Title

High-Speed Particle-Laden Flows : Eulerian-Lagrangian Methods, Multi-Scale Modeling and Numerical/Physical Instabilities

Author

Davis, Sean Lin Sheng

Publication Date

2015

UNIVERSITY OF CALIFORNIA, SAN DIEGO

High-Speed Particle-Laden Flows: Eulerian-Lagrangian Methods, Multi-Scale Modeling and Numerical/Physical Instabilities

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Engineering Sciences (Mechanical and Aerospace Engineering)

by

Sean Lin Sheng Davis

Committee in charge:

University of California, San Diego

Professor Sutanu Sarkar, Co-Chair
Professor David J. Benson
Professor Juan C. del Alamo
Professor H. S. Udaykumar

San Diego State University

Professor Gustaaf B. Jacobs, Co-Chair
Professor Fletcher J. Miller

2015

The dissertation of Sean Lin Sheng Davis is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Co-Chair

Co-Chair

University of California, San Diego

2015

DEDICATION

To my parents, who let me find my own path and always encouraged me to reach for the stars.

EPIGRAPH

There is almost no area of human life today that is not touched by the effects of science and technology. Yet are we clear about the place of science in the totality of human life - what exactly it should do and by what it should be goverened?

—Dalai Lama XIV

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to express my gratitude towards Dr. Gustaaf Jacobs, who has been an excellent academic advisor and has patiently guided me through difficult research challenges, while remaining an ever positive influence. Not only has he pushed me academically, but he has reinforced the importance of outreach and dissemination of knowledge.

Also to my grandparents: Pop always made sure there was a computer in the house, which got me started on my path (by constantly having to fix it); and Grandpop who taught me, “Who the hell deserves it more than me.”

I cannot describe the gratitude I feel towards my entire family. My parents, Peter and Peggy, have always taught me the importance of hard work while still maintaining a fun and adventurous life. My sister, Jackie, has always been a supportive influence and really pushed me towards the completion of my goals. Last but certainly not least, Laura, you have been patient with me, loved me and shared in my passions; I’m truly inspired by your determination each and every day.

I would also like to acknowledge a former coworker whom I think of often, Mamadou Ndiaye. His ever positive outlook on life, even through the hardest of times, helps many people remember what’s important.

Sections 5.1 to 5.6, in full, are a reprint of the material as it appears in Evaluation of Convergence Behavior of Metamodeling Techniques for Bridging Scales in Multi-scale Multimaterial Simulation. Sen, Oishik; Davis, Sean; Jacobs, Gustaaf; Udaykumar, H.S. Elsevier Press, 2015. The dissertation/thesis author was the primary investigator and author of this material.

Sections 6.1 to 6.5, in full, are to be submitted to the AIAA Journal under the title, Carbuncles in high resolution Eulerian-Lagrangian simulations of Shock-Particle-laden flow. Davis, Sean; Don, W.S.; Jacobs, Gustaaf. The dissertation/thesis author was the

primary investigator and author of this material.

Sections 7.1 to 7.4, in full, are a reprint of the material as it appears in Dispersion of a cloud of particles by a moving shock: Effects of the shape, angle of rotation, and aspect ratio. Davis, Sean; Dittman, Thomas; Jacobs, Gustaaf; Don, Wai Sun, Springer Publishing, 2013. The dissertation/thesis author was the primary investigator and author of this material.

Sections 8.1 to 8.7, in full, are a reprint of the material as it appears in The Effect of Non-Uniform Mass Loading on the Linear, Temporal Development of Particle-Laden Shear Layers. Senatore, Giacomo; Davis, Sean; Jacobs, Gustaaf, AIP Publishing, 2015. The dissertation/thesis author was the primary investigator and author of this material.

I would also like to thank Lockheed Martin, the California Space Grant Consortium and the Air Force Office of Scientific Research for their financial support and backing.

This research would not have been possible without the assistance from Dr. W. S. Don and the use of his WENO-Z libraries. Thanks also to my co-chair, Dr. Sutanu Sarkar and committee members, Dr. David Benson, Dr. Juan Carlos del Alamo, Dr. Fletcher Miller and Dr. H.S. Udaykumar for their invaluable assistance in perfecting this dissertation.

VITA

2009	Bachelors of Engineering in Mechanical Engineering with a concentration in Aeronautics, McGill University, Montreal, QC
2010-2012	Lockheed Leadership Scholarship recipient
2012	Intern at Los Alamos National Laboratory, Los Alamos, NM
2013-2015	Teaching Assistant for Compressible Flow (undergraduate) and Turbulence (graduate) courses, San Diego State University, San Diego, CA
2015	Doctor of Philosophy in Engineering Sciences (Mechanical and Aerospace Engineering), Joint Doctoral Program, San Diego State University and University of California San Diego, San Diego, CA

PUBLICATIONS AND PRESENTATIONS

Journal Publications

G. Senatore, S. Davis and G. Jacobs, The effect of non-uniform mass loading on the linear, temporal development of particle-laden shear layers, *Physics of Fluids*, 27 (3):033302, 2015.

S. Davis, T. Dittmann, G. Jacobs and W.S.Don, Dispersion of a cloud of particles by a moving shock: Effects of shape, angle of incidence and aspect ratio, Journal of Applied Mechanics and Technical Physics, published by Springer, *Journal of Applied Mechanics and Technical Physics*, 54 (6):900-912, 2013. *Published in both English and Russian*.

O. Sen, S. Davis, G. Jacobs and H.S. Udaykumar, Evaluating Convergence and Accuracy of Techniques to Bridge Scales in a Multi-Scale Solver, *Journal of Computational Physics, In Press.*

S. Davis, G. Jacobs and W. S. Don, Carbuncles in high resolution Eulerian-Lagrangian simulations of particle-laden flow, *AIAA Journal, Submitted for Publication*.

S. Davis, O. Sen, H. S. Udaykumar and G. Jacobs, Reynolds Averaged Modeling of Subgrid Particle Cloud Dynamics, *Journal of Multiphase Flow, Under Preparation*.

O. Sen, S. Davis, G. Jacobs and H. S. Udaykumar, Multi-Scale Modeling of the 1D Interaction of a Shock and a Dense Cloud of Particles, *AIAA Journal, Under Preparation*.

Peer-Reviewed Conference Proceedings

S. Davis, O. Sen, H. S. Udaykumar and G. Jacobs, Coupling of Micro-Scale and Macro-Scale Eulerian-Lagrangian Models for the Computation of Shocked Particle-Laden Flows, *ASME International Mechanical Engineering Congress and Exposition*, 2013 Nov 15-21, San Diego, CA.

S. Davis, G. Jacobs and W.S. Don, Carbuncles in high resolution Eulerian-Lagrangian simulation of shock-particle-laden flow, *21st AIAA CFD Conference*, 2013 June 24-27, San Diego, CA.

S. Davis, T. Dittmann, G. Jacobs and W.S. Don, High-Fidelity Eulerian-Lagrangian Methods for Simulation of Three Dimensional, Unsteady, High-Speed, Two-Phase Flows in High-Speed Combustors, *47th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, 2011 Jul 31-Aug 2, San Diego, CA.

Select Presentations

S. Davis, G. Senatore and G. Jacobs, The Effect of Varying Stokes Number on the Growth Rate of Instabilities in Particle-Laden Shear Layers, *American Physical Society Division of Fluid Dynamics*, San Francisco, CA, Nov. 2014.

S. Davis, O. Sen, H. S. Udaykumar and G. Jacobs, Coupling of Micro-Scale and Macro-Scale Eulerian-Lagrangian Models for the Computation of Shocked Particle-Laden Flows, *ASME International Mechanical Engineering Congress and Exposition*, San Diego, CA, Nov. 2013

S. Davis, G. Jacobs and W.S. Don, Carbuncles in high resolution Eulerian-Lagrangian simulation of shock-particle-laden flow, *21st AIAA CFD Conference*, San Diego, CA, June 2013.

S. Davis, W. S. Don and G. Jacobs, Effect of initial cloud shape and orientation on particle dispersion in the accelerated flow behind a shock, *American Physical Society Division of Fluid Dynamics*, San Diego, CA, Nov. 2012.

S. Davis, T. Dittmann, G. Jacobs and W.S. Don, High-Fidelity Eulerian-Lagrangian Methods for Simulation of Three Dimensional, Unsteady, High-Speed, Two-Phase Flows in High-Speed Combustors, *47th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, San Diego, CA, Aug. 2011.

S. Davis, S. Sarkar and G. Jacobs, Cloud Dispersion in Shocked Particle-Laden Flow, *US National Congress on Computational Mechanics*, Raleigh, NC, July 2013, *Poster Presentation*.

ABSTRACT OF THE DISSERTATION

High-Speed Particle-Laden Flows: Eulerian-Lagrangian Methods, Multi-Scale Modeling and Numerical/Physical Instabilities

by

Sean Lin Sheng Davis

Doctor of Philosophy in Engineering Sciences (Mechanical and Aerospace Engineering)

University of California, San Diego, 2015

Professor Gustaaf B. Jacobs, Co-Chair
Professor Sutanu Sarkar, Co-Chair

High-speed particle-laden flows are studied through a combination of high-order Eulerian Lagrangian (EL) method development, multi-scale modeling and the analysis of normal, shear and wake instabilities. The EL code is based on a Weighted Essentially Non-Oscillatory (WENO) discretization, which captures discontinuities sharply while ensuring higher-order resolution in smoother areas. The favorable characteristics of WENO methods are extended to the EL framework with an Essentially Non-Oscillatory (ENO) scheme to interpolate the carrier phase from the Eulerian grid to the particle location.

A high fidelity multi-scale method is introduced that couples full-resolution micro-scale statistics with the macro-scale. Using a Taylor expansion of the drag correction factor and Reynolds averaging of the particle transport equation, the Subgrid Particle Reynolds Stress Equivalent (SPARSE) model is derived. A mantle is constructed to provide closure for the particle drag and subgrid particle dynamics models. The efficacy of several metamodeling techniques in building a mantle with a target uncertainty using the least number of support points is compared.

Grid aligned shocks at high mach numbers create normal instabilities that bleed into the particle phase. These so-called "carbuncles" have a predominantly numerical nature and can be mitigated by adjusting the nonlinear WENO power parameters.

Instabilities that are induced by the accelerated flow behind a moving shock in the wake of a cloud of particles are analyzed. The initial shape, orientation and dimensionality of the particle cloud with respect to an oncoming normal shock determines the particle dispersion at later times. Dispersion characteristics are matched with results reported in literature. Streamlined cloud shapes exhibit a lower dispersion as compared to clouds with an initially blunt shape. Lower particle number density areas away from the heavily populated cloud core accelerate more.

Growth rates of shear instabilities computed with the EL code are compared with the results of a linear stability analysis of particle-laden shear layers. The growth rate of shear layers with non-uniformly laden low Stokes number particles is greater than an unladen shear layer whereas a shear layer with high Stokes number particles dampens the growth.

Chapter 1

Introduction

1.1 Motivation

Particle-laden and droplet-laden flows play an important role in high-speed technologies such as solid rocket propulsion systems and high-speed liquid-fuel combustors. Shock waves occur in scramjet combustors and interact with fuel particles in the supersonic flow. These environments exhibit tremendous physical complexity requiring the simultaneous resolution of particle-turbulence, shock-turbulence, and shock-particle interactions. Capturing finite shock discontinuities without oscillations and resolving turbulence while simultaneously tracing particle paths through the high-speed flow requires complex models and advanced methods. The large range of spatial and temporal scales, furthermore, pose high demands on both experimental and computational analysis.

Because of the high velocities, experimental analysis is limited to large scale observation or the examination of a single short term event. Experiments studying instabilities such as Richtmeyer Meshkov [3] and those encountered when a shock impacts an initially stationary cloud of droplets [151] have elucidated some of the fundamental mixing occurring on small scales at early times in the interaction of a shock and particles. Later time analysis of the interaction of a blast wave and metallic particles [163, 48, 47] show unique instabilities, which cannot be fully resolved using current measurement techniques. To understand the fundamental interactions occurring in the mixing and transport of both the carrier and particle phases on a process-scale problem, computational analysis is needed.

However, first principle computations are expensive and models are not optimal. The direct analysis of particle-laden shocked flow requires the computation of the entire flow over each particle, the tracking of individual solid or liquid complex particle boundaries along their paths, and the capturing of shock waves in the moving framework. These individual components are difficult to resolve on their own and currently barely within reach, even with current advances in computational architecture. The combined difficulty of resolving shocks and particle paths has an immense complexity, scale range and size, which can currently only be realistically analyzed in highly idealized situations with only a few particles [119, 90].

1.2 Numerical Scheme

Process-scale computations typically demand simplified macro-scale governing equations and simulation techniques. For example, in the problem of a shock wave interacting with a dusty gas, a common practice is to define a computational particle as an agglomerate of a number of dust particles [28]. The system is then modeled using a mixed Eulerian-Lagrangian (EL) viewpoint [18, 68]. Particle paths are traced in a Lagrangian reference frame while the fluid governing equations are solved in a fixed Eulerian frame. In this approach, the computational particles are modeled as singular point sources, which couple with the carrier fluid through momentum exchange modeled via source terms in the fluid equations [30, 68, 34, 33]. In high-speed particle-laden flows, the small time scales and range of interphase velocity differences causes the particle phase to rapidly disperse. Although an Eulerian-Eulerian approach is effective at capturing the large scale mixing of the particle and fluid phase [127, 128], dissipation errors arise while tracking the number density in low particle volume fraction environments. EL methods are better able to compute the dispersion patterns in sparse flows because individual computational particles are tracked.

In the Eulerian frame, discontinuities arise, which necessitate the use of shock capturing methods that can accurately model the sharp jump across a shock in a flow field in a stable manner. Traditionally, robustness considerations constrained shock-capturing schemes to only those which rely on low-order methods [50, 149, 83]. Although this is often good enough to capture the short time dynamics of the shock, numerical diffusion dissipates important smaller scale flow structures in the resultant wake. These dissipation errors extend throughout the domain, smearing both shocks and turbulence over longer time periods. The high resolution computation of particle-laden flows with a large range of spatial and temporal scales is limited by the use of lower order methods. Dispersion and diffusion errors plague the accuracy of the solution.

To accurately trace discontinuities while simultaneously resolving small-scale turbulence in the flow, high-order shock-capturing methods have been devised. Essentially Non-Oscillatory (ENO) schemes, first presented by Harten et. al. [56], provide a framework for the implementation of robust high-order finite difference methods for the solution of hyperbolic equations. These schemes maintain a high-order of accuracy while adaptively adjusting the stencil near discontinuities to avoid spurious Gibbs Oscillations. The stencil is adaptively chosen to provide the “smoothest” reconstruction, thereby avoiding stencils containing discontinuities [83].

Weighted ENO (WENO) schemes, first introduced in [85], improve upon on the adaptive stencil techniques of the ENO scheme. An optimal order of accuracy is obtained at smooth parts of the solution while enough dissipation is retained over discontinuous stencils to damp out oscillations [70]. This is accomplished through a convex combination of all the possible ENO substencils. Improved WENO methods have also been derived, such as the WENO-Z scheme [5, 18, 49, 35], with enhancements to decrease dissipation errors and ensure a high-order accuracy in smoother regions of the flow.

In the Lagrangian frame, Particle-Source-in-Cell (PSIC) methods, introduced by Crowe et. al. [30, 28], have proven to be effective models for solving larger scale problems. In the case where the number of physical particles to trace is prohibitively large, groups of physical particles are amalgamated into a single computational particle with an increased source factor, this scheme has been deemed the Cloud-In-Cell (CIC) method [10]. In PSIC and CIC methods, particles are modeled as points and their motion is forced by the drag exerted on them by the fluid [68, 66, 34, 33]. The fluid velocity and temperature are interpolated from the Lagrangian frame and used explicitly in the particle governing equations. To maintain the high-order accuracy of the solver and avoid oscillations in the interpolation, a high-order method is required [137]. The ENO method, discussed above, preserves both the high-order accuracy and robustness of the fluid solver [68].

The motion of the solid phase is governed by drag laws derived from Stokes' drag law [136], which assumes very low Reynolds number, $Re << 1$. More complicated heuristic drag laws governing the motion of particles in higher Reynolds number particle-laden flows and even accounting for compressibility have been formulated [26, 89]. Additional terms such as the slip coefficient [141], various shape factors [143], viscosity ratios for droplets [43], etc. have also been incorporated into these empirical models.

In recent studies summarized in books such as Crowe et. al. [28], significant effort has gone towards modeling particle drag in shocked, accelerated flows. Boiko et. al. [15, 13] determined the drag of a droplet behind a shock by comparing the known relaxation times of a hard sphere to their experimentally measured droplet relaxation times. Sun et al. [139] numerically studied the dynamic drag coefficient of a spherical particle behind a shock wave. A reflected bow shock was observed in front of the spherical particle, and, as the shock wave traversed over the sphere, a Mach reflection formed. The Mach reflection proceeded to the rear center of the sphere before converging with the Mach reflection from the other side. This caused shock focusing to occur and a region of very high pressure at the rear of the spherical particle. This region of high pressure resulted in a brief negative drag. Their numerical data matched experimental data within 10%. Loth [88] investigated the effect of compressibility and rarefaction on a spherical particle. Boiko et al. [14] also studied different shaped particles. In a comparison of a cubical and a spherical particle they found that the drag is predominantly a function of the frontal area of the particle. They conclude that the relative bluntness of the shapes do not significantly affect the particle dynamics.

In addition to improving the closure models for the particle drag, small scale perturbations in the particle phase must be accounted for to accurately trace particle dispersion patterns. Much effort has gone into modeling the complex sub-grid turbulent structures in the fluid governing equations [80, 99, 39, 134, 4]. By contrast, little effort has been devoted to the study of sub-scale particle fluctuations under the computational particle assumptions of CIC. Several articles [22, 102, 146] have studied the inclusion of small-scale particle-fluid energy transfer in their turbulence models. However, these studies focus on the effect that the particles have on the fluid rather than the influence of individual modeled particles on the averaged computational particle dynamics.

By coupling micro and macro-scale methods, closure models for the drag and sub-scale particle perturbation terms can be obtained using small-scale, full resolution computations. For example, Hasegawa et. al. [57] used the statistics determined from DNS computations to close modeled terms in macro-scale large eddy simulations (LES) to study mass transfer across turbulent flows with high Schmidt numbers. Other methods, such as the energy minimization multi-scale method [154] use multi-scale methods to study gas-solid two-phase flows. The closure of these terms with statistics generated directly from the meso-scale creates a highly robust and accurate method.

The complexity of the relationship between the particle drag and fluid motion as well as the requirement that numerous coupling parameters be considered, necessitates an accurate, efficient algorithm to represent the behavior of the system. Improving the accuracy of the sub-grid model in the macro-scale code requires additional full-resolution experiments, which consume significant computational resources. The requirement to limit the time and expense of performing supplementary DNS studies necessitates an efficient, accurate coupling algorithm to interpolate the sub-grid particle drag statistics and generate a representative mantle. The required number of input points to build this mantle must not grow exponentially as new parameters are added i.e. the scheme cannot suffer from the *curse of dimensionality*. Because the drag model function is not known *a priori*, the coupling algorithm must also show defined convergence. Methods such as the Stochastic Collocation (SC) [7, 160, 93], Kriging [84, 145, 86, 133], and Radial Basis Function Artificial Neural Networks (RBFANN) [23, 110, 58] fit these requirements.

