CartaBlanca Theory Manual: Multiphase Flow Equations and Numerical Methods

(LAUR-07-3621)

Duan Z. Zhang, W. Brian VanderHeyden[1], Qisu Zou, Xia Ma and Paul T. Giguere

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

December 13, 2007

[1]Current address: BP North America, 150 West Warrenville Road, Naperville, IL 60563

Abstract

CartaBlanca is a flexible software environment for prototyping physical models and simulation of a wide range of physical systems. It employs modern discretization schemes and solution methods for nonlinear physics problems on unstructured grids. CartaBlanca adopts an object-oriented, component-like design using the Java programming language. CartaBlanca is implemented with the finite-volume method and material point method (MPM). For the finite-volume method, the Arbitrary Lagrangian-Eulerian (ALE) method is used to provide flexibility with regard to physical models. Optionally, MPM can be used to effectively trace large deformation of materials while avoiding mesh tangling issues in Lagrangian methods and numerical diffusion issues in Eulerian methods. The Jacobian-Free Newton Krylov (JFNK) method is used for the solution of the nonlinear algebraic systems arising from the discretization of the governing partial differential equations. CartaBlanca has been used to simulate multiphase flows, fluid-structure interactions, heat transfer and solidification, and free surface flows.

The basic equations solved in CartaBlanca are based on multiphase flow theory. Single phase flow is treated as a special case of multiphase flow. Considerable work has been devoted to the study of disperse and continuous multiphase flows. In disperse multiphase flows, there is only one continuous phase and all other phases are in the form of particles, droplets or bubbles with sizes small compared to the macroscopic length scale of the flow. A continuous multiphase flow, in contrast, contains more than one continuous phase occupying regions or forming interconnected networks with length scales comparable to the macroscopic length

scale of the flow. Models are better developed for disperse multiphase flows. There are relatively few studies conducted of continuous multiphase flows as compared to the number of studies of disperse multiphase flows. For generality, the equations solved in CartaBlanca are based on equations obtained from an ensemble averaging technique (Zhang et al. 2006). All the phases are allowed to have their own stress (or pressure) fields evolving according to their constitutive relations (or equations of state). The microscopic density of a phase is calculated according to an evolution equation for the microscopic density, with closure terms subject to a continuity constraint. The traditionally used single pressure, or equilibrium pressure model is also available and is treated as a special case of the multipressure model. In the case that all the phases are incompressible, the multipressure model reduces to the traditional equilibrium pressure model.

Contents

Chapter 1

Introduction

CartaBlanca is an object-oriented component-based simulation and prototyping software package that enables both analysts and code developers to solve a wide range of nonlinear hydrodynamics and fluid-structure interaction problems on unstructured grids and graphs.

CartaBlanca is written entirely in Java; therefore it provides scientists and engineers with developer-friendly, modular software to use in producing large-scale computational models, and the code was designed to be readily extendible to new physical models. CartaBlanca allows users to solve a wide variety of nonlinear physics problems, including multiphase flows, interfacial flows, free surface flows, heat transfer, solidifying flows, and complex material responses involving fluid-structure interactions and solid-solid interactions. CartaBlanca makes use of the powerful, state-of-the-art Jacobian-Free Newton-Krylov (JFNK) method to solve nonlinear equations in a flexible unstructured grid finite-volume scheme. CartaBlanca couples the material point method (MPM), a latest development of the Particle-in-Cell (PIC) method, that can be used to model discrete objects, with its multiphase flow treatment to model fluid interaction with solid materials that can undergo deformation, damage, and failure. MPM method can also be used to model solid-solid interactions.

Calculations can be run in 1D, 2D, or 3D on a wide variety of unstructured grids with triangular, quadrilateral, tetrahedral, and hexahedral elements. This design allows CartaBlanca to handle complex geometrical shapes and mathematical domains. Cartesian, cylindrical, or

spherical coordinates can be used.

This document gives a detailed description of the codes physics and numerical basis, including the governing conservation equations, their closure models and discretization, available constitutive models, and the numerical solution methods.

This manual is one of three documents that comprise the main CartaBlanca documentation set. The other two are the CartaBlanca User's Manual and the CartaBlanca Programmers Manual. The User's Manual provides a comprehensive guide to the use of CartaBlanca to obtain results for the broad range of problem domains in hydrodynamics and fluid-structure interaction for which the code is applicable. The Programmers Manual describes the codes structure, computational flow, and database; it references relevant sections of the Theory Manual.

A good introduction to CartaBlancas motivation, design, and capabilities can be found at the CartaBlanca website:

http://www.lanl.gov/projects/CartaBlanca/

The code employs modern discretization schemes and solution methods for nonlinear physics problems on unstructured grids. The JFNK is used for the solution of the nonlinear algebraic systems arising from the discretization of the governing partial differential equations. CartaBlanca is implemented with a combination of a finite-volume method to calculate multiphase fluid flows and an MPM algorithm that is embedded in the multiphase flow framework that is used to enable fluid-structure interaction simulations.

For the finite-volume method, an Arbitrary Lagrangian-Eulerian (ALE) method is used to provide flexibility with regard to physical models. Optionally MPM can be used to effectively trace large deformations of materials while avoiding mesh tangling issues associated with Lagrangian methods and numerical diffusion issues in Eulerian methods.

The basic equations solved in CartaBlanca were derived from multiphase flow theory. Single phase flow is treated as a special case of multiphase flow. Considerable work has been

devoted to the study of disperse and continuous multiphase flows. In disperse multiphase flows, there is only one continuous phase; all other phases are in the form of particles, droplets or bubbles with sizes small compared to the macroscopic length scale of the flow. A continuous multiphase flow, in contrast, contains more than one continuous phase that occupy regions or form interconnected networks with length scales comparable to the macroscopic length scale of the flow. Models are better developed for disperse multiphase flows. There are relatively few studies conducted of continuous multiphase flows as compared to the number of studies of disperse multiphase flows. For generality, equations solved in CartaBlanca are based on the equations obtained from an ensemble averaging technique (Zhang et al. 2006). All phases are allowed to have their own stress (or pressure) fields evolving according to their constitutive relations (or equations of state). The microscopic densities of the phases are calculated according to evolution equations for microscopic density with closure terms subject to a continuity constraint. A traditionally used single pressure, or equilibrium pressure, model is also available and is treated as a special case of the multipressure model. In the case that all phases are incompressible, the multipressure model reduces to the traditional equilibrium pressure model.

This document is organized in a top-down fashion. First the basic governing partial differential equations are derived. Then the closure relations and the code's available constitutive models are described. Next, the discretization logic used on the governing equations is described: first for the finite volume ALE method, then, for MPM. Finally, we discussed the solution of the resulting nonlinear algebraic systems with the JFNK method.

Chapter 2 derives the basic forms of the multiphase flow equations the code solves, based on an ensemble phase average method. Derivations are given for an averaged transport equation for a generic quantity, and equations for mass and volume transport, momentum transport, energy transport, and species (components of a given phase) and scalar transport.

Chapter 3 describes the code's phase interaction models. After derivation of a continuity

constraint, CartaBlanca's closure of the velocity gradient tensor is developed. Both an equilibrium pressure model and a multipressure model are derived. Issues related to the numerical implementation on Eulerian meshes are then discussed. The next major sections develop the code's auxiliary stress and the interfacial force logic, and auxiliary heat flux and interfacial energy flux logic, respectively. Finally, the code's current treatment of phase change is described, and the need for further work on phase change modeling is indicated.

Chapter 4 describes the available constitutive models. Currently these comprise models for materials that are described as Rigid Body, Incompressible, Linear, Noble Abel gas, Mie-Gruneisen, Maxwell, Kelvin-Voigt, Johnson-Cook, or Tepla (tension plasticity). A CartaBlanca phase can consist of more than one species, and a constitutive model is chosen for a given species.

Chapter 5 develops the code's Arbitrary Lagrangian-Eulerian (ALE) Method that is used to for fluid flow, which is based on finite-volume methodology.

Chapter 6 describes CartaBlanca's implementation of MPM, which is the latest development of PIC method.

Efficient and accurate solution of the resulting systems of nonlinear equations is done in CartaBlanca with a JFNK method. Chapter 7 starts with a description of the JFNK method. The following sections in Chapter 7 give an extensive discussion of the code's preconditioning logic.

Appendix A gives a discussion on the relationship between probability theory and the averaged equations derived in Chapter 2.

Chapter 2

Basic forms of multiphase flow equations

Averaged equations for multiphase flows can be derived using a number of approaches. The averaged equations implemented in CartaBlanca are obtained using the ensemble phase average approach (Zhang et al. 2006).

We consider an ensemble of flows and denote a flow belonging to the ensemble as $\mathcal{F}$. Let $C_i(\boldsymbol{x}, t, \mathcal{F})$ be the indicator function of phase i, such that $C_i(\boldsymbol{x}, t, \mathcal{F}) = 1$ if the spatial point $\boldsymbol{x}$ is occupied by phase i in flow $\mathcal{F}$ at time t, and $C_i(\boldsymbol{x}, t, \mathcal{F}) = 0$, otherwise. The ensemble phase average $< q_i >$ for a quantity q_i pertaining to phase i is defined as

$$\theta_i(\boldsymbol{x}, t) < q_i > (\boldsymbol{x}, t) = \int C_i(\boldsymbol{x}, t, \mathcal{F}) q_i(\boldsymbol{x}, t, \mathcal{F}) d\mathcal{P}, \tag{2.1}$$

where the volume fraction θ_i of phase i at this point at time t is defined by setting $q_i =< q_i >= 1$ in (2.1)

$$\theta_i(\boldsymbol{x}, t) = \int C_i(\boldsymbol{x}, t, \mathcal{F}) d\mathcal{P}, \tag{2.2}$$

and integral $\int(\cdot)d\mathcal{P}$ denotes the average over all possible flows in the ensemble. Further description of the ensemble average and the associated probability can be found in Appendix A and in a recent paper (Zhang et al. 2006).

2.1 Averaged transport equation

For a generic quantity q_i pertaining to phase i, the averaged transport equation for its average $< q_i >$ can be written as (Zhang et al. 2006, Zhang and Prosperetti 1994, 1997)

$$\frac{\partial}{\partial t}(\theta_i < q_i >) + \nabla \cdot (\theta_i < \boldsymbol{u}_i q_i >) = \theta_i < \frac{\partial q_i}{\partial t} + \nabla \cdot (\boldsymbol{u}_i q_i) > + \int \dot{C}_i q_i d\mathcal{P}, \tag{2.3}$$

where $\boldsymbol{u}_i(\boldsymbol{x}, t)$ is the velocity of phase i and

$$\dot{C}_i = \frac{\partial C_i}{\partial t} + \boldsymbol{u}_i \cdot \nabla C_i. \tag{2.4}$$

The last term in (2.3) represents a source or a sink to quantity q_i due to phase change in the flows in the ensemble.

2.2 Mass and volume transport

The averaged mass conservation equation can be obtained from the averaged transport equation (2.3) by setting $q_i = \rho_i^0$, where ρ_i^0 is the material density, or the microscopic density, of phase i. After using the mass conservation equation

$$\frac{\partial \rho_i^0}{\partial t} + \nabla \cdot (\boldsymbol{u}_i \rho_i^0) = 0, \tag{2.5}$$

for the microscopic density ρ_i^0, we have

$$\frac{\partial}{\partial t}(\theta_i < \rho_i^0 >) + \nabla \cdot (\theta_i \widetilde{\boldsymbol{u}}_i < \rho_i^0 >) = \int \rho_i^0 \dot{C}_i d\mathcal{P}, \tag{2.6}$$

where $\widetilde{\boldsymbol{u}}_i$ is the Favre averaged velocity defined as $\widetilde{\boldsymbol{u}}_i < \rho_i^0 >=< \boldsymbol{u}_i \rho_i^0 >$.

By setting $q_i = 1$ in (2.3) one finds the evolution equation for the volume fraction

$$\frac{\partial \theta_i}{\partial t} + \widetilde{\boldsymbol{u}}_i \cdot \nabla \theta_i = \int \dot{C}_i d\mathcal{P} - \int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \cdot \nabla C_i d\mathcal{P}. \tag{2.7}$$

Multiplying (2.7) by $< \rho_i^0 >$ and then subtracting the resulting equation from (2.6) we have the averaged evolution equation for the average of the microscopic density.

$$\begin{aligned} \theta_i \left[\frac{\partial < \rho_i^0 >}{\partial t} + \nabla \cdot (\widetilde{\boldsymbol{u}}_i < \rho_i^0 >) \right] &= \int (\rho_i^0 - < \rho_i^0 >) \dot{C}_i d\mathcal{P} \\ &+ < \rho_i^0 > \int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \cdot \nabla C_i d\mathcal{P}. \end{aligned} \tag{2.8}$$

In many cases, for instance with boiling or chemical reactions, models for phase changes are available for $\dot{C}_i$. Models for the interface integral $\int(\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i)\cdot\nabla C_i d\mathcal{P}$ need to be specified for a given physical problem. Constraints for these models are discussed in the following Chapter.

2.3 Momentum transport

The microscopic momentum equation for the material of phase i can be written as

$$\frac{\partial}{\partial t}(\rho_i^0 \boldsymbol{u}_i) + \nabla\cdot(\rho_i^0 \boldsymbol{u}_i\boldsymbol{u}_i) = \nabla\cdot\boldsymbol{\sigma}_i + \rho_i^0\boldsymbol{b}, \tag{2.9}$$

where $\boldsymbol{\sigma}_i$ is the stress tensor of the phase i material and $\boldsymbol{b}$ is the body force per unit mass. By setting $q_i = \rho_i^0\boldsymbol{u}_i$ in the transport equation (2.3) and the using (2.9) one finds the averaged momentum equation for phase i,

$$\begin{aligned}&\frac{\partial}{\partial t}(\theta_i < \rho_i^0 > \widetilde{\boldsymbol{u}}_i) + \nabla\cdot(\theta_i < \rho_i^0 > \widetilde{\boldsymbol{u}}_i\widetilde{\boldsymbol{u}}_i)\\ &= \theta_i < \nabla\cdot\boldsymbol{\sigma}_i > + \nabla\cdot(\theta_i\boldsymbol{\sigma}_i^{Re}) + \int \dot{C}_i\rho_i^0\boldsymbol{u}_i d\mathcal{P} + \theta_i < \rho_i^0 > \widetilde{\boldsymbol{b}},\end{aligned} \tag{2.10}$$

where $\widetilde{\boldsymbol{b}} = < \rho_i^0\boldsymbol{b} > / < \rho_i^0 >$ is the Favre averaged body force, and

$$\boldsymbol{\sigma}_i^{Re} = - < \rho_i^0(\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i)(\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) > \tag{2.11}$$

is the Reynolds stress resulting from velocity fluctuations. To further study the averaged momentum equation we introduce an auxiliary macroscopic stress field $\boldsymbol{\sigma}_{Ai}(\boldsymbol{x}, t)$ defined for phase i. In different fields related to multiphase flows the choice of this auxiliary stress is different as we will discuss later. For any such stress the first term on the right hand side of (2.10) can be written as

$$\theta_i < \nabla\cdot\boldsymbol{\sigma}_i > = \theta_i\nabla\cdot\boldsymbol{\sigma}_{Ai} + \nabla\cdot[\theta_i(< \boldsymbol{\sigma}_i > - \boldsymbol{\sigma}_{Ai})] + \boldsymbol{f}_i, \tag{2.12}$$

where

$$\boldsymbol{f}_i = -\int(\boldsymbol{\sigma}_i - \boldsymbol{\sigma}_{Ai})\cdot\nabla C_i d\mathcal{P} \tag{2.13}$$

is the interfacial force.

Substituting (2.12) into the momentum equation (2.10) one finds

$$\frac{\partial}{\partial t}(\theta_i < \rho_i^0 > \widetilde{\boldsymbol{u}}_i) + \nabla \cdot (\theta_i < \rho_i^0 > \widetilde{\boldsymbol{u}}_i \widetilde{\boldsymbol{u}}_i) = \theta_i \nabla \cdot \boldsymbol{\sigma}_{Ai} + \nabla \cdot [\theta_i (< \boldsymbol{\sigma}_i > - \boldsymbol{\sigma}_{Ai})]$$
$$+ \quad \nabla \cdot (\theta_i \boldsymbol{\sigma}_i^{Re}) + \int \dot{C}_i \rho_i^0 \boldsymbol{u}_i d\mathcal{P} + \boldsymbol{f}_i + \theta_i < \rho_i^0 > \widetilde{\boldsymbol{b}}. \tag{2.14}$$

The interfacial force $\boldsymbol{f}_i$ defined in (2.13) depends on the choice of the macroscopic stress field $\boldsymbol{\sigma}_{Ai}$. The sum of the interfacial forces is

$$\sum_{i=1}^{M} \boldsymbol{f}_i = -\int \sum_{i=1}^{M} \boldsymbol{\sigma}_i \cdot \nabla C_i d\mathcal{P} + \sum_{i=1}^{M} \boldsymbol{\sigma}_{Ai} \cdot \nabla \theta_i, \tag{2.15}$$

where M is the number of phases in the system. The first term in (2.15) represents the effects of normal stress jumps, such as the surface tension, and is independent of the choice of the stress $\boldsymbol{\sigma}_{Ai}$. In the Rayleigh-Taylor mixing problem studied by Glimm *et al.* (1999), and Saltz *et al.* (2000), the stress $\boldsymbol{\sigma}_{Ai}$ is simply zero. In studies of two-phase flow in porous media (Bentsen, 2003), the stress is chosen to be the average stress of the phases, $\boldsymbol{\sigma}_{Ai} =< \boldsymbol{\sigma}_i >$. All such choices are allowed provided that the interfacial forces defined in (2.13) are modeled accordingly, although some choices may facilitate or complicate closure development for a given practical problem. For instance, for a particle suspension under gravity, one can choose $\boldsymbol{\sigma}_{Ai}$ to be zero for the particle phase, as long as the model for the interfacial force $\boldsymbol{f}_i$ includes effects of buoyancy.

A typical choice for the stress $\boldsymbol{\sigma}_{Ai}$ in disperse multiphase flows is $\boldsymbol{\sigma}_{Ai} =< \boldsymbol{\sigma}_c >$, $(i = 1, \cdots, M)$ where $< \boldsymbol{\sigma}_c >$ is the average stress for the continuous phase. With this choice, under the assumption that the particle (or droplet or bubble) size is small compared to the macroscopic length scale, the averaged momentum equation (2.14) can be written in the form derived by Zhang and Prosperetti (1994, 1997). Studying one-dimensional Rayleigh-Taylor mixing, Glimm *et al.* (1999), and Saltz *et al.* (2000) introduced a two-pressure model in which $\boldsymbol{\sigma}_{Ai} = \mathbf{0}$ and the interfacial terms are modeled as proportional to $\nabla \theta_i$. For instance, $\boldsymbol{f}_i$ is modeled as $p_i^* \nabla \theta_i$, where p_i^* is the pressure for phase i averaged on the interface. The gradient in the volume fraction provides a natural length scale for the interfacial force. Apparently

these models are specifically devised for Rayleigh-Taylor mixing problems, where the length scale in the problem is dominated by the length scale represented by the inverse of the volume fraction gradient, and cannot be extended easily to more general cases since the interfacial force is not necessarily zero when the gradient of the volume fraction vanishes.

For continuous multiphase flows, the concepts, such as drag and added mass forces, need to be reconsidered, if they can be meaningfully defined. Their relations to the interfacial force $\boldsymbol{f}_i$ also need to be reexamined. Furthermore, for continuous multiphase flows, one has to explicitly consider the stress difference or average stresses of the all phases. Models for such interfacial interactions implemented in CartaBlanca are discussed in the next chapter.

2.4 Energy transport

Similar to the derivation of the momentum equations, to derive the averaged equation for the internal energy e_i, let $q_i = e_i$ in transport equation (2.3) and then use the microscopic transport equations for the internal energy,

$$\frac{\partial}{\partial t}(\rho_i^0 e_i) + \nabla \cdot (\mathbf{u}_i \rho_i^0 e_i) = \boldsymbol{\tau}_i : \dot{\boldsymbol{\varepsilon}}_i + \nabla \cdot \boldsymbol{q}_i - p_i \nabla \cdot \mathbf{u}_i + s_{ei}, \tag{2.16}$$

to find

$$\begin{aligned} & \frac{\partial}{\partial t}(\theta_i < \rho_i^0 > \widetilde{e}_i) + \nabla \cdot (\theta_i \mathbf{u}_i < \rho_i^0 > \widetilde{e}_i) + \nabla \cdot (\theta_i < \rho_i^0 \mathbf{u}_i' e_i' >) \\ = \;& \theta_i (< \boldsymbol{\tau}_i : \dot{\boldsymbol{\varepsilon}}_i > + < \nabla \cdot \boldsymbol{q}_i > - < p_i \nabla \cdot \mathbf{u}_i > + < s_{ei} >) + \int \dot{C}_i \rho_i^0 e_i dP, \end{aligned} \tag{2.17}$$

where $\boldsymbol{\tau}_i$ is the deviatoric stress, $\dot{\boldsymbol{\varepsilon}}_i$ is the strain rate, $\boldsymbol{q}_i$ is the heat flux, s_{ei} is the heat source for phase i, $\widetilde{e}_i$ is the Favre averaged internal energy defined by $< \rho_i^0 > \widetilde{e}_i =< \rho_i^0 e_i >$ for phase i, and $\mathbf{u}_i'$ and e_i' are the fluctuation components of the velocity and internal energy defined as $\mathbf{u}_i' = \mathbf{u}_i - \widetilde{\mathbf{u}}_i$ and $e_i' = e_i - \widetilde{e}_i$. By adding $\int \left[\frac{\partial p_i}{\partial t} + \nabla \cdot (\boldsymbol{u}_i p_i) \right] d\mathcal{P}$ to both sides of (2.17) and using the definition $h_i = e_i + p_i/\rho_i^0$ for enthalpy, we find the averaged enthalpy equation

$$\begin{aligned} & \frac{\partial}{\partial t}(\rho_i \widetilde{h}_i) + \nabla \cdot (\mathbf{u}_i \rho_i \widetilde{h}_i) + \nabla \cdot (\theta_i < \mathbf{u}_i' \rho_i^0 h_i' >) \\ = \;& \theta_i (< \boldsymbol{\tau}_i : \dot{\boldsymbol{\varepsilon}}_i > + < \nabla \cdot \boldsymbol{q}_i > + < \dot{p}_i > + < s_{ei} >) + \int \dot{C}_i \rho_i^0 h_i dP, \end{aligned} \tag{2.18}$$

where $\widetilde{h}_i$ is the Favre averaged enthalpy defined by $< \rho_i^0 > \widetilde{h}_i =< \rho_i^0 h_i >$, $h_i' = h_i - \widetilde{h}_i$ is the fluctuation component of the enthalpy, and

$$\dot{p}_i = \frac{\partial p_i}{\partial t} + \mathbf{u}_i \cdot \nabla p_i \tag{2.19}$$

is the total derivative of the pressure.

