

Article

An ALE Framework with an HLLC-2D Riemann Solver for Reactive Gas–Particle Flows

Jianqiao Zhang ¹, Xianggui Li ² and Wei Yan ^{1,*}¹ School of Mathematics, Jilin University, Changchun 130012, China; jqzhang22@mails.jlu.edu.cn² School of Applied Science, Beijing Information Science and Technology University, Beijing 100101, China; lixg@bistu.edu.cn

* Correspondence: wyanmath@jlu.edu.cn

Abstract

We propose a coupled gas–particle two-phase model for particle transport in a compressible carrier gas with interphase momentum and energy exchange, and we incorporate a diffusion-based mechanism to represent gas–particle reactions. The governing equations are discretized in an Arbitrary Lagrangian–Eulerian (ALE) finite-volume framework using an HLLC-type two-dimensional Riemann solver (HLLC-2D). The solver employs a nodal-conservation construction that enforces consistency between numerical fluxes and nodal contact velocities, which helps reduce spurious oscillations near discontinuities on moving meshes. In addition, a particle-search-based Courant–Friedrichs–Lewy (CFL)-like time-step restriction is introduced to enhance robustness in coupled simulations. Numerical tests are presented to assess the method and to illustrate particle-induced modifications of wave dynamics, as well as reaction-driven variations in velocity and temperature fields.

Keywords: two-phase flow; gas–particle interaction; ALE framework; HLLC-2D Riemann solver

MSC: 65M08; 76M12; 76T25; 76T15

1. Introduction

The modeling and numerical simulation of gas–particle multiphase flows in compressible media have long been an important topic in computational fluid dynamics. Such problems arise in a wide range of engineering and natural applications, including internal flows in solid rocket motors (nozzle problems), combustion in engine chambers, plume dynamics of ballistic weapons, and volcanic eruptions. In these scenarios, particles are entrained by the background flow and may also act as reactive constituents, leading to complex two-way interactions with the surrounding gas.

A common modeling strategy employs the compressible Euler equations for the carrier gas, augmented by interphase momentum and energy exchange terms that represent drag and heat transfer between the gas and particles. These interphase couplings typically enter the Euler system as source terms, while the particle phase is described by additional equations whose specific form depends on the particle loading regime (e.g., dilute versus dense). The resulting coupled system provides a unified framework for the evolution of both phases. Furthermore, particles may undergo chemical reactions during transport and generate new species. Such reactions are highly sensitive to local thermodynamic conditions (e.g., high temperature, high pressure, and gas reactivity), which increases model complexity and requires careful formulation of the reactive source mechanisms.



Academic Editor: Efstratios Tzirtzilakis

Received: 29 January 2026

Revised: 14 February 2026

Accepted: 17 February 2026

Published: 22 February 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

From the numerical viewpoint, solutions of the compressible Euler equations generally contain shock waves and contact discontinuities, whose accurate and stable resolution remains a central challenge in computational mathematics. The presence of a particle phase further complicates the problem: stiff or strongly varying source terms associated with interphase exchange may destabilize the flow solver and amplify spurious oscillations near discontinuities. Consequently, numerical methods for gas–particle systems must balance physical fidelity, conservation, and robustness, particularly in the presence of strong nonlinearities and discontinuous solutions.

The nomenclature “two-phase flow” was first introduced in the literature [1]. During the formative period of this topic, systematic research predominantly centered on experimental investigations and phenomenological characterization. Guided by these experimental findings, researchers progressively formulated physical frameworks and governing equations for two-phase flow problems [2–6], including the work of Campbell et al. [7], who conducted a series of studies on particle dynamics.

Through various experimental configurations, researchers have carried out systematic observations and analyses of transport dynamics and interphase interactions in shock-induced ejecta and gas–particle mixing flows. Buttler et al. [8] employed photonic Doppler velocimetry (PDV) to quantify the velocity evolution of metal ejecta, and subsequent developments in PDV-based diagnostics have enabled improved recovery of ejecta velocity and size distributions [9]. Chemical reactions between phases in reactive gases constitute another important aspect of these flows. Buttler et al. [10] investigated the transport and breakup of reactive cerium ejecta in reactive gases, and more recent numerical studies have further examined solid cerium ejecta transporting in vacuum as well as in non-reactive and reactive gas environments [11].

However, due to the intrinsic complexity of gas–particle systems and the high cost of experimental investigations, the development of physical models and numerical methods has increasingly become a primary research focus, both as predictive tools and as complementary approaches for interpreting and validating experimental observations. Recent reviews have emphasized the multi-scale challenges in high-speed compressible particle-laden flows, including shock–particle interactions and interphase coupling mechanisms [12]. As representative numerical approaches, smoothed particle hydrodynamics (SPH) has been used to simulate shock-induced ejecta formation and microjetting from shocked metal surfaces under extreme loading [13]. A particle model, referred to as multi-phase particle-in-cell (MP-PIC), was introduced in [14–16] and has continued to be extended toward broader compressible regimes; for example, compressible MP-PIC (CMP-PIC) formulations have been proposed to improve applicability across flow patterns [17], and related developments have been reported in recent studies [18]. MP-PIC has also been applied to shock interaction with particle curtains [19]. In addition, ejecta source and transport modeling capabilities have been developed in the FLAG hydrocode framework [20]. Chertock et al. [21] introduced a low-dissipative hybrid finite-volume–particle method for dusty gas flows. In [22], we investigated a discontinuous Galerkin (DG) scheme on moving meshes for the ejecta-mixing equations.

While the aforementioned studies have significantly advanced the understanding of gas–particle interactions, several limitations remain and motivate further refinement. Two issues are particularly relevant to coupled, shock-dominated regimes. First, in many particle-trajectory-focused studies, comparatively less emphasis is placed on accurately resolving the background compressible flow; consequently, the fluid solvers adopted in practice may exhibit insufficient resolution and nonphysical oscillations in the presence of shock waves, and quantitative validation against reference solutions or experiments is often limited. Second, energy exchange associated with temperature variation and chemical

reactions is sometimes neglected or treated in a simplified manner, and numerical tests frequently provide insufficient diagnostics on temperature evolution, which restricts the reliable computation and assessment of the gas internal energy budget.

Building upon previous research, we develop a coupled gas–particle trajectory transport model for particle motion in compressible gas, with explicit attention to interphase interactions and energy exchange. The model accounts for three primary physical processes: (i) drag induced by viscosity and relative velocity, which governs interphase momentum exchange; (ii) heat transfer driven by temperature differences; and (iii) reaction-induced energy variation modeled through a diffusion-based framework, without imposing temporal consistency between reactions and fluid motion. By emphasizing gas–particle coupling, the model further investigates how these effects modify the background compressible flow. Notably, particle loading introduces additional complexity in shock dynamics and yields nontrivial variations in both flow motion and energy distribution, which places stringent demands on the accuracy and stability of the numerical discretization.

To address these challenges, we propose a cell-centered Arbitrary Lagrangian–Eulerian (ALE) finite-volume framework for reactive gas–particle flows, combining a coupled particle-trajectory transport model with interphase drag/heat exchange and a diffusion-based reaction treatment. The gas-phase equations are discretized on moving meshes using an HLLC-2D nodal Riemann solver, and the time step is further constrained by a particle-search-based CFL-like condition to ensure consistent particle transport and coupling. The main novelty is a genuinely two-dimensional, node-compatible flux construction: a nodal contact velocity is determined from local conservation and used to build discontinuous edge fluxes, which improves robustness on moving meshes and mitigates shock instabilities observed in ALE schemes based on one-dimensional Riemann solvers. Numerical tests are presented to assess accuracy and stability and to quantify particle-induced modifications of the carrier-gas dynamics.

The rest of the paper is organized as follows. In Section 2, we introduce the coupled gas–particle trajectory transport model with chemical reactions between phases. A cell-centered ALE scheme and a two-dimensional nodal solver are presented in Section 3. In Section 4, we describe the time discretization and a particle-searching strategy that provides a CFL-like condition to restrict the time step. In Section 5, we present numerical tests, analyze the results, and draw conclusions.

2. Particle Trajectory Model with Enthalpy

2.1. Problem Statement, Notation and Assumptions

Consider the following problem in a region Ω of two-dimensional space that multiple particles are mixed in the fluid (the total volume is less than a certain percentage). Regard each particle as an individual, interacting with the fluid separately and research the motion of each particle independently. Since our particle concentration is less than a certain ratio, the particle flow is considered to be dilute particle flow, so it is assumed that particles do not affect each other. And the fluid has initial density, velocity, pressure and interacts with particles to exchange momentum and energy at the same time.

In the following, the subscript p denotes particle-phase quantities, while the subscript g denotes gas-phase quantities. Let $\rho_g, \mathbf{v}_g = (v_{gx}, v_{gy}), E_g, \mu_g, T_g, C_g$ denote the gas density, velocity, energy, viscosity, thermal conductivity, respectively; P denotes the pressure of the fluid. $\mathbf{u}_p = (u_{px}, u_{py}), \rho_p, \mathbf{v}_p = (v_{px}, v_{py}), T_p, c_p, r_p$ denote the coordinate, density, velocity, temperature, specific heat capacity and radius of particle, respectively. In particular, if there are multiple particles in the region Ω with different physical quantities, we can additionally use a subscript m to identify them. Namely $\mathbf{u}_{pm}, \rho_{pm}, \mathbf{v}_{pm}, T_{pm}, c_{pm}, r_{pm}$ denote the variable

of the different particles, in which $m = 1, \dots, m(\Omega)$, $m(\Omega)$ is the total number of particles in the region Ω .

To investigate the interactions between two phases, the force analysis of the particles in the system is essential. During the motion, particles are subject to gravity, drag, differential pressure, Basset force, Magnus force, Saffman force, etc. In general, not all of these forces are equally important. It is necessary to compare the magnitude of the above forces in order to estimate the relative importance of the various forces. Under normal conditions, other forces are several orders of magnitude smaller than the drag. So for the convenience of analysis, above forces except drag are not considered in the following model.