1.3 Physical and/or Numerical Instabilities

As discussed above, ENO and WENO schemes are used to mitigate Gibbs Oscillations. However, stronger shocks in very high-speed flows create other instabilities when using high-order shock capturing methods. In situations where a very strong, slow moving shock is aligned with the numerical grid, anomalous bumps and oscillations can occur in the shock profile which have earned the nickname *carbuncle* [111]. These phenomena are typically studied in the case of solid bodies immersed in high-speed flows

[112, 111, 152]. Many "cures" have been proposed to eliminate carbuncles from numerical solvers [108, 120]. However, these methods generally add signficant dissipation, limiting the resolution of small-scale structures over long time periods. Experiments by Holden [60] have revealed that carbuncle-like phenomenon may arise naturally by the injection of dust particles along the stagnation line ahead of a blunt cylinder. Robinet et. al. [114] exhibited a new mode in the inviscid Euler equations, which could develop into a carbuncle. These conclusions have lead some researchers to the hypothesis that carbuncles are incurable [38]. Aside from a few idealized cases, however, experiments have not shown these instabilities to be naturally occurring. Therefore, using numerical methods that mitigate the effect of carbuncles on the carrier flow is desirable.

In addition to modeling and numerical instabilities, physical instabilities arise in the interaction between particles and fluids. Limited studies have been performed on the dynamics of a large number of particles in high-speed flow. Olim et al. [106] studied the attenuation of a normal shock wave in a homogeneous gas particle mixture. Kiselev et al. [78] compared simulations based on Boiko's empirical particle models to shock tube experiments on the dispersion of a cloud of plexiglas and bronze particles in the accelerated flow behind a moving shock [15]. Not only did they visualize the particle dynamics and dispersion, they also matched some of their results quantitatively to the experimental dispersions.

The experiment in [15] showed that the interaction of a shock and a particle cloud produces a turbulent wake and strong shear layers, which contribute heavily to the resultant particle dispersion. The growth rate of instabilities in particle-laden shear layers, therefore, is of great importance to the fundamental understanding of particle mixing and transport in high-speed multi-phase flows. The first analytical linear stability analysis (LSA) of a viscous, incompressible particle-laden shear layer was performed by Saffman in 1961 [118]. Saffman found that particles with a small Stokes number (St) destabilize a viscous flow because of an increase in the effective bulk mixture density and therefore Reynolds number (Re) of the flow. However, large St particles stabilize the flow because of the viscous dissipation of fluid moving around the large inertia particles.

1.4 Outline and Contributions

The primary contribution of this thesis is the elucidation and mitigation (in the case of spurious numerical errors) of the inherent instabilities which arise in the computation of high-speed particle laden flows. The first few chapters of the manuscript lay the foundation for the numerical tools used in this study the particle-laden flows. Improvements to the macro-scale modeling of computational particles by the addition of micro-scale information is addressed in the middle portion of the thesis. In the later chapters, the macro-scale computational method is used to study several numerical and/or physical instabilities. Solutions to mitigate the spurious numerical instabilities while maintaining the high fidelity of the solver are offered. The main contributions of this thesis can be summarized as follows:

- The effect of sub-scale individual particle dynamics on the averaged computational particle momentum is investigated. Additional terms are added to the particle phase governing equations to capture the response of the cloud to micro-scale perturbations in the particle phase. The inclusion of an averaged carrier phase velocity is analyzed and compared the the classical assumption that the fluid velocity can be interpolated at a single point.

- A multi-scale scheme, that couples a full resolution, micro-scale DNS code [119] with a macro-scale EL code [68] to provide a dynamic closure model to solve the particle momentum equations over all parameter spaces is developed. A series of small-scale DNS experiments generate particle statistics, which are then used to close the sub-grid models in a macro-scale EL solver. A coupling algorithm interpolates the correlation for momentum flux between the particle and fluid phases and determines the number of realizations that are required to obtain a target uncertainty in the data. The macro-scale code interrogates this coupling algorithm to determine the statistics needed to close the governing equations.

- *Normal* instabilities called “Carbuncles,” which arise in the numerical analysis of a very strong grid-aligned shock encountering an obstruction, are shown, for the first time, to occur in the interaction between an high-speed flow and a cloud of particles. Damping of these instabilities is accomplished by adjusting the power parameters of the WENO-Z nonlinear weights.

- *Wake* instabilities in the interaction of a right running normal shock and a cloud of inertial particles is studied. The influence of individual particles as well as the effect of the full macro-scale particle cloud on the fundamental mixing and transport of the particle phase is analyzed. Particle dispersion patterns are characterized based on the initial shape and orientation of the particle cloud. The strong shear layers formed by the rapid decrease in fluid velocity through blunter cloud shapes cause greater spanwise particle dispersion.

- The effect of particles on the linear growth of *shear* instabilities is explored from both a theoretical linear stability analysis and a computational approach. The growth rate of the shear layer instability increases when non-uniformly laden with fast responding particles as opposed to previous studies, which concluded that the addition of particles always stabilizes the shear layer. A bulk energy budget analysis shows a net transfer of energy from the small Stokes number particles to the carrier phase.

In the next section, governing equations for both the carrier phase and dispersed phase are presented. In Chapter 3, the high-order EL method is discussed in full detail. A multi-scale method coupling drag statistics from micro-scale computations with the macro-scale EL solver is explained in Chapter 5. Chapter 4 details the averaged subgrid particle momentum model, showing the improvements over existing CIC methods. Numerical instabilities are examined in Chpater 6 and methods to mitigate these spurious oscillations are given. In Chapter 7, physical instabilities arising from the interaction of particles and fluid are studied. The interaction of a normal shock and a particle cloud of differing shape and orientation is studied to elucidate the importance of both macro- and micro-scale particle-cloud-shock interactions on the full-scale problem. The shear layers, which cause much of the spanwise dispersion of the cloud in the above case, are studied in greater detail through both a Linear Stability Analysis (LSA) and computations. Conclusions and recommendations are reserved for the final chapter.

Chapter 2

The Physical Model and Governing Equations

In the particle-source-in-cell (PSIC) method, the Eulerian continuum equations are solved for the carrier flow in the Eulerian frame, while particles are traced along their paths in the Lagrangian frame. In the following, we present the coupled system of Euler equations that govern the gas flow and kinematic equations that govern the particle motion. We shall denote the subscript p for the particle variables and f for the gas variables at the particle position. Variables without subscript refer to the gas variables unless specified otherwise.

2.1 Carrier Phase

2.1.1 Dimensional Form

This section presents the equations of motion for a compressible Newtonian fluid, also known as the Navier-Stokes equations. The conservative, dimensional form of the conservation of mass, momentum and energy in Cartesian coordinates are given by

$$\mathbf{Q}_t^* + \mathbf{F}_x^{a*} + \mathbf{G}_y^{a*} + \mathbf{H}_z^{a*} = \mathbf{F}_x^{\nu*} + \mathbf{G}_y^{\nu*} + \mathbf{H}_z^{\nu*} + \mathbf{S}, \tag{2.1}$$

where the superscript * denotes dimensional variables, the bold face shows a vector and

$$\mathbf{Q}^* = \begin{bmatrix} \rho^* \\ \rho^* u^* \\ \rho^* v^* \\ \rho^* w^* \\ \rho^* e^* \end{bmatrix}, \tag{2.2}$$

where

$$\mathbf{F}^{a*} = \begin{bmatrix} \rho^* u^* \\ p^* + \rho^* u^{*2} \\ \rho^* u^* v^* \\ \rho^* u^* w^* \\ u^*(\rho^* e^* + p^*) \end{bmatrix}, \mathbf{G}^{a*} = \begin{bmatrix} \rho^* v^* \\ \rho^* u^* v^* \\ p^* + \rho^* v^{*2} \\ \rho^* v^* w^* \\ v^*(\rho^* e^* + p^*) \end{bmatrix}, \mathbf{H}^{a*} = \begin{bmatrix} \rho^* w^* \\ \rho^* u^* w^* \\ \rho^* v^* w^* \\ p^* + \rho^* w^{*2} \\ w^*(\rho^* e^* + p^*) \end{bmatrix}, \tag{2.3}$$

are the advective fluxes and

$$\mathbf{F}^{\nu *} = \begin{bmatrix} 0 \\ \tau^*_{11} \\ \tau^*_{12} \\ \tau^*_{13} \\ u^*\tau^*_{11} + v^*\tau^*_{12} + w^*\tau^*_{13} + \kappa^* T^*_x \end{bmatrix},$$

$$\mathbf{G}^{\nu *} = \begin{bmatrix} 0 \\ \tau^*_{21} \\ \tau^*_{22} \\ \tau^*_{23} \\ u^*\tau^*_{21} + v^*\tau^*_{22} + w^*\tau^*_{23} + \kappa^* T^*_x \end{bmatrix},$$

$$\mathbf{H}^{\nu *} = \begin{bmatrix} 0 \\ \tau^*_{31} \\ \tau^*_{32} \\ \tau^*_{33} \\ u^*\tau^*_{31} + v^*\tau^*_{32} + w^*\tau^*_{33} + \kappa^* T^*_x \end{bmatrix}, \tag{2.4}$$

are the viscous fluxes where

$$\begin{aligned}
\tau^*_{11} &= 2\mu^*(u^*_x - (u^*_x + v^*_y + w^*_z)/3), \\
\tau^*_{11} &= 2\mu^*(v^*_y - (u^*_x + v^*_y + w^*_z)/3), \\
\tau^*_{11} &= 2\mu^*(w^*_z - (u^*_x + v^*_y + w^*_z)/3), \\
\tau^*_{12} &= \tau^*_{21} = \mu^*(v^*_x + u^*_y), \\
\tau^*_{13} &= \tau^*_{31} = \mu^*(w^*_x + u^*_z), \\
\tau^*_{23} &= \tau^*_{32} = \mu^*(w^*_y + v^*_z).
\end{aligned} \tag{2.5}$$

The subscript on the stress tensor, τ^*, defines the direction of τ^* and the plane it is acting on with 1, 2 and 3 denoting the x, y and z direction respectively. In the above equations ρ^* is the density; u^*, v^* and w^* are the velocities in the x^*, y^* and z^* directions respectively; p^* is the pressure; T^* is the temperature; $\rho^* e^*$ is the sum of the internal and kinetic energy per unit volume given by $\rho^* c_v^* T^* + 1/2\rho^*(u^{*2} + v^{*2} + w^{*2})$, where c_v^* is the constant volume specific heat of the fluid; μ^* is the fluid dynamic viscosity; and κ^* is the dynamic conductivity. The source term, $\mathbf{S}$, accounts for the effect of the particles on the carrier gas and will be discussed in more detail below. The fluid is assumed to be an ideal gas where intermolecular forces are negligible, and obeys the equation of state,

$$p^* = \rho^* R^* T^*, \tag{2.6}$$

where R^* is the gas constant.

2.1.2 Non-Dimensional Form

To solve Equation 2.1, the non-dimensional form is favored because disparity in the physical variables' units can generate undesirable errors. To avoid such spurious errors, the equations are non-dimensionalized by reference length, L_f^*, density, ρ_f^*, velocity, U_f^*, and temperature, T_f^* scales, leading to the non-dimensional variables:

$$\begin{aligned}
\rho &= \frac{\rho^*}{\rho_f^*}, & u &= \frac{u^*}{U_f^*}, \\
v &= \frac{v^*}{U_f^*}, & w &= \frac{w^*}{U_f^*}, \\
T &= \frac{T^*}{T_f^*}, & p &= \frac{p^*}{\rho_f^* U_f^*}, \\
x &= \frac{x^*}{L_f^*}, & x &= \frac{x^*}{L_f^*}, \\
z &= \frac{z^*}{L_f^*}, & t &= \frac{t^* U_f^*}{L_f^*}.
\end{aligned} \tag{2.7}$$

Substituting these non-dimensional values into Equation 2.1, yields the non-dimensional Navier-Stokes equation in conservative form,

$$\mathbf{Q}_t + \mathbf{F}_x^a + \mathbf{G}_y^a + \mathbf{H}_z^a = \frac{1}{Re_f}(\mathbf{F}_x^\nu + \mathbf{G}_y^\nu + \mathbf{H}_z^\nu) + \mathbf{S}, \tag{2.8}$$

where

$$\mathbf{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e \end{bmatrix} \tag{2.9}$$

where

$$\mathbf{F}^a = \begin{bmatrix} \rho u \\ p + \rho u^2 \\ \rho uv \\ \rho uw \\ u(\rho e + p) \end{bmatrix}, \; \mathbf{G}^a = \begin{bmatrix} \rho v \\ \rho uv \\ p + \rho v^2 \\ \rho vw \\ v(\rho e + p) \end{bmatrix}, \; \mathbf{H}^a = \begin{bmatrix} \rho w \\ \rho uw \\ \rho vw \\ p + \rho w^2 \\ w(\rho e + p) \end{bmatrix}, \tag{2.10}$$

are the advective fluxes and

$$\mathbf{F}^\nu = \begin{bmatrix} 0 \\ \tau_{11} \\ \tau_{12} \\ \tau_{13} \\ u\tau_{11} + v\tau_{12} + w\tau_{13} + \frac{1}{(\gamma-1)M_f^2 Pr} T_x \end{bmatrix},$$

$$\mathbf{G}^\nu = \begin{bmatrix} 0 \\ \tau_{21} \\ \tau_{22} \\ \tau_{23} \\ u\tau_{21} + v\tau_{22} + w\tau_{23} + \frac{1}{(\gamma-1)M_f^2 Pr} T_x \end{bmatrix},$$

$$\mathbf{H}^\nu = \begin{bmatrix} 0 \\ \tau_{31} \\ \tau_{32} \\ \tau_{33} \\ u\tau_{31} + v\tau_{32} + w\tau_{33} + \frac{1}{(\gamma-1)M_f^2 Pr} T_x \end{bmatrix}, \tag{2.11}$$

are the viscous fluxes, where

$$\begin{aligned}
\tau_{11} &= 2(u_x - (u_x + v_y + w_z)/3), \\
\tau_{11} &= 2(v_y - (u_x + v_y + w_z)/3), \\
\tau_{11} &= 2(w_z - (u_x + v_y + w_z)/3), \\
\tau_{12} &= \tau_{21} = v_x + u_y, \\
\tau_{13} &= \tau_{31} = w_x + u_z, \\
\tau_{23} &= \tau_{32} = w_y + v_z. \qquad (2.12)
\end{aligned}$$

with

$$\rho e = \frac{p}{\gamma - 1} + \frac{\rho(u^2 + v^2 + w^2)}{2}. \qquad (2.13)$$

The inviscid fluxes, denoted with an 'a' superscript, remain unchanged from Equation 2.1. However, the viscous fluxes, denoted with the 'ν' superscript, give rise to three non-dimensional parameters: the Reynolds number, $Re_f = \frac{U_f^* L_f^* \rho_f^*}{\mu^*}$, which indicates the influence of the viscous fluxes as compared to the advective fluxes; the Prandtl number, $Pr = \frac{c_p^* \mu^*}{\kappa^*}$ where c_p^* is the constant pressure specific heat of the fluid, which is the ratio of the viscous and thermal diffusivity; and the reference Mach number, $M_f = \frac{U_f^*}{c^*}$ where $c^* = \sqrt{\gamma R^* T_f^*}$ is the reference speed of sound and $\gamma = \frac{c_p^*}{c_v^*}$. The ideal gas equation of state in non-dimensional form is given as

$$p = \frac{\rho T}{\gamma M_f^2} \qquad (2.14)$$

In the sections where we analyze high-speed shock-particle interaction, we assume Re_f large and neglect the viscous fluxes. We therefore do not explicitly model viscous effects such as boundary layers or shock waves in the governing Eulerian equations for the gas flow.

2.2 Dispersed Phase

2.2.1 Dimensional Form

Particles are tracked individually in the Lagrangian frame using the analytical solution of the flow around a spherical particle corrected for high particle Reynolds and Mach number. The kinematic equation describing the particle's position $\mathbf{x_p}^*$, is given as

$$\frac{d\mathbf{x_p}^*}{dt^*} = \mathbf{v_p}^*, \tag{2.15}$$

where $\mathbf{v_p}^*$ is the particle velocity vector.

The particles' acceleration is governed by Newton's second law,

$$m_p^* \frac{d\mathbf{v_p}^*}{dt^*} = \sum \mathbf{F}^*(t^*). \tag{2.16}$$

The particle forcing term, $\mathbf{F}^*(t^*)$, has been the subject of many studies [28, 105, 117, 97]. In most practical cases for which the density of the particle is much greater than the density of the carrier phase and the particle size is much smaller than the smallest length scale of the carrier phase, it suffices to consider the quasi-steady viscous drag force of the particle. With particles assumed spherical, we take the drag as the Stokes drag [158],

$$\mathbf{F}^*_{\mathbf{Stokes}} = C_{D_s} 6\pi \rho^*_{fp} \nu^* a^*_p (\mathbf{v_f}^* - \mathbf{v_p}^*), \tag{2.17}$$

where $\mathbf{v_f}^*$ is the velocity of the gas at the particle position, ν^* is the kinematic viscosity of the fluid, ρ^*_{fp} the ratio of the particle to fluid densities and a^*_p is the particle radius.

The term C_{D_S} is a correction factor, which, in this study, corrects for high Reynolds and Mach numbers. By substituting the equation for particle mass, $m_p^* = 4\pi a_p^{*3}\rho_p^*/3$, and the Stokes drag into Equation 2.16, the governing equation for particle velocity becomes,

$$\frac{d\mathbf{v_p}^*}{dt^*} = C_{D_s}\frac{18\nu^*\rho_{fd}^*}{\rho_d^* d_d^{*2}}\left(\mathbf{v_f}^* - \mathbf{v_p}^*\right). \tag{2.18}$$

The term on the right hand side describes the particle acceleration resulting from the velocity difference between the particle and the gas.

2.2.2 Non-Dimensional Form

Equations 2.15 and 2.18 are non-dimensioqnalized with the reference variables from (2.7). leading to the non-dimensional forms:

$$\begin{aligned} \frac{d\mathbf{x_p}}{dt} &= \mathbf{v_p}, \\ \frac{d\mathbf{v_p}}{dt} &= C_{D_s}\frac{\mathbf{v_f}-\mathbf{v_p}}{\tau_p}. \end{aligned} \tag{2.19}$$

The particle time constant $\tau_p = Re_f d_p^2 \rho_p/18$ is a measure for the reaction time of the particle to the changes in the carrier gas. Here, Re_f is a characteristic fluid Reynolds number used to compute the drag on a particle.

The empirical correction factor, C_{D_s}, was introduced by Boiko et. al. [15], adjusts the drag force to within 10% of measured particle acceleration for higher relative particle Reynolds number up to $Re_p = |\mathbf{v_f} - \mathbf{v_p}| d_p/\nu = 1\times 10^4$ and relative particle Mach number up to $M_f = |\mathbf{v_f} - \mathbf{v_p}|/\sqrt{T_f} = 1.2$ and is given by

$$C_{D_s} = \frac{3}{4}\left(24 + 0.38 Re_p + 4\sqrt{Re_p}\right)\left(1 + \exp\left[\frac{-0.43}{M_f^{4.67}}\right]\right). \tag{2.20}$$

From the first law of thermodynamics and Fourier's law for heat transfer, the equation for temperature is derived as,

$$\frac{dT_p}{dt} = \frac{1}{3}\frac{Nu}{Pr}\left(\frac{T_f - T_p}{\tau_p}\right), \tag{2.21}$$

where $Pr = 1.4$ is the Prandtl number, taken as its typical value for air in the following study. $Nu = 2 + \sqrt{Re_p}Pr^{0.33}$ is the Nusselt number corrected for high Reynolds number.

