Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 227 (2008) 3159–3173
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Material point method applied to multiphase flows

Duan Z. Zhang *, Qisu Zou, W. Brian VanderHeyden [1], Xia Ma

Theoretical Division, Fluid Dynamics Group (T-3, B216), Los Alamos National Laboratory, Los Alamos, NM 87545, USA

Received 5 June 2007; received in revised form 9 November 2007; accepted 19 November 2007
Available online 22 Janauary 2008

Abstract

The particle-in-cell method (PIC), especially the latest version of it, the material point method (MPM), has shown significant advantage over the pure Lagrangian method or the pure Eulerian method in numerical simulations of problems involving large deformations. It avoids the mesh distortion and tangling issues associated with Lagrangian methods and the advection errors associated with Eulerian methods. Its application to multiphase flows or multi-material deformations, however, encounters a numerical difficulty of satisfying continuity requirement due to the inconsistence of the interpolation schemes used for different phases. It is shown in Section 3 that current methods of enforcing this requirement either leads to erroneous results or can cause significant accumulation of errors. In the present paper, a different numerical method is introduced to ensure that the continuity requirement is satisfied with an error consistent with the discretization error and will not grow beyond that during the time advancement in the calculation. This method is independent of physical models. Its numerical implementation is quite similar to the common method used in Eulerian calculations of multiphase flows. Examples calculated using this method are presented.

Keywords: Particle-in-cell; Material point method; Multiphase flow; Fluid–structure interaction

1. Introduction

Particle-in-cell (PIC) method has been used in computational fluid mechanics since 1960s [1,2]. Since then many have used, improved and generalized the method to various cases. The work of Sulsky et al. [3] provides a mathematical foundation for the improved version of the method. The method is regarded as a numerical approach of seeking a weak solution to the governing equations. The numerical schemes described in that work and the sequential extensions of it are often referred as the material point methods (MPM). A material point method uses both Eulerian meshes and Lagrangian (material) points to represent a material. In such methods Eulerian meshes stay fixed and the Lagrangian points move through the Eulerian meshes during the material motion and deformation.

* Corresponding author. Tel.: +1 505 665 4428; fax: +1 505 665 5926.
E-mail address: dzhang@lanl.gov (D.Z. Zhang).
[1] Present address: BP North America, 150 West Warrenville Road, Naperville, IL 60563, USA.

0021-9991/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2007.11.021

A distinguished advantage of the particle-in-cell method is its capability of tracking motion of material undergoing large deformation while avoid mesh tangling issues associated with Lagrangian meshes. The use of Lagrangian points in the method avoids the need to advect state variables, such as stress and strain of the material through the Eulerian mesh and avoids the numerical diffusion issues associated with such advection. One of the disadvantages of the material point method is its computational cost. It is more computationally expensive compared to both pure Eulerian and Lagrangian methods because it has to trace motions of the material points and to update quantities on mesh nodes at the same time. Despite this disadvantage, often it is the only credible choice for a practical problem that involves large deformations and complex moving boundaries [4].

The advantage of the material point method is highly desired for solving problems involving interactions of different materials or fluids. Materials involved in such cases often have different constitutive relations. For instance, in the case of porous solid materials filled with fluid, the fluid in the pores can often be modeled as Newtonian and its stress can be calculated from the strain rate. The stress in the solid is often related to strain, which needs to be calculated accumulatively following the path of the material deformation. In this case, we often choose to calculate the fluid motion using an Eulerian mesh and to calculate the solid motion using both the Eulerian mesh and Lagrangian material points, so that the stress and the deformation history of the solid can be traced using the material points. The method has been used to calculate fluid–structure interactions [5] and to calculate the interaction between a tungsten projectile and a steel block [6].

In all of these calculations, materials are regarded as interpenetrating continua. The location of the materials is described by the spatial distribution of the volume fractions of the materials. The continuity condition requires that the volume fractions of all the materials sum to one. This continuity requirement is used to determine the pressure in many calculations of multiphase flows using Eulerian methods [7]. As we shall see in Section 3, due to the difference in the approximations between the material point method and the Eulerian method, erroneous results for pressure will be obtained if the same scheme as used in Eulerian methods for multiphase flows is used to calculate the pressure. To avoid such numerical error, evolution equations for pressure and specific volume are developed [5] based on the assumption of the pressure equilibriums, so that the direct enforcement of the continuity condition can be circumvented by directly advancing these quantities using the evolution equations. It is proved [8] that the set of evolution equations for pressure and specific volume is equivalent to one of the models that satisfies the continuity requirement. Other models are possible.

Many of the problems to be solved using the material point method are continuous multiphase or multi-material problems where the characteristic length scales of the materials are comparable to the length scale of the problem domain. This type of problems is beyond the regime of disperse multiphase flows or particle reinforced composite materials, where there is only one continuous phase, or material, and all other phases or materials are in the form of particles, droplets or bubbles with the characteristic size small compared to the problem domain. Compared to disperse multiphase problems, the models for continuous multiphase flows are still in their infancy. Even for disperse multiphase flows, there are still unsettled issues related to pressure calculations. Evolution equations [5,8] for pressure, the material densities and volume fractions are used to compute physical interactions involved in deformations that determine volume fractions. However, using these equations directly to advance these quantities in time can lead to significant error accumulation as we show in Section 3.

In this paper we introduce a numerical scheme that satisfies the continuity requirement to higher order of accuracy in the sense of weak solutions for the evolution equations for the volume fractions. The nature of weak solutions is consistent with the material point method.

In the following section we briefly describe the material point method. The approach described is not new and has been used by many others [3,4,9]. The reason we list them here is to rewrite them in the context of multiphase flows and to provide the basis for our discussion of the new method of satisfying the continuity constraint. Many of the equations listed are used in later discussions. By briefly describing the method here we also provide a complete description of the method as a convenience to the readers.

2. The material point method

Let q_k be an average quantity contained in a unit mass of phase k at location $\boldsymbol{x}$ and time t. The averaged transport equation for this quantity q_k can be written in the following Lagrangian form [8]:

$$\theta_k \rho_k^0 \frac{\mathrm{d}q_k}{\mathrm{d}t} = \rho_k \left(\frac{\partial q_k}{\partial t} + \boldsymbol{u}_k \cdot \nabla q_k \right) = \nabla \cdot (\theta_k \boldsymbol{L}_k) + \rho_k G_k, \tag{1}$$

where θ_k is the volume fraction, ρ_k^0 is the average material density, $\rho_k = \theta_k \rho_k^0$ is the macroscopic density, $\boldsymbol{u}_k$ is the average velocity of phase k, $\boldsymbol{L}_k$ is the q_k-flux tensor that is one order higher than q_k, and $\rho_k G_k$ is the source density for q_k.

After taking inner product with a continuous trial function, h_k, Eq. (1) can be written as

$$\left(\rho_k \frac{\mathrm{d}q_k}{\mathrm{d}t}, h_k \right) = (\nabla \cdot (\theta_k \boldsymbol{L}_k) + \rho_k G_k, h_k), \tag{2}$$

where $(\cdot, \cdot)$ denotes the inner product. The inner product of two functions q_k and h_k is defined as

$$(q_k, h_k) = \int_\Omega q_k h_k \, \mathrm{d}v, \tag{3}$$

where the integration is over the entire computational domain Ω, and v is the volume in the domain.

The material point method seeks an approximate weak solution to (1) in a subspace of continuous functions in which all functions take the following form:

$$q_k(\boldsymbol{x}, t) = \sum_{j=1}^{N} q_{kj}(t) S_j(\boldsymbol{x}), \tag{4}$$

where N is the number of mesh nodes in the domain, q_{kj} is the value of q_k at node j and S_j is the shape function associated with the node.

