Plume-SPH 1.0: A 3D dust-gas volcano plume model based on smoothed particle hydrodynamics

Zhixuan Cao[1], Abani Patra[1], Marcus Bursik[2], E. Bruce Pitman[3], and Matthew Jones[4]

[1]Department of Mechanical and Aerospace Engineering, University at Buffalo, SUNY, New York, United State
[2]Department of Geology, University at Buffalo, SUNY, New York, United State
[3]Department of Mathematics, University at Buffalo, SUNY, New York, United State
[4]Center for Computational Research, University at Buffalo, SUNY, New York, United State

Correspondence to: Abani Patra (abani.patra@gmail.com)

Abstract. Volcanic ash, if persists in the atmosphere, will pose great threat on aircrafts and communities. VATDs (Volcanic ash transport and dispersion models) usually take output of plume models as source terms. The accuracy of source terms is crucial for VATDs. All of existing 3D (three dimensional) plume models are using mesh based methods. SPH (Smoothed particle hydrodynamics), as a mesh free method has several advantages over mesh based methods in modeling of multiphase free boundary flow. As an initial effort on exploiting the feasibility and advantage of SPH in volcano plume modeling we adopt a relative simpler model, a 3D dusty-gas dynamic model, targeting at capturing the more basic features of volcano plume. In this model, erupted material is assumed to be well mixed and represented by one phase while air is another phase. Immediate dynamic and thermal dynamic equilibrium between air and erupted material are assumed. Effect of wind is not considered. Several newly developed techniques in SPH are adopted and trimmed to address numerical challenges in simulating multiphase compressible turbulent flow with classical SPH method. The $SPH-\varepsilon$ turbulence model is adopted to capture sub-particle mixing. Heat exchange due to turbulence is then calculated with Reynolds analogy. Corrected formulism of SPH is used to handle tensile instability and deficiency of particle distribution near the boundaries. We also propose feasible ways to impose velocity inlet and pressure outlet boundary conditions, both of which are scarce in traditional implementations of SPH. The core solver of our model is parallelized with MPI (message passing interface) obtaining good weak scalability and strong scalability. The model is first verified by comparing velocity and concentration distribution along axis and cross section with experimental results of JPUE (jet or plume which is ejected from a nozzle into a uniform environment). The top height of Pinatubo eruption (15 June 1991) simulated by our model is consistent with both observation and existing 3D plume models. Profiles of several integrated variables are compared with these by existing 3D plum models and further verifies the correctness of our model. Analysis on the plume evolution process illustrated that this model is able to reproduce the basic physics of plume development. The comparison also implies that turbulence model plays a significant role in 3D volcano plume modeling.

1 Introduction

1.1 Volcanic ash hazards

The dynamics of explosive eruption clouds has been the central issue of volcanology science for a long time. Primary hazards associated with explosive grey volcanoes include pyroclastic density currents (flows and surges), the widespread deposition of airfall tephra, and the threats posed by volcanic ash. In addition, there are also other secondary hazards associated with volcanoes (Lockwood and Hazlett, 2013). Simulation of all possible hazards with one model is difficult due to diversity among these hazards. Our concern in current work is the risks that volcanic ash poses on aircrafts and communities. During volcanic eruptions, VATDs (volcanic ash transport and dispersion models) are used to forecast the location and movement of ash clouds over hours to days. Those VATDs use eruption source parameters, such as plume height, mass eruption rate, duration, and the mass fraction distribution of erupted debris finer than about 4Φ (or 63μm), which can remain in the cloud for many hours or days. Observational data for such parameters are usually unavailable in the first minutes or hours after an eruption is detected. Moreover, these input parameters are subject to change during an eruption, requiring rapid re-assignment of new parameters. Usually, plume models are used to provide source terms for VATDs.

1.2 Existing plume models

The basic physics of volcanic plume has been understood for more than half a century (Morton et al., 1956; Settle, 1978; Wilson et al., 1978). Several volcano plume models have been developed in the past few decades. Specific aspects of a plume have been studied by 1D (one dimensional) plume models which describe the steady state solution of a plume under idealized boundary conditions. The effects of magma type and vent conditions (Woods, 1988; Woods and Bower, 1995), atmospheric conditions (Woods, 1993; Sparks et al., 1997; Bursik, 2001), external surface water (Koyaguchi and Woods, 1996), thermal disequilibrium, and particle fallout (Woods and Bursik, 1991) have been assessed in increasing details through 1D numerical simulations. Recently developed 1D models (eg. (Mastin, 2007; Woodhouse et al., 2013)) tend to account for more physics effects in one model. FPLUME-1.0 (Folch et al., 2016) accounts for wind effect, entrainment of moisture, water phase change, particle fallout and re-entrainment and even wet aggregation of ash. However, In these 1D models, the entrainment of air is evaluated based on two coefficients: entrainment coefficient for the vertical plume and the entrainment coefficient that describes the effect of wind. Different 1D models adopt different entrainment coefficients based on specific formulation or calibration against well-documented case studies. The feedback from plume to atmosphere is ignored in 1D models. Even though determination of essential parameters such as the entrainment is not based on first principles, such simple models nevertheless allow us to investigate the importance of physical mechanisms in a volcanic plume. In addition, these simplified models requires little computational resource and can run on standard personal computers or on web sites in very short time. As a result, 1D software for volcano plume development (such as (Bursik, 2010; Mastin, 2011; Vitturi, 2015)), combined with VATDs (such as (Bursik et al., 2013; Draxler and Rolph, 2015)) are widely used in research and practice. While these 1D models can generate well-matched results with 3D (three dimensional) models for weak plumes, much greater variability is observed for strong plume scenarios, especially for local variables (Costa et al., 2016). In addition, the need for greater skills in associated hazards

forecasts with 1D models requires 2D (two dimensional) and 3D models.

The development of 2D and 3D, time-dependent, and multiphase numerical models for volcano plume has provided new explanations for many features of explosive volcanism. One of these 3D models is PDAC (Pyroclastic Dispersion Analysis Code) (Neri et al., 2003) which is a non-equilibrium, multiphase, 3D compressible flow model. Conservation equations for each phase are solved separately with finite volume method. And a parallel computing version of PDAC was developed lately (Ongaro et al., 2007). Advanced numerical techniques, such as second order scheme and semi-explicit time upgrading, was also adopted afterwards to improve the accuracy of PDAC (Carcano et al., 2013). Another 3D model, SK-3D (Suzuki et al., 2005) is a three-dimensional time-dependent fluid dynamics model that can reproduce the entrainment process of eruption clouds with relative simpler physics but high order of accuracy and high spatial resolution, which is applicable to time-dependent phenomena in actual volcanological simulations. A serials of simulations based on SK-3D was conducted, including determination of the entrainment coefficients of eruption columns as a function of height (Suzuki and Koyaguchi, 2010), study the effect of wind field on entrainment coefficient (Suzuki and Koyaguchi, 2013), establishment of the relationship between the observable quantities of the eruption clouds and the eruption conditions at the vent (Suzuki and Koyaguchi, 2009) (previously, such relationship is based on 1D models and greatly depends on empirical constants), investigation of the effect of the intensity of turbulence in the umbrella cloud on dispersion and sedimentation of tephra (Koyaguchi et al., 2009). While the SK-3D focuses on accurately capturing the entrainment caused by turbulent mixture with higher resolution and numerical method of higher order, PADC takes the disequilibrium between different phases into account and hence is a true multiphase model. Another 3D model, ATHAM (Active Tracer High-Resolution Atmospheric Model) (Oberhuber et al., 1998) focuses more on microphysics within the plume. As pyroclastic flow is not the initial concern of ATHAM, immediately dynamics and thermodynamics equilibrium is assumed in ATHAM. The dynamic core of ATHAM solves the compressible Euler equations for momentum, pressure and temperature of the gas particle mixture (Oberhuber et al., 1998). The subgrid-scale turbulence closure scheme differentiates between the horizontal and vertical directions (Herzog et al., 2003) to capture turbulent mixing. The cloud microphysics predicts the mass of hydrometeors in liquid and ice phase (Herzog et al., 1998). Additional modules include radiation (Langmann et al., 1997), gas phase chemistry (Trentmann et al., 2002) and gas scavenging by hydrometeors (Textor et al., 2003) were added lately. A further extension was made to include particle aggregation in 2006 (Textor et al., 2006b, a). However, the resolution of ATHAM is still pretty coarse compared with SK-3D and PDAC. Besides strengthening their strength (ATHAM has been adding more and more microphysics, PDAC was extended to consider more phases), these models are learning from each other. PDAC is trying to include effect of microphysics into the model while ATHAM wants to extend its ability to modeling pyroclastic flow. Both are using finer and finer resolution. Recently, a first order, non-equilibrium compressible 3D model, ASHEE (Cerminara et al., 2016a), is derived based on three dimensional N phases Eulerian transportation equation. ASHEE is valid for low concentration and low Strokes number region and much faster than N-phase Eulerian model. The model is based an open source numerical solver OpenFOAM adopting unstructured Finite Volume method.

To summarize, each 3D model has its own focus based on problem of interest. Different simplification was made based on their scope. Improvements have been making since their birth. Accuracy (depends on comprehensiveness of the model, resolution of discretization, numerical error, and order of accuracy) and simulation time (depends on number of governing equations,

resolution, numerical methods and parallel techniques) are always a couple of conflict considerations in 3D plume simulations.

1.3 Features of SPH

All of these existing 3D models are based on mesh based methods (Eulerian methods). And there is no 3D plume model based on mesh free method (Lagrange methods) yet. Actually, Lagrangian methods have several features that are suitable for volcano plume simulation. Among such Lagrangian methods, Smoothed Particle Hydrodynamics (Gingold and Monaghan, 1977; Lucy, 1977), even though relatively newer compared to well designed mesh based methods such as Finite Difference, has shown good agreement with experiments by many applications in fluid dynamic area. And it currently is, by far the most widely used scheme. Specifically, the reason we choose SPH as the numerical method for volcanic plume simulation are as following:

- For mesh based method, either interface tracking (Lagrangian) (Harlow et al., 1965; Wrobel and Brebbia, 1991; Cheng and Armfield, 1995) or interface capturing (Eulerian) (Hirt and Nichols, 1981; Youngs, 1982; Gerlach et al., 2006; Gopala and van Wachem, 2008) methods are used to reconstruct the flow interface of free boundary flow. High computational cost, a tendency to form numerical instabilities and the inability to track complex topological changes are the significant drawbacks of Lagrangian techniques (Hirt and Nichols, 1981; Unverdi and Tryggvason, 1992; Anderson et al., 1998). For interface capturing (Eulerian) method, the surrendering of surface detail before the phase transport calculation means that interface reconstruction is required between time steps to recover the interface information, which need additional numerical effort (Hirt and Nichols, 1981; Youngs, 1982). Being able to adaptively adjust the discretization and automatically construct the interface, SPH does not require additional numerical effort for interface construction and therefor more suitable for volcano plume simulation.

- Advection term in the Navier-Stokes equations does not appear explicitly in discretized formula of SPH (as illustrated in Eq. (12) to Eq. (15)). So advection term will be treated exactly with SPH (Monaghan, 2005).

- It is easy to include various physics effect (like self gravity, radiative cooling and chemical reaction) into the model. It does not require a major overhaul and re-tooling every time new physics is introduced (Monaghan and Kocharyan, 1995). This implies that accounting for more physics is easier for SPH model.

- With more than one material, each described by its own set of particles, interface problems between phases are often trivial for SPH but difficult for mesh based schemes. So multiphase flow can be easily handled by SPH. Adding of new phases to the model also does not require a major overhaul and re-tooling. As will be shown in later paragraph, adding of new phases will lead to adding of several lines into the source code for new phases and additional interaction terms between existing phases and newly added phases.

As discussed in the previous paragraph, existing 3D plume models focus on one or several specific aspects of plume and have been extended to be more comprehensive by accounting for more physics or more phases. Easy extensibility and capability

of handling multiples phase flow without additional numerical effort greatly facilitate future extension of SPH models. As volcanic plume are in nature multiphase and without pre-defined boundary in the atmosphere, SPH is a suitable numerical method for plume modeling. The more basic physics, such as entrainment of air and thermal expansion, are essential for all plume modeling while some other physics, such as water condense and aggregation ect., are important in specific scenario. As an initial effort on exploiting advantages of SPH in volcano plume modeling, we focus on capturing more basic features in plume development. So our current model is far away from comprehensive. Fortunately, the easy extension feature of SPH facilitate developing of a more comprehensive model in the future.

1.4 Our contribution

As SPH is a relative new method in computational fluid dynamics, implementation of SPH for compressible multiphase turbulent flow is scarce. Several remedies of classical SPH are adopted in our implementation. One of the most popular implementations of SPH is simulating of free surface flow, such as breaking-waves and flood. Less attention was paid to velocity inlet and pressure outlet boundary conditions which are required in plume modeling. We impose pressure boundary condition by adding extra layers of static ghost particles. Additional constrain on time step is used to avoid growing up of numerical fluctuations near the pressure boundary. We impose velocity inlet boundary condition by placing several layers of ghost particles moving with eruption velocity. Turbulence model is crucial for reproducing the entrainment of air. There are several turbulence models proposed for SPH method (Issa, 2005; Violeau and Issa, 2007). We adopt a LANS (Lagrangian Averaged Navier-Stokes) turbulence model (Monaghan, 2011), which was originally proposed for incompressible, and extend it for compressible flow accounting for turbulent heat exchange. Corrected formulism of SPH is adopted to bypass the well-known tensile instability issues of classical SPH.