2.3 Source term S

Each particle generates a momentum and energy that affects the carrier flow. The volume averaged summation of all these contributions gives a continuum source contribution on the momentum and energy equation in (2.8) as:

$$\mathbf{S_m}(\mathbf{x}) = \sum_{i=1}^{N_p} \mathbf{K}(\mathbf{x_p}, \mathbf{x})\mathbf{W_m}, \tag{2.22}$$

$$\mathbf{S_e}(\mathbf{x}) = \sum_{i=1}^{N_p} \mathbf{K}(\mathbf{x_p}, \mathbf{x})(\mathbf{W_m} \cdot \mathbf{v_p} + \mathbf{W_e}), \tag{2.23}$$

where $\mathbf{K}(x,y) = \mathbf{K}(|x-y|)/V$ is a normalized distribution function that distributes the influence of each particle onto the carrier flow. This distribution function should ideally be a dirac delta function but is most often a regularized delta function [137]. $\mathbf{W_m} = m_p f_1(\mathbf{v_f} - \mathbf{v_p})/\tau_p$ and $\mathbf{W_e} = m_p(Nu/(3Pr))\,(T - T_p)\,/\tau_p$ are weigh functions describing the momentum and energy contribution of one particle, respectively. The term m_p is the mass of one spherical particle which can be derived from τ_p. N_p is the total number of particles in a finite volume V.

Chapter 3

Numerical Scheme

In this section, the high-order EL numerical method employed by this thesis is expounded upon. First, the WENO-Z scheme, introduced by Borges et. al. [18], is summarized in Section 3.0.1, which is reprinted from [32]. The particle solver is then described in more detail in the following sections.

3.0.1 Carrier Phase Solver

The advective fluxes in the carrier flow equations (2.8) are discretized spatially with the high order improved WENO conservative finite difference scheme (WENO-Z [18, 35]) on a uniform mesh. Time integration is performed with a third order Runge-Kutta TVD scheme.

For simplicity, we describe a generic one dimensional formulation of the method when applied to hyperbolic conservation laws in the form of

$$\frac{\partial \mathbf{u}}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{F}(\mathbf{u}) = 0. \tag{3.1}$$

The two-dimensional formulation follows naturally by tensor extension.

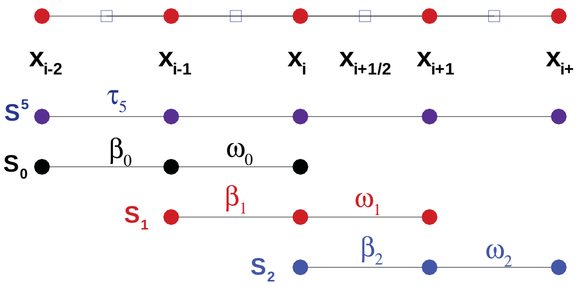


Figure 3.1: The uniformly spaced Eulerian grid, with cell centers x_i and cell boundaries $x_{i+\frac{1}{2}}$, and the 5-points stencil S^5, composed of three 3-points substencils S_0, S_1, S_2, used in the fifth-order WENO reconstruction step.

Consider a uniformly spaced grid, consisting of cell centers $x_i = i\Delta x$ and cell boundaries $x_{i+\frac{1}{2}} = x_i + \frac{1}{2}[\Delta x]$, $i = 0, \ldots, N$, where Δx is the uniform cell size (see Fig. 3.1). The conservative semi-discretized form of (3.1), by the method of lines, yields a system of ordinary differential equations

$$\frac{du_i(t)}{dt} = -\frac{1}{\Delta x}\left(h_{i+\frac{1}{2}} - h_{i-\frac{1}{2}}\right), \tag{3.2}$$

where $u_i(t)$ is a cell-averaged of $u(x,t)$ at x_i, and $h_{i\pm\frac{1}{2}} = h(x_{i\pm\frac{1}{2}})$ is the flux function $h(x)$ at $x_{i\pm\frac{1}{2}}$.

At the cell faces, $h_{i\pm\frac{1}{2}}$ are interpolated from the known flux function $F(x)$ at the cell centers, $f_i = F(x_i)$. A fifth order WENO scheme uses a five point global stencil S^5, which is subdivided into three substencils $\{S_0, S_1, S_2\}$ with each substencil containing three grid cells. The fifth degree polynomial approximation $\hat{f}_{i\pm\frac{1}{2}} = h_{i\pm\frac{1}{2}} + O(\Delta x^5)$ is constructed through a convex combination of the interpolated values of the second order interpolation polynomial $\hat{f}^k(x)$ at $x = x_{i\pm\frac{1}{2}}$, multiplied by the nonlinear weights, ω_k. The weights are defined in each one of the substencils S_k:

$$\hat{f}_{i\pm\frac{1}{2}} = \sum_{k=0}^{2} \omega_k \hat{f}^k(x_{i\pm\frac{1}{2}}), \tag{3.3}$$

and $\hat{f}^k(x)$ at the cell boundary $x_{i+\frac{1}{2}}$ in substencil S_k can be found as

$$\hat{f}^k(x_{i+\frac{1}{2}}) = \hat{f}^k_{i+\frac{1}{2}} = \sum_{j=0}^{2} c_{k,j} f_{i-k+j}, \quad i = 0, \ldots, N, \tag{3.4}$$

where the $c_{k,j}$ are Lagrangian interpolation coefficients [129].

The regularity (smoothness) of $\hat{f}^k(x)$ in the substencil S_k is measured by the lower order local smoothness indicators, β_k, which are given by

$$\beta_k = \sum_{l=1}^{2} \Delta x^{2l-1} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left(\frac{d^l}{dx^l} \hat{f}^k(x) \right)^2 dx. \tag{3.5}$$

On a substencil, S_k, containing a discontinuity, the β_k term is of order $O(1)$ whereas in smooth regions, all β_k are approximately equal and are of order $O(\Delta x^2)$ [35].

One can build a global optimal order smoothness indicator, τ_5 for a fifth order WENO scheme, as a linear combination of β_k, that is,

$$\tau_5 = |\beta_0 - \beta_2|, \tag{3.6}$$

with a leading order truncation error of $O(\Delta x^5)$ [35].

Using β_k and τ_5, the unnormalized and normalized nonlinear weights α_k^z and ω_k^z are defined as

$$\alpha_k^z = d_k \left(1 + \left(\frac{\tau_5}{\beta_k + \varepsilon} \right)^p \right), \quad \omega_k^z = \frac{\alpha_k^z}{\sum_{j=0}^{2} \alpha_j^z}, \quad k = 0, \ldots, r-1, \tag{3.7}$$

where d_k are the ideal weights so that if the function is smooth, $\omega_k^z = d_k$ and the scheme becomes a fifth order central finite difference scheme. The sensitivity parameter $\varepsilon = 10^{-10}$ is chosen in this study to avoid division by zero. The power parameter, p, amplifies the separation of scales between the smooth and non-smooth substencils. In the presence of a discontinuity, a larger p increases the numerical dissipation of the scheme near the discontinuity by decreasing the weight of the substencil containing the shock as compared to the smooth substencils. Because the substencil containing the discontinuity is given essentially zero weight, the method becomes more biased towards a second order upwinding ENO scheme [35].

At each grid point, the first order Lax-Friedrichs flux splitting is used as the low order building block to split the Euler flux, ignoring the source term, into the positive and negative fluxes. The positive and negative fluxes are then decomposed into the characteristic variables via the left eigenvectors and eigenvalues of the Euler flux, which are obtained via the linearized Riemann solver of Roe [115]. The characteristic variables are then reconstructed via the improved high order weighted essentially non-oscillatory (WENO-Z) scheme as discussed above. The reconstructed characteristic variables are then projected back into the physical space as the numerical flux via the right eigenvectors (see Castro et al. [18] or Shu et al. [129] for further details.)

3.1 Particle Tracing Algorithm

Simulation of the particle phase is performed with Lagrangian tracing of individual computational particle paths using the governing equations from Section 2.2. Lagrangian tracking of the particles consists of three stages per particle, including searching the element a particle is located in, interpolating the field variables to the particle location, and pushing the particle forward with a time integration method matching the

WENO integration scheme.

Locating the Eulerian cell in which the particle resides, $\mathbf{i_p}$, is computed on the structured grid,

$$\mathbf{i_p} = INT\left[\frac{\mathbf{x_p} - \mathbf{X_0}}{\mathbf{X_N} - \mathbf{X_0}}\right](\mathbf{i_N} - \mathbf{i_0}) + \mathbf{i_0} \tag{3.8}$$

where $\mathbf{x_p}$ are the particle locations, $\mathbf{X_0}$ and $\mathbf{X_N}$ are the lower and upper bounds of the domain respectively and $\mathbf{i_0}$ and $\mathbf{i_N}$ are the first and last cell node indices. The *INT* function truncates the bracketed quotient to an integer, causing the searching algorithm to the choose the lower host cell node and thereby creating an erroneous numerical bias in the solution. For most cases, the grid cells are small enough that this numerical error is negligible. However, for symmetric problems, the host cells below the centerline are adjusted by adding one to $\mathbf{i_p}$.

Following Jacobs and Hesthaven [67], to avoid aliasing errors and an unphysical numerical total energy increase, the order of the interpolation scheme must match the order of the solver in the fluid phase. The time integration method of the particle phase governing equations has to equal the the carrier phase solver as well. To determine the field variables at the particle location the ENO interpolation introduced by Jacobs and Don [68] suited to flows containing shock discontinuities is used. ENO interpolation [56] was shown to prevent Gibbs oscillations which often plague the accuracy of the centered interpolation over shocks.

In smooth flow areas without shocks, the WENO-Z method uses a central difference scheme. Under these conditions, centered interpolation to the particle position is more accurate and therefore preferred. Lagrange interpolating polynomials of degree k,

$$\mathbf{P}_k(x_p) = \sum_{i=i_p-k/2}^{i_p+k/2} \mathbf{Q}(x_i) l_i(x_p), \tag{3.9}$$

where i_p represents the nearest node to the particle position found in Equation 3.8 are used to compliment the WENO scheme. The number of points k should be equal to the number of points used as the order of the WENO scheme [67]. In the case of the fifth order WENO scheme the Lagrange interpolating polynomials would be of degree $k = 5$.

In shocked regions the centered interpolation will produce undesirable Gibbs oscillations. With an ENO interpolation [129], these oscillations are essentially removed. ENO interpolation is only necessary in WENO-domains identified by the smoothness indicator. In those domains, the interpolating points are determined based on the smoothness of the function indicated by the divided differences. The k-th degreed divided differences are determined first.

The 0-th order divided differences of Q are defined by:

$$Q[x_i] \equiv Q(x_i). \tag{3.10}$$

The j-th degree divided difference for $j \geq 1$ are defined by

$$Q[x_i, \cdots, x_{i+j}] \equiv \frac{Q[x_{i+1}, \cdots, x_{i+j}] - Q[x_i, \cdots, x_{i+j-1}]}{x_{i+j} - x_i}. \tag{3.11}$$

Starting from a two point stencil, x_{i_p}, x_{i_p+1}, the interpolation stencil is expanded to k points based on a comparison of the divided differences of the the increasing order at i_p. The smallest second order divided differences at i_p of the two potential three point stencils $\min\left\{Q[x_{i_p-1}, x_{i_p}, x_{i_p+1}], Q[x_{i_p}, x_{i_p+1}, x_{i_p+2}]\right\}$ indicates the smoothest interpolation stencil and is therefore chosen. This procedure is repeated until a k point interpolant is found. The Lagrange interpolant in (3.9) then interpolates to the particle position.

In two dimensions, the same procedure can be used along the separate dimension on the tensor grid. The divided differences are determined along horizontal and vertical lines in the grid. In 3D, this method is extended further to include the additional dimension. A note on the effectiveness of this ‘method of lines’ approach will be discussed in Section 6.6. With the 1D approach outline above, we find the left most and bottom most grid point of the interpolation stencil with $k \times k$ points for each grid point in the domain.

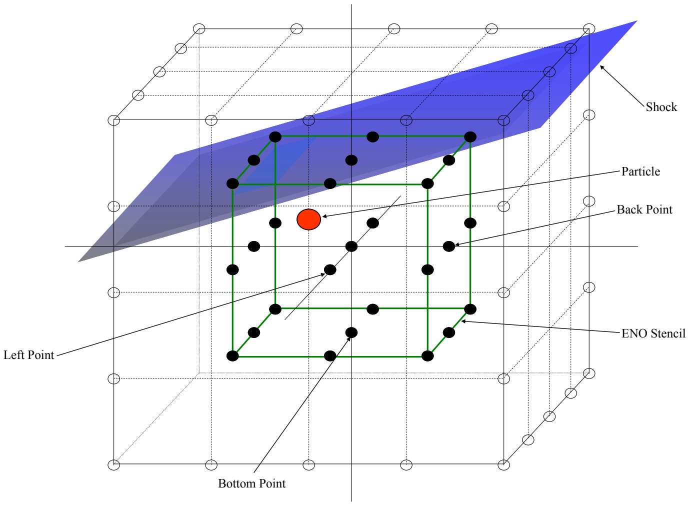


Figure 3.2: Three-dimensional ENO stencil for interpolation to a particle located near to a shock. The back, left and bottom points of the interpolation stencil were determined based on the divided differences along the x, y, and z grid lines at the particle’s nearest grid point.

We give an example of a three-dimensional ENO stencil in Figure 3.2. The particle’s nearest grid point is found to the back, bottom, and left of the particle. The back, bottom and left points of the ENO stencil are determined by comparison of the divided difference along the x, y, and z gridlines crossing the nearest grid point. If a particle is located in a cell with a shock, then the ENO is one-sided to the back, bottom and left of the particle. We note that if two shocks cross the k interpolation stencil, then this procedure will fail to recognize the second shock. This is, however, mostly a rare short-lived event and can be adapted by using lower-order ENO interpolation if necessary.

3.2 Evaluating the Source Term in the Eulerian Phase

To determine the particle influence on the carrier flow from Equation 2.23, the high-order spline interpolation discussed in [2] is used. The high-order weighing reduces aliasing and noise in the source terms that couple the particles to the gas flow. The spline S_k is constructed by the convolution of the square nearest-grid-point or zero order weighting function. For large k the spline approaches a Gaussian function. The 0^{th} mode of the function in wave space is free of aliasing errors and the higher components of the function in wave space are smaller than the equivalent Lagrangian interpolations.

3.3 Parallel Implementation

One additional advantage of the EL method is that it does not employ global numerical techniques and is therefore straightforward to parallelize. Information is passed through the element faces and only local stencils are needed for computation. These features make the method very suitable for parallel implementation, since the computational domain can be naturally split into smaller subdomains and allocated to different processors. Particle paths can be trivially mapped by switching the particles on or off as they enter or leave a domain. Message passing, which is accomplished using the Message Passing Interface (MPI) library, is restricted to the small number of ghost cells needed on the boundaries of a subdomain and the passage of particles across interprocessor interfaces. The focus of this section will be on the parallel implementation of the particle routines, for more information on the WENO libraries used see [18, 94, 92, 68, 21].

Because the computation of the fluid governing equations is much more computationally expensive, the dispersed phase is partitioned according to the carrier phase subdomains i.e. particle governing equations are computed according to the processors on which their accompanying domains are allocated. Using this method, the particle and carrier phase information does not need to be passed across processors to track particles or to compute the particle source term in the fluid governing equations. An obvious drawback of dividing the dispersed phase along uniform subdomains is the high probability that the particle load balance per processor will not be uniformly spread. However, the number of operations needed to compute the fluxes and integrate each discrete carrier phase point significantly outweighs the computational resources required to track a particle.

Searching for a particle within a grid partition is done trivially on the uniform grid. The information required for the exchange of particles between processors is stored in a 'particle' matrix accessible locally and a global 'particle' matrix available to all processors. The two matrices share the information needed for particles to be exchanged:

- the particle number,
- the processor the particle is currently allocated on,
- the processor the particle is being exchanged to.

The local matrix contains information about each of the particles leaving the host processor while the global matrix contains assembly information for all of the local matrices at each iteration.

To exchange particles, the local matrix is first determined on each processor for the total number of particles leaving as follows:

1. The subdomain each particle resides on in determined.
2. If the particle is located on another subdomain than the host processor, it is determined which subdomain it has crossed into. This is a trivial task on a structured grid.

3. The particle number, the current host processor and the new processor are saved in the local particle matrix.

Next, the MPI_ALLGATHER operation compiles all local particle matrices into the global particle matrix. This operation is the most computationally expensive part of the parallel process because it requires the stoppage of all processes and the passing of data to all processors. Currently, the local ‘particle’ matrix is the same size for all processors and the global matrix is the size of each local matrix times the number of processors but it is noted that a variable sized gather operation would be more efficient.

Following the gather operation, the global ‘particle’ matrix is sorted to place all particles moving between two processors in sequential order in the global particle matrix. With the sorted global matrix, the stored information (location, velocity, temperature, etc.) for all particles that move between processors are exchanged with non-blocking send, MPI_ISEND, and non-blocking receive, MPI_IRECV, operations. These routines pass messages between two processors without blocking other operations from being performed, such as the updating of all particles not being sent/received.

Chapter 4

Subgrid Particle Averaged Reynolds Stress Equivalent Model

This chapter derives and presents *a priori* testing on the Subgrid Particle Averaged Reynolds Stress Equivalent (SPARSE) model. It is not computationally feasible to trace every particle in the domain for process-scale shocked particle-laden flow problems. In CIC methods, groups of individual physical particles are modeled as single macro-scale computational particles [10]. Much like the modeling of the Reynolds stress in RANS turbulence computations, the SPARSE model includes fluctuation terms in the particle momentum governing equation, which account for the influence of the sub-scale physical perturbations on the averaged macro-scale particle cloud velocity. By taking a Reynolds decomposition of the particle momentum equation and Reynolds averaging, a governing equation for the average particle velocity is derived. Additionally, the assumption in the CIC model that the fluid velocity can be interpolated at a single point is refuted and an average particle cloud fluid velocity is used in the computations. The first section presents a derivation of the SPARSE model. A description of the 1D *a priori* testing follows in Section 4.2.1, where the two enhanced components of SPARSE are shown to reduce errors in the modeling of a cloud of particles with a single computational particle. Finally, in Section 4.2.2, a full three dimensional isotropic turbulence case is studied to examine the efficacy of the SPARSE model.

