Simulation of Smoke in OpenFOAM Framework

Lakshman Anumolu and Imaduddin Ahmed
ME964-Fall 2012
University of Wisconsin Madison

December 21, 2012

In this work, we present the simulation of smoke in OpenFOAM with various approximations made on smoke particles like dusty gas approximation and equilibrium approximation [Druzhinin, 1994]. Results for 3D test cases are presented and are rendered using mitsuba [Jakob, 2010].

1 Introduction

For the past several years, there has been a lot of work in modelling complex simulations pertaining to motion of gases. In the work of [Fedkiw et al., 2001], a new approach to numerically simulate smoke was developed which incorporates the model proposed by Steinhoff and Underhill [Steinhoff and Underhill, 1994] known as "vorticity confinement". Passive advection of smoke particles is considered in the works published thus far in the field of computer graphics, to the best of our knowledge. In this work we will show the effect of particle mass which affects the extent to which the motion of smoke is influenced by the underlying flow field. We will show these results with the help of particle time constant (τ_p).

2 Modelled Approach

The numerical approach developed in [Fedkiw et al., 2001] treats the smoke particles as passive particles moving with the flow and in order to achieve realistic gas flow, they adapt the approach developed by [Steinhoff and Underhill, 1994]. The incompressible Euler equations that are used to model smoke's velocity are given by

$$\nabla \cdot \mathbf{U} = 0, \tag{1}$$

$$\frac{\partial \mathbf{U}}{\partial t} + (\mathbf{U} \cdot \nabla)\mathbf{U} = -\nabla p + \mathbf{f}_{external}, \tag{2}$$

where $\mathbf{f}_{external}$ accounts for external forces like buoyance and vorticity confinement forces. Buoyancy forces due to gravity and temperature are modelled in $\mathbf{f}_{external}$ [Fedkiw et al., 2001]. In order to account for turbulent structures in smoke motion, vorticity confinement force was adapted from [Steinhoff and Underhill, 1994], and is given by

$$\mathbf{f}_{conf} = \epsilon_{conf} \Delta x \left(\frac{\nabla|\nabla \times \mathbf{U}|}{|\nabla|\nabla \times \mathbf{U}||} \times (\nabla \times \mathbf{U}) \right), \tag{3}$$

where the positive dimensional constant ϵ_{conf} controls the amount of small scale detail added back into the flow field. In our work we have chosen a value of 0.8 s^{-1} for this constant.

2.1 Numerical Implementation

The solution procedure to solve momentum equation given by Eq. (2) is implemented in two different ways into OpenFOAM [OpenCFD-Ltd, 2012]. In one formulation, we included external forces in cell-centered form and in another, we considered face-centered formulation in aim of reducing the instabilities that might result in numerical simulation. In case of cell-centered formulation, the velocity predictor step is achieved by solving the following equation in finite volume formulation,

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot ((\mathbf{U} \otimes \mathbf{U}) = \mathbf{f}_{external}. \tag{4}$$

Details regarding finite volume formulation are presented in [Deshpande et al., 2012], though in their work the authors described the implementation for two-phase flows, the numerical approach extends to solve the velocity predictor equation Eq. (4). Velocity flux is computed using the obtained predicted velocity and is used to correct the velocity flux field by solving the pressure Poisson equation [Deshpande et al., 2012].

As a second approach, we included the external forces in face formulation, i.e. external force $\mathbf{f}_{external}$ is excluded from Eq. (4) and is included while computing velocity flux field. This is then proceeded by correcting the velocity flux field using PISO iterations.

The above described two approaches are tested on a test case. The test case contains domain of dimensions $5m \times 5m \times 5m$, and fluid is injected into the domain through a circular inlet of diameter $0.5m$ from the bottom of the domain with a velocity $3m/s$. We arbitrarily considered density of smoke to be 1, which is injected at the inlet and tracked it as a group of passive particles. Figure 1 shows the computational domain consisting of $80 \times 80 \times 80$ grid cells. A solid sphere of radius $0.75m$ is placed at the center of the domain, i.e. directly above the inlet.