Simulation of volcano plume with acceptable accuracy requires fine resolutions (very high particle counts) that cannot be accomplished without parallel computing within given time window. The core solver of our model is parallelized by MPI (message passing interface standard) parallelism. In addition, a dynamic load balancing strategy is also developed. What's more, imposing of some types of boundary conditions (such as eruption boundary condition) requires dynamically adding and removing of particles during simulation. To address this issue, we adopt an efficient data management scheme based on time dependent SFC (space filling curve) and hash table. The computational cost is further reduced by adjusting simulation domain adaptively.

The organization of this paper is as following: The physics model of the plume is first presented in section 2 and leads to a complete mathematical description of the volcano plume (governing equations and boundary conditions). In section 3 we will briefly introduce the numerical tool, SPH method. Both the fundamental discretization formulism and techniques that used to handle specific issues involved in plume modeling will be discussed. Verification and validation with numerical tests is presented in section 4. In section 5, a discussion on future work is given following a brief summary.

2 Physical Model

2.1 Description of the model

During an explosive eruption, a volcanic jet erupts out from a vent with a speed of several tens to more than 150 meters per second, driven by expanding gas. The jet is initially denser than the surrounding atmosphere and begins to decelerate through negative buoyancy and turbulent interaction with surrounding air. Cauliflower-like vortices are generated along jet margins, within which process, air is entrained and heated up, reducing the bulk density of the entire jet, in many cases, to less than that of the surrounding atmosphere. Once it becomes buoyant, such a jet develops into a plinian or subplinian plume, rising kilometers to tens of kilometers until its heat is exhausted that buoyancy is lost. Jets that lose their momentum before becoming buoyant collapse back onto the ground and transform into pyroclastic flows, surges and ignimbrites. When plume rises up, relative coarse particle might separate with main stream of the plume falling down onto the ground and probably be re-entrained into the plume at a lower height (Ernst et al., 1996). Within this process, erupted vapor will condense to liquid (droplet) and even further to ice. Latent heat released from phase change of erupted vapor will further heat up entrained air and make the bulk density further dilute. The entrained vapor might also experience a similar process and impose impact on plume development. Particle aggregation process (Carey and Sigurdsson, 1982; Taddeucci et al., 2011), either due to presence of liquid water, resulting from particle collision or driven by electrostatic forces might occur inside plume and thereby affect the sedimentation. All in all, the process of plume development is essentially a multiphase turbulent mixing process coupled with heat transfer and many other microphysics and chemical reaction.

As an initial effort on exploiting the feasibility and advantage of SPH in plume modeling, our model is designed to describe an injection of well mixed solid and volcanic gas from a circular vent above a flat surface into a stratified stationary atmosphere following SK-3D (Suzuki et al., 2005). In this model, molecular viscosity and physics heat conduction is neglected since turbulent exchange coefficients are dominant. Erupted material consist of solid with different size and mixture of gases (various constituent) is assumed to be well mixed and behave like a single phase fluid (phase 2) which is valid for eruptions with fine particles and ash. Air (also a mixture of different gases) is assumed to be another phase (phase 1). Immediate thermodynamics equilibrium is assumed so that no separate energy equation is needed for each phase. As a result, there is only one energy equation for both phases (heat exchange term between different phases will not show up under this assumption). Immediate dynamics equilibrium is assumed so that no separate momentum equation is needed for each phase. As a result, there is only one vector momentum equation for both phases (drag force term will not show up with this assumption).

Because of above assumptions, all other micro-physics process (like phase change of H_2O aggregation, decomposition, absorption of gas on the surface of solid, solution of gas into the liquid) and chemical process is not be considered in this model. These ignored microphysics factors will come to play critical roles under particular eruption conditions. Capturing of these processes needs a more comprehensive model. One critical element in plume development, the effect of wind, is also not considered in this model yet. Introducing of wind effect in our model will need imposing of pressure boundary conditions in a dynamic way, which requires more numerical effort and algorithm design. To summarize, our model is not valid for eruptions where wind effect plays a significant role in its development, usually refereed as weak plume. Our model is also lack ability in modeling

plumes with large particles or eruptions in which microphysics plays non-ignorable roles, such as a eruption of *El Chichón* volcano at April 4th (Sigurdsson et al., 1984; Folch et al., 2016). At present stage, we are more interested in more basic (and more critical) aspects of volcano plume and therefor devote our effort to this relative simpler model. It is worthwhile to mention here that because SPH is adopted as our numerical method, adding of these physics into our model would require much less work in terms of programming compared to mesh based methods.

2.2 Governing equations

Based above assumptions, the governing equation of our model is given as following, which is the same as governing equations of SK-3D (Suzuki et al., 2005).

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{1}$$

$$\frac{\partial \rho \xi}{\partial t} + \nabla \cdot (\rho \xi \mathbf{v}) = 0 \tag{2}$$

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v} + p\mathbf{I}) = \rho \mathbf{g} \tag{3}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot [(\rho E + p)\mathbf{v}] = \rho \mathbf{g} \cdot \mathbf{v} \tag{4}$$

ξ is the mass fraction of ejected material. $E = e + K$ is total energy which is a summation of kinetic energy K and internal energy e. An additional equation is required to close the system. In this model, the equation for closing the system is the EOS (equation of state) for ideal gas.

$$p = (\gamma_m - 1)\rho e \tag{5}$$

Where

$$\gamma_m = R_m / C_{vm} + 1 \tag{6}$$

$$R_m = n_g R_g + n_a R_a \tag{7}$$

$$C_{vm} = n_s C_{vs} + n_g C_{vg} + n_a C_{va} \tag{8}$$

$$n_a = 1 - \xi \tag{9}$$

$$n_g = \xi n_{g0} \tag{10}$$

$$n_s = \xi - n_g \tag{11}$$

Where, C_v is specific heat with constant volume, n is mass fraction, R is gas constant. The subscript m represents mixture of ejected material and air, s represents solid portion in ejected material, g represents gas portion in the ejected material and a represents air. In mesh based method, governing equations in Eulerian form, Eq. (1) to Eq. (4), will be directly discretized. For SPH, governing equations in Lagrange form is needed. By deducting Kinetic energy from energy equation, subtracting mass conservation from momentum equation, combining transient term and advection term into material derivative term, the governing equation is put into the final formulism, in which advection term does not show up explicitly.

$$\frac{D\rho}{Dt} + \rho\nabla\cdot\mathbf{v} = 0 \tag{12}$$

$$\frac{D\rho\xi}{Dt} + \rho\xi\nabla\cdot\mathbf{v} = 0 \tag{13}$$

$$\frac{D\mathbf{v}}{Dt} + \frac{\nabla P}{\rho} = \mathbf{g} \tag{14}$$

$$\frac{De}{Dt} + \frac{P\nabla\cdot\mathbf{v}}{\rho} = 0 \tag{15}$$

Governing equations in Lagrange form, Eq. (12) to Eq.(15), and Eulerian form, Eq. (1) to Eq. (4), are actually equivalent as deriving of Lagrange form from Eulerian form is purely based on mathematics without any additional physics assumption.

2.3 Boundary conditions

In current model the initial domain is a box. The boundaries are categorized into velocity inlet (a circle area at the center of the bottom of the box), wall boundary (box bottom), pressure outlet (other faces of the box).

2.3.1 Velocity inlet

At the vent, temperature of erupted material T, eruption velocity $\mathbf{v}$, the mass fraction of vapor in erupted material n_{g0} and mass discharge rate $\dot{M}$ is given. The pressure of erupted material p is assumed to be the same as ambient pressure for pressure-balanced eruption. The radius of vent is determined from ρ, $\dot{M}$ and $\mathbf{v}$. Equation (16) (19) gives velocity outlet boundary condition wrote in terms of primitive variables.

$$\rho = const = p/(R_m T) \tag{16}$$

$$\xi = const = 1 \tag{17}$$

$$\mathbf{v} = const = \{u, v, w\}^T \tag{18}$$

$$\frac{\partial e}{\partial n} = \dot{M}e/(\pi r^2) \tag{19}$$

Where r is the radius of the vent.

2.3.2 Non-slip wall boundary

Velocity is zero for non-slip wall boundary. If assume the boundary to be adiabatic, heat flux should be zero on the boundary. The flux of mass should also be zero. As a result, internal energy flux, which consists of heat flux and energy flux carried by mass flux, will vanish on the wall boundary. Equation (20) (23) gives no-slip wall boundary condition wrote in terms of primitive variables.

$$\frac{\partial \rho}{\partial n} = const = 0 \tag{20}$$

$$\frac{\partial \xi}{\partial n} = const = 0 \tag{21}$$

$$\mathbf{v} = const = \{0,0,0\}^T \tag{22}$$

$$\frac{\partial e}{\partial n} = 0 \tag{23}$$

2.3.3 Open outlet pressure boundary condition

The pressure of the surrounding atmosphere is given. Except for the pressure, boundary conditions for density, velocity, and energy on the outlet should depend on the solution. As we ignore the viscosity, the shear stress is gone and norm stress (whose magnitude equals to pressure) will balance with ambient pressure.

$$p = p_a(h) \tag{24}$$

3 SPH method

SPH is a mesh free Lagragian method. In SPH, the domain is discretized as a set of particles or discretization points and the position of each particle (discretization point) is updated at every time step. Approximation of all field variables (velocity, density and pressure, ect.) is obtained by interpolation based on discretization points. The physical laws (such as conservation laws of mass, momentum and energy) written in the form of PDEs (partial differential equations) or ODEs (ordinary differential equations) need to be transformed into the Lagrangian particle formalism of SPH. Using a kernel function that provides the weighted estimation of the field variables at any point, the integral equations are evaluated as sums over neighbor particles. Thus, field variables associated to the particle are updated based on its neighbors. The kernel function has a compact support determined by smoothing length of each particle. There are several review papers (Monaghan, 1992, 2005; Rosswog, 2009; Price, 2012; Monaghan, 2012), giving a pretty comprehensive view over SPH, we will only refer here to the representation of the constitutive equations in SPH and put more focus on specific numerical techniques for plume modeling.

3.1 Fundamental principle

There are several routines for discretizing governing equations (PDEs or ODEs) with SPH. There is slight difference between discretized formulism obtained by different routines. We present here one of them following JJ Monaghan (Monaghan, 1992,

2005). The start point of approximating a function with SPH is one translation property of Dirac function.

$$A(\mathbf{x}) = \int_{-\infty}^{\infty} A(\mathbf{x}\prime)\delta(\mathbf{x}\prime - \mathbf{x})d\mathbf{x}\prime \tag{25}$$

The Dirac function in Eq. (25) can be approximated by a weighting function $w(\mathbf{x} - \mathbf{x}\prime, h)$ (or $w(\mathbf{x}\prime - \mathbf{x}, h)$) which will tend to a Dirac function when $h \to 0$:

$$\lim_{h\to 0} w(\mathbf{x}\prime - \mathbf{x}, h) = \delta(\mathbf{x}\prime - \mathbf{x}) \tag{26}$$

The function $A(\mathbf{x})$ then can be approximated by:

$$A(\mathbf{x}) \approx < A(\mathbf{x}) > = \int_{\Omega} A(\mathbf{x}\prime)w(\mathbf{x} - \mathbf{x}\prime, h)d\mathbf{x}\prime + O(h^2) \tag{27}$$

Where h is the smoothing length, determining the interaction distance. According to Eq. (27), the order of accuracy of SPH method is second order. However, in practice, second order can not be achieved because there is no guarantee on the symmetry of particle distribution in real simulation (Price, 2012). Recall that $d\mathbf{x}\prime = \frac{dm\prime}{\rho\prime}$, the integration equation, Eq. (27), can be approximated by summation and lead to an approximation of the function at particle (discretization point) a:

$$< A(\mathbf{x}) > \approx \sum_b m_b \frac{A_b}{\rho_b} w(\mathbf{x} - \mathbf{x}_b, h) \tag{28}$$

Where the summation is over all the particles within the region of compact support of the weighting function. Gradient terms may be straightforwardly calculated by taking the derivative of Eq. (28), giving

$$\begin{aligned} < \nabla A(\mathbf{x}) > &= \frac{\partial}{\partial \mathbf{x}} \int_{\Omega} A(\mathbf{x}\prime)w(\mathbf{x} - \mathbf{x}\prime, h)d\mathbf{x}\prime + O(h^2) \\ &\approx \sum_b m_b \frac{A_b}{\rho_b} \nabla w(\mathbf{x} - \mathbf{x}_b, h) \end{aligned} \tag{29}$$

For vector quantities the expressions are similar, simply replacing A with $\mathbf{A}$ in Eq. (28) and and Eq. (29), giving

$$< \mathbf{A}(\mathbf{x}) > \approx \sum_b m_b \frac{\mathbf{A}_b}{\rho_b} w(\mathbf{x} - \mathbf{x}_b, h) \tag{30}$$

$$< \nabla \cdot \mathbf{A}(\mathbf{x}) > \approx \sum_b m_b \frac{\mathbf{A}_b}{\rho_b} \cdot \nabla w(\mathbf{x} - \mathbf{x}_b, h) \tag{31}$$

$$< \nabla \times \mathbf{A}(\mathbf{x}) > \approx \sum_b m_b \frac{\mathbf{A}_b}{\rho_b} \times \nabla w(\mathbf{x} - \mathbf{x}_b, h) \tag{32}$$

$$< \nabla^j \mathbf{A}^i(\mathbf{x}) > \approx \sum_b m_b \frac{\mathbf{A}_b^i}{\rho_b} \nabla^j w(\mathbf{x} - \mathbf{x}_b, h) \tag{33}$$

3.2 Weighting function

As described in previous section, the weighting function (or kernel function) is used to replace Dirac function and should satisfy Eq. (26). So it can be viewed as approximation form of Dirac function and satisfy normalization condition:

$$\int w(\mathbf{x}-\mathbf{x\prime},h)d\mathbf{x\prime}=1 \tag{34}$$

Besides normalization, weighting function of particle a has to be symmetric with respect to a to ensure that neighbor particles locate at the same distance away will contribute equally to SPH summation equation, see Eq. (35). Weighting function also needs to satisfy conditions such as positivity and compact support. In addition, kernel function must be monotonically decreasing with the distance between particles.