By taking the trial function h_k in the same form as (4)

$$h_k = \sum_{\ell=1}^{N} \delta q_{k\ell} S_\ell(\boldsymbol{x}), \tag{5}$$

we can define the weak solution to (1) in this subspace as q_k in form (4) with q_{kj} satisfying the following equation for any trial function of form (5):

$$\sum_{\ell=1}^{N} \sum_{j=1}^{N} m_{k\ell j} \frac{\mathrm{d}q_{kj}}{\mathrm{d}t} \delta q_{k\ell} = \sum_{\ell=1}^{N} \delta q_{k\ell} \left[(\rho_k G_k, S_\ell) - (\theta_k \boldsymbol{L}_k, \nabla S_\ell) + \int_{\partial\Omega} \theta_k \boldsymbol{L}_k \cdot \boldsymbol{n} S_\ell(\boldsymbol{x}) \mathrm{d}S \right], \tag{6}$$

where $\boldsymbol{n}$ is the outward normal on the boundary of the domain Ω, and

$$m_{k\ell j} = \int_\Omega \rho_k S_\ell(\boldsymbol{x}) S_j(\boldsymbol{x}) \mathrm{d}v. \tag{7}$$

Since $\delta q_{k\ell}$ is arbitrary, we have

$$\sum_{j=1}^{N} m_{k\ell j} \frac{\mathrm{d}q_{kj}}{\mathrm{d}t} = \left[(\rho_k G_k, S_\ell) - (\theta_k \boldsymbol{L}_k, \nabla S_\ell) + \int_{\partial\Omega} \theta_k \boldsymbol{L}_k \cdot \boldsymbol{n} S_\ell(\boldsymbol{x}) \mathrm{d}S \right]. \tag{8}$$

This is a system of linear equations for the rate of change of q_k at the nodes. To avoid solving this system of coupled equations, we note that $m_{k\ell j}$ is non-zero only for the nodes (j's) that are within the support (non-zero region) of the shape function S_ℓ. Since these nodes are in the vicinity of node ℓ, the rate $\mathrm{d}q_{kj}/\mathrm{d}t$ can be approximated as $\mathrm{d}q_{k\ell}/\mathrm{d}t$ with a spatial discretization error $\mathrm{O}[(\Delta x)^d]$, where $d = 1$ if the rate is continuous in the space or node ℓ is a boundary node; and $d = 2$ if the rate is smooth (first order differentiable), in the space. With this approximation the system of the linear equations is decoupled and can then be solved as

$$\frac{\mathrm{d}q_{k\ell}}{\mathrm{d}t} = \frac{1}{m_{k\ell}} \left[(\rho_k G_k, S_\ell) - (\theta_k \boldsymbol{L}_k, \nabla S_\ell) + \int_{\partial\Omega} \theta_k \boldsymbol{L}_k \cdot \boldsymbol{n} S_\ell(\boldsymbol{x}) \mathrm{d}S \right] + \mathrm{O}[(\Delta x)^d], \tag{9}$$

where

$$m_{k\ell} = \sum_{j=1}^{N} m_{k\ell j} = \int_\Omega \rho_k S_\ell(\boldsymbol{x}) \mathrm{d}v, \tag{10}$$

since $\sum_{j=1}^{N} S_j(\boldsymbol{x}) = 1$. This approximate way of decoupling the system of equations is equivalent to approximating the matrix formed by elements $m_{k\ell j}(\ell, j = 1, 2, \ldots, N)$ with a diagonal matrix in which the diagonal elements are the sum of the elements in the corresponding rows of the original matrix. The error introduced by this approximation is of order $(\Delta x)^d$. Because of this error, only the linear or bi-linear shape functions are usually used in a material point method. In the present paper, we also restrict ourselves in using these shape functions. This approximation is also known to cause artificial energy dissipation of order $(\Delta t)^2$ in a dynamic system [9].

To calculate the inner product in the first term on the right-hand side of (9), we again write G_k in the form of (4) to find

$$\begin{aligned}(\rho_k G_k, S_\ell) &= \int_\Omega \rho_k \sum_{j=1}^{N} G_{kj} S_j(\boldsymbol{x}) S_\ell(\boldsymbol{x}) \mathrm{d}v \\ &= G_{k\ell} \int_\Omega \rho_k \sum_{n=1}^{N} S_n(\boldsymbol{x}) S_\ell(\boldsymbol{x}) \mathrm{d}v \{1 + \mathrm{O}[(\Delta x)^d]\} \\ &= G_{k\ell} m_{k\ell} \{1 + \mathrm{O}[(\Delta x)^d]\}. \end{aligned} \tag{11}$$

We have again taken advantage of local support of the shape function as in (9) and approximated G_{kj} with $G_{k\ell}$ within the support of S_ℓ. This equation enables the source term G_k, such as the interaction force between phases, to be calculated at the nodes.

To calculate the second inner product on the right-hand side of (9), we now introduce an approximate scheme to calculate the inner product using material point quantities. In a material point method, the domain occupied by phase k material is divided into non-overlapping Lagrangian regions. The union of these regions covers entire domain occupied by phase k material. In principle, these Lagrangian regions are independent of the mesh in the computational domain, but for the convenience often these Lagrangian regions are formed by dividing a mesh cell into several such regions based on the domain occupied by phase k material specified in the initial condition. The sizes of these regions are fractions of the typical mesh size Δx. Initially, the material points are centroids of these Lagrangian regions. The mass and the volume of the region are assigned to the material point. These Lagrangian regions move and deform with the material and hence the material points also move with the material. Although the material point may not be the centroid of the Lagrangian region after a deformation, the distance between them is of order $\mathrm{O}[(\Delta x)^d]$, where $d = 1$ for continuous and $d = 2$ for smooth displacement fields. Because the region is a Lagrangian region, the mass of the material point is constant during the material deformation without a phase change. In this way the shape of the Lagrangian region is not tracked in the computation, but the volume v_{kp} of the region for phase k represented by material point p can be calculated by $v_{kp} = m_{kp}/\rho^0_{kp}$, where m_{kp} is the mass of the material point and ρ^0_{kp} is the material density at the material point. With the material points so constructed, the inner product $(\rho_k q_k, h_k)$ can be approximately calculated based on the Lagrangian regions as

$$(\rho_k q_k, h_k) = \sum_{p=1}^{N_{kp}} m_{kp} q_{kp} h_{kp} + \mathrm{O}[(\Delta x)^d], \tag{12}$$

where N_{kp} is the number of material points for phase k material in the domain, and q_{kp} and h_{kp} are the values of q_k and h_k evaluated at material point p. The error estimate in (12) is obtained from the relation between the material points and the centroids of the Lagrangian regions.

In this way to seek a weak solution for (1) as defined in (6), the second inner product on the right-hand side of (9) is approximated as

$$(\theta_k \boldsymbol{L}_k, \nabla S_\ell) = (\rho_k \boldsymbol{L}_k/\rho^0_k, \nabla S_\ell) \approx \sum_{p=1}^{N_{kp}} v_{kp} \boldsymbol{L}_k(\boldsymbol{x}_{kp}, t) \cdot \nabla S_\ell(\boldsymbol{x}_{kp}). \tag{13}$$

For the case where there are not enough material points in the elements surrounding node ℓ, the accuracy of this approximation could be reduced. For more accurate and stable calculation, this inner product could be calculated using Gaussian integration scheme, but this subject is beyond the scope of the present work and we

will not discuss it further. Since S_ℓ has a local support, the summation only needs to be carried out for the material points in the elements surrounding node ℓ.

With the right-hand side calculated using (11) and (13) we can write (9) as

$$\frac{\mathrm{d}q_{k\ell}}{\mathrm{d}t} \approx G_{k\ell} - \frac{1}{m_{k\ell}} \sum_{p=1}^{N_{kp}} v_{kp} \boldsymbol{L}_k(\boldsymbol{x}_{kp}, t) \cdot \nabla S_\ell(\boldsymbol{x}_{kp}) + \frac{1}{m_{k\ell}} \int_{\partial\Omega} \theta_k \boldsymbol{L}_k \cdot \boldsymbol{n} S_\ell(\boldsymbol{x}) \mathrm{d}S. \tag{14}$$

The last term in (14) represents the effect of boundaries.