In addition to the force analysis, the exchange of heat between the particles and the fluid is also significant. In general, interphase heat transfer may involve conduction, convection, radiation, and reaction-induced energy exchange. In this work, the interphase heat exchange is modeled by the convective heat-transfer law $hA(T_g - T_p)$; the interfacial conductive contribution is effectively embedded in the coefficient h , and radiative transfer is neglected since it remains much smaller than convection over the temperature range considered.

Throughout, give the following assumptions:

- (1) Consider ideal compressible gas with ideal gas equation of state $P = (\gamma - 1)\rho\epsilon$, where γ is the adiabatic index.
- (2) Viscosity and heat transfer are only considered in interactions.
- (3) The particles are ideal round or spherical, uniform in diameter.
- (4) There are no collisions and interactions among particles.
- (5) The reaction between the ejecta and fluid is homogeneous which only depends on the diffusion efficiency K .

2.2. Governing Equations

According to the above assumptions, we introduce the equation of the system.

For the fluid phase, the equations for two-dimensional compressible particle trajectory model with enthalpy are as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} + \frac{\partial \mathbf{G}(\mathbf{U})}{\partial y} = \mathbf{S}_p, \quad (1)$$

where the state vector and flux vectors are

$$\mathbf{U} = \begin{bmatrix} \rho_g \\ \rho_g v_{gx} \\ \rho_g v_{gy} \\ \rho_g E_g \end{bmatrix}, \quad \mathbf{F}(\mathbf{U}) = \begin{bmatrix} \rho_g v_{gx} \\ P + \rho_g v_{gx}^2 \\ \rho_g v_{gx} v_{gy} \\ \rho_g E_g v_{gx} + P v_{gx} \end{bmatrix}, \quad \mathbf{G}(\mathbf{U}) = \begin{bmatrix} \rho_g v_{gy} \\ \rho_g v_{gx} v_{gy} \\ \rho_g v_{gy}^2 + P \\ \rho_g E_g v_{gy} + P v_{gy} \end{bmatrix}. \quad (2)$$

For the particle phase, each particle has:

$$\begin{aligned} \frac{\partial \mathbf{u}_p}{\partial t} &= \mathbf{v}_p, \\ \frac{\partial \mathbf{v}_p}{\partial t} &= \mathbf{F}_p, \\ \pi r_p^2 \rho_p c_p \frac{\partial T_p}{\partial t} &= Q_{pr} - Q_{pc}, \end{aligned} \quad (3)$$

where the $\mathbf{F}_p$ and Q_{pr}, Q_{pc} indicate the interactions between the particle phase and fluid phase.

And $\mathbf{S}_p$ is the source term between the flow and particles.

$$\mathbf{S}_p = \begin{bmatrix} \dot{M} \\ F_{px} \\ F_{py} \\ E_p \end{bmatrix},$$

In the above model, the motion of every particles with index m will be calculated respectively; actually, we have

$$\mathbf{F}_p = \begin{bmatrix} F_{px} \\ F_{py} \end{bmatrix} = - \sum_{m=1}^{m(\Omega)} \mathbf{F}_{d_m} - \dot{M} \mathbf{v}_g,$$

$$E_p = - \sum_{m=1}^{m(\Omega)} E_{p_m} = \sum_{m=1}^{m(\Omega)} \mathbf{F}_{d_m} \cdot (\mathbf{v}_g - \mathbf{v}_{p_m}) - Q_{p_{rm}} + Q_{p_{cm}}.$$

where $\mathbf{F}_p$ is the force source item coming from the drag $\mathbf{F}_{d_m}$ between phases and momentum term $\dot{M} \mathbf{v}_g$ carried away by the gas in chemical reactions. E_p is the exchange of energy including the heat ($Q_{pr} - Q_{pc}$) and the kinetic energy $\mathbf{F}_d \cdot \mathbf{v}_p$ from drag work. $\dot{M}$ is the change ratio of mass because of the reaction between phases.

Then give the specific definition of drag source item $\mathbf{F}_{d_m}$. The force $\mathbf{F}_{d_m}$ comes from the drag due to friction. There are two drag models usually used during computing which are the Stoke's model and Crowe's model.

For the Stoke's model [23], the following formulation is applied:

$$\mathbf{F}_{d_m} = 6\pi r_p \mu_g (\mathbf{v}_g - \mathbf{v}_{p_m}),$$

where r_p is the radius of the particles and μ_g is the gas viscosity.

For the Crowe's model [24], we have

$$\mathbf{F}_{d_m} = \frac{1}{2} C_D K_{p_m} \rho_g |\mathbf{v}_g - \mathbf{v}_{p_m}| (\mathbf{v}_g - \mathbf{v}_{p_m}),$$

where C_D is the drag coefficient and K_{p_m} is the upwind area. Drag coefficient C_D is related to the Reynolds number Re , the following definition is applied [25,26]:

$$C_D = \frac{24}{Re} f_{drag}(Re). \quad (4)$$

In formulation (4), $f_{drag}(Re)$ depends on the Reynolds number Re :

$$Re = \frac{2\rho_g r_{p_m} |\mathbf{v}_g - \mathbf{v}_{p_m}|}{\mu_g},$$

and

$$C_D = \begin{cases} \frac{24}{Re}, & Re < 0.2, \\ \frac{24}{Re} (1 + 0.15 Re^{0.687}), & 0.2 \leq Re \leq 800, \\ 0.5, & Re > 800. \end{cases}$$

The calculation method of C_D is not same in different research. The above definition is given by Wang et al. in [26].

2.3. Reactions and Energy

The heat source item $Q_p(Q_{pc}, Q_{pr})$ comes from the convection heat transfer [27] and reaction between phases, respectively. First analyze the procedure of reaction and suppose there is an ejecta with radius r_p . Typically, the reaction proceeds uniformly from the surface of the particles towards the interior, and the reaction rate depends on the diffusion

efficiency D_2 . Reactions are usually accompanied by heat absorption and release, and this exchange of heat can affect the physical state of the phases. For convenience, we assume that the particles remain in the solid state. Take the reaction between Ce and D_2 as an example to illustrate this. The chemical reaction equation is: $Ce + D_2 \rightarrow CeD_2 + \Delta H$, where $\Delta H = -206 \text{ kJ/mol}$. The following Figure 1 gives the details:

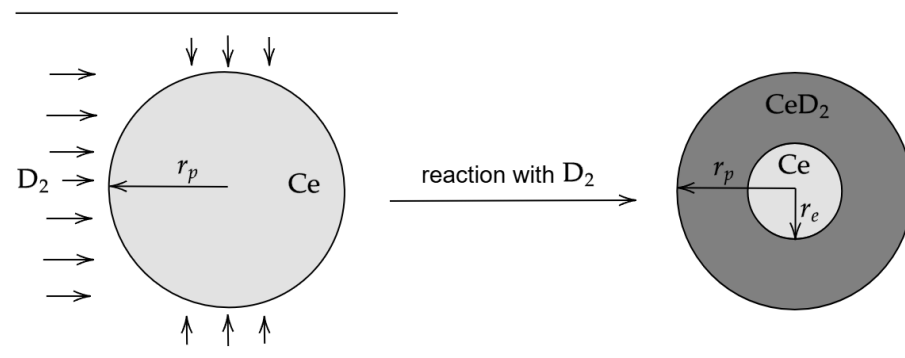


Figure 1. Reaction between Ce and D_2 .

We employ a diffusion model to characterize the spread of the reaction procedure. We have:

$$r_e = r_p - \sqrt{Kt}$$

so the reaction rate has:

$$\frac{dM_p}{dt} = -\pi\rho_p r_e \sqrt{\frac{K}{t}}.$$

The reason of using the time index t_1 there but not the index t is that the reaction is not always with the motion of fluid except under the certain conditions of pressure and temperature.

According to the heat ΔH , we have the energy release:

$$Q_{pr} = \frac{dM_p}{dt} \Delta H.$$

Because of the difference of temperature between a particle and surrounding fluid, there is a convection heat transfer Q_{pc} , the total energy exchange is:

$$Q_p = Q_{pr} - Q_{pc}.$$

And Q_{pc} is:

$$Q_{pc} = \frac{2\mu_g}{Pr} C_{pg} \pi r_p (T_g - T_p) Nu_p,$$

where C_{pg} is constant-pressure specific heat capacity, Nu_{pm} is Nusselt numbers, Pr is Prandtl numbers of flow, of which specific definitions are as follows:

$$Pr = \frac{4\gamma}{9\gamma - 5},$$

and

$$Nu_p = 2 + 0.6Pr^{1/3}Re^{1/2}.$$

Based on Equations (1) and (3), we need to prove that the above gas–particle two phase system satisfies conservation law. It is obvious that the conservation of momentum because the drag between the particle and the fluid is a pair of interacting forces, namely the right items of the second Equations (1) and (3). The item $F_{pm}v_{pm}$ is the energy of the

fluid, which is converted into the kinetic energy of the particle, and the convection heat transfer energy Q_{p_m} is generated by the temperature difference.

3. Numerical Scheme

In this section, we construct a numerical scheme using HLLC-type Riemann solver based on the nodal conservation to solve the system (1) and (3).

3.1. ALE Scheme

For the subsequent discretization process, it is convenient to express Equation (1) in the form of a moving control volume formulation:

$$\frac{d}{dt} \int_{\Omega} \mathbf{U} dx dy + \int_{\partial\Omega} [(\mathbf{F}, \mathbf{G})\mathbf{N} - (\mathbf{w} \cdot \mathbf{N})\mathbf{U}] dl = \int_{\Omega} \mathbf{S}_p dx dy, \quad (5)$$

$\mathbf{w}$ denotes the moving velocity of the control volume Ω while $\mathbf{N}$ denotes the unit outward normal direction along the boundary of Ω . When $\mathbf{w} = \mathbf{u}$, the system simplifies into a Lagrangian formulation, and when $\mathbf{w} = \mathbf{0}$, it takes on an Eulerian form.

It is important to note that in Equation (5), the integration domain Ω is related to time t ; that is, $\Omega = \Omega(t)$. For convenience, we denote it as Ω .