$$w(\mathbf{x}-\mathbf{x\prime},h)=w(\mathbf{x\prime}-\mathbf{x},h) \tag{35}$$

There is a wide variety of possible weighting functions, such as spline functions (with different orders) and Gaussian function. Generally, the accuracy increases with the order of the polynomials of the kernel function, but the computational time also increases as number of interactions increase. We are adopting a truncated Gaussian function as the weighting function in our simulation.

$$w(\mathbf{x}-\mathbf{x}_b)=\begin{cases}\frac{1}{(h\sqrt{\pi})^d}exp[-(\frac{\mathbf{x}-\mathbf{x\prime}}{h})^2] & |\mathbf{x}-\mathbf{x\prime}|\leq 3h\\ 0 & \text{Otherwise}\end{cases} \tag{36}$$

Where d is number of dimensions. The derivative of weighting function:

$$\nabla w(\mathbf{x}-\mathbf{x}_b)=\begin{cases}-2(\frac{\mathbf{x}-\mathbf{x\prime}}{h^2})\frac{1}{(h\sqrt{\pi})^d}exp[-(\frac{\mathbf{x}-\mathbf{x\prime}}{h})^2] & |\mathbf{x}-\mathbf{x\prime}|\leq 3h\\ 0 & \text{Otherwise}\end{cases} \tag{37}$$

3.3 Discretization of governing equations

The basic interpolation given in Eq. (28) to Eq. (33) provide a general way of obtaining SPH expressions of governing equations. The problem is that using these expressions “as is" in general leads to quite poor gradient estimates. Various tricks are used to conserve linear and angular momentum and thermal energy (Monaghan, 1992). Special treatments are also need for second order derivative terms. We only refer here one of these possible discretization forms of compressible Euler equations with SPH:

$$<\rho_a>=\sum m_b w_{ab}(h_a) \tag{38}$$

$$<\frac{d\mathbf{v}_a}{dt}>=\sum_b m_b(\frac{p_b}{\rho_b^2}+\frac{p_a}{\rho_a^2}+\Pi_{ab})\nabla_a w_{ab}(h)+\mathbf{g} \tag{39}$$

$$<\frac{de_a}{dt}>=0.5\sum_b m_b\mathbf{v}_{ab}(\frac{p_b}{\rho_b^2}+\frac{p_a}{\rho_a^2}+\Pi_{ab})\cdot\nabla_a w_{ab}(h) \tag{40}$$

Where, $\mathbf{v}_{ab} = \mathbf{v}_a - \mathbf{v}_b$. Π is artificial viscosity term, which will be discussed in section 3.4. However, these discretized formulations will not guarantee conservation of entropy. We can obtain conservation of entropy by giving up conservation of mass or thermal energy adopting an alternative discretized formulations (Monaghan, 1992). As a Lagrangian method, particle position will also been update at every time step.

$$< \frac{d\mathbf{x}_a}{dt} >= \mathbf{v}_a \tag{41}$$

Adding new physics and new phases into the model is trivial in terms of discretization. For example, adding of new source (or sink) into Eq. (38), adding a drag force into Eq. (39) anding a heat exchange term into Eq. (40) will lead to the new discretization form:

$$< \rho_a >= \sum m_b w_{ab}(h_a) + \dot{\rho}(\mathbf{x}, t) \tag{42}$$

$$< \frac{d\mathbf{v}_a}{dt} >= \sum_b m_b (\frac{p_b}{\rho_b^2} + \frac{p_a}{\rho_a^2} + \Pi_{ab}) \nabla_a w_{ab}(h) + \mathbf{g} + D \sum_b m_b \frac{\mathbf{v}_b - \mathbf{v}_a}{\rho_b} \tag{43}$$

$$< \frac{de_a}{dt} >= 0.5 \sum_b m_b \mathbf{v}_{ab} (\frac{p_b}{\rho_b^2} + \frac{p_a}{\rho_a^2} + \Pi_{ab}) \cdot \nabla_a w_{ab}(h) + \sum_b \frac{m_b}{\rho_b} (\kappa_a + \kappa_b) \frac{(T_a - T_b)}{\mathbf{r}_a - \mathbf{r}_b} w_{ab}(h) \tag{44}$$

Where the source term $\dot{\rho}$ can be a "sink" of erupted vapor due to phase change which is considered in ATHAM (Oberhuber et al., 1998). D is drag force coefficient. κ is heat conduction coefficient. Other physics can be added easily in a similar way. Adding of these new terms will lead to modification at several lines in the source code. The drag force term should show up only when dynamics dis-equilibrium between different phases is considered. In that case, each phase will need one set of governing equations of Navier-Strokes type. Adding of new phases into SPH code only needs to add several new lines for the new phase besides adding interaction terms.

3.4 Artificial viscosity

In classical SPH, shock waves are handled by introducing artificial viscosity to smear out discontinuities. Π, in discretized momentum equation, Eq. (39), and energy equation, Eq. (40), represents artificial viscosity term which is essentially a second order derivative term. As in the case of first derivatives, second derivatives can be estimated by differentiating a SPH interpolation twice. However, such formulism has two disadvantages: First, it is very sensitive to particle disorder. Second, the second derivative of the kernel can change sign and will lead to unphysical representation (viscous dissipation cause decrease of the entropy). One of the most commonly used form of artificial viscosity is:

$$\Pi_{ab} = -\frac{\nu}{\bar{\rho}_{ab}} \frac{\mathbf{v}_{ab} \cdot \mathbf{x}_{ab}}{\mathbf{x}_{ab}^2 + (\eta h)^2} \tag{45}$$

Absolute(kinematic) viscous coefficient ν is defined as:

$$\nu = \alpha \bar{h}_{ab} \bar{c}_{ab} \tag{46}$$

Where

$$\bar{c}_{ab} = \frac{c_a + c_b}{2} \tag{47}$$

$$\bar{\rho}_{ab} = \frac{\rho_a + \rho_b}{2} \tag{48}$$

$$\mathbf{v}_{ab} = \mathbf{v}_a - \mathbf{v}_b \tag{49}$$

$$\mathbf{x}_{ab} = \mathbf{x}_a - \mathbf{x}_b \tag{50}$$

The artificial viscosity term Π_{ab} is a Galilean invariant and vanishes for rigid rotation. It will produce a repulsive force between two particles when they are approaching each other and an attractive force when they are receding from each other.

The SPH viscosity can be related to a continuum viscosity by converting the summation to integrals (Monaghan, 2005). It has been shown that the shear viscosity coefficient $\eta = \frac{\rho\alpha hc}{8}$ and the bulk viscosity coefficient $\zeta = \frac{5\eta}{3}$ for two dimensional and $\eta = \frac{\rho\alpha hc}{10}$, $\zeta = \frac{5\eta}{3}$ for three dimensional. An extra term was added to ν considering aspects of the dissipative term in shock solutions based on Riemann solvers and lead to:

$$\Pi_{ab} = \begin{cases} \dfrac{-\alpha\mu_{ab}\bar{c}_{ab} + \beta\mu_{ab}^2}{\bar{\rho}_{ab}} & \mathbf{v}_{ab}\cdot\mathbf{x}_{ab} < 0 \\ 0 & \mathbf{v}_{ab}\cdot\mathbf{x}_{ab} > 0 \end{cases} \tag{51}$$

Where

$$\mu_{ab} = \frac{h\mathbf{v}_{ab}\cdot\mathbf{x}_{ab}}{\mathbf{x}_{ab}^2 + (\eta h)^2} \tag{52}$$

α and β are free parameters that must be adjusted for each case. $\alpha = 1$ and $\beta = 2$ are recommended by Monaghan for best results. In our simulation, these two parameters are calibrated to $\alpha = 0.3$ and $\beta = 0.6$. η is usually taken as 0.1 for preventing singularities when $\mathbf{x}_{ab} = 0$.

3.5 Time step

The physical properties (velocity, density, position and pressure) change every time step due to particle interactions. We have to consider the Courant condition, which is in spirit similar to the Courant condition for the grid-based methods, to determine the amount of the time step Δt.

$$\Delta t = CFL \min_a \left\{ \frac{\left[\frac{m_a}{\rho_a}\right]^{\frac{1}{d}}}{c_a} \right\} \tag{53}$$

Where c_a is sound speed at particle a, d is number of dimensions. First order Euler time integration, with $CFL = 0.2$ has been used in three dimensional calculations.

3.6 Tensile instability and corrected derivatives

The classical SPH method suffered from tensile instability and boundary deficiency. To address these difficulties, Chen (Chen et al., 1999) proposed a corrective SPH method. For 1D case, by employing a Taylor expansion for $A(x)$ about x_a, multiplying

both side by kernel function and then do intergration over the domain, gives

$$\int_{\Omega} A(x)w(x-x_a,h)dx = A_a \int_{\Omega} w(x-x_a,h)dx + A_{xa} \int_{\Omega} (x-x_a)w(x-x_a,h)dx + ... \tag{54}$$

Ignore derivative terms higher than first order, and write the intergral in particle approximation form:

$$A_a = \frac{\sum_b m_b \frac{A_b}{\rho_b} w(x_a - x_b, h)}{\sum_b m_b \frac{1}{\rho_b} w(x_a - x_b, h)} \tag{55}$$

Equation 55 implies that $\sum_b m_b \frac{1}{\rho_b} w(x-x_b,h) = 1$, which can be viewd as the approximation form of Eq. 34. The first order derivative term can be obtained in a similar way and lead to:

$$\nabla A_a = \frac{\sum_b m_b \frac{A_b - A_a}{\rho_b} \nabla_a w(x_a - x_b, h)}{\sum_b m_b \frac{x_b - x_a}{\rho_b} \nabla_a w(x_a - x_b, h)} \tag{56}$$

For higher dimensions problem, expression for function approximation are exactly the same as Eq. (55) even though the derivation is different. The first order derivative can be solved explicitly or numerically from:

$$\mathbf{M}_{\alpha\beta a} \mathbf{A}_{\alpha a} = \mathbf{F}_{\beta a} \tag{57}$$

where

$$\mathbf{M}_{\alpha\beta a} = \sum_b m_b \frac{\mathbf{x}_b^\alpha - \mathbf{x}_a^\alpha}{\rho_b} \nabla_a^\beta w(\mathbf{x}_a - \mathbf{x}_b, h) \tag{58}$$

$$\mathbf{A}_{\alpha a} = \frac{\partial A}{\partial x^\alpha}(\mathbf{x}_a) \tag{59}$$

$$\mathbf{F}_{\beta a} = \sum_b m_b \frac{A_b - A_a}{\rho_b} \nabla_a^\beta w(\mathbf{x}_a - \mathbf{x}_b, h) \tag{60}$$

α and β are direcion indices, ranging from 1 to 3 for 3D problem.

3.7 Multiphase SPH

Numerical simulation of multiphase flows is usually difficult due to the existence of interface between two fluids. And the interface is usually moving or deformable. This present severe challenges to conventional Eulerian grid-based numerical methods. As we discussed in the introduction section, special algorithms to treat and track the interface between different phases is required for mesh based method to deal with multiphase problems. What's worse, the engulfment process during plume development makes geometry of interface very complicated. SPH is able to handle multiphase problems without additional numerical effort for dealing with interface, which makes it suitable for volcano plume simulation. Take the governing equation adopted in our model as an example. Different phases of fluid (air and erupted material) will be represented by two different

sets of SPH particles (or discretization points). Based on assumption we made in section 2, only density need to be updated respectively for each phase. The updating of density is exactly the same as Eq. (38) in spirit. Particles of phase 1 will not be counted while evaluate density of phase 2 and vice verse. Updating of density is based on the following discretized equations.

$$< \rho_\alpha^a >= \frac{\sum m_b w_{\alpha b}(h_\alpha)}{\sum \frac{m_b}{\rho_b} w_{\alpha b}(h_\alpha) + \sum \frac{m_j}{\rho_j} w_{\alpha j}(h_\alpha)} \tag{61}$$

$$< \rho_\alpha^{sg} >= \frac{\sum_j m_j w_{\alpha j}(h_\alpha)}{\sum \frac{m_b}{\rho_b} w_{\alpha b}(h_\alpha) + \sum \frac{m_j}{\rho_j} w_{\alpha j}(h_\alpha)} \tag{62}$$

Where the subscript a and b represents air particles (phase 1) while i and j represents erupted material. β and α represents either erupted material or air. ρ_a^a is density of phase 1 (air). ρ_i^{sg} is density of phase 2 (erupted material). $\rho = \rho^a + \rho^{sg}$ is density of mixture of air and erupted material. By definition, the mass fraction is updated according to Eq. (63).

$$< \xi_\alpha >= \frac{\rho_\alpha^{sg}}{\rho_\alpha} \tag{63}$$

In the areas far away from interface, updating of density is exactly the same as that for single phase flow. For example, in the right side and left side (or blue areas) in Fig. 1, where there are only air particles, Eq. (62) will lead to zero and total density will be:

$$\rho_\alpha = \rho_\alpha^a = \frac{\sum m_b w_{\alpha b}(h_\alpha)}{\sum \frac{m_b}{\rho_b} w_{\alpha b}} \tag{64}$$

Which is a special case of Eq. (55). For these areas occupied by only particles of phase 2 and far away from interface, similarly, the equation for density update will become:

$$\rho_\alpha = \rho_\alpha^{sg} = \frac{\sum m_j w_{\alpha j}(h_\alpha)}{\sum \frac{m_j}{\rho_j} w_{\alpha j}} \tag{65}$$

That is to say, the same density updating equation can be applied for both phases and no additional numerical treatment needed to locate where the interface is.