4.1 Derivation of the Model

The particle governing equations from 2.19 are defined as,

$$\begin{aligned}\frac{d\mathbf{x_p}}{dt} &= \mathbf{v_p},\\ \frac{d\mathbf{v_p}}{dt} &= f(\mathbf{u}-\mathbf{v_p})\cdot(\mathbf{u}-\mathbf{v_p}),\end{aligned} \tag{4.1}$$

where $\mathbf{v_p}$ is the particle velocity vector, $\mathbf{u}$ is the fluid velocity vector and the particle correction factor, $f(\mathbf{u}-\mathbf{v_p}) = {}^{C_{Ds}}/_{\tau_p}$, is a function of $\mathbf{u}$ and $\mathbf{v_p}$ equivalent to the drag correction factor divided by the particle time constant. To simplify the derivation, the interphase velocity difference, $\mathbf{a}$, is defined as $\mathbf{a} = \mathbf{u} - \mathbf{v_p}$, or in vector notation, $a_i = u_i - v_i$, making the governing equations,

$$\begin{aligned}\frac{d\mathbf{x_p}}{dt} &= \mathbf{v_p},\\ \frac{d\mathbf{v_p}}{dt} &= f(\mathbf{a})\cdot\mathbf{a}.\end{aligned} \tag{4.2}$$

Under the assumption that a group of particles can be modeled with a single computational particle in CIC, only the averaged velocity, $\overline{\mathbf{v_p}}$, and averaged position, $\overline{\mathbf{x_p}}$, of the particles are computed, where the averaging is defined as

$$\overline{\eta} = \frac{1}{N}\sum_{i=1}^{N}\eta_i. \tag{4.3}$$

To find the average values, first a Reynolds decomposition, $\mathbf{g} = \overline{\mathbf{g}} + \mathbf{g}'$, is performed to split the velocities and positions into their average, $\overline{\mathbf{g}}$, and fluctuating, $\mathbf{g}'$, terms. By averaging the decomposed governing equations and noting that because of the linearity of the derivative operator,

$$\overline{\frac{d\eta}{dt}} = \frac{d\overline{\eta}}{dt}, \tag{4.4}$$

the computational particle velocity and position can be written,

$$\begin{aligned} \frac{d\overline{(\overline{\mathbf{x_p}} + \mathbf{x_p}')}}{dt} &= \overline{\overline{\mathbf{v_p}} + \mathbf{v_p}'}, \\ \frac{d\overline{(\overline{\mathbf{v_p}} + \mathbf{v_p}')}}{dt} &= \overline{f(\overline{\mathbf{a}} + \mathbf{a}') \cdot (\overline{\mathbf{a}} + \mathbf{a}')}, \end{aligned} \tag{4.5}$$

or

$$\begin{aligned} \frac{d\overline{\mathbf{x_p}}}{dt} &= \overline{\mathbf{v_p}}, \\ \frac{d\overline{\mathbf{v_p}}}{dt} &= \overline{f(\overline{\mathbf{a}} + \mathbf{a}') \cdot (\overline{\mathbf{a}} + \mathbf{a}')}. \end{aligned} \tag{4.6}$$

Here, $\overline{\mathbf{a}} = \overline{\mathbf{u}} - \overline{\mathbf{v_p}}$ where $\overline{\mathbf{u}}$ is the average of the fluid velocity at all particle locations and $\mathbf{a}' = \mathbf{u}' - \mathbf{v_p}'$ where $\mathbf{u}'$ are the fluid velocity fluctuations at the particle positions and $\mathbf{v}'_\mathbf{p}$ are the fluctuations in the particle phase. Traditionally, the average of the correction factor term has been taken as the correction factor for the average interphase velocity,

$$\overline{f(\overline{\mathbf{a}} + \mathbf{a}')} = f(\overline{\mathbf{a}}),$$

leading to the classical first order model,

$$\begin{aligned}\frac{d\overline{\mathbf{x_p}}}{dt} &= \overline{\mathbf{v_p}},\\ \frac{d\overline{\mathbf{v_p}}}{dt} &= f(\overline{\mathbf{a}})\cdot(\overline{\mathbf{a}}),\end{aligned} \tag{4.7}$$

where the average fluid velocity is taken as the fluid velocity at the average particle position, $\overline{\mathbf{u}} = \mathbf{u}(\overline{\mathbf{x_p}})$ [28]. Equation 4.7 is a first order model under the assumptions of CIC because the correction factor, f, is a first order truncation of the full Taylor series expansion of the correction factor.

To be consistent with the high-order methods used in the EL solver, a higher order model for the average particle governing equations is required. Additional terms from the Taylor expansion of the correction factor, $f(\mathbf{a})$, around $\overline{\mathbf{a}}$, where $\mathbf{a} = (a_1, a_2, a_3) = (\overline{a_1} + a'_1, \overline{a_2} + a'_2, \overline{a_3} + a'_3)$, must be taken,

$$\begin{aligned}f(\mathbf{a}) =& f(\overline{\mathbf{a}}) + \frac{df(\overline{\mathbf{a}})}{da_1}((\overline{a_1} + a'_1) - \overline{a_1}) + \frac{df(\overline{\mathbf{a}})}{da_2}((\overline{a_2} + a'_2) - \overline{a_2})...\\ &+ \frac{df(\overline{\mathbf{a}})}{da_3}((\overline{a_3} + a'_3) - \overline{a_3}) + O(a'^2),\\ =& f(\overline{\mathbf{a}}) + \frac{df(\overline{\mathbf{a}})}{da_1}a'_1 + \frac{df(\overline{\mathbf{a}})}{da_2}a'_2 + \frac{df(\overline{\mathbf{a}})}{da_3}a'_3 + O(a'^2).\end{aligned} \tag{4.8}$$

Assuming that the fluctuations in the interphase velocity difference are small, the expansion is truncated at $O(a'^2)$.

Using (4.8) in the particle momentum equation (4.2) yields,

$$\begin{aligned}\frac{dv_1}{dt} &= (f(\overline{\mathbf{a}}) + \frac{df(\overline{\mathbf{a}})}{da_1}a'_1 + \frac{df(\overline{\mathbf{a}})}{da_2}a'_2 + \frac{df(\overline{\mathbf{a}})}{da_3}a'_3)(\overline{a_1} + a'_1),\\ \frac{dv_2}{dt} &= (f(\overline{\mathbf{a}}) + \frac{df(\overline{\mathbf{a}})}{da_1}a'_1 + \frac{df(\overline{\mathbf{a}})}{da_2}a'_2 + \frac{df(\overline{\mathbf{a}})}{da_3}a'_3)(\overline{a_2} + a'_2),\\ \frac{dv_3}{dt} &= (f(\overline{\mathbf{a}}) + \frac{df(\overline{\mathbf{a}})}{da_1}a'_1 + \frac{df(\overline{\mathbf{a}})}{da_2}a'_2 + \frac{df(\overline{\mathbf{a}})}{da_3}a'_3)(\overline{a_3} + a'_3).\end{aligned} \tag{4.9}$$

Algebraically expanding,

$$
\begin{aligned}
\frac{dv_1}{dt} =& f(\bar{\mathbf{a}})\overline{a_1} + \frac{df(\bar{\mathbf{a}})}{da_1}\overline{a_1}a_1' + \frac{df(\bar{\mathbf{a}})}{da_2}\overline{a_1}a_2' + \frac{df(\bar{\mathbf{a}})}{da_3}\overline{a_1}a_3' ... \\
& f(\bar{\mathbf{a}})a_1' + \frac{df(\bar{\mathbf{a}})}{da_1}a_1'a_1' + \frac{df(\bar{\mathbf{a}})}{da_2}a_1'a_2' + \frac{df(\bar{\mathbf{a}})}{da_3}a_1'a_3', \\
\frac{dv_2}{dt} =& f(\bar{\mathbf{a}})\overline{a_2} + \frac{df(\bar{\mathbf{a}})}{da_1}\overline{a_2}a_1' + \frac{df(\bar{\mathbf{a}})}{da_2}\overline{a_2}a_2' + \frac{df(\bar{\mathbf{a}})}{da_3}\overline{a_2}a_3' ... \\
& f(\bar{\mathbf{a}})a_2' + \frac{df(\bar{\mathbf{a}})}{da_1}a_1'a_2' + \frac{df(\bar{\mathbf{a}})}{da_2}a_2'a_2' + \frac{df(\bar{\mathbf{a}})}{da_3}a_2'a_3', \\
\frac{dv_3}{dt} =& f(\bar{\mathbf{a}})\overline{a_3} + \frac{df(\bar{\mathbf{a}})}{da_1}\overline{a_3}a_1' + \frac{df(\bar{\mathbf{a}})}{da_2}\overline{a_3}a_2' + \frac{df(\bar{\mathbf{a}})}{da_3}\overline{a_3}a_3' ... \\
& f(\bar{\mathbf{a}})a_3' + \frac{df(\bar{\mathbf{a}})}{da_1}a_1'a_3' + \frac{df(\bar{\mathbf{a}})}{da_2}a_2'a_3' + \frac{df(\bar{\mathbf{a}})}{da_3}a_3'a_3',
\end{aligned} \tag{4.10}
$$

and averaging yields the second order model,

$$
\begin{aligned}
\frac{d\overline{v_{p1}}}{dt} =& f(\bar{\mathbf{a}})\overline{a_1} + \frac{df(\bar{\mathbf{a}})}{da_1}\overline{a_1'a_1'} + \frac{df(\bar{\mathbf{a}})}{da_2}\overline{a_1'a_2'} + \frac{df(\bar{\mathbf{a}})}{da_3}\overline{a_1'a_3'}, \\
\frac{d\overline{v_{p2}}}{dt} =& f(\bar{\mathbf{a}})\overline{a_2} + \frac{df(\bar{\mathbf{a}})}{da_1}\overline{a_1'a_2'} + \frac{df(\bar{\mathbf{a}})}{da_2}\overline{a_2'a_2'} + \frac{df(\bar{\mathbf{a}})}{da_3}\overline{a_2'a_3'}, \\
\frac{d\overline{v_{p3}}}{dt} =& f(\bar{\mathbf{a}})\overline{a_3} + \frac{df(\bar{\mathbf{a}})}{da_1}\overline{a_1'a_3'} + \frac{df(\bar{\mathbf{a}})}{da_2}\overline{a_2'a_3'} + \frac{df(\bar{\mathbf{a}})}{da_3}\overline{a_3'a_3'}.
\end{aligned} \tag{4.11}
$$

The particle phase stress terms, or "Reynolds Stress Equivalent" terms, $\overline{a_1'a_1'}$, $\overline{a_1'a_2'}$, $\overline{a_1'a_3'}$, $\overline{a_2'a_2'}$, $\overline{a_2'a_3'}$ and $\overline{a_3'a_3'}$ require a modeling effort which is beyond the scope of this thesis. Future directions for the closure of these terms will be discussed further in Section 9.3. In the present study, these terms will be closed using an *a priori* approach where the perturbations of the physical particles are used directly in the SPARSE model.

The second important feature of the SPARSE model is the approximation of the average fluid velocity through the entire cloud. Classically, the fluid velocity used in the particle governing equation is interpolated at the modeled particle location. However, the average fluid velocity in the cloud is not equal to the fluid velocity at the averaged particle location, $\overline{u_{fluid}(\mathbf{x})} \neq u_{fluid}(\overline{\mathbf{x}})$. A model governing the average cloud behavior must therefore account for changes in the fluid throughout the volume of the approximated particle cloud. This term is once again closed in an *a priori* manner in the following numerical studies.

4.2 Numerical Experiments

The accuracy of a single computational particle modeling the average motion of a cloud containing thousands of physical particles is analyzed. Single particle velocities and positions, computed with the first order and SPARSE models, are compared with the average velocities and positions of a cloud of thousands of particles as they interact with the fluid field. In the following sections, the computations tracking the full physical cloud of particles will be referred to as the “exact” case. Errors are characterized by the ability of the modeled particle to compute the average of the exact particle locations and velocities over time. Sub-scale particle perturbations and average fluid velocities needed for closure in the SPARSE model are taken at each time step from the exact particle cloud statistics in an *a priori* fashion. In the future, these models will be derived using statistics from micro-scale computations as described in Section 5.

4.2.1 One Dimensional Analysis of SPARSE in Frozen Fluid Field

The one-dimensional temporal evolution of a cloud of particles through a frozen fluid phase is analyzed. To isolate errors in the modeling of the particle cloud dynamics, the background fluid velocity is governed by an explicit function, independent of the particle phase. The exact case computes the interaction of 10,000 particles with the fluid, which the first and SPARSE models trace only a single particle. Average velocities and locations for the exact particle cloud are then compared with the modeled computational particle location and velocity. The first order model in 1D is defined by,

$$\begin{aligned}
\frac{d\overline{x_p}}{dt} &= \overline{v_p}, \\
\frac{d\overline{v_p}}{dt} &= f(\overline{a}) \cdot \overline{a},
\end{aligned} \tag{4.12}$$

and the SPARSE model simplifies to

$$\begin{aligned}
\frac{d\overline{x_p}}{dt} &= \overline{v_p}, \\
\frac{d\overline{v_p}}{dt} &= f(\overline{a}) \cdot \overline{a} + \frac{df(\overline{a})}{da}\overline{a'a'}.
\end{aligned} \tag{4.13}$$

Physical particles are initialized at the locations and velocities,

$$\begin{aligned}
x_p &= 5.0 + \sigma_x \cdot d_{cloud}, \\
v_p &= 5.0 + \sigma_v \cdot \gamma,
\end{aligned} \tag{4.14}$$

where d_{cloud} is the cloud diameter, set to 1.0, γ is the maximum amplitude of the initial velocity perturbations and σ_i is an evenly distributed random number such that $-1.0 \leq \sigma_i \leq 1.0$. The initial velocity perturbations are set to a reference scale on the order of the average fluid velocity, $\gamma = 10$. The modeled particle position and velocity are initialized as the mean of the exact locations and velocities. Errors in the particle phase shown in the following analysis are calculated as

$$\begin{aligned}
\varepsilon_{x_p} &= \frac{|\overline{x_{p,exact}} - x_{p,model}|}{\overline{x_{p,exact}}}, \\
\varepsilon_{v_p} &= \frac{|\overline{v_{p,exact}} - v_{p,model}|}{\overline{v_{p,exact}}},
\end{aligned} \tag{4.15}$$

where ε_{x_p} is the error in the modeled particle position and ε_{v_p} is the error in the modeled particle velocity. Time integration is performed using an first order explicit time stepping routine with a time step of $dt = 10^{-5}$, where modeling errors dominate. The fluid velocities at exact and first order model particle positions are computed directly using the fluid velocity equations.

The SPARSE model improves upon existing point source particle modeling in two ways: inclusion of the sub-scale particle fluctuations and the modeling the fluid velocity through the entire cloud rather than at a single point. Differing flow fields are presented to highlight the importance of including each of these components. Improvements to the SPARSE model with the inclusion of the Taylor expansion of the sub-scale perturbation terms is highlighted first followed by the analysis of errors stemming from the assumption of the average fluid velocity through a cloud being interpolated at a single point.

Effect of the Reynolds Stress Equivalent Terms

The importance of including the sub-scale particle fluctuations is shown by using a constant fluid velocity,

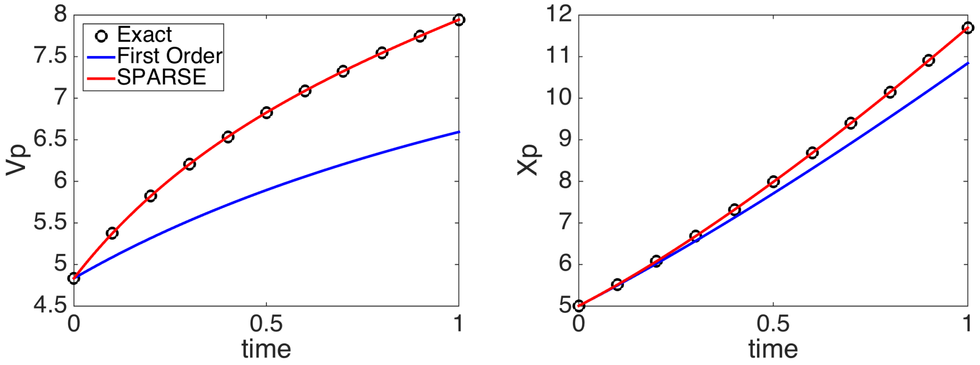
$$u_{fluid}(x,t) = 10, \tag{4.16}$$

to eliminate errors from the assumption in the first order model that $\overline{u_{fluid}(x)} = u_{fluid}(\bar{x})$. This profile is chosen to analyze only that error caused by truncating the correction factor in the first order model at $O(a')$ and thereby eliminating terms which include information about the sub-scale particle perturbations.

Any differences between the average velocity and location of the exact particle cloud and the modeled particles' velocity and location arises because of the truncation of the Taylor series expansion of the particle correction factor. To highlight this error, the particle correction factor, $f(a)$, is set to a linearly dependent function of a,

$$f(a) = \frac{0.1a}{St}, \tag{4.17}$$

where St is the particle Stokes number, set to 1.0 in the following numerical experiments. This model is chosen to be on a similar scale with other low Reynolds number correction factors such as the one proposed by Schiller and Naumann [122]. Because the correction factor is linearly dependent on the interphase velocity difference, all second order or higher derivatives are zero, $d^2f/da^2 = 0$. The Taylor expansion of the correction factor in the SPARSE model is therefore exact, leading to a modeling error in the SPARSE computation on the order of machine precision. Analyzing the computed particle cloud velocities and locations in Figure 4.1, the exact and modeled cases are collinear. The error profiles for the velocity and location, Figure 4.2a and 4.2b respectively, shows the importance of including the second order term in the Taylor expansion of the correction factor. Truncating the expansion at $O(a')$ causes an error, which rises to 17.1% at time $t = 1$ whereas the SPARSE model has an error on the order of 10^{-15}.



(a) (b)

Figure 4.1: Particle (a) velocities and (b) locations using the mean of the exact particle locations as well as the first order and SPARSE models with a uniform background fluid velocity and a linear correction factor.

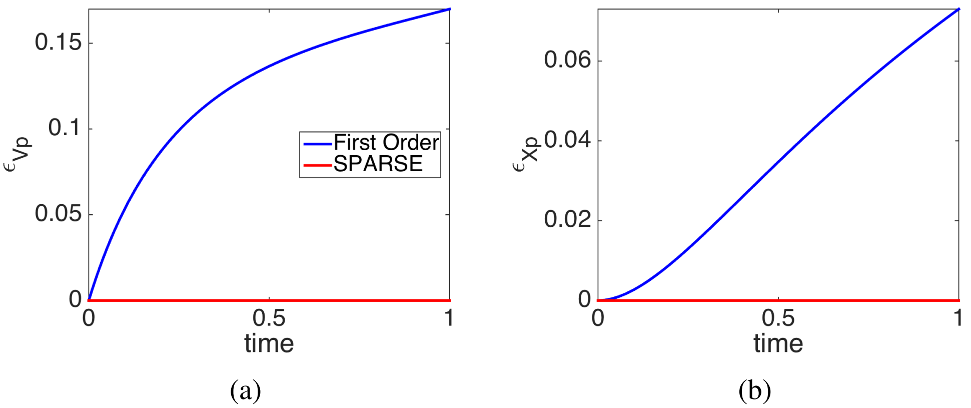
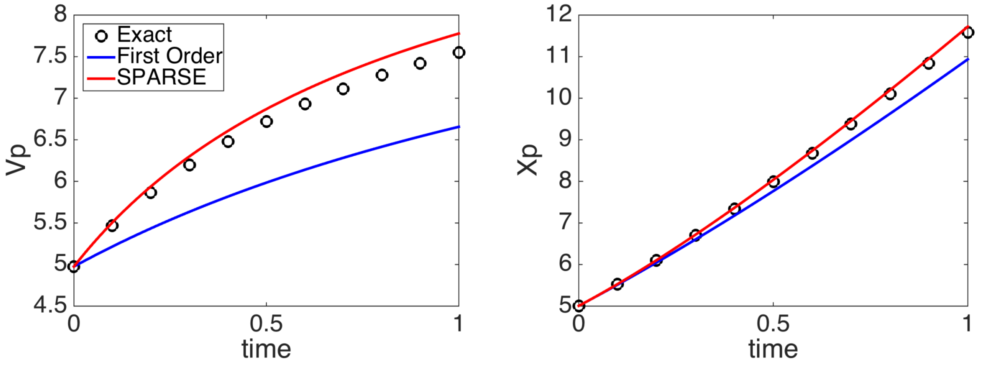


Figure 4.2: Modeling error of the mean particle (a) velocity and (b) location using the first order and SPARSE models with a uniform background fluid velocity and a linear correction factor.