3.1.1. Notations on a Generic Polygonal Grid

Every cell Ω_c within the mesh is assigned a distinct index represented by c . To refer to an adjacent cell Ω_d that shares a common edge with Ω_c , we employ the subscript d . The edge where two units intersect is defined as $c \cap d$ or denoted as k , as shown in Figure 2. $F(c)$ denotes the set of adjacent cells surrounding the cell Ω_c . $Q(c)$ is the set of all vertices of a cell Ω_c . Each vertex within the mesh is allocated in the index q , and $C(q)$ and $K(q)$ are defined as the sets of cells and edges respectively surrounding the vertex q , i.e.,

$$C(q) = \{\Omega_c : \text{cells surrounding the vertex } q\},$$

$$K(q) = \{k : \text{edges surrounding the vertex } q\}.$$

The physical quantities of fluid for example the density ρ_{g_c} , pressure P_{g_c} , velocity $\mathbf{v}_{g_c}$, energy E_{g_c}, e_{g_c} are defined in the center of Ω_c . The grid moving velocity, denoted as $\mathbf{w}_q$, is defined at the node q . Let $\mathbf{N}_d^c = ((n_x)_d^c, (n_y)_d^c)$ represents the unit normal vector of cell Ω_c along edge $c \cap d$, with the subscript d and the superscript c indicating that the vector direction originates from Ω_c and extends towards Ω_d .

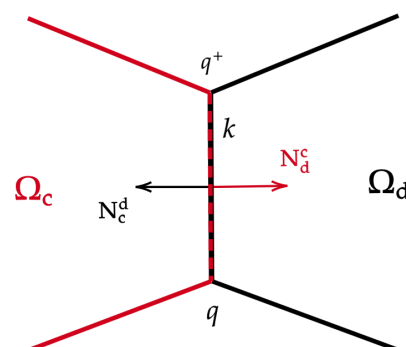


Figure 2. Notations on the mesh edge.

Based on the edge's normal direction between two adjacent cells, we utilize indices L and R to represent the states on the left and right sides of an edge, respectively. For instance, $\mathbf{U}_L$ and $\mathbf{U}_R$ are state vectors in the left and right cells of the edge $c \cap d$.

To compute integration of the variable $\mathbf{F}$ across edge $[M_a, M_{a+1}]$, denote by $\mathbf{F}_{a,a+1}$ the fluxes on the face, $L_{a,a+1}$ is the length of $[M_a, M_{a+1}]$, we have

$$L_{a,a+1} \mathbf{F}_{a,a+1} \cdot \mathbf{N}_{a,a+1} = \int_{M_a}^{M_{a+1}} \mathbf{F} \cdot \mathbf{N} dl. \quad (6)$$

Using the above transformation, we discretize Equation (5) on an unstructured mesh. Define cell-centered state variable $\mathbf{U}_c$ and resource term $\mathbf{S}_{p_c}$, then we have:

$$\begin{aligned} \frac{d}{dt} |\Omega_c| \mathbf{U}_c &= |\Omega_c| \mathbf{S}_{p_c} - \sum_d \int_d (\mathcal{T}_d^c)^{-1} [\mathbf{F}(\mathcal{T}_d^c \mathbf{U}) - (\mathbf{w}_d^c \cdot \mathbf{N}_d^c) \mathcal{T}_d^c \mathbf{U}] dl \\ &= |\Omega_c| \mathbf{S}_{p_c} - \sum_d L_d^c (\mathcal{T}_d^c)^{-1} \mathbf{F}_d^c(\mathbf{w}_d^c \cdot \mathbf{N}_d^c, \mathcal{T}_d^c \mathbf{U}_c, \mathcal{T}_d^c \mathbf{U}_d), \end{aligned} \quad (7)$$

where $|\Omega_c|$ represent the volume of cell Ω_c . $\mathbf{w}_d^c$ is the moving velocity defined at the center of cell edge $c \cap d$. L_d^c is the length of the edge $c \cap d$. $\mathcal{T}_d^c$ and $(\mathcal{T}_d^c)^{-1}$ are the rotation matrix and its inverse as follows:

$$\mathcal{T}_d^c = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & (n_x)_d^c & (n_y)_d^c & 0 \\ 0 & -(n_y)_d^c & (n_x)_d^c & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (\mathcal{T}_d^c)^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & (n_x)_d^c & -(n_y)_d^c & 0 \\ 0 & (n_y)_d^c & (n_x)_d^c & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Then compute the Riemann problem to approximate the numerical flux $\mathbf{F}_d^c$ along the normal direction of edge $c \cap d$:

$$\begin{aligned} \mathbf{U}_t + \mathbf{F}_x &= \mathbf{S}_p, \\ \mathbf{U}(x, 0) &= \begin{cases} \mathbf{U}_L = \mathcal{T}_d^c \mathbf{U}_c, & x < 0, \\ \mathbf{U}_R = \mathcal{T}_d^c \mathbf{U}_d, & x > 0, \end{cases} \end{aligned} \quad (8)$$

and the numerical flux can be represented as follows:

$$\mathbf{F}_d^c(w, \mathbf{U}_L, \mathbf{U}_R) = \mathbf{F}^w - w \mathbf{U}^w, \quad (9)$$

where $w = \mathbf{w}_d^c \cdot \mathbf{N}_d^c$ is the normal projection of the moving velocity $\mathbf{w}_d^c$, $\mathbf{U}^w$ and $\mathbf{F}^w$ are the projection of the state and flux functions onto the direction $x/t = w$, respectively.

3.1.2. Flux Formula of HLLC-2D on Moving Mesh

We use the two-dimensional Riemann solver HLLC-2D method proposed in [28] to evaluate the numerical flux. The construction of this method hinges on two important points. Firstly, it involves a nodal solver that relies on the vertex q with state vectors $\{\mathbf{U}_c, c \in C(q)\}$ assigned to neighbor cells. Then the nodal contact velocity $\mathbf{u}_q^*$ is computed through this solver. Secondly, it involves the utilization of the HLLC-2D solver, an expanded scheme of the classical one-dimensional HLLC Riemann solver. In Riemann problem (8), we have the nodal contact velocity $\mathbf{u}_q^*$, as well as the state vectors $\mathbf{U}_c$ and $\mathbf{U}_d$, which are on the sides of edge $c \cap d$. Through solving the Riemann problem, we could get the approximate numerical flux $\mathbf{F}_c^d$ and $\mathbf{F}_d^c$ and further solving Equation (7).

In the following evaluation, assume that the nodal contact velocity $\mathbf{u}_q^*$ has already been calculated, in alignment with the preceding chapter's description. From Equation (8), the initial values of the equation are:

$$\mathbf{U}_L = \mathcal{T}_d^c \mathbf{U}_c, \quad \mathbf{U}_R = \mathcal{T}_d^c \mathbf{U}_d,$$

and the fluxes are $\mathbf{F}_L = \mathbf{F}(\mathbf{U}_L)$, $\mathbf{F}_R = \mathbf{F}(\mathbf{U}_R)$.

In contrast to the classical one-dimensional Riemann solver, HLLC-2D method deviates by introducing an artificial contact wave speed denoted as $S_* = \mathbf{u}_q^* \cdot \mathbf{N}_d^c$, which is consistent with the nodal velocity $\mathbf{u}_q^*$.

Establish two velocities S_L and S_R and suppose $S_L \leq S_* \leq S_R$. As a result, the HLLC-2D solver comprises 4 constant state regions separated by S_L, S_* and S_R . Within these regions, the vector of conservation quantities is defined as follows:

$$\mathbf{U}^w = \begin{cases} \mathbf{U}_L, & \text{if } w \leq S_L, \\ \mathbf{U}_L^*, & \text{if } S_L < w \leq S_*, \\ \mathbf{U}_R^*, & \text{if } S_* < w \leq S_R, \\ \mathbf{U}_R, & \text{if } w > S_R, \end{cases} \quad (10)$$

where $\mathbf{U}_L^*, \mathbf{U}_R^*$ are the same as the classical HLLC solver. Within each state region, we introduce the two approximate fluxes which is from the different direction of the edge $c \cap d$:

$$\mathbf{F}_H^w = \begin{cases} \mathbf{F}_{H,1}, & \text{if } w \leq S_L, \\ \mathbf{F}_{H,2}, & \text{if } S_L < w \leq S_*, \\ \mathbf{F}_{H,3}, & \text{if } S_* < w \leq S_R, \\ \mathbf{F}_{H,4}, & \text{if } w > S_R, \end{cases} \quad (11)$$

where $H = L, R$, and the above flux scheme ensure strictly satisfies the Rankine–Hugoniot conditions.