3.8 Turbulence modeling with SPH

Typically, turbulence fluctuations in a volcanic column occur on very different length scales, ranging from millimeters up to hundreds of meters. The entrainment of air and heat transfer are dominated by such turbulent fluctuations as turbulent exchange coefficient would be several magnitudes larger than molecular viscosity (and heat conduction) coefficient. One way to include enough turbulence in the model is to use SPH-SPS (SPH sub-particle-scale) method. Another way is to use extremely fine resolution (The direct numerical simulation method) at the expense of much higher computational cost that results from both increase in number of particles and decrease in time steps constrained by CFL condition. Here we choose the SPH-SPS method. Even though free surface flow are in nature turbulent, detailed research in turbulence using SPH method is rather scarce. Among the first papers on turbulence are the implementations of the α turbulence model (Holm, 1999) by Monaghan (Monaghan, 2002)

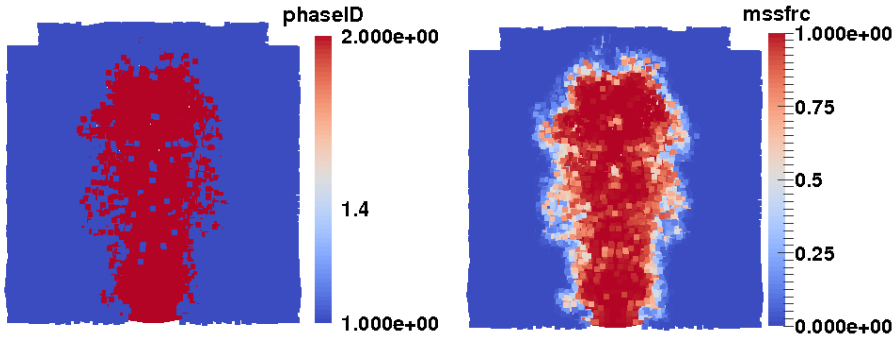

Figure 1. In the left figure, the blue particles (phaseID=1) represent air particles the red ones (phaseID=2) represent erupted material. The right figure shows corresponding mass fraction. Mass fraction are evaluated based on Eq. (61) and Eq. (61) without any interface track or capture method.

and the RANS (Reynolds averaged Navier-Stokes) approach by Violeau (Violeau and Issa, 2007). A first wall-bounded LES (large eddy simulation) simulation with SPH was performed by Issa (Issa, 2005). An improved version of $SPH-\alpha$ method is proposed in 2011 by Monaghan (Monaghan, 2011), which we refer to as $SPH-\varepsilon$ method. We will adopt $SPH-\varepsilon$ method in our model. As most of these papers constrain their focus on incompressible or weakly compressible flow. To simulate turbulence in plume, we need to extend existing incompressible turbulent modeling methods to compressible flow.

$SPH-\alpha$ model (Monaghan, 2002) is a powerful extension of the XSPH algorithm which reduces disorder at short length scales and retains the constants of the motion. But in that paper, energy conservation equation is discretized in a way that is exactly the same as usual SPH. There is no special treatment on energy conservation equation. And the recent improved version (Monaghan, 2011) of $SPH-\alpha$ method focus on incompressible flow only. Actually, there are several additional terms in the averaged compressible Navier-Stokes equations besides the Reynolds stress term (Rumsey, 2014). The averaged momentum equation for compressible flow are in the same form as that for incompressible flow, all of other additional terms except for turbulent stress term are in the energy equation. However, most turbulence modeling focuses on the Reynolds stress terms which are either solved directly or defined via a constitutive relation. Less attention is typically given to the other terms. Most commonly, a Reynolds analogy is used to model the turbulent heat flux. Simulations of heat transfer or other scalar transfer in turbulent flow simply involve adding transport terms for thermal energy or species concentration, at the expense of greater storage and longer computing times but without other difficulties (Cebeci, 2013). We adopt this strategy in our simulation of turbulent heat transfer with $SPH-\varepsilon$. PDAC (Neri et al., 2003) also adopted a Reynolds analogy in its turbulence modeling. The additional terms associated with molecular diffusion and turbulent transport in the energy equation are either modeled in different ways or neglected sometimes (Rumsey, 2014). We neglect these terms in our simulation.

3.8.1 $SPH-\varepsilon$ turbulence model

In $SPH-\varepsilon$ method (Monaghan, 2011), a turbulence model is constructed within the framework of SPH in such a way that general principles such as conservation of energy, momentum and circulation are satisfied using the ideas associated with the LANS (Lagrangian averaged Navier-Stokes equations). The basic idea of $SPH-\varepsilon$ is that determine a smoothed velocity $\widehat{\mathbf{v}}$ by a linear operation on the unsmoothed velocity $\mathbf{v}$. The SPH particles move with this smoothed velocity and the average motion of the fluid is determined by Eq. (66):

$$\frac{d\mathbf{x}_a}{dt} = \widehat{\mathbf{v}}_a \tag{66}$$

Lagrangian equations contain extra term which represents the stresses induced by the smoothing. Once the form of the smoothing is chosen these stresses are determined. The typical LANS model uses a smoothed velocity $\widehat{\mathbf{v}}$ defined in terms of the unsmoothed velocity $\mathbf{v}$ by:

$$\widehat{\mathbf{v}}(\mathbf{x}) = \int \mathbf{v}(\mathbf{x}\prime) G(|\mathbf{x}\prime - \mathbf{x}|, l) d\mathbf{x}\prime \tag{67}$$

Where G satisfies:

$$\int G(|\mathbf{x}\prime - \mathbf{x}|, l) d\mathbf{x}\prime = 1 \tag{68}$$

and is a member of a sequence of functions which tends to the δ function in the limit where $l \to 0$. A typical example is Gaussian. The length scale l determines the characteristic width of the kernel and the distance over which the velocity is smoothed.

It is common practice in $LANS-\alpha$ to use a differential equation for the smoothing rather than the integral form and finally reach to system of equations that need to be solved implicitly. In $SPH-\varepsilon$ method, a XSPH smoothing is adopted which will conserve linear and angular momentum. In this way solving of system of equations is avoided and it also makes the method simple to implement and cheap for computation. The discretized form of momentum equation is obtained through lengthy derivation based on Lagrangian equations. Derivation details and other discussions, such as conservation of momentum and energy, satisfactory of circulation theory, energy/momentum spectrum analysis based on Founrier transfer are detailed in Monaghan's paper (Monaghan, 2011). Here we just have a brief summary on key steps.

The smooth that Monaghan adopted is:

$$\widehat{\mathbf{v}}(\mathbf{x}) = \mathbf{v}(\mathbf{x}) + \epsilon \int (\mathbf{v}(\mathbf{x}\prime) - \mathbf{v}(\mathbf{x})) G(|\mathbf{x}\prime - \mathbf{x}|, l) d\mathbf{x}\prime \tag{69}$$

As function G has the same feature as kernel function w, SPH approximation of the integration will lead to:

$$\widehat{\mathbf{v}}(\mathbf{x}) = \mathbf{v}(\mathbf{x}) + \epsilon \sum_b m_b \frac{(\mathbf{v}_b - \mathbf{v})}{\rho_b} G(|\mathbf{x}_b - \mathbf{x}|, l) \tag{70}$$

By making the replacement:

$$\frac{G(|\mathbf{x}_b - \mathbf{x}_a|, l_a)}{\rho_b} \to \frac{K_{ab}}{M} \tag{71}$$

Where $K_{ab} = l^d G_{ab}$, $M = \rho_0 l^d$ in which d is the dimension and ρ_0 is initial density. $SPH - \varepsilon$ turbulence model is obtained after lengthy derivation:

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b [m_b(\frac{p_b}{\rho_b^2} + \frac{p_a}{\rho_a^2}) \bigtriangledown_a w_{ab}(h_a)] + \sum_b m_b \frac{\varepsilon}{2} \frac{\mathbf{v}_{ab} \cdot \mathbf{v}_{ab}}{M} \bigtriangledown_a K_{ab} \tag{72}$$

Instead of using Eq. (72) which is from Monaghan's paper, we make inverse replacement of Eq. (71) and notice that if l is uniform and keep constant:

$$\nabla K_{ab} = \nabla(l^d G_{ab}) = l^d \nabla G_{ab} \tag{73}$$

So the discretized momentum equation with $SPH - \varepsilon$ turbulence model in our simulation will be:

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b [m_b(\frac{p_b}{\rho_b^2} + \frac{p_a}{\rho_a^2}) \bigtriangledown_a w_{ab}(h_a)] + \sum_b m_b \Phi_{ab} \bigtriangledown_a G_{ab}(l_a) \tag{74}$$

where

$$\Phi_{ab} = \frac{\varepsilon}{2} \frac{\mathbf{v}_{ab} \cdot \mathbf{v}_{ab}}{\rho_b} \tag{75}$$

and take $\varepsilon = 0.8$ as recommended by JJ Monaghan.

3.8.2 Turbulence heat transfer

We adopt Reynolds analogy to get heat transfer coefficient due to turbulence. Prandtl number is defined as:

$$Pr = \frac{C_p \mu}{\kappa} \tag{76}$$

Where, μ is dynamic viscosity, κ is thermal conductivity. And μ can be written in term of absolute viscosity (kinematic viscosity) as:

$$\mu = \rho \nu \tag{77}$$

Then

$$\kappa = \frac{C_p \mu}{Pr} \tag{78}$$

Typical value of Pr_t for air is 0.7 $\sim$ 0.9 . We take $Pr_t = 0.85$ for gases as recommended by Kays, William M (Kays, 1994) from summarizing of experimental results. JJ Monaghan (Monaghan, 2005) summarized the simulation of viscosity and heat conduction in his review on SPH. We will refer to his summary in our following discussion. The additional term in discretized momentum equation, Eq. (74), is the turbulent shear stress term. Recall that viscosity term can be discretized with SPH as shown in Eq. (45). Comparing of this term to physical viscous term for compressible gas shows that it has both bulk viscosity and shear viscosity, where shear viscosity coefficient is (Monaghan, 2005):

$$\nu_t = S\nu \tag{79}$$

with

$$S = \begin{cases} \frac{1}{6} & for \quad d=3 \\ \\ \frac{5}{24} & for \quad d=2 \end{cases} \tag{80}$$

The turbulent viscosity coefficient can be inferred from that form if we can reformulate the "turbulence stress" term in a form which is similar to molecular shear viscous term. Reformulate the "turbulence stress" term:

$$\sum_b \frac{\varepsilon}{2} \frac{m_b}{\rho_b} \mathbf{v}_{ab} \cdot \mathbf{v}_{ab} \nabla_a G_{ab}(l_a) = \sum_b \frac{\varepsilon}{2S} m_b \frac{\mathbf{v}_{ab}}{\rho_b} \frac{S \mathbf{v}_{ab} \cdot \mathbf{x}_{ab}}{x_{ab}^2} \frac{x_{ab}^2}{\mathbf{x}_{ab}} \nabla_a G_{ab}(l_a) \tag{81}$$

Then the turbulent viscosity coefficient will be

$$\nu_t = \frac{\varepsilon}{2S} \frac{\mathbf{v}_{ab} \cdot \mathbf{x}_{ab}}{\rho_b} \tag{82}$$

Please notice that turbulent viscosity term has opposite sigh with molecular viscosity term in discretized momentum equation and there is a minus sign in the expression of Π_{ab}, they will cancelled out. However, the above equation is correct only for 1D situation. For 2D or 3D, it is not easy to get the explicit expression of viscosity.

We adopt an alternative way: obtaining a value for each pair of particles instead of persisting on getting an analytical expression. Choosing smoothing function as the same as SPH kernel and the smoothing scale l as the same as smoothing length h, the ratio between turbulent shear stress and physical shear stress is:

$$\begin{aligned} \Upsilon_{ab} &= \frac{\frac{\varepsilon}{2} \frac{\mathbf{v}_{ab} \cdot \mathbf{v}_{ab}}{\rho_b}}{\frac{S\nu}{\rho_{ab}} \frac{\mathbf{v}_{ab} \cdot \mathbf{x}_{ab}}{x_{ab}^2 + \eta^2 h_{ab}^2}} \\ &= \frac{\varepsilon (x_{ab}^2 + \eta^2 h_{ab}^2)}{2S\nu} \frac{\mathbf{v}_{ab} \cdot \mathbf{v}_{ab}}{\mathbf{v}_{ab} \cdot \mathbf{x}_{ab}} \end{aligned} \tag{83}$$

Υ_{ab} is essentially equivalent to ratio between turbulent viscous effect attributed by particle b on particle a and molecular viscous effect attributed by particle b on particle a. Turbulent viscosity can be easily obtained by:

$$\begin{aligned} \nu_{t,ab} &= \nu \Upsilon_{ab} \\ &= \frac{\varepsilon (x_{ab}^2 + \eta^2 h_{ab}^2)}{2S} \frac{\mathbf{v}_{ab} \cdot \mathbf{v}_{ab}}{\mathbf{v}_{ab} \cdot \mathbf{x}_{ab}} \end{aligned} \tag{84}$$

The corresponding turbulent thermal conductivity should be

$$\kappa_{t,ab} = \frac{\varepsilon \overline{C_{p,ab}} \overline{\rho_{ab}} (x_{ab}^2 + \eta^2 h_{ab}^2) \mathbf{v}_{ab} \cdot \mathbf{v}_{ab}}{2S Pr_t \mathbf{v}_{ab} \cdot \mathbf{x}_{ab}} \tag{85}$$

The term that used to prevent singularity now can be removed.

$$\kappa_{t,ab} = \frac{\varepsilon \overline{C_{p,ab}} \overline{\rho_{ab}} x_{ab}^2 \mathbf{v}_{ab} \cdot \mathbf{v}_{ab}}{2S Pr_t \mathbf{v}_{ab} \cdot \mathbf{x}_{ab}} \tag{86}$$