Similar to (2.12), for the heat flux we can also introduce an auxiliary heat flux $\boldsymbol{q}_{iA}$ and write

$$\theta_i < \nabla \cdot \boldsymbol{q}_i >= \theta_i \nabla \cdot \boldsymbol{q}_{Ai} + \nabla \cdot [\theta_i (< \boldsymbol{q}_i > -\boldsymbol{q}_{Ai})] + Q_i, \tag{2.20}$$

where

$$Q_i = -\int (\boldsymbol{q}_i - \boldsymbol{q}_{Ai}) \cdot \nabla C_i d\mathcal{P}. \tag{2.21}$$

In this way the quantity Q_i is related to the heat fluxed across the interface of the phases. Clearly closures are needed for this term and many other terms in equations (2.17) and (2.18). Their closures depend on the physical processes to be simulated. Closures implemented in CartaBlanca will be discussed in Chapter 3.

2.5 Species and scalar transport

There are phases that contain different species. For a physical process in which the dynamics of the relative motion between species is not important, such as diffusion of the species within the phase, the species treatments in CartaBlanca can be used. In CartaBlanca, a phase is allowed to contain several species. Often, in such problems individual species velocity can be easily related to the phase velocity, by adding a diffusion velocity for instance. For this reason, species in CartaBlanca do not have individual velocities. Only the velocity of the phase is calculated.

The volume fraction of species j of phase i is defined similarly to the volume fraction of the phase in (2.2) as

$$\theta_{ij} = \int C_{ij} d\mathcal{P}, \tag{2.22}$$

where C_{ij} is the indicator function of for species j of phase i. That is $C_{ij}(\boldsymbol{x}, t, \mathcal{F}) = 1$, if the point $\boldsymbol{x}$ is occupied by species j of phase i at time t for flow $\mathcal{F}$ in the ensemble, and $C_{ij}(\boldsymbol{x}, t, \mathcal{F}) = 0$ otherwise. We note that

$$\sum_{j=1}^{M_i} C_{ij} = C_i, \tag{2.23}$$

where M_i is the number of species contained in phase i. Using this relation we find

$$\theta_i = \sum_{j=1}^{M_i} \theta_{ij}. \tag{2.24}$$

The average species microscopic density is defined as

$$< \rho_{ij}^0 >= \frac{1}{\theta_{ij}} \int \rho_{ij}^0 C_{ij} d\mathcal{P}. \tag{2.25}$$

For a phase containing M_i species, the microscopic density of the phase can be calculated as

$$\theta_i < \rho_i^0 >= \sum_{j=1}^{M_i} \int \rho_{ij}^0 C_{ij} d\mathcal{P}. \tag{2.26}$$

The mass fraction β_{ij} for species j of phase i is defined as

$$\beta_{ij} = \frac{\theta_{ij} < \rho_{ij}^0 >}{\theta_i < \rho_i^0 >}, \tag{2.27}$$

Using (2.26) and (2.27) we have

$$\sum_{j=1}^{M_i} \beta_{ij} = 1. \tag{2.28}$$

In CartaBlanca, for species calculations, only species mass fractions are stored; microscopic density and volume fraction are not stored.

For a scalar $\phi_{ij}(\boldsymbol{x}, t, \mathcal{F})$ pertaining to species j and phase i, its average is defined as

$$\widetilde{\phi}_{ij}(\boldsymbol{x}, t) =< \rho_i^0 C_{ij} \phi_{ij} > / < \rho_i^0 C_{ij} >=< \rho_i^0 C_{ij} \phi_{ij} > /(\beta_{ij} < \rho_i^0 >), \tag{2.29}$$

By setting $q_i = \rho_i^0 C_{ij} \phi_{ij}$ in (2.3), one finds the transport equation

$$\begin{aligned} & \frac{\partial}{\partial t}(\theta_i < \rho_i^0 C_{ij} \phi_{ij} >) + \nabla \cdot (\theta_i < \boldsymbol{u}_i \rho_i^0 C_{ij} \phi_{ij} >) \\ = \quad & \theta_i < \frac{\partial}{\partial t}(\rho_i^0 C_{ij} \phi_{ij}) + \nabla \cdot (\boldsymbol{u}_i \rho_i^0 C_{ij} \phi_{ij}) > + \int \dot{C}_i \rho_i^0 C_{ij} \phi_{ij} d\mathcal{P}. \end{aligned} \tag{2.30}$$

Decomposing $\boldsymbol{u}_i$ as $\boldsymbol{u}_i = \widetilde{\boldsymbol{u}}_i + \boldsymbol{u}_i'$, and $C_{ij}\phi_{ij} = \beta_{ij}\widetilde{\phi}_{ij} + (C_{ij}\phi_{ij})'$, where $< \rho_i^0 \boldsymbol{u}_i' >= \boldsymbol{0}$ and $< \rho_i^0 (C_{ij}\phi_{ij})' >= 0$, we can write (2.30) as

$$\begin{aligned} & \frac{\partial}{\partial t}(\theta_i < \rho_i^0 > \beta_{ij}\widetilde{\phi}_{ij}) + \nabla \cdot (\theta_i < \rho_i^0 > \widetilde{\boldsymbol{u}}_i \beta_{ij}\widetilde{\phi}_{ij}) \\ = \;& \theta_i < \rho_i^0 \dot{C}_{ij}\phi_{ij} > + \theta_i < \rho_i^0 > \beta_{ij}\widetilde{\dot{\phi}}_{ij} + \int \dot{C}_i \rho_i^0 C_{ij}\phi_{ij} d\mathcal{P} \\ - \;& \nabla \cdot (\theta_i < \rho_i^0 (C_{ij}\phi_{ij})' \boldsymbol{u}_i' >), \end{aligned} \tag{2.31}$$

after using the microscopic continuity equation $\partial \rho_i^0 / \partial t + \nabla \cdot (\boldsymbol{u}_i \rho_i^0) = 0$ for the phase. The first term in the right hand side is the source term due to species change while the third term is the source due to the phase change. The second term is the average rate of change of ϕ_{ij}, and the last term represents the correlation between the velocity fluctuations and the fluctuations of $(C_{ij}\phi_{ij})$.

Using the mass conservation equation (2.6) we can write (2.31) in the Lagrangian form

$$\begin{aligned} & \theta_i < \rho_i^0 > \left[\frac{\partial}{\partial t}(\beta_{ij}\widetilde{\phi}_{ij}) + \widetilde{\boldsymbol{u}}_i \cdot \nabla (\beta_{ij}\widetilde{\phi}_{ij}) \right] \\ = \;& \theta_i < \rho_i^0 \dot{C}_{ij}\phi_{ij} > + \theta_i < \rho_i^0 > \beta_{ij}\widetilde{\dot{\phi}}_{ij} + \int \dot{C}_i \rho_i^0 (C_{ij}\phi_{ij})' d\mathcal{P} \\ - \;& \nabla \cdot (\theta_i < \rho_i^0 (C_{ij}\phi_{ij})' \boldsymbol{u}_i' >). \end{aligned} \tag{2.32}$$

By setting $\phi_{ij} = \widetilde{\phi}_{ij} = 1$, in the case of no phase and species changes, equation (2.32) becomes

$$\theta_i < \rho_i^0 > \left(\frac{\partial \beta_{ij}}{\partial t} + \widetilde{\boldsymbol{u}}_i \cdot \nabla \beta_{ij} \right) = -\nabla \cdot (\theta_i < \rho_i^0 C_{ij}' \boldsymbol{u}_i' >). \tag{2.33}$$

The last term in (2.33) can be related to the relative motion between the species and the phase, for instance by diffusion.

The averaged equations derived in this chapter contain integrals on the phase interfaces. They represent the phase interactions and need to be modeled before the averaged equations can be solved. In the next chapter we discuss constraints to these models and models that are currently implemented in CartaBlanca.

Chapter 3

Phase interaction models

As discussed in Chapter 2, averaged transport equations contain integrals involving ∇C_i. These integrals not only represent phase interaction but also relate the gradient of an averaged quantity $< q_i >$ to the average of the gradient of the quantity as (Zhang et. al, 2006)

$$\theta_i < \nabla q_i >= \theta_i \nabla < q_i > - \int (q_i - < q_i >) \nabla C_i d\mathcal{P}. \tag{3.1}$$

In this chapter we discuss models for such phase integrals. We start with the model for the velocity gradient because the average stress in the momentum equation of a phase is often directly related to the average deformation gradient or average velocity gradient $< \nabla \boldsymbol{u}_i >$ of the material and the average velocity gradient is not a primary variable in the averaged equation system. Therefore to calculate stresses in multiphase flows, a closure to the interface integral in the last term of (3.1) needs to be specified for $q_i = \boldsymbol{u}_i$. That is, for the velocity, relation (3.1) becomes

$$\theta_i < \nabla \boldsymbol{u}_i >= \theta_i \nabla < \boldsymbol{u}_i > - \int (\boldsymbol{u}_i - < \boldsymbol{u}_i >) \nabla C_i d\mathcal{P}. \tag{3.2}$$

The integral is a tensor. In the following section we focus on the trace of this integral because it is related to the continuity and mass conservation of the multiphase flows. The continuity condition provides a constraint to closures of the integral. Closure models for the full tensor satisfying this continuity constraint are discussed in later sections.

3.1 Continuity constraint

Continuity of multiphase flows requires that the volume fractions of all phases sum to one. In CartaBlanca as in many computations, the volume fractions are calculated as a ratio between the macroscopic density, $\rho_i = \theta_i < \rho_i^0 >$, and the average microscopic density $< \rho_i^0 >$. The macroscopic density is calculated using (2.6) with the velocity obtained from the momentum equations of the system. The average microscopic density, $< \rho_i^0 >$, is determined by finding an average pressure $< p_i >$ according to the equation of state for phase i such that

$$\sum_{i=1}^{M} \frac{\rho_i}{< \rho_i^0 > (< p_i >, < T_i >)} = \sum_{i=1}^{M} \theta_i = 1, \tag{3.3}$$

where $< T_i >$ is the average temperature of the phase and M is the number of phases. Here we imply an assumption that there are averaged equations of state for all the phases.

Regardless of the numerical method used to solve this equation, any method that provides a way to determine the volume fraction and the microscopic density is equivalent to making closure assumptions about the interface integral in the last terms on the right hand sides of equations (2.7) and (2.8). If phase changes are present, models for the terms related to $\dot{C}_i$ should be provided. Assuming the models related to the phase changes are given, the approach of determining the pressures and the microscopic densities described above is equivalent to making a closure assumption for the interface integral $\int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \cdot \nabla C_i d\mathcal{P}$.

From (3.2) we find

$$\int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \nabla C_i d\mathcal{P} = \theta_i (\nabla \widetilde{\boldsymbol{u}}_i - < \nabla \boldsymbol{u}_i >) - \nabla (\theta_i < \rho_i^{0\prime} \boldsymbol{u}_i' > / < \rho_i^0 >), \tag{3.4}$$

after using $\widetilde{\boldsymbol{u}}_i - < \boldsymbol{u}_i > = < \rho_i^{0\prime} \boldsymbol{u}_i'' > / < \rho_i^0 >$, where $\boldsymbol{u}_i'' = \boldsymbol{u}_i - < \boldsymbol{u}_i >$ and $\rho_i^{0\prime} = \rho_i^0 - < \rho_i^0 >$ are the fluctuation components of the velocity and the microscopic material density ρ_i^0. In many multiphase flows, the last term in (3.4) can be neglected. For instance, if the density fluctuations are caused by pressure fluctuations, then using the equation of state we have $\rho_i^{0\prime} = p'/c_i^2$, where c_i is the speed of sound for phase i and p' is the pressure fluctuation. The

momentum equation for the material can be used to find p' is of order $< \rho_i^0 >< \boldsymbol{u}_i > \cdot \boldsymbol{u}_i''$, and the last term in equation (3.4) can be estimated as $O(\theta_i < \boldsymbol{u}_i > \boldsymbol{u}_i'' \cdot \boldsymbol{u}_i''/c_i^2)$. If the velocity fluctuation is small compared to the sound speed, which is true for many practical cases of multiphase flows, the last term of (3.4) can be neglected. There are cases, however, such as the Rayleigh-Benard convection, in which the correlation between the velocity fluctuation and the microscopic density fluctuation cannot be neglected. In CartaBlanca we restrict ourselves to the cases where the correlation $< \rho_i^{0\prime} \boldsymbol{u}_i'' >$ is negligible. Under this restriction equation (3.4) implies that a closure for $\int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \nabla C_i d\mathcal{P}$ provides a relation between the gradient of the Favre averaged velocity $\nabla \widetilde{\boldsymbol{u}}_i$ and the average of the velocity gradient $< \nabla \boldsymbol{u}_i >$ of the material.

Closures of the interface integral cannot be arbitrary; they are constrained by the following microscopic continuity condition,

$$\sum_{i=1}^{M} \frac{\partial C_i}{\partial t} = 0. \tag{3.5}$$

This is the continuity condition at the interfaces of the phases. Using this condition, we find the constraint for the closure after differentiating the definition (2.2) with respect to time.

$$\sum_{i=1}^{M} \int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \cdot \nabla C_i d\mathcal{P} = \sum_{i=1}^{M} \left(\int \dot{C}_i d\mathcal{P} - \widetilde{\boldsymbol{u}}_i \cdot \nabla \theta_i \right). \tag{3.6}$$

This constraint (3.6) to the closure relation is equivalent to the requirement of (3.3) because by differentiating (3.3) and then using (2.8) we have

$$\frac{\partial}{\partial t} \sum_{i=1}^{M} \frac{\rho_i}{< \rho_i^0 >} = \sum_{i=1}^{M} \left[\int \dot{C}_i d\mathcal{P} - \int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \cdot \nabla C_i d\mathcal{P} - \widetilde{\boldsymbol{u}}_i \cdot \nabla \theta_i \right]. \tag{3.7}$$

If (3.6) is satisfied, we have $\frac{\partial}{\partial t} \sum_{i=1}^{M} \theta_i = 0$. If the initial volume fractions of all the phases sum to one, their sum will be one during the system evolution. Conversely, if (3.3) is satisfied, the left hand side of (3.7) vanishes and (3.6) is satisfied.

Equation (3.3) or (3.6) provides a constraint to the closures for the interface integral, $\int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \cdot \nabla C_i d\mathcal{P}$, but does not specify a model for the integral for each individual phase.

For an incompressible phase i, $\nabla \cdot \boldsymbol{u}_i = 0$. Substituting this into (3.4) we find the closure for the interface integral,

$$\int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \cdot \nabla C_i d\mathcal{P} = \theta_i \nabla \cdot \widetilde{\boldsymbol{u}}_i, \tag{3.8}$$

if we neglect the effects of density-velocity correlation in the last term of (3.4). Using (3.8) and (2.8) we find

$$\theta_i \frac{d < \rho_i^0 >}{dt} = \theta_i \left(\frac{\partial < \rho_i^0 >}{\partial t} + \widetilde{\boldsymbol{u}}_i \cdot \nabla < \rho_i^0 > \right) = \int (\rho_i^0 - < \rho_i^0 >) \dot{C}_i d\mathcal{P}. \tag{3.9}$$

This equation shows that with the presence of phase change, $\frac{d<\rho_i^0>}{dt}$ is not necessarily zero for a microscopically incompressible phase. To understand this we can imagine that the incompressible phase consists of two species with different densities. If the phase change happens to one of the species, the average microscopic density of the phase changes.

In the case where all the phases are incompressible, the constraints (3.6) and (3.8) lead to the familiar continuity condition for the mixture

$$\nabla \cdot \sum_{i=1}^{M} \theta_i \widetilde{\boldsymbol{u}}_i = 0. \tag{3.10}$$

It is known that this constraint on the average velocities is equivalent to (3.3) for multiphase flows where all phases are incompressible.

For compressible phases, the interface integral needs to be modeled. The models for the integral are not unique. In the following sections we describe the models implemented in CartaBlanca.

3.2 Velocity gradient models

With the continuity constraint discussed in the previous section, we now develop a closure of the velocity gradient tensor. We assume it can be written in the following form

$$< \nabla \boldsymbol{u}_i > = \boldsymbol{\alpha}_i \cdot \nabla \widetilde{\boldsymbol{u}}_i + \boldsymbol{B}_i, \tag{3.11}$$

where $\boldsymbol{\alpha}_i$ is a fourth order tensor and $\boldsymbol{B}_i$ is a second order tensor. We show that the traditional equilibrium pressure model can be viewed as a special case of this form and that a multipressure model can be developed in this form.

To further simplify the model we assume that $\boldsymbol{\alpha}_i$ is isotropic. After the use of the representation theorem for isotropic tensors (Gurtin 1981), we find that the fourth order tensor $\boldsymbol{\alpha}_i$ can be represented by two parameters α_{bi} and α_{di} corresponding to the bulk part and the deviatoric part of the deformation.

$$< \nabla \boldsymbol{u}_i >= \frac{1}{3}[\alpha_{bi}(\nabla \cdot \widetilde{\boldsymbol{u}}_i) + B_i]\boldsymbol{I} + \alpha_{di}\left[\nabla \widetilde{\boldsymbol{u}}_i - \frac{1}{3}(\nabla \cdot \widetilde{\boldsymbol{u}}_i)\boldsymbol{I}\right]. \tag{3.12}$$

Under this assumption for the average velocity gradient $< \nabla \boldsymbol{u}_i >$, using (3.4), the closure for the interface integral can be written as

$$\int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \cdot \nabla C_i d\mathcal{P} = \theta_i \left[(1 - \alpha_{bi})\nabla \cdot \widetilde{\boldsymbol{u}}_i - B_i\right], \tag{3.13}$$

and the continuity constraint (3.6) is equivalent to

$$\sum_{i=1}^{M} \theta_i B_i = \sum_{i=1}^{M} \left[\nabla \cdot (\theta_i \widetilde{\boldsymbol{u}}_i) - \alpha_{bi}\theta_i \nabla \cdot \widetilde{\boldsymbol{u}}_i - \int \dot{C}_i d\mathcal{P}\right], \tag{3.14}$$

if the velocity-density correlation in the last term of (3.4) is negligible.

Using (3.13), equation (2.7) can be written as

$$\frac{\partial \theta_i}{\partial t} + \nabla \cdot (\theta_i \widetilde{\boldsymbol{u}}_i) = \alpha_{bi}\theta_i \nabla \cdot \widetilde{\boldsymbol{u}}_i + \int \dot{C}_i d\mathcal{P} + \theta_i B_i, \tag{3.15}$$

and (2.8) can be written as

$$\theta_i \frac{d < \rho_i^0 >}{dt} = -\theta_i < \rho_i^0 > [\alpha_{bi} \nabla \cdot \widetilde{\boldsymbol{u}}_i + B_i] + \int (\rho_i^0 - < \rho_i^0 >)\dot{C}_i d\mathcal{P}. \tag{3.16}$$

If phase i is incompressible, comparing (3.13) to (3.8) we find $\alpha_{bi} = 0$ and $B_i = 0$.

The continuity constraint does not impose restrictions on the parameter α_{di}. Although it can be easily changed, in CartaBlanca, currently $\alpha_{di} = 1$ is the default choice. This implies that the deviatoric component of $< \nabla \boldsymbol{u}_i >$ is assumed to be the same as the deviatoric component of $\nabla < \boldsymbol{u}_i >$.

The continuity constraint for the parameters B_i and α_{bi} does not uniquely specify them. Their specification is related to pressures in the multiphase flow system. Currently two pressure models are implemented in CartaBlanca: an equilibrium pressure model and a multi-pressure model. These models are described in the following sections. Users should choose one of them in the input specification according to physics of the problem they are solving.

3.2.1 Equilibrium pressure assumption

Under the equilibrium pressure assumption all phases have the same pressure. This assumption is often generalized such that the time derivatives of the pressures are the same for all the phases ($\partial < p_i > /\partial t = \partial p/\partial t$) to accommodate the effect of surface tension.

By differentiating (3.3) and using (2.6) we find

$$\frac{\partial p}{\partial t} = \sum_{i=1}^{M} \frac{1}{< \rho_i^0 >} \left[\int \rho_i^0 \dot{C}_i d\mathcal{P} - \nabla \cdot (\theta_i < \rho_i^0 > \widetilde{\boldsymbol{u}}_i) \right] / \sum_{i=1}^{M} \frac{\theta_i}{c_i^2 < \rho_i^0 >}, \tag{3.17}$$

where

$$c_i^2 = \frac{\partial < p_i >}{\partial < \rho_i^0 >}. \tag{3.18}$$

In principle c_i defined here is different from the speed of sound of the material. By expanding the equation of state in the vicinity of the average density and the average temperature, after averaging, one finds that the averaged equation of state differs from the original equation of state for the material by quadratic terms in density and temperature fluctuations. If these fluctuations result from velocity fluctuations, similarly to the analysis following (3.4), using the momentum and energy equations for the material, one can estimate that these quadratic terms are negligible for flows with velocity fluctuation of a small Mach number. Under this assumption, c_i as defined in (3.18) can be approximated by the speed of sound of the material.

Using (3.18) we have

$$\frac{\partial < \rho_i^0 >}{\partial t} = \frac{1/c_i^2}{\sum_{i=1}^{M} \theta_i/(c_i^2 < \rho_i^0 >)} \sum_{i=1}^{M} \frac{1}{< \rho_i^0 >} \left[\int \rho_i^0 \dot{C}_i d\mathcal{P} - \nabla \cdot (\theta_i < \rho_i^0 > \widetilde{\boldsymbol{u}}_i) \right]. \tag{3.19}$$

Substituting (3.19) into (2.8), one finds

$$\begin{aligned}
&\int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i)\cdot\nabla C_i d\mathcal{P} \\
= \;& \frac{\theta_i/(c_i^2 < \rho_i^0 >)}{\sum_{i=1}^M (\theta_i/c_i^2 < \rho_i^0 >)} \sum_{i=1}^M \frac{1}{< \rho_i^0 >}\left[\int \rho_i^0 \dot{C}_i d\mathcal{P} - \nabla\cdot(\theta_i < \rho_i^0 > \widetilde{\boldsymbol{u}}_i)\right] \\
+ \;& \frac{\theta_i \nabla\cdot(< \rho_i^0 > \widetilde{\boldsymbol{u}}_i)}{< \rho_i^0 >} - \frac{1}{< \rho_i^0 >}\int (\rho_i^0 - < \rho_i^0 >)\dot{C}_i d\mathcal{P},
\end{aligned} \tag{3.20}$$

or $\alpha_{bi} = 0$ and

$$\begin{aligned}
B_i \;=\; & \frac{1/(c_i^2 < \rho_i^0 >)}{\sum_{i=1}^M (\theta_i/c_i^2 < \rho_i^0 >)} \sum_{i=1}^M \frac{1}{< \rho_i^0 >}\left[\nabla\cdot(\theta_i < \rho_i^0 > \widetilde{\boldsymbol{u}}_i) - \int \rho_i^0 \dot{C}_i d\mathcal{P}\right] \\
- \;& \frac{\widetilde{\boldsymbol{u}}_i\cdot\nabla < \rho_i^0 >}{< \rho_i^0 >} + \frac{1}{\theta_i < \rho_i^0 >}\int (\rho_i^0 - < \rho_i^0 >)\dot{C}_i d\mathcal{P},
\end{aligned} \tag{3.21}$$

If we further assume that the pressure gradients for all the phases are the same, applying (3.18) to calculate $\nabla < \rho_i^0 >$ in $\nabla\cdot(\theta_i < \rho_i^0 > \widetilde{\boldsymbol{u}}_i)$ in the right hand side of (3.17) we can rewrite (3.17) as

$$\frac{\partial p}{\partial t} + \boldsymbol{u}_s\cdot\nabla p = \sum_{i=1}^M \left[\frac{1}{< \rho_i^0 >}\int \rho_i^0 \dot{C}_i d\mathcal{P} - \nabla\cdot(\theta_i \widetilde{\boldsymbol{u}}_i)\right] / \sum_{i=1}^M \frac{\theta_i}{c_i^2 < \rho_i^0 >}, \tag{3.22}$$

where $\boldsymbol{u}_s$ is the sonic average velocity (Kashiwa and Rauenzahn, 1994) defined as

$$\boldsymbol{u}_s = \frac{\sum_{i=1}^M \theta_i \widetilde{\boldsymbol{u}}_i/(c_i^2 < \rho_i^0 >)}{\sum_{i=1}^M \theta_i/(c_i^2 < \rho_i^0 >)}. \tag{3.23}$$

This closure for $\int(\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i)\cdot\nabla C_i d\mathcal{P}$, or equivalently equation (3.19), implies that the local microscopic density change is not directly related to the velocity field of the individual phase, but rather is related to the mixture motion. For a disperse multiphase flow, where the typical size of the particles (or droplets or bubbles) is small compared to the macroscopic length scale of the flow, it is true that the microscopic density change of the disperse phase is not directly related to its velocity field. On the other hand, for the continuous phase one expects a more direct relation between the microscopic density change and the velocity field of the phase. Clearly it is advantageous to have a more flexible relation between the microscopic density

changes and the velocity fields for both the disperse and continuous phases. The multipressure model described in the next section provides such flexibility.