According to the (9) and (11). Numerical fluxes encircling the vertex q on the moving mesh are as follows:

$$\mathbf{F}_{d,q}^c(w, \mathbf{U}_L, \mathbf{U}_R) = \mathbf{F}_L^w - w\mathbf{U}^w = \begin{cases} \mathbf{U}_L(u_L - w) + \mathbf{D}_L, & \text{if } w \leq S_L, \\ \mathbf{U}_L^*(S_* - w) + \mathbf{D}_L^*, & \text{if } S_L < w \leq S_*, \\ \mathbf{U}_R^*(S_* - w) + \mathbf{D}_L^*, & \text{if } S_* < w \leq S_R, \\ \mathbf{U}_R(u_R - w) + \mathbf{D}_R - \mathbf{D}_R^* + \mathbf{D}_L^*, & \text{if } w > S_R, \end{cases} \quad (12)$$

and

$$\mathbf{F}_{c,q}^d(w, \mathbf{U}_L, \mathbf{U}_R) = \mathbf{F}_R^w - w\mathbf{U}^w = \begin{cases} \mathbf{U}_L(u_L - w) + \mathbf{D}_L - \mathbf{D}_L^* + \mathbf{D}_R^*, & \text{if } w < S_L, \\ \mathbf{U}_L^*(S_* - w) + \mathbf{D}_R^*, & \text{if } S_L \leq w < S_*, \\ \mathbf{U}_R^*(S_* - w) + \mathbf{D}_R^*, & \text{if } S_* \leq w < S_R, \\ \mathbf{U}_R(u_R - w) + \mathbf{D}_R, & \text{if } w \geq S_R, \end{cases} \quad (13)$$

where $\mathbf{D}_H = (0, P, 0, Pu)^T$, $\mathbf{D}_H^* = (0, P_H^*, 0, P_H^* S_*)^T$ for $H = L, R$ are vectors, including the pressure and work terms on the ends and in the middle two regions, respectively. Pressures are

$$\begin{cases} P_L^* = p_L + \rho_L(S_L - u_L)(S_* - u_L), \\ P_R^* = p_R + \rho_R(S_R - u_R)(S_* - u_R). \end{cases} \quad (14)$$

An analogous derivation can be given at vertex q^+ , and flux $\mathbf{F}_{d,q^+}^c$ is evaluated. For the edge $c \cap d$, the final flux on it is:

$$\mathbf{F}_d^c(w, \mathcal{T}_d^c \mathbf{U}_c, \mathcal{T}_d^c \mathbf{U}_d) = \frac{1}{2} \left(\mathbf{F}_{d,q}^c(\mathbf{w}_q \cdot \mathbf{N}_d^c, \mathcal{T}_d^c \mathbf{U}_c, \mathcal{T}_d^c \mathbf{U}_d) + \mathbf{F}_{d,q^+}^c(\mathbf{w}_{q^+} \cdot \mathbf{N}_d^c, \mathcal{T}_d^c \mathbf{U}_c, \mathcal{T}_d^c \mathbf{U}_d) \right). \quad (15)$$

3.1.3. Nodal Solver

In Section 3.1.2, we have assumed that the nodal contact velocity is already calculated. And since $\mathbf{F}_d^c \neq \mathbf{F}_c^d$ may happen, the conservation in cells is not satisfied. Therefore we introduce a nodal solver that consistently evaluates the $\mathbf{u}_q^*$, ensuring the scheme preserves local conservation of mass, momentum and energy. This procedure can be derived from the global conservation within domain Ω . By maintaining the local nodal conservation including mass, momentum and total energy, contact velocity could be calculated. By using the pressure Equation (14), we have:

$$\sum_{k \in K(q)} L_k(\alpha_{L,k} + \alpha_{R,k}) [\mathbf{u}_q^* \cdot \mathbf{N}_k - v_k^*] \mathbf{N}_k = 0, \quad (16)$$

where

$$\alpha_{L,k} = -\rho_{L,k}(S_{L,k} - u_{L,k}), \quad \alpha_{R,k} = \rho_{R,k}(S_{R,k} - u_{R,k}), \quad (17)$$

and v_k^* represents the contact velocity in the classical one-dimensional HLLC Riemann solver for edge k

$$v_k^* = \frac{P_{L,k} - P_{R,k} + \alpha_{L,k}u_{L,k} + \alpha_{R,k}u_{R,k}}{\alpha_{L,k} + \alpha_{R,k}}. \quad (18)$$

Solve Equation (16) to calculate the contact velocity $\mathbf{u}_q^*$ at each vertex and we have

$$\mathbf{u}_q^* = M^{-1} \sum_{k \in K(q)} L_k(\alpha_{L,k} + \alpha_{R,k}) v_k^* \mathbf{N}_k, \quad (19)$$

where

$$M = \begin{bmatrix} \sum_{k \in K(q)} L_k(\alpha_{L,k} + \alpha_{R,k}) n_{x,k}^2 & \sum_{k \in K(q)} L_k(\alpha_{L,k} + \alpha_{R,k}) n_{x,k} n_{y,k} \\ \sum_{k \in K(q)} L_k(\alpha_{L,k} + \alpha_{R,k}) n_{x,k} n_{y,k} & \sum_{k \in K(q)} L_k(\alpha_{L,k} + \alpha_{R,k}) n_{y,k}^2 \end{bmatrix}.$$

3.1.4. Boundary Condition

In this ALE method, the conventional zero-order extrapolation method is employed to enforce boundary conditions. For velocity-based boundaries, there are two different cases:

- (1) When the three boundary nodes are not collinear, a straightforward extrapolation method is implemented.
- (2) When the three boundary nodes are collinear, we employ the mirror extrapolation method, which involves constructing a symmetrical point for each vertex connected to a boundary node.

The above method ensures the same scheme (19) as used for internal vertex when calculating $\mathbf{u}_q^*$.

4. Time Discretization and Particle Searching

We discretize in time for the system (7) that describes the evolution of the variables in the cell Ω_i . Suppose that there are $m(\Omega_i)$ particles in cell Ω_i , of which the index is $m = 1, \dots, m(\Omega_i)$, then we use the second-order Runge–Kutta method to obtain the following fully discretized system:

$$\hat{\mathbf{U}}_i^{n+1} = \frac{\Delta t^n}{|\Omega_i^{n+1}|} \sum_m \mathbf{S}_{p_i}^n + \frac{|\Omega_i^n|}{|\Omega_i^{n+1}|} \mathbf{U}_i^n - \frac{\Delta t^n}{|\Omega_i^{n+1}|} [\sum_d L_d^i(\mathcal{T}_d^i)^{-1} \mathbf{F}_d^i(\mathbf{w}_d^i \cdot \mathbf{N}_d^i, \mathcal{T}_d^i \mathbf{U}_i^n, \mathcal{T}_d^i \mathbf{U}_d^n)], \quad (20)$$

$$\bar{\mathbf{U}}_i^n = \frac{1}{2}(\mathbf{U}_i^n + \hat{\mathbf{U}}_i^{n+1}), \quad (21)$$

$$\mathbf{U}_i^{n+1} = \frac{\Delta t^n}{|\Omega_i^{n+1}|} \sum_m \mathbf{s}_{p_i}^n + \frac{|\Omega_i^n|}{|\Omega_i^{n+1}|} \mathbf{U}_i^n - \frac{\Delta t^n}{|\Omega_i^{n+1}|} \sum_d L_d^i (\mathcal{T}_d^i)^{-1} \mathbf{F}_d^i (\mathbf{w}_d^i \cdot \mathbf{N}_d^i, \mathcal{T}_d^i \bar{\mathbf{U}}_i^n, \mathcal{T}_d^i \bar{\mathbf{U}}_d^n), \quad (22)$$

and similarly

$$\begin{aligned} \hat{\mathbf{v}}_{p_m}^{n+1} &= \mathbf{v}_{p_m}^n + \frac{\Delta t^n}{\pi r_{p_m}^2 \rho_{p_m}} \mathbf{F}_{p_m}^n, \\ \hat{\mathbf{F}}_{p_m}^{n+1} &= \mathbf{F}_{p_m} \left(\frac{1}{2} (\hat{\mathbf{v}}_{p_m}^{n+1} + \mathbf{v}_{p_m}^n) \right), \\ \hat{T}_{p_m}^{n+1} &= T_{p_m}^n + \Delta t^n \frac{1}{\pi r_{p_m}^2 \rho_{p_m} c_{p_m}} Q_{p_m}^n, \\ \hat{Q}_{p_m}^{n+1} &= Q_{p_m} \left(\frac{1}{2} (T_{p_m}^n + \hat{T}_{p_m}^{n+1}) \right). \end{aligned}$$

so that

$$\begin{aligned} \mathbf{v}_{p_m}^{n+1} &= \mathbf{v}_{p_m}^n + \frac{\Delta t^n}{\pi r_{p_m}^2 \rho_{p_m}} \hat{\mathbf{F}}_{p_m}^{n+1}, \\ \mathbf{u}_{p_m}^{n+1} &= \mathbf{u}_{p_m}^n + \Delta t^n \hat{\mathbf{v}}_{p_m}^{n+1}, \\ T_{p_m}^{n+1} &= T_{p_m}^n + \Delta t^n \frac{1}{\pi r_{p_m}^2 \rho_{p_m} c_{p_m}} \hat{Q}_{p_m}^{n+1}. \end{aligned} \quad (23)$$

4.1. Particle Searching

Before computing the time step $\Delta t^n = t^{n+1} - t^n$, we need to give the process of searching the particles during a time step. In the numerical simulation, the interactions between the particles and flow are calculated in each cell instead of the whole space. That is, if consider the cell Ω_i , the momentum and energy exchanged between the two phases are that all the particles in cell Ω_i interact with the fluid in Ω_i . So before computing the physical properties, we should know which particles are in the cell. At the beginning of each time, the coordinates of particles are given. And after every time steps, we should search the particles and confirm the cell position of all particles. In our numerical simulations, we use the following procedure to locate particles:

- (1) Suppose that the coordinates of particles in the cell Ω_i at time t^n are known, namely $\mathbf{u}_m^n, m = 1, \dots, m(\Omega_i)$.
- (2) Solve Equation (23) and get the coordinate of particle $m, \mathbf{u}_m^{n+1}$, at time t^{n+1} .
- (3) Search the particle m in the updated geometrical characteristics of the cell. First determine whether the particle m is in the cell Ω_i . If so, proceed to step 5.
- (4) If not, search the particle m in the cells Ω_c which are the surrounding cells of Ω_i , $c = 1, \dots, n(i)$.
- (5) Update the position of particle m .

Using the above method, we could get the updated position of particle m under the condition that we give a CFL-like time step limitation to ensure the movement range of the particles. In Figure 3 we give the schematic diagrams of particle searching in unstructured mesh in Ω_i , and $n(i) = 6$ this moment:

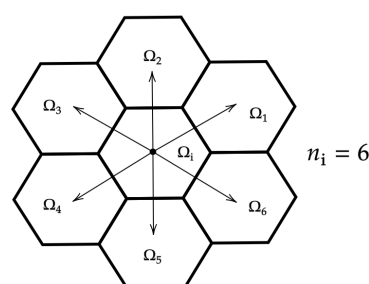


Figure 3. Searching the surrounding cells.

4.2. Time Step Limitation

To ensure stability under the explicit scheme, the following time step constraints must be satisfied:

- (1) An a priori CFL condition is required to ensure compliance with the entropy inequality.
- (2) The variation in cell volume is geometrically consistent with the motion of the mesh vertices.
- (3) Particles must be efficiently searched.