We also need to prevent singularity, so:

$$\kappa_{t,ab} = \begin{cases} 0 & if \quad \mathbf{v}_{ab} = 0 \quad or \quad \mathbf{x}_{ab} = 0 \\ \dfrac{\varepsilon \overline{C_{p,ab}} \overline{\rho_{ab}} x_{ab}^2 \mathbf{v}_{ab} \cdot \mathbf{v}_{ab}}{2 Pr_t \mathbf{v}_{ab} \cdot \mathbf{x}_{ab}} & \text{otherwise} \end{cases} \tag{87}$$

The heat conduction equation without source term is:

$$C_p \frac{dT}{dt} = \frac{1}{\rho} \nabla (\kappa \nabla T) \tag{88}$$

Second spatial derivative can be approximated with SPH by following JJ Monaghan (Monaghan, 2005).

$$C_p \frac{dT}{dt} = \sum_b \frac{m_b}{\rho_a \rho_b} (\kappa_a + \kappa_b)(T_a - T_b) F_{ab}(h_a) \tag{89}$$

Where

$$F_{ab} \mathbf{x}_{ab} = \nabla_a w_{ab} \tag{90}$$

And $F_{ab} \leq 0$, which guarantees that heat flux will flow from hot to cold. Plug the turbulent thermal conductivity into the heat conduction equation:

$$\begin{aligned} C_p \frac{dT}{dt} &= \sum_b \frac{m_b}{\rho_a \rho_b} (\kappa_a + \kappa_b)(T_a - T_b) F_{ab}(h_a) \\ &= 2 \sum_b \frac{m_b}{\rho_a \rho_b} \frac{\overline{C_{p,ab}} \overline{\rho_{ab}} \varepsilon x_{ab}^2 \mathbf{v}_{ab} \cdot \mathbf{v}_{ab}}{2 Pr_t S \mathbf{v}_{ab} \cdot \mathbf{x}_{ab}} (T_a - T_b) F_{ab}(h_a) \end{aligned} \tag{91}$$

Notice that the number 2 at the front of the equation comes from integration approximation of second order derivative (Cleary and Monaghan, 1999). By further simplification, we get:

$$C_p \frac{dT}{dt} = \frac{\varepsilon}{S Pr_t} \sum_b \frac{m_b}{\rho_a \rho_b} \frac{\overline{C_{p,ab}} \overline{\rho_{ab}} x_{ab}^2 \mathbf{v}_{ab} \cdot \mathbf{v}_{ab}}{\mathbf{v}_{ab} \cdot \mathbf{x}_{ab}} (T_a - T_b) F_{ab}(h_a) \tag{92}$$

3.8.3 Discretized governing equations with $SPH - \varepsilon$ turbulence model

Plug discretized turbulent stress term and turbulent heat transfer term into the momentum and energy equation, we got new discretized governing equation:

$$< \frac{d\mathbf{v}_\alpha}{dt} > = -\sum_b [m_b (\frac{p_b}{\rho_b^2} + \frac{p_\alpha}{\rho_\alpha^2} + \Pi_{\alpha b} - \Phi_{\alpha b}) \bigtriangledown_\alpha w_{\alpha b}(h_\alpha)] - \sum_j [m_j (\frac{p_j}{\rho_j^2} + \frac{p_\alpha}{\rho_\alpha^2} + \Pi_{\alpha j} - \Phi_{\alpha j}) \bigtriangledown_\alpha w_{\alpha j}(h_\alpha)] + \mathbf{g} \tag{93}$$

With

$$\Phi_{\alpha\beta} = \frac{\varepsilon}{2} \frac{\mathbf{v}_{\alpha\beta} \cdot \mathbf{v}_{\alpha\beta}}{\rho_\beta} \tag{94}$$

$$< \frac{de_\alpha}{dt} > = 0.5 \sum_b [m_b \widehat{\mathbf{v}_{\alpha b}} (\frac{p_b}{\rho_b^2} + \frac{p_\alpha}{\rho_\alpha^2} + \Pi_{\alpha b} - \Phi_{\alpha b}) \bigtriangledown_\alpha w_{\alpha b}(h_\alpha)] + 2 \sum_b \frac{m_b}{\rho_\alpha \rho_b} \kappa_{t,\alpha b} (T_\alpha - T_b) F_{\alpha b}(h_\alpha)$$
$$+ 0.5 \sum_j [m_j \widehat{\mathbf{v}_{\alpha b}} (\frac{p_j}{\rho_j^2} + \frac{p_\alpha}{\rho_\alpha^2} + \Pi_{\alpha j} - \Phi_{\alpha j}) \bigtriangledown_\alpha w_{\alpha j}(h_\alpha)] + 2 \sum_j \frac{m_j}{\rho_\alpha \rho_j} \kappa_{t,\alpha j} (T_\alpha - T_j) F_{\alpha j}(h_\alpha) \quad (95)$$

with $\kappa_{t,\alpha\beta}$ given by Eq. (87). As the particle-scale movement of flow is based on smoothed velocity, the velocity in the energy equation should also be smoothed. The filtering process will be done according to Eq. (70). Position of particles will be updated according to Eq. (66). Smoothed velocity is also used while computing artificial viscosity Π_{ab}.

3.9 Boundary conditions in SPH

3.9.1 Wall boundary condtion

Traditionally either ghost particles that mirror SPH particles across the boundary (Ferrari et al., 2009), or boundary forces (Monaghan and Kajtar, 2009) have been used to impose the wall boundary conditions. One disadvantage of the later one is that the boundary forces tend to corrupt the solution in the neighborhood of such boundary forces. In addition, the natural way of imposing eruption boundary condition is using eruption ghost particles. To impose boundary conditions in a consistent way we adopt a modified version of ghost particle method (Kumar et al., 2013) for wall boundary conditions. Stationary ghosts particles are deployed the same way as initial deployment of real particles. Instead of enforcing the symmetry particle by particle, symmetric field across the boundary is explicitly enforced. Ghost positions are reflected into the domain and physical properties are calculated at these reflected positions by SPH interpolations. It should be noted that ghost particles are not used for these interpolations. Assign all properties at the corresponding reflect position to the ghost particle except for velocity. We assign a velocity to each wall ghost particles that is the negative of the interpolated velocity at its corresponding reflection inside the fluid domain. By this way, the no-slip wall boundary condition (Eq. (22)) is imposed naturally. These wall ghost particles will serve as neighbors in momentum and energy update. More implementation details about this method can be found in (Kumar et al., 2013). As these wall ghost particles keep stationary, there will be no mass flux on the boundary (Eq. (20) and (21)). In addition, as temperature is also symmetric with respect to the boundary, the gradient of temperature will vanish on the wall boundary and hence no internal energy flux on the wall boundary (Eq. (23)).

3.9.2 Eruption boundary condition

The natural way of imposing eruption boundary condition is using ghost particles which moves with eruption velocity and bears the temperature of erupt material. A parabolic velocity profile which represents a fully developed Hagen-Poiseuille flow is used to determine the particle velocity. The detailed shape of the parabolic profile will determined based on averaged eruption velocity (Eq. 18). The correctness of eruption boundary condition for density (Eq. (16)) can be guaranteed by choosing mass of erupted ghost particles given distribution of these particles. The internal energy associated with these particles are set

to a value so that Eq. (19) is satisfied. The mass fraction of erupted material (Eq. (17)) is automatically satisfied as all particles in the eruption conduit are of phase 2. The density, momentum, and internal energy of these eruption ghost particles will not be updated before they move above ground. As soon as they move out from erupt conduit, these ghost particle will be shift to real particles and their physics properties and position will be updated based on discretized governing equations. New ghost particle will be added at the bottom of the eruption conduit as these existing ghost particles move upwards.

3.9.3 Pressure boundary condition

Another boundary in this model is pressure outlet boundary. For flow in a straight channel, it is possible to treat the exit the same as the entry with a prescribed velocity profile. For flow with more complex channel, an exit far downstream of the flow disturbance is also feasible. However, the natural boundary condition (Eq. (24)) is more suitable for plume simulation where the outlet is open atmosphere. The way we impose pressure boundary condition is: adding several layers of pressure ghost particles surrounding the real atmosphere particles. Physics properties except for velocity (pressure, density, temperature) are determined based on the elevation of particles. Velocity is set to zero for static atmosphere. The physics properties for pressure ghost particles will not be updated while these for real particles will be updated at every time step. As position of all pressure ghost particles are keep constant, we are essentially imposing a static pressure boundary condition. Real particles that moves out the pressure boundary will be removed.

As simulation progress, change in position and physical properties of real particles near pressure boundary boundaries might corrupt pressure boundary condition that was established initially. This shortcoming is relieved by choosing a larger computational domain so that boundaries that might be corrupted will locate far away from turbulent mixing area. In addition, to avoid enlarging fluctuation, we add another constrain on time step:

$$\Delta t \leq CFL_p \frac{h}{v} \tag{96}$$

Where CFL_p is a safety coefficient which has similar function as the normal CFL number. Too small CFL_p will slow down simulation while too large CFL_p will lose its ability of mitigate numerical fluctuation near the boundary. The proper CFL_p is determined by series of simulation tests.

In subfigure b of Fig. 2, different type of boundary conditons are marked with different color. The red portion are main computational domain.

3.10 Parallellism and performance

One disadvantage of 3D model is that it usually takes much longer time than 1D models to complete one simulation. This disadvantage further preventing doing simulation with finer resolution and accounting for more physics in one model. Non-intrusive uncertainty analysis which is commonly adopted in hazard forecasting requires finishing multiple simulations within a given time window. High performance computing provides a powerful routine for relieving the misery. Among existing CPU parallel SPH schemes, most of them focus on neighbors searching algorithm and dynamic load balancing (eg. (Ferrari et al.,

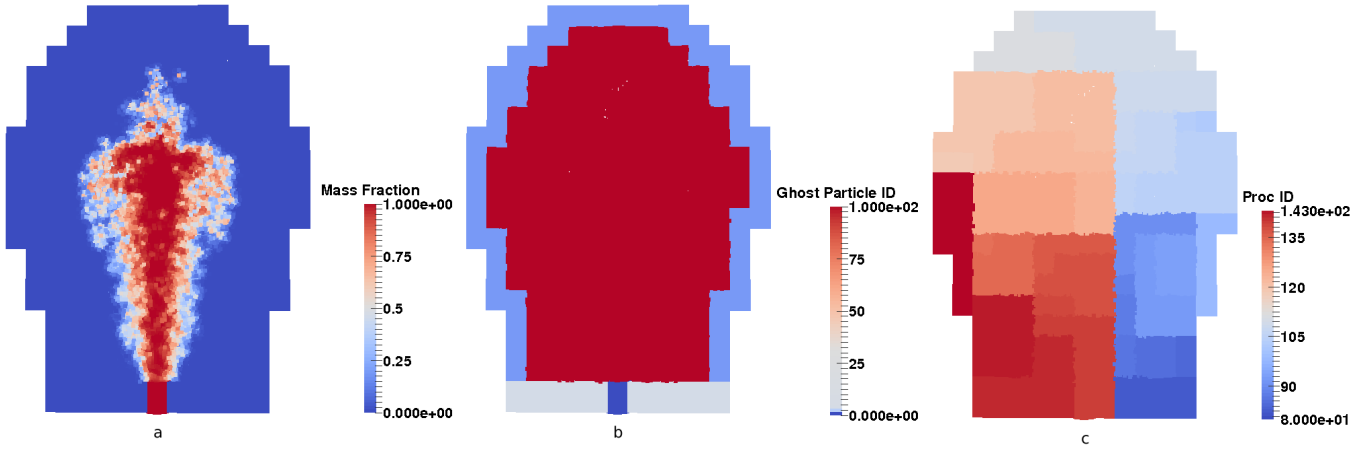

Figure 2. A cross section view of the simulation domain. Subfigure a shows the mass fraction of a plume simulation. Subfigure b shows all boundary conditions: the dark blue is for velocity inlet boundary, the light blue is for pressure outlet boundary, the gray is for no-slip wall boundary condition. Ghost particle ID of All real particles are set to 100, so is 100 in b. Subfigure c shows the cross section view of domain decomposition based on SFC. The simulation is conducted on 144 processors (12 nodes with 12 cores each node), so there are 144 subdomains in total.

2009; Crespo et al., 2015)). Less attention has been paid to developing of more flexible data management schemes for more complicated problems. Motivated by techniques developed for mesh based methods, we develop a complete framework for parallelizing SPH program with MPI standard model allowing flexible and efficient data access.