In many physical applications, the particle correction factor is a function of the Reynolds number, which in turn is a function of the absolute value of the interphase velocity difference, a [28]. Using a correction factor of

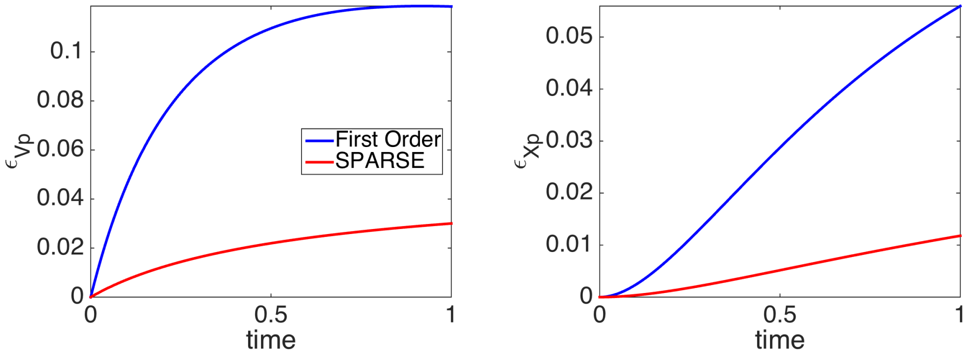
$$f(a) = \frac{0.1|a|}{St}, \tag{4.18}$$

provides a more realistic computation of the particle-fluid interaction and demonstrates the improved accuracy of the SPARSE model with a non-linear correction factor. The first order model does not account for the sub-scale kinetic energy in the particle phase, causing the modeled particle to lag behind the average particle position of the exact cloud, as seen in Figure 4.3. As the particle velocity approaches the constant fluid velocity, the derivative of the correction factor is discontinuous for $a = 0$ causing errors in the modeling of the average particle velocity with the SPARSE model, as is evident in Figure 4.4. However, because the SPARSE model accounts for the perturbations in the particle phase, the computed velocity is nearly 10% more accurate at time $t = 1$.



(a) (b)

Figure 4.3: Particle (a) velocities and (b) locations using the mean of the exact particle locations as well as the first order and SPARSE models with a uniform background fluid velocity and an absolute value correction factor.



(a) (b)

Figure 4.4: Modeling error of the mean particle (a) velocity and (b) location using the first order and SPARSE models with a uniform background fluid velocity and an absolute value correction factor.

Effect of Modeling the Average Cloud Velocity

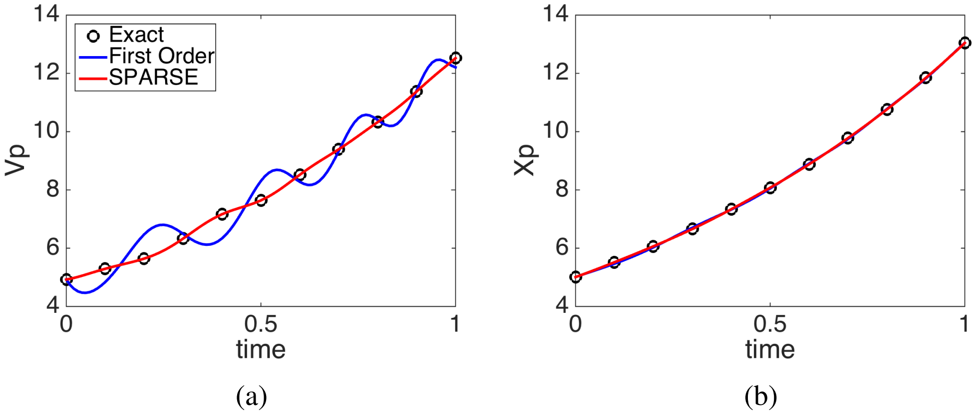
The use of a constant correction factor, such at Stokes' drag [136],

$$f(a) = \frac{24}{St}, \tag{4.19}$$

eliminates truncation errors in the Taylor expansion of the correction factor in the first order model because the first derivative of $f(a)$ is zero. Combining the constant correction factor and a spatially varying fluid velocity,

$$u_{fluid} = \cos(\pi x) + x, \tag{4.20}$$

isolates the errors caused by approximating the fluid velocity throughout the entire cloud with a single point sample.



(a)

(b)

Figure 4.5: Particle (a) velocities and (b) locations using the mean of the exact particle locations as well as the first order and SPARSE models with a spatially varying background fluid velocity.

As stated in the derivation in Section 4, the erroneous assumption that $\overline{u_{fluid}(\mathbf{x_p})}$ = $u_{fluid}(\overline{\mathbf{x_p}})$ creates approximation errors in the first order numerical model. Errors as high as 16.6% are seen in the first order modeling of the fluid velocity in Figure 4.6a whereas the average velocity and location of the cloud is computed exactly with the SPARSE model. The perturbations in the fluid velocity are periodic so the errors in the modeled particle velocity are oscillatory and only cause a maximum 1% error in the particle location in Figure 4.6b. However, as will be shown in Section 4.2.2, the initial error in modeling the fluid velocity can cause the modeled particle to diverge entirely from the physical cloud's path when the perturbations are no longer periodic.

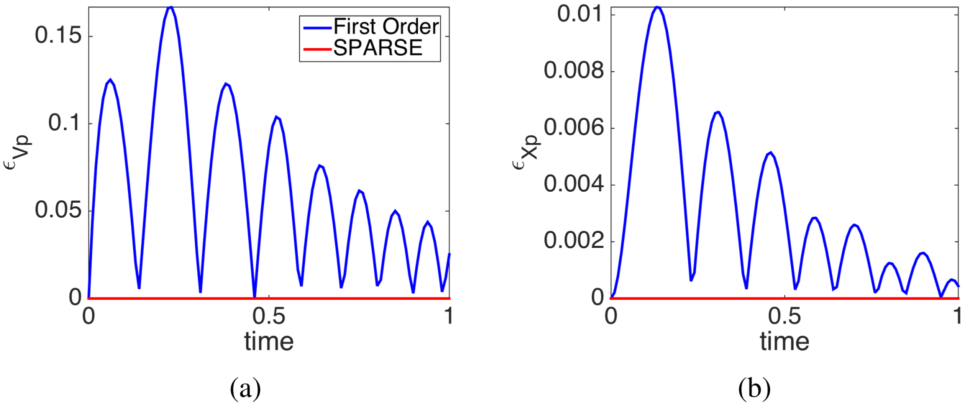


Figure 4.6: Modeling error of the mean particle (a) velocity and (b) location using the first order and SPARSE models with a spatially varying background fluid velocity.

4.2.2 Three Dimensional Analysis of SPARSE in Decaying Isotropic Turbulence

In this section, a simulation of decaying isotropic turbulence similar to the one performed in [11, 12] with a Fourier pseudo-spectral method is reproduced with the addition of passive particles using the high-order EL method. The particles' inertia is assumed small so the forcing from the particle phase does not influence the fluid phase i.e. the source term in the fluid governing equations is zero. The purpose of this test is twofold: validation of the fluid solver and testing of the SPARSE model. First, the time integration of initially isotropic turbulence is compared to literature. Next, the velocities in the reference case initialization are multiplied by a scaling factor of five to demonstrate the error in the first order model, particularly in the assumption that the fluid velocity needed to compute the particle trajectory can be approximated with the velocity at a single point in space.

Initial Conditions

The isotropic turbulence simulation is performed in a cube with periodic boundary conditions on all sides. Because of the absence of specified boundary conditions and, by definition of decaying turbulence, the lack of a forcing source term in the governing flow creates a difficulty in generating the turbulence. Following [116], Blaisdell et. al. [11] generated turbulence by specifying initial energy spectra that produce a correlated flow field with turbulent characteristics. Full details of this procedure can be found in [11] and are not repeated here. The specific equations used for the case Blaisdell et. al. refer to as “iga96” are, however, presented in the following.

Initial conditions are produced by specifying the solenoidal, E_s, and dilatational, E_d, velocity, density, E_ρ, and temperature, E_T, spectra. Each spectrum is initialized as a

Table 4.1: Initial amplitudes of the spectra for case "iga96" from Blaisdell et. al. [11].

Spectrum	Initial Value
E_s	2.8125×10^{-4}
E_d	3.1250×10^{-5}
E_ρ	1.2500×10^{-3}
E_T	1.2500×10^{-3}

tophat distribution with the amplitudes specified in Table 4.1, which span all wavenumbers between $k = 8$ and $k = 16$.

Using E_s, the Fourier coefficients for the solenoidal velocity field can be calculated as,

$$\begin{bmatrix} \hat{u}_s(k) \\ \hat{v}_s(k) \\ \hat{w}_s(k) \end{bmatrix} = \begin{bmatrix} \frac{\alpha^{ran} k k_2 + \beta^{ran} k_1 k_3}{k k_{12}} \\ \frac{\beta^{ran} k_2 k_3 - \alpha^{ran} k k_1}{k k_{12}} \\ \frac{-\beta^{ran} k_{12}}{k} \end{bmatrix}, \tag{4.21}$$

where α^{ran} and β^{ran} are set to

$$\alpha^{ran} = \sqrt{\frac{E_s(k)}{4\pi k^2}} e^{i\phi_1} \cos(\phi_3), \tag{4.22}$$

$$\beta^{ran} = \sqrt{\frac{E_s(k)}{4\pi k^2}} e^{i\phi_2} \sin(\phi_3). \tag{4.23}$$

The terms ϕ are uniformly distributed randomly generated numbers on the domain $[0, 2\pi]$, i is the imaginary number $\sqrt{-1}$, k is the magnitude of the wavenumber vector, (k_1, k_2, k_3), and k_{12} is $\sqrt{k_1^2 + k_2^2}$. If $k_{12} = 0$, then $\hat{u}_s = \alpha^{ran}$ and $\hat{v}_s = \beta^{ran}$. The dilation velocity Fourier coefficients are given as,

$$\begin{bmatrix} \hat{u}_d(k) \\ \hat{v}_d(k) \\ \hat{w}_d(k) \end{bmatrix} = \begin{bmatrix} \frac{\gamma^{ran} k_1}{k} \\ \frac{\gamma^{ran} k_2}{k} \\ \frac{\gamma^{ran} k_3}{k} \end{bmatrix}, \tag{4.24}$$

where

$$\gamma^{ran} = \sqrt{\frac{E_d(k)}{4\pi k^2}} e^{i\phi_4}. \tag{4.25}$$

The density and temperature Fourier coefficients are determined with,

$$\hat{\rho}(k) = \sqrt{\frac{E_\rho(k)}{4\pi k^2}} e^{i\phi_5}, \tag{4.26}$$

$$\hat{T}(k) = \sqrt{\frac{E_T(k)}{4\pi k^2}} e^{i\phi_6}. \tag{4.27}$$

The initialized variables are defined on the equidistant grid at the points,

$$x_i = \frac{(l_i - 1)L_f}{N}, \text{ where } l_i = 1, ..., N, \tag{4.28}$$

with the wavenumbers,

$$k_i = \frac{2\pi n_i}{L_f}, \text{ where } n_i = -N/2 + 1, ..., N/2, \tag{4.29}$$

for each direction, $i = 1, 2, 3$. The reference length, $L_f = 2\pi$, is used in the discrete Fourier transform using the coefficients from Equations 4.21, 4.24 and 4.27,

$$q(l_1, l_2, l_3) = \sum_{n_1=-N/2+1}^{N/2} \sum_{n_2=-N/2+1}^{N/2} \sum_{n_3=-N/2+1}^{N/2} \hat{q}(n_1, n_2, n_3) e^{2\pi i \left(\frac{n_1(l_1-1)}{N} + \frac{n_2(l_2-1)}{N} + \frac{n_3(l_3-1)}{N}\right)}. \tag{4.30}$$

The flow field conditions u, v, w, ρ and T are computed with a Fast Fourier Transform (FFT), which initializes a flow field correlated to the tophat spectrum on the uniform physical grid.

Computations

Computations are performed on a $128 \times 128 \times 128$ domain that spans 2π in each dimension, $(x,y,z) = (0:2\pi, 0:2\pi, 0:2\pi)$. The average initial density, $\overline{\rho}_0$, and temperature, $\overline{T}_0$ are set to 1.0, which leads to an average initial fluctuating Mach number, $M_0 = 0.05$ where

$$M_0 = \frac{\sqrt{\overline{u_i' u_i'}}}{c(\overline{T}_0)}, \tag{4.31}$$

where u_i' is the fluctuating fluid velocity and is written in index notation. The reference fluid Reynolds number is set to $Re_f = 2357$.

The decay of turbulent kinetic energy, TKE = $1/2\overline{u_i' u_i'}$, computed with the EL code (using a 4th order central difference method for the fluxes) is compared with the results from Blaisedell et. al. [11] in Figure 4.7 for the "iga96" case described above. The referenced case study used an $N = 96$ Fourier spectral method. The oscillatory behavior is a result of the exchange of energy between the kinetic and internal acoustic energy, also known as pressure-dilation [77]. The results agree well with the referenced solution.

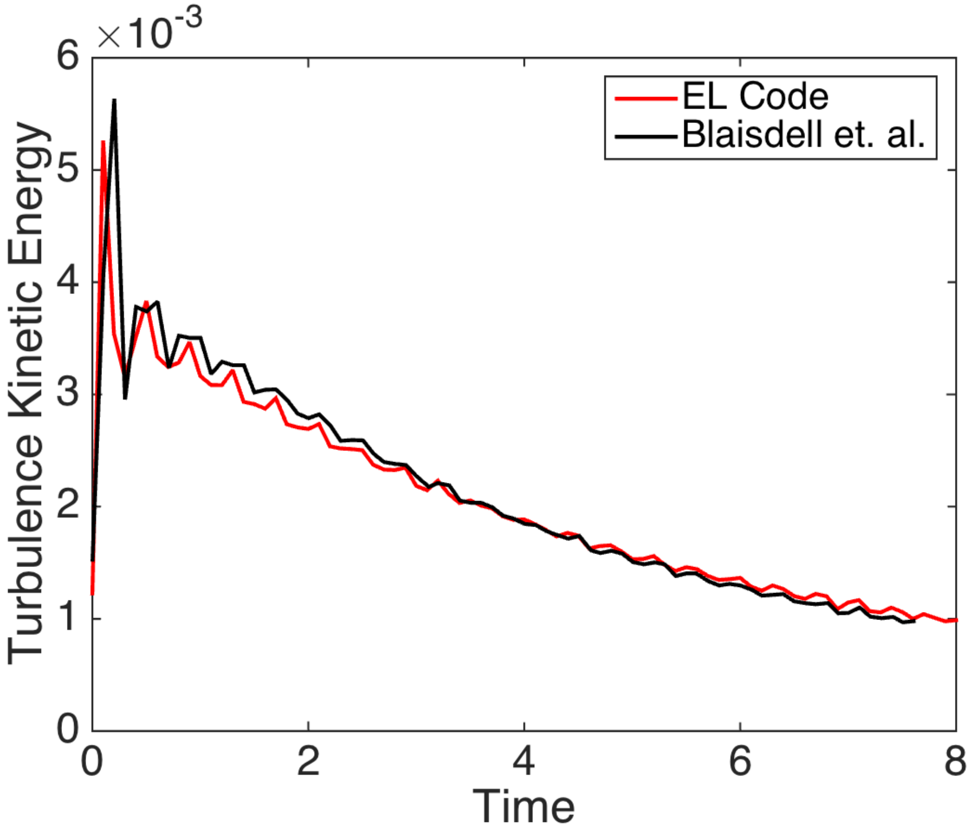


Figure 4.7: Comparison of the turbulent kinetic energy (TKE) versus time in isotropic turbulence using the high-order EL code and a Fourier spectral method performed by Blaisdell et. al. [11].

To compare the dynamics of the average particle cloud using the SPARSE and first order point particle models with the exact particle dispersion, a higher energy isotropic turbulence system is desired to increase the particle cloud dispersion and movement. The physical velocities computed with the spectral initialization are multiplied by a scaling factor of 5.0, raising the initial fluctuating Mach number to $M_0 = 0.25$. The kinetic energy transferred to the particle phase is increased to show greater particle spreading and the effects of modeling both the average fluid velocity through the cloud and the sub-scale particle perturbations.

Physical particles are uniformly initialized over a domain stretching approximately 3 grid cells in each direction with $30^3 = 27,000$ total physical particles. Each physical particle is given an initial perturbation of $\sigma \cdot \gamma$ where $\gamma = 5.0$ is the maximum amplitude of the initial velocity perturbations and σ is an evenly distributed random number such that $-0.5 \leq \sigma \leq 0.5$. Particles are assumed light, with a Stokes number of $St = 0.01$, and therefore do not effect the fluid phase. The particle correction factor of Boiko et. al. [15] from (5.15) is used in this experiment to reflect the drag modeling used in the EL method in the remainder of this dissertation.

The time lapse in Figure 4.8, shows the maximum physical particle cloud spread over time (in green) as compared to the average particle location for the first order model (in red) and the SPARSE model (in blue). The modeled computational particles are shown with the initial particle cloud diameter. The helicity of the fluid is projected onto the boundaries with a range of -0.1157:0.1157 or a variation of 10% of the initial maximum helicity.

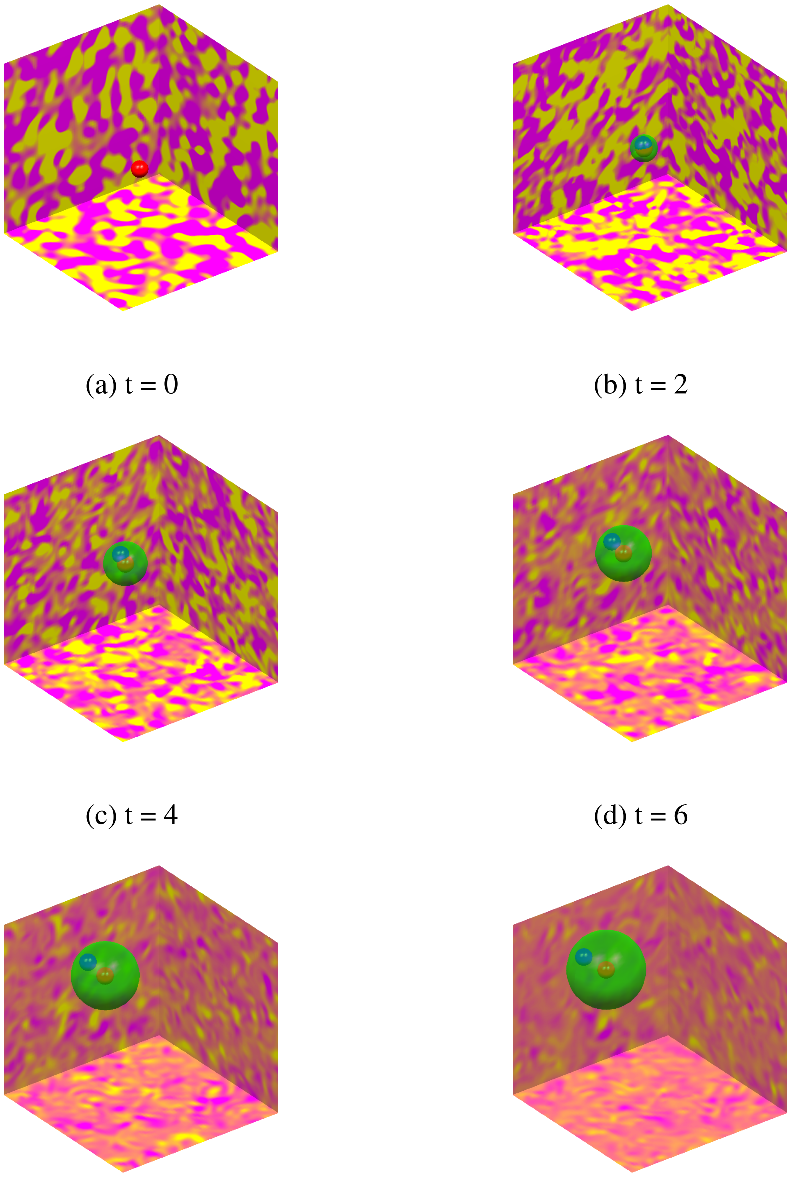
The statistics for the magnitude of the average particle cloud distance from the origin and velocity,

$$\begin{aligned} |\mathbf{x_p}| &= \sqrt{x_p^2 + y_p^2 + z_p^2}, \\ |\mathbf{v_p}| &= \sqrt{u_p^2 + v_p^2 + w_p^2}, \end{aligned} \tag{4.32}$$

for the exact, first order and SPARSE models are shown in Figure 4.9. While both the full particle cloud and the single modeled particles are initially entrained in the same turbulent eddy, the physical cloud spreads over time. Because the turbulence is isotropic, the average velocity in the box is zero, so as the cloud spreads and the fluid velocity is sampled over a larger area, the fluid velocity through the cloud approaches zero. Because the fluid velocity is interpolated at a single point in the first order model, this decay is delayed. Additionally, the kinetic energy present in the initialization of the particle cloud causes the particles to deviate from the fluid path, further increasing the particle cloud spread. Errors in the magnitude of the distance from the origin and velocity,

$$\begin{aligned} \varepsilon_{|\mathbf{x_p}|} &= |\overline{\mathbf{x_{p,exact}}}| - |\mathbf{x_{p,model}}|, \\ \varepsilon_{|\mathbf{v_p}|} &= |\overline{\mathbf{v_{p,exact}}}| - |\mathbf{v_{p,model}}|, \end{aligned} \tag{4.33}$$

are shown in Figure 4.10. The SPARSE model accounts for the initial kinetic energy and models the average fluid velocity though the cloud appropriately, as is evident in SPARSE modeling errors below 0.5%.