Assume that the physical properties and geometric characteristics of the cell Ω_i are known at the time t^n . We will compute the variables' value at time t^{n+1} .

4.2.1. CFL Condition

First we derive a CFL condition to ensure positive entropy production within cell Ω_i during the computation. At time t^n , we denote by λ_i^n the inner circle radius of the cell Ω_i . Then

$$\Delta t_e = C_e \min_{i=1,\dots,I} \frac{\lambda_i^n}{\sqrt{(u_{gi}^n)^2 + (v_{gi}^n)^2 + c_i^n}}, \quad (24)$$

where C_E is a strictly positive coefficient and c_i is the sound speed of the cell Ω_i . Numerically, the value of C_E between 0.3 and 0.4 is usually taken.

4.2.2. Criterion on the Variation of Volume

We also establish a criterion for the variation of cell volume. At time t^n , we can get the contact velocity of vertex, we have:

$$\Delta t_v = C_v \min_{i=1,\dots,I} \left\{ \frac{|A_i^n|}{|A_i'|} \right\}, \quad (25)$$

where C_v can be set 0.1 and

$$|A_i'| = \frac{1}{2} \sum_f L_f^c (\mathbf{u}_q^* + \mathbf{u}_{q+}^*) \cdot \mathbf{N}_f^c. \quad (26)$$

4.2.3. Particle Searching Limitation

Referring to Section 4.1, suppose that particle m is located in Ω_i ; we check the cell Ω_i and its surrounding cells Ω_c , $c = 1, \dots, n(i)$. Over t^n to t^{n+1} , we restrict particle motion to a local neighborhood to ensure efficient search. In Section 4.2.1, we use λ_k^n to denote the inner circle radius of the cell Ω_i and the cell Ω_c . We have:

$$\Delta t_p = C_p \min_m \min_{k=1,\dots,n(i)+1} \frac{\lambda_k^n}{|\mathbf{v}_{sum}^n|}, \quad (27)$$

where $n(m)$ is the number of cells containing and surrounding the particle, $\mathbf{v}_{sum}^n = \mathbf{v}_{pm}^n + \sum_{\Omega_i} \mathbf{u}_{qi}^n$ is the sum of velocity of particle m and the nodal velocities of cell Ω_i . In numerical simulation we take $C_p = 0.5$.

The proposed CFL-like restriction introduces a particle-based bound on the time step. While this adds an $O(N_p \cdot C_{local})$ particle contribution to the time-step evaluation beyond the standard CFL condition, it is primarily introduced to control the cost of particle localization. In the particle-trajectory model, particle localization is required at every step to deposit interphase source terms consistently; without localization, one cannot determine, at the discrete level, which control volume should receive each particle's contribution. If particles are allowed to overstep multiple cells within one step, the host-cell update may require

a global search whose cost is $O(N_p \cdot C_{\text{global}})$. By limiting the particle displacement, the host cell of each particle can be updated via a local neighbor check around its previous cell.

Finally, the next time step $\Delta t^n = t^{n+1} - t^n$ is given by

$$\Delta t^n = \min(\Delta t_e, \Delta t_v, \Delta t_p, 1.01 * \Delta t^{n-1}). \quad (28)$$

4.3. Algorithm for Full Discretization System

Procedure. At time $t = t^n$, the coupled update from t^n to t^{n+1} is carried out as follows:

1. Initialization. For each cell Ω_i ($i = 1, \dots, I$), initialize particle variables $\mathbf{u}_{p_m}^n, \rho_{p_m}^n, \mathbf{v}_{p_m}^n, T_{p_m}^n, r_{p_m}^n$ for $m = 1, \dots, m(\Omega_i)$; initialize fluid variables $\mathbf{v}_g^n, E_g^n, \rho_g^n, T_g^n$; and geometrical quantities $L_k^n, \mathbf{N}_k^n$ for $k \in K(q)$.
2. Nodal solver. Compute the contact velocity $\mathbf{u}_q^*$ at vertex q by solving the equations in Section 3.1.3.
3. Time-step limitation. Determine the time step Δt^n using Equation (28).
4. Grid-moving strategy (ALE). Use a grid-moving strategy (e.g., harmonic mapping) to update vertex coordinates and compute the nodal velocity

$$\mathbf{w}_q = \frac{\mathbf{x}_q^{n+1} - \mathbf{x}_q^n}{\Delta t^n}.$$

5. Interphase interaction. Compute interaction forces $\mathbf{F}_p$ and energy exchange E_p at time t^n .
6. HLLC-2D Riemann solver. For each edge $\Omega_i \cap \Omega_d$, compute the nodal fluxes $\mathbf{F}_{d,q}^i$ and $\mathbf{F}_{d,q^+}^i$, and take their average

$$\mathbf{F}_d^i = \frac{1}{2}(\mathbf{F}_{d,q}^i + \mathbf{F}_{d,q^+}^i).$$

7. Update flow variables. Update the cell-averaged flow variables in each cell Ω_i .
8. Update particle variables. Update particle variables using Equation (23), and determine particle positions at time t^{n+1} .
9. Loop. If the simulation continues, set $n \rightarrow n + 1$ and return to Step 1.

5. Numerical Tests and Results

In the following section, we present several examples to validate the above numerical scheme. We begin by several classical one-dimensional particle transport tests in quiescent fluid, in fluid under constant acceleration and in fluid under sinusoidal acceleration. The relationship between particle size and particle motion path is further explored in this part. Next we explore the motion of fluid in gas–particle Sod test. Finally, we give a two-dimensional practical test case about Pb plate ejecta motion under the high speed shock. For generality, we calculate these numerical tests on unstructured mesh.

5.1. Particle Transport Tests

In this test, we investigate the dynamics of particles immersed in a fluid across different flow conditions, a problem of fundamental importance in both engineering and natural applications.

5.1.1. Transport in a Quiescent Fluid

Consider the basic case of a particle subject only to drag. Initialized with a nonzero velocity at the fluid center, the particle exhibits a monotonically decreasing velocity, with deceleration gradually diminishing, as shown in Figure 4. At the same time, the velocity of the fluid reaches the peak rapidly within a short period and then decreases. The velocity

difference between the particle and fluid ultimately converges, contrary to the intuition that the fluid velocity keeps increasing until they become consistent.

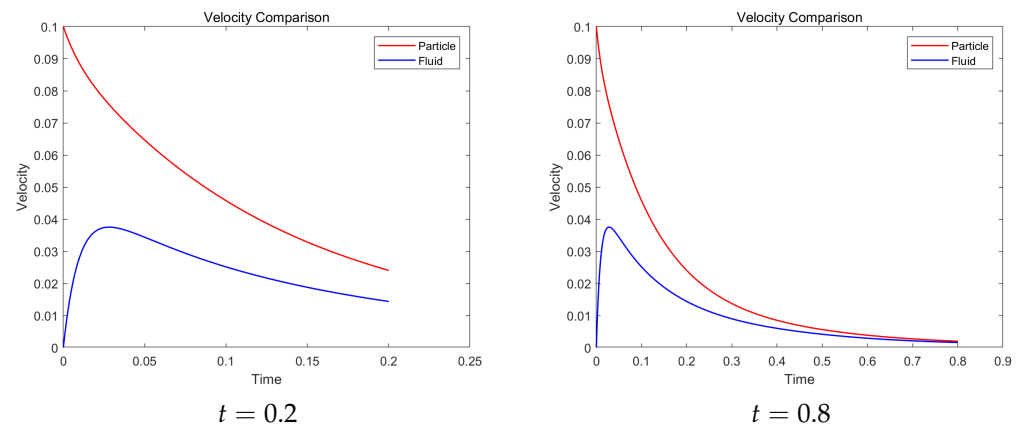


Figure 4. Drag model.

5.1.2. Motion Under Constant Acceleration

In the second series of tests, according to the Figure 5, the different-size particles are given a nonzero initial velocity into a quiescent fluid with a constant acceleration. At the start of the motion, the fluid velocity is less than all the particles when the particles are enforced to decelerate, and then accelerate when the fluid velocity exceeds the particle velocity. Analyzing the curves of $r = 0.0001$ and $r = 0.001$, it can be observed that once the particle velocity is lower than that of the fluid, the particle accelerates with the same acceleration as the fluid, consistently lagging behind it. Similarly, when the particle becomes sufficiently large, $r = 1$ for instance, it is scarcely affected.

In addition, it is obvious that the particles eventually move with the different velocity differences from the fluid. Suppose that the acceleration of fluid is a_g , the velocity difference is $\Delta v = v_g - v_p$. There is the equation that $a_g = F_d/m_p = a_p$ when the motion reaches the balance. In a general physical computation, drag force is positive correlation with the Δv , we have $F_d = \kappa \Delta v$ in which κ is a variable positive coefficient related to radius, density and viscosity when we used the different drag force model affecting the final velocity difference. For instance, in Stoke's drag model, there are $\kappa = 6\pi r_p \mu_g$ and $\Delta v = a_g r_p / 6\mu_g$ which is consistent with the numerical results.

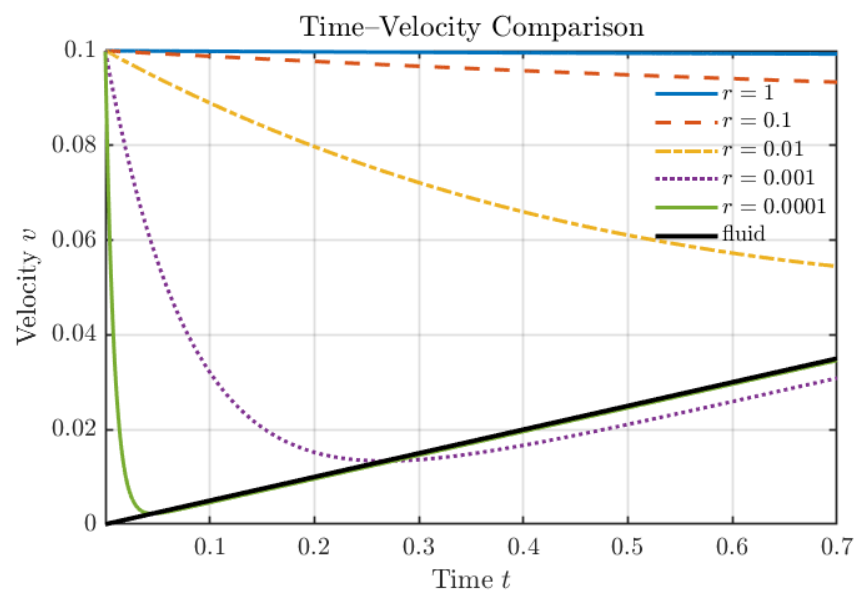


Figure 5. Time–Velocity in a constant acceleration fluid.