With the rate of change for q_k calculated we can advance q_k to the next time step as

$$q_{k\ell}^L = q_{k\ell}^n + \frac{\mathrm{d}q_{k\ell}}{\mathrm{d}t} \Delta t, \tag{15}$$

where the superscript n denotes time step n and the superscript L denotes that this time advancement follows the material point, the Lagrangian value, since the derivative is the solution to the evolution equation (1) in the Lagrangian form. To update q_k values on material points we interpolate the rate of change, or the material derivative, from the nodes to material points as

$$q_{kp}^{n+1} = q_{kp}^n + \sum_{\ell=1}^{N} (q_{k\ell}^L - q_{k\ell}^n) S_\ell(\boldsymbol{x}_{kp}^n). \tag{16}$$

It is important to note that we interpolate the change $q_{k\ell}^L - q_{k\ell}^n$, not $q_{k\ell}^L$, to the material points. In this way, the change of material point values is only caused by the right-hand side of (14). If the right-hand side of (14) vanishes, the material point value does not change. Therefore, such node to material point interpolation does not introduce numerical diffusion to the solution. If we interpolate Lagrangian node values $q_{k\ell}^L$ to material points, significant numerical diffusion would occur. We also note that the value for the shape function S_ℓ is evaluated at the time step n. This is because material points are Lagrangian points. In this Lagrangian step they follow the motion of the material and the shape function is defined in the coordinate system that moves and deforms with the material. Therefore there is no relative motion between the material points and the coordinate system during the time advancement, and the values of the shape functions remain unchanged at the material point locations. Following the motion of the material, the new positions $\boldsymbol{x}_{kp}$ of the material points representing phase k are calculated as

$$\boldsymbol{x}_{kp}^{n+1} = \boldsymbol{x}_{kp}^n + \boldsymbol{u}_{kp} \Delta t, \tag{17}$$

where velocity $\boldsymbol{u}_{kp}$ is the velocity $\boldsymbol{u}_k$ for phase k material at the location of material point p,

$$\boldsymbol{u}_{kp} = \sum_{\ell=1}^{N} \boldsymbol{u}_{k\ell}^L S_\ell(\boldsymbol{x}_{kp}^n), \tag{18}$$

with $\boldsymbol{u}_{k\ell}^L$ calculated using $q_k = \boldsymbol{u}_k$ in (15). Again, the value for the shape function S_ℓ in (18) is evaluated at the time step n for the same reason as in (16). The velocity $\boldsymbol{u}_{kp}$ used to advance the material point positions is not the material point velocity, but rather the velocity interpolated from the nodes, because material points are Lagrangian points following the motion and deformation of the material, while a material point velocity should be regarded as the averaged momentum per unit mass carried by the material point. The velocity gradient used to calculate the stresses or to advance the strains on material points is calculated by differentiating the shape functions in (18).

Note that the change rate $\mathrm{d}q_{k\ell}/\mathrm{d}t$ calculated in (14) is the Lagrangian rate following the motion of the material. The time advanced $q_{k\ell}^{n+1}$ on a fixed node needs to be calculated using the updated values q_{kp}^L on material points. To obtain the scheme of calculating node values using material point values, we note that the inner product $(\rho q, h)$ can also be calculated using (4) to find

$$(\rho_k q_k, h_k) = \sum_{\ell=1}^{N} \sum_{j=1}^{N} m_{k\ell j} q_{kj} h_{k\ell}, \tag{19}$$

where $m_{k\ell j}$ is defined in (7).

Comparing (12) with (19) and then using (4) for h_k we find

$$\sum_{\ell=1}^{N}\sum_{j=1}^{N} m_{k\ell j} q_{kj} h_{k\ell} = \sum_{\ell=1}^{N}\sum_{p=1}^{N_{kp}} m_{kp} q_{kp} h_{k\ell} S_\ell(\boldsymbol{x}_{kp}) + \mathrm{O}[(\Delta x)^d]. \tag{20}$$

Since h_k is an arbitrary smooth function, we have

$$\sum_{j=1}^{N} m_{k\ell j} q_{kj} = \sum_{p=1}^{N_{kp}} m_{kp} q_{kp} S_\ell(\boldsymbol{x}_{kp})\{1 + \mathrm{O}_{\mathrm{w}}[(\Delta x)^d]\}, \tag{21}$$

where the subscript w in the error term reminds that this error estimate is given in the sense of the weak solution. A common mistake is to treat (21) as a pointwise approximation with $d = 2$ thinking that (21) can be derived from (20) by letting $h_{k\ell} = 1$ and $h_{kj} = 0 (j \neq \ell)$. The problem with this derivation is that such defined function h_k is not a smooth function, whereas (20) only holds for a smooth function if $d = 2$. Therefore we have only $d = 1$ if (21) is regarded as a pointwise approximation. Instead of treating both sides of (21) as pointwise individual values, we should regard the values specified by both sides of (21) as an approximate way of representing two smooth functions by their values on the mesh nodes. The error estimate $\mathrm{O}_{\mathrm{w}}[(\Delta x)^d]$ in (21) states that, for the two functions represented by both sides of (21), the inner products of these two functions with another smooth function approach each other quadratically as mesh size decreases. As a consequence of this, if the values of both sides of (21) are averaged over the mesh nodes within a fixed physical domain, the difference between the averages approaches zero quadratically as mesh refines.

For $q_k = 1$ in (21), we find that $m_{k\ell}$ defined in (10), can be approximately calculated using

$$m_{k\ell} = \sum_{j=1}^{N} m_{k\ell j} = \sum_{p=1}^{N_{kp}} m_{kp} S_\ell(\boldsymbol{x}_{kp})\{1 + \mathrm{O}_{\mathrm{w}}[(\Delta x)^d]\}. \tag{22}$$

Eq. (21) is a set of coupled equations for q_{kj} at the nodes. Again because of the local support of the shape functions, by approximating q_{kj} with $q_{k\ell}$ we have

$$q_{k\ell} = \frac{\sum_{p=1}^{N_{kp}} m_{kp} q_{kp} S_\ell(\boldsymbol{x}_{kp})}{m_{k\ell}}\{1 + \mathrm{O}[(\Delta x)^d]\} = \frac{\sum_{p=1}^{N_{kp}} m_{kp} q_{kp} S_\ell(\boldsymbol{x}_{kp})}{\sum_{p=1}^{N_{kp}} m_{kp} S_\ell(\boldsymbol{x}_{kp})}\{1 + \mathrm{O}[(\Delta x)^d]\}. \tag{23}$$

By letting $q_k = 1/\rho_k^0$ in (23), we find

$$\rho_{k\ell}^0 = \frac{\sum_{p=1}^{N_{kp}} m_{kp} S_\ell(\boldsymbol{x}_{kp})}{\sum_{p=1}^{N_{kp}} v_{kp} S_\ell(\boldsymbol{x}_{kp})}\{1 + \mathrm{O}[(\Delta x)^d]\}, \tag{24}$$

for the material density at node ℓ, where $v_{kp} = m_{kp}/\rho_k^0$ is the volume of the material point. Although for $d = 2$ the error estimates in (21) and (22) are only in the sense of weak solution, the error estimates in (23) and (24) are pointwise provided that the displacements from the initial material point positions are spatially smooth, because the node quantity is approximated as the mass weighted mean of its surrounding material points and the distance between a material point and the centroid of the Lagrangian region is of order $\mathrm{O}[(\Delta x)^2]$. The first order errors in Δx in the numerator and in the denominator cancel each other in this case.

Because the main subject of this paper is about the material point method, which seeks a weak solution to the equations, and also because the second order spatial accuracy can be simply obtained by averaging over several neighboring nodes for smooth fields, there is little advantage to distinguish whether an error estimate is pointwise or is in the sense of weak solution. We will not distinguish them further, and drop the subscript w in the error estimate.