(a) t = 0

(b) t = 2

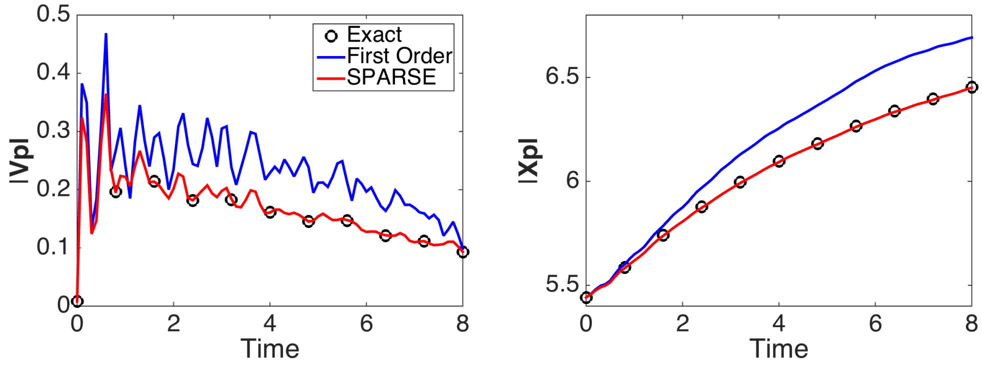
(c) t = 4

(d) t = 6

(e) t = 8

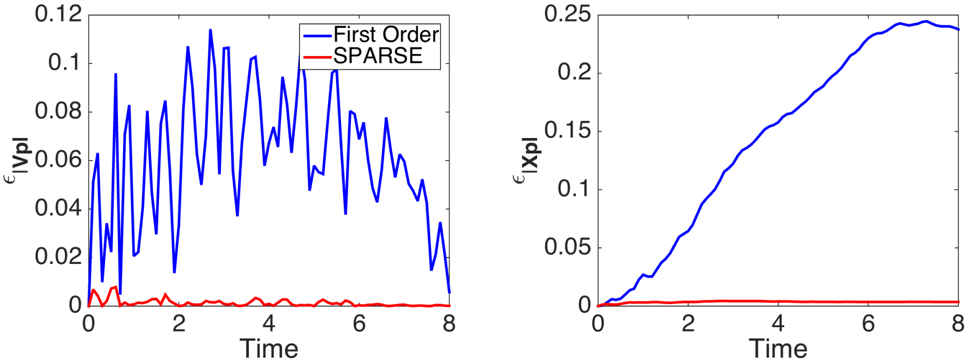
(f) t = 10

Figure 4.8: Projected helicity and particle cloud locations for the 3D isotropic decaying turbulence case at times t = (a) 0.0, (b) 1.6, (c) 3.2, (d) 4.8, (e) 6.4 and (f) 8.0. The first order modeled particle is shown in blue, the SPARSE modeled particle is shown in red and the maximum spread of the physical particle cloud is shown in green.



(a) (b)

Figure 4.9: The magnitude of the average (a) velocity and (b) particle distance from the origin is shown when computed using the first order model, SPARSE model and average over the physical particles.



(a) (b)

ma

Figure 4.10: The error in computing the magnitude of the average (a) velocity and (b) particle distance from the origin is shown when computed using the first order model, SPARSE model and average over the physical particles.

Chapter 5

Multi-Scale Computation of Particle-Laden Flows

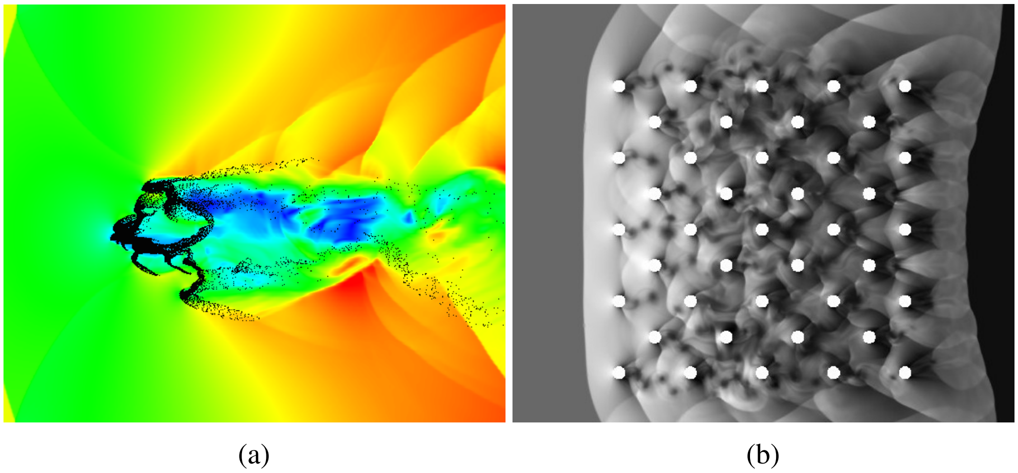
In collaboration with Professor Udaykumar's group at the University of Iowa, the foundation for a hybrid micro-scale DNS and macro-scale Eulerian-Lagrangian model for the multi-scale computation of shocked particle-laden flows has been laid out. The hybrid code couples a full resolution, micro-scale DNS code [119] with the macro-scale EL code [68] to provide a dynamic closure model to solve the particle momentum equations over all parameter spaces. The following is a reprint of [124] describing the coupling of the micro- and macro-scales.

5.1 Introduction

5.1.1 Motivation and Applications

A wide variety of problems in multi-material dynamics including the passage of a shockwave through a gas laden with particles [42], problems involving crack propagation in heterogeneous materials such as bones [54, 53, 52, 55] or concrete structures [147, 148] involve the intricate coupling of physics at two or more distinct length and time scales. Further examples of such problems include modeling of heterogenous explosives [8, 75, 98, 74], flow of mixtures including sediment transport in river beds [1], flow through fluidized beds [59] and flow of blood, i.e. plasma carrying cells and macromolecules [107]. In such systems, the physics of the micro/meso-scale needs to be represented in macro-scale simulations. This can be achieved by averaging over the heterogenous micro/meso-scale. In such volume-averaged macro models [138], or homogenized models [37, 68, 128], micro/meso physics appear in the form of closure terms in the macro-scale equations.

Process-scale computations typically demand macro-scale governing equations and simulation techniques. For example, in the problem of a shock wave interacting with a dusty gas, the number of dust particles is extremely large. To follow the evolution of the gas-solid mixture, a common practice is to define a computational particle as an agglomerate of a number of dust particles and to adopt a mixed Eulerian-Lagrangian viewpoint [68], as in Figure 5.1a. Particle paths are traced in a Lagrangian reference frame while solving the fluid equations in a fixed Eulerian frame. In this approach, the computational particles are modeled as singular point sources, which couple with the carrier fluid through momentum exchange modeled via source terms in the fluid equations [68, 34, 33]. The source terms close the unresolved momentum exchange between the fluid and solid (particle) phases, providing the forces on the particles. For small particle Reynolds numbers and incompressible flow, the drag on a spherical particle can be determined analytically using Stokes drag law [136]. A range of empirical drag laws exist, which incorporate the effect of inertia [26], compressibility [16, 88], slip coefficients [141], various shape factors [143] and/or viscosity ratio for droplets [43] for more complex flows. In general, closure models are obtained in the form of correlations developed in a physical experiment.


(a)

(b)

Figure 5.1: Two-dimensional examples of the (a) macroscale interaction of a large number of modeled particles with a right running normal shock and (b) full resolution meso-scale computation of a small number of particles interacting with a right running normal shock.

Empirical closure models such as drag correlations are only applicable in limited parameter spaces. To overcome this limitation, high resolution micro-scale methods that resolve the dynamics at the particle scale, as can be seen in Figure 5.1b [91], can be used as surrogates for physical experiments to obtain closure models connecting the meso-scale physics to the macro-scale. In [90] for example, an artificial neural network (ANN) is used to construct a closure model for particle-laden shocked flow based on computational experiments. The neural network then supplies closure terms (drag force) to the macro-scale simulation. Further examples of closure terms constructed from computational experiments using an ANN can be seen in [54, 53, 52, 55, 147, 148, 34].

5.1.2 Bridging Scales in a Multiscale Multimaterial Model

There are three components to the multiscale modeling approach described above: a macro-scale solver which computes the interaction of a large number of particles with a carrier flow, a meso-scale solver, which resolves the fine-scale particle-fluid dynamics of a smaller number of particles and a closure model which calculates the drag and other relevant parameters from the meso-scale solver for use in the macro-scale solver. Generation of a closure model derived from an ensemble of full-resolution meso-scale computations requires quantifying the output from the meso-scale dynamics (for example, drag forces) under a number of different input parameters such as shock strength, particle loading, particle size distribution, etc.

5.1.3 Metamodels as Surrogates to Bridge Scales

A metamodel, or a 'model of a model' [79], builds a hypersurface from a limited amount of input/output data and approximates the output over a much wider parameter space. An excellent overview of metamodeling techniques is given in [131, 72, 73]. Several studies have compared metamodels for reconstructing hypersurfaces from computational experiments. A review of the challenges and concerns in metamodeling techniques can be found in [153] and [25]. In addition, Jin et. al. [71] compared the hypersurfaces approximated by a Polynomial Response Surface Method (RSM), a Kriging method, a Radial Basis Function Neural Network (RBFANN), and Multivariate Adaptive Regression splines (MARS) for 14 different test functions. Fang et. al. [40] compared the RBFANN method and the RSM method, with the express purpose of reconstructing hypersurfaces in multi-objective crashworthiness optimization. However, these studies have been limited to comparing the quality of approximation only for a given number of input points, and not over a range of input points.

The choice of a "good" metamodeling technique depends on the application and the purpose of the metamodel. Because metamodels are constructed from expensive numerical computations in multi-scale modeling and because the multiscale method should converge with increasing degrees of freedom, convergence of the metamodels with respect to the number of input points for a wide variety of hypersurfaces warrants careful investigation. This study shows that some metamodelling techniques converge faster than others only for a certain classes of hypersurface. Furthermore, some metamodels converge faster when the number of sampled input points is low, while other metamodels converge faster when the number of inputs is high. The focus of the current work is to examine the rate of convergence of the following three classes of metamodels for their suitability in bridging scales in a multiscale framework:

1. An interpolation method; Stochastic Collocation (PSC) methods [160, 7, 93] - the Polynomial Stochastic Collocation method (PSC) and the Adaptive Stochastic Collocation Method (ASC) are chosen as representative interpolation methods.

2. A fitting method; the RBFANN method [23, 110, 58] is the fitting method considered in the study.

3. A method which first fits a global response surface and then interpolates local departures from the global fit; the Kriging method [84, 145, 86] and the Dynamic Kriging Method (DKG) [133] are chosen in this study.

The methods, represent, in their respective classes as approximators, typical and state-of-the-art techniques for assimilating and representing the complex relationships between input parameters and the resultant outputs in a multi-dimensional parameter space.

The modeling methods are summarized in Section 5.2. In Section 5.3, the evaluation criteria that is used to assess the performance of the metamodels is discussed, and their convergence behavior is analyzed in Sections 5.4 and 5.5 for several analytic functions and existing empirical particle drag models. To elucidate their metamodeling capabilities, the approximation error of the metamodels on several known functions is analyzed.

These functions are chosen to highlight the strengths and weaknesses of each class of metamodels in approximating a wide variety of hypersurfaces; irregularly sampled data and presence of noise in the input data are also studied. Empirical drag laws are then considered to illustrate the complexities that will be encountered by the metamodeling techniques in building a closure model for a full multi-scale solver. Conclusions drawn from the quantification of the modeling error are discussed in Section 5.6. To the best of the authors' knowledge, no comparison of the rate of convergence of these classes of metamodels, with the express purpose of ascertaining the suitability of the methods as an ingredient in the framework of multiscale modeling, have been previously performed.

5.2 Inter-scale Coupling Methods

The metamodeling techniques are summarized in the following section. Broadly, the problem of metamodeling is the estimation of the value of a function $f(\mathbf{x})$ at a point $\mathbf{x}_0$ where $f(\mathbf{x}_0)$ is unknown and the value of $f(\mathbf{x}) : \mathbb{C}^n \to \mathbb{C}$ is only known at certain discrete (distinct) points, $\mathbf{x}_j$. Here, $\mathbb{C}^n[\mathbf{0},\mathbf{1}]$ is a bounded subspace of $\mathbb{R}^n$, with $\mathbf{0}$ and $\mathbf{1}$ being an n-dimensional vector with all entries 0 and 1 respectively. The points, $\mathbf{x}_j, j = 1, 2, ..., N$, are the "input points" to the metamodel.

5.2.1 Stochastic Collocation Methods

Stochastic Collocation (SC) interpolation methods [160, 7, 93] rely on sparse grids generated using the Smolyak algorithm [132] to build a multivariate interpolation method by recursively taking the tensor products of univariate interpolation formulae. The resultant nodal architecture takes advantage of the recursive nature of the algorithm to enhance sparsity, i.e. limit the number of new data points required to improve the order of accuracy of the interpolation function. In SC methods, the estimated value of the function, $\widetilde{f}(\mathbf{x}_0)$, is given by [160, 7, 93],

$$\widetilde{f}(\mathbf{x}_0) = \sum_{j_1=1}^{m_{i_1}} \cdots \sum_{j_n=1}^{m_{i_n}} f(x_{j_1}^{i_1}, \ldots, x_{j_n}^{i_n})(a_{j_1}^{i_1} \otimes \ldots \otimes a_{j_n}^{i_n}), \tag{5.1}$$

where i is the level of interpolation, m_i is the number of input points required for level i and a_j are the basis functions used in n dimensions. An overview of the SC methods used in this study follow. For a more detailed derivation the reader is referred to [160, 7, 93].

The convergence of the SC method depends on the choice of basis function as well as the nature of the hypersurface being interpolated. In this study, two SC methods are considered, which will be referred to as the Polynomial Stochastic Collocation (PSC) method and the Adaptive Stochastic Collocation (ASC) method. The input points for PSC method are based on the the end knots of a Chebyshev polynomial on a Clenshaw-Curtis grid (Figure 5.2a) [160, 7, 93]. The PSC method is particularly effective in interpolating globally smooth functions because of the fit based on Lagrange polynomials. However, for steeper gradients and highly localized features, the PSC method displays Gibbs phenomena. Adaptive methods are therefore required to avoid these spurious oscillations in the solution. The support nodes in a Clenshaw-Curtis grid are not suited for adaptivity because they must be predetermined at each level. In the ASC method, input/output pairs are therefore located on a Newton-Cotes grid with equidistant nodes [93] (Figure 5.2b). The grid is locally refined around points where the hierarchical surplus, defined as,

$$\mathbf{w}_{\mathbf{j}}^{\mathbf{i}} = f(x_{j_1}^{i_1}, \ldots, x_{j_N}^{i_N}) - \tilde{f}^{i-1}(x_{j_1}^{i_1}, \ldots, x_{j_N}^{i_N}), \tag{5.2}$$

at level i on point j is above a threshold value, ε. Unless otherwise noted, $\varepsilon = 0.001$ will be used here. Local linear spline functions are used as the polynomial bases in the ASC method because high order polynomial basis functions suffer from Gibbs's phenomenon on uniform grids [93].

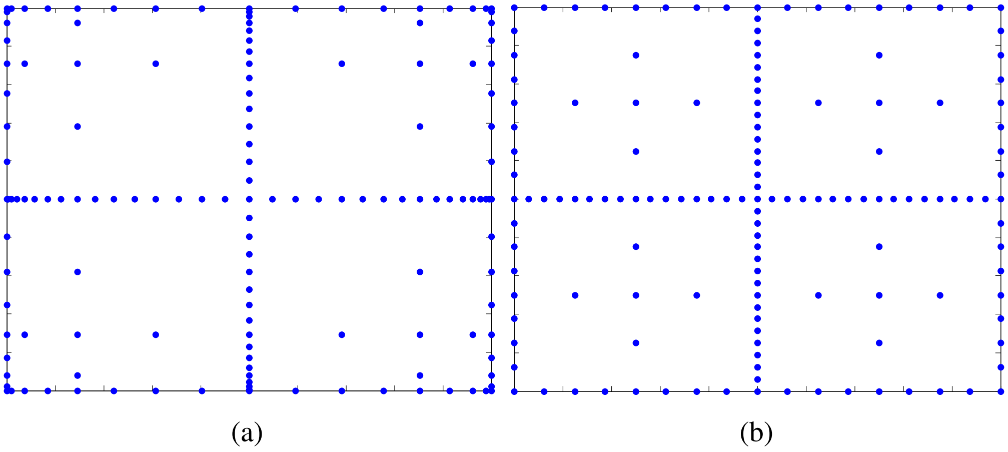


Figure 5.2: The distribution of nodes from a two-dimensional fifth level Smolyak Sparse Grid based on (a) Chebyshev polynomial end knots and (b) a uniform Newton-Cotes grid.

A major advantage of SC methods is the availability of an *a priori* error estimate. For the PSC method, the interpolation error in the maximum norm [7, 93] is on the order of

$$||f(x_0) - \tilde{f}^i(x_0)||_\infty = O\left(N^{-2}|log_2 N|^{3(n-1)}\right) \tag{5.3}$$

where n is the number of dimensions and N is the total number of interpolation points. In the ASC method, the additional error depends on the threshold hierarchical surplus value, ε [93],

$$||\tilde{f}^i(x_0)_{\mathrm{PSC}} - \tilde{f}^i(x_0)_{\mathrm{ASC}}||_\infty \leq N_2\varepsilon \tag{5.4}$$

where N_2 is the difference between the full Smolyak sparse grid and the number of input points in the locally refined grid.