5.2. Multiphase Sod Test

We consider a shock tube problem in the domain $[0, 1] \times [0, 0.1]$ following the classical setup proposed in [29]. The solution consists of a left rarefaction, a contact discontinuity and a right-moving shock. The length of the shock tube is normalized to unity. To investigate the influence of particle–fluid interaction, we conduct a series of numerical experiments under varying particle sizes. In all cases, particles are uniformly distributed across the computational domain and the flow phase is modeled as an ideal gas. The initial flow field is piecewise constant, with a high-pressure state on the left and a low-pressure state on the right of $x = 0.5$. Specifically, the left state is given by $(\rho_l, P_l, u_l) = (9.6, 1,013,250, 0)$, while the right state is $(\rho_r, P_r, u_r) = (1.2, 101,325, 0)$, as reported in [17]. We employ a uniform mesh in the domain with the mesh resolution $N_x = 100$ and $N_y = 10$.

The physical interactions and resulting flow behaviors are analyzed based on high-resolution numerical simulations. In the numerical tests, we investigate the influence of gas–particle coupling by varying the particle radius to obtain different particle volume fractions: 0%, 0.01%, and 4%. The corresponding initial conditions are summarized in Table 1. The final average CPU times for the three cases are about 5.5 s, 6.1 s, and 6.3 s on the same workstation, respectively.

Table 1. Initial conditions for particles.

Case	Volume Fraction (%)	Radius	Velocity
1	0	0	0
2	0.01	0.0001	0
3	4	0.002	0

The simulations are conducted in a two-dimensional domain to assess the stability of the proposed algorithm. For clarity, both one- and two-dimensional results are presented. Although the Sod shock tube is essentially one-dimensional, we also compute it on a 2D mesh because the proposed HLLC-2D solver is constructed in a two-dimensional manner, with flux and nodal velocity evaluations involving coupled contributions from cells around a vertex. The 2D setup, therefore, serves as a consistency and robustness check of the 2D construction. In particular, the 2D results confirm that the algorithm effectively eliminates spurious oscillations near discontinuities, even in the presence of particles, and coincide with the corresponding 1D results.

Figures 6 show the flow field (density, velocity, energy and pressure) at time $t = 0.0004$ under the condition of different volume fractions. Different colors denote different particle volume fractions. The results show that, when the particle loading is sufficiently small, its influence on the carrier flow remains negligible. As the volume fraction increases, the drag effect becomes more pronounced, leading to a reduction in wave propagation speed and a decrease in the overall flow velocity, while the internal energy in the central region rises. Although discontinuities are still present in the flow field, the presence of particles acts in a manner analogous to a weak viscous term, smoothing the variations across contact discontinuity and shock. This behavior is consistent with the findings reported in [17].

To further verify the stability of the proposed scheme, simulations were conducted in a two-dimensional domain. We give the results in Figure 7. The results show that no spurious oscillations arise near discontinuities, and the flow field remains consistent along the y -direction. Even with particle–fluid interactions, the scheme demonstrates strong robustness.

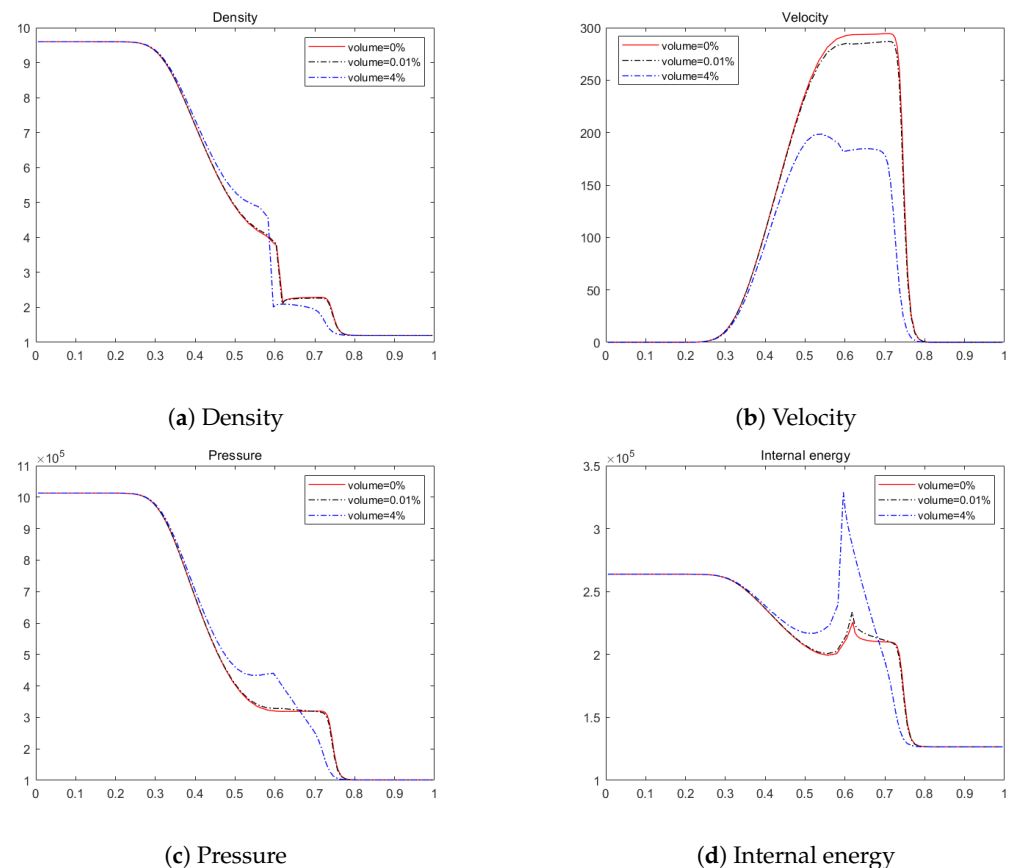


Figure 6. One-dimensional profiles for different particle volume fractions.

To compare the effects of Eulerian, Lagrangian, and ALE mesh strategies, we consider the case with a particle volume fraction of 4% and perform simulations under three settings: a fixed mesh (Eulerian), a Lagrangian mesh, and a moving-mesh ALE formulation. The corresponding results are shown in Figure 8. For this test regime and the present mesh resolution, the three strategies yield very similar profiles for density, pressure, and internal energy; only a modest discrepancy is observed in the fluid velocity in localized regions, while the overall trends remain consistent. We emphasize that this observation does not imply that the three approaches necessarily achieve comparable accuracy for more general two-phase flow problems. In an ALE framework, moving mesh strategies can significantly affect numerical dissipation and the resolution of discontinuities. The mesh-moving strategy adopted in this work follows [30], and formulating our scheme within an ALE framework is motivated by the need for greater geometric flexibility across different two-phase problems. For instance, highly skewed or excessively flattened cells may hinder particle search, whereas an adaptive ALE mesh can help maintain cell quality and thereby improve the robustness and applicability of the coupled particle-flow computation.

Specifically when the particle volume fraction is set to 0%, the coupled system degenerates to the classical compressible Euler equations; the slight smearing of the shock and contact discontinuity on coarse meshes is mainly attributable to the numerical viscosity of shock-capturing schemes, and a detailed accuracy and convergence study of the HLLC-2D Riemann solver for the Sod test is available in our previous work [28].

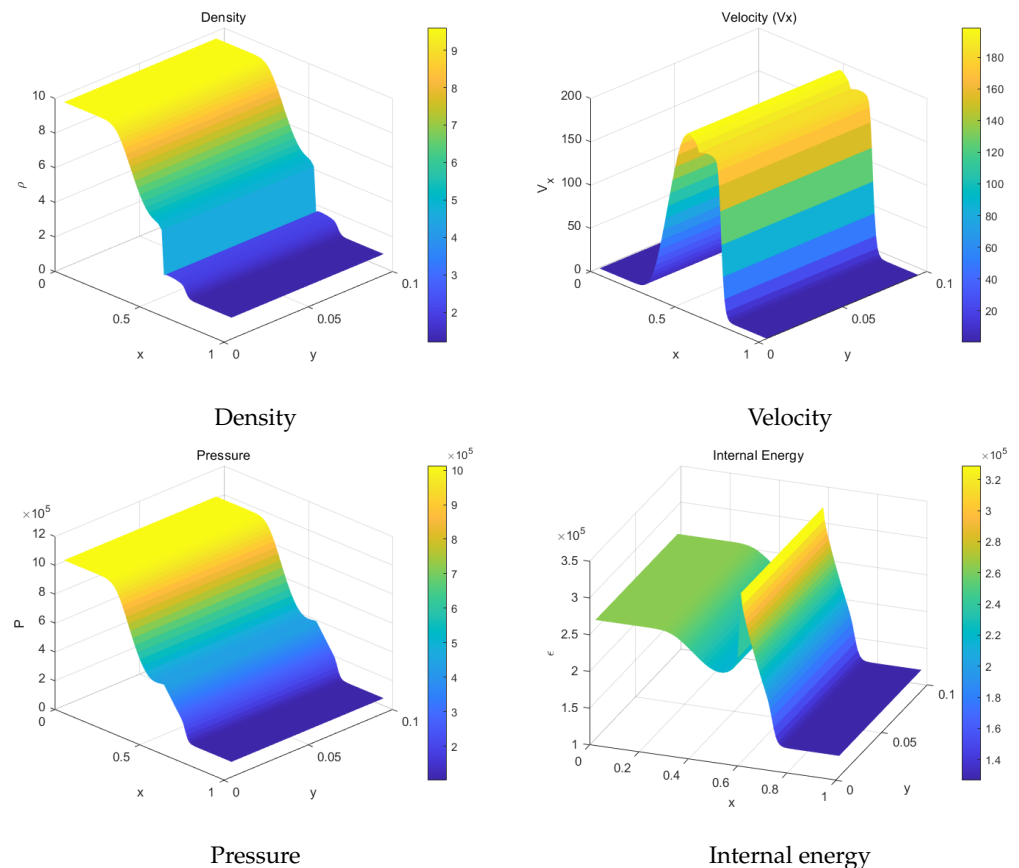


Figure 7. Two-dimensional result profiles under 4%.