3. Calculation of volume fractions and pressures

Using (10) and (22), the macroscopic density $\rho_{k\ell}$ at node ℓ can be approximated as

$$\rho_{k\ell} = \frac{m_{k\ell}}{V_\ell} + \mathrm{O}[(\Delta x)^d] = \frac{\sum_{p=1}^{N_{kp}} m_{kp} S_\ell(\boldsymbol{x}_{kp})}{V_\ell}\{1 + \mathrm{O}[(\Delta x)^d]\}, \tag{25}$$

where

$$V_\ell = \int_\Omega S_\ell(\boldsymbol{x})\mathrm{d}v \tag{26}$$

is the volume associated with the node. The volume fraction θ_k can be approximately calculated as

$$\theta_{k\ell} = \frac{\rho_{k\ell}}{\rho^0_{k\ell}} = \frac{\sum_{p=1}^{N_{kp}} v_{kp} S_\ell(\boldsymbol{x}_{kp})}{V_\ell}\{1 + \mathrm{O}[(\Delta x)^d]\}. \tag{27}$$

For phases that are not represented by material points, the macroscopic density is calculated from the mass conservation equation [7,8]:

$$\frac{\partial \rho_k}{\partial t} + \nabla \cdot (\boldsymbol{u}_k \rho_k) = \rho^0_{kc}\dot{\phi}_k, \tag{28}$$

where $\dot{\phi}_k$ is the average rate of volume generation due to phase change and ρ^0_{kc} is the material density at phase change. The volume fractions, $\theta_{k\ell}$, at node ℓ for those phases are approximated as the ratio $\rho_{k\ell}/\rho^0_{k\ell}$ with the same order of the spatial discretization error $\mathrm{O}[(\Delta x)^d]$. Since the material densities are functions of pressures of the phases, the pressures can be found by enforcing the continuity condition

$$\sum_{k=1}^{M} \theta_k = 1, \tag{29}$$

where M is the number of the phases in the calculation. In many computations of multiphase flows using Eulerian methods [7], this equation is indeed used to find the pressures. However using the volume fraction approximately calculated using (27) for the phases represented by material points in this equation will result in failure of the calculation. To show this, we define a function

$$f(\Delta p, t) \equiv \sum_{k=1}^{M} \frac{\rho_k}{\rho^0_k} = \sum_{k=1}^{M} \theta_k(\Delta p, t), \tag{30}$$

where Δp is the pressure increment. In this definition for function f, the material density ρ^0_k is a function of the pressure increment through the equation of state. If a time explicit method is used, the macroscopic density is independent of the pressure increment. If a time implicit method is used the macroscopic density is a function of the pressure increment through the pressure gradient term in the momentum equation. The pressures for the phases may be the same or different, depending on the physical models used.

If an Eulerian method is used for all the phases as in many multiphase flow calculations [7], the consistence in spatial discretization for all the phases ensures that $f(0, t^n) = 1$ is exact at the beginning of the time step from n to $n+1$. For the cases where calculations of the volume fractions are different as described above, one can only have $f(0, t^n) = 1 + \mathrm{O}[(\Delta x)^d]$. In other words, the sum of the volume fractions is not exactly one if the volume fractions are calculated using different sets of material points for different phases, or if some of the volume fractions are calculated using an Eulerian method and the others are calculated using the material point method. We now show that this numerical error propagates and eventually contaminates the solution if we require the function defined in (30) to be one in finding the pressure increment Δp, even though all the methods of calculating the volume fractions are equally valid and have the same order of accuracy individually. The use of (30) to find Δp is equivalent to solving the following equation:

$$f(0, t^n) + \frac{\partial f}{\partial t}\Delta t + \frac{\partial f}{\partial \Delta p}\Delta p = 1, \tag{31}$$

where, according to the Taylor theorem, the partial derivatives are evaluated at a point between t^n and $t^n + \Delta t$ for time and between 0 and Δp for the pressure increment. Since $f(0, t^n) = 1 + \mathrm{O}[(\Delta x)^d]$, we have

$$\frac{\partial f}{\partial t} + \frac{\partial f}{\partial \Delta p}\frac{\Delta p}{\Delta t} = \frac{\mathrm{O}[(\Delta x)^d]}{\Delta t} \tag{32}$$

for Δp. In many calculations of dynamical problems the time step is proportional to the mesh size Δx; therefore the error on the right-hand side of (32), or $\Delta p/\Delta t$ is $[\mathrm{O}(\Delta x)^{d-1}]$. In this way, the pressure calculated using $p_k^{n+1} = p_k^n + \Delta p$ accumulates such errors. For a problem with characteristic time T, within $T/\Delta t$ time steps, or at the end of period T, the accuracy of the pressure reduces to $\mathrm{O}[(\Delta x)^{d-1}]$, and in d such periods the pressure calculation fails. The material point method described in Section 1 has a second order spatial accuracy in the region where relevant quantities vary smoothly. For sharp interface regions or boundaries the accuracy reduces to first order. For such cases the pressure calculated using (32) is erroneous (i.e. the error is zeroth order). Indeed, if this method were used to calculate the pressure in the case of an elastic body translating through a mesh with the same uniform velocity as the surrounding air, one would see pressure changes where none should exist. For problems involving diffusion, if an explicit method is used, the problem will be even worse because Δt is restricted by $(\Delta x)^2$.

To satisfy the continuity requirement while avoiding the numerical error, efforts [5] have been made to directly solve the evolution equations for material density or pressures. This approach is proved to be equivalent to directly solving the evolution equations for the volume fractions [8] and has been practiced for the equilibrium pressures model [5]. The evolution equations for the volume fraction and the average material density can be written as [8]

$$\frac{\partial \theta_k}{\partial t} + \nabla \cdot (\theta_k \boldsymbol{u}_k) = \theta_k \langle \nabla \cdot \boldsymbol{u}_k \rangle + \dot{\phi}_k, \tag{33}$$

$$\theta_k \left(\frac{\partial \rho_k^0}{\partial t} + \boldsymbol{u}_k \cdot \nabla \rho_k^0 \right) = \theta_k \rho_k^0 \langle \nabla \cdot \boldsymbol{u}_k \rangle + (\rho_{kc}^0 - \rho_k^0)\dot{\phi}_k, \tag{34}$$

where $\langle \nabla \cdot \boldsymbol{u}_k \rangle$ is the average rate of the volumetric expansion in the material of phase k. The angular bracket, $\langle \cdot \rangle$ represents the ensemble average [8]. The divergence of the averaged velocity $\nabla \cdot \boldsymbol{u}_k$ is not necessary equal to the average rate of the volumetric expansion in a multiphase flow. For instance, in a multiphase flow containing air and sand grains, the divergence of the velocity field for sand is quite different from the volumetric expansion of sand grains. In fact, in most practical situations the sand grains can be regarded as rigid with zero volumetric expansion, while the divergence of the average velocity of sand is not usually zero as sand grains accumulate or disperse in the flow.

