{n+1}(x^{n+1})$ is an unknown variable for the particle in x^{n+1} at *t*, the position of the particle at *t-Δt* is:

$$x^n = x^{n+1} - U\Delta t = x_i - U\Delta t. \tag{12}$$

If $U < 0$, x^n will be between x_i and x_{i+1}. We can obtain $\phi^n(x^n)$ by a linear interpolation

$$\phi^n(x^n) = \frac{|U|\Delta t}{\Delta x}\phi_{i+1}^n + \frac{\Delta x - |U|\Delta t}{\Delta x}\phi_i^n = \phi_i^n + |U|\Delta t\frac{\phi_{i+1}^n - \phi_i^n}{\Delta x} \tag{13}$$

where the notation $\phi_i^n = \phi^n(x^n = x_i)$. Using the concerpt of semi-Lagrangian method, (10) can be reformatted as

$$\phi^{n+1}(x^{n+1}) = \phi^n(x^n). \tag{14}$$

Substituting (13) into (14) yields

$$\phi_i^{n+1} = \phi^{n+1}(x^{n+1}) = \phi^n(x^n) = \phi_i^n + |U|\Delta t\frac{(\phi_{i+1}^n - \phi_i^n)}{\Delta x}. \tag{15}$$

The ϕ_i^{n+1} is the solution of (10) by using the semi-Lagrangian solver with linear interpolation. To evaluate the accuracy of ϕ_i^{n+1}, we first expand ϕ_{i+1}^n using Taylor series

$$\phi_{i+1}^n = \phi_i^n + \left(\frac{\partial\phi}{\partial x}\right)_i^n \Delta x + \left(\frac{\partial^2\phi}{\partial x^2}\right)_i^n \frac{\Delta x^2}{2} + \left(\frac{\partial^3\phi}{\partial x^3}\right)_i^n \frac{\Delta x^3}{6} + O(\Delta x^4). \tag{16}$$

Substituting (16) into (15) gives

$$\phi_i^{n+1} = \phi_i^n + |U| \triangle t\left[\left(\frac{\partial\phi}{\partial x}\right)_i^n + \left(\frac{\partial^2\phi}{\partial x^2}\right)_i^n \frac{\Delta x}{2} + O(\Delta x^2)\right]. \tag{17}$$

Considering $U < 0$, (17) can be rearranged as

$$\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} = -U\left(\frac{\partial \phi}{\partial x}\right)_i^n + \left(\frac{\partial^2 \phi}{\partial x^2}\right)_i^n \frac{|U|\Delta x}{2} + O(\Delta x^2). \tag{18}$$

Finally, we can format (18) as

$$\frac{\partial \phi}{\partial t} + U\frac{\partial \phi}{\partial x} = \frac{|U|\Delta x}{2}\frac{\partial^2 \phi}{\partial x^2} + O(\Delta x^2). \tag{19}$$

The (19) is the actual equation solved by the semi-Lagrangian approach and is called "modeled equation" of (10). Although (19) is derived with $U < 0$, same conclusion can be obtained with $U > 0$. Compared to (10), (19) has extra terms $\frac{|U|\Delta x}{2}\frac{\partial^2 \phi}{\partial x^2} + O(\Delta x^2)$. The dominant term $\frac{|U|\Delta x}{2}\frac{\partial^2 \phi}{\partial x^2}$ is called "numerical diffusion" term since it is analogous to the diffusion term $\nu\frac{\partial^2 \phi}{\partial x^2}$ and artificially generated by the numerical scheme. Accordingly, $\frac{|U|\Delta x}{2}$ is named "numerical viscosity".

It is noteworthy that the numerical viscosity produced by the linear interpolation in the semi-Lagrangian method is the same as that by the first-order explicit upwind scheme (Ferziger, 2002). To illustrate it, we simulate the transportation of waves in inviscid fluid using the FFD with both methods. The flow domain is one-dimensional and the length of the domain is L. Suppose the flow velocity U is constant, the flow property ϕ can be described by the advection equation (10). If the initial condition is

$$\phi(x, t = 0) = \phi_0(x), x \in [0, L], \tag{20}$$

the exact solution at time t will be

$$\phi(x,t) = \phi_0\,(x-ut), x \in [0,L], \tag{21}$$

Two flows with different initial profiles are simulated. One is a sine wave

$$\phi_0\,(x) = \begin{cases} \sin\left(\frac{4\pi}{L}x\right), & x \in [0,0.5L] \\ 0 & x \in [0.5L,L] \end{cases}. \tag{22}$$

The other is a square wave

$$\phi_0\,(x) = \begin{cases} 1, & x \in [0.25L, 0.5L] \\ 0 & x \in others \end{cases}. \tag{23}$$

The parameters are U = 1 m/s and L = 1 m. The simulations were performed using two uniform meshes (50 and 100 grids) and two time step sizes (0.005 s and 0.01 s). The profiles at t = 0.25 s predicted by the semi-Lagrangian method with linear interpolation (Figure 1) are the same as those by the first-order upwind scheme (Figure 2). Both methods under-predict the amplitudes due to the damping effect of numerical viscosity. Since the numeric viscosity is proportion to the mesh size Δx, it can be reduced by using finer grids. For instance, the results become better when the Δx decreases from 0.02 m to 0.01 m although the CFL number remains the same.

Compared to the physical viscosity of air, the numerical viscosity is significant for indoor airflow simualtions. For instance, a comfortable speed of indoor air flow $|U|$ should be around 0.4 m/s (ASHRAE, 1995). A typical mesh size Δx for indoor airflow simulation is about 0.1 m. Then the numerical viscosity $0.5|U|\Delta x$ is about 2×10^{-2} m^2/s, which is 1200 times larger than the kinematic viscosity of air at room temperature (~1.568×10^{-5} m^2/s). The

numerical viscosity has both positive and negative impacts on the simulation. On one side, it can partially act as turbulent viscosity since there are not treatments for turbulence in the FFD model. It can also stabilize the simulation which is critical for computer games. On the other side, the numerical viscosity is not turbulent viscosity so that the FFD model has difficulties to precisely predict the turbulent flows. In addition, simply adding turbulence models into the FFD cannot improve the prediction for turbulence (Zuo, 2009) since the flow will become too viscous with both numerical and turbulent viscosities. To improve the accuracy of the FFD model, it is necessary to reduce the numerical viscosity.