5.3. The Motion of Cerium in Reactive Gas

Detonation simulations are widely used as benchmark problems for coupled particle transport and interphase heat exchange. We consider a sealed square container (Figure 9) with a cerium plate placed at the bottom, which generates ejecta under shock loading. When the carrier gas is helium (He), interphase chemical reactions are absent, whereas reactions occur in a deuterium (D_2) environment as described in Section 2.3. We therefore compare particle heating and transport in He and D_2 under otherwise identical conditions.

The parameters used in the numerical experiments were taken from [11]. The cerium plate is approximately 0.6 cm in length, and the ejecta mass is about 35 mg/cm. The initial particle speed ranges from roughly 1.1 km/s to 1.4 km/s, with a small transverse component whose magnitude depends on the mass ratio. To better represent realistic conditions, the particle radius is not prescribed as a uniform value but varies from 1 μm to 10 μm . The total number of simulated particles is up to 10^6 . In what follows, we compare particle motion and heat-transfer effects under different background gases.

We first investigate the particle motion in He, where no reaction occurs during the process. Neglecting the initial stage in which the cerium plate is impacted by the shock wave, we analyze the particle dynamics over 2 μs ; the numerical results are shown in Figure 10. At $t = 0$, the particles start to move. As the particles evolve, they gradually spread outward and form layered structures due to differences in their initial velocities and radii. A key feature of the enthalpy-based particle-trajectory model is that it resolves the trajectory of each individual particle; this is also illustrated by the second panel of Figure 10, where the boundary of the particle cloud is captured clearly. Although the particle phase exhibits a coherent overall trend, each particle follows its own distinct trajectory.

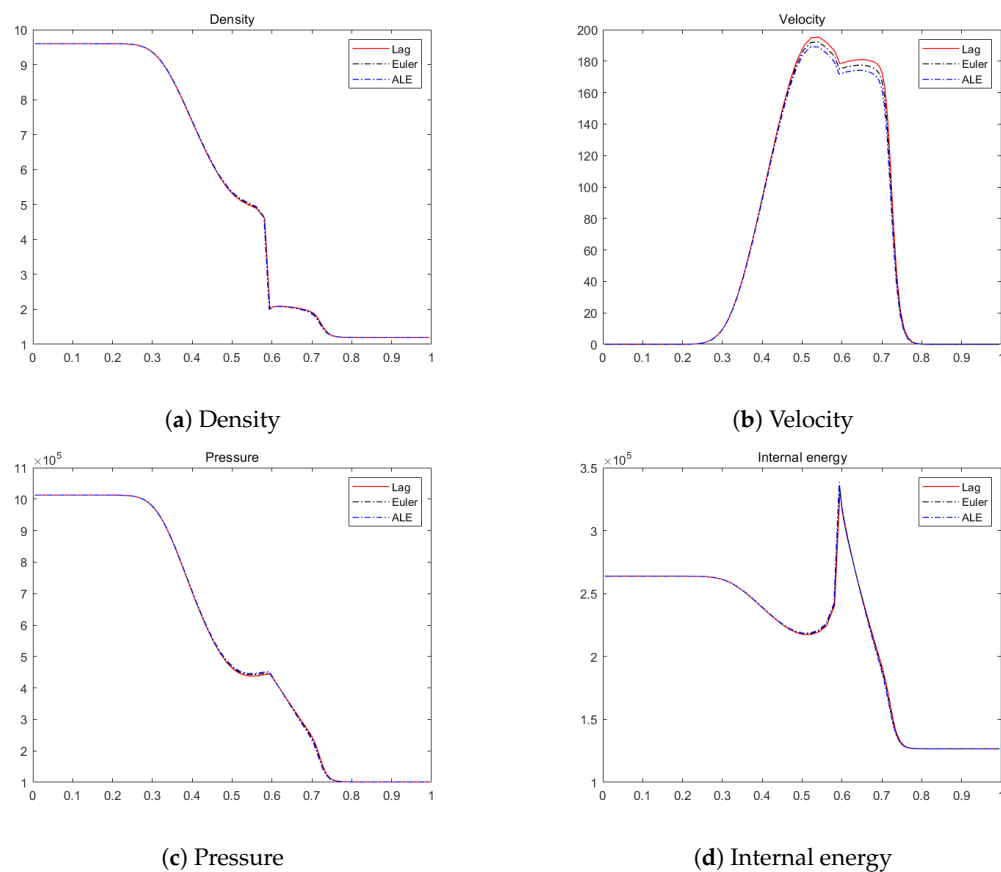


Figure 8. Comparison at 4% volume fraction using Eulerian, Lagrangian, and ALE mesh strategies.

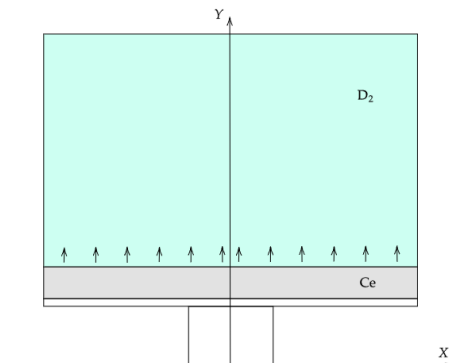


Figure 9. Reaction of Ce in D_2 .

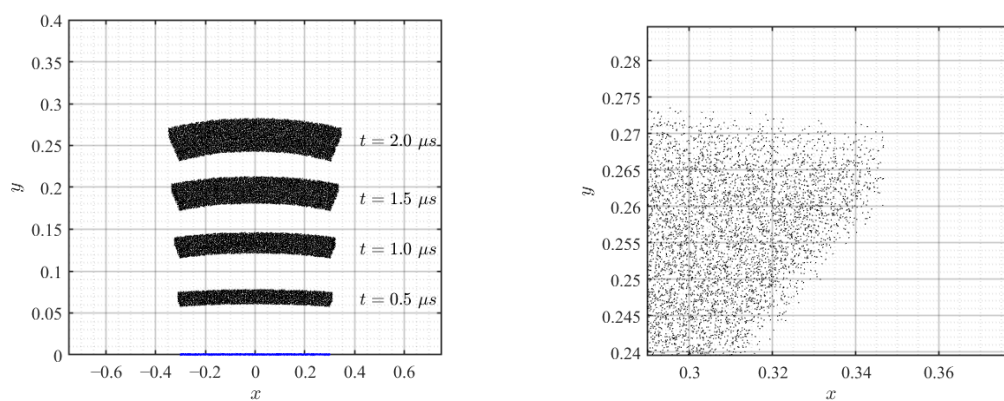


Figure 10. Particle motion at $t = 2 \mu s$.

We then perform numerical tests in D_2 with reactions taken into account. By comparing the results in Figure 11, it is evident that the presence of reactions has a pronounced impact on the temperature evolution. In He, we assume that the particles and the gas share the same initial temperature, which weakens the interphase heat exchange; consequently, only a slight temperature decrease is observed for particles near the boundary. In contrast, in D_2 the temperature rises markedly due to chemical reactions and shows clear spatial variations. Particles with larger radii and lower speeds exhibit more pronounced heating, indicating a higher extent of reaction. Regarding the velocity distribution in Figure 12, the overall speed field becomes progressively stratified in time as the particles are transported, with the highest speeds located in the central region.

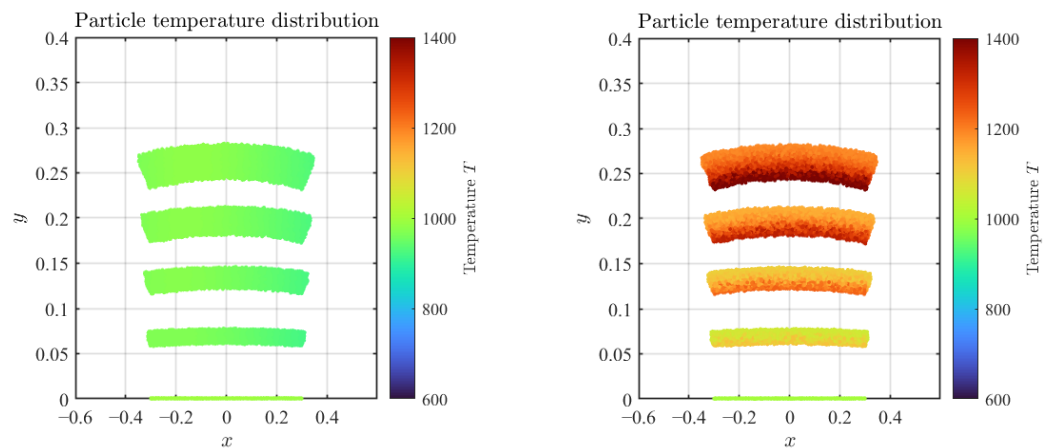


Figure 11. Temperature Change.

5.4. Conclusions

Due to strong interphase coupling, the gas flow and particle motion are tightly linked. In regimes where either the carrier-gas velocity or particle velocity becomes large, a purely Lagrangian description may suffer from severe mesh distortion and loss of robustness. To improve applicability across a wider range of flow conditions, we formulated the proposed HLLC-type Riemann solver within an Arbitrary Lagrangian–Eulerian (ALE) moving-mesh finite-volume framework, which allows mesh motion to be decoupled from the material motion while retaining conservative updates for the coupled system.