The summation of (33) over all the phases leads to

$$\frac{\partial}{\partial t} \sum_{k=1}^{M} \theta_k + \nabla \cdot (\boldsymbol{u}_\mathrm{m}) = \sum_{k=1}^{M} (\theta_k \langle \nabla \cdot \boldsymbol{u}_k \rangle + \dot{\phi}_k), \tag{35}$$

where

$$\boldsymbol{u}_\mathrm{m} = \sum_{k=1}^{M} \theta_k \boldsymbol{u}_k \tag{36}$$

is the mixture velocity and M is the number of phases. The continuity condition (29), is equivalent to

$$\nabla \cdot \boldsymbol{u}_\mathrm{m} - \sum_{k=1}^{M} (\theta_k \langle \nabla \cdot \boldsymbol{u}_k \rangle + \dot{\phi}_k) = 0, \tag{37}$$

since if this equation is satisfied the sum of the volume fractions is one if it is initially so. Using this relation, models for the average rate $\langle \nabla \cdot \boldsymbol{u}_k \rangle$ of the volumetric expansion have been derived [5,8]. For a multipressure model

$$\langle \nabla \cdot \boldsymbol{u}_k \rangle = \alpha_k \nabla \cdot \boldsymbol{u}_k + B_k, \tag{38}$$

where α_k is a parameter specified by the model and the term B_k is related to compressibilities of the phases

$$B_k = \frac{1/(\langle \rho_k^0 \rangle c_k^2)}{\sum_{i=1}^{N} \theta_i/(\langle \rho_i^0 \rangle c_i^2)} \sum_{i=1}^{M} \left[\nabla \cdot (\theta_i \widetilde{\boldsymbol{u}}_i) - \alpha_i \theta_i \nabla \cdot \widetilde{\boldsymbol{u}}_i - \dot{\phi}_i \right], \tag{39}$$

with c_k being the speed of sound in phase k material. The equilibrium pressure or the single pressure model can be treated [8] as a special case of the multipressure model by setting $\alpha_k = 0$.

Using (38) and (39) one can calculate $\langle \nabla \cdot \boldsymbol{u}_k \rangle$ at time level n (or at time level $n+1$, if an implicit method is used) and advance the volume fractions and the material densities in time according to (33) and (34). In this scheme, Eq. (37) is only exactly satisfied at time t^n (or t^{n+1} for an implicit method) since the models are derived based on this equation. During time interval between t^n and t^{n+1}, Eq. (37) is not exactly satisfied but with an error $O(\Delta t)$. Using (35), after a time integration, we find that the sum of the volume fractions at time level $n+1$ is

$$\sum_{k=1}^{M} \theta_k^{n+1} = \sum_{k=1}^{M} \theta_k^n + O[(\Delta x)^d \Delta t] + O[(\Delta t)^2], \tag{40}$$

where the error term $O[(\Delta x)^d \Delta t]$ results from the calculation of $\langle \nabla \cdot \boldsymbol{u}_k \rangle$ using approximated volume fractions and the error term $O[(\Delta t)^2]$ results from the $O(\Delta t)$ error in satisfying (37) during the time interval from t^n to t^{n+1}. Although such scheme is usable to find the pressures and volume fractions, the error accumulation could be significant. The $O(\Delta t)$ error in continuity constraint (37) leads to an error in the pressure. This error in pressure can be estimated as $O(\rho_k c_k^2 \Delta t/T)$, where T is the characteristic time scale of the problem. Comparing the advection terms in the momentum equation with this error, we find that this error in pressure can only be neglected if $O[(\Delta t/T)/M_k^2]$ is small, where M_k is the Mach number for phase k. For low Mach number flows this error could be significant although it decreases with the time step Δt.

According to this analysis, the most important error is not the error in the sum of the volume fractions but the error in satisfying (37), because the later error carries significant dynamical consequence while the error in the volume fraction sum is kinematic. In the time advancing scheme directly using the evolution equations, the kinematic error can be generated, because the sum of the volume fractions at time level t^{n+1} can deviate from unity even if the sum is exactly one at the previous time level and the motion of the material point does not introduce any additional error. This volume fraction error causes error in the mixture compressibility as implied by (39). To regulate this error, the volume fractions need to be normalized by redefining the volume fractions as $\theta_k/\sum_{i=1}^{M} \theta_i$. However such normalization does not affect the error already contained in the material densities and pressures. In this way the kinematic error generated in the time advancing scheme spreads and becomes the dynamical error. Indeed, this is the reason for using (29) instead of this scheme for pressure calculations in modern Eulerian methods for multiphase flow.

In a numerical calculation, although it is difficult to isolate errors and to prevent the kinematic error from spreading into and becoming dynamical errors, such error spreading can be minimized. To do so, we now introduce a numerical method that spreads the kinematic error less than the scheme of directly advancing the volume fractions or material densities using the values for $\langle \nabla \cdot \boldsymbol{u}_k \rangle$ calculated at some specific time levels. The new method does not generate kinematic errors during the enforcement of the continuity constraint although it cannot prevent the generation of the error of $O(\Delta t)$ due to the motion of the material points. In this scheme, if the volume fraction sum is one at the beginning of a time step, the volume fraction would remain to be one if the motion of the material points were not to create a kinematic error. In other words, in the new scheme, although the kinematic process still introduces an error of $O(\Delta t)$, Eq. (37) is satisfied within the error limited to $O[(\Delta x)^d \Delta t]$, instead of $O(\Delta t)$, and then the pressure error caused by the kinematic error becomes $O[(\Delta x/L)^d \Delta t/T/M_k^2]$, where L is the characteristic length scale in the problem. In this way, although the scheme is still first order in the time step, the extra factor $(\Delta x/L)^d$ reduces the error in pressure caused by the kinematic error resulting from using different but equivalently valid approaches to approximate the volume fractions. This new scheme is based on the weak solution to the evolution equations for the volume fractions and is model independent. Its numerical implementation is quite similar to the commonly used methods in pure Eulerian calculations.

4. Weak solution for volume fraction equations

To derive this scheme, we add and subtract $\nabla \cdot (\theta_k \boldsymbol{u}_\mathrm{m})$ from the left-hand side of (33), and rewrite the equation as

$$\frac{\mathrm{d}_\mathrm{m}\theta_k}{\mathrm{d}_\mathrm{m}t} + \nabla \cdot [\theta_k(\boldsymbol{u}_k - \boldsymbol{u}_\mathrm{m})] + \theta_k \nabla \cdot \boldsymbol{u}_\mathrm{m} = \theta_k \langle \nabla \cdot \boldsymbol{u}_k \rangle + \dot{\phi}_k, \tag{41}$$

where

$$\frac{\mathrm{d}_\mathrm{m}\theta_k}{\mathrm{d}_\mathrm{m}t} = \frac{\partial \theta_k}{\partial t} + \boldsymbol{u}_\mathrm{m} \cdot \nabla \theta_k \tag{42}$$

is the material derivative following the mixture velocity. To seek an approximate weak solution, we multiply both sides of (41) by a trial function h in the form of (4) defined on the frame moving with the mixture velocity $\boldsymbol{u}_\mathrm{m}$, and then integrate the resulting equation over the entire computational domain to find

$$\sum_{j=1}^{N} h_j \frac{\mathrm{d}_\mathrm{m} v_{kj}}{\mathrm{d}_\mathrm{m}t} = -\sum_{j=1}^{N} h_j \left\{ \int_\Omega S_j(\boldsymbol{x}) \nabla \cdot [\theta_k(\boldsymbol{u}_k - \boldsymbol{u}_\mathrm{m})] \mathrm{d}v + \int_\Omega S_j(\boldsymbol{x}) \Big(\theta_k \langle \nabla \cdot \boldsymbol{u}_k \rangle + \dot{\phi}_k - \theta_k \nabla \cdot \boldsymbol{u}_\mathrm{m} \Big) \mathrm{d}v \right\}, \tag{43}$$

where

$$v_{kj} = \int_\Omega \theta_k(\boldsymbol{x}, t) S_j(\boldsymbol{x}) \mathrm{d}v \tag{44}$$

and

$$\frac{\mathrm{d}_\mathrm{m} v_{kj}}{\mathrm{d}_\mathrm{m}t} = \frac{\partial v_{kj}}{\partial t} + \int_\Omega S_j(\boldsymbol{x}) \boldsymbol{u}_\mathrm{m} \cdot \nabla \theta_k \mathrm{d}v. \tag{45}$$