A HYBRID INTERPOLATION

As shown in the previous analysis, the numerical viscosity is caused by the low order linear interpolation in the semi-Lagrangian solver. Thus, it is possible to minimize it by using higher order interpolation schemes, such as a third-order interpolation

$$\phi(x) = \phi_i^n + \frac{|U|\Delta t(\phi_{i+1}^n - \phi_{i-1}^n)}{2\Delta x} + \frac{(|U|\Delta t)^2\left((\phi_{i+1}^n + \phi_{i-1}^n) - 2\phi_i^n\right)}{2(\Delta x)^2} \tag{24}$$

To analyze formula (24), we first expand ϕ_{i-1}^n using Taylor series

$$\phi_{i-1}^n = \phi_i^n - \left(\frac{\partial \phi}{\partial x}\right)_i^n \Delta x + \left(\frac{\partial^2 \phi}{\partial x^2}\right)_i^n \frac{\Delta x^2}{2} - \left(\frac{\partial^3 \phi}{\partial x^3}\right)_i^n \frac{\Delta x^3}{6} + O(\Delta x^4). \tag{25}$$

Suppose $U < 0$, substituting (16) and (25) into (24) gives

$$\frac{\partial \phi}{\partial t} + U\frac{\partial \phi}{\partial x} = \frac{U^2 \Delta t}{2}\frac{\partial^2 \phi}{\partial x^2} - \frac{U\Delta x^2}{6}\frac{\partial^3 \phi}{\partial x^3} + \frac{U^2 \Delta t \Delta x^2}{24}\frac{\partial^4 \phi}{\partial x^4} + O(\Delta x^4). \quad (26)$$

The (26) is a modeled equation of (10) by solving it using the semi-Lagrangian method with third-order interpolation. Different than the diffusion term $\frac{|U|\,\Delta\, x}{2}\frac{\partial^2 \phi}{\partial x^2}$ in (19), the numerical diffusion term $\frac{U^2 \Delta t}{2}\frac{\partial^2 \phi}{\partial x^2}$ in (26) is proportion to the temporal discretization. When $\Delta\, t \to 0$, both $\frac{U^2 \Delta t}{2}\frac{\partial^2 \phi}{\partial x^2}$ and $\frac{U^2 \Delta t \Delta x^2}{24}\frac{\partial^4 \phi}{\partial x^4}$ vanish so that (26) becomes

$$\frac{\partial \phi}{\partial t} + U\frac{\partial \phi}{\partial x} = -\frac{U\Delta x^2}{6}\frac{\partial^3 \phi}{\partial x^3}. \quad (27)$$

The additional third-order derivative term $-\frac{U\Delta x^2}{6}\frac{\partial^3 \phi}{\partial x^3}$ is called "numerical dispersion", which can generate oscillations in the simulation.

The third-order interpolation is evaluated by simulating the same sine and square waves with the same settings as those used for the linear interpolation in the previous section. Compared to Figure 1, Figure 3 shows that the third-order interpolation can significantly improve the prediction for sine waves. It also computes better amplitude for the square wave but introduces oscillations due to the numerical dispersion. Similar to the linear interpolation, we can also improve the predictions of the third-order interpolation by using smaller grids and time steps.

The above studies show that the linear interpolation can smooth the profile but damp the amplitude, while the third-order interpolation can maintain the amplitude but introduce

oscillations. To utilize the merits of both schemes, we propose a hybrid scheme of two interpolations

$$\phi(x) = \begin{cases} \phi_i + (x - x_i)\dfrac{(\phi_{i+1} - \phi_{i-1})}{2\Delta x} + \dfrac{(x - x_i)^2}{2}\dfrac{(\phi_{i+1} + \phi_{i-1} - 2\phi_i)}{(\Delta x)^2}, & (\phi_{i+1} - \phi_i)(\phi_{i+1} - \phi_{i-1}) > 0 \\ \phi_i + (x - x_i)\dfrac{\phi_{i+1} - \phi_i}{\Delta x}, & (\phi_{i+1} - \phi_i)(\phi_{i+1} - \phi_{i-1}) \le 0 \end{cases} \quad (28)$$

When the profile is monotonic, the third-order interpolation is adopted to precisely capture the amplitude. Otherwise, the linear interpolation is used to avoid possible oscillations at the non-monotonic part.

With the same settings, the hybrid interpolation method computes better results for both sine and square waves (Figure 4) than the linear interpolation (Figure 1) and the third-order interpolation (Figure 3). Thus, the hybrid interpolation seems to be a promising solution for the FFD model. Considering both cases use ideal flows that are not sufficient to assess the performance of the hybrid scheme, we will evaluate the hybrid method by simulating four different indoor.

SIMULATION RESULTS

Flow in a square lid-driven cavity

The two-dimensional flow in a square lid-driven cavity (Ghia, 1982, Kumar, 2009) is like a circulated flow in a room (Figure 5). Based on the lid velocity U_0 and square height L, the Reynolds numbers Re of the studied flows are 100 and 1000, respectively. When the

Reynolds number increases, the flow profiles change from smooth to non-smooth. Four different meshes (17 × 17, 33 × 33, 65 × 65, and 129 × 129) are used to evaluate the mesh dependence. The accuracy of the FFD model can be improved by using smaller time step size since the FFD is a first-order method in time (Hu, 2010). Thus, various time step sizes are also adopted until the solution is independent of time step size. The high quality data of Ghia et al. (Ghia, 1982) is used as reference .

Figure 6 compares the predicted horizontal velocity U along the vertical mid-section ($x = 0.5L$) at various Reynolds numbers. All the results were obtained by using a fine mesh of 129 × 129 grids with a time step of 0.005s. When Re = 100, the flow profile is smooth. Both linear and hybrid interpolation correctly compute the velocity profiles (Figure 6a). When Re = 1000, the flow profile becomes sharp (Figure 6b). The linear interpolation damps the peak values of velocity due to the large numerical viscosity. With less numerical viscosity, the hybrid interpolation properly predicts the velocity.

To measure the model performance (accuracy and computing speed) with different meshes, we define the relative error to the reference data as

$$E_{REL,i} = \frac{\phi_i - \phi_{ref,i}}{\phi_{ref,i}}, \phi = U, V, i \in [1, N], \tag{29}$$

where N is the number of sampled data and $\phi_{ref,i}$ the value reference data at point i. The averaged value or root mean square of $E_{REF,i}$ may not correctly represent the overall performance since a small $\phi_{ref,i}$ can yield huge $E_{REF,i}$. For instance, the flow velocity near the wall is almost zero and it can magnify the relative error even the absolute error is tiny.

To avoid this problem, we evaluate the accuracy using a probability of confidence interval for the relative error $E_{REF,i}$:

$$\Pr\left(|E_{REF,i}| < \delta\right), i \in [1, N], \tag{30}$$

where δ is constant. Using the probability of confidence interval, we can properly obtain the overall accuracy.

Figure 7 compares the Pr for $\delta = 10\%$ at Re = 1000. For both methods, the Pr increases if we refine the grids. The Pr of hybrid method increases at a speed much faster than the linear method. Applying a mesh with 257 × 257 grids, the hybrid interpolation can has Pr $(|E_{REF,i}| < 10\%) = 100\%$ and the Pr by the linear interpolation is less than 60%. To get Pr = 60%, the hybrid scheme only needs a grid resolution of 70 × 70 and the total number of grids is only 7% of that required by the linear interpolation. Figure 7 also compares the computing time needed by the FFD with two interpolations using different grids. For the same grid distribution, the hybrid method needs about 10% more CPU time than the linear one. But the difference of these two approaches in computing time is significantly smaller than that in accuracy. As a result, the hybrid method is able to provide more accurate results with less computing time than the linear interpolation. For instance, the hybrid method with 128 × 128 grids uses 654s of computing time to get $Pr = 97\%$. As a comparison, the linear interpolation with 256 × 256 grids needs 3290s of computing time but only has $Pr = 58\%$.