5.2.2 Artificial Neural Networks Using Radial Basis Functions

A Radial Basis Function Artificial Neural Network (RBFANN) is a fitting method, particularly suited to function approximation and pattern recognition [126], which comprises a (finite) set of identical basis functions, called Radial Basis Functions (RBF) centered around several distinct points in the input space. In an RBFANN method, the estimated value of the function at a point $\mathbf{x}_0$, is given by [23]

$$\widetilde{f}(\mathbf{x_0}) = \sum_{k=1}^{M} \lambda_k \phi\left(\mathbf{x_0}, \mathbf{x}_{c_k}; \theta_{c_k}\right) \tag{5.5}$$

where λ_k is the weight associated with each of the basis functions $\phi\left(\mathbf{x_0}, \mathbf{x}_{c_k}; \theta_{c_k}\right)$, while the parameter $\mathbf{x}_{c_k}$ is the position of the basis function and the parameter θ_{c_k} is the shape parameter. A typical choice of the basis function is a Gaussian $\phi\left(\mathbf{x_0}, \mathbf{x}_{c_k}; \theta_{c_k}\right) = exp\left(-d_{c_k}^2/\theta_{c_k}^2\right)$ where $d_{c_k} = \|\mathbf{x_0} - \mathbf{x}_{c_k}\|$, is the Euclidean Norm; this basis function has non-compact support [58, 157, 159, 101]. If the Gaussians are equally spaced, they form a set of Riesz Bases for the input space [44].

The weights in (5.5) are given by

$$\boldsymbol{\lambda} = \mathbf{H}^{\dagger}\mathbf{f} \tag{5.6}$$

where $\mathbf{H} = H_{jk} = \phi\left(\left\|\mathbf{x}_j - \mathbf{x}_{c_k}\right\|; \theta_{c_k}\right); j = 1, 2, ..., N,\ \ k = 1, 2, ..., M,$ with N being the number of inputs and M the number of Gaussians used. $\mathbf{H}^{\dagger}$ is the pseudoinverse of the interpolation matrix, $\boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 & \lambda_2 & \cdots & \lambda_M \end{bmatrix}^T$ and $\mathbf{f} = \begin{bmatrix} f(\mathbf{x}_1) & f(\mathbf{x}_2) & \cdots & f(\mathbf{x}_N) \end{bmatrix}^T$. Equation (5.6) is thus the least-squares solution to the overdetermined system of equations given by $\mathbf{H}\boldsymbol{\lambda} = \mathbf{f}$. Because $\mathbf{H}$ is a dense matrix and often ill-conditioned [46, 41], a singular value decomposition of the interpolation matrix is performed to obtain $\mathbf{H}^{\dagger}$. The determination of optimal parameters, M, $\mathbf{x}_{c_k}$ and θ_{c_k} of an RBFANN is a subject of active reseach [45, 82, 23, 24, 9, 31, 36, 62, 104]. In the current approach the parameters are determined using an unsupervised training process (which means that a non-linear optimization algorithm is not performed to determine the parameters). The RBF algorithm involves the following stages.

1. *Determination of M:* In order to avoid "memorizing" [140] the inputs, in a typical RBFANN, the number of RBFs is chosen to be less than the number of inputs. However, in the context of a multiscale framework, RBFANN is used to "learn" from fully-resolved micro-scale computational experiments. Since such computations are expensive, the method must create a hypersurface from as few inputs as possible. The number of Gaussians are therefore chosen to be approximately 1.1 times the number of inputs.

2. *Determination of* $\mathbf{x}_{c_k}, k = 1, 2, ..., M$. The RBFs are initially uniformly distributed in the domain and are updated by a K-means clustering algorithm [100] to avoid the possibility of an empty cluster in case of non-uniform inputs.

3. *Determination of the shape parameter,* θ_{c_k}*, of the RBFs:* The shape parameter is chosen to be equal to the mean distance of an RBF to its five nearest neighboring RBF such that they span the entire domain of the input space [100].

5.2.3 Kriging methods

The third class of metamodels studied are derived from Kriging methods, which have their origin in mining and geostatistical applications involving spatially and temporally correlated data [27, 96]. Kriging methods combine a global (polynomial) model which fits to the given response surface, while the local departures from the global fit are estimated using semi-variogram models [84]. The resulting approximation interpolates the sampled input points. In a Kriging method, the estimated value of the function, denoted by $\tilde{f}(\mathbf{x}_0)$ can be expressed as [145],

$$\tilde{f}(\mathbf{x}_0) = \sum_{l=0}^{r} \lambda_l p_l(\mathbf{x}_0) + Z(\mathbf{x}_0) \tag{5.7}$$

where $p_l(\mathbf{x}_0)$ denotes the l-th order term in a polynomial basis functions of maximum order r and λ_l is the least-squares solution to the set of normal equations $\tilde{f}(\mathbf{x}_j) = \lambda_l p_l(\mathbf{x}_j)$, $j = 1,2,...,N$ [133]. In Equation(5.7), $Z(\mathbf{x}_0)$ is a realization of a Gaussian random process with zero mean, $E\left[Z(\mathbf{x}_j)\right] = 0$ and a covariance structure, $E\left[Z(\mathbf{x}_j)Z(\mathbf{x}_q)\right] = \sigma^2 R_{jq}$, $j,q = 1,2,...,N$ where $\sigma^2$is the process variance, and R_{jq} is the correlation model of the process [164] . The correlation model, R_{jq} is often chosen to be of the form [164]

$$\mathbf{R} = R_{jq} = R(\theta, \mathbf{x}_j, \mathbf{x}_q) = \prod_{k=1}^{n} \gamma_k(\theta_k, d_k), \tag{5.8}$$

with a shape parameter θ where $d_k = (x_{kj} - x_{kq})$, $k = 1,2,...,n$. An example of such a correlation model is a Gaussian model, where $\gamma_k = e^{-\theta_k d_k^2}$, $k = 1,2,...,n$. Although any value of θ would provide an interpolation formula, the optimal value of θ in (5.8) is selected to maximize the following log-likelihood function of the model parameters [133]

$$l = -\frac{N}{2}\ln\left[2\pi\sigma^2\right] - \frac{1}{2}\ln\left[|\mathbf{R}|\right] - \frac{1}{2\sigma^2}(\mathbf{f} - \mathbf{P}\boldsymbol{\lambda})^T \mathbf{R}^{-1}(\mathbf{f} - \mathbf{P}\boldsymbol{\lambda}) \tag{5.9}$$

where $\mathbf{f} = \left\lfloor \begin{array}{cccc} f(\mathbf{x}_1) & f(\mathbf{x}_2) & \cdots & f(\mathbf{x}_N) \end{array} \right\rfloor^T$, $\boldsymbol{\lambda} = \left\lfloor \begin{array}{cccc} \lambda_0 & \lambda_1 & ... & \lambda_r \end{array} \right\rfloor^T$ and $\mathbf{P} = P_{jl} = p_l(\mathbf{x}_j)$. In the current work, the Kriging method with a first-order mean structure and a Gaussian correlation model is applied using a MATLAB code, DACE [86].

An improved version of Kriging algorithm called the Dynamic Kriging Method with adaptive sampling (DKG) [164, 133] has been implemented as well. The detailed algorithm can be found in [133], the key aspects of the algorithm are summarized here.

1. The DACE code uses a modified Hooke and Jeeves algorithm [86] to find the optimum value for θ. However, this method often fails to provide a global optimum of θ and has therefore been replaced in the DKG method by a Global Pattern Search (GPS) algorithm [164].

2. The choice of the correlation model and the mean structure depends on the hypersurface to be approximated [87] and is not known a priori. In the DACE code, the order of the mean-structure, r, and the correlation function are selected by the user a priori. However, the DKG method compares between three mean structures corresponding to $r = 0,1$ and 2, using a Cross-Validation (CV) error estimate [133]. The method also evaluates seven different correlation models, as listed in Table (5.1) and selects the best one, i.e. the model which maximizes the likelihood function.

3. In a typical Kriging model, samples are either supplied by the user heuristically or are supplied by a sampling strategy like the Latin Hypercube Sampling method [135, 51] or Latin Centroidal Voronoi Tessellation (LCVT) method [20].These methods usually generate input points in the domain uniformly. However, the Dynamic Kriging model is integrated with an adaptive sampling strategy, which selects more samples at highly non-linear portions of the hypersurface, thus aiming to obtain a better approximation using a parsimonious distribution of input points. The sample insertion criterion is described in detail in [164].

Table 5.1: List of correlation functions

Correlation Function	$\gamma_k(\theta, d_k)$
Exponential	$exp(-\theta_k \lvert d_k \rvert)$
General Exponential	$exp\left(-\theta_k \lvert d_k \rvert^{\theta_{n+1}}\right); 0 < \theta_{n+1} \leq 2$
Gaussian	$exp\left(-\theta_k d_k^2\right)$
Linear	$\max\{0, 1 - \theta_k \lvert d_k \rvert\}$
Spherical	$1 - 1.5\xi_k + 0.5\xi_k^3,\ \xi_k = min\{1, \theta_k \lvert d_k \rvert\}$
Cubic	$1 - 3\xi_k^2 + 2\xi_k^3,\ \xi_j = min\{1, \theta_k \lvert d_k \rvert\}$
Spline	$1 - 1.5\xi_k^2 + 30\xi_k^3$ for $0 \leq \xi_k \leq 0.2$ $1.25(1-\xi_k)^3$ for $0.2 < \xi_k < 1$; $\xi_k = \theta_k \lvert d_k \rvert$ 0 for $\xi_k \geq 1$

5.3 Evaluation of Techniques for Metamodeling

In order to be used for the generation of closure models in a multiscale modeling framework, it is desirable that the metamodels described in the previous sections satisfy certain restrictions on the error behavior and model construction. These include:

1. *Parsimonious representation*: Computational experiments are expensive to perform and a single high-resolution realization can take several hours to compute, even on multiprocessor architectures. Thus, the metamodel should be accurate and converge rapidly when supplied with information obtained from a minimum number of high resolution simulations (input data points).

2. *Monotonic convergence:* The inclusion of additional meso-scale simulations must result in improved approximation of the closure model. Because the closure model will not be known *a priori*, monotonic convergence is required so the modeling error can be estimated and additional micro-scale computations can be performed to improve the accuracy of the metamodel.

3. *Multidimensional representations*: Since multidimensional parameter spaces are expected, the method must be easily extendable to multiple input dimensions

without suffering from the "curse of dimensionality", i.e. the number of input points should not increase exponentially with the increase in the dimension of the input space. This consideration obviates the use of methods like Lagrange interpolation and discourages the use of methods which operate on a dense grid of input points.

4. *Flexibility and Re-use*: Since the metamodelling technique relies on expensive high resolution simulations as inputs, previous results must be utilized when expanding the parameter space. In addition, if the parameter space is expanded to include a larger domain of approximation, the augmented parameter space and corresponding data values must be included in generating an updated hypersurface. This becomes difficult, for example, when a metamodelling approach relies on fixed collocation points (for example, Gauss points in the computational domain, zeroes of a Chebyshev polynomial, etc) for constructing closure models because additional interpolation would be needed to fit data onto the predefined nodes.

5. *Treatment of noisy data*: Since numerical noise is expected in the meso-scale results, it is preferred to have the metamodel filter noisy data to build a smooth approximation without adding excessive filtering errors.

5.4 Analysis of Convergence Behavior of the Metamodeling Techniques for Analytical Functions

To evaluate and compare the metamodels in light of the above mentioned criteria, hypersurfaces for several predetermined functions are generated and the approximations are compared to the exact functions. The method of comparing the metamodels is as follows:

1. *Training the metamodels:* A number (say N) of input points of a given function are provided to the metamodels as training points. These sample inputs are spaced at regular intervals along each axis throughout the parameter space for the RBFANN and the Kriging method. Because the input points of a PSC method are predetermined for each level of refinement, a comparable number ($\sim N$) of nodal collocation points are provided as inputs for the PSC method. In the DKG and ASC methods, the sample insertion criteria is adjusted such that when the maximum number of samples reach N, no further inputs are generated and the training process is terminated. Thus, the approximation of the metamodels are evaluated at a comparable number of training points.

2. *Building the hypersurface to test accuracy:* Once trained, the metamodel is used to predict the value of the function at $S = 100 \times 100$ uniformly distributed points in the domain of approximation.

3. *Evaluating the Approximation Error and the Rate of Convergence:* Because metamodels are constructed from analytical functions, it is possible to compare the predicted values of the metamodels with the exact values of the function at these S points. In order to quantify the accuracy of approximation at these points, a normalized sum-of-squares error is calculated:

$$\varepsilon_{L^2} = \sqrt{\frac{\sum_{j=1}^{S} \left(f\left(\mathbf{x}_j\right) - \widetilde{f}\left(\mathbf{x}_j\right) \right)^2}{\sum_{j=1}^{S} \left(f\left(\mathbf{x}_j\right) \right)^2}}, \tag{5.10}$$

where $f\left(\mathbf{x}_j\right)$ is the exact value of the function at the point $\mathbf{x}_j$ and $\widetilde{f}\left(\mathbf{x}_j\right)$ is the value approximated by a metamodel. The error, ε_{L^2}, is calculated and plotted for different values of N.

4. *Local Error Estimation:* Since the error measure ε_{L^2} is a global measure of approximation errors, a normalized local error field, $\delta\left(\mathbf{x}_j\right)$, defined as

$$\delta\left(\mathbf{x}_j\right) = \frac{f\left(\mathbf{x}_j\right) - \widetilde{f}\left(\mathbf{x}_j\right)}{\frac{1}{S}\sum\limits_{l=1}^{S} f\left(\mathbf{x}_l\right)}, \tag{5.11}$$

   is calculated to quantify the local approximation error of a hypersurface by a metamodel at the points $\mathbf{x}_j$, $j = 1, 2, ..., S$. Representative plots showing the contours of the local error field for a given value of N are also shown in the subsequent section.

For the purpose of illustration, representative contours of the hypersurfaces and local error plots are shown for $N = 144$ points.

5.4.1 Harmonic Test Function

The first function considered is a smooth harmonic function, given by

$$f(x,y) = \sin(2\pi x)\cos(4\pi y) + 2, \tag{5.12}$$

where x and y range from 0 to 1.

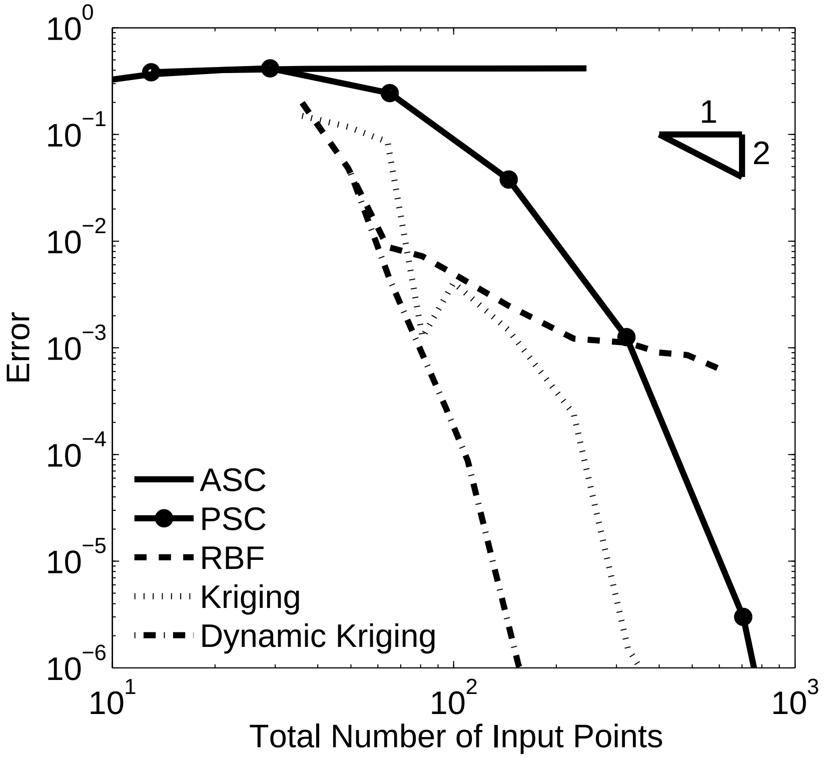


Figure 5.3: Error plot showing the convergence rates on approximating the smooth harmonic test function from (5.12).

As the convergence rates of the errors defined by Equation (5.10) shown in Figure 5.3 demonstrate, if the number of input points is below 60, the hypersurface is best approximated by the RBFANN and DKG methods. However, as the number of input points increases, the rate of convergence of the RBFANN method decreases. The value of the shape parameter, $\theta^{\mathbf{c}_j}$, decreases as the number of input points increases for the RBFANN method. The system of equations defined by (5.6) becomes ill-conditioned and the SVD solver essentially “filters” out higher frequencies of the interpolation matrix. As the higher frequencies are removed, the accuracy of the representation decreases and the rate of convergence decreases if the number of inputs approaches and exceeds 100 points.

Table 5.2: Correlation parameters estimated by the Kriging method to approximate the hypersurface given by (5.12).

Number of Inputs	θ(Kriging)	θ(DKG)
36	6.13	0.8898
64	6.13	0.4523
81	0.322	0.3741
100	0.161	0.3683
144	0.161	0.2863

The error in approximation by the Kriging method, applied using the DACE code, does not decrease monotonically, evinced by the spikes seen in Fig. 5.4. The rate of convergence changes when the number of input points changes from 64 to 81 and from 81 to 100. To further investigate this, the value of the shape parameter θ, as estimated in the Kriging Method is examined and compared with that obtained by the DKG method in Table 5.2. The value of θ for the Kriging method undergoes sharp changes when the number of input points change from 64 to 81 and from 81 to 100. This corresponds to those points in Figure 5.3 when the rate of convergence of the Kriging method also changes. In comparison, the shape parameter estimated by the DKG method decreases uniformly (Table 5.2) and the error in the DKG method also decreases uniformly (Figure 5.3). This numerical example illustrates the advantage of the GPS algorithm for determining the optimum value of θ over the modified Hooke and Jeeves algorithm used in DACE.

The error in approximation of the PSC method is initially constant, as shown in Fig. 5.3, and decreases as the number of input points exceeds 30, finally decreasing spectrally when the number of inputs increases beyond 60. In contrast, the error of approximation of the ASC method does not change for any given number of input points. The sinusoidal variation of the function along the *x* axis is satisfactorily reconstructed, but the cosine waves in the *y* direction are not recognized all together, as is shown in the approximation of the hypersurface by the ASC method in Fig. 5.4. The ASC method initially operates on a mesh where the collocation nodes lie along the boundaries and the centerlines of the input domain (as seen in Figure 5.2). Because of the adequate representation of the sinusoidal values at the boundary and along the centerlines, the hierarchical surplus falls below the threshold value and input nodes are not successively refined on the interior of the domain. Hence, the hypersurface reconstructed by the ASC method is constant with respect to the coordinates along the y axis causing an aliasing error. The sinusoidal variation of the function along the x axis is satisfactorily reconstructed, but the cosine waves in the y direction are not predicted by the ASC method.