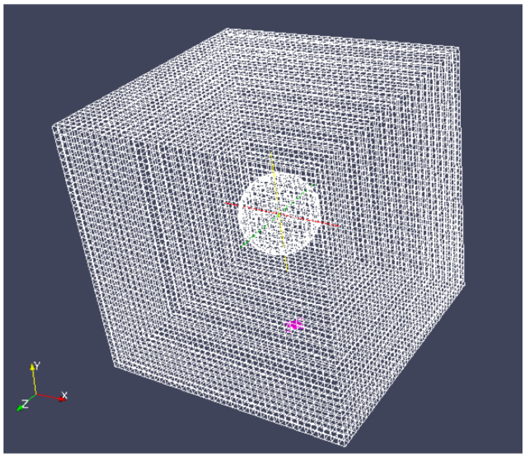
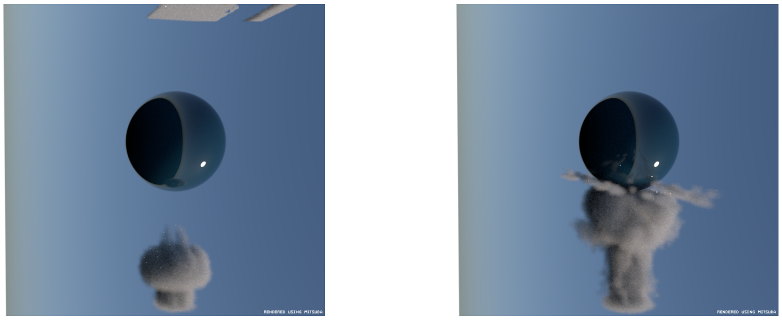


Figure 1: Computational grid with sphere centered at the center of the domain and the inlet colored in pink.

Figures 2a and 2b show the results at times $t = 0.55$s and $t = 1.30$s for cell-centered and face-centered formulations respectively. Though the face-centered formulation continued till $t = 1.30$s, both cases crashed after this time, indicating that the considered formulation could not completely aleviate the issue of numerical errors. Hence we considered using OpenFOAM's inbuilt momentum formulations which will be described in the coming sections.



(a) cell-centered formulation

(b) face-centered formulation

Figure 2: Simulations obtained when the momentum equation (Eq. (2)) is solved using (a) cell centered formulation and (b) face centered formulation. *The rendered artifacts in the top of the images are erroneous and therefore should be ignored.*

3 Solving the Navier-Stokes Equation

As described above, in this section, we solve the following momentum equation which is already implemented into OpenFOAM,

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot ((\mathbf{U} \otimes \mathbf{U}) = -\nabla p + \nabla \cdot (\nu \nabla \mathbf{U}) \, . \tag{5}$$

The velocity field obtained after solving Eq. (5) is used to advect the smoke. In the results presented here we considered $1.33 \times 10^{-5} \mathrm{m}^2/\mathrm{s}$ for ν. As mentioned previously, we used a density field (ρ_{smoke}) of '1' to track the smoke. In our work we considered two cases to advect the smoke field. In the first the smoke is made of passive particles, i.e. the density field is advected by using the velocity field obtained from Eq. (5). In the other case, we used an equilibrium approximation [Druzhinin, 1994], in which the density field is advected by using the following velocity field

$$\mathbf{U}_{eq} = \mathbf{U} - \tau_p \left(\frac{\partial \mathbf{U}}{\partial t} + \mathbf{U} \cdot \nabla \mathbf{U} - \mathbf{g} \right) , \tag{6}$$

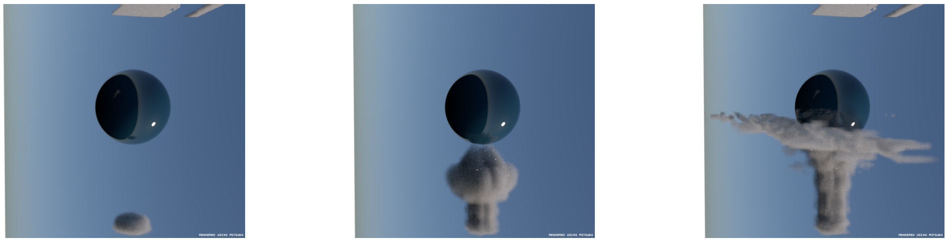
where τ_p denotes the particle time constant and $\mathbf{g}$ is the acceleration due to gravity which is $(0, -9.81, 0) m/s^2$.