Flow over a backward facing step

The airflow that goes into a room through an open window is like a flow passing over a backward facing step. There are many experimental data (Armaly, 1983, Durst, 1985, Driver, 1985) and numerical simulations (Kim, 1985, Saldana, 2005, Barton, 1994, Le, 1997) related to this flow, although their geometries and flow conditions may vary case by case. We simulate the flow studied by Armaly et al (1983) because the flow domain is close to a vertical wall with an open window and the data has been widely used for numerical validation (Liu, 2009, Das, 2009, Lian, 2009) due to its high quality. Figure 8 gives the schematic view of the experiment. The height of the step H is half of the channel height. The small channel is long enough to ensure that the flow is fully developed when it reaches the step at $x = 0$. The Reynolds number Re is based on H and the averaged inlet velocity U_{in}.

Armaly et al (1983) measured the flow within a wide range of Reynolds numbers ($70 < Re < 8000$). They found that the flow was laminar when $Re < 1200$ and the flow pattern was two-dimensional when $Re < 400$. Since the FFD has no treatments for turbulence, we only simulate the laminar flow with $Re = 100$ and 389. To save computing time, we do not simulate the flow in the long small channel. Instead, we use a parabolic profile for laminar flow as inlet boundary condition at the intersection of the small and large channel. The computing domain is two-dimensional since the flows are two-dimensional. Four different meshes (25×25, 50×50, 100×100 and 200×200) are used to achieve a mesh independent solution.

Table 1 compares the calculated reattachment length from the step with the experimental data (Armaly, 1983). The grids are 100 × 100 for the flow at Re = 100 and 200 × 200 for Re = 389. The time step size is 0.02s for both cases. The data is normalized by H and the error is a relative error to the experimental data. Due to the numerical viscosity, the FFD model computes a shorter reattachment length for all the cases. With less numerical viscosity, the hybrid interpolation has better predictions than the linear interpolation. When the Reynolds number is small (Re = 100), the hybrid scheme computes the reattachment length precisely and the relative error is less than 1%. The linear interpolation also gets reasonable results with a relative error of 4%. When the Reynolds number increases (Re = 389), the errors increases for both methods. But the hybrid scheme is still more accurate than the linear one.

Similar to the previous cases, the hybrid scheme has better performance than the linear scheme when the grid is coarse. With the refinement of grids, the predictions of these two methods are getting closer. For instance, at Re = 389, the hybrid interpolation calculates better velocity profiles than the linear one when a coarse grid is applied (Figure 9). If the grid is fine, both methods provide similar results (Figure 10). The reason is that the numerical viscosity in the linear interpolation is proportion to grid size and using smaller grid can reduce it.

Natural convection in a square cavity

The flows in the previous two cases are isothermal. To evaluate the impact of the hybrid and linear interpolations for non-isothermal flows, a natural convective flow in a square

cavity is simulated. This case is widely used for validation of the numerical program (Davis, 1983b, Kimura, 1983, Mergui, 1996). As shown in Figure 11, the upper and lower walls are isolated from heat transfer. Temperatures on the two side walls are T_1 and T_2, respectively. To simplify the configuration, we set $T_2 = -T_1$ and the reference temperature $T_{ref} = 0$. The length of the cavity L is 0.1 m, the kinematic viscosity of fluid ν is 1.59×10^{-5} m^2/s, the thermal diffusivity α is 2.2495×10^{-5} m^2/s, the thermal expansion coefficient β is 3.0×10^{-3} 1/K, and the gravitational acceleration is 10.0 m/s^2. Thus, the Rayleigh number Ra only depends on the temperature difference $\Delta T = T_1 - T_2$.

The two interpolations are compared by using flows in three different Rayleigh numbers ($Ra = 10^3$, 10^4 and 10^5). When the $Ra = 10^5$, the convection plays a more important role in the heat transfer than the diffusion. To evaluate the mesh dependence, all the simulations are conducted on four various meshes (16 × 16, 32 × 32, 64 × 64 and 128 × 128).

To compare with the benchmark solution, we normalize the results as follows

$$X = \frac{x}{L}, Y = \frac{y}{L}, U = \frac{uL}{a}, V = \frac{vL}{a}, \tag{31}$$

where X and Y are normalized coordinates at x and y directions, respectively. U and V are normalized velocities for u and v, respectively. Another important result is the averaged Nusselt number $\overline{Nu}$:

$$\overline{Nu} = \frac{\bar{h}L}{k}, \tag{32}$$

where $\bar{h}$ is the averaged heat transfer coefficient and k the thermal conductivity. To calculate $\overline{Nu}$, we need to get $\bar{h}$ from the equation for heat transfer between the fluid and the wall

$$\bar{h}(T_1 - T_2)A_S = \int_{A_S} q'' dA_S = -k \int_{A_S} \frac{dT}{dx} dA_S, \tag{33}$$

where q'' is the heat flux and A_S the area of vertical surface. Substituting (33) into (32) leads to

$$\overline{Nu} = \frac{L}{T_2 - T_1} \frac{\int_{A_S} \frac{dT}{dx} dA_S}{A_S}. \tag{34}$$

Table 2 compares the simulated results using the finest mesh (128 × 128) and a time step size of 0.025s with the benchmark solution (Davis, 1983a). The U_{max} value is the maximum value of U velocity on the central vertical plane and Y_{max} is the position of U_{max}. Similarly, V_{max} and X_{max} are the maximum V velocity and its position on the central horizontal plane. The FFD with both linear and hybrid interpolations provide reasonable solutions for U_{max}, V_{max}, X_{max}, Y_{max}, and $\overline{Nu}$. The hybrid interpolation computes the results generally closer to the benchmark data than the linear interpolation. But the difference between the hybrid interpolation and the linear interpolation is not significant. Similar phenomena are observed in a comparison of predicted temperature contours (for instance, $Ra = 10^5$ in Figure 12). Using the finest grids, the temperature contours computed by the linear and hybrid interpolations are close to each other.

Figure 13 compares the relative errors of $\overline{Nu}$ calculated by the FFD with two interpolations using different grid resolutions for different flows. Similar to previous cases, the hybrid interpolation computes a solution with less error than the linear scheme when the grid resolution is the same. One exception is the flow at $Ra = 10^5$ with the grid of 16 × 16. The liner interpolation provides slightly better results than the hybrid interpolation. The reason is that the FFD model is not able to properly capture the flow property with such coarse grid. For instance, the relative errors are larger than 20% for both methods. Thus, it is not meaningful to compare the interpolations methods if the FFD model fails. As long as the FFD model works, using the hybrid interpolation can provide higher accuracy than the linear interpolation.

Three-dimensional flow in a cubic lid-driven cavity

The flows studied in previous three cases are two-dimensional. To study the impact of hybrid interpolation in predicting three dimensional flows, we now evaluate it using a three-dimensional flow in a cubic lid-driven cavity (Figure 14) with the reference data from Ku et al (Ku, 1987). The size of the cavity is 1 m × 1 m × 1 m and the Reynolds number is 1000.

Figure 15 presents the predicted vertical velocity V along the horizontal centerline by using different grid resolutions. The FFD predicts the profiles with large errors using a coarse grid (30 × 30 × 30) and a time step size of 0.1 s (Figure 15a). Similar to other cases, we can improve the predictions (Figure 15b) by using a finer grid (60 × 60 × 60) and a smaller time

step size (0.01 s). With less numerical viscosity, the hybrid interpolation can provide better predictions than the linear interpolation for both settings.