Since h_j is arbitrary, we then have

$$\frac{\mathrm{d}_\mathrm{m} v_{kj}}{\mathrm{d}_\mathrm{m}t} = -\int_\Omega S_j(\boldsymbol{x}) \nabla \cdot [\theta_k(\boldsymbol{u}_k - \boldsymbol{u}_\mathrm{m})] \mathrm{d}v + \int_\Omega S_j(\boldsymbol{x}) \Big(\theta_k \langle \nabla \cdot \boldsymbol{u}_k \rangle + \dot{\phi}_k - \theta_k \nabla \cdot \boldsymbol{u}_\mathrm{m} \Big) \mathrm{d}v. \tag{46}$$

Summing (46) over all phases, we find that, in the sense of a weak solution, to satisfy (37) within an error of $\mathrm{O}[(\Delta x)^d \Delta t]$ is equivalent to satisfying the following equation within the same order of error:

$$\frac{\mathrm{d}_\mathrm{m}}{\mathrm{d}_\mathrm{m}t} \sum_{k=1}^{M} v_{kj} = -\int_\Omega S_j(\boldsymbol{x}) \nabla \cdot \sum_{k=1}^{M} [\theta_k(\boldsymbol{u}_k - \boldsymbol{u}_\mathrm{m})] \mathrm{d}v + V_j \mathrm{O}[(\Delta x)^d \Delta t], \tag{47}$$

where V_j is defined in (26). Using (36), we find that the first term on the right-hand side of (47) is zero if the continuity condition (29) is satisfied exactly. When the approximated volume fractions are used, $\sum_{k=1}^{M} [\theta_k(\boldsymbol{u}_k - \boldsymbol{u}_\mathrm{m})]$ is a quantity of $\mathrm{O}[(\Delta x)^d]$. To avoid this error, we now change the way of calculating the mixture velocity as

$$\boldsymbol{u}_\mathrm{m} = \sum_{k=1}^{M} \theta_k \boldsymbol{u}_k \Big/ \sum_{k=1}^{M} \theta_k. \tag{48}$$

This mixture velocity is, of course, the same as the definition (36) for the mixture velocity if the continuity condition (29) is satisfied exactly and has the same order of accuracy as that calculated using the original definition (36) when (29) is only satisfied with an error $\mathrm{O}[(\Delta x)^d]$. In this way the first term on the right and side of (47) vanishes identically; and (47) becomes

$$\frac{\mathrm{d}_\mathrm{m}}{\mathrm{d}_\mathrm{m}t} \sum_{k=1}^{M} v_{kj} = \frac{\partial}{\partial t} \sum_{k=1}^{M} v_{kj} + \int_\Omega S_j(\boldsymbol{x}) \boldsymbol{u}_\mathrm{m} \cdot \nabla \sum_{k=1}^{M} \theta_k \mathrm{d}v = V_j \mathrm{O}[(\Delta x)^d \Delta t]. \tag{49}$$

To calculate v_{kj} defined in (44) for the phases represented by the material points, we can approximate the volume integral by the sum over all material point volumes as in (21) with $q_k = 1/\rho_k^0$ and then use (27) to find

$$v_{kj} = (\rho_k/\rho_k^0, S_j) = \{1 + \mathrm{O}[(\Delta x)^d]\} \sum_{p=1}^{N_{kp}} v_{kp} S_j(\boldsymbol{x}_{kp}) = \theta_{kj} V_j \{1 + \mathrm{O}[(\Delta x)^d]\}. \tag{50}$$

For the phases that are calculated using an Eulerian method we can simply approximate v_{kj} by

$$v_{kj} = \theta_{kj} V_j \{1 + \mathrm{O}[(\Delta x)^d]\} \tag{51}$$

for $d \leqslant 2$.

Since V_j is a constant in time, by dividing V_j across (49), the equation becomes

$$\left[\frac{\partial}{\partial t} \sum_{k=1}^{M} \theta_{kj} + \left(\boldsymbol{u}_{\mathrm{m}} \cdot \nabla \sum_{k=1}^{M} \theta_k\right)_j\right] \{1 + \mathrm{O}[(\Delta x)^d]\} = \mathrm{O}[(\Delta x)^d \Delta t], \tag{52}$$

where the subscript j outside the round brackets denotes that it is evaluated at node j. By integrating this equation from time t^n to t^{n+1} we have

$$\left[\sum_{k=1}^{M} \theta_{kj}^{n+1} - \sum_{k=1}^{M} \theta_{kj}^{n} + \int_{t^n}^{t^{n+1}} \left(\boldsymbol{u}_{\mathrm{m}} \cdot \nabla \sum_{k=1}^{M} \theta_k\right)_j \mathrm{d}t\right] \{1 + \mathrm{O}[(\Delta x)^d]\} = \mathrm{O}[(\Delta x)^d (\Delta t)^2]. \tag{53}$$

Within an error $\mathrm{O}(\Delta t)$, the integrand in the time integral can be approximated as $(\boldsymbol{u}_{\mathrm{m}}^{\eta} \cdot \nabla \sum_{k=1}^{M} \theta_{kj}^{\eta})$ $[1 + \mathrm{O}(\Delta t)]$, where $\eta = n$ for a time explicit method and $\eta = n + 1$ for a time implicit method. Substituting this into (53), we find

$$\left(\sum_{k=1}^{M} \theta_{kj}^{n+1} - \sum_{k=1}^{M} \theta_{kj}^{n} + \boldsymbol{u}_{\mathrm{m}}^{\eta} \cdot \nabla \sum_{k=1}^{M} \theta_{kj}^{\eta} \Delta t\right) \{1 + \mathrm{O}[(\Delta x)^d]\} + \boldsymbol{u}_{\mathrm{m}}^{\eta} \cdot \nabla \sum_{k=1}^{M} \theta_{kj}^{\eta} \mathrm{O}(\Delta t)^2 = \mathrm{O}[(\Delta x)^d (\Delta t)^2]. \tag{54}$$

Noting $\sum_{k=1}^{M} \theta_{kj}^{\eta} = 1 + \mathrm{O}[(\Delta x)^d]$, we find that $\nabla \sum_{k=1}^{M} \theta_{kj}^{\eta} = \mathrm{O}[(\Delta x)^d]$, and that the last term on the left-hand side can be combined into the right-hand side. Therefore, with the mixture velocity $\boldsymbol{u}_{\mathrm{m}}$ calculated using (48), if

$$\sum_{k=1}^{M} \theta_{kj}^{n+1} - \sum_{k=1}^{M} \theta_{kj}^{n} + \boldsymbol{u}_{\mathrm{m}}^{\eta} \cdot \nabla \sum_{k=1}^{M} \theta_{kj}^{\eta} \Delta t = \mathrm{O}[(\Delta x)^d (\Delta t)^2] \tag{55}$$

holds, then Eq. (37) is satisfied within an error of $\mathrm{O}[(\Delta x)^d \Delta t]$. Interestingly, Eq. (55) implies that the sum of the volume fractions is "incompressible" following the mixture velocity calculated using (48).

In the numerical implementation described in the following section, Eq. (55) is used to enforce the continuity condition (37).

5. Numerical implementation

After substituting (38) into (34), the evolution equation of the material density becomes

$$\theta_k \left(\frac{\partial \rho_k^0}{\partial t} + \boldsymbol{u}_k \cdot \nabla \rho_k^0\right) = \theta_k \rho_k^0 \alpha_k \nabla \cdot \boldsymbol{u}_k + (\rho_{kc}^0 - \rho_k^0) \dot{\phi}_k + \theta_k \rho_k^0 B_k. \tag{56}$$

With this evolution equation for the material density, the numerical scheme introduced in the present paper can be implemented in the time advancement from t^n to t^{n+1} as follows:

(1) For phases calculated using the Eulerian method, ρ_k is the value of macroscopic density solved from the mass conservation equation (28).
(2) For the phases calculated using the material point method, the macroscopic density ρ_k is calculated using (25). The volume V_ℓ defined in (26) is not typically calculated in a material point method. In many meshes, such as triangle and quadrilateral elements in two dimensional problems and tetrahedral and hexahedral elements in three dimensional problem, this volume can be approximated by the control volume, V_ℓ^C, of node ℓ, provided that the meshes are not significantly distorted.