In the limit of an incompressible phase absent phase change, from this equilibrium pressure model, we have

$$B_i = -\frac{\widetilde{\boldsymbol{u}}_i \cdot \nabla < \rho_i^0 >}{< \rho_i^0 >}, \tag{3.24}$$

which leads to

$$\frac{\partial < \rho_i^0 >}{\partial t} = 0, \tag{3.25}$$

instead of

$$\frac{d < \rho_i^0 >}{dt} = \frac{\partial < \rho_i^0 >}{\partial t} + \widetilde{\boldsymbol{u}}_i \cdot \nabla < \rho_i^0 >= 0. \tag{3.26}$$

Therefore the equilibrium pressure model is not suitable for incompressible phases with a variable density. This deficiency of the equilibrium pressure model can be overcome by the multipressure model we now introduce.

3.2.2 Multipressure model

To accommodate a more flexible relation between $< \nabla \cdot \boldsymbol{u}_i >$ and $\nabla \cdot \widetilde{\boldsymbol{u}}_i$ for different phases, one can choose coefficient α_{bi} according to the connectivity or morphology, material properties, and volume fraction of the phase. This will be an input parameter in CartaBlanca. Currently it is set to be one. For continuous multiphase flows, satisfactory results can often be found by simply assuming $\alpha_{bi} = \theta_i$, or $\alpha_{bi} = 1$ if the volume fraction is sufficiently large ($> 90\%$) or the phase is well connected. To determine B_i, we assume the pressure increase $\partial p_i / \partial t$ caused by B_i for all the phases is the same. One may consider $\partial p_i / \partial t$ to be due to the propagation of fast pressure waves in the system, while the underlying non-equilibrium state would need a much slower convective time scale to equilibrate. Using this assumption, (3.16) and the equation of state (3.18) we have

$$< \rho_i^0 > c_i^2 B_i = -\partial p_c / \partial t, \tag{3.27}$$

where $\partial p_c/\partial t$ is the same for all the phases with the subscript denoting the pressure increment is common for all the phases. Solving equations (3.27) and (3.14) we find

$$B_i = \frac{1/(<\rho_i^0> c_i^2)}{\sum_{i=1}^N \theta_i/(<\rho_i^0> c_i^2)} \sum_{i=1}^M \left[\nabla \cdot (\theta_i \widetilde{\boldsymbol{u}}_i) - \alpha_{bi} \theta_i \nabla \cdot \widetilde{\boldsymbol{u}}_i - \int \dot{C}_i d\mathcal{P} \right], \tag{3.28}$$

and

$$\begin{aligned} & \int (\boldsymbol{u}_i - \widetilde{\boldsymbol{u}}_i) \cdot \nabla C_i d\mathcal{P} = (1 - \alpha_{bi}) \theta_i \nabla \cdot \widetilde{\boldsymbol{u}}_i \\ - \quad & \frac{\theta_i/(<\rho_i^0> c_i^2)}{\sum_{i=1}^N \theta_i/(<\rho_i^0> c_i^2)} \sum_{i=1}^M \left[\nabla \cdot (\theta_i \widetilde{\boldsymbol{u}}_i) - \alpha_{bi} \theta_i \nabla \cdot \widetilde{\boldsymbol{u}}_i - \int \dot{C}_i d\mathcal{P} \right], \end{aligned} \tag{3.29}$$

after using (3.13).

For an incompressible phase i, in the absence of phase change, the multipressure model introduced in this subsection leads to the correct evolution equation (3.26), instead of (3.25), for the microscopic density since $\alpha_{bi} = 0$ and $B_i = 0$ in (3.16).

3.3 Numerical implementations on Eulerian meshes

With the closure for $< \nabla \cdot \boldsymbol{u}_i >$ chosen according to the equilibrium pressure model or to the multipressure model described in the last section, in this section we introduce numerical implementations for Eulerian methods. A full description of CartaBlanca's Arbitrary Lagrangian-Eulerian (ALE) numerical method is given in Chapter 5. A different implementation of the pressure models is necessary for material point methods (MPM) and is described in Chapter 6.

The volume fractions and microscopic densities for all the phases can be calculated from evolution equations (2.7) and (2.8). Equation (3.3) is redundant because both the equilibrium pressure closure (3.20) and multipressure closures (3.29) satisfy constraint (3.6), which is equivalent to (3.3) as proved in Section 3.1 provided that the initial volume fractions sum to one. In this way one can avoid solving (3.3), which is typically nonlinear and requires an implicit method. This implementation of the closures can reduce the amount of calculation

and could be significant in an explicit numerical scheme. However there is a shortcoming with this implementation and it should be used with care. Since this explicit scheme enforces $\sum_{i=1}^{M} \partial\theta_i/\partial t = 0$ at every time step, instead of $\sum_{i=1}^{M} \theta_i = 1$, error accumulation over time may result with the sum of the volume fractions noticeably deviating from one, especially in cases with large Courant numbers and a large number of time steps. To avoid this possible numerical error, we use a method described in Section 4 of Chapter 5.

Finally, we note that for cases, such as slow expansion of uniformly distributed large gas bubbles in a compressible viscous fluid in a closed container without macroscopic motion ($\widetilde{\boldsymbol{u}}_i = 0$), an additional model term,

$$\sum_{j=1}^{M} \frac{\theta_j}{\tau_{ij}} \frac{(p_j - p_i)}{\sqrt{<\rho_i^0><\rho_j^0>}c_i c_j},$$

to the trace of (3.12) is needed (with corresponding changes in (3.13) - (3.16) and (3.29)), to account for the process of pressure equilibration among the phases due to exchange of volume among them, where $\tau_{ij}(=\tau_{ji})$ is a time scale for pressure equilibration. In CartaBlanca, we currently assume that the equilibration time scales are very long compared to the dynamics and this term is therefore neglected. We include it here, however, for possible future implementation and to show how the multipressure model can include the case of pressure equilibration.

3.4 The auxiliary stress and interfacial force

Similarly to the velocity gradient discussed in the previous sections, relation (3.1) also applies to stress and its divergence. However, to accommodate many commonly used models for multiphase flows the average of the stress gradient is written slightly differently by introducing the auxiliary stress $\boldsymbol{\sigma}_A$ as in (2.12). The interface integral to be modeled is then defined in (2.13). As mentioned in Chapter 2 models for the interface force depend on the choice of $\boldsymbol{\sigma}_A$.

In CartaBlanca, our choice for $\boldsymbol{\sigma}_A$ is

$$\boldsymbol{\sigma}_A = -p_A \boldsymbol{I} + \mu_e [\nabla \boldsymbol{u}_m + (\nabla \boldsymbol{u}_m)^T], \tag{3.30}$$

where p_A is an auxiliary pressure, μ_e is the effective viscosity and $\boldsymbol{u}_m$ is the mixture velocity defined as $\boldsymbol{u}_m = \sum_{i=1}^{M} \theta_i \widetilde{\boldsymbol{u}}_i$. In CartaBlanca the pressure, p_A, is the pressure of the system if the single pressure model is used. When the multipressure model is used, p_A is the pressure of one of the phases specified by the user in the input specification. For disperse multiphase flows, which contain only one continuous phase, it is suggested that the user choose the pressure as the pressure of the continuous phase, to be consistent with commonly used interface force models.

The interfacial force $\boldsymbol{f}_i$ on phase i is assumed as additive for all the phases that phase i interacts with, $\boldsymbol{f}_i = \sum_{k=1}^{M} \boldsymbol{f}_{ik}$, where $\boldsymbol{f}_{ik}$ is the force between phases i and k, which is modeled as a summation of the drag and the added mass force between the phases.

$$\boldsymbol{f}_{ik} = \theta_i \theta_k K_{ik} \rho_{ik}^0 (\widetilde{\boldsymbol{u}}_k - \widetilde{\boldsymbol{u}}_i) + A_{ik} \rho_{ik}^0 \left(\frac{\partial \widetilde{\boldsymbol{u}}_k}{\partial t} + \widetilde{\boldsymbol{u}}_k \nabla \cdot \widetilde{\boldsymbol{u}}_k - \frac{\partial \widetilde{\boldsymbol{u}}_i}{\partial t} - \widetilde{\boldsymbol{u}}_i \nabla \cdot \widetilde{\boldsymbol{u}}_i \right), \tag{3.31}$$

where $K_{ik} = K_{ki}$ is the momentum exchange coefficient, $A_{ik} = A_{ki}$ is the added mass coefficient and ρ_{ik}^0 is the reference density specified in the user input. Currently the added mass coefficient A_{ik} is an input parameter and the momentum exchange coefficient is calculated as

$$K_{ik} = \frac{3}{4} C_d \frac{|\widetilde{\boldsymbol{u}}_k - \widetilde{\boldsymbol{u}}_i|}{d_{ik}}, \tag{3.32}$$

$$C_d = C_{d\infty} + \frac{24}{Re_{ik}} + \frac{6}{1 + \sqrt{Re_{ik}}}, \quad Re_{ik} = \frac{|\widetilde{\boldsymbol{u}}_k - \widetilde{\boldsymbol{u}}_i| d_{ik}}{\nu_{ik}}, \tag{3.33}$$

where the drag coefficient $C_{d\infty}$ for infinite Reynolds number, the length scale d_{ik} and the kinematic viscosity ν_{ik} are input parameters.

Under this phase interaction model, the effect of the phase stress enters the momentum equations only through the divergence of the stress difference appearing in the second terms on the right hand sides of (2.12) and (2.14). Models for the phase stress $< \boldsymbol{\sigma}_i >$ are discussed in Chapter 4.

3.5 The auxiliary heat flux and interfacial energy flux

Similarly to the momentum exchange, the auxiliary heat flux $\boldsymbol{q}_{Ai}$ is calculated using the mixture field information as

$$\boldsymbol{q}_{Ai} = -K_e \nabla T_m, \tag{3.34}$$

where K_e is the effective heat conductivity specified in the user input and $T_m = \sum_{i=1}^{M} \theta_i \widetilde{T}_i$ is the mixture temperature.

Similarly to momentum interactions, the interfacial energy flux is modeled as

$$Q_i = \sum_{k=1}^{M} H_{ik} \theta_i \theta_k (\widetilde{T}_k - \widetilde{T}_i), \tag{3.35}$$

with H_{ik} as an input parameter.

In many practical problems, the enthalpy can be related to temperature as

$$h_i = C_i(\widetilde{T}_i - T_i^f) + h_i^f, \tag{3.36}$$

where C_i is the heat capacity, T_i^f is the formation temperature and h_i^f is the formation enthalpy.

If the the heat capacity, the formation temperature the formation enthalpy can be treated as constants, by substituting (3.36) into (2.18) and then using (2.20), (3.34) and (3.35) we find

$$\begin{aligned} \frac{\partial}{\partial t}(\rho_i C_i \widetilde{T}_i) \quad &+ \quad \nabla \cdot (\rho_i C_i \widetilde{T}_i \widetilde{\boldsymbol{u}}_i) = -\theta_i \nabla \cdot (K_e \nabla T_m) + \sum_{k=1}^{M} H_{ik} \theta_i \theta_k (\widetilde{T}_k - \widetilde{T}_i) \\ &+ \quad \int \dot{C}_i \rho_i^0 h_i dP + \theta_i < \dot{p}_i > + \theta_i (< s_{ei} > + < \boldsymbol{\tau}_i : \dot{\boldsymbol{\varepsilon}}_i >) \\ &+ \quad \nabla \cdot [\theta_i (< \boldsymbol{q}_i > - \boldsymbol{q}_{Ai})] - \nabla \cdot (\theta_i < \mathbf{u}_i' \rho_i^0 h_i' >). \end{aligned} \tag{3.37}$$

Currently, in CartaBlanca, the effect of phase change $\int \dot{C}_i \rho_i^0 h_i dP$ are specified by related chemical reaction models, the pressure change term is approximated as

$$< \dot{p}_i > \approx \frac{\partial < p_i >}{\partial t} + \widetilde{\boldsymbol{u}}_i \cdot \nabla < p_i > . \tag{3.38}$$

Unlike momentum interactions where the effect of stress difference is accounted for, the heat flux difference $< \boldsymbol{q}_i > - \boldsymbol{q}_{Ai}$ is currently neglected together with heat generation due to internal friction $< \boldsymbol{\tau}_i : \dot{\boldsymbol{\varepsilon}}_i >$ and the fluctuation term $< \mathbf{u}_i' \rho_i^0 h_i' >$. Clearly better models are needed here.

3.6 Phase change

Multiphase flow calculation in CartaBlanca is based on the system of equations for multiphase flow described above. In this equation system all effects of phase changes are represented by terms involving $\dot{C}_i$.

Currently in CartaBlanca, for phase change between two phases, we simply assume the quantities, such as density, velocity and enthalpy for newly generated material are the same as the corresponding average values of the donor phase. Under this assumption CartaBlanca has several experimental phase change models implemented for chemical reactions related to high explosive materials, with the rate of the chemical reaction related to temperature and concentration of the involved materials. Depending on the physical problems to be calculated, models for phase change vary greatly. This part of CartaBlanca needs to be further developed. In the meantime, users are expected to provide their own model terms related to $\dot{C}_i$ to consider phase changes.

Chapter 4

Constitutive models

The constitutive relation for a phase in CartaBlanca is viewed as a composite material comprised of species in the phase. The equation of state for a phase is written in the form of (3.18). The speed of sound for the phase is a function of temperature, densities of the species in the phase and the composition of the species. Currently the Voigt assumption is used to find the speed of sound for a phase comprised of more than one species:

$$c_i^2 = \frac{1}{\theta_i} \sum_{j=1}^{M_i} \theta_{ij} c_{ij}^2 (< T_i >, < \rho_{ij}^0 >), \tag{4.1}$$

where θ_{ij} is the volume fraction, c_{ij} is the sound speed, and $< \rho_{ij}^0 >$ is the microscopic density of species j, and the sum is over the total number M_i of species contained in phase i. The species volume fractions satisfy $\sum_{j=1}^{M_i} \theta_{ij} = \theta_i$. For a single species phase or for species with the same sound speeds, using (2.24) we find the model reduces to the equation of state for those species.

In CartaBlanca, the species volume fractions and densities are not stored. To calculate the sound speed for phase i, the density for species j in (4.1) is calculated as

$$< \rho_{ij}^0 >= \rho_{ij}^0 (< T_i >, p_{ij}), \tag{4.2}$$

where $p_{ij} = \theta_{ij} p_i / \theta_i$ is the partial pressure of the species. Since θ_{ij} is not stored in CartaBlanca, we approximate θ_{ij}/θ_i by the mass fraction β_{ij} in the calculation of partial pressure. This calculated density is used to find the volume fraction θ_{ij} by using (2.27). This procedure of

calculating volume fraction for the species can be viewed as a first step in an iterative process to find the species volume fraction θ_{ij}/θ_i within the phase. Using (2.27) we have

$$\frac{\theta_{ij}}{\theta_i} = \frac{\beta_{ij} < \rho_i^0 >}{\theta_i(< \rho_{ij}^0 >_0 + \frac{\theta_{ij}}{\theta_i}\Delta p_i / c_{ij}^2)}, \tag{4.3}$$

where $< \rho_{ij}^0 >_0$ is the reference density and Δp_i is the difference between the current pressure p_i and the pressure at the reference density. If the derivative of the right hand side of (4.3) with respect to the unknown θ_{ij}/θ_i is less than one the iteration is guaranteed to converge. By differentiating the right hand side and using (2.27) again we find the derivative to be $-\Delta p_{ij}/(< \rho_{ij}^0 > c_{ij}^2)$, where $\Delta p_{ij} = (\theta_{ij}/\theta_i)\Delta p_i$ is the difference of the partial pressure. If the pressure change is less than $(< \rho_{ij}^0 > c_{ij}^2)$, the algorithm can be used to calculate the volume fraction and to approximate the phase wave speed. For more complicated problems, where the concept of partial pressure does not apply, better models are needed here.

In the following sections we describe the constitutive models currently available for species in CartaBlanca.

4.1 Rigid body

This is a special constitutive relation; it is used when there is only one species in the phase and the deformation of the phase is negligible. In this constitutive relation the microscopic density of the material is a constant and the sound speed is set to zero so that the time step will not be affected by this phase.

4.2 Incompressible

For an incompressible material the density is set to be a constant specified by user input and the square of the sound speed is set to machine infinity (10^{64}).

4.3 Linear

This is a constitutive model for a fluid with the following equation of state:

$$\rho^0 = \frac{A + Bp}{1 + C(T - D)}, \tag{4.4}$$

where A, B, C and D are model parameters specified by user input.

4.4 Noble Abel gas

This is a constitutive model for a fluid with the following equation of state:

$$\rho^0 = \frac{p}{Ap + BT}, \tag{4.5}$$

where A and B are model parameters specified by user input .

4.5 Mie-Gruneisen equation of state

Mie-Gruneisen equations are often used for condensed matter and for materials under shock and impact. In this equation of state the density, enthalpy and pressure are related as

$$p = \frac{1}{1+\gamma}\left\{p_h\left[1 - \frac{\gamma}{2}\left(\frac{\rho^0}{A} - 1\right)\right] + \gamma\rho^0(h - h_0)\right\}, \tag{4.6}$$

where

$$p_h = \begin{cases} K_1\left(\frac{\rho^0}{A} - 1\right) & \text{if } \frac{\rho^0}{A} < 1 \\ K_1\left(\frac{\rho^0}{A} - 1\right) + K_2\left(\frac{\rho^0}{A} - 1\right)^2 + K_3\left(\frac{\rho^0}{A} - 1\right)^3 & \text{Otherwise.} \end{cases} \tag{4.7}$$

and γ, A, h_0, K_1, K_2 and K_3 are model parameters specified by user input.

4.6 Maxwell model

Viscoelastic materials can be modeled in CartaBlanca with either a Maxwell or Kelvin-Voigt model. Maxwell materials can be considered as a viscous damper (dashpot) in series with an

elastic spring. The stress $\boldsymbol{\sigma}$ in this model is decomposed into pressure and a deviatoric part $\boldsymbol{s}$ as

$$\boldsymbol{\sigma} = -p\boldsymbol{I} + \boldsymbol{s}. \tag{4.8}$$

The pressure p is calculated as

$$\frac{dp}{dt} = -\frac{p}{\tau_p} + Btr(\dot{\boldsymbol{\epsilon}}) + \gamma \boldsymbol{s} : \ddot{\boldsymbol{\epsilon}}_d, \tag{4.9}$$

where τ_p is the relaxation time for pressure, B is the bulk modulus, $\dot{\boldsymbol{\epsilon}}$ is the strain rate, γ is the Gruneisen coefficient and $\ddot{\boldsymbol{\epsilon}}_d$ is the rate of change of the deviatoric strain rate.

The evolution equation for the deviatoric stress $\boldsymbol{s}$ is

$$\frac{d\boldsymbol{s}}{dt} + \boldsymbol{s} \cdot \Omega - \Omega \cdot \boldsymbol{s} = -\frac{\boldsymbol{s}}{\tau_d} + 2G\dot{\boldsymbol{\epsilon}}_d, \tag{4.10}$$

where $\Omega = \frac{1}{2}[\nabla\widetilde{\boldsymbol{u}} - (\nabla\widetilde{\boldsymbol{u}})^T]$ is the spin tensor, τ_d is the relaxation time for deviatoric stress, $\dot{\boldsymbol{\epsilon}}_d$ is the deviatoric strain rate and G is the shear modulus.

4.7 Kelvin-Voigt model

In the Kelvin-Voigt model, the stress is separated into the elastic part $\boldsymbol{\sigma}_E$ and the viscous part $\boldsymbol{\sigma}_V$.

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}_V + \boldsymbol{\sigma}_E \tag{4.11}$$

The viscous part $\boldsymbol{\sigma}_V$ is calculated as

$$\boldsymbol{\sigma}_V = \mu_b tr(\dot{\boldsymbol{\epsilon}})\boldsymbol{I} + 2\mu\dot{\boldsymbol{\epsilon}} \tag{4.12}$$

where μ_b is the bulk viscosity, μ is the shear viscosity and $\dot{\boldsymbol{\epsilon}}$ is the strain rate.

The elastic part is calculated by solving the following evolution equation

$$\frac{d\boldsymbol{\sigma}_E}{dt} + \boldsymbol{\sigma}_E \cdot \boldsymbol{\Omega} - \boldsymbol{\Omega} \cdot \boldsymbol{\sigma}_E + \frac{1}{2}tr(\dot{\boldsymbol{\epsilon}})\boldsymbol{\sigma}_E = Btr(\dot{\boldsymbol{\epsilon}})\boldsymbol{I} + 2G\dot{\boldsymbol{\epsilon}}, \tag{4.13}$$

where B is the bulk modulus, G is the shear modulus and $\dot{\boldsymbol{\epsilon}}$ is the strain rate. The last term on the left hand side is necessary to ensure the energy conservation by accounting for the effect of volume change in cases of large deformations as we shall show in Section 6.6.