DISUCSSIONS

The current formulation of the hybrid interpolation is only for structured uniform mesh. The formula will need to be modified if it is applied for non-uniform structured meshes and unstructured meshes. However, the principle of the hybrid scheme of low and high order interpolation should remain the same.

Since the FFD had no turbulence model, we only studied the performance of hybrid schemes for laminar flows. However, most indoor airflow is turbulent (Lariani, 2009). Our previous work (Zuo, 2009) tried to predict turbulent flow by adding turbulence models in the FFD. Unfortunately, it did not provide satisfactory results due to the large numerical viscosity in the linear interpolation. Considering the FFD with the hybrid interpolation has significantly less numerical viscosity, it is possible to further improve its accuracy for turbulent flow by adding turbulence models.

Various approaches can be taken to accelerate the computing speed of the FFD model. The current FFD is a first-order method in time. Thus, it is possible to apply higher order temporal scheme in the FFD so that it can use larger time step sizes without losing accuracy. We can also apply advanced numerical schemes, such as the multigrid method (Kumar, 2010), to reduce the computing time. In addition, parallelization is another way to

speed up the simuation. For instance, we have acclerated the FFD simulation 30 times by computing in parallel on a graphics processing unit (Zuo, 2010a).

CONCLUSIONS

This study proposed a hybrid scheme of linear and third-order interpolations to reduce both the numerical viscosity in the FFD model. The results show that:

With the same special and temporal resolutions, the hybrid interpolation can provide more accurate results than the linear interpolation, but it needs about 10% more computing time.

When refining the grid resolution, the error of FFD model using hybrid interpolation can be reduced at a faster rate than that using the linear interpolation.

The FFD model with hybrid interpolation can achieve the same accuracy with much less grids and computing time compared to the FFD with the liner interpolation.

ACKONWLEDGEMENT

Wangda Zuo was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Building Technologies of the U.S. Department of Energy, under Contract No. DE-AC02-05CH11231. He also wants to thank the support of the visiting scholar program at Chongqing University's Key Laboratory of Three Gorges Reservoir Region' s Eco-Environments under Ministry of Education.

Mingang Jin and Qingyan Chen would like to thank the U.S. Federal Aviation Administration (FAA) Office of Aerospace Medicine for funding this project through the National Air Transportation Center of Excellence for Research in the Intermodal Transport Environment under Cooperative Agreement 10-C-RITE-PU. Although the FAA has sponsored this project, it neither endorses nor rejects the findings of this research. The presentation of this information is in the interest of invoking technical community comment on the results and conclusions of the research.

REFERENCES

Armaly, B.F., Durst, F., Pereira, J.C.F. and Schonung, B. (1983) "Experimental and Theoretical Investigation of Backward-Facing Step Flow", *Journal of Fluid Mechanics*, **127**, 473-496.

Ashrae (1995) Thermal Environmental Conditions for Human Occupancy, In: Ashrae (ed), Vol. ASHRAE Standard 55a-1995.

Axley, J. (2007) "Multizone Airflow Modeling in Buildings: History and Theory", *HVAC&R Research*, **13**, 907-928.

Barton, I.E. (1994) "Laminar-Flow Past an Enclosed and Open Backward-Facing Step", *Physics of Fluids*, **6**, 4054-4056.

Chen, Q. (2009) "Ventilation performance prediction for buildings: A method overview and recent applications", *Building and Environment*, **44**, 848-858.

Chorin, A.J. (1967) "A Numerical Method for Solving Incompressible Viscous Flow Problems", *Journal of Computational Physics*, **2**, 12-26.

Courant, R., Friedrichs, K. and Lewyt, H. (1928) "Uber die partiellen differenzgleichungen der mathematischen physik", *Mathematische Annalen*, **100**, 32-74.

Das, M.K. and Devangre, R.B. (2009) "Conjugate Mixed Convection Heat Transfer in Plane Laminar Wall Jet Flow", *Numerical Heat Transfer Part a-Applications*, **56**, 60-75.

Davis, G.D. (1983a) "Natural-Convection of Air in a Square Cavity - a Bench-Mark Numerical-Solution", *International Journal for Numerical Methods in Fluids*, **3**, 249-264.

Davis, G.D. and Jones, I.P. (1983b) "Natural-Convection in a Square Cavity - a Comparison Exercise", *International Journal for Numerical Methods in Fluids*, **3**, 227-248.

Driver, D.M. and Seegmiller, H.L. (1985) "Features of a Reattaching Turbulent Shear-Layer in Divergent Channel Flow", *AIAA Journal*, **23**, 163-171.

Durst, F. and Schmitt, F. (1985) "Experimental Studies of High Reynolds Number Backward-Facing Step Flow". In: Proceedings of the 5th Symposium on Turbulent Shear Flows, pp. 519-524.

Fedkiw, R., Stam, J. and Jensen, H.W. (2001) "Visual simulation of smoke". In: Proceedings of SIGGRAPH 2001, pp. 15-22.

Ferziger, J.H. and Peric, M. (2002) *Computational methods for fluid dynamics,* Berlin, New York, Springer.

Ghia, U., Ghia, K.N. and Shin, C.T. (1982) "High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method", *Journal of Computational Physics*, **48**, 387-411.

Hu, J., Zuo, W. and Chen, Q. (2010) "Impact of time-splitting schemes on the accuracy of FFD simulations". In: Proceedings of the 7th International Indoor Air Quality, Ventilation and Energy Conservation in Buildings Conference (IAQVEC 2010), pp. 55-60.

Kim, J. and Moin, P. (1985) "Application of a Fractional-Step Method to Incompressible Navier-Stokes Equations", *Journal of Computational Physics*, **59**, 308-323.

Kimura, S. and Bejan, A. (1983) "The Heatline Visualization of Convective Heat-Transfer", *Journal of Heat Transfer-Transactions of the Asme*, **105**, 916-919.

Ku, H.C., Hirsh, R.S. and Taylor, T.D. (1987) "A Pseudospectral Method for Solution of the three-dimensional Incompressible Navier-Stokes Equations", *Journal of Computational Physics*, **70**, 439-462.

Kumar, D.S., Dass, A.K. and Dewan, A. (2010) "A Multigrid-Accelerated Code on Graded Cartesian Meshes for 2d Time-Dependent Incompressible Viscous Flows", *Engineering Applications of Computational Fluid Mechanics*, **4**, 71-90.

Kumar, D.S., Kumar, K.S. and Das, M.K. (2009) "A Fine Grid Solution for a Lid-Driven Cavity Flow Using Multigrid Method", *Engineering Applications of Computational Fluid Mechanics*, **3**, 336-354.

Lariani, A., Nesreddine, H. and Galanis, N. (2009) "Numerical and Experimental Study of 3d Turbulent Airflow in a Full Scale Heated Ventilated Room", *Engineering Applications of Computational Fluid Mechanics*, **3**, 1-14.

Le, H., Moin, P. and Kim, J. (1997) "Direct Numerical Simulation of Turbulent Flow Over a Backward-Facing Step", *J. Fluid Mechanics*, **330**, 349-374.

Leonard, B.P. (1979) "Stable and Accurate Convective Modelling Procedure Based on Quadratic Upstream Interpolation", *Computer Methods in Applied Mechanics and Engineering*, **19**, 59-98.