Special care is needed to advect smoke using Eq. (6) in OpenFOAM, i.e. we must use the MULES formulation in OpenFOAM to make sure the values of density field (which tracks smoke) must be bounded. In order to make this implementation, we use the following advection equation for the smoke's density field

$$\frac{\partial \rho_{smoke}}{\partial t} + \nabla \cdot (\mathbf{U}_{eq} \rho_{smoke}) = \rho_{smoke} \nabla \cdot \mathbf{U}_{eq}. \tag{7}$$

Note that the right hand side term in Eq. (7) does not go to zero, hence it is not neglected in this work. Equation (7) is solved using finite volume formulation and first order Euler method is used for time integration.

In this work, we again consider the same test case described in sec. 2.1 and the results with various τ_p values i.e. $\tau_p = 0, 0.0001$ and 0.01 are shown in figs. 3, 4 and 5. Although the rendered images shown in figs. 3b, 4b and 5b cannot be easily distinguished, a clear distinction can be seen from fig. 6, where the smoke with particle time constant of 0.01 did not reach the sphere. This is because the heavier particles take more time to respond to the flow.



(a) t=0.15 s

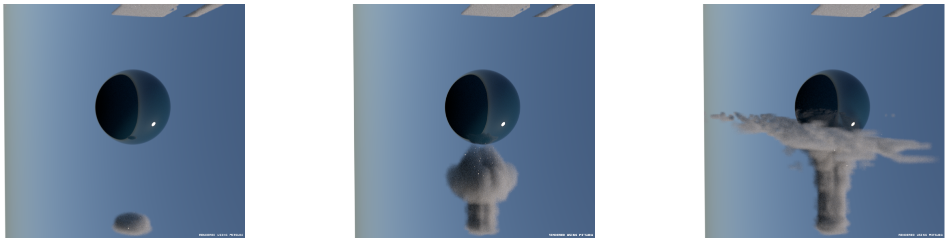
(b) t=1.00 s

(c) t=1.85 s

Figure 3: Simulation at different times for $\tau_p = 0$. *The rendered artifacts in the top of the images are erroneous and therefore should be ignored.*

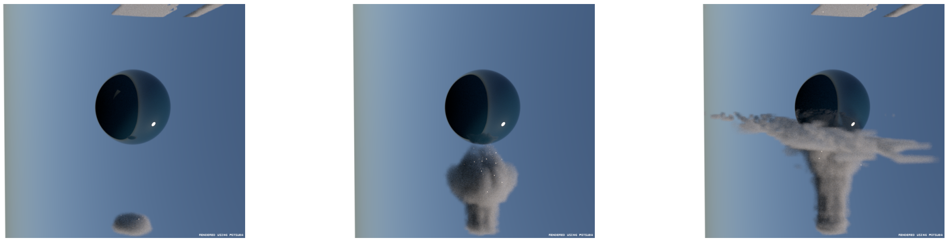
4 Conclusions and Future work

Smoke simulation is performed by using two different methods in the framework of OpenFOAM. In one approach, we implemented the modelled momentum equation proposed in [Fedkiw et al., 2001], but the finite volume formulation that is implemented showed instabilities in the results. This motivated us to move to using existing momentum solution procedure, where we used both the dusty gas approximation and the equilibrium approximation to advect smoke. Results suggest that, smoke made of heavier particles take



(a) t=0.15 s (b) t=1.00 s (c) t=1.85 s

Figure 4: Simulation at different times for $\tau_p = 0.0001$. *The rendered artifacts in the top of the images are erroneous and therefore should be ignored.*



(a) t=0.15 s (b) t=1.00 s (c) t=1.85 s

Figure 5: Simulation at different times for $\tau_p = 0.01$. *The rendered artifacts in the top of the images are erroneous and therefore should be ignored.*

longer time to respond to the flow. Future work can focus on simulating similar cases with a larger time constant, where a significant lag in particle response to the flow field can be seen.

5 Acknowledgements

The authors would like to acknowledge Suraj Deshpande for giving a demo on making a mesh using Pointwise. The authors would also like to acknowledge Professor Mario Trujillo because of whom we could use the cluster Nieves to run some of the cases for this project. Finally, the authors would like to acknowledge Joshua Leach for maintaining the cluster.