4.8 Johnson-Cook model

The Johnson-Cook model adds a plasticity model to the Kelvin-Voigt model. Stress calculation for this constitutive model contains two parts, an elastic part and a plastic flow part. The elastic stress increases as in the Kelvin-Voigt model. The plastic flow part starts by calculating the yield stress σ_{eq} as

$$\sigma_{eq} = (Y_0 + B_{jc}\epsilon_p^n)[1 + C\ln(\dot{\epsilon}_p/\dot{\epsilon}_0)]\left[1 - \left(\frac{T - T_0}{T_m - T_0}\right)^m\right], \tag{4.14}$$

where C, n, m, Y_0 and B_{jc} are material parameters and $\dot{\epsilon}_0$ is the characteristic strain rate, T is the temperature, T_m is the melting temperature, T_0 is the reference temperature and ϵ_p and $\dot{\epsilon}_p$ are the effective plastic strain and the rate of the plastic strain. For many practical applications with large deformation, in CartaBlanca we approximate the effective plastic strain by an effective strain ϵ_e. The rate of the effective strain is calculated as

$$\begin{aligned}\dot{\epsilon}_e &= \sqrt{2[(\dot{\epsilon}_1 - \dot{\epsilon}_2)^2 + (\dot{\epsilon}_2 - \dot{\epsilon}_3)^2 + (\dot{\epsilon}_3 - \dot{\epsilon}_1)^2]/9} \\ &= \sqrt{2|2[tr(\dot{\boldsymbol{\epsilon}})]^2 - 6(\dot{\epsilon}_{xx}\dot{\epsilon}_{yy} + \dot{\epsilon}_{xx}\dot{\epsilon}_{zz} + \dot{\epsilon}_{yy}\dot{\epsilon}_{zz} - \dot{\epsilon}_{xy}\dot{\epsilon}_{xy} - \dot{\epsilon}_{xz}\dot{\epsilon}_{xz} - \dot{\epsilon}_{yz}\dot{\epsilon}_{yz})|/9},\end{aligned} \tag{4.15}$$

where $\dot{\epsilon}_1$, $\dot{\epsilon}_2$ and $\dot{\epsilon}_3$ are the three principal rates of strain, and the $\dot{\epsilon}$'s with double subscripts are the components of the rates of strain under the coordinate system used. The effective strain is the time integration of this effective strain rate. The effective stress is calculated similarly as

$$\begin{aligned}\sigma_e &= \sqrt{2[(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2]/9} \\ &= \sqrt{2|2[tr(\boldsymbol{\sigma})]^2 - 6(\sigma_{xx}\sigma_{yy} + \sigma_{xx}\sigma_{zz} + \sigma_{yy}\sigma_{zz} - \sigma_{xy}\sigma_{xy} - \sigma_{xz}\sigma_{xz} - \sigma_{yz}\sigma_{yz})|/9},\end{aligned} \tag{4.16}$$

where σ_1, σ_2, and σ_3 are the three principal stress rates, and the σ's with double subscripts are the components of the stress under the coordinate system used. If the effective stress σ_e is greater than the yield stress calculated from (4.14) then each deviatoric component of the stress is reduced by a factor σ_{eq}/σ_e to make the effective stress equal to the yield stress σ_{eq}. The pressure, or the isotropic component of the stress, is kept unaltered in this step.

4.9 Tepla - tension plasticity model

4.10 Sesame table

Chapter 5

Arbitrary Lagrangian-Eulerian method

CartaBlanca is based on the finite volume method for solution of its governing conservation equations, using the integral formulation of the conservation equations. CartaBlanca takes advantage of the flexibility of Arbitrary Lagrangian-Eulerian methods (ALE) in implementing physical models. In an ALE method, physics represented by the terms in the right hand side of the transport equations described in the previous chapters can be calculated separately from the convection terms on the left hand side. This facilitates implementation of new physical models into a numerical code. This advantage is further enhanced in CartaBlanca by employing object-oriented coding and the use of the Java language. The disadvantage of the ALE method is the time splitting error introduced by the separation of the calculations of the right hand side of the equations and the advection terms on the left hand side of an equation. A numerical scheme with a second order accuracy in time has yet to be developed for the ALE method.

5.1 Control volume and conservation laws

Let $V_a(t)$ be a volume enclosed by a surface $S_a(t)$ moving with an arbitrary velocity $\boldsymbol{u}_a(\boldsymbol{x},t)$, and $q(\boldsymbol{x},t)$ be a continuous field quantity. Using the transport theorem (Liu, 2002), we have

$$\frac{d}{dt}\int_{V_a(t)} q(\mathbf{x},t)dV = \int_{V_a(t)} \frac{\partial}{\partial t} q(\mathbf{x},t)dV + \int_{S_a(t)} q\mathbf{u}_a \cdot \mathbf{n}dS, \tag{5.1}$$

where $\boldsymbol{n}$ is an outward normal vector on the surface $S_a(t)$.

For a quantity q, such as mass density, momentum or energy, satisfying the following transport equation

$$\frac{\partial q}{\partial t} + \nabla \cdot (\mathbf{u}q) = f_q, \tag{5.2}$$

where f_q is the source term for quantity q, equation (5.1) can be written as

$$\frac{d}{dt}\int_{V_a(t)} q(\boldsymbol{x},t)dV + \int_{S_a(t)} q(\boldsymbol{u}-\boldsymbol{u}_a)\cdot \boldsymbol{n}dS = \int_{V(t)} f_q dV. \tag{5.3}$$

To understand the physical meaning of the equations above let q be the density ρ of the material. Then $f_q = 0$ for cases without a mass source and equation (5.3) becomes

$$\frac{d}{dt}\int_{V_a(t)} \rho(\mathbf{x},t)dV + \int_{S_a(t)} \rho(\mathbf{u}-\mathbf{u}_a)\cdot \mathbf{n}dS = 0. \tag{5.4}$$

For a Lagrangian control volume, $\mathbf{u}_a = \mathbf{u}$, this equation takes the familiar form

$$\frac{d}{dt}\int_{V_a(t)} \rho(\mathbf{x},t)dV = 0, \tag{5.5}$$

and for an Eulerian control volume, $\mathbf{u}_a = 0$, equation (5.4) becomes

$$\frac{d}{dt}\int_{V_a(t)} \rho(\mathbf{x},t)dV + \int_{S_a(t)} \rho\mathbf{u}\cdot \mathbf{n}dS = 0. \tag{5.6}$$

The motion of volume V_a in equation (5.3) is independent of the motion of the material inside the volume. One can prescribe any motion to the volume without affecting the physics we study. We now introduce an arbitrary moving volume that is $V(t_1)$ at time t_1, such as a computational cell in a numerical calculation. This volume changes to $V(t_2)$ at time t_2. We

suppose the volume moves and deforms with the material inside the volume, but "jumps" to its final position $V(t_2)$ at the end of the time interval. That is, the velocity $\mathbf{u}_a$ takes the following form

$$\mathbf{u}_a = \mathbf{u} + \mathbf{d}\delta(t - t_2^-), \tag{5.7}$$

where $\boldsymbol{u}$ is the velocity of the material, t_2^- denotes that the δ-function happens right before the end of the time interval and $\mathbf{d}$ is the displacement necessary to jump to a prescribed position at the end of the time interval. After integrating (5.3) over the time duration, we have

$$\begin{aligned}\int_{V(t_2)} q(\mathbf{x}, t_2)dV - \int_{V(t_1)} q(\mathbf{x}, t_1)dV = \int_{t_1}^{t_2} \left(\int_{V_a(t)} f_q dV \right) dt \\ + \int_{t_2^-}^{t_2} \int_{S_a(t)} q(\mathbf{x}, t)\delta(t - t_2^-)\, \mathbf{d} \cdot \mathbf{n}\, dSdt.\end{aligned} \tag{5.8}$$

The last term in (5.8) can be evaluated as

$$\int_{t_2^-}^{t_2} \int_{S_a(t)} q(\mathbf{x}, t)\delta(t - t_2^-)\, \mathbf{d} \cdot \mathbf{n}\, dSdt = \int_{\Delta V} q(\mathbf{x}, t_2)dV, \tag{5.9}$$

where $\Delta V = V(t_2) - V(t_2^-)$ is the volume swiped through by the surface bounding the volume during the jump at the end of the time interval. We divide the volume ΔV into inflow volume V_i and outflow volume V_o relative to the final volume at the end of the jump, and then use (5.9) to write

$$\begin{aligned}\int_{V(t_2)} q(\mathbf{x}, t_2)dV - \int_{V(t_1)} q(\mathbf{x}, t_1)dV = \int_{t_1}^{t_2} \left(\int_{V_a(t)} f_q dV \right) dt \\ + \int_{V_i} q(\mathbf{x}, t_2)dV - \int_{V_o} q(\mathbf{x}, t_2)dV.\end{aligned} \tag{5.10}$$

If the volume is fixed during the time interval, the inflow and outflow volumes coincide with the material volume fluxed in and out of the volume.

Equation (5.10) implies that the total change of a quantity with its density represented by q inside the volume V_a during the time interval can be calculated in two steps, the Lagrangian step and the remapping step. The Lagrangian step, corresponding to the first term on the

right hand side of (5.10), can be performed by following the motion of the material without regard to the the motion of the control volume because the volume integral in this term is a bound value and its time integral from t_2^- to t_2 vanishes. The remapping step accounts for the "jump" of the control volume at the end of the motion. During the jump the control volume gains volume V_i and loses volume V_o, therefore gaining and losing the quantity contained in these volumes. This jump happens at the end of the time interval, therefore the quantity q contained in the inflow and outflow volumes is evaluated at time t_2^-, which is the same as the value of q evaluated at time t_2 if we assume q is continuous in time.

For a finite volume method, the control volume in (5.10) is taken to be the control volume defined by a computational cell. The ALE scheme is built on the basis of this relation.

5.2 Numerical discretization

In a numerical calculation, often the computational domain is divided into many sub-domains, called cells. The quantities of the continuous fields are approximated by their values at either cell centers or at the nodes of the computational mesh. CartaBlanca uses a node based scheme with an unstructured grid. The control volumes used in CartaBlanca are median mesh control volumes. A control volume surrounding node i is the union of sub-volumes from elements surrounding node i. Each sub-volume from an element containing node i is bounded by the element boundaries containing node i and the mid planes passing the centroid of the elements and the middle point of the edges containing node i. Figure 5.1 illustrates the construction of such a control volume in two dimensions. The shaded area surrounding node i is the control volume for the node.

A node quantity q_i defined at node i is the averaged value of q over the control volume, calculated as

$$q_i = \frac{1}{V}\int_V q dV, \tag{5.11}$$

where V is the control volume.

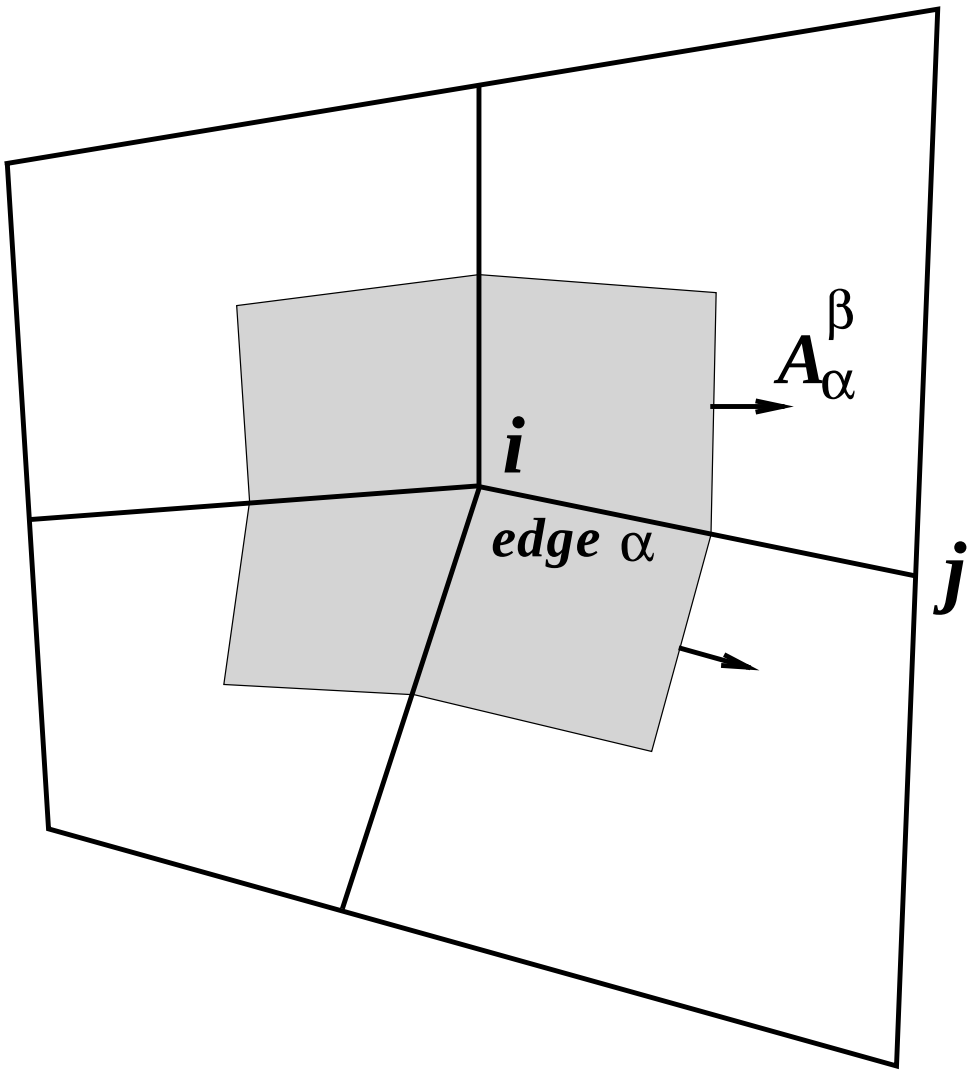

Figure 5.1: Illustration of a node and a control volume.

In the ALE scheme, we first perform the Lagrangian step, corresponding to the time integral term in (5.10). In this way we can ignore the advection term in equation (5.2) and define a Lagrangian quantity q^L as

$$q^L = q^n + \overline{f}_q \Delta t. \tag{5.12}$$

where $\overline{f}_q$ is the time and volume averaged source of q in the computational cell during the time interval Δt, and q^n is the value of quantity q at time step n. In an explicit scheme, this average is approximated by the value at the end of the last time step. Often the source f_q for q is a function of q itself, such as drag force in the momentum equations for two-phase flows. In such cases, the source term $\overline{f}_q$ can be expressed as a function of q^L, and equation (5.12) is solved implicitly in CartaBlanca. The source term $\overline{f}_q$ may contain many terms representing different physical mechanisms. Typically, the terms representing mechanisms with short time scales, such as drag force and added mass force in the momentum exchange for two-phase flows, especially for tight coupling of the two phases, need to be treated implicitly. For instance using (2.14) and the closures described in Chapter 3, the Lagrangian velocity $\mathbf{u}_k^L$ for phase k is calculated as

$$\begin{aligned} \frac{\mathbf{u}_k^L - \mathbf{u}_k^n}{\Delta t} &= -\nabla p/\rho_k^0 + \nabla \cdot [\theta_k(\boldsymbol{\sigma}_k + p) + \boldsymbol{\sigma}_k^{Re}]/\rho_k + \mathbf{g}_k \\ + \quad & \frac{1}{\rho_k^0} \sum_{l=1}^{N} \theta_l C_{kl} \left(\frac{\mathbf{u}_l^L - \mathbf{u}_l^n}{\Delta t} - \frac{\mathbf{u}_k^L - \mathbf{u}_k^n}{\Delta t} \right) - \frac{1}{\rho_k^0} \sum_{l=1}^{N} \theta_l K_{kl} (\mathbf{u}_l^L - \mathbf{u}_k^L). \end{aligned} \tag{5.13}$$

In (5.13) the material acceleration of phase k is calculated as $(\mathbf{u}_k^L - \mathbf{u}_k^n)/\Delta t$, using Lagrangian velocities. To find the Lagrangian velocities for each phase, a system of equations for all the phases is solved. Except for the terms involving divergence and the gradient, all the terms only contain local variables. When the divergence and the gradient terms are calculated, these equations can be solved locally without referring to the values of their neighbors. The gradient of quantity q, such as the pressure in (5.13), at node i is calculated using values of q on the surfaces bounding the control volume of the node as

$$\nabla q = \frac{1}{V} \int_V \nabla q dV = \frac{1}{V} \int_S q \mathbf{n} dS. \tag{5.14}$$

In the discretized form, this is written as

$$\nabla q = \frac{1}{V} \sum_{\alpha=1}^{m} q_\alpha \mathbf{A}_\alpha, \tag{5.15}$$

where m is the total number of edges connected to the node, q_α is the value q at the mid-point of edge α, and the area $\boldsymbol{A}_\alpha$ is the vector sum over the vector surface areas bounding the control volume of node i that have a point in edge α. More precisely,

$$\boldsymbol{A}_\alpha = \sum_{\beta=1} \boldsymbol{A}_\alpha^\beta, \tag{5.16}$$

where the summation is over all the surfaces that touch the edge α, and $\mathbf{A}_\alpha^\beta$ is a surface vector of the β-th surface. The magnitude of $\mathbf{A}_\alpha^\beta$ equals the area of the surface and the direction of the vector is in the normal direction to the surface.

The pressure p_α at the mid point of the edge connecting nodes i and j is calculated as follows to ensure the mixture acceleration produced by the pressure gradient is continuous across the cell boundary.

$$(p_i - p_\alpha)/\rho_{mi} = (p_\alpha - p_j)/\rho_{mj}, \quad \text{or} \quad p_\alpha = \left(\frac{p_i}{\rho_{mi}} + \frac{p_i}{\rho_{mj}}\right) / \left(\frac{1}{\rho_{mi}} + \frac{1}{\rho_{mj}}\right), \tag{5.17}$$

where ρ_m is the mixture density, the sum of the macroscopic densities of all the phases.

For the Lagrangian step, the macroscopic density at the mid-point of the edge is the average value of the two nodes on both ends of the edge (see Fig. 5.1). Quantities other than the pressure and the macroscopic density at the mid-point are calculated as the average weighted by the macroscopic densities at the two end nodes of the edge as

$$q_\alpha = \frac{\rho_i q_i + \rho_j q_j}{\rho_i + \rho_j}, \tag{5.18}$$

where i and j are the two ends of edge α.

The divergences of a vector and a tensor such as velocity $\mathbf{u}_k$ and stress $\boldsymbol{\sigma}_k$ are calculated similarly,

$$\nabla \cdot \mathbf{u}_k = \frac{1}{V} \sum_{\alpha=1}^{m} \mathbf{u}_{k\alpha} \cdot \mathbf{A}_\alpha, \tag{5.19}$$

$$\nabla \cdot \boldsymbol{\sigma}_k = \frac{1}{V} \sum_{\alpha=1}^{m} \boldsymbol{\sigma}_{k\alpha} \cdot \mathbf{n}_\alpha A_\alpha. \tag{5.20}$$

After the Lagrangian step, the remapping step is performed as

$$q_k^{n+1} = q_k^L - \frac{1}{V} \sum_{\alpha=1}^{m} \hat{q}_{k\alpha} \mathbf{u}_{k\alpha} \cdot \mathbf{A}_\alpha \Delta t, \tag{5.21}$$

where the summation over surfaces $\boldsymbol{A}_\alpha$ is used to approximate the integrals over the inflow and outflow volume defined in (5.10). Depending on the sign of the inner product $\mathbf{u}_{k\alpha} \cdot \mathbf{A}_\alpha$ the volume $\mathbf{u}_{k\alpha} \cdot \mathbf{A}_\alpha \Delta t$ is an inflow or outflow volume. The quantity $\hat{q}_{k\alpha}$ is the averaged value inside such an inflow or outflow volume associated with phase k, and is calculated differently from (5.18) to ensure stability of the calculation, as we discuss in the following section.

5.3 Advection schemes

The remapping step described in the end of last section requires the calculation of the fluxing velocities and quantities to be fluxed on the interface of a control volume. In this section we describe methods used in CartaBlanca to calculate these quantities that minimize numerical instabilities.

The fluxing velocity across face α of a control volume is calculated based on the momentum equation

$$\begin{aligned} & \frac{\boldsymbol{u}_{k\alpha}^L - \boldsymbol{u}_{k\alpha}^n}{\Delta t} = -\nabla_\alpha p_k / \rho_\alpha^0 + \nabla_\alpha \cdot [\theta_k(\boldsymbol{\sigma}_k + p) + \boldsymbol{\sigma}_{Re}]/\rho_{k\alpha} + \mathbf{g}_{k\alpha} \\ + \quad & \frac{1}{\rho_k^0} \sum_{l=1}^{N} \theta_{l\alpha} C_{kl} \left(\frac{\boldsymbol{u}_{l\alpha}^L - \boldsymbol{u}_{l\alpha}^n}{\Delta t} - \frac{\boldsymbol{u}_{k\alpha}^L - \boldsymbol{u}_{k\alpha}^n}{\Delta t} \right) \\ - \quad & \frac{1}{\rho_{k\alpha}^0} \sum_{l=1}^{N} \theta_{l\alpha} K_{kl} (\boldsymbol{u}_{l\alpha}^L - \boldsymbol{u}_{k\alpha}^L). \end{aligned} \tag{5.22}$$

This face velocity is used only to compute the fluxing volumes, the inflow and outflow volumes in (5.10). In this calculation we only need the normal component on the surface. To obtain the normal velocity, we project $\mathbf{u}_{k\alpha}^L$ on the normal direction of surface α. Equation (5.22)

then becomes

$$\begin{aligned} & \boldsymbol{u}^L_{k\alpha}\cdot\mathbf{n}_\alpha = \boldsymbol{u}^n_{k\alpha}\cdot\mathbf{n}_\alpha - \nabla_\alpha p_k\cdot\mathbf{n}_\alpha\Delta t/\rho^0_\alpha + \nabla_\alpha\cdot[\theta_k(\boldsymbol{\sigma}_k+p)+\boldsymbol{\sigma}_{Re}]\cdot\mathbf{n}_\alpha\Delta t/\rho_{k\alpha} \\ + & \frac{1}{\rho^0_k}\sum_{l=1}^{N}\theta_{l\alpha}C_{kl}\left[(\boldsymbol{u}^L_{l\alpha}\cdot\mathbf{n}_\alpha-\boldsymbol{u}^n_{l\alpha}\cdot\mathbf{n}_\alpha)-(\boldsymbol{u}^L_{k\alpha}\cdot\mathbf{n}_\alpha-\boldsymbol{u}^n_{k\alpha}\cdot\mathbf{n}_\alpha)\right] \\ - & \frac{\Delta t}{\rho^0_{k\alpha}}\sum_{l=1}^{N}\theta_{l\alpha}K_{kl}(\boldsymbol{u}^L_{l\alpha}\cdot\mathbf{n}_\alpha-\boldsymbol{u}^L_{k\alpha}\cdot\mathbf{n}_\alpha)+\mathbf{g}_{k\alpha}\cdot\mathbf{n}_\alpha\Delta t. \end{aligned} \tag{5.23}$$

The pressure gradient on the cell interface α is calculated according to the following algorithm for a generic quantity q:

$$\nabla_\alpha q = \frac{(\nabla q)_i+(\nabla q)_j}{2}+\left[q_j-q_i-\frac{(\nabla q)_i+(\nabla q)_j}{2}\cdot\boldsymbol{d}_\alpha\right]\cdot\frac{\boldsymbol{d}_\alpha}{|\boldsymbol{d}_\alpha|^2}, \tag{5.24}$$

where $\boldsymbol{d}_\alpha=\boldsymbol{x}_j-\boldsymbol{x}_i$ is the distance vector from node i to node j sharing the interface α. In this algorithm the gradient along the direction connecting nodes i and j is replaced by the finite difference between the two nodes. This calculated pressure gradient is projected to the face normal $\boldsymbol{n}_\alpha=\boldsymbol{A}_\alpha/|\boldsymbol{A}_\alpha|$ to calculate $\nabla_\alpha p\cdot\mathbf{n}_\alpha$. When the face normal $\boldsymbol{n}_\alpha$ is coincident with the distance normal $\boldsymbol{d}_\alpha/|\boldsymbol{d}_\alpha|$, the projected pressure gradient reduces to

$$\nabla_\alpha p\cdot\mathbf{n}_\alpha=\frac{p_j-p_i}{\Delta r_{ji}}, \tag{5.25}$$

where Δr_{ji} is the distance from node i to node j. For a given pressure field, eq. (5.23) can be solved to find the fluxing volume $\mathbf{u}^L_{k\alpha}\cdot\mathbf{n}_\alpha A_\alpha dt$ across face α.