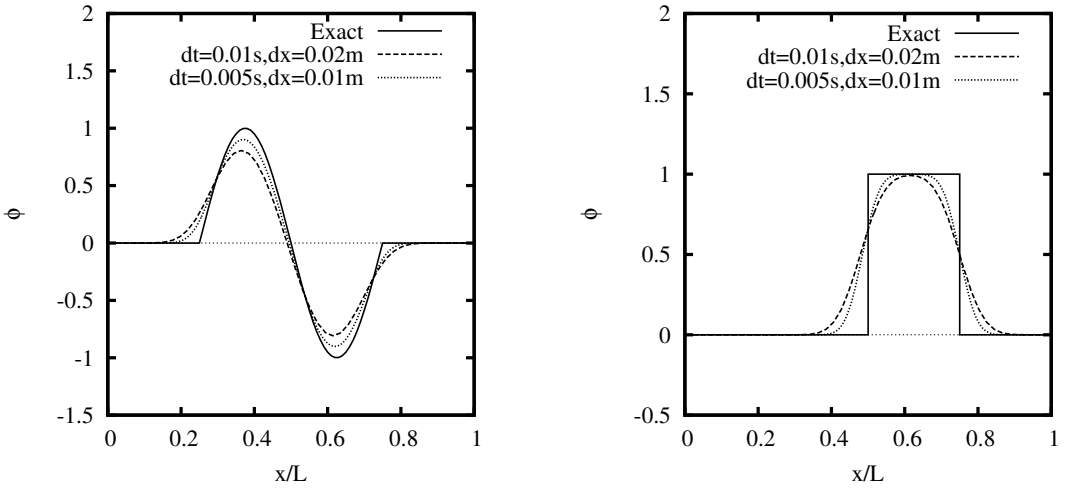
Lian, C.Z., Xia, G.P. and Merkle, C.L. (2009) "Solution-limited time stepping to enhance reliability in CFD applications", *Journal of Computational Physics*, **228**, 4836-4857.

Liu, J. (2009) "Open and traction boundary conditions for the incompressible Navier-Stokes equations", *Journal of Computational Physics*, **228**, 7250-7267.

Mergui, S. and Penot, F. (1996) "Natural convection in a differentially heated square cavity: Experimental investigation at Ra=1.69x10(9)", *International Journal of Heat and Mass Transfer*, **39**, 563-574.

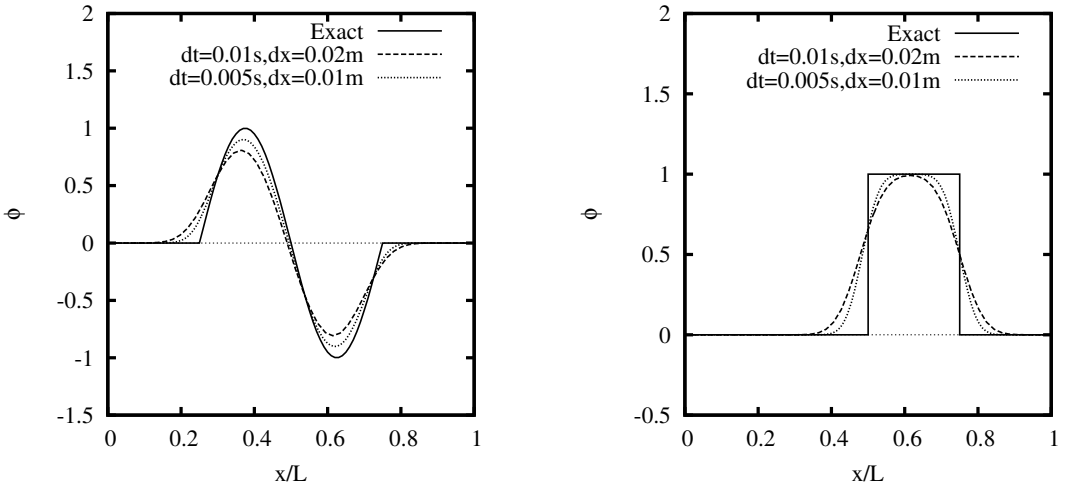
Molemaker, J., Cohen, J.M., Patel, S. and Yong Noh, J. (2008) Low Viscosity Flow Simulations for Animation, *Symposium on Computer Animation*.

Robert, A. (1981) "A stable numerical integration scheme for the primitive meteorological equations", *Atmosphere Ocean*, **19**, 35-46.
Saldana, J.G.B., Anand, N.K. and Sarin, V. (2005) "Numerical Simulation of Mixed Convective Flow Over a Three-Dimensional Horizontal Backward Facing Step", *J. Heat Transfer*, **127**, 1027-1036.
Song, O.-Y., Shin, H. and Ko, H.-S. (2005) "Stable but nondissipative water", *ACM Transactions on Graphics*, **24**, 81-97.
Stam, J. (1999) "Stable Fluids". In: Proceedings of 26th International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'99), pp. 121-128.
Zuo, W. and Chen, Q. (2009) "Real-time or faster-than-real-time simulation of airflow in buildings", *Indoor Air*, **19**, 33-44.
Zuo, W. and Chen, Q. (2010a) "Fast and informative flow simulations in a building by using fast fluid dynamics model on graphics processing unit", *Building and Environment*, **45**, 747-757.
Zuo, W., Hu, J. and Chen, Q. (2010b) "Improvements in FFD Modeling by Using Different Numerical Schemes", *Numerical Heat Transfer Part B-Fundamentals*, **58**, 1-16.



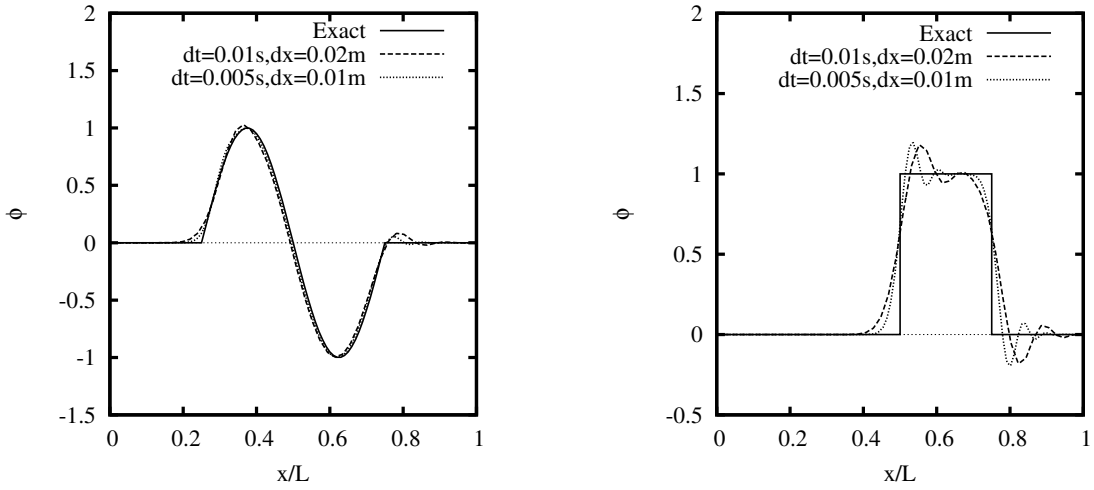
(a) sine wave (b) square wave

Figure 1 Predicted time advection of sine and square waves by the semi-Lagrangian method with linear interpolation.