Any implementation of SPH code requires efficient searching and updating of neighbors during simulation. Of the many choices possible for this we adopted a background grid which was proposed by Monaghan and Lattanzio (Monaghan and Lattanzio, 1985) and is quite popular in parallel SPH. The background grid is also used for domain decomposition in SPH. We refer to the elements of background grid, namely squares for two dimensions and cubic for three dimensions, as buckets. As for the actual storage of data representing the physical quantities associated to each particle, different strategies have been adopted in existing implementations of SPH. In both SPHysics and DualSPHysics (Crespo et al., 2015), the physical quantities of each particle (position, velocity, density...) are stored in arrays, and the particles (and the arrays with particle data) are reordered following the order of the cells. This has two advantages: 1) access pattern is more regular and more efficient, 2) it is ease to identify the particles that belong to a cell by using a range since the first particle of each cell is known. But adding, deleting and especially accessing of particles are cumbersome. Ferrari (Ferrari et al., 2009) adopted linked lists using pointers so that particles can be deleted or added during the simulation. Storage problems caused by fix-size arrays are thereby also eliminated. We define C++ classes which contain all data of particles and buckets. As for the management of data, we adopt hash tables to store pointers to particles and buckets, which give us not only flexibility of deleting and adding elements, but also quicker access compared with linked list. Instead of using the "nature manner" to number particles, we adopt SFC based index to give each particle and background bucket an unique identifier – a strategy known to preserve some locality at minimal cost. The

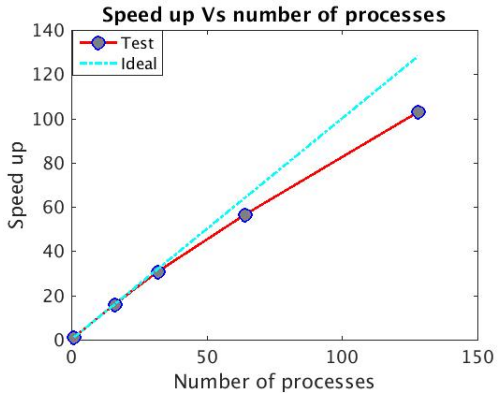

Figure 3. Strong scalability test results

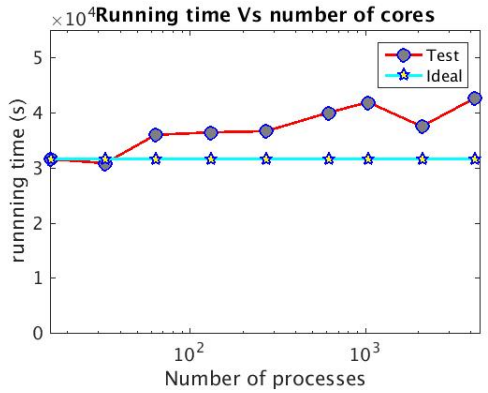

Figure 4. Weak scalability test results

SFC based numbering strategy is further extended to include time step information so that particles added at the same position but different time will have different identifiers. As for domain decomposition, even though more complicated graph-based partitioning tools (Biswas and Oliker, 1999) might get higher quality decomposition, they requires much more effort in programming and computation. So we adopt an easy-programming scheme based on SFC (Patra and Kim, 1999). Subfigure c of Fig. 2 shows a cross section view of domian decomposition.

We will provide a supplementary documentation describing details about the data structure, domain decomposition, load balancing and domain adjusting.

Experiments have been carried out on CCR (the computational cluster of Center for Computational Research) at Buffalo. E5645 CPUs running at 2.40GHz clock rate with 4GB memory per core on a Q-Logic Infiniband is used in these tests. Each node comprise of two sockets with 6 of these cores. Memory and level 3 cache are shared on each node. The initial domain is $[-4.8km, 4.8km] \times [-4.8km, 4.8km] \times [0km, 6km]$. Almost linear speed up is observed in our strong scalability test (Fig. 3). The weak scalability test is conducted with the same initial domain and various smoothing lengths. Each

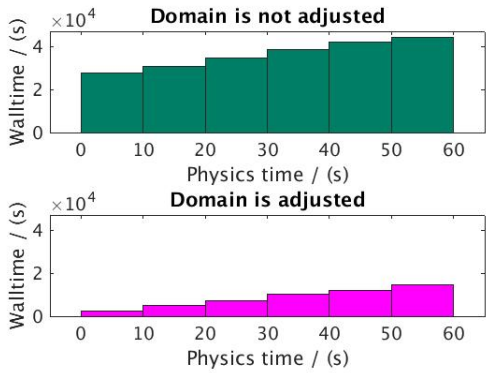

Figure 5. The effect of domain adjusting on simulation time. The figure on the top shows simulation time without domain adjusting, the figure on the bottom shows simulation time with domain adjusting. Different bins represent simulation time up to specific physics time indicated by x axis.

simulation is run for 400 steps. The average number of real particles of each process is kept constant at 25900. As shown in Fig. 4, simulation times increase around $1/3$ when number of cores increase from 16 to 4208. For the test problem in this section, the volcanic plume will finally reach to a region of $[-30km\ 30km] \times [-30km\ 30km] \times [1.5km\ 40km]$ after around 400 seconds of eruption. When numerical simulation goes up to 90 seconds, the plume is still within a region of $[-10km\ 10km] \times [-10km\ 10km] \times [0km\ 25km]$. This implies that adjusting of domain can avoid computing large number of uninfluenced air particles, especially for the beginning stage of simulation. A domain adjusting algorithm is designed and implemented in our code. Figure 5 shows that simulation time of the test problem is greatly reduced when we adopt the domain adjusting strategy in our code.

4 Verification and validation

We will now present a series of numerical simulations to verify and validate the model. Plume-SPH is first verified by a JPUE simulation. Velocity and mass fraction distribution both along the central axis and cross transverse are compared with experimental results. The pattern of ambient particles entrainment is also clearly shown. Then a simulation of representative strong volcano plume is conducted. Both global variables and local variable are comparable with simulation results from existing 3D plume models. Analysis on evolution of plume shows detailed air entrainment mechanism.

4.1 Simulation of JPUE

JPUE can be considered as a simplified volcanic plume. While the effect of stratified atmosphere and the effect of expansion due to high temperature in volcanic plume are not represented, JPUE reproduces the entrainment due to turbulent mixing which is one of the key elements in volcanic plume development. There exist consistently good experimental data (List, 1982; Dimotakis et al., 1983; Papanicolaou and List, 1988) that describe the JPUE flow field giving an insight into details of JPUE,

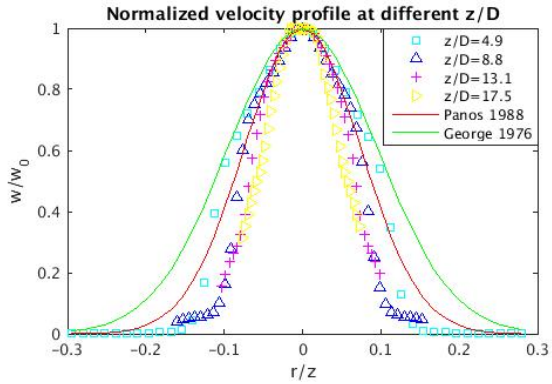

Figure 6. Dimensionless velocity distribution across the cross-section.

such as transverse velocity and concentration profile. In this section, we will verify that our code and the $SPH-\varepsilon$ turbulence model is able to reproduce feature of turbulent entrainment by a JPUE simulation.

As many of these experiments were conducted with liquid, we replace the original equation of state (Eq. (5)) with a weakly compressible Tait equation of state (Becker and Teschner, 2007) (see Eq. (97)) to avoid solving the Poisson equation.

$$p = B[(\frac{\rho}{\rho_0})^{\gamma} - 1] \tag{97}$$

with $\gamma = 7$ and B is evaluated by:

$$B = \frac{\rho_0 c^2}{\gamma} \tag{98}$$

Where c is speed of sound in the liquid. Energy equation is actually decoupled from momentum and mass conservation equation by using this EOS. In addition, the "atmosphere" is assumed to be uniform and gravity is set to be zero. We set the temperature and density of ejected material from the vent the same as its surrounding ambient. This will further simplify the scenario for the convenience of studying turbulent mixing. One overall feature of JPUE is "self-similarity" which means that the evolution of the JPUE is determined solely by the local scale of length and velocity which theoretically account for the fact that the rate of entrainment at the edge of JPUE is proportional to a characteristic velocity at each height. As a results, physical and numerical experiments do not necessary to have exactly the same setups and are compared on a non-dimensional basis.

A three dimensional axisymmetric JPUE which ejects from a round vent is simulated and results are compared with experiments (George et al., 1977; Papanicolaou and List, 1988) for verification purpose. Experimental data of concentration and velocity distribution across the cross-section was fit into a Gaussian profile: $\frac{\varphi}{\varphi_c} = exp[-coef(\frac{r}{z})^2)]$ (solid line) even though there is no priori reason. Panos (Papanicolaou and List, 1988) also fit concentration distribution and jet width based on velocity along central line into a straight line : $\frac{\varphi_0}{\varphi_c} = slope(\frac{z}{D} + intercept)$ (dash line). Where φ is either velocity or concentration, the subscript c represents the centreline, subscript 0 represents the cross-sectionally averaged exit value. r is the distance from the centreline on any cross-section. z is the axial distance from the origin of the jet transverse section under consideration. D is the diameter of vent. The coefficient $coef$ for concentration is 80 and 50 respectively according to Panos and George. $coef$

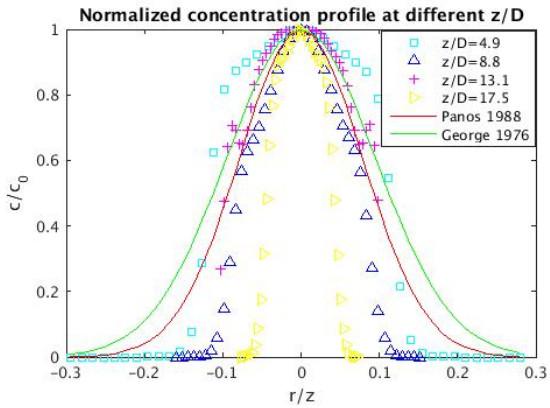

Figure 7. Dimensionless concentration distribution across the cross-section.

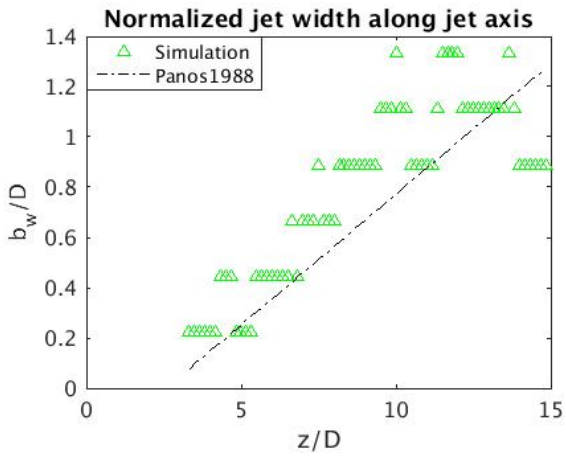

Figure 8. Dimensionless velocity distribution along centraline.

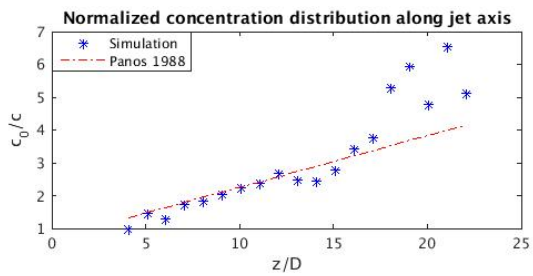

Figure 9. Dimensionless concetration distribution along centraline.

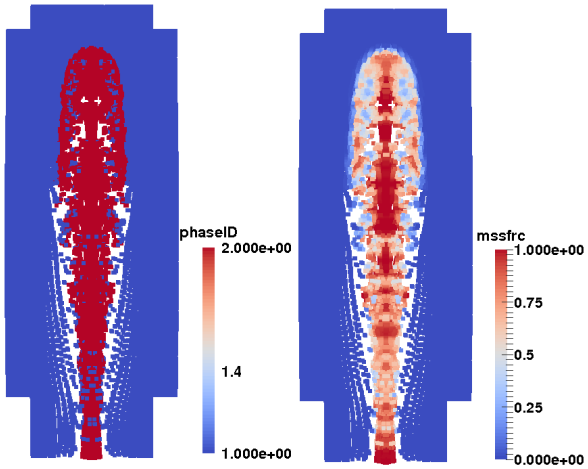

Figure 10. Left figure shows particle distribution. The right figure shows the mass fraction of erupted material at the moment corresponding to left figure. Particles of phase 1 (blue) are gradually entrained and mixed with erupted particles (red) as jet flows down stream. What need to be noticed is that the mass of particle of phase 1 (blue) is 8 time of that of phase 2 (red). It is a trade off between resolution and computational expense to give particle of two phases different particles mass.

for velocity is 90 and 55 respectively according to Panos and George. $slope$ for jet width based on velocity is 0.104 and for concentration is 0.157. $intercept$ for jet width based on velocity is 2.58 while that for the concentration is 4.35.

Although both velocity and concentration are found to be well matched with experimental results, a small disparity in both velocity and concentration are observed near the boundary of the jet. There are several factors that would attribute to such disparity. Reynolds number is not reported in many experiments assuming a high enough Reynolds number. In addition, some details of the experiments, such as exit velocity profile, viscosity of the experimental liquid, are not reported, neither. These factors prevent us from numerically reproducing these experiments in an exact way as they were conducted. However, the features of JPUE is correctly reproduced with our code.

We also investigated the mixing due to turbulence in JPUE simulation by checking the mixture of two phases. It is shown in Fig. 10 that the ejected material and ambient fluids are mixed through eddies at the outer shear region. And the inner dense core dispersed gradually due to erosion of the outer shear region. Hence, our confidence in the numerical correctness of our code is greatly reinforced.

4.2 Simulation on Plume

The development of volcano plume is more complicated than JPUE in several aspects. Besides turbulent entrainment of ambient fluids, development of volcano plume also involves heating up of entrained air and expanding in a stratified atmosphere. A strong eruption column without wind is tested in this section for the purpose of further verification and validation. Input parameters, material properties and atmosphere are chosen to be the same as the strong plume no wind case in an inter-

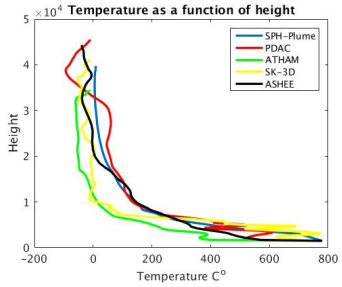

Figure 11. Temperature as a function of height

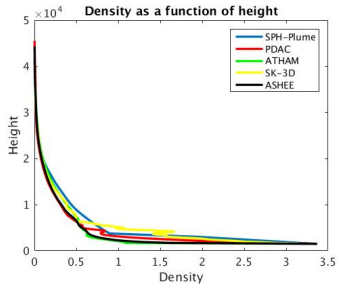

Figure 12. Density of the strong plume without wind after reaching its top height

comparison study on eruptive column models (Costa et al., 2016). The strong plume scenario was based on climactic phase of the Pinatubo eruption (Philippines, 15 June 1991). Material properties are selected based on properties of Pinatubo and Shinmoe-dake eruption. Both global variable and local variables are compared with existing models. Then plume evolution process is investigated.