To perform the remapping step (5.21), we need to calculate a fluxed quantity $\hat{q}_{k\alpha}$ in the inflow and outflow volumes. As mentioned at the end of the last section, using the average of the values at the adjacent nodes as the value in the inflow and outflow volumes can introduce undesired numerical error and cause instability in the calculation. To avoid such numerical instability, the following methods are used to calculate the value of $\hat{q}_{k\alpha}$ on the interface, or in the inflow and outflow volumes.

5.3.1 Upwind advection

It is commonly known that a stable method with first order accuracy in spatial discretization is to take the value of the cell center upstream of the face as the value $\hat{q}_{k\alpha}$ in the fluxing

volume. The upstream cell is called the donor cell. However, this approach introduces a large amount of "numerical viscosity" that can cause unphysical diffusion of the solution.

5.3.2 van Leer limiting

To achieve second order accuracy, one needs to account for the effects of the gradient of the quantity q. With a piecewise-linear representation the value of q at $\mathbf{x}$ in a cell can be written as

$$q = q_i + \nabla q_i \cdot \mathbf{r}, \tag{5.26}$$

where $\mathbf{r} = \mathbf{x} - \mathbf{x}_i$ is the distance vector from the node i. The gradient ∇q_i is calculated using (5.15) with the value q_α taken as the averaged value of the neighboring cell centers with different weights, as described in Section 5.2.

However, such a scheme can result in unphysical situations. Consider a 1D problem, where the velocity field is a constant 1.0. Suppose that the values of ρ at nodes (cell centers) $m-1, m, m+1$ are 2.0, 0.1, 0.0, respectively, and $\Delta x = 1.0$. Now consider the fluxing density at the face between cell m and $m+1$. In this case, the donor cell is m. Based on (5.15), $\nabla\rho$ is actually equal to $\rho[m+1] - \rho[m]/2\Delta x = -1.0$. From (5.26), the fluxed ρ on the face is $\rho = 0.1 + (-1.0 * 0.5) = -0.4$, an unphysical negative value. It is apparent that this approach may destroy the monotonicity of a solution in the case of a uniform velocity.

To avoid these problems, we multiply the gradient ∇q in (5.26) by a limiter ℓ_q such that the value in the cell is calculated using

$$q = q_i + \ell_q \nabla q_i \cdot \mathbf{r}, \tag{5.27}$$

to ensure the value of q does not lie outside the maximum, q_{max}, and minimum, q_{min} of the neighboring cell centers. The use of a gradient limiter first appeared in the work of van Leer (1979) for a one-dimensional calculation. Dukowicz and Kodis (1987) extend the method to multi-dimensional cases as:

- First, obtain a trial gradient ∇q_i for the cell centers, e.g. use (5.15).

- Second, find the limiter ℓ_q based on

$$\ell_q = min\{1, \ell_{min}, \ell_{max}\}, \tag{5.28}$$

where

$$\ell_{max} = max\left\{0, \frac{q_{max} - q_i}{max\{q_e\} - q_i}\right\}, \quad \ell_{min} = max\left\{0, \frac{q_{min} - q_i}{min\{q_e\} - q_i}\right\}, \tag{5.29}$$

in which q_{max} and q_{min} are the maximum and minimum of the q-values at the surrounding nodes connected by edges, and $max\{q_e\}$ and $min\{q_e\}$ are the maximum and minimum of the q-values at the mid-point of the edges calculated using (5.26) with the trial gradient. The subscript e emphasizes that these values are calculated at the mid-point of the edges. This is the Barth-modified version of the van Leer limiting method (VanderHeyden and Kashiwa, 1998). In the original version of the van Leer limiting method (Dukowicz and Kodis, 1987), the values $max\{q_e\}$ and $min\{q_e\}$ in (5.29) are $max\{q_v\}$ and $min\{q_v\}$, the maximum and minimum of q-values at vertexes calculated using (5.26) with the trial gradient.

The limited gradient, $\ell_q \nabla q_i$, is then used as the gradient of q on the nodes. In the remapping step (5.21), when the density $\rho_{k\alpha}$ is calculated using (5.27), one can ensure positive density everywhere in the computational domain provided that the density after the Lagrangian step is positive. Also, the application of the van Leer advection scheme is monotone. That is, for a uniform velocity flow an initially monotonic density distribution will remain monotonic after the remapping step.

5.3.3 Compatibility of conserved quantities

Besides the advection of density, we also need to advect other quantities, such as momentum and temperature or enthalpy. The quantity q in the remapping step can be a compound quantity, such as enthalpy multiplied by density ($q_k = \rho_k h_k$). A desired property of a remapping scheme is to ensure no new maximum or minimum is created through the remapping step,

especially in the case of a constant velocity field. A remapping or advection scheme with this property is called compatible.

Using the advection scheme with the gradient limiter for both ρ_k and $q_k = \rho_k h_k$ does not guarantee such a property. Although monotonicity is preserved for ρ_k and $q_k = \rho_k h_k$, respectively, monotonicity is not necessarily preserved for $h_k = q_k/\rho_k$ since q_k and ρ_k are independent quantities. An additional scheme to calculate the fluxed quantity h_k is needed to ensure the monotonicity of h_k. We now describe such a scheme commonly used in CFD codes. This scheme was built on the van Leer scheme and was proposed by VanderHeyden and Kashiwa (1998).

In the following discussion, we use h (omitting subscript k) as an example; the treatment of other quantities such as the momentum components is similar. In this scheme, we again want to find a limiter ℓ_h to limit the gradient of h at the nodes such that the value in cell i calculated using

$$h = h_i + \ell_h \rho_i \nabla h_i \cdot \mathbf{r}/(\rho_i + \ell_\rho \nabla \rho_i \cdot \mathbf{r}) \tag{5.30}$$

does not lie outside the maximum, h_{max}, and minimum, h_{min} of the neighboring cell centers, where $\mathbf{r} = \mathbf{x} - \mathbf{x}_i$, ∇h_i is a trial gradient for the cell centers calculated using (5.15), and ℓ_ρ is the limiter for the macroscopic density calculated as described in the last subsection.

The calculation of ℓ_h is similar to that in (5.29):

$$\ell_h = min\{1, \ell_{min}, \ell_{max}\}, \tag{5.31}$$

where

$$\ell_{max} = max\left\{0, \frac{h_{max} - h_i}{max\{h_e\} - h_i}\right\}, \quad \ell_{min} = max\left\{0, \frac{h_{min} - h_i}{min\{h_e\} - h_i}\right\}, \tag{5.32}$$

in which h_{max} and h_{min} are the maximum and minimum of the h-values at the surrounding nodes connected by edges, and $max\{h_e\}$ and $min\{h_e\}$ are the maximum and minimum of the h-values at the mid-point of the edges calculated using (5.30) with $\ell_h = 1$. Again, we are using the Barth implementation of this scheme in CartaBlanca.

5.4 Calculation of pressure, density and volume fraction

The pressure at the nodes is obtained by solving the continuity equations and evolution equations for volume fractions and then enforcing the condition satisfied by volume fractions (3.3). The solution to the continuity equation (2.6) is divided into the Lagrangian step and the remapping step. In the Lagrangian step, the Lagrangian macroscopic density is calculated as

$$\rho_k^L = \rho_k^n + S_{mk}\Delta t, \tag{5.33}$$

where $S_{mk} = \int \rho_k^0 \dot{C}_k d\mathcal{P}$ is the mass source. To calculate the macroscopic density ρ_k^{n+1} at the end of the time step, we first calculate an interim macroscopic density

$$\rho_k^* = \rho_k^L - \frac{1}{V}\sum_{\alpha=1}^{m} \hat{\rho}_{k\alpha}\mathbf{u}_{k\alpha}\cdot\mathbf{n}_\alpha A_\alpha \Delta t, \tag{5.34}$$

where the summation is over all the faces around the node.

Similarly, the Lagrangian step for solving the evolution equation (3.15) for volume fractions can be written as

$$\theta_k^L = \theta_k^n + S_{vk}\Delta t, \tag{5.35}$$

where

$$S_{vk} = \alpha_{bi}\theta_i \nabla\cdot\widetilde{\boldsymbol{u}}_i + \int \dot{C}_i d\mathcal{P} \tag{5.36}$$

is the volume source. We note that the last term $\theta_k B_k$ in (3.15) is not contained in the volume source here. In this numerical scheme, the effect of this term is calculated later. After the following remapping step an interim volume fraction is calculated as

$$\theta_k^* = \theta_k^L - \frac{1}{V}\sum_{\alpha=1}^{m} \hat{\theta}_{k\alpha}\boldsymbol{u}_{k\alpha}\cdot\mathbf{n}_\alpha A_\alpha \Delta t. \tag{5.37}$$

Such calculated volume fractions do not sum to one because the effect of the last term in (3.15) has yet to be accounted for. To account for this effect we proceed to calculate an interim microscopic density $< \rho_k^0 >^* = \rho_k^*/\theta_k^*$ and the corresponding pressure p_k^* from the

averaged equation of state. Finally to ensure all volume fractions sum to one, we find a common pressure increment Δp for all the phases such that the following equation is satisfied

$$\sum_{i=1}^{M} \frac{\rho_i^*}{< \rho_i^0 > (p_i^* + \Delta p)} = 1. \tag{5.38}$$

The final pressure for each phase at the end of the time step is then $p_k^{n+1} = p_k^* + \Delta p$ and the final microscopic density is already calculated from solving (5.38), that is $< \rho_i^0 > (p_i^* + \Delta p)$. The volume fraction is then $\theta_k^{n+1} = \rho_i^* / < \rho_i^0 >^{n+1}$. In this scheme, the contribution of the last term in (3.15) to the volume fractions is not directly added through the source term (5.36) in the Lagrangian step but is accounted for in the process of solving (5.38) to find the common pressure increment Δp. The difference between θ_k^{n+1} and θ_k^* is the contribution of $\theta_k B_k$.

The scheme described above is for the CartaBlanca implementation of the multipressure model. For the single pressure model, the steps of calculating the interim volume fraction, the interim microscopic density and the interim pressure (equations (5.35) - (5.37)) are skipped. The pressure p_i^* in (5.38) is replaced with p^n.

For an explicit time advancement scheme, the normal face velocity $\boldsymbol{u}_{k\alpha} \cdot \mathbf{n}_\alpha$ used in (5.34) for the calculation of the interim macroscopic density ρ_k^* is calculated using the pressure gradient of the last time step (time level n). Equation (5.38) is then local and does not involve the pressures at the neighboring nodes. In this case, the interim macroscopic density ρ_k^* is taken as the macroscopic density ρ_k^{n+1} at the end of the time step.

For an implicit method, the normal face velocity is calculated using the updated (time level $n+1$) pressure gradient. The macroscopic density ρ_k^* is a function of the gradient of Δp; thus equation (5.38) is nonlocal and involves pressures at neighboring nodes. Iteration methods described in Chapter 7 are used in CartaBlanca to solve these coupled equations on the nodes in the domain. The macroscopic density ρ_k^{n+1} at the end of the time step is then the value of the interim macroscopic density ρ_k^* after the iterations are converged.

Chapter 6

Material point method

6.1 Introduction

In the Arbitrary Lagrangian-Eulerian (ALE) method introduced in the previous chapter, the primary variable is the Eulerian velocity, not the Lagrangian displacement of the material. This is a necessary choice for a numerical method intended to handle large deformation of materials, to avoid mesh tangling. However such a choice introduces issues associated with numerical diffusion. For instance, in a fluid-structure interaction problem, calculation of stress in the solid material needs to know the strain in the material, which needs to be calculated by integrating the strain rate over the entire deformation history of a material point in a numerical method with velocity as the primary variable. Numerical diffusion associated with advection schemes for Eulerian methods or ALE methods makes the task of following a material point very difficult.

In this chapter, we introduce the Material Point Method (MPM), to overcome both the mesh tangling issues associated with Lagrangian methods and the numerical diffusion issues associated with Eulerian methods.

In the MPM, materials are represented by both an Eulerian grid and Lagrangian points. During a deformation, the Eulerian grid stays fixed while the Lagrangian points move. Each material point carries basic quantities such as mass, microscopic density, velocity, etc. These quantities, or changes of the quantities, are interpolated back and forth between the grid and

the Lagrangian points using shape functions. Depending on the type of elements, different shape functions are used. Currently quadrilateral elements and hexahedral elements with bi-linear shape functions are available in CartaBlanca.

There are three requirements for a shape function S_n of node n to satisfy in MPM, as in a finite element method. They are

$$\sum_{n=1}^{N} S_n(\boldsymbol{x}) = 1, \tag{6.1}$$

$$\sum_{n=1}^{N} \boldsymbol{x}_n S_n(\boldsymbol{x}) = \boldsymbol{x}, \tag{6.2}$$

where $\boldsymbol{x}$ is any point in the computational domain, and N is the total number of nodes in the domain, and that S_n has a local support (non-zero region) within the elements surrounding node n. Requirements (6.1) and (6.2) ensure that a rigid body motion of a material does not cause stress in the material; the last requirement ensures that related equations can be solved locally as we shall discuss in the following sections.

For multiphase flow calculations in CartaBlanca, the user can specify that a phases is calculated using either MPM or the ALE method. The mathematical guiding principles for solving the equations are different in the ALE method and MPM. The numerical scheme used in a Material Point Method is based on the weak solution of the governing partial differential equations, as discussed in the following sections. To properly calculate interactions between phases calculated using MPM and phases calculated using the ALE method, we discuss a numerical scheme to combine these two methods in section 5.

6.2 Weak form of the equations

Let q_k be a quantity contained in a unit mass of phase k at location $\boldsymbol{x}$ and time t. The evolution equation for q_k can be written in the following Lagrangian form:

$$\rho_k \frac{dq_k}{dt} = \rho_k \left(\frac{\partial q_k}{\partial t} + \widetilde{\boldsymbol{u}}_k \cdot \nabla q_k \right) = \nabla \cdot (\theta_k \boldsymbol{L}_k) + \rho_k G_k, \tag{6.3}$$

where $\boldsymbol{L}_k$ is the tensor that is one order higher than q_k, and G_k is the source density for q_k.

MPM seeks an approximate weak solution of this equation. The weak solution of (6.3) is defined as a function q that satisfies the following relation for any continuous trial function, h_k:

$$\left(\rho_k \frac{dq_k}{dt}, h_k\right) = \left(\nabla \cdot (\theta_k \boldsymbol{L}_k) + \rho_k G_k, h_k\right), \tag{6.4}$$

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

$$(q_k, h_k) = \int_\Omega q_k h_k dv, \tag{6.5}$$

where the integration is over the entire computational domain Ω, and v is the volume in the domain. MPM seeks an approximate solution of (6.3) in a subspace of continuous functions in which all functions take the following form:

$$q_k(\boldsymbol{x}, t) = \sum_{n=1}^{N} q_{kn}(t) S_n(\boldsymbol{x}), \tag{6.6}$$

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

By taking the trial function h_k in the same form as (6.6),

$$h_k = \sum_{\ell=1}^{N} \delta q_{k\ell} S_\ell(\boldsymbol{x}), \tag{6.7}$$

we can write equation (6.4) in this subspace as

$$\sum_{\ell=1}^{N}\sum_{n=1}^{N} m_{k\ell n} \frac{dq_{kn}}{dt} \delta q_{k\ell} = \sum_{\ell=1}^{N} \delta q_{k\ell} \left[(\rho_k G_k, S_\ell) - (\theta_k \boldsymbol{L}_k, \nabla S_\ell) + \int_{\partial\Omega} \theta_k \boldsymbol{L}_k \cdot \boldsymbol{n} S_\ell(\boldsymbol{x}) dS\right], \tag{6.8}$$

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

$$m_{k\ell n} = \int_\Omega \rho_k S_\ell(\boldsymbol{x}) S_n(\boldsymbol{x}) dv. \tag{6.9}$$

All such $m_{k\ell n}$'s form a mass matrix with element in the ℓ-th row and n-th column being $m_{k\ell n}$.

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

$$\sum_{n=1}^{N} m_{k\ell n} \frac{dq_{kn}}{dt} = \left[(\rho_k G_k, S_\ell) - (\theta_k \boldsymbol{L}_k, \nabla S_\ell) + \int_{\partial\Omega} \theta_k \boldsymbol{L}_k \cdot \boldsymbol{n} S_\ell(\boldsymbol{x}) dS\right]. \tag{6.10}$$

This is a system of linear equations for the rates of change of q_k at the nodes. To avoid solving this system of equations, we note that $m_{k\ell n}$ is non-zero only for the nodes 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 dq_{kn}/dt can be approximated as $dq_{k\ell}/dt$. With this approximation the system of linear equations is decoupled and can then be solved as

$$\frac{dq_{k\ell}}{dt} = \frac{1}{m_{k\ell}} \left[(\rho_k G_k, S_\ell) - (\theta_k \boldsymbol{L}_k, \nabla S_\ell) + \int_{\partial\Omega} \theta_k \boldsymbol{L}_k \cdot \boldsymbol{n} S_\ell(\boldsymbol{x}) dS \right], \tag{6.11}$$

where

$$m_{k\ell} = \sum_{n=1}^{N} m_{k\ell n} = \int_\Omega \rho_k S_\ell(\boldsymbol{x}) dv. \tag{6.12}$$

The second equality is a result of property (6.1) for shape functions. This approximate way of decoupling the system of equations is equivalent to approximating the element $m_{k\ell n}$ in the mass matrix by $m_{k\ell}\delta_{\ell n}$. The error introduced by this approximation is of the same order as the error introduced by the spatial discretization. The effect of this approximation on energy dissipation is discussed in the last section of this chapter. To calculate the inner product in the first term on the right hand side of (6.11), we write G_k in the form of (6.6) to find

$$(\rho_k G_k, S_\ell) = \int_\Omega \rho_k \sum_{n=1}^{N} G_{kn} S_n(\boldsymbol{x}) S_\ell(\boldsymbol{x}) dv \approx G_{k\ell} \int_\Omega \rho_k \sum_{n=1}^{N} S_n(\boldsymbol{x}) S_\ell(\boldsymbol{x}) dv = G_{k\ell}\, m_{k\ell}. \tag{6.13}$$

We have again taken advantage of local support of the shape function as in (6.11) and approximated G_{kn} with $G_{k\ell}$ within the support of S_ℓ. This equation states that the source term G_k, such as the interaction force between phases, can be calculated at the nodes.

To calculate the second inner product on the right hand side of (6.11), we now introduce an approximate scheme to calculate the inner product using particle quantities. We approximate the inner product $(\rho_k q_k, h_k)$ as

$$(\rho_k q_k, h_k) \approx \sum_{p=1}^{N_{kp}} m_{kp} q_{kp} h_{kp}, \tag{6.14}$$

where N_{kp} is the number of phase k particles in the domain, and m_{kp} is the mass of particle

p. In this way,

$$(\theta_k \boldsymbol{L}_k, \nabla S_\ell) = (\rho_k \boldsymbol{L}_k / \rho_k^0, \nabla S_\ell) \approx \sum_{p=1}^{N_{kp}} v_{kp} \boldsymbol{L}_k(\boldsymbol{x}_p, t) \cdot \nabla S_\ell(\boldsymbol{x}_p), \tag{6.15}$$

where $v_{kp} = m_{kp}/\rho_{kp}^0$ is the volume of particle p, and ρ_{kp}^0 is the microscopic density of the particle. Since S_ℓ has a local support, the summation only needs to be carried out for the particles in the elements surrounding node ℓ.

With the right hand side calculated using (6.13) and (6.15) we can write (6.11) as

$$\frac{dq_{k\ell}}{dt} = G_{k\ell} - \frac{1}{m_{k\ell}} \sum_{p=1}^{N_{kp}} v_{kp} \boldsymbol{L}_k(\boldsymbol{x}_p, t) \cdot \nabla S_\ell(\boldsymbol{x}_p) + \frac{1}{m_{k\ell}} \int_{\partial\Omega} \theta_k \boldsymbol{L}_k \cdot \boldsymbol{n} S_\ell(\boldsymbol{x}) dS. \tag{6.16}$$

The last term in (6.16) represents the effect of boundaries. In CartaBlanca, surfaces of particle domains with forces acting on them are currently treated as interior to the computational domain and the surface forces are considered as the phase interaction forces between the phases occupying the domains on either side of the surface. In other words, there is a layer of elements outside the domain that particles occupy. In this way the surface integral vanishes and the phase interaction forces on the surface is treated as the body forces and is calculated as described in (6.13). The only term calculated using particle quantities is the second term. This term is an approximate way of calculating $(\theta_k \boldsymbol{L}_k, \nabla S_\ell)$ as described in (6.15). For smoothly varying θ_k and $\boldsymbol{L}_k$, one can prove that the error of this approximation in (6.15) is of second order in the mesh size. For a discontinuous field, the error could be larger; especially if the discontinuity results in a very small mass $m_{k\ell}$ of the node, the second term in (6.16) could become singular and cause instability in the calculation. To regulate this instability, for a node with mass $m_{k\ell}$ smaller than a lower bound mass m_{klb}, (currently is set to be 1/4 of the smallest particle mass), we limit the inner product calculated for the node using (6.15) by a factor $m_{k\ell}/m_{klb}$. This is equivalent to setting the lower bound m_{klb} for $m_{k\ell}$ in the second term of (6.16). This modification only affects those nodes whose surrounding elements contains particles far away from the node. In this case, the volume fraction θ_k is small in the region close to the node, therefore the inner product $(\theta_k \boldsymbol{L}_k, \nabla S_\ell)$ should be reduced accordingly.

With an exception for section 6.4, the rest of this chapter does not involve phase interactions. For simplicity we drop the subscript k when there is little possibility for confusion.

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

$$q_\ell^L = q_\ell^n + \frac{dq_\ell}{dt}\Delta t. \tag{6.17}$$

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 (6.3), the evolution equation in the Lagrangian form. This corresponds to (5.12) in the ALE method. To update q values on particles we interpolate the rate of change, or the material derivative, from the nodes to particles as

$$q_p^{n+1} = q_p^n + \sum_{\ell=1}^{N}(q_\ell^L - q_\ell^n)S_\ell(\boldsymbol{x}_p). \tag{6.18}$$

It is important to note that we interpolate the change $q_\ell^L - q_\ell^n$, not q^L, to the particles. In this way, the change of particle values is only caused by the right hand side of (6.16). If the right hand side of (6.16) vanishes, the particle value does not change. Therefore, such node to particle interpolation does not introduce numerical diffusion to the solution. If we interpolate Lagrangian node values q_ℓ^L to particles, significant numerical diffusion will occur. We also note that the value for the shape function S_ℓ is evaluated at the time step n. This is because particles are Lagrangian points; from time step n to $n+1$ 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 particles and the coordinate system during the time advancement, and the values of the shape functions remain unchanged at the particle locations. Following the motion of the material, the new positions of the particles are calculated as

$$\boldsymbol{x}_p^{n+1} = \boldsymbol{x}_p^n + \boldsymbol{u}_p\Delta t. \tag{6.19}$$

where velocity $\boldsymbol{u}_p$ is

$$\boldsymbol{u}_p = \sum_{\ell=1}^{N} \boldsymbol{u}_\ell^L . S_\ell(\boldsymbol{x}_p). \tag{6.20}$$

In (6.20), again, the value for the shape function S_ℓ is evaluated at the time step n for the same reason as in (6.18). The velocity $\boldsymbol{v}_p$ used to advance the particle positions is not the particle velocity, but rather the velocity interpolated from the nodes, because particles are Lagrangian points following the motion and deformation of the material, while a particle velocity should be understood as the averaged momentum per unit mass carried by the particle. The velocity gradient used in the calculation of the stresses on particles is also calculated using (6.20) (with the same parameter η). Since the spatial variation of the velocity is represented by the shape functions, the velocity gradients are calculated by differentiating the shape functions.