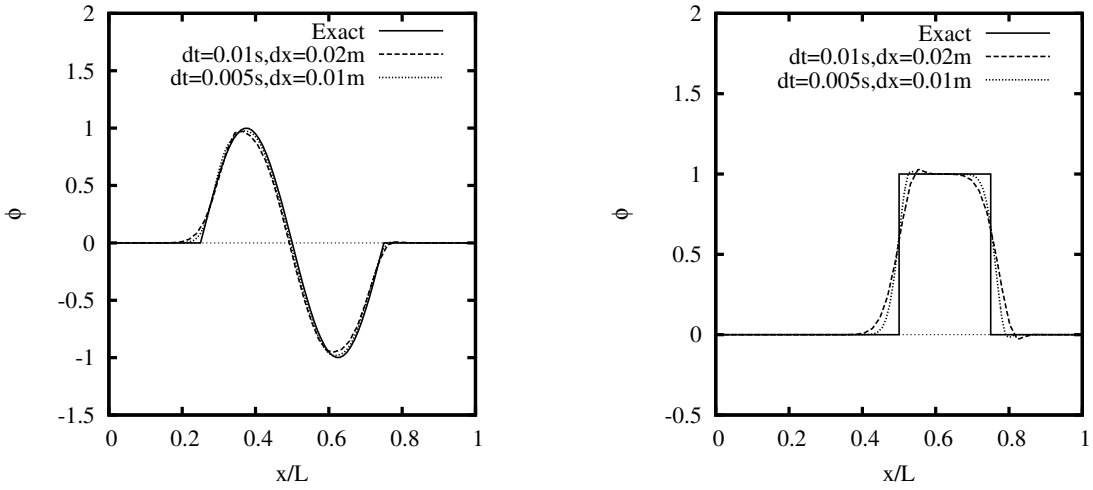
(a) sine wave (b) square wave

Figure 2 Predicted time advection of sine and square waves by the first-order upwind scheme.



(a) sine wave (b) square wave

Figure 3 Predicted time advection of sine and square waves by the semi-Lagrangian method with third-order interpolation.



(a) sine wave (b) square wave

Figure 4 Predicted time advections of sine and square waves by using the semi-Lagrangian method with hybrid interpolation.

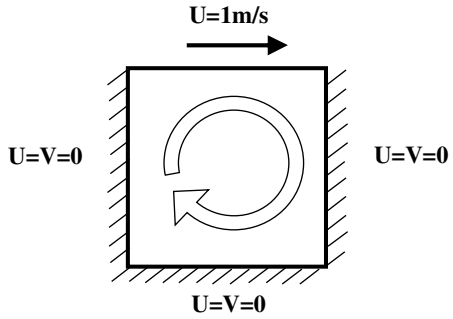
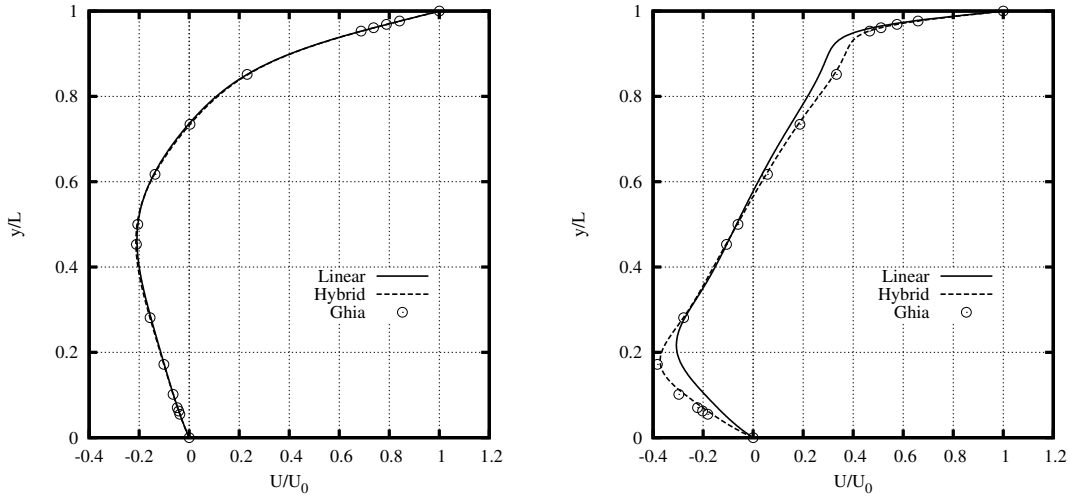


Figure 5 Schematic view of a flow in a lid-driven cavity



(a) U at Re =100 (b) U at Re =1000

Figure 6 Comparison of velocity profiles at different Reynolds numbers computed by the FFD with linear and hybrid interpolations with a mesh of 129 x 129.

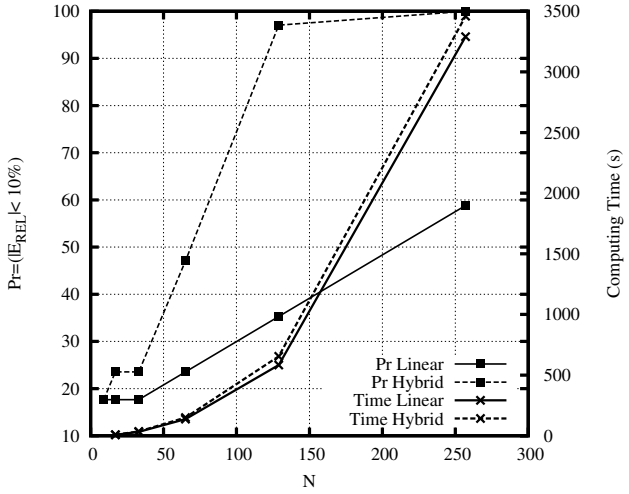


Figure 7 Comparison of probabilities of confidence interval for relative error less than 10% in predictions of linear and hybrid interpolation at $Re = 1000$. The grid resolution is $N \times N$.