(3) For the phases calculated using the material point method, calculate interim microscopic density $\langle \rho^0_{kn} \rangle^*$ on material points using (56) by neglecting the effects of B_k, and with the velocity divergence calculated by taking the divergence of velocity (18). Then interpolate such calculated interim material density to nodes using (24).
(4) For phases calculated using the Eulerian method, the material density at the node is calculated by solving (56) with the B_k term neglected.
(5) Use the equation of state for the phase to find an interim pressure p^*_k for all the phases according to the calculated interim material densities at the nodes.
(6) Find a common pressure increment Δp such that the volume fractions, calculated as the ratio $\rho_{kj}(p^*_k + \Delta p)/\rho^0_{kj}(p^*_k + \Delta p)$ between the macroscopic density and the material density, satisfy (55), or equivalently

$$\sum_{k=1}^{M} \left[\frac{\rho_{kj}(p^*_k + \Delta p)}{\rho^0_{kj}(p^*_k + \Delta p)} \right]^{n+1} = \left[\sum_{k=1}^{M} \theta_{kj} \right]^{n} - \left(\boldsymbol{u}^{\eta}_{\mathrm{m}} \cdot \nabla \sum_{k=1}^{M} \theta^{\eta}_{k} \right)_j \Delta t. \tag{57}$$

The right-hand side of (57) can be regarded as a predicted volume fraction sum at time t^{n+1} using (52). The last term in this equation can be evaluated at time t^{n+1} or t^n ($\eta = n+1$ or n), depending on whether an implicit or explicit method is used. For numerical codes equipped with a good divergence operator, it is often convenient to calculate the factor in the last term as

$$\left(\boldsymbol{u}^{\eta}_{\mathrm{m}} \cdot \nabla \sum_{k=1}^{M} \theta^{\eta}_{k} \right)_j = \nabla \cdot \left(\boldsymbol{u}^{\eta}_{\mathrm{m}} \sum_{k=1}^{M} \theta^{\eta}_{k} \right)_j - \left(\sum_{k=1}^{M} \theta^{\eta}_{k} \nabla \cdot \boldsymbol{u}^{\eta}_{\mathrm{m}} \right)_j. \tag{58}$$

In this approach the effects of B_k in (56) is accounted for by finding Δp and then changing the material densities and volume fractions accordingly to satisfy (57). In an explicit time advancing scheme Eq. (57) is point wise and involves only the local variables at the node j, and therefore can be solved easily. For implicit schemes, a mesh wide system must be solved.

The method of satisfying the continuity requirement introduced here is independent of physical models. The equilibrium pressure model can be implemented as a special case of the procedures outlined above by skipping steps 3–5. This method is similar to the typical pressure calculation procedures in an Eulerian code for multiphase flows; therefore it can be implemented without significant modification of an Eulerian code.

Using (57) we also find that the change of the volume fraction sum during this procedure of finding the pressures is $O[(\Delta x)^d \Delta t]$. Therefore, the error accumulation due to this pressure calculation is limited to $O[(\Delta x)^d]$, the same order of error due to the spatial discretization. In the case of spatially smoothly varying volume fractions, $d = 2$ and the error in the sum of the volume fractions is second order in Δx.

6. Numerical examples

The numerical procedures outlined in Section 5 have been implemented into a numerical code. Several cases of fluid–structure interactions and multi-material interactions have been calculated. In our recent paper [8] we have presented a numerical example of multiphase flow involving spalling of porous material with air filled pores calculated by the new method described here. In this example, the air flow around the porous solid and inside the pores of the solid is calculated using the Eulerian approach and the motion of the solid is calculated using the material point method. Good agreement between the numerical result and the theoretical results is observed.

To confirm the error analysis presented in this paper we perform a one-dimensional calculation of a porous solid bar immersed in a heavy gas in domain $[0, 1]$ cm with periodic boundary conditions on both ends. The material properties of the solid bar are set as copper with initial density of 8.9 g/cm^3. The initial volume fraction for the solid is set as $0.5 + 0.4\sin(2\pi x)$ ($0 \leqslant x \leqslant 1$). The solid bar is modeled as an elastic material with Young's modulus 125 GPa. The gas is assumed to be an ideal gas, with density calculated as Bp, where $B = 1.15 \times 10^{-5}$ g/cm^3/Pa. The initial pressure is set to be one atmosphere ($p = 1.013 \times 10^5$ Pa). In this example, the gas is 1000 times heavier and stiffer than the air so that appreciable gas–solid interactions can take

place. The gas–solid mixture is set into vibration by imposing initial velocity $100\sin(2\pi x)$ cm/s to both phases. In the numerical calculations, eight different grid sizes with $\Delta x = 1/10, 1/20, 1/40, 1/80, 1/160, 1/320, 1/640$ and 1/1280 are used, and the time steps are fixed to be $\Delta t = 0.5$ ns, corresponding to Courant number 0.25 for the solid for the finest mesh. The results from the finest mesh are used as the standard. After one and a half natural periods of the vibration, when the errors are largest, the relative ℓ_2 errors are calculated for the velocities, gas pressure, the material densities and the volume fractions using the definition $\sqrt{\sum (q_i - q_{si})^2 / \sum q_{si}^2}$, where the summations are over all the nodes in the computational domain, q_i is the quantity of concern at node i and q_{si} is the standard value of q_i calculated with the finest mesh. The results are shown in Fig. 1. The figure confirms the errors for these quantities decay with second order dependence on the spatial discretization $1/\Delta x$.

To illustrate the applicability of the new method to practical cases involving large deformations, we present another example and compare our results with experiment. In this example we simulate the penetration of a projectile into a steel block. This problem has been treated by Anderson et al. [10] using an Eulerian method Code (CTH). In contrast to their simulation that uses a single velocity field, we track three velocity fields, the velocity of projectile, the velocity of the target material and the velocity of the surrounding air. Although the effects of air are negligible in the penetration process and a multiphase treatment is not the most efficient approach to this particular problem, we intentionally choose to simulate this problem using the multiphase or multi-material formulation [8] combined with the newly introduced numerical scheme of satisfying the continuity requirement to demonstrate the capabilities of such a formulation and the numerical scheme. If we pursue this problem further, to consider the effect of air drag on the motion of fragments generated by the impact, the air motion is evidently important. If the air is replaced by water, the effect of water flow is even more important for the motion of the fragments. For these cases, the multiphase treatment of the problem is not only convenient but could also be the only choice if the fragments sizes are smaller than the grid resolution. For this reason, the purpose of our calculation is not to compare the numerical results obtained using different codes or physical models, but rather to show that the multiphase or multi-material formulation [8] combined with the new numerical scheme of satisfying the continuity requirements can be used to produce results in good agreement with the experimental data. Therefore the theoretical framework and the numerical method can potentially be used in more complicated problems, such as the fragments–fluid interactions, or problems of continuous multiphase flows, including fluid–structure interaction problems. Indeed, we have used this combination to investigate various problems related to continuous multiphase flows, multi-material

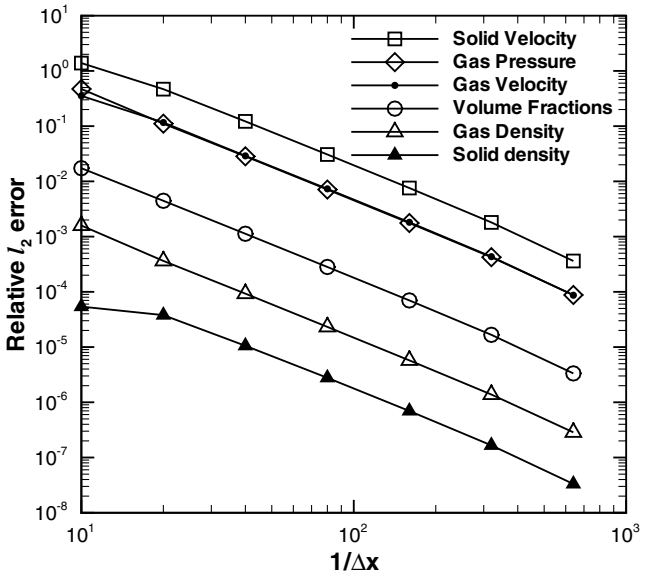

Fig. 1. Convergence of the relative ℓ_2 errors with respect to the spatial discretization Δx.

interactions and fluid–structure interactions. Because of the complexity involved, direct comparison to either theoretical result and experiment are non-trivial. Fortunately, the projectile–target interaction problem treated in this work is an exception and direct comparison to experimental results is possible for this case.