Note that the change rate $dq_{k\ell}/dt$ calculated in (6.16) 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}^{n+1} on particles. To obtain the scheme of calculating node values using particle values, we note that the inner product $(\rho q, h)$ can also be calculated using (6.6) to find

$$(\rho q, h) = \sum_{\ell=1}^{N} \sum_{n=1}^{N} m_{\ell n} q_n h_\ell, \tag{6.21}$$

where $m_{\ell n}$ is defined in (6.9) (with subscript k dropped) and can be approximately calculated by setting $q = S_n(\boldsymbol{x})$ and $h = S_\ell(\boldsymbol{x})$ in (6.21) as

$$m_{\ell n} = (\rho S_\ell, S_n) \approx \sum_{p=1}^{N_p} m_p S_\ell(\boldsymbol{x}_p) S_n(\boldsymbol{x}_p), \tag{6.22}$$

where N_p is the total number of the particles representing the phase. Comparing (6.14) with (6.21) and then using (6.6) we find

$$\sum_{\ell=1}^{N} \sum_{n=1}^{N} m_{\ell n} q_n h_\ell \approx \sum_{\ell=1}^{N} \sum_{p=1}^{N_p} m_p q_p h_\ell S_\ell(\boldsymbol{x}_p). \tag{6.23}$$

Since h_ℓ is arbitrary in (6.14), we have

$$\sum_{n=1}^{N} m_{\ell n} q_n \approx \sum_{p=1}^{N_p} m_p q_p S_\ell(\boldsymbol{x}_p). \tag{6.24}$$

For $q = 1$ in (6.24), we find that m_ℓ (with subscript k denoting phase omitted), defined in (6.12), can be approximately calculated as

$$m_\ell = \sum_{n=1}^{N} m_{\ell n} \approx \sum_{p=1}^{N_p} m_p S_\ell(\boldsymbol{x}_p). \tag{6.25}$$

Equation (6.24) is a set of coupled equations for q_n at the nodes. Again because of the local support of the shape functions, by approximating q_n with q_ℓ we have

$$q_\ell \approx \frac{\sum_{p=1}^{N_p} m_p q_p S_\ell(\boldsymbol{x}_p)}{m_\ell} = \frac{\sum_{p=1}^{N_p} m_p q_p S_\ell(\boldsymbol{x}_p)}{\sum_{p=1}^{N_p} m_p S_\ell(\boldsymbol{x}_p)}. \tag{6.26}$$

The node quantity is approximated as the mass weighted mean of its surrounding particles. In particular, by letting $q = 1/\rho^0$, we find

$$\rho_n^0 = \frac{\sum_{p=1}^{N_p} m_p S_n(\boldsymbol{x}_p)}{\sum_{p=1}^{N_p} v_p S_n(\boldsymbol{x}_p)} \tag{6.27}$$

for microscopic densities on the nodes, where $v_p = m_p/\rho^0$ is the volume of the particle.

This approximate way to solve equation (6.24) is equivalent to approximating the mass matrix consisting of elements $m_{\ell n}$ by a diagonal mass matrix with elements $m_\ell \delta_{\ell n}$. This approximation saves a large amount of computation by avoiding the full solution of the coupled linear equations, but introduces a numerical dissipation to the method. The effects of this dissipation will be discussed in section 6.

6.3 Solution of the mass conservation equation

In a Lagrangian method the motion of the material is followed, therefore mass conservation is automatically satisfied. However, in the Material Point Method, one needs to calculate macroscopic density defined at an Eulerian point in the domain. The evolution of the macroscopic density satisfies the mass conservation equation (2.6) written in the Eulerian description. To find the macroscopic density, we seek an approximate weak solution to the equation on the Eulerian frame by using shape functions defined on the Eulerian frame. These shape functions are the shape functions described in the previous section. The only difference is that,

in this section and the following section on volume fraction evolution, they are defined on the Eulerian frame; in the previous section they are defined on the Lagrangian frame moving with the material.

We multiply equation (2.6) by an arbitrary continuous function h and then integrate over the entire domain to find

$$\frac{\partial}{\partial t}(\rho, h) = (\rho, \boldsymbol{u} \cdot \nabla h) - \int_{\partial\Omega} \rho h \boldsymbol{u} \cdot \boldsymbol{n} dS + \int_{\Omega} r h dv, \tag{6.28}$$

where $r = \int \rho^0 \dot{C} d\mathcal{P}$ is the mass source for the phase. After restricting the function h in the subspace spanned by the shape functions defined in the Eulerian frame as in (6.6), noting the value of h on the nodes is arbitrary, using the approximation (6.14), and then integrating over the time interval, we find

$$\begin{aligned}\sum_{p=1}^{N_p^{n+1}} m_p^{n+1} S_n(\boldsymbol{x}_p^{n+1}) = \sum_{p=1}^{N_p^n} [m_p^n S_n(\boldsymbol{x}_p^n) + m_p^n \boldsymbol{v}_p \cdot \nabla S_n(\boldsymbol{x}_p^n) \Delta t] \\ -\Delta t \int_{\partial\Omega} \rho S_n(\boldsymbol{x}) \boldsymbol{u} \cdot \boldsymbol{n} dS + \Delta t \int_{\Omega} r S_n(\boldsymbol{x}) dv,\end{aligned} \tag{6.29}$$

where N_p^{n+1} and N_p^n are the number of particles in the computational domain at time level $n+1$ and n. The surface integral on the right hand side represents the mass of the phase fluxed into or out of the computational domain and is calculated as

$$-\Delta t \int_{\partial\Omega} \rho S_n(\boldsymbol{x}) \boldsymbol{u} \cdot \boldsymbol{n} dS = \sum m_p^{in} S_n(\boldsymbol{x}_p^{in}) - \sum m_p^{out} S_n(\boldsymbol{x}_p^{out}), \tag{6.30}$$

where the first and the second summations are respectively over all particles flowing into or out of the domain during the time step. The volume integral on the right hand side is calculated as

$$\Delta t \int_{\Omega} r S_n(\boldsymbol{x}) dv = \sum_{p=1}^{N^n} (m_p^{n+1} - m_p^n) S_n(\boldsymbol{x}_p), \tag{6.31}$$

where the mass change on particles is calculated by interpolating the mass change rate at nodes to particle locations,

$$m_p^{n+1} = m_p^n + m_p^n \sum_{\ell=1}^{N} \frac{r_\ell}{\rho_\ell} S_\ell(\boldsymbol{x}_p) = m_p^n \sum_{\ell=1}^{N} \left(1 + \frac{r_\ell}{\rho_\ell}\right) S_\ell(\boldsymbol{x}_p). \tag{6.32}$$

By summing (6.32) over all particles and then using (6.25), we find $\sum_{p=1}^{N_p^n} m_p^{n+1} = \sum_{\ell=1}^{N}(1 + r_\ell/\rho_\ell)m_\ell^n$. This ensures the conservation of the total mass between the particles and nodes. Using (6.30), (6.31) and (6.25), equation (6.29) can be rewritten as

$$\begin{aligned} m_\ell^{n+1} &= \sum_{p=1}^{N_p^n} m_p^{n+1} S_n(\boldsymbol{x}_p^{n+1}) + \sum m_p^{in} S_n(\boldsymbol{x}_p^{in}) - \sum m_p^{out} S_n(\boldsymbol{x}_p^{out}) \\ &= \sum_{p=1}^{N_p^{n+1}} m_p^{n+1} S_n(\boldsymbol{x}_p^{n+1}), \end{aligned} \tag{6.33}$$

because the contribution from a particle that is included in the first summation above but flows out of the domain during the time advancement is canceled exactly by the corresponding term involving m_p^{out}, even if, sometimes for the smoothness of the solution, one chooses to regard $S_n(\boldsymbol{x}_p^{out}) \neq 0$ for particles outside the domain. This has yet to be implemented in CartaBlanca.

Equation (6.33) proves that the calculation of the node mass using (6.25) is consistent with the weak solution of the mass conservation equation (2.6). In other words the mass conservation equation is automatically satisfied with this material point method.

6.4 Weak solution for volume fraction equations

As mentioned in Chapter 2, only two of the three equations, (2.6), (2.7) and (2.8) are independent. Since (6.25) is the solution of the mass conservation equation (2.6) as proved in the last section, we need to solve only one of the two remaining equations. In the material point method implemented in CartaBlanca, we choose to solve the evolution equation (2.7) for the volume fraction. But before seeking a weak solution of the equation we first write it in a form so that the solution of it can be found with intuition.

By letting $q_k = 1$ in (2.3) we find

$$\frac{\partial \theta_k}{\partial t} + \nabla \cdot (\theta_k < \boldsymbol{u}_k >) = \theta_k < \nabla \cdot \boldsymbol{u}_k > + \int \dot{C}_k d\mathcal{P}. \tag{6.34}$$

Since the sum of the volume fractions equals one, we have

$$\nabla \cdot \boldsymbol{u}_m = \sum_{k=1}^{M} \left(\theta_k < \nabla \cdot \boldsymbol{u}_k > + \int \dot{C}_k d\mathcal{P} \right), \tag{6.35}$$

where $\boldsymbol{u}_m = \sum_{k=1}^{M} \theta_k < \boldsymbol{u}_k >$ is the mixture velocity and M is the number of phases. This equation is equivalent to (3.6) after using (3.4) and neglecting the correlation between the velocity fluctuation and the microscopic density fluctuation. By neglecting this correlation we can also neglect the difference between $< \boldsymbol{u}_k >$ and $\widetilde{\boldsymbol{u}}_k$, and the mixture velocity can also be calculated as $\boldsymbol{u}_m = \sum_{k=1}^{M} \theta_k \widetilde{\boldsymbol{u}}_k$. Under this approximation, we then add and subtract $\nabla \cdot (\theta_k \widetilde{\boldsymbol{u}}_m)$ on the left hand side of (6.34) and write it as

$$\frac{d_m \theta_k}{d_m t} + \nabla \cdot [\theta_k (\widetilde{\boldsymbol{u}}_k - \boldsymbol{u}_m)] + \theta_k \nabla \cdot \boldsymbol{u}_m = \theta_k < \nabla \cdot \boldsymbol{u}_k > + \int \dot{C}_k d\mathcal{P}, \tag{6.36}$$

where

$$\frac{d_m \theta_k}{d_m t} = \frac{\partial \theta_k}{\partial t} + \boldsymbol{u}_m \cdot \nabla \theta_k \tag{6.37}$$

is the material derivative following the mixture velocity. To seek an approximate weak solution we multiply both sides of (6.36) by a trial function h in the form of (6.6) defined on the frame moving at the mixture velocity $\boldsymbol{u}_m$, then integrate the resulting equation over the entire computational domain to find

$$\begin{aligned} \sum_{n=1}^{N} h_n \frac{d_m v_{kn}}{d_m t} = & - \sum_{n=1}^{N} h_n \left\{ \int_\Omega S_n(\boldsymbol{x}) \nabla \cdot [\theta_k (\widetilde{\boldsymbol{u}}_k - \boldsymbol{u}_m)] dv \right. \\ + & \left. \int_\Omega S_n(\boldsymbol{x}) \left(\theta_k < \nabla \cdot \boldsymbol{u}_k >) + \int \dot{C}_k d\mathcal{P} - \theta_k \nabla \cdot \boldsymbol{u}_m \right) dv \right\}, \end{aligned} \tag{6.38}$$

where

$$v_{kn} = \int_\Omega \theta_k(\boldsymbol{x}, t) S_n(\boldsymbol{x}) dv. \tag{6.39}$$

Since h_n is arbitrary, we then have

$$\begin{aligned} \frac{d_m v_{kn}}{d_m t} = & - \int_\Omega S_n(\boldsymbol{x}) \nabla \cdot [\theta_k (\widetilde{\boldsymbol{u}}_k - \boldsymbol{u}_m)] dv \\ + & \int_\Omega S_n(\boldsymbol{x}) \left(\theta_k < \nabla \cdot \boldsymbol{u}_k > + \int \dot{C}_k d\mathcal{P} - \theta_k \nabla \cdot \boldsymbol{u}_m \right) dv. \end{aligned} \tag{6.40}$$

Summing (6.40) over all phases, we find that in the sense of a weak solution the condition (6.35) is equivalent to

$$\frac{d_m}{d_m t} \sum_{k=1}^{M} v_{kn} = 0, \tag{6.41}$$

because the first term on the right hand side of (6.40) sums to zero after the use of the definition for mixture velocity.

To calculate v_{kn} defined in (6.39), we can approximate the volume integral by the sum over all particle volumes as in (6.21),

$$v_{kn} \approx \sum_{p=1}^{N_p} v_p S_n(\boldsymbol{x}_p). \tag{6.42}$$

Since the shape function has support only in the elements surrounding node n the sum only involves particles in those elements. The volume v_{kn} is often regarded as the phase k volume associated with node n. For this reason, the ratio

$$\theta^A_{kn} = v_{kn}/V^c_n, \tag{6.43}$$

where V^c_n is the control volume surrounding node n (Figure 5.1), is regarded as the approximated volume fraction at the node. Since in CartaBlanca implementation of MPM , v_{kn} is calculated using (6.42), there is no guarantee that the sum of such ratios over all the phases equals one. For this reason, we call the ratio the apparent volume fraction θ^A_{kn} for phase k at node n.

For phases that are not represented by particles, the volume v_{kn} defined in (6.39) can be approximately calculated as $\theta_{kn} \int S_n dv$. For median meshes constructed as described in Chapter 5, one can approximate $\int S_n dv$ as V^c_n, the control volume of node n provided the mesh is not significantly distorted. In this case, the apparent volume fraction defined in (6.43) can be approximated by volume fraction θ_{kn} calculated from the ALE method described in Chapter 5.

Since V^c_n is a constant, by dividing V^c_n across (6.41) we find

$$\frac{\partial}{\partial t} \sum_{k=1}^{M} \theta^A_{kn} + \boldsymbol{u}_m \cdot \nabla \sum_{k=1}^{M} \theta^A_{kn} = 0. \tag{6.44}$$

In CartaBlanca the continuity condition is enforced by satisfying this equation at every node in the domain. The convection term in (6.44) is calculated as

$$\boldsymbol{u}_m \cdot \nabla \sum_{k=1}^{M} \theta_{kn}^A = \nabla \cdot \left(\boldsymbol{u}_m \sum_{k=1}^{M} \theta_{kn}^A \right) - \sum_{k=1}^{M} \theta_{kn}^A \nabla \cdot \boldsymbol{u}_m, \tag{6.45}$$

and the divergence is calculated using (5.19) with the surface quantities calculated as described in the last chapter.

Similarly to the definition of the apparent volume fraction, the ratio m_{kn}/V_n^c, where m_{kn} is the lumped mass of phase k to node n calculated using (6.25), is also regarded as the apparent macroscopic density ρ_{kn}^A for the node. In the next section we will discuss its use in enforcing the continuity condition for the material point method implemented in CartaBlanca. Although the apparent volume fraction and apparent macroscopic density calculated here are not the true volume fraction and the true macroscopic density, their ratio

$$\frac{\rho_{kn}^A}{\theta_{kn}^A} = \frac{\sum_{p=1}^{N_p} m_p S_n(\boldsymbol{x}_p)/V_n^c}{\sum_{p=1}^{N_p} v_p S_n(\boldsymbol{x}_p)/V_n^c} \tag{6.46}$$

is the averaged microscopic density as calculated using (6.27).

6.5 The use of apparent volume fractions

In the Material Point Method introduced in this chapter, the equation for the volume fraction is solved differently than the way described in Chapter 5 for an Arbitrary Lagrangian-Eulerian method, because the discretization error of the Material Point Method makes the apparent volume fractions not sum to one. The method described in Chapter 5 fails to solve the equations for volume fractions as we now show. We define a function

$$f(p,t) = \sum_{k=1}^{M} \frac{\rho_k^A(p,t)}{< \rho_k^0 > (p)} = \sum_{k=1}^{M} \theta_k^A. \tag{6.47}$$

This function can only be calculated with an error of $O(\Delta x)^d$, where d is the order of accuracy of the spatial discretization. If we enforce (5.38), after the Taylor expansion of $f(p,t)$ we have

to solve

$$f(p^n, t^n) + \frac{\partial f}{\partial t}\Delta t + \frac{\partial f}{\partial p}\Delta p = 1. \tag{6.48}$$

If the numerical scheme for calculating volume fraction is of order d in accuracy at time t^n, that is it contains an error of $O(\Delta x)^d$ at time t^n or $f(p^n, t^n) = 1 + O(\Delta x)^d$, and equation (6.48) becomes

$$\frac{\partial f}{\partial t} + \frac{\partial f}{\partial p}\frac{\Delta p}{\Delta t} = \frac{O(\Delta x)^d}{\Delta t}, \tag{6.49}$$

for Δp. In many calculations the time step is proportional to the mesh size Δx, therefore the error on the right hand side of (6.49) is of $O(\Delta x)^{d-1}$. In this way, the accuracy is reduced by an order in each time step, and the numerical scheme fails within a finite number of time steps. Indeed, if this method were used, one would see stress change for an elastic body when it translates undeformed through the mesh.

The approach introduced in the previous section solves evolution equation (6.34) for the volume fraction. As long as each term in the equation is calculated with first order accuracy, the solution for θ_k is first order in accuracy after the time integration. Since the numerical differentiation method employed in CartaBlanca only introduces second order errors, the divergences involved in (6.45) are accurate to first order in Δx.

Although the approach to solving for volume fractions is based on principles different from the ALE method in Chapter 5, the numerical implementation is quite similar and can be combined with the ALE method as follows.

1. For the phases calculated using MPM, calculate the interim apparent macroscopic density ρ_{kp}^{A*} as $m_{k\ell}/V_\ell^c$, with $m_{k\ell}$ calculated using (6.25). For phases calculated using the ALE method, ρ_k^{A*} is the value of macroscopic density ρ_k^* obtained from (5.34).

2. For the phases calculated using MPM, calculate interim microscopic density $< \rho_{kn}^0 >^*$ on particles using (3.16) by neglecting the effects of B_k, and with the velocity divergence calculated by taking the divergence of velocity (6.20).

3. For the phases calculated using MPM, interpolate the calculated interim microscopic density to nodes using (6.27). For phases calculated using the ALE method, the microscopic density at the node is calculated as described in section 5.4. Namely, $< \rho_k^0 >= \rho_k^*/\theta_k^*$ with θ_k^* calculated using (5.37).

4. Use the equation of state for the phase to find an interim pressure p_{kn}^*.

5. Calculate the expected sum of the apparent volume fractions as

$$\left(\sum_{k=1}^{M} \theta_{k\ell}^A\right)^{n+1} = \left(\sum_{k=1}^{M} \theta_{k\ell}^A\right)^{n} - \left(\boldsymbol{u}_m \cdot \nabla \sum_{k=1}^{M} \theta_{k\ell}^A\right) \Delta t, \tag{6.50}$$

with the convection term calculated using (6.45) and the mixture velocity normalized as

$$\boldsymbol{u}_m = \sum_{k=1}^{M} \theta_k^A \widetilde{\boldsymbol{u}}_k / \sum_{k=1}^{M} \theta_k^A. \tag{6.51}$$

6. Find a common pressure increment Δp such that

$$\sum_{k=1}^{M} \frac{\rho_{kn}^{A*}}{< \rho_{k\ell}^0 >(p_{k\ell}^* + \Delta p)} = \left(\sum_{k=1}^{M} \theta_{k\ell}^A\right)^{n+1}. \tag{6.52}$$

If the single pressure model is used, $\alpha_{bk} = 0$ in (3.16), steps 2, 3 and 4 are skipped, and $p_{k\ell}^*$ is replaced by $p_{k\ell}^n$ in (6.52).

6.6 Conservations affected by node value calculation

As mentioned in section 2, to avoid solving systems of equations (6.24) for node values, the mass matrix consisting of elements $m_{\ell n}$ is approximated by a diagonal mass matrix with elements $m_\ell \delta_{\ell n}$. This approximation saves a large amount of calculation. In this section we examine the effect of this approximation on conservation of mass, momentum and energy.

Summing over (6.25) for all nodes, we find that mass conservation is automatically satisfied in this scheme for mapping particle values to nodes.

According to (6.24) the momentum at a node is the sum of the momenta of the surrounding particles,

$$\sum_{n=1}^{N} m_{\ell n}\mathbf{u}_n \approx \sum_p m_p \mathbf{u}_p S_\ell(\boldsymbol{x}_p). \tag{6.53}$$

By summing over both sides of (6.53) for all ℓ's, one finds the total momentum conservation between the grid description and particle description,

$$\sum_{\ell=1}^{N}\sum_{n=1}^{N} m_{\ell n}\mathbf{u}_n = \sum_p m_p \mathbf{u}_p, \tag{6.54}$$

after the use of (6.1). As we show later, the momentum conservation is still satisfied after the approximation of the mass matrix.