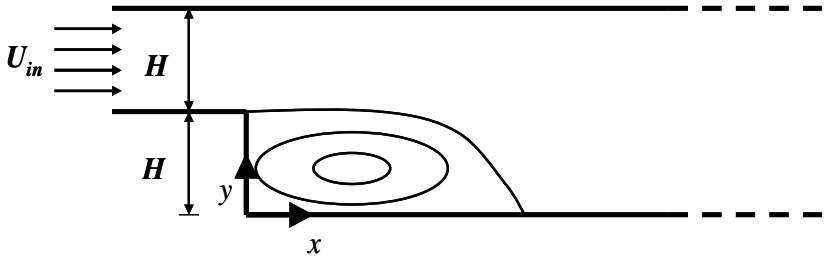
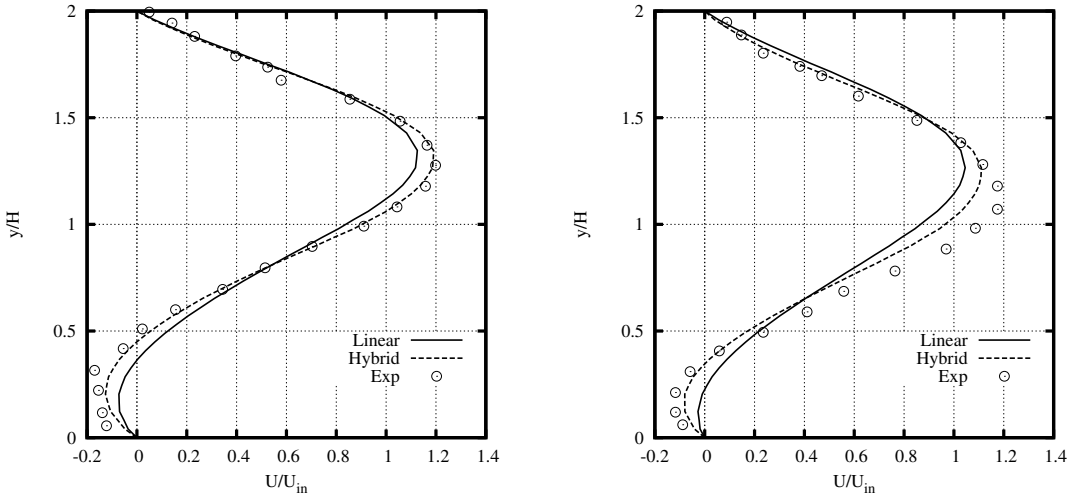
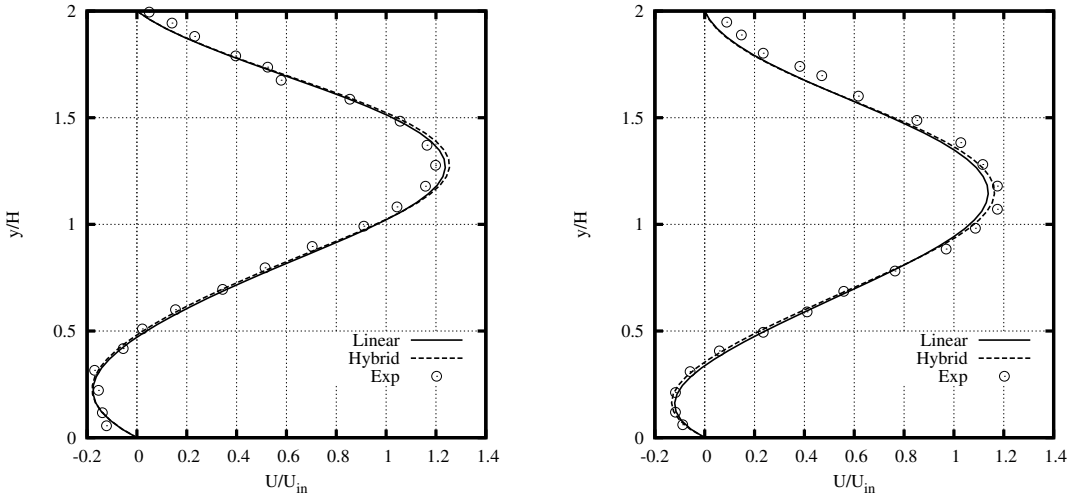


Figure 8 Schematic view of a flow passing the backward facing step.



(a) $x = 4.80H$ (b) $x = 6.12H$

Figure 9 Comparison of computed horizontal velocities of backward facing step flow at $Re = 389$ by using a coarse mesh with 25×25 grids.



(a) $x = 4.80H$ (b) $x = 6.12H$

Figure 10 Comparison of computed horizontal velocities of backward facing step flow at $Re = 389$ by using a fine mesh with 200×200 grids.

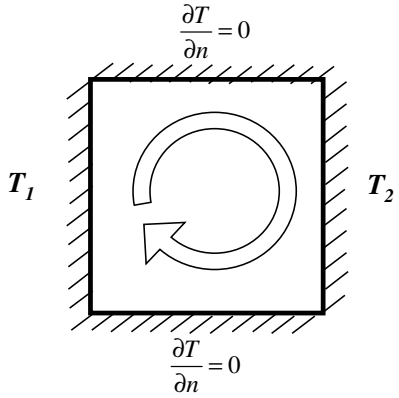
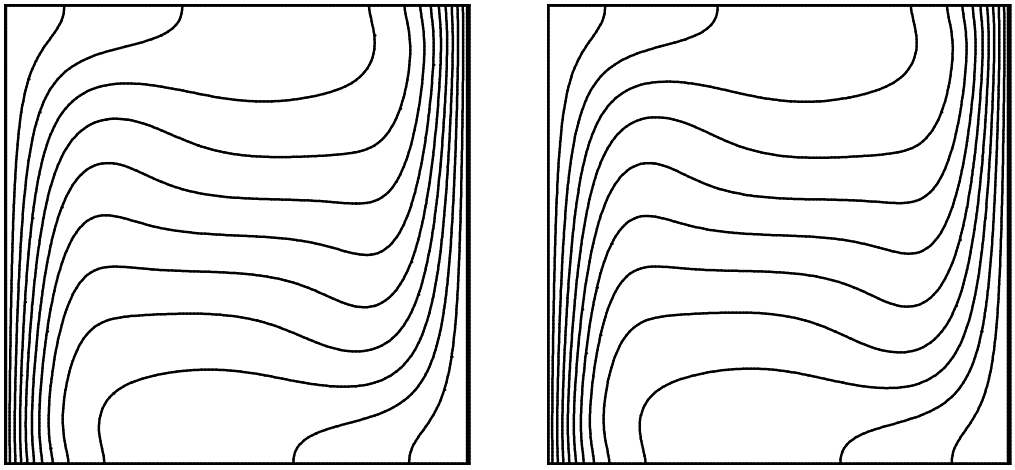
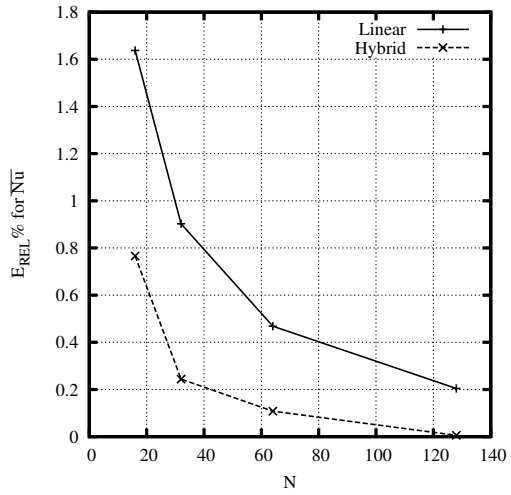


Figure 11 Sketch of the natural convection flow in a square cavity.