In this example of projectile–target interaction, a 5 cm tungsten rod is shot into a steel block of 4.95 cm in thickness. The steel block is initially at rest and the impact velocity of the tungsten rod is 1.7 km/s. The constitutive relation of both the tungsten and the steel are described using the Johnson–Cook model [11], in which the yield stresses of the metals are functions of the effective plastic strain and the temperature. The model parameters used in this simulation are the same as those used by Anderson et al. [10]. Fig. 2 shows a snap shot of the penetration process. Our numerical results are compared with experimental data in Fig. 3.

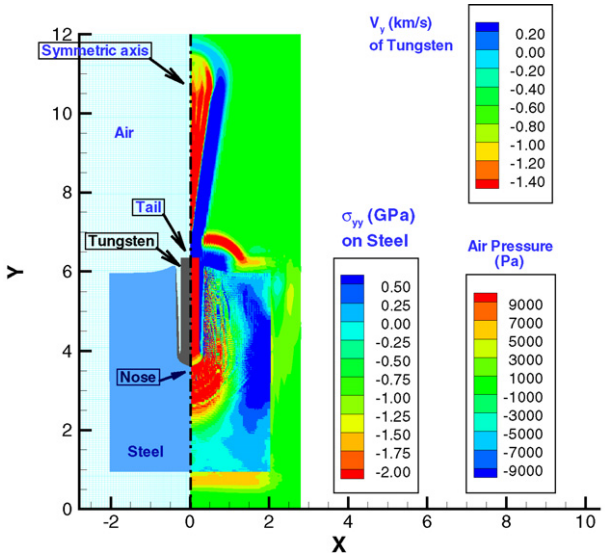

Fig. 2. A snapshot of the projectile–target interaction. The region left to the symmetry axis shows the relative locations of the air, the tungsten rod and the steel block. In region right to the symmetry axis, we plot contours of air pressure, tungsten velocity and stress σ_{yy} in the steel block in the corresponding regions.

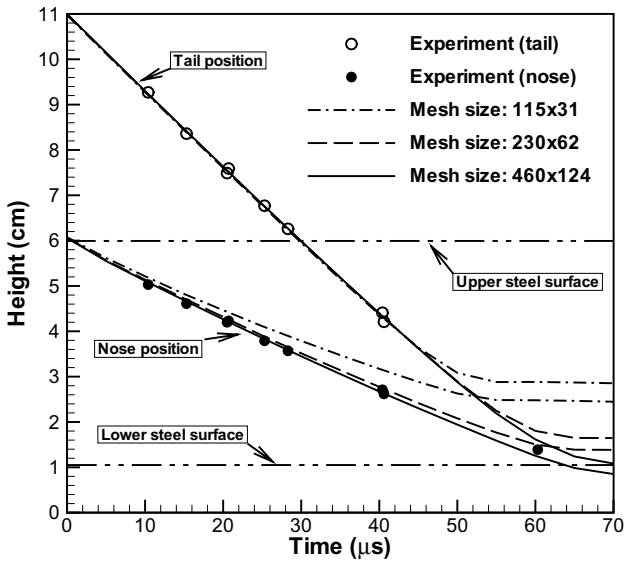

Fig. 3. Comparison with experimental results. Results calculated using different mesh sizes are plotted.

7. Conclusions

The material point method has been successfully applied in many calculations of solid material undergoing large deformations. Its direct application to physical systems described by multi-velocity fields, such as multiphase flows and fluid–structure interactions, encounters a numerical difficulty of satisfying the continuity requirement due to the spatial discretization error. Although there is an approach to circumvent this difficulty by solving an evolution equation for the pressure or the material density based on the models for volumetric deformation rate, this approach is restricted to the single pressure model and is shown to cause significant error accumulation. The error in satisfying the continuity requirement in the approach is $O(\Delta t)$; and the error in pressure is $O[(\Delta t/T)/M_k^2]$, where M_k is the Mach number in phase k, T is the characteristic time scale of the problem and Δt is the time step in the calculation. For low Mach number flows this error could be significant although it decreases with the time step.

Based on the weak solution to the evolution equations for volume fractions, in the present paper, we introduced another method of enforcing the continuity requirement within the error $O[(\Delta x/L)^d \Delta t/T]$, where Δx is the typical mesh size, L is the characteristic length scale of the problem, and d is the order of the discretization error. For a smoothly spatial varying field $d = 2$. Although this method is still first order in the time step, there is an extra factor $(\Delta x/L)^d$ in it to reduce the error, and the error in the pressure caused by different but equivalently valid methods used to approximate the volume fractions can be maintained at $O[(\Delta x/L)^d \Delta t/T/M_k^2]$. The new approach is independent of physical models and is quite similar to the method of satisfying the continuity requirement in an Eulerian method for multiphase flows; therefore it can be implemented easily without significant change to that part of an Eulerian code. Numerical examples calculated using this new approach are presented and show good agreement with theoretical and experimental results.

Acknowledgment

This work was performed under the auspices of the United States Department of Energy with support from the Army Research Office.

References

[1] F.H. Harlow, The particle-in-cell computing method for fluid dynamics, Methods Comput. Phys. 3 (1963) 219.
[2] F.H. Harlow, A.A. Amsden, Fluid Dynamics, A LASL Monograph, Los Alamos National Laboratory Report LA-4700, 1991.
[3] D. Sulsky, S.-J. Zhou, H.L. Schreyer, Application of a particle-in-cell method to solid mechanics, Comput. Phys. Commun. 87 (1995) 236.
[4] S.G. Bardenhagen, J.U. Brackbill, D. Sulsky, The material-point method for granular materials, Comput. Methods Appl. Mech. Eng. 187 (2000) 529.
[5] B.A. Kashiwa, A Multifield Model and Method for Fluid–Structure Interaction Dynamics, Los Alamos Technical Report LA-UR-01-1136, Los Alamos National Laboratory, Los Alamos, NM, 2001.
[6] X. Ma, Q. Zou, D.Z. Zhang, W.B. VanderHeyden, G.W. Wathugala, T.K. Hasselman, Application of a FILP-MPM-MFM method for simulating weapon–target interaction, in: Proceedings of the 12th International Symposium on Interaction of the Effects of Munitions with Structures, New Orleans, Louisiana, September 13–16, 2005.
[7] A. Prosperetti, G. Tryggvason, Computational Methods for Multiphase Flow, Cambridge University Press, 2006 (Chapter 10).
[8] D.Z. Zhang, W.B. VanderHeyden, Q. Zou, Nely T. Padial-Collins, Pressure calculations in disperse and continuous multiphase flows, Int. J. Multiphase Flow. 33 (2007) 86.
[9] S.J. Cummins, J.U. Brackbill, An implicit particle-in-cell method for granular materials, J. Comput. Phys. 180 (2002) 506.
[10] C.E. Anderson, V. Hohler, J.D. Walker, A.J. Stilp, Time-resolved penetration of long rods into steel targets, Int. J. Impact Eng. 16 (1) (1995) 1.
[11] G.R. Johnson, W.H. Cook, Fracture characteristic of three metals subjected to various strains, strain rates, temperature and pressure, Eng. Fract. Mech. 21 (1985) 31.