The total kinetic energy calculated based on the node variables is

$$K_g = \frac{1}{2}(\boldsymbol{u}, \rho\boldsymbol{u}) = \frac{1}{2}\sum_{\ell=1}^{N}\sum_{n=1}^{N} m_{\ell n}\mathbf{u}_\ell \cdot \mathbf{u}_n, \tag{6.55}$$

and based on the particles is

$$K_p = \frac{1}{2}\sum_p m_p \mathbf{u}_p^2. \tag{6.56}$$

Multiplying both sides of (6.53) by $\mathbf{u}_\ell$ and then summing the resulting equation over all ℓ's we have

$$2K_g = \sum_p m_p \mathbf{u}_p \cdot \sum_\ell \mathbf{u}_\ell S_{\ell p}. \tag{6.57}$$

Using the inequality

$$\mathbf{u}_p \cdot \sum_k \mathbf{u}_\ell S_\ell(\boldsymbol{x}_p) \le \frac{1}{2}\left[\mathbf{u}_p^2 + \left(\sum_\ell \mathbf{u}_\ell S_\ell(\boldsymbol{x}_p)\right)^2\right], \tag{6.58}$$

equation (6.57) can be written as

$$\begin{aligned} 2K_g &\le \frac{1}{2}\sum_p m_p \mathbf{u}_p^2 + \frac{1}{2}\sum_p m_p \left(\sum_n \mathbf{u}_n S_n(\boldsymbol{x}_p)\right)^2 \\ &= K_p + \frac{1}{2}\sum_p m_p \left(\sum_n \mathbf{u}_n S_n(\boldsymbol{x}_p)\right)\cdot\left(\sum_\ell \mathbf{u}_\ell S_\ell(\boldsymbol{x}_p)\right). \end{aligned} \tag{6.59}$$

After exchanging the order of summation and the use of (6.22) for $m_{\ell n}$, one finds

$$2K_g \le K_p + \frac{1}{2}\sum_{\ell=1}^{N}\sum_{n=1}^{N} m_{\ell n}\mathbf{u}_\ell \mathbf{u}_n = K_p + K_g, \tag{6.60}$$

or

$$K_g \leq K_p. \tag{6.61}$$

This shows that the kinetic energy calculated on the nodes is smaller than or equal to that calculated on the particles. However we now prove the difference is kept constant during a time advancement without the approximation for the mass matrix. We now consider the change in the kinetic energy. Using (6.18) one finds

$$\begin{aligned} & 2(K_p^{n+1} - K_p^n) \\ = & \sum_p m_p \left[(\mathbf{u}_p^{n+1})^2 - (\mathbf{u}_p^n)^2 \right] \\ = & 2 \sum_{\ell=1}^{N} (\mathbf{u}_\ell^L - \mathbf{u}_\ell^n) \cdot \sum_p m_p \mathbf{u}_p^n S_\ell(\boldsymbol{x}_p) \\ + & \sum_p m_p \sum_{\ell=1}^{N} (\mathbf{u}_\ell^L - \mathbf{u}_\ell^n) S_\ell(\boldsymbol{x}_p) \cdot \sum_{j=1}^{N} (\mathbf{u}_j^L - \mathbf{u}_j^n) S_j(\boldsymbol{x}_p), \end{aligned} \tag{6.62}$$

where superscripts n and $n+1$ denotes the values at the end of time steps n and $n+1$, and the superscript L denotes the values at the end of the Lagrangian step. Using (6.53) on $\sum_p m_p \mathbf{u}_p^n S_\ell(\boldsymbol{x}_p)$, and (6.22) on $\sum_p m_p S_\ell(\boldsymbol{x}_p) S_j(\boldsymbol{x}_p)$, one finds

$$\begin{aligned} & 2(K_p^{n+1} - K_p^n) \\ = & \sum_{\ell=1}^{N} \sum_{j=1}^{N} m_{\ell j} \left[2\mathbf{u}_j^n \cdot (\mathbf{u}_\ell^L - \mathbf{u}_\ell^n) + (\mathbf{u}_\ell^L - \mathbf{u}_\ell^n) \cdot (\mathbf{u}_j^L - \mathbf{u}_j^n) \right] \\ = & \sum_{\ell=1}^{N} \sum_{j=1}^{N} m_{\ell j} \mathbf{u}_\ell^L \cdot \mathbf{u}_j^L - \sum_{\ell=1}^{N} \sum_{j=1}^{N} m_{\ell j} \mathbf{u}_\ell^n \cdot \mathbf{u}_j^n \\ = & 2(K_g^L - K_g^n). \end{aligned} \tag{6.63}$$

Combining (6.61) and (6.63) we note that while the kinetic energy calculated on the grid is less than or equal to that calculated on the particles, the changes in the kinetic energy in every time step are the same when calculated on the grid and the particles. In proving this relation, we used (6.53) to convert particle velocities to the grid. However, calculation of grid velocities using (6.53) requires the inversion of the mass matrix. To avoid this matrix inversion, the

mass matrix is approximated by the diagonal matrix with elements m_ℓ as defined in (6.25); that is,

$$m_{\ell n} \approx m_\ell \delta_{\ell n}. \tag{6.64}$$

With this approximation the velocity $\boldsymbol{u}_n$ on node n can be obtained from (6.26) as the mass weighted average of the velocities of the surrounding particles, and (6.54) becomes

$$\sum_{\ell=1}^{N} m_\ell \boldsymbol{u}_\ell = \sum_{p=1}^{N_p} m_p \boldsymbol{u}_p. \tag{6.65}$$

This shows that the total momentum is conserved under this approximation.

With the mass matrix approximation the total kinetic energy on the grid k_g can be written as

$$k_g = \frac{1}{2} \sum_{\ell=1}^{N} m_\ell \boldsymbol{u}_\ell^2. \tag{6.66}$$

Using (6.26) we can replace $\sum_p m_p \mathbf{u}_p^n S_{\ell p}$ in (6.62) with $m_\ell \boldsymbol{u}_\ell^n$ and write the equation as

$$2(K_p^{n+1} - K_p^n) = 2 \sum_{\ell=1}^{N} m_\ell \boldsymbol{u}_\ell^n \cdot (\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n) + \sum_{\ell=1}^{N} \sum_{j=1}^{N} m_{\ell j} (\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n) \cdot (\boldsymbol{u}_j^L - \boldsymbol{u}_j^n). \tag{6.67}$$

As a consequence we find

$$\begin{aligned} & 2(k_g^L - k_g^n) - 2(K_p^{n+1} - K_p^n) \\ = & \sum_{\ell=1}^{N} \sum_{j=1}^{N} (m_\ell \delta_{kj} - m_{\ell j})(\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n) \cdot (\mathbf{u}_j^L - \boldsymbol{u}_j^n) \\ = & \sum_{\ell=1}^{N} (m_\ell - m_{\ell\ell})(\boldsymbol{u}_\ell^L - \mathbf{u}_\ell^n)^2 - \sum_{\ell \neq j} m_{\ell j} (\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n) \cdot (\mathbf{u}_j^L - \boldsymbol{u}_j^n). \end{aligned} \tag{6.68}$$

Noting that $m_{\ell j} = m_{j\ell} \geq 0$, we have

$$\begin{aligned} & 2(k_g^L - k_g^n) - 2(K_p^{n+1} - K_p^n) \\ \geq & \sum_{\ell=1}^{N} (m_\ell - m_{\ell\ell})(\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n)^2 - \frac{1}{2} \sum_{\ell \neq j} m_{\ell j} [(\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n)^2 + (\boldsymbol{u}_j^L - \boldsymbol{u}_j^n)^2] \\ = & \sum_{\ell=1}^{N} (m_\ell - m_{\ell\ell})(\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n)^2 - \frac{1}{2} \sum_{\ell \neq j} [m_{\ell j} (\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n)^2 + m_{j\ell} (\boldsymbol{u}_j^L - \boldsymbol{u}_j^n)^2] \\ = & \sum_{\ell=1}^{N} (m_\ell - \sum_j m_{\ell j})(\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n)^2 = 0, \end{aligned} \tag{6.69}$$

or the kinetic energy dissipation D resulting from the lump sum of the mass matrix:

$$D = (k_g^L - k_g^n) - (K_p^{n+1} - K_p^n) \geq 0. \tag{6.70}$$

Since the velocity difference $\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n$ is proportional to the time step Δt, using (6.68) the energy dissipation D is seen to be proportional to $(\Delta x)^2$.

The change of the kinetic energy calculated on the particles is always less than or equal to the kinetic energy change calculated on the grid if the approximation (6.64) is used (Cummins and Brackbill, 2002). In other words the approximation introduces the numerical dissipation (6.70). To illustrate this dissipation, let us now consider the motion of a linear elastic body free of body and boundary forces. In this case, the change of grid kinetic energy defined in (6.66) during the Lagrangian step from time step n to $n+1$ can be calculated as

$$\begin{aligned} k_g^L - k_g^n &= \frac{1}{2}\sum_{\ell=1}^{N} m_\ell(\boldsymbol{u}_\ell^L + \boldsymbol{u}_\ell^n)(\boldsymbol{u}_\ell^L - \boldsymbol{u}_\ell^n) \\ &= -\frac{1}{2}\sum_{\ell=1}^{N}(\boldsymbol{u}_\ell^L + \boldsymbol{u}_\ell^n)\sum_{p=1}^{N_p} v_p \boldsymbol{\sigma}_p^n : \nabla S_\ell(\boldsymbol{x}_p)\Delta t \\ &= -\sum_{p=1}^{N_p} v_p \boldsymbol{\sigma}_p^n : \dot{\boldsymbol{\varepsilon}}_p^{L-1/2}\Delta t, \end{aligned} \tag{6.71}$$

where we have used (6.18) for velocity, and strain rate $\dot{\boldsymbol{\varepsilon}}_p^{L-1/2}$ is the symmetric part of the velocity gradient $\nabla \boldsymbol{u}^{L-1/2}$ at particle p calculated using velocity $\boldsymbol{u}^{L-\alpha} = (1-\alpha)\boldsymbol{u}_\ell^L + \alpha \boldsymbol{u}_\ell^n$ with $\alpha = 1/2$.

$$\nabla \boldsymbol{u}^{L-\alpha} = \sum_{\ell}^{N}[(1-\alpha)\boldsymbol{u}_\ell^L + \alpha \boldsymbol{u}_\ell^n]S_\ell(\boldsymbol{x}_p). \tag{6.72}$$

The change in the kinetic energy calculated in (6.71) does not account for the effects of regulating the particle force term in (6.16) as discussed in section 2. The effect of this particle force modification can be accounted for by setting a floor on the node mass m_ℓ to m_{lb} in the definition (6.66) of the grid kinetic energy. Since the both m_ℓ and m_{lb} are small compared to the particle mass, their difference can be neglected.

The elastic potential energy U_p on particles can be calculated from the stress $\boldsymbol{\sigma}_p$ stored for each particle as

$$U_p = \frac{1}{2}\sum_p v_p \boldsymbol{\sigma}_p \cdot \boldsymbol{C}_p^{-1} \cdot \boldsymbol{\sigma}_p = \frac{1}{2}\sum_p v_p \boldsymbol{\varepsilon}_p \cdot \boldsymbol{C}_p \cdot \boldsymbol{\varepsilon}_p, \tag{6.73}$$

where v_p is the particle volume, $\boldsymbol{\sigma}_p$ is the stress, $\boldsymbol{\varepsilon}_p$ is the strain and $\boldsymbol{C}_p$ is the elastic stiffness tensor on particle p. Stress on a particle is calculated by discretizing (4.13) as

$$\boldsymbol{\sigma}_p^{n+1} = \frac{\boldsymbol{\sigma}_p^n + \boldsymbol{C}_p \cdot \dot{\boldsymbol{\varepsilon}}_p^{L-\alpha}\Delta t}{\sqrt{1 + tr(\dot{\boldsymbol{\varepsilon}}_p^{L-\alpha})\Delta t}}, \tag{6.74}$$

where $\dot{\boldsymbol{\varepsilon}}_p^{(L-\alpha)}$ is the strain rate calculated as the symmetric part of the velocity gradient defined in (6.72). Currently $\alpha = 0$ in our MPM implementation. The denominator in (6.74) results from the last term on the right hand side of ((4.13)). The particle volume change can be calculated as

$$v_p^{n+1} = v_p^n[1 + tr(\dot{\boldsymbol{\varepsilon}}_p^{L-\alpha})\Delta t]. \tag{6.75}$$

The change of the potential energy can be calculated as

$$\begin{aligned} U_p^{n+1} - U_p^n &= \frac{1}{2}\sum_{p=1}^{N_p}[v_p^{n+1}\boldsymbol{\sigma}_p^{n+1} \cdot \boldsymbol{C}_p^{-1} \cdot \boldsymbol{\sigma}_p^{n+1} - v_p^n \boldsymbol{\sigma}_p^n \cdot \boldsymbol{C}_p^{-1} \cdot \boldsymbol{\sigma}_p^n] \\ &= \sum_{p=1}^{N_p} v_p^n \boldsymbol{\sigma}_p^n : \dot{\boldsymbol{\varepsilon}}_p^{L-\alpha}\Delta t + \frac{1}{2}\sum_{p=1}^{N_p} v_p^n \dot{\boldsymbol{\varepsilon}}_p^{L-\alpha} \cdot \boldsymbol{C}_p \cdot \dot{\boldsymbol{\varepsilon}}_p^{L-\alpha}(\Delta t)^2, \end{aligned} \tag{6.76}$$

where we have used the symmetry property of the elastic stiffness tensor ($(C_p)_{ij\ell m} = (C_p)_{\ell mij}$). Using (6.71) we have

$$\begin{aligned} k_g^L + U_p^{n+1} = k_g^n + U_p^n &+ \sum_{p=1}^{N_p} v_p^n \boldsymbol{\sigma}_p^n : (\dot{\boldsymbol{\varepsilon}}_p^{L-\alpha} - \dot{\boldsymbol{\varepsilon}}_p^{L-1/2})\Delta t \\ &+ \frac{1}{2}\sum_{p=1}^{N_p} v_p^n \dot{\boldsymbol{\varepsilon}}_p^{L-\alpha} \cdot \boldsymbol{C}_p \cdot \dot{\boldsymbol{\varepsilon}}_p^{L-\alpha}(\Delta t)^2. \end{aligned} \tag{6.77}$$

Finally, after using (6.70)we find

$$\begin{aligned} K_g^{n+1} + U_p^{n+1} = K_g^n + U_p^n &+ \sum_{p=1}^{N_p} v_p^n \boldsymbol{\sigma}_p^n : (\dot{\boldsymbol{\varepsilon}}_p^{L-\alpha} - \dot{\boldsymbol{\varepsilon}}_p^{L-1/2})\Delta t \\ &+ \frac{1}{2}\sum_{p=1}^{N_p} v_p^n \dot{\boldsymbol{\varepsilon}}_p^{L-\alpha} \cdot \boldsymbol{C}_p \cdot \dot{\boldsymbol{\varepsilon}}_p^{L-\alpha}(\Delta t)^2 - D. \end{aligned} \tag{6.78}$$

Noting that the difference $\dot{\boldsymbol{\varepsilon}}_p^{L-\alpha} - \dot{\boldsymbol{\varepsilon}}_p^{L-1/2}$ is proportional to time step Δt, the terms summing over particles in (6.77) and (6.78) are of second order in the time step. Since D is proved to be always positive in (6.70), the consequence of lump summing the mass matrix is a numerical energy dissipation. Relation (6.78) show the numerical scheme used in CartaBlanca has a numerical error on energy conservation of second order in both spatial and time discretization.

6.7 Calculation of stress acceleration

In the CartaBlanca implementation of the material point method, the momentum equation (2.14) is rewritten as

$$\begin{aligned} & \frac{\partial}{\partial t}(\theta_k < \rho_k^0 > \widetilde{\boldsymbol{u}}_k) + \nabla \cdot (\theta_k < \rho_k^0 > \widetilde{\boldsymbol{u}}_k \widetilde{\boldsymbol{u}}_k) = \theta_k \nabla \cdot \boldsymbol{\sigma}_{Ak} + \nabla \cdot (\theta_k < \boldsymbol{\sigma}_k >^N) \\ + \quad & \nabla \cdot [\theta_k (< \boldsymbol{\sigma} >_k^P - \boldsymbol{\sigma}_{Ak})] + \int \dot{C}_k \rho_k^0 \boldsymbol{u}_k d\mathcal{P} + \boldsymbol{f}_k + \theta_k < \rho_k^0 > \widetilde{\boldsymbol{b}}, \end{aligned} \tag{6.79}$$

where $< \boldsymbol{\sigma}_k >^N$ denotes that the stress is calculated on the mesh nodes, and $< \boldsymbol{\sigma}_k >^P$ denotes that the stress is calculated on the material points. The sum

$$< \boldsymbol{\sigma}_k >^N + < \boldsymbol{\sigma}_k >^P = < \boldsymbol{\sigma}_k > + \boldsymbol{\sigma}_k^{Re}. \tag{6.80}$$

The second term on the right hand side of (6.79) is treated as a part of the body force, $\rho_k G_k$, in (6.3); and the third term is treated as $\nabla \cdot (\theta_k \boldsymbol{L}_k)$ in the equation. The acceleration due to the node stress $< \boldsymbol{\sigma}_k >^N$ is calculated using (6.13); and the acceleration due to the stress on the material points is calculated using (6.15). In a material point calculation, since the way of treating the body force carries an error related to lump summing the mass matrix as discussed in Section 6.2, therefore it is preferred that all the stresses are calculated on the material points, while the stress calculated on the nodes provides a supplementary role. For instance, using the Kelvin-Voigt model, the elastic stress component on the material points are calculated in an incremental manner. To include the viscous stress on the material points in an incremental manner, one needs to calculate $d\dot{\boldsymbol{\varepsilon}}_k/dt$, which includes a term $\widetilde{\boldsymbol{u}}_k \cdot \nabla \dot{\boldsymbol{\varepsilon}}_k$. Since the shape function used in this MPM implementation is either linear or bi-linear for

the velocity, $\nabla\dot{\boldsymbol{\varepsilon}}_k$ cannot be calculated accurately. The formulation of (6.79) with $< \boldsymbol{\sigma}_k >^N$ provides an option to calculate the viscous component of the stress on the nodes.

In (6.79) the subtraction of the auxiliary stress $\boldsymbol{\sigma}_{Ak}$ is done on the stress on the material points instead of on the node stress to provide stability in the calculation. In the case of constant and equal stresses for $< \boldsymbol{\sigma}_k >^P$ and $< \boldsymbol{\sigma}_{Ak} >$, the third term on the right hand side of (6.79) is identically zero. If the stress $< \boldsymbol{\sigma}_k >^N = 0$ when the material is at rest, then no motion will be caused in this situation. On the other hand, if $< \boldsymbol{\sigma}_{Ak} >$ is subtracted from $< \boldsymbol{\sigma}_k >^N$, instead of from $< \boldsymbol{\sigma}_k >^P$, and is calculated on the nodes, the inconsistent numerical errors from the different ways of calculating the accelerations caused by the stresses on the material points and on the nodes result in a non-zero value for the sum of the second and the third terms in (6.79); and then cause an artificial motion of the material.

Chapter 7

Solver

CartaBlanca employs the Jacobian-Free Newton-Krylov method introduced by Brown and Saad in the early 1990s (Dana .) In a Newton's method a set of nonlinear equations $\boldsymbol{F}(\boldsymbol{u}) = 0$ is solved iteratively starting from an initial guess $\boldsymbol{u}_0$. Sequential improvement of the solution is made by solving

$$\boldsymbol{J}(\boldsymbol{u}_n)\boldsymbol{\delta}_n = -\boldsymbol{F}(\boldsymbol{u}_n), \tag{7.1}$$

and letting $\boldsymbol{u}_{n+1} = \boldsymbol{u}_n + \boldsymbol{\delta}_n$, where $\boldsymbol{J}$ is the Jacobian of the system.

In a Newton-Krylov method, the linear equation system (7.1) is solved using a Krylov method. For a linear equation system, a Krylov method starts from an initial guess $\boldsymbol{\delta}^{(0)}$. The initial residual $\boldsymbol{r}^{(0)}$ is obtained as

$$\boldsymbol{r}^{(0)} = -\boldsymbol{F} - \boldsymbol{J}\boldsymbol{\delta}^{(0)}. \tag{7.2}$$

To find the solution of (7.1), the initial guess is corrected as $\boldsymbol{\delta}_n = \boldsymbol{\delta}^{(0)} + \boldsymbol{z}$ and the correction $\boldsymbol{z}$ is found by solving

$$\boldsymbol{J}\boldsymbol{z} = \boldsymbol{r}^{(0)}. \tag{7.3}$$

The approximate solution for the linear equations (7.3) is found iteratively, in the Krylov subspace

$$K_m = span\{\boldsymbol{r}^{(0)}, \boldsymbol{J}\boldsymbol{r}^{(0)}, ..., \boldsymbol{J}^{m-1}\boldsymbol{r}^{(0)}\}, \tag{7.4}$$

where m is the number of iterations. For a non-singular Jacobian $\boldsymbol{J}$, we can show that in every iteration the dimension of the Krylov space is either increased by one or a solution for (7.3) is found in K_m. Suppose K_m has dimension m, or equivalently the vectors in the list (7.4) are linearly independent. If vector $\boldsymbol{J}^m\boldsymbol{r}^{(0)}$ is linearly independent of the vectors in the list (7.4) then the Krylov space K_{m+1} has dimension $m+1$. If $\boldsymbol{J}^m\boldsymbol{r}^{(0)}$ is linearly dependent on the the vectors in the list (7.4) then we have

$$\boldsymbol{J}^m\boldsymbol{r}^{(0)} = \alpha_0\boldsymbol{r}^{(0)} + \sum_{i=1}^{m-1} \alpha_i\boldsymbol{J}^i\boldsymbol{r}^{(0)}. \tag{7.5}$$

The coefficient α_0 cannot be zero in (7.5), otherwise by factoring $\boldsymbol{J}$ we find a nonzero solution for $\boldsymbol{J}\boldsymbol{x} = \boldsymbol{0}$, which is impossible for a non-singular Jacobian $\boldsymbol{J}$. Rewriting (7.5) we find $\boldsymbol{J}\boldsymbol{z} = \boldsymbol{r}^{(0)}$, where

$$\boldsymbol{z} = \frac{1}{\alpha_0}\left(\boldsymbol{J}^{m-1}\boldsymbol{r}^{(0)} - \sum_{i=1}^{m-1} \alpha_i\boldsymbol{J}^{i-1}\boldsymbol{r}^{(0)}\right) \in K_m. \tag{7.6}$$

Since the solution space has finite dimension, a solution will eventually be found using the Krylov method. In many practical systems a sufficiently accurate solution is found before the number of iterations reaches the dimension of the entire space. The basis of the Krylov space is constructed by the Gram-Schmidt orthogonalization process, therefore we only need to calculate $\boldsymbol{J}\boldsymbol{z}$; the matrix $\boldsymbol{J}$ is not needed explicitly and a Jacobian-free method can be used to solve the equations. The product $\boldsymbol{J}\boldsymbol{z}$ is calculated approximately as

$$\boldsymbol{J}\boldsymbol{z} \approx \frac{F(\boldsymbol{u}_n + \varepsilon\boldsymbol{z}) - F(\boldsymbol{u}_n)}{\varepsilon}, \tag{7.7}$$

where ε is a small scalar perturbation parameter.

7.1 Preconditioning

The solution of equation (7.3) using a Krylov method inevitably requires preconditioning. Preconditioning accelerates the Krylov solution method by improving the condition of the matrix the Krylov method directly sees. To explain the hybrid preconditioning done in CartaBlanca, it is useful to review some basic concepts. To start, recall that preconditioning requires the

construction of a matrix operator that approximates the original matrix operator. The approximate operator can then be used to transform (7.3) into a system that is easier for the Krylov method to solve. This can be done in two ways, called right and left preconditioning.

7.1.1 Right preconditioning

For right preconditioning, we transform (7.3) as

$$\boldsymbol{J}\boldsymbol{R}^{-1}\boldsymbol{R}\boldsymbol{z} = \boldsymbol{r}^0, \tag{7.8}$$

$$\widetilde{\boldsymbol{J}}\boldsymbol{y} = \boldsymbol{r}^0, \tag{7.9}$$

where $\widetilde{\boldsymbol{J}} = \boldsymbol{J}\boldsymbol{R}^{-1}$ and $\boldsymbol{y} = \boldsymbol{R}\boldsymbol{z}$. If $\boldsymbol{R}$ is chosen properly such that $\widetilde{\boldsymbol{J}}$ is better conditioned, then the Krylov iteration can be accelerated. The Krylov subspace defined in (7.4) for $\widetilde{\boldsymbol{J}}$ is

$$\widetilde{K_m} = span\{\boldsymbol{r}^{(0)}, \widetilde{\boldsymbol{J}}\boldsymbol{r}^{(0)}, ..., \widetilde{\boldsymbol{J}}^{m-1}\boldsymbol{r}^{(0)}\}. \tag{7.10}$$

After the solution of (7.9) is found in this Krylov subspace, the solution of (7.3), or (7.8), can be found with $\boldsymbol{z} = \boldsymbol{R}^{-1}\boldsymbol{y}$.