4.2.1 Global and local variables

One of the key global quantities of great interest is the altitude to which the plume rises. The top height predicted by our model is around 40 km which agrees well with other plume models. For example, the height predicted by PDAC is 42500 m, by SK-3D is 39920 m, by ATHAM is 33392 m, by AHSEE is 36700 m. As for local variables, the profiles of integrated temperature, density, mass fraction of entrained air, gas mass fraction, mass fraction of solid and the radius of the plume as a function of height are compared with existing 1D and 3D models in Fig. 11 14. To get rid of significant fluctuations in time and space we did time averaging and spatial integration of the dynamic 3D flow fields by following Cerminara (Cerminara et al., 2016b). As particles distribute in the space in a disordered manner. We first project simulation results (on disordered particles) onto a pre-defined grid before doing time average and spatial integration. The interpolation method is the basic SPH kernel based interpolation.

The profiles of local variables match well with simulation results of existing 3D models in general sense. The basic phenomena in volcano plume development is correctly captured by our model. As the height increase, the amount of entrained air will

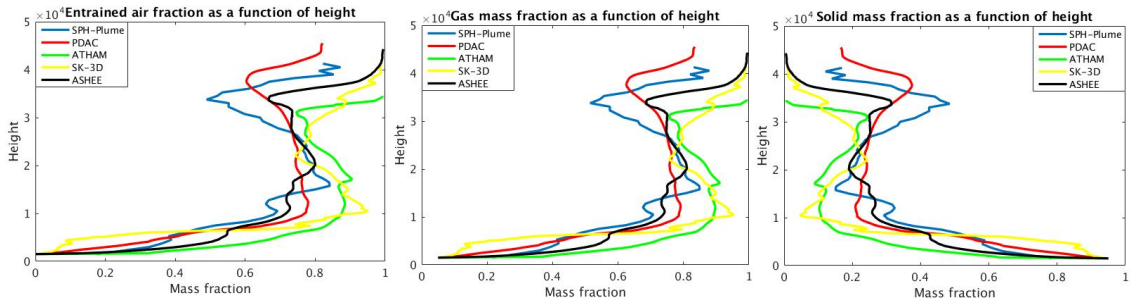

Figure 13. The mass fraction of entrained air, gas, and solid as a function of height.

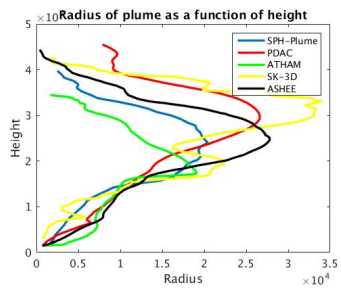

Figure 14. Radius of the strong plume without wind after reaching its top height

increase. Around the neutral height, where the umbrella expanding, the entrainment of air shows a slight decrease due to lack of air surrounding the column at that height. The profile for gas, which account for both air and vapor, shows a very similar tendency as that of entrained air. First of all, vapor condense is not considered in our model. In addition, as we assume that erupted material behaves like a single phase water, the mass fraction of gas will simply be a function of entrained air (Eq. (99)). Among these 3D models, ATHAM takes vapor condense into account and Eq. (99) does not hold for ATHAM. However, the profile of entrained air and profile of gas predicted by ATHAM are still very close to each other which implies that ignoring of water phase change is a valid assumption for eruptions similar to this test case (strong plume with erupted water fraction in erupted material less than 5%). Because air occupies a larger portion of the gas and plays a more significant role. As for mass fraction of solid, similarly, we have Eq. (100) or Eq. (101). PDAC which treat particles of two different sizes as two separate phases got similar mass fraction profile. Which implies that our model is at least valid for eruptions similar to the test case.

$$n_a + n_g = n_a + (1 - n_a) n_{g0} \tag{99}$$

$$n_s = (1 - n_a)(1 - n_{g0}) \tag{100}$$

$$n_s = 1 - (n_a + n_g) \tag{101}$$

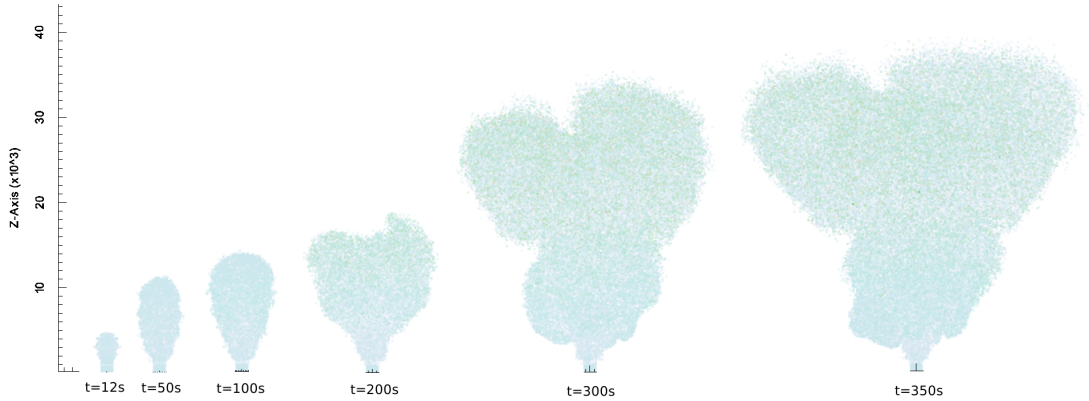

Figure 15. This figure shows the dynamic evolution process of the Pinatubo with time. At the very beginning, the plume is driven by initial momentum and rises up quickly, mixing and entraining surround air. Around 100s, the jet stage finishes and it becomes a plume as there were enough air entrained. The plume keeps rising up until reaches the top height then starts spreading.

With more cool air entrained into the plume and mixing with hot erupted material, the temperature of the plume decreases as the height increase as shown in Fig. 11. In the meanwhile, bulk density decrease due to entrainment and expansion (Fig. 12). Our model adopts the same assumptions and governing equations as SK-3D. However, there is still a big disparity between the profiles of local variables of our model and 3K-3D. One of the big differences between our model and 3K-3D is that we adopt a LANS type of turbulence model while 3K-3D adopts a LES (large eddy simulation) turbulence model. This implies that turbulence model plays a critical role in plume simulation.

4.2.2 Evolution of plume

Figure 15 shows the overall evolution process of plume. Plume keeps rising up and entraining air into the plume until it reaches its top height(around t=300 seconds). Afterwards the mixtures of erupted material and air then fall back to neutral height and starts spreading radially. Detailed velocity quiver plot in Fig. 16 shows entraining of air into the plume at the edge of column. The evolution of the plume further verifies our model.

5 Conclusions

A new plume model was developed based on SPH method. Advanced numerical techniques in SPH were exploited and tailored for this model. High performance computing was used to mitigate confliction between accuracy (depends on comprehensiveness of the model, resolution, order of accuracy of numerical methods, scheme for time upgrading.) and simulation time (depends on, comprehensiveness, resolution, order of accuracy of numerical methods, scheme for time upgrading ... and computational techniques). The correctness of the code and model was verified and validated by a series of test simulations.

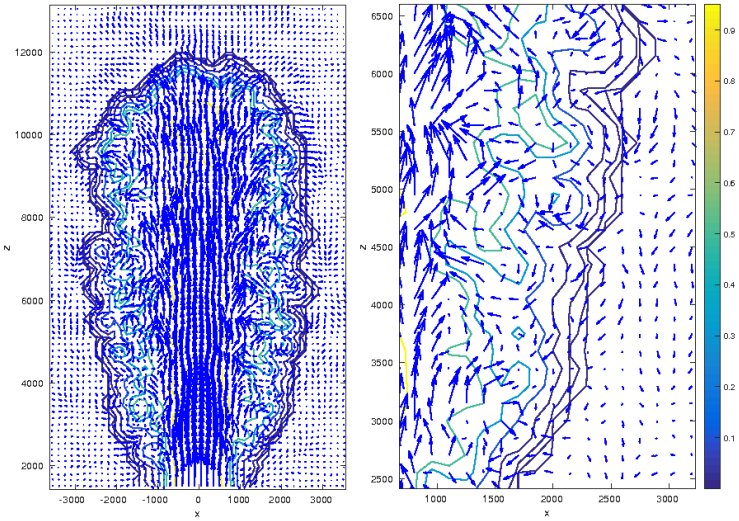

Figure 16. Contour of mass fraction and velocity quiver at 50 seconds in plane x=0, the column keeps rising up and at the same time entraining air into the plume. The zoomed view (right figure) of the quiver plot clearly shows mechanism of air entrainment. The contour levels of mass fraction are 0.7, 0.5, 0.3, 0.1, 1×10^{-1}, 1×10^{-2}, 1×10^{-3}, 1×10^{-4} and 1×10^{-5}.

Currently existing 3D models focus on certain aspect of the volcano plume (PDAC on pyroclastic flow, ATHAM on microphysics, and SK-3D on entrainment with higher accuracy and higher order of accuracy) and hence, naturally, different assumptions were made in these models. However, these different aspects of volcano plume are not independent and actually are coupled. In addition, there is no absolute boundary to determine which kind of hazard is dominant in certain eruption. So it is necessary to simulate all associated hazards in one model. Actually, effort has already been put on developing more comprehensive plume models. For example, a large-particle module (LPM) was added to ATHAM to track the paths of rocky particles (pyroclastic or tephra) within the plume and predicts where these particles fall (Kobs, 2009). We were also motivated by such a tendency of plume modeling to choose SPH as our numerical tool besides its ability in dealing with interface for multiphase flows. As mentioned in the introduction section, SPH method has good extension features and adding of new physics and phases requires tiny modification on the code compared with mesh based methods. The last but not the least, the dramatic development of computational power makes it possible to establish a comprehensive model. Even though right now, current computational capacity does not allow us to have too comprehensive model, the easy-extension feature of SPH makes it feasible to keep adding new physics into the model when necessary. What presented in this paper is actually our initial effort and results towards developing a first principle based plume model with comprehensive physics, adopting proper numerical tools and power of high performance computing. More advanced numerical techniques, such as adaptive particle size, Godnuv-SPH, semi-explicit time advancing scheme and better data management strategies and algorithms are on our list to exploit. In the near future, effect of wind field will be take into account.

6 Code availability

The Plume-SPH code with a user manual providing instructions for installation, running and visualization are archived in github (https://github.com/Plume-SPH/plume-sph.git). The input data for all simulations presented in this work can also be found in the repository.

7 Data availability

Data for running the test cases, including meteorological data, properties of erupted material and air, and eruption condition data can be found in source code repository. Raw data of simulation results are archived in google drive. Access permission will be given upon request. Post-processed data for plotting figures is provided as supplement along with the manuscript. Please note that some of these pictures are generated by visualizing the raw data with visualization tools, there is no separate data for these figures.

Competing interests. Authors have the standard conflict with employers and listed sponsors

Acknowledgements. All developers of Titan-2D, especially Dinesh Kumar who developed the GSPH version of Titan-2D, are greatly appreciated as Plume-SPH is based on their code. Advice and data (which is not shown in this paper) given by Suzuki Yujiro gave us great help at the initial stage of model establishment and therefor are greatly appreciated. We appreciate Tomaso Esposti Ongaro for providing simulation data of PDAC (were also provided by Antonio Costa later together with simulation results of other plume models) which helped us in doing early verification and improvement. We appreciate Antonio Costa for providing simulation results of existing 3D models, which are used in verification and validation section of this paper. We thank Matteo Cerminara for his helping on post-processing of plume simulation results. Computational results reported here were performed at the Center for Computational Research at the University at Buffalo. This project is supported by Grants No. NSF 1131074 from the National Science Foundation.

References

Anderson, D., McFadden, G. B., and Wheeler, A.: Diffuse-interface methods in fluid mechanics, Annual review of fluid mechanics, 30, 139–165, 1998.

Becker, M. and Teschner, M.: Weakly compressible SPH for free surface flows, in: Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation, pp. 209–217, Eurographics Association, 2007.

Biswas, R. and Oliker, L.: Experiments with repartitioning and load balancing adaptive meshes, in: Grid Generation and Adaptive Algorithms, pp. 89–111, Springer, 1999.

Bursik, M.: Effect of wind on the rise height of volcanic plumes, Geophys. Res. Lett, 28, 3621–3624, 2001.

Bursik, M. I.: bent: A model of plumes in crossflow, https://vhub.org/resources/267, 2010.

Bursik, M. I., Carbonara, A. U., and Zawicki, S. M.: puffin, https://vhub.org/resources/114, 2013.

Carcano, S., Bonaventura, L., Esposti Ongaro, T., and Neri, A.: A semi-implicit, second-order-accurate numerical model for multiphase underexpanded volcanic jets, Geoscientific Model Development, 6, 1905–1924, 2013.

Carey, S. N. and Sigurdsson, H.: Influence of particle aggregation on deposition of distal tephra from the May 18, 1980, eruption of Mount St. Helens volcano, Journal of Geophysical Research: Solid Earth, 87, 7061–7072, 1982.

Cebeci, T.: Analysis of turbulent flows with computer programs, Butterworth-Heinemann, 2013.