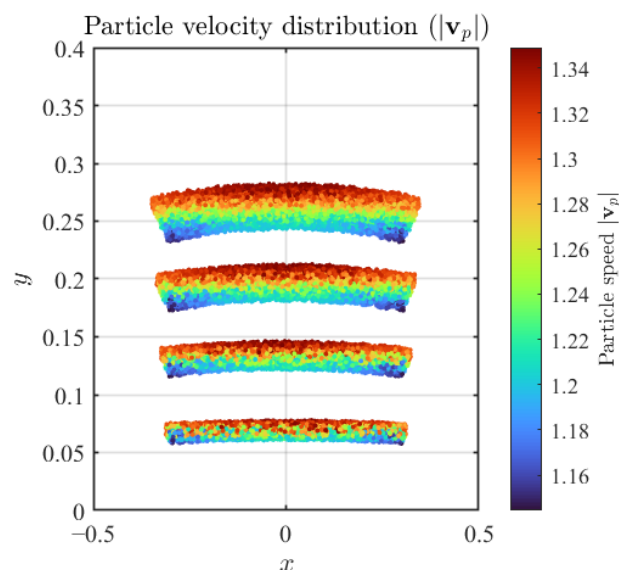


Figure 12. Velocity distribution.

The numerical results indicate that particle–fluid interactions have a pronounced impact on the flow evolution. In particular, interphase drag acts as an effective dissipative mechanism that attenuates wave structures and can modify propagation characteristics when the coupling is sufficiently strong; therefore, in particle-laden high-speed regimes, the feedback of particles on the carrier gas should be included for predictive simulations. The tests also show that the computed dynamics are sensitive to the choice of the gas viscosity coefficient. When the temperature difference between the gas and particles remains small, the thermal exchange term plays a secondary role relative to momentum exchange; however, in reactive settings the reaction-induced energy transfer leads to appreciable temperature variations and must be incorporated. Finally, our results suggest that a common-pressure approximation is reasonable at low particle volume fractions, whereas at higher loadings noticeable gas–particle pressure differences may arise and should be accounted for in the coupled modeling.

Author Contributions: Conceptualization, J.Z. and X.L.; methodology, J.Z.; software, J.Z.; validation, W.Y., J.Z. and X.L.; writing—original draft preparation, J.Z.; writing—review and editing, W.Y., J.Z. and X.L.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (grant numbers 12272059, 11871113, 22341302), the National Key Research and Development Program of China (grant number 2020YFA0713602), and the Key Laboratory of Symbolic Computation and Knowledge Engineering of the Ministry of Education, Jilin University.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Tangren, R.; Dodge, C.; Seifert, H. Compressibility effects in two-phase flow. *J. Appl. Phys.* **1949**, *20*, 637–645. [\[CrossRef\]](#)
2. Farrell, M.; Lun, C.; Savage, S. A simple kinetic theory for granular flow of binary mixtures of smooth, inelastic, spherical particles. *Acta Mech.* **1986**, *63*, 45–60. [\[CrossRef\]](#)
3. Jenkins, J.T.; Savage, S.B. A theory for the rapid flow of identical, smooth, nearly elastic, spherical particles. *J. Fluid Mech.* **1983**, *130*, 187–202. [\[CrossRef\]](#)
4. Lun, C.K.; Savage, S.B.; Jeffrey, D.; Chepurniy, N. Kinetic theories for granular flow: Inelastic particles in Couette flow and slightly inelastic particles in a general flowfield. *J. Fluid Mech.* **1984**, *140*, 223–256. [\[CrossRef\]](#)
5. Savage, S.; Jeffrey, D. The stress tensor in a granular flow at high shear rates. *J. Fluid Mech.* **1981**, *110*, 255–272. [\[CrossRef\]](#)
6. Sinclair, J.; Jackson, R. Gas–particle flow in a vertical pipe with particle–particle interactions. *AIChE J.* **1989**, *35*, 1473–1486. [\[CrossRef\]](#)
7. Campbell, C.S.; Gong, A. The stress tensor in a two-dimensional granular shear flow. *J. Fluid Mech.* **1986**, *164*, 107–125. [\[CrossRef\]](#)
8. Buttler, W.T.; Oro, D.M.; Dimonte, G.; Terrones, G.; Morris, C.; Bainbridge, J.; Hogan, G.E.; Hollander, B.J.; Holtkamp, D.B.; Kwiathowski, K. *Ejecta Model Development at pRad(u)*; Los Alamos National Lab. (LANL): Los Alamos, NM, USA, 2010.
9. Don Jayamanne, J.A.; Outerovitch, R.; Ballanger, F.; Bénier, J.; Blanco, E.; Chauvin, C.; Hereil, P.; Tailleur, J.; Durand, O.; Pierrat, R.; et al. Recovering particle velocity and size distributions in ejecta with photon Doppler velocimetry. *J. Appl. Phys.* **2024**, *136*, 083111. [\[CrossRef\]](#)
10. Buttler, W.T.; Schulze, R.K.; Charonko, J.J.; Cooley, J.C.; Hammerberg, J.E.; Schwarzkopf, J.D.; Sheppard, D.G.; Goett, J.J., III; Grover, M.; La Lone, B.M.; et al. Understanding the transport and break up of reactive ejecta. *Phys. D* **2021**, *415*, 132787. [\[CrossRef\]](#)
11. Lyn, S.; Shi, X.; Han, D.; Ma, Z.; Ma, D.; Sun, H.; Wang, P. Numerical simulations of solid cerium ejecta transporting in vacuum and in non-reactive and reactive gases. *J. Appl. Phys.* **2023**, *133*, 17. [\[CrossRef\]](#)
12. Capecelatro, J.; Wagner, J. Gas–Particle Dynamics in High-Speed Flows. *Annu. Rev. Fluid Mech.* **2024**, *56*, 475–510. [\[CrossRef\]](#)
13. Liang, Y.; Wu, L.; Shao, J.; Xiao, Y. Numerical study on the shock-induced ejecta formation from a metal surface. *Phys. Fluids* **2024**, *36*, 063324. [\[CrossRef\]](#)
14. Andrews, M.J.; O’Rourke, P. The multiphase particle-in-cell (MP-PIC) method for dense particulate flows. *Int. J. Multiph. Flow* **1996**, *22*, 379–402. [\[CrossRef\]](#)

15. O'Rourke, P.; Snider, D. Progress Toward Inclusion of Collision Effects in MP-PIC. In Proceedings of the NETL 2011 Multiphase Flow Workshop, Pittsburgh, PA, USA, 16–18 August 2011.
16. Snider, D.M. An incompressible three-dimensional multiphase particle-in-cell model for dense particle flows. *J. Comput. Phys.* **2001**, *170*, 523–549. [[CrossRef](#)]
17. Tian, B.; Zeng, J.; Meng, B.; Chen, Q.; Guo, X.; Xue, K. Compressible multiphase particle-in-cell method (CMP-PIC) for full pattern flows of gas–particle system. *J. Comput. Phys.* **2020**, *418*, 109602. [[CrossRef](#)]
18. Li, P.; Wang, Z.; Zhang, Y.; Ren, W.; Zhang, X.; Lu, X. A compressible semi-resolved computational fluid dynamics-discrete element method coupling model for fluid–solid systems with heat transfer. *Phys. Fluids* **2024**, *36*, 103315. [[CrossRef](#)]
19. Marayikkottu Vijayan, A.; Levin, D.A. Study of shock interaction with a particle curtain using the Multiphase Particle-in-Cell (MP-PIC) approach. In *AIAA SciTech Forum*; AIAA Paper 2023-0281; American Institute of Aeronautics and Astronautics (AIAA): Reston, VA, USA, 2023.
20. Fung, J.; Harrison, A.K.; Chitanvis, S.; Margulies, J. Ejecta source and transport modeling in the FLAG hydrocode. *Comput. Fluids* **2013**, *83*, 177–186. [[CrossRef](#)]
21. Chertock, A.; Cui, S.; Kurganov, A. Hybrid finite-volume-particle method for dusty gas flows. *Smaj J. Comput. Math.* **2017**, *3*, 139–180. [[CrossRef](#)]
22. Zong, X.; Zhang, J.; Zhang, J.; Li, X. A Discontinuous Galerkin Method on Moving Mesh for Ejecta Mixing. *Comput. Math. Math. Phys.* **2025**, *65*, 457–467. [[CrossRef](#)]
23. Ukai, S.; Balakrishnan, K.; Menon, S. On Richtmyer–Meshkov instability in dilute gas–particle mixtures. *Phys. Fluids* **2010**, *22*, 104103. [[CrossRef](#)]
24. Crowe, C.T.; Schwarzkopf, J.D.; Sommerfeld, M.; Tsuji, Y. *Multiphase Flows with Droplets and Particles*; CRC Press: Boca Raton, FL, USA, 2011.
25. Morsi, S.; Alexander, A. An investigation of particle trajectories in two-phase flow systems. *J. Fluid Mech.* **1972**, *55*, 193–208. [[CrossRef](#)]
26. Wang, P.; Sun, H.; Shao, J.; Qin, C.; Li, X. Numerical simulation on mixing process of ejecta and gas. *Wuli Xuebao/Acta Phys. Sin.* **2012**, *61*, 234703. [[CrossRef](#)]
27. Chen, D.; Sun, H.; Pei, W.; Yu, X.; Ma, D. Numerical investigation on the influence of gas–particle two-way coupling to the shock fluid in the two-dimensional Lagrangian framework. *Acta Phys. Sin.* **2016**, *65*, 084703. [[CrossRef](#)]
28. Shen, Z.; Yan, W.; Yuan, G. A robust and contact resolving Riemann solver on unstructured mesh, part II, ALE method. *J. Comput. Phys.* **2014**, *268*, 456–484. [[CrossRef](#)]
29. Sod, G.A. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *J. Comput. Phys.* **1978**, *27*, 1–31. [[CrossRef](#)]
30. Tang, H.; Tang, T. Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws. *SIAM J. Numer. Anal.* **2003**, *41*, 487–515. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.