In many numerical calculations, the right preconditioner is chosen according to the physics related to the problem as we shall discuss in section 7.2. The right preconditioner chosen is often quite "close" (in physical meaning) to the matrix $\boldsymbol{J}$. In this way the Krylov iteration procedure with the right preconditioner can be viewed as an iterative prediction-correction method to find the solution for (7.3). In Krylov iterations we can view $\boldsymbol{z}' = \boldsymbol{R}^{-1}\boldsymbol{r}_0^n$, where $\boldsymbol{r}_0^n$ is the residual at the end of n Krylov iterations, as the predicted solution of (7.3) and then substitute this solution into (7.3) to find the residual $\boldsymbol{r}_0^{n+1} = \boldsymbol{J}\boldsymbol{z}' = \widetilde{\boldsymbol{J}}\boldsymbol{r}_0^n$ in the Krylov subspace $\widetilde{K_m}$ for the next iteration. In a Jacobian-free method, the vector $\widetilde{\boldsymbol{J}}\boldsymbol{y}$ is calculated using (7.7) as

$$\widetilde{\boldsymbol{J}}\boldsymbol{y} = \boldsymbol{J}\boldsymbol{R}^{-1}\boldsymbol{y} \approx \frac{F(\boldsymbol{u}_n + \varepsilon\boldsymbol{R}^{-1}\boldsymbol{y}) - F(\boldsymbol{u}_n)}{\varepsilon}. \tag{7.11}$$

In this way each Krylov iteration requires the calculation $\boldsymbol{z} = \boldsymbol{R}^{-1}\boldsymbol{y}$. Therefore $\boldsymbol{R}$ should be chosen such that $\boldsymbol{z}$ can be solved for easily. In many cases the choice of the preconditioner

depends on the physical problem to be solved. We shall discuss this further in section 7.2.

7.1.2 Left Preconditioning

Similarly to right preconditioning, we find a left preconditioner $\boldsymbol{L}$, as we shall discuss in the following section, such that $\boldsymbol{L}^{-1}\boldsymbol{J}\boldsymbol{z}$ is better conditioned to accelerate the Krylov iterations. We transform (7.3) as

$$\boldsymbol{L}^{-1}\boldsymbol{J}\boldsymbol{z} = \boldsymbol{L}^{-1}\boldsymbol{r}^0. \tag{7.12}$$

Using (7.2) we can write

$$\boldsymbol{L}^{-1}\boldsymbol{J}\boldsymbol{z} = -\boldsymbol{L}^{-1}\boldsymbol{F} - \boldsymbol{L}^{-1}\boldsymbol{J}\boldsymbol{\delta}^{(0)}. \tag{7.13}$$

In many cases the preconditioner $\boldsymbol{L}$ is a function of $\boldsymbol{u}_n$ or $\boldsymbol{z}$. In solving (7.13) the value of $\boldsymbol{u}_n$ is kept fixed. The solution $\boldsymbol{u}_{n+1} = \boldsymbol{u}_n + \boldsymbol{z}$ from this approach does not satisfy $\boldsymbol{L}^{-1}(\boldsymbol{u}_{n+1})\boldsymbol{J}(\boldsymbol{u}_{n+1})\boldsymbol{z} = \boldsymbol{L}^{-1}(\boldsymbol{u}_{n+1})\boldsymbol{r}^0$; instead it satisfies $\boldsymbol{L}^{-1}(\boldsymbol{u}_n)\boldsymbol{J}(\boldsymbol{u}_n)\boldsymbol{z} = \boldsymbol{L}^{-1}(\boldsymbol{u}_n)\boldsymbol{r}^0$, or (7.3).

7.2 Hybrid preconditioning

In a numerical calculation to accelerate the convergence of the Krylov iteration, often both left and right preconditioners are applied. With right and left preconditioners, equation (7.3) is written as

$$\boldsymbol{L}^{-1}\boldsymbol{J}\boldsymbol{R}^{-1}\boldsymbol{R}\boldsymbol{z} = -\boldsymbol{L}^{-1}\boldsymbol{r}^{(0)}. \tag{7.14}$$

Specification of preconditioners is not unique and is a subject of research. In CartaBlanca, the left operator preconditions inter-field coupling at a given node only; and the right operator preconditions inter-node coupling within a field on a field-by-field basis.

To perform the left preconditioning step for inter-field coupling, a small matrix is assembled from the elements of the Jacobian that describe the interaction among the field variables at a node. We neglect the interactions with all other variables on other nodes. This results in an $N \times N$ system of equations at each node where N is the number of field variables. We invert

these systems and report the resulting transformed residuals, the right hand side of (7.14). Thus, the left preconditioner matrix is a block diagonal matrix composed of $N \times N$ blocks for each node in the mesh.

From this result, the transformed system is then subjected to right preconditioning to address the inter-node coupling in each field. This is done in a conventional fashion from within the Krylov solver itself.

To explain further, let $\boldsymbol{J}_{ij}^{fg}$ be the element of the Jacobian matrix at node i interacting with node j for field f interacting with field g. Also, let $\boldsymbol{z}_i^f$ and $\boldsymbol{r}_i^f$ be, respectively, the correction for the solution and the residual for field f at node i. Then the equation for the residual for field f at node i is

$$\sum_g \sum_j \boldsymbol{J}_{ij}^{fg} \boldsymbol{z}_j^g = -\boldsymbol{r}_i^f. \tag{7.15}$$

For preconditioning inter-field coupling at node i, we extract only the equations for that node and ignore effects of other nodes,

$$\sum_g \boldsymbol{J}_{ii}^{fg} \boldsymbol{z}_i^g = -\boldsymbol{r}_i^f - \sum_g \sum_{j \neq i} \boldsymbol{J}_{ij}^{fg} \boldsymbol{z}_j^g. \tag{7.16}$$

The left preconditioner $\boldsymbol{L}$ is then a block diagonal, $N \times N$ matrix, $\boldsymbol{J}_{ii}^{fg}$.

To perform the inter-node, or right, preconditioning for field f, we extract only the equations for that field and drop the effects of other fields,

$$\sum_j \boldsymbol{J}_{ij}^{ff} \boldsymbol{z}_j^f = -\boldsymbol{r}_i^f. \tag{7.17}$$

In other words, the right preconditioner matrix $\boldsymbol{R}$ is constructed by replacing the corresponding elements in an identity matrix, with the same rank as $\boldsymbol{J}$, by $\boldsymbol{J}_{ij}^{ff}$. Such a constructed preconditioner $\boldsymbol{R}$ does not contain inter-field elements ($\boldsymbol{J}_{ij}^{fg}, f \neq g$). Because of this property, in a numerical calculation, the right preconditioner matrix $\boldsymbol{R}$ does not need to be formed explicitly. The solution for equation $\boldsymbol{y} = \boldsymbol{R}\boldsymbol{z}$ is found field by field without the need to consider inter-field interactions.

In this hybrid preconditioning approach, the physics of inter-node interactions for a field is mostly calculated by right preconditioning and the physics of inter-field interactions at a node are mostly calculated by left preconditioning. The Krylov iterations ensure the final solution correctly accounts for the remaining physics of inter-node and inter-field interactions.

In solving $\boldsymbol{y} = \boldsymbol{R}\boldsymbol{z}$, approximate solutions are obtained using smoothers such as Jacobi, SSOR and ILU0 are available in CartaBlanca as user options. In addition, the Conjugate Gradient method can also be used. In order to promote parallel algorithmic scaling, a 2-level method (Knoll et al. 1999) can be used to bring a zeroth order multigrid character to the inter-node preconditioning operation.

7.2.1 Right preconditioner for pressure

Equation (3.3) for pressure is preconditioned by extracting the operator acting on the pressure variable. Following the technique of Kashiwa et. al. (1994), we find the pressure correction operator to be

$$V^n \sum_k \frac{\rho_k}{\rho_k^0 c_k^2} p^L - \frac{(\Delta t)^2}{\rho_k^0} \sum_e \sum_k \frac{\rho_{ke}}{\rho_{ke}^0} \nabla p^L \cdot \boldsymbol{n}_e A_e, \tag{7.18}$$

where p^L is the Lagrangian step, time advanced pressure, and subscripts e denote the average over the time interval of the conserved quantity passing through the face e. The pressure gradient in the normal direction of the surface e is

$$\nabla p^L \cdot \boldsymbol{n}_e = \frac{p^L_{left} - p^L_{right}}{|\boldsymbol{x}_{left} - \boldsymbol{x}_{right}|} \frac{\boldsymbol{x}_{left} - \boldsymbol{x}_{right}}{|\boldsymbol{x}_{left} - \boldsymbol{x}_{right}|} \cdot \boldsymbol{n}_e, \tag{7.19}$$

where *left* and *right* refer to the logical left and right nodes for a given face. The quantity $\boldsymbol{x}$ is the position vector of the node. Pressure operator (7.18) is used as the right preconditioner for the pressure equation.

7.2.2 In-function left preconditioner for coupling terms

In this section we present a method for solving the energy equation in temperature form, for two phases k and ℓ. Extension to velocity coupling is identical. To be able to consider tight

temperature coupling between phases, the Lagrangian temperature of the phase is obtained by solving

$$T_k^L = T_k^n + \frac{\Delta t}{\rho_k^n C_k}\left[\theta_k \sum_{\ell \neq k} \theta_\ell H_{k\ell}(T_\ell^L - T_k^L) + \theta_k \nabla \cdot (K_m \nabla T_m)\right]. \tag{7.20}$$

This corresponds to the use of the left preconditioner $\boldsymbol{L}$ in the temperature equation (3.37)

$$\boldsymbol{L} = \begin{bmatrix} 1 + \frac{\theta_\ell H_{k\ell}}{\rho_k^0 C_k} & -\frac{\theta_\ell R_{k\ell}}{\rho_k^0 C_k} \\ -\frac{\theta_k H_{k\ell}}{\rho_\ell^0 C_\ell} & 1 + \frac{\theta_k R_{k\ell}}{\rho_\ell^0 C_\ell} \end{bmatrix}. \tag{7.21}$$

This is applied on a per-node basis to the residual.

The Lagrangian step using (5.12) for the enthalpy equation (2.18) can be written as

$$h_k^L = h_k^n + \Delta t\left[\theta_k \sum_{\ell \neq k} \theta_\ell H_{k\ell}(T_\ell^L - T_k^L) + \theta_k \nabla \cdot (K_m \nabla T_m)\right]. \tag{7.22}$$

The preconditioner for the enthalpy equation is slightly more complicated because the temperature stays at a constant T_{pc} during phase change. The left preconditioner for the enthalpy is

$$\boldsymbol{L} = \begin{bmatrix} 1 + \frac{\theta_\ell H_{k\ell}}{\rho_k^0 C_k}\beta_k & -\frac{\theta_\ell H_{k\ell}}{\rho_k^0 C_k}\beta_k \\ -\frac{\theta_k H_{k\ell}}{\rho_\ell^0 C_\ell}\beta_\ell & 1 + \frac{\theta_k H_{k\ell}}{\rho_\ell^0 C_\ell}\beta_\ell \end{bmatrix}, \tag{7.23}$$

where

$$\beta = \begin{cases} 1/C_\ell & \text{if } h < h_\ell^{pc} \\ T_{pc} & \text{if } h_\ell^{pc} \leq h \leq h_k^{pc} \\ 1/C_k & \text{if } h > h_k^{pc} \end{cases} \tag{7.24}$$

in the case that phase ℓ has lower enthalpy than phase k during the phase change between phases ℓ and k.

7.2.3 Right preconditioner for temperature

The right side preconditioner for the energy equation extracted from the time derivative term and the diffusion term of (3.37). It is given as

$$V^n \sum_k \frac{\rho_k}{\rho_k^0 C_k^2}\widetilde{T}_k^L - \frac{\Delta t}{\rho_k^0 C_k}\sum_e \sum_k K_m \nabla \widetilde{T}_k^L \cdot \boldsymbol{n}_e A_e, \tag{7.25}$$

where C_k is the heat capacity of phase k, K_m is the mixture conductivity, and the normal component of the temperature gradient is calculated as in (7.19) with p replaced by T.

References

1. Ash, R.B., 1972, *Real Analysis and Probability*, Academic Press, Inc., Orlando, FL.

2. Brown, P. N. and Saad, Y.,, 1990, Hybrid Krylov methods for nonlinear systems of equations. *SIAM J. on Scientific and Statistical Computing* **11** (3), pp450-481.

3. Cummins, S.J. and Brackbill, J.U., 2002, An implicit particle-in-cell method for granular materials. *J. Comput. Phys.* **180** pp 506-548.

4. Drew, D. A. and Passman, S. L., 1999, Chapter 9 Ensemble averaging, *Theory of Multicomponent fluids*, Springer-Verlag. New York, NY.

5. Dukowicz, J.K. and Kodis, J.W., 1987, Accurate conservative remapping, *SIAM J. Sci. Stat. Comput.* **31**, pp 305.

6. Glimm, J., Saltz, D. and Sharp, D.H., 1999, Two-phase modeling of a fluid mixing layer. *J. Fluid Mech.* **378** pp 119-143.

7. Gurtin, M.E., 1981, Chapter 3, section 10, *An introduction to continuum mechanics.* Academic Press, London.

8. Kashiwa, B. A., Padial, N. T., Rauenzahn, R. M. and VanderHeyden, W. B., 1994, A Cell-Centered ICE Method for Multiphase Flow Simulations, FED-Vol. 185, Numerical Methods in Multiphase Flows, ASME, 185:159-176.

9. Kashiwa, B. A. and Rauenzahn, R. M., 1994 A multimaterial formalism, *Numerical Method in Multiphase flows* ed. by Crowe, C.C., Johnson, R., Prosperetti, A., Sommerfeld, M. and Tsuji, Y. FED-Vol 185, ASME New York.

10. Knoll, D.A., Keyes, D.E., 2004, Jacobian-free Newton-Krylov methods: a survey of approaches and applications, J. Comput. Phys. 193, 357-397.

11. Knoll, D.A., Kothe, D.B. and Lally, B., 1999, A new Non-linear solution method for phase-change problems. *Numerical Heat Transfer, Part B*, **35**, pp 439-459.

12. Liu, I., 2002, Chapter 2, *Continuum Mechanics*, Springer, New York.

13. Saltz, D., Lee W. and Hsiang, T.-R., 2000, Two-phase flow analysis of unstable fluid mixing in one-dimensional geometry. *Phys. Fluid.* **12**(10), pp 2461-2477.

14. VanderHeyden W. B. and Kashiwa, B.A., 1998, Compatible fluxes for van Leer Advection, *J. Comp. Phys.* **146**, pp 1-28.

15. van Leer, B., 1979, Toward the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method. *J. comp. Phys.* **32**. pp 101-136.

16. Yosida, K. 1966, *Functional analysis*, Springer-Verlag, New York.

17. Zhang, D. Z. and Prosperetti, A., 1994, Averaged equations fro inviscid disperse two-phase flow. *J. Fluid Mech.*, **267**, 185-219.

18. Zhang, D.Z. and Prosperetti, A., 1997, Momentum and energy equations for disperse two-phase flow and their closure for dilute suspensions. *int. J. Multiphase flow.* **23**(3) 425-453.

19. Zhang, D. Z., VanderHeyden, W.B., Zou, Q. and Padial-Collins, N. T., 2007, Pressure calculation in disperse and continuous multiphase flows. *Int. J. Multiphase Flow.* **33**, pp. 86-100.

Appendix: Probability and average

Following statistical mechanics, Zhang and Prosperetti (1994, 1997) derived averaged equations by averaging over an ensemble of two-phase flows. They used the Liouville equation in a phase space comprised of the positions and velocities of particles. For the potential and Stokes flows they treated, the motion of the continuous fluid is uniquely determined by the motions and positions of the particles; therefore the continuous fluid does not possess additional degrees of freedom, and the phase space has a finite dimension. This is no longer true for flows with finite Reynolds number, in which the degrees of freedom of the continuous fluid are infinite. Thus the concept of probability defined in a finite-dimensional phase space needs to be extended to handle these systems. For such an extension, we note that the probability of finding flows in a given subset of the ensemble is defined by the nature of the physics involved, and is independent of the parameters, or degrees of freedom, that we choose to describe them. This notion of parameter independence is similar to the notion of coordinate system independence in describing physical systems; different coordinate systems can be used to describe the same set of flows. In different descriptions of the flows, the phase space and the probability density are different, but the probability of finding flows belonging to the subset of the ensemble is independent of the description. The probability density defined in a phase space is merely a representation of the probability defined by the physical process. This description-independent probability can be used to treat systems with finite or infinite degrees of freedom, because degrees of freedom, finite or infinite, are merely descriptions of the system. This type of probability is common in real analysis and modern probability the-

ory. Drew and Passman (1999) used this probability as a conceptual starting point to derive averaged equations for multiphase flows.

Following Drew and Passman, we now introduce a probability $\mathcal{P}$ defined on a collection of subsets of the flows in the ensemble Ω. A subset in the collection is called an event in probability theory. The probability is a set function that maps an event (a subset in the collection), to a real number between 0 and 1. To ensure that such a probability is well defined and has the properties with which we are already familiar, there are certain conditions that need to be satisfied by the set function and by the collection of the subsets. Almost all physical systems of interest satisfy these conditions; therefore we do not list them here. Readers interested in more details are referred to textbooks on real analysis and probability theory, (e.g. Ash 1972). The focus of this Appendix is the connection between this probability and the probability density defined in the phase space used by Zhang and Prosperetti (1994, 1997) in the derivation of the averaged equations. This connection is particularly useful because the probability defined on the collection of subsets is quite abstract and difficult to manipulate. The probability defined in the phase space, although not as general, is easy to apply. For instance, the small particle approximation (Zhang and Prosperetti 1994, 1997) can only be obtained by using the probability density defined on particle configurations as derived in the following.

The average of function, f, which depends on a spatial position $\boldsymbol{x}$, time t and flow $\mathcal{F}$ in the ensemble, is the probability integral (associated with the probability measure) over all possible flows in the ensemble, and is denoted as $\int f(\boldsymbol{x}, t, \mathcal{F}) d\mathcal{P}$. Here, we assume that all the functions of interest satisfy the integrability condition under the probability $\mathcal{P}$ (Yosida, 1966, Ash 1972). For such a probability to be useful we further assume that differentiation with respect to both position $\boldsymbol{x}$ and time t can be exchanged freely with the probability integral for the functions of interest.

For a system with N particles at time t the probability density of finding a particle

configuration $\mathcal{C}^N = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N; \boldsymbol{v}_1, \cdots, \boldsymbol{v}_N\}$ in which a particle α is at a specified position $\boldsymbol{x}_\alpha$ with a specified velocity $\boldsymbol{v}_\alpha$ for $\alpha = 1, \cdots, N$ is related to the probability $\mathcal{P}$ as follows.

$$P(\mathcal{C}^N, t) = \int \delta[\boldsymbol{x}_1 - \boldsymbol{y}_1(\mathcal{F}, t)] \cdots \delta[\boldsymbol{x}_N - \boldsymbol{y}_N(\mathcal{F}, t)] \\ \delta[\boldsymbol{v}_1 - \boldsymbol{w}_1(\mathcal{F}, t)] \cdots \delta[\boldsymbol{v}_N - \boldsymbol{w}_N(\mathcal{F}, t)] d\mathcal{P}, \tag{A.1}$$

where $\boldsymbol{y}_\alpha(\mathcal{F}, t)$ and $\boldsymbol{w}_\alpha(\mathcal{F}, t)$ are the position and velocity of particle α in flow $\mathcal{F}$ at time t. In this definition the δ-functions select the flows with configuration $\mathcal{C}^N$ from the ensemble; and the integral averages over all the flows satisfying the configuration. The conditional average $\overline{q}(\boldsymbol{x}, t|\mathcal{C}^N)$ of a quantity $q(\boldsymbol{x}, t, \mathcal{F})$ given the configuration $\mathcal{C}^N$ can be calculated by averaging over all flows satisfying the configuration as

$$P(\mathcal{C}^N, t)\overline{q}(\boldsymbol{x}, t|\mathcal{C}^N) = \int q(\boldsymbol{x}, t, \mathcal{F}) \delta[\boldsymbol{x}_1 - \boldsymbol{y}_1(\mathcal{F}, t)] \cdots \delta[\boldsymbol{x}_N - \boldsymbol{y}_N(\mathcal{F}, t)] \\ \delta[\boldsymbol{v}_1 - \boldsymbol{w}_1(\mathcal{F}, t)] \cdots \delta[\boldsymbol{v}_N - \boldsymbol{w}_N(\mathcal{F}, t)] d\mathcal{P}. \tag{A.2}$$

In particular, the conditionally averaged acceleration $\overline{\dot{\boldsymbol{v}}}_\alpha$ of particle α can be calculated by

$$P(\mathcal{C}^N, t)\overline{\boldsymbol{v}}_\alpha(t|\mathcal{C}^N) = \int \dot{\boldsymbol{w}}_\alpha(\mathcal{F}, t) \delta[\boldsymbol{x}_1 - \boldsymbol{y}_1(\mathcal{F}, t)] \cdots \delta[\boldsymbol{x}_N - \boldsymbol{y}_N(\mathcal{F}, t)] \\ \delta[\boldsymbol{v}_1 - \boldsymbol{w}_1(\mathcal{F}, t)] \cdots \delta[\boldsymbol{v}_N - \boldsymbol{w}_N(\mathcal{F}, t)] d\mathcal{P}. \tag{A.3}$$

By differentiating (A.1) with respect to time t and then using (A.2), we find a generalized Liouville equation,

$$\frac{\partial P}{\partial t} + \sum_{\alpha=1}^{N} [\nabla_{\boldsymbol{x}_\alpha} \cdot (\boldsymbol{v}_\alpha P) + \nabla_{\boldsymbol{v}_\alpha} \cdot (\overline{\dot{\boldsymbol{v}}}_\alpha P)] = 0, \tag{A.4}$$

where we have used $\nabla_{\boldsymbol{y}_\alpha} \delta(\boldsymbol{x}_\alpha - \boldsymbol{y}_\alpha) = -\nabla_{\boldsymbol{x}_\alpha} \delta(\boldsymbol{x}_\alpha - \boldsymbol{y}_\alpha)$ and $\nabla_{\boldsymbol{w}_\alpha} \delta(\boldsymbol{v}_\alpha - \boldsymbol{w}_\alpha) = -\nabla_{\boldsymbol{v}_\alpha} \delta(\boldsymbol{v}_\alpha - \boldsymbol{w}_\alpha)$, and exchanged differentiation with the probability integration. For systems that can be uniquely described by the particle configuration $\mathcal{C}^N$, the average sign (over-bar) for the acceleration $\overline{\dot{\boldsymbol{v}}}_\alpha$ is not necessary ($\overline{\dot{\boldsymbol{v}}}_\alpha = \dot{\boldsymbol{w}}_\alpha$), and equation (A.4) becomes the Liouville equation.

The probability density defined in (A.1) is for distinguishable particles, while the probability density P_{zp} used by Zhang and Prosperetti (1994, 1997) is for indistinguishable particles.

These two probabilities are connected simply by $P_{zp} = \sum P(\mathcal{C}^N, t)$, where the summation is over all $N!$ possible permutations of particle numberings. Therefore the probability density defined in (A.1) normalizes to one, while P_{zp} normalizes to $N!$. Using this connection, one can show that generalized Liouville equation (A.4) is also satisfied by P_{zp}. This result implies that probability used here is consistent with the probability used by Zhang and Prosperetti (1994, 1997) for the cases they treated.

Because we have derived the generalized Liouville equation (A.4) and the transport equation (2.3) without referring to the phase space, the equations obtained by Zhang and Prosperetti (1994, 1997) can be used in disperse multiphase flows with finite Reynolds numbers.