Cerminara, M., Ongaro, T. E., and Berselli, L.: ASHEE-1.0: a compressible, equilibrium-Eulerian model for volcanic ash plumes, Geoscientific Model Development, 9, 697–730, 2016a.

Cerminara, M., Ongaro, T. E., and Neri, A.: Large Eddy Simulation of gas–particle kinematic decoupling and turbulent entrainment in volcanic plumes, Journal of Volcanology and Geothermal Research, 2016b.

Chen, J., Beraun, J., and Jih, C.: An improvement for tensile instability in smoothed particle hydrodynamics, Computational Mechanics, 23, 279–287, 1999.

Cheng, L. and Armfield, S.: A simplified marker and cell method for unsteady flows on non-staggered grids, International journal for numerical methods in fluids, 21, 15–34, 1995.

Cleary, P. W. and Monaghan, J. J.: Conduction modelling using smoothed particle hydrodynamics, Journal of Computational Physics, 148, 227–264, 1999.

Costa, A., Suzuki, Y., Cerminara, M., Devenish, B., Ongaro, T. E., Herzog, M., Van Eaton, A., Denby, L., Bursik, M., Vitturi, M. d., et al.: Results of the eruptive column model inter-comparison study, Journal of Volcanology and Geothermal Research, 2016.

Crespo, A., Domínguez, J., Rogers, B., Gómez-Gesteira, M., Longshaw, S., Canelas, R., Vacondio, R., Barreiro, A., and García-Feal, O.: DualSPHysics: Open-source parallel CFD solver based on Smoothed Particle Hydrodynamics (SPH), Computer Physics Communications, 187, 204–216, 2015.

Dimotakis, P. E., Miake-Lye, R. C., and Papantoniou, D. A.: Structure and dynamics of round turbulent jets, Physics of Fluids, 26, 3185–3192, 1983.

Draxler, R. and Rolph, G.: HYSPLIT (HYbrid Single-Particle Lagrangian Integrated Trajectory) Model access via NOAA ARL READY Website (). NOAA Air Resources Laboratory, Silver Spring, 2015.

Ernst, G. G., Sparks, R. S. J., Carey, S. N., and Bursik, M. I.: Sedimentation from turbulent jets and plumes, Journal of Geophysical Research: Solid Earth, 101, 5575–5589, 1996.

Ferrari, A., Dumbser, M., Toro, E. F., and Armanini, A.: A new 3D parallel SPH scheme for free surface flows, Computers & Fluids, 38, 1203–1217, 2009.

Folch, A., Costa, A., and Macedonio, G.: FPLUME-1.0: An integral volcanic plume model accounting for ash aggregation, Geoscientific Model Development, 9, 431, 2016.

George, W. K., Alpert, R. L., and Tamanini, F.: Turbulence measurements in an axisymmetric buoyant plume, International Journal of Heat and Mass Transfer, 20, 1145–1154, 1977.

Gerlach, D., Tomar, G., Biswas, G., and Durst, F.: Comparison of volume-of-fluid methods for surface tension-dominant two-phase flows, International Journal of Heat and Mass Transfer, 49, 740–754, 2006.

Gingold, R. A. and Monaghan, J. J.: Smoothed particle hydrodynamics: theory and application to non-spherical stars, Monthly notices of the royal astronomical society, 181, 375–389, 1977.

Gopala, V. R. and van Wachem, B. G.: Volume of fluid methods for immiscible-fluid and free-surface flows, Chemical Engineering Journal, 141, 204–221, 2008.

Harlow, F. H., Welch, J. E., et al.: Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, Physics of fluids, 8, 2182, 1965.

Herzog, M., Graf, H.-F., Textor, C., and Oberhuber, J. M.: The effect of phase changes of water on the development of volcanic plumes, Journal of Volcanology and Geothermal Research, 87, 55–74, 1998.

Herzog, M., Oberhuber, J. M., and Graf, H.-F.: A prognostic turbulence scheme for the nonhydrostatic plume model ATHAM, Journal of the atmospheric sciences, 60, 2783–2796, 2003.

Hirt, C. W. and Nichols, B. D.: Volume of fluid (VOF) method for the dynamics of free boundaries, Journal of computational physics, 39, 201–225, 1981.

Holm, D. D.: Fluctuation effects on 3D Lagrangian mean and Eulerian mean fluid motion, Physica D: Nonlinear Phenomena, 133, 215–269, 1999.

Issa, R.: Numerical assessment of the Smoothed Particle Hydrodynamics gridless method for incompressible flows and its extension to turbulent flows, 2005.

Kays, W. M.: Turbulent Prandtl number—where are we?, Journal of Heat Transfer, 116, 284–295, 1994.

Kobs, S.: Modeling particle motion and near-vent deposition in explosive volcanic eruptions, State University of New York at Buffalo, 2009.

Koyaguchi, T. and Woods, A. W.: On the formation of eruption columns following explosive mixing of magma and surface-water, Journal of Geophysical Research: Solid Earth (1978–2012), 101, 5561–5574, 1996.

Koyaguchi, T., Ochiai, K., and Suzuki, Y. J.: The effect of intensity of turbulence in umbrella cloud on tephra dispersion during explosive volcanic eruptions: Experimental and numerical approaches, Journal of Volcanology and Geothermal Research, 186, 68–78, 2009.

Kumar, D., Patra, A. K., Pitman, E. B., and Chi, H.: Parallel Godunov smoothed particle hydrodynamics (SPH) with improved treatment of Boundary Conditions and an application to granular flows, Computer Physics Communications, 184, 2277–2286, 2013.

Langmann, B., Herzog, M., and Graf, H.-F.: Radiative forcing of sulfate aerosols as determined by a regional circulation-chemistry transport model, Journal of Aerosol Science, 28, S389–S390, 1997.

List, E.: Turbulent jets and plumes, Annual review of fluid mechanics, 14, 189–212, 1982.

Lockwood, J. P. and Hazlett, R. W.: Volcanoes: global perspectives, John Wiley & Sons, 2013.

Lucy, L. B.: A numerical approach to the testing of the fission hypothesis, The astronomical journal, 82, 1013–1024, 1977.

Mastin, L. G.: A user-friendly one-dimensional model for wet volcanic plumes, Geochemistry, Geophysics, Geosystems, 8, 2007.

Mastin, L. G.: Plumeria 2.3.1, https://vhub.org/resources/1194, 2011.

Monaghan, J.: SPH compressible turbulence, Monthly Notices of the Royal Astronomical Society, 335, 843–852, 2002.

Monaghan, J.: Smoothed particle hydrodynamics, Reports on progress in physics, 68, 1703, 2005.

Monaghan, J.: Smoothed particle hydrodynamics and its diverse applications, Annual Review of Fluid Mechanics, 44, 323–346, 2012.

Monaghan, J. and Kocharyan, A.: SPH simulation of multi-phase flow, Computer Physics Communications, 87, 225–235, 1995.

Monaghan, J. J.: Smoothed particle hydrodynamics, Annual review of astronomy and astrophysics, 30, 543–574, 1992.

Monaghan, J. J.: A turbulence model for Smoothed Particle Hydrodynamics, European Journal of Mechanics-B/Fluids, 30, 360–370, 2011.

Monaghan, J. J. and Kajtar, J. B.: SPH particle boundary forces for arbitrary boundaries, Computer Physics Communications, 180, 1811–1820, 2009.

Monaghan, J. J. and Lattanzio, J. C.: A refined particle method for astrophysical problems, Astronomy and astrophysics, 149, 135–143, 1985.

Morton, B., Taylor, G., and Turner, J.: Turbulent gravitational convection from maintained and instantaneous sources, in: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, vol. 234, pp. 1–23, The Royal Society, 1956.

Neri, A., Esposti Ongaro, T., Macedonio, G., and Gidaspow, D.: Multiparticle simulation of collapsing volcanic columns and pyroclastic flow, Journal of Geophysical Research: Solid Earth (1978–2012), 108, 2003.

Oberhuber, J. M., Herzog, M., Graf, H.-F., and Schwanke, K.: Volcanic plume simulation on large scales, Journal of Volcanology and Geothermal Research, 87, 29–53, 1998.

Ongaro, T. E., Cavazzoni, C., Erbacci, G., Neri, A., and Salvetti, M.-V.: A parallel multiphase flow code for the 3D simulation of explosive volcanic eruptions, Parallel Computing, 33, 541–560, 2007.

Papanicolaou, P. N. and List, E. J.: Investigations of round vertical turbulent buoyant jets, Journal of Fluid Mechanics, 195, 341–391, 1988.

Patra, A. and Kim, D.: Efficient mesh partitioning for adaptive hp finite element meshes, in: In International Conference on Domain Decomposition Methods, Citeseer, 1999.

Price, D. J.: Smoothed particle hydrodynamics and magnetohydrodynamics, Journal of Computational Physics, 231, 759–794, 2012.

Rosswog, S.: Astrophysical smooth particle hydrodynamics, New Astronomy Reviews, 53, 78–104, 2009.

Rumsey, C.: Implementing Turbulence Models into the Compressible RANS Equations, http://turbmodels.larc.nasa.gov/implementrans.html, 2014.

Settle, M.: Volcanic eruption clouds and the thermal power output of explosive eruptions, Journal of Volcanology and Geothermal Research, 3, 309–324, 1978.

Sigurdsson, H., Carey, S., and Espindola, J.: The 1982 eruptions of El Chichón volcano, Mexico: stratigraphy of pyroclastic deposits, Journal of Volcanology and Geothermal Research, 23, 11–37, 1984.

Sparks, R. S. J., Bursik, M., Carey, S., Gilbert, J., Glaze, L., Sigurdsson, H., and Woods, A.: Volcanic plumes, Wiley, 1997.

Suzuki, Y. and Koyaguchi, T.: A three-dimensional numerical simulation of spreading umbrella clouds, Journal of Geophysical Research: Solid Earth (1978–2012), 114, 2009.

Suzuki, Y. and Koyaguchi, T.: Numerical determination of the efficiency of entrainment in volcanic eruption columns, Geophysical Research Letters, 37, 2010.

Suzuki, Y. J. and Koyaguchi, T.: 3D numerical simulation of volcanic eruption clouds during the 2011 Shinmoe-dake eruptions, Earth, Planets and Space, 65, 581–589, 2013.

Suzuki, Y. J., Koyaguchi, T., Ogawa, M., and Hachisu, I.: A numerical study of turbulent mixing in eruption clouds using a three-dimensional fluid dynamics model, Journal of Geophysical Research: Solid Earth, 110, 2005.

Taddeucci, J., Scarlato, P., Montanaro, C., Cimarelli, C., Del Bello, E., Freda, C., Andronico, D., Gudmundsson, M., and Dingwell, D.: Aggregation-dominated ash settling from the Eyjafjallajökull volcanic cloud illuminated by field and laboratory high-speed imaging, Geology, 39, 891–894, 2011.

Textor, C., Graf, H.-F., Herzog, M., and Oberhuber, J.: Injection of gases into the stratosphere by explosive volcanic eruptions, Journal of Geophysical Research: Atmospheres (1984–2012), 108, 2003.

Textor, C., Graf, H.-F., Herzog, M., Oberhuber, J. M., Rose, W. I., and Ernst, G.: Volcanic particle aggregation in explosive eruption columns. Part II: Numerical experiments, Journal of Volcanology and Geothermal Research, 150, 378–394, 2006a.

Textor, C., Graf, H.-F., Herzog, M., Oberhuber, J. M., Rose, W. I., and Ernst, G. G.: Volcanic particle aggregation in explosive eruption columns. Part I: Parameterization of the microphysics of hydrometeors and ash, Journal of Volcanology and Geothermal Research, 150, 359–377, 2006b.

Trentmann, J., Andreae, M., Graf, H.-F., Hobbs, P., Ottmar, R., and Trautmann, T.: Simulation of a biomass-burning plume: Comparison of model results with observations, Journal of Geophysical Research: Atmospheres (1984–2012), 107, AAC–5, 2002.

Unverdi, S. O. and Tryggvason, G.: A front-tracking method for viscous, incompressible, multi-fluid flows, Journal of computational physics, 100, 25–37, 1992.

Violeau, D. and Issa, R.: Numerical modelling of complex turbulent free-surface flows with the SPH method: an overview, International Journal for Numerical Methods in Fluids, 53, 277–304, 2007.

Vitturi, M. d. M.: PlumeMoM, https://vhub.org/resources/3541, 2015.

Wilson, L., Sparks, R., Huang, T., and Watkins, N.: The control of volcanic column heights by eruption energetics and dynamics, Journal of Geophysical Research: Solid Earth (1978–2012), 83, 1829–1836, 1978.

Woodhouse, M., Hogg, A., Phillips, J., and Sparks, R.: Interaction between volcanic plumes and wind during the 2010 Eyjafjallajökull eruption, Iceland, Journal of Geophysical Research: Solid Earth, 118, 92–109, 2013.

Woods, A.: The fluid dynamics and thermodynamics of eruption columns, Bulletin of Volcanology, 50, 169–193, 1988.

Woods, A. W.: Moist convection and the injection of volcanic ash into the atmosphere, Journal of Geophysical Research: Solid Earth (1978–2012), 98, 17 627–17 636, 1993.

Woods, A. W. and Bower, S. M.: The decompression of volcanic jets in a crater during explosive volcanic eruptions, Earth and Planetary Science Letters, 131, 189–205, 1995.

Woods, A. W. and Bursik, M. I.: Particle fallout, thermal disequilibrium and volcanic plumes, Bulletin of Volcanology, 53, 559–570, 1991.

Wrobel, L. C. and Brebbia, C.: Computational Modelling of Free and Moving Boundary Problems: Fluid flow, vol. 1, Walter de Gruyter, 1991.

Youngs, D. L.: Time-dependent multi-material flow with large fluid distortion, Numerical methods for fluid dynamics, 24, 273–285, 1982.