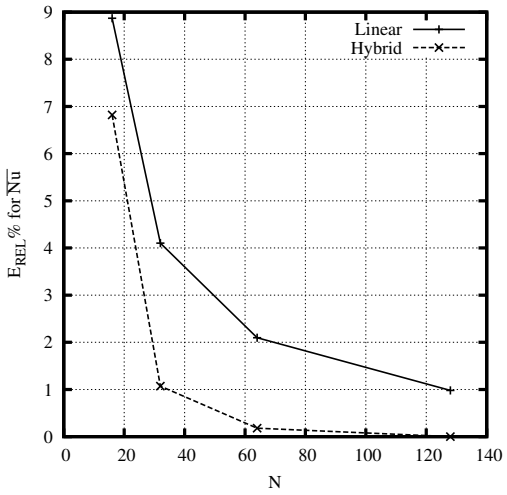


(a) Linear (b) Hybrid

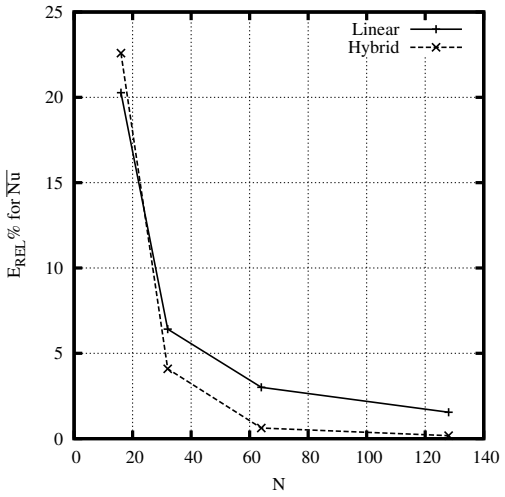
Figure 12 Comparison of temperature contours at $Ra = 10^5$.



(a) $Ra = 10^3$



(b) $Ra = 10^4$



(c) $Ra = 10^5$

Figure 13 Comparison of relative error in averaged Nusselt number predicted by the linear and hybrid interpolations with different grids for different Rayleigh numbers. Grid resolution is $N \times N$.

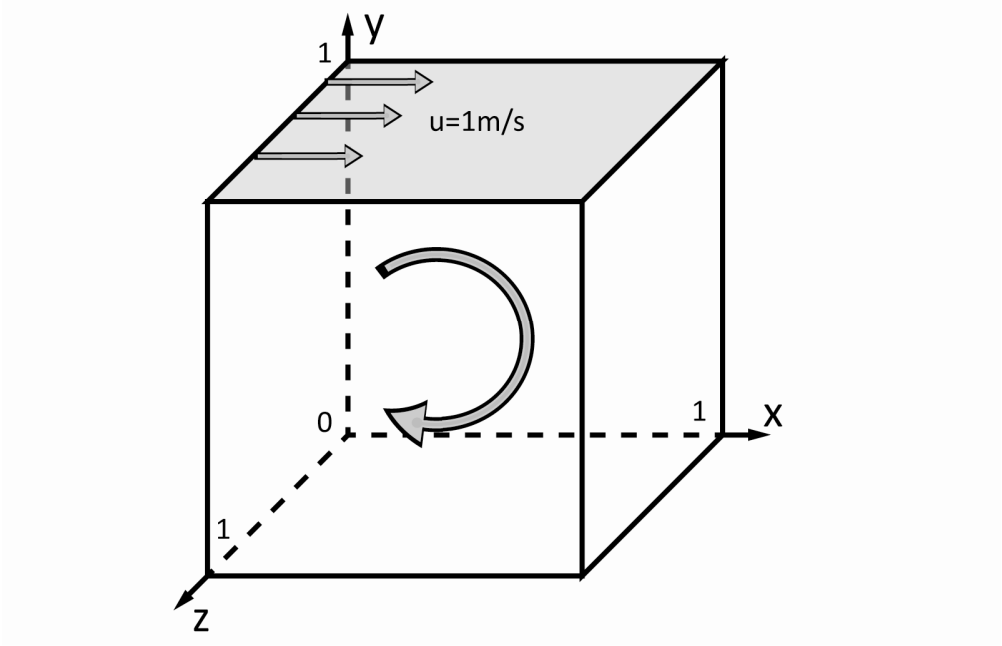
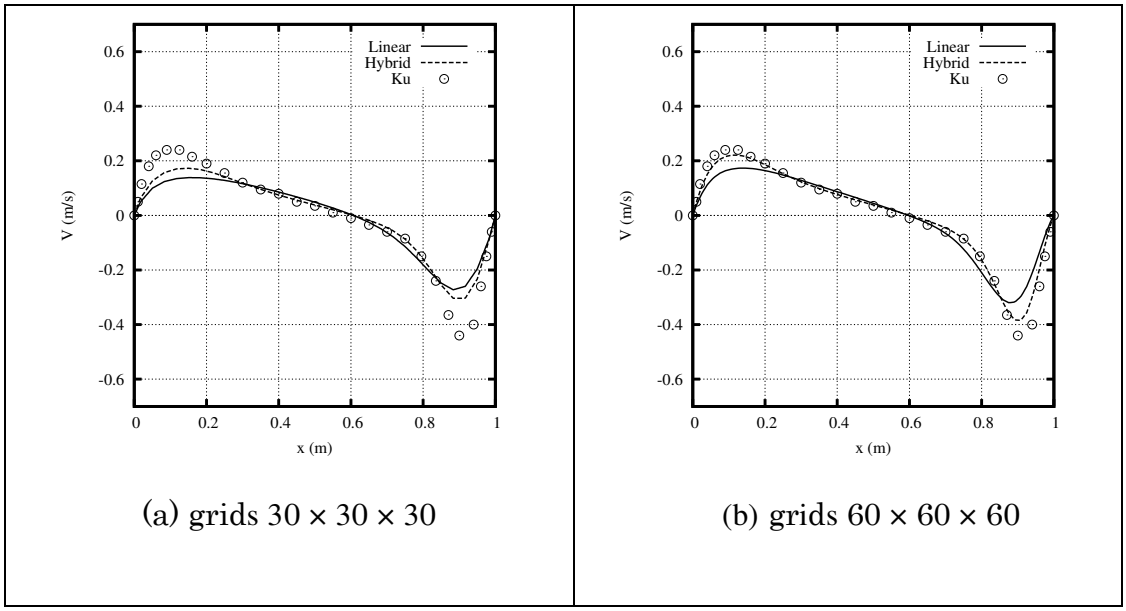


Figure 14 Schematic of the flow in a cubic lid-driven cavity



(a) grids 30 × 30 × 30

(b) grids 60 × 60 × 60

Figure 15 Comparison of velocity V along the horizontal centerline in the cubic lid-driven cavity predicted by the FFD with hybrid and linear interpolation schemes with different grid resolutions. The reference data is from Ku et al. (Ku, 1987)

Table 1 Comparison of normalized reattachment lengths computed by the FFD with different interpolation schemes.

Re	Experiment	Bilinear	Error	Hybrid	Error
100	2.99	2.87	-4.0%	2.97	-0.7%
389	8.59	7.80	-9.2%	7.95	-7.5%

Table 2 Comparison of simulation results with the finest mesh.

$Ra = 10^3$			
Variable	Linear	Hybrid	Benchmark
U_{max}	3.634	3.647	3.649
Y_{max}	0.815	0.815	0.813
V_{max}	3.687	3.705	3.697
X_{max}	0.178	0.178	0.178
$\overline{Nu}$	1.116	1.118	1.118
$Ra = 10^4$			
U_{max}	16.201	16.167	16.178
Y_{max}	0.822	0.822	0.823
V_{max}	19.613	19.662	19.617
X_{max}	0.123	0.115	0.119
$\overline{Nu}$	2.222	2.243	2.243
$Ra = 10^5$			
U_{max}	35.54	34.85	34.73
Y_{max}	0.855	0.855	0.855
V_{max}	69.39	68.86	68.59
X_{max}	0.068	0.068	0.066
$\overline{Nu}$	4.449	4.514	4.519