References

S.S. Deshpande, L. Anumolu, and M.F. Trujillo. Evaluating the performance of the two-phase flow solver interfoam. *Computational Science and Discovery*, 5:014016–36, 2012.

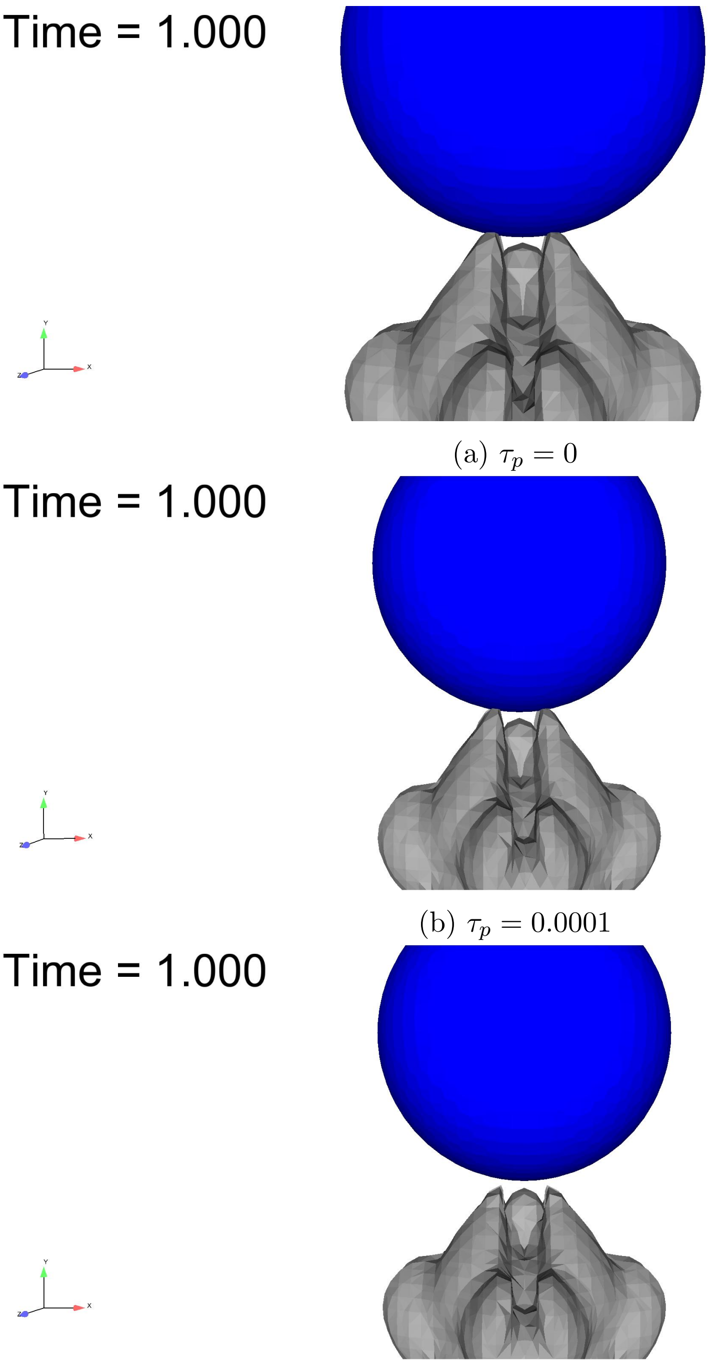
O.A. Druzhinin. Concentration waves and flow modification in a particle-laden circular vortex. *Physics of Fluids*, 6:3276–3284, 1994.

R. Fedkiw, J. Stam, and H.W. Jensen. Visual simulation of smoke. In *SIGGRAPH '01, Proceedings of the* 28^{th} *annual conference on Computer graphics and interactive techniques*, 2001.

Wenzel Jakob. Mitsuba renderer, 2010. http://www.mitsuba-renderer.org.

OpenCFD-Ltd. Openfoam 2.1.1, 2012. http://www.openfoam.org/.

J. Steinhoff and D. Underhill. Modification of the euler equations for "vorticity confinement": Application to the computation of interacting vortex rings. *Physics of Fluids*, 8:2738–2744, 1994.



(a) $\tau_p = 0$

(b) $\tau_p = 0.0001$

(c) $\tau_p = 0.01$

Figure 6: Closeup during collision with sphere.