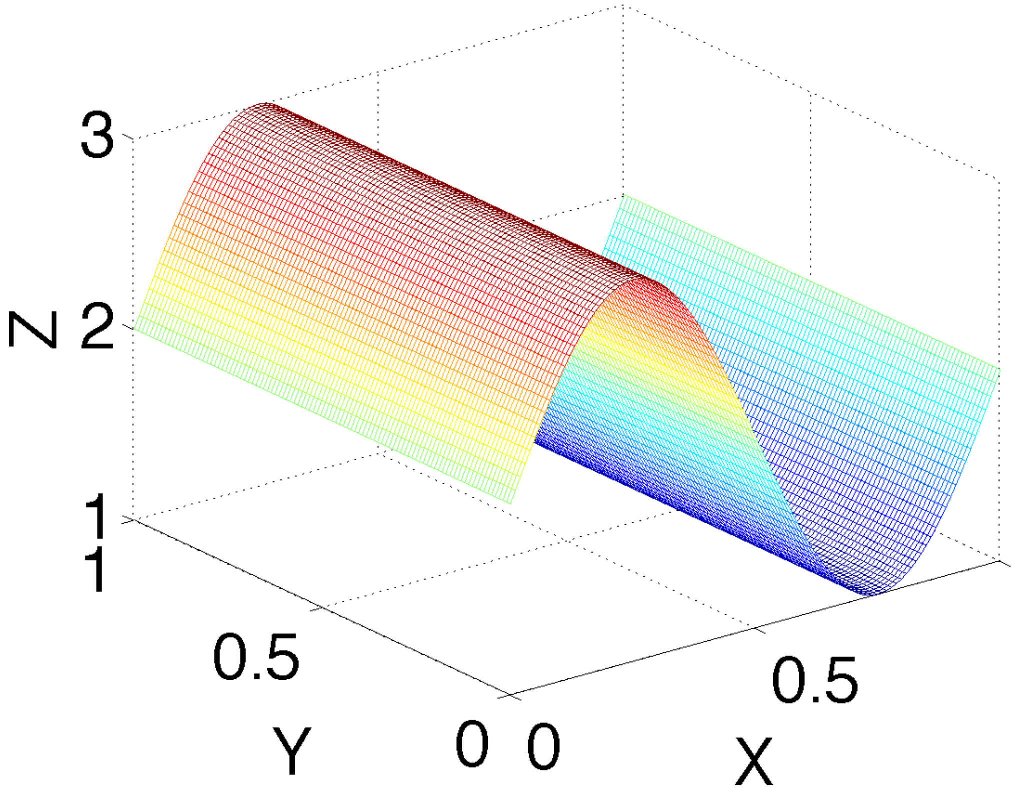


Figure 5.4: The hypersurface of the function given by (5.12) as approximated by the ASC method.

To appropriately characterize the convergence of the ASC on a smooth function, the aliasing error is eliminated by introducing slight phase-shifts in the sine and cosine waves of the function,

$$f(x,y) = \sin(2\pi x + 0.25)\cos(4\pi y + 0.5) + 2, \tag{5.13}$$

where x and y range from 0 to 1. The aliasing error seen in Fig. 5.4 does not appear in this function because the boundaries of the domain no longer trace a simple sine wave. The Newton-Cotes grid in Figure 5.2 does not line up directly with the sine wave so grid refinement occurs on the interior of the domain away from the centerlines and the sum-of-squares error converges per the theoretical expectation when the number of input points is greater than 20, which is shown in Figure 5.5.

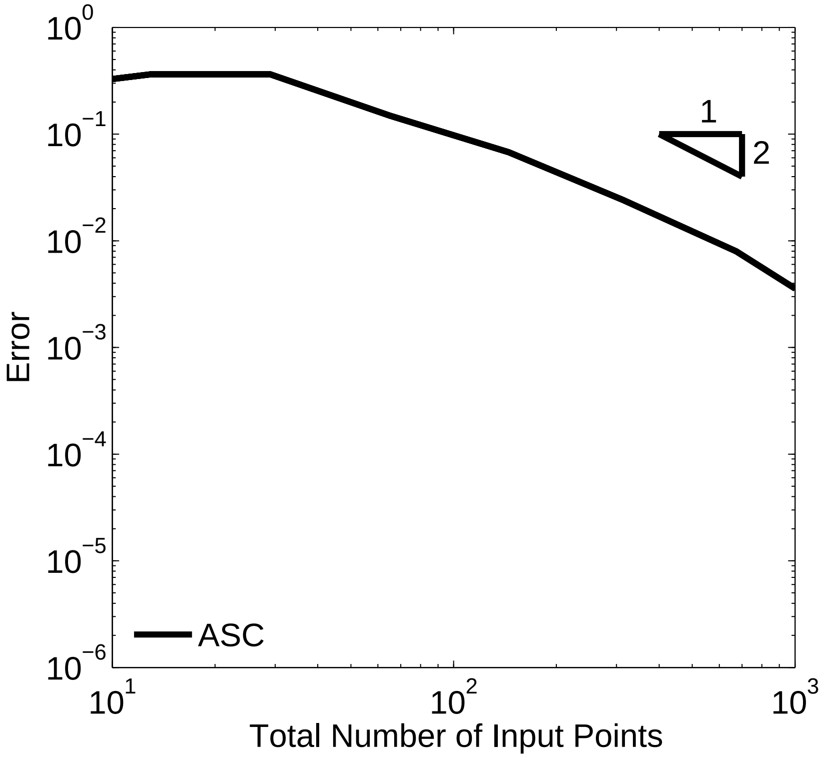


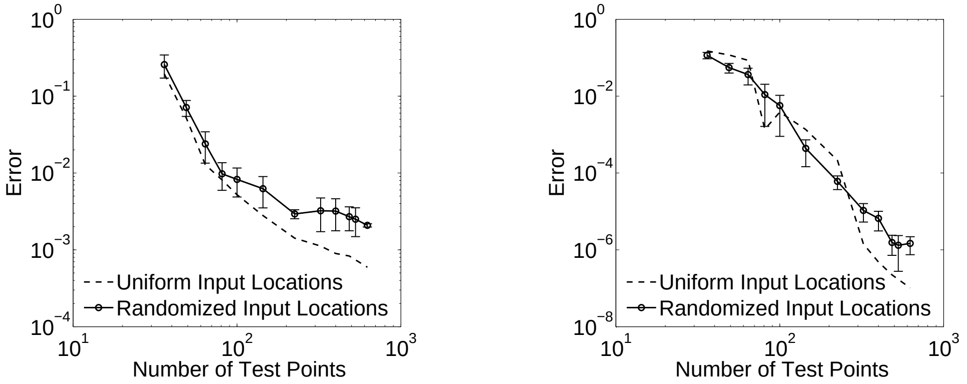
Figure 5.5: Error convergence of the approximation of the shifted harmonic test function from (5.13) using an ASC method.

5.4.2 Hypersurface Reconstructed from Irregularly Spaced Input Points

In a computational experiment, it is possible to specify the input points in several different ways. Input points can be specified at the nodes of a structured grid, i.e. at strictly regular intervals along each axis. In sampling methods like Latin Hypercube

Sampling Method [135], LCVT method [20] input points are distributed uniformly along the domain, although the points may not be necessarily specified at strictly regular intervals along each axis. The resulting input grid, in such a case, is unstructured. Because in a multiscale model it is not always possible or advisable to perform mesoscale experiments at strictly regular intervals in the parameter space, the input grid of a metamodel may not necessarily be structured. Therefore, a metamodeling approach which is fairly insensitive to the distribution of input points is preferable. In this section, a comparison is made between the hypersurfaces created by regularly spaced input points and irregularly spaced input points using the RBFANN and Kriging methods. Note that the PSC method is trained from inputs placed at specific locations in the input domain (see Fig 5.2), while the ASC method and the DKG method are integrated with a sampling strategy and are therefore not tested in this section.

To train the RBFANN methods and the Kriging methods, the harmonic function defined by Equation (5.12) is considered. Here, the input points are chosen at random within the domain. Because the location of a given number of input points in the domain is not unique, five such random input distributions are used to calculate the average normalized sum-of-squares error, given by Equation (5.10), while the standard deviation of the error for the input distributions (for a fixed number of input points) is used to obtain a prima facie measure of the confidence interval. The avergae error is then compared to the error calculated from the approximation of a structured grid of training points. Figures 5.6a and 5.6b show the rate of convergence of the error for the RBFANN and Kriging methods respectively.



(a)

(b)

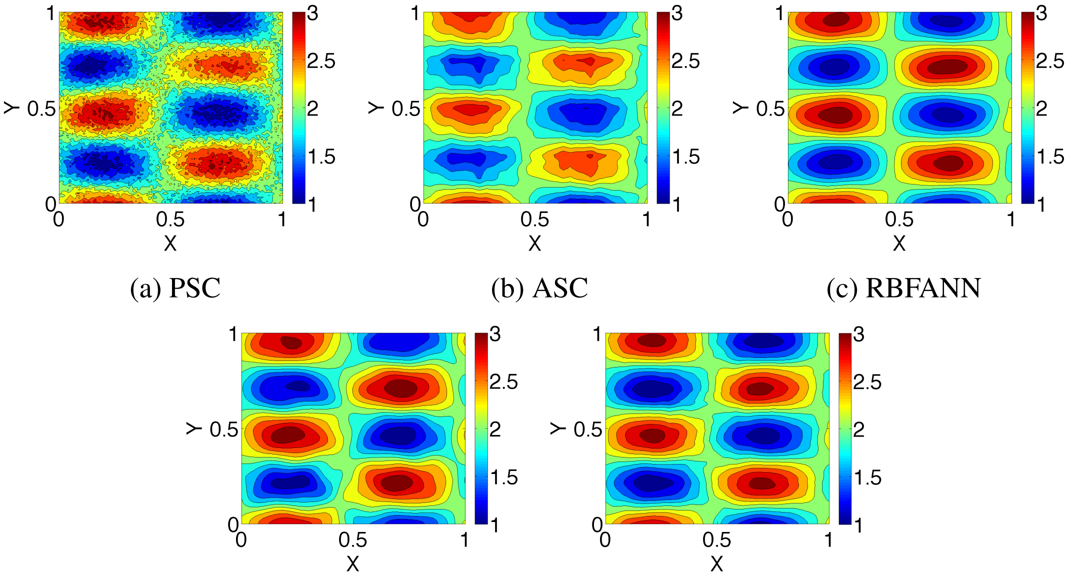
Figure 5.6: Error plot showing the convergence rates on interpolating the harmonic test function from Eqn. 5.12 using (a) an RBF method and (b) a Kriging method. The error bars show the standard deviation from the mean error for randomly distributed input points.

The convergence rates shown in Figures 5.6a and 5.6b demonstrate that structured grids give the lowest error for the given harmonic function. However, the convergence of the Kriging Method trained with an unstructured grid of sample points closely follows the convergence when using a structured grid of input points. However, if the number of input points increases beyond 81, the rate of convergence of the RBFANN method trained with regular input points is significantly different from that trained with random input points. Because the current architecture of the RBFANN uses more Gaussians than the number of input points, the RBFANN is sensitive to the distribution of input points. Thus, the rate of convergence of the current RBFANN model not only depends on the number of training points, but also on the location of the input points in the domain. Unless otherwise mentioned, an RBFANN method will be trained with regularly spaced inputs in subsequent sections of the present work.

5.4.3 Reconstruction of a Hypersurface from Noisy Data

Noise is unavoidable in solutions obtained from computational and physical experiments. While it is possible to filter out noise from any approximation obtained from a metamodel with a pre/post processing algorithm, additional errors may be added if a filter is used. Therefore, a metamodel that is relatively insensitive to noise is preferred.

To analyze the metamodels' response to numerical noise, each of the metamodels is given a fixed number of inputs from Equation (5.13) and a (white) noise of maximum amplitude 0.1 is superposed onto the training samples. The metamodels are then used to predict the hypersurface of Equation (5.13). The contour plots for each of the reconstructed hypersurfaces are shown in Figure 5.7.



(a) PSC

(b) ASC

(c) RBFANN

(d) Kriging

(e) DKG

Figure 5.7: Representative hypersurfaces for the noisy shifted harmonic function using (a) a PSC method, (b) an ASC method, (c) an RBF ANN, (d) a Kriging and (e) a DKG method.

The hypersurface predicted using the RBFANN method is noise-free. Because an RBFANN filters out all frequencies beyond a certain limit [44], it can filter out the noise components most effectively. As seen in Figure 5.7, the hypersurfaces predicted using the SC methods are noisy because the PSC and ASC are interpolation methods in which the reconstructed hypersurface must pass through all the values given at the training points exactly. Also, the Kriging and DKG methods have an inherent mean-structure which filters out the noise partially, but the process of minimization of the departure from the local fit result in an interpolation. Therefore, as can be seen in Figure 5.7, the noise from the training data is retained in the hypersurface predicted by these methods. This section thus demonstrates that there may be cases of noisy input data which may not be satisfactorily reconstructed by the SC methods and the Kriging family of methods. Unless otherwise mentioned, the training data in the subsequent sections of the work is noise-free.

5.4.4 A Radially Symmetric Steep Gradient Test Function

The previous examples provided valuable insight into the convergence of the metamodels for a smooth function. To study the response of the metamodels to a hypersurface with steep gradients localized in a region in the interior of the input domain, consider the function,

$$f(x,y) = \begin{cases} \frac{C_1}{2\sqrt{0.4}} \ln \left| \frac{\sqrt{x^2+y^2}+\sqrt{0.4}}{\sqrt{x^2+y^2}-\sqrt{0.4}} \right| & \text{if } \sqrt{x^2+y^2} \leq \sqrt{0.3}, \\ \frac{C_2}{2\sqrt{0.2}} \ln \left| \frac{\sqrt{x^2+y^2}-\sqrt{0.2}}{\sqrt{x^2+y^2}+\sqrt{0.2}} \right| & \text{if } \sqrt{x^2+y^2} > \sqrt{0.3}, \end{cases} \tag{5.14}$$

where

$$C_1 = \frac{16\sqrt{0.4}}{\ln\left|\frac{\sqrt{0.3}+\sqrt{0.4}}{\sqrt{0.3}-\sqrt{0.4}}\right|}$$
$$C_2 = \frac{16\sqrt{0.2}}{\ln\left|\frac{\sqrt{0.3}-\sqrt{0.2}}{\sqrt{0.3}+\sqrt{0.2}}\right|}$$

and x and y range from 0 to 1. This function is an integral of Equation (65) from [93]. The contours of $f(x,y) =$ constant are radially symmetric with respect to the origin. Steep gradients arise along the arc $\sqrt{x^2+y^2} = \sqrt{0.3}$, while the hypersurface varies more slowly as the distance from the arc increases. The exact hypersurface is shown in Fig. 5.9a.

Figure 5.8 shows the convergence of the errors of the metamodels with respect to the number of input points. In the convergence plot (Figure 5.8), the convergence rates for the metamodels are similar if the number of input points is below 900. However, the magnitude of the error is higher for the SC methods because the nodes are concentrated along the boundaries and centerlines of the domain, while the regions of high gradient are radially symmetric. The input points are therefore not collocated with the highly nonlinear regions of the hypersurface and the local features of the hypersurface are therefore not well resolved. The maximum values of the function are underrepresented when using the SC methods and oscillations occur. These oscillations increase as the distance between a point and the arc $\sqrt{x^2+y^2} = \sqrt{0.3}$ decreases resulting in the corresponding higher local errors seen in Figures 5.10a and 5.10b.

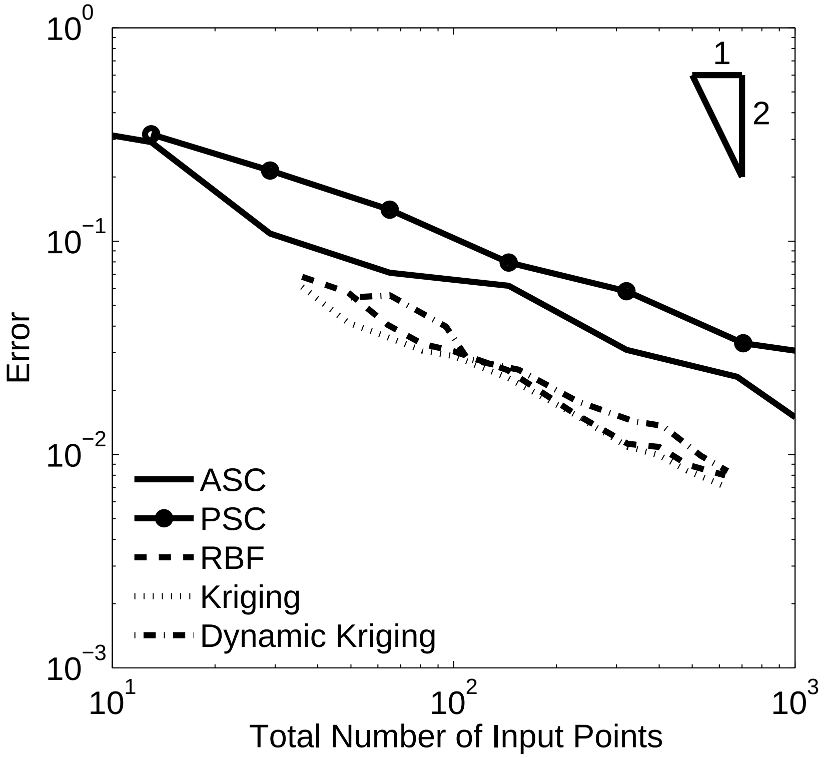
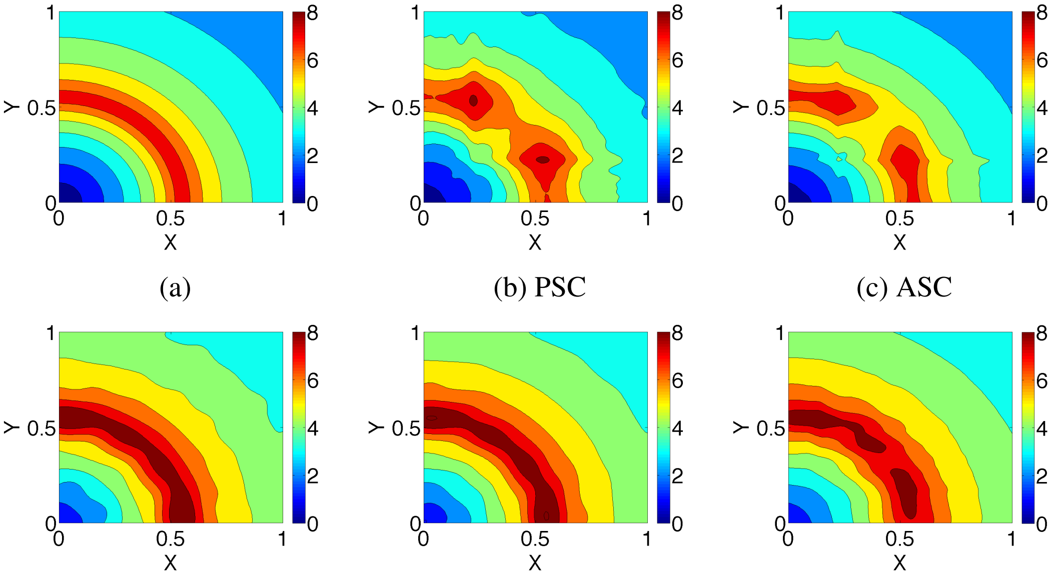


Figure 5.8: Error plot showing the convergence rates on approximating the test function from (5.14) using a PSC method, an ASC method, an RBFANN, a Kriging method and the DKG method.

Figure 5.9 shows that the hypersurfaces reconstructed by the RBFANN, Kriging and the DKG methods also display spurious oscillations, but these oscillations are smaller compared to those seen in the PSC reconstruction. When the number of input points is greater than 900, the PSC method converges exponentially because a larger number of input nodes fall on the arc containing the steeper gradients. If the number of input points increases beyond the scope of this study ($> 10^3$), the ASC method converges exponentially because of the increased number of nodes near the steep gradients. An example of the node distribution for such a case (i.e. $N > 10^3$) is shown in Figure 5.11.



(a)

(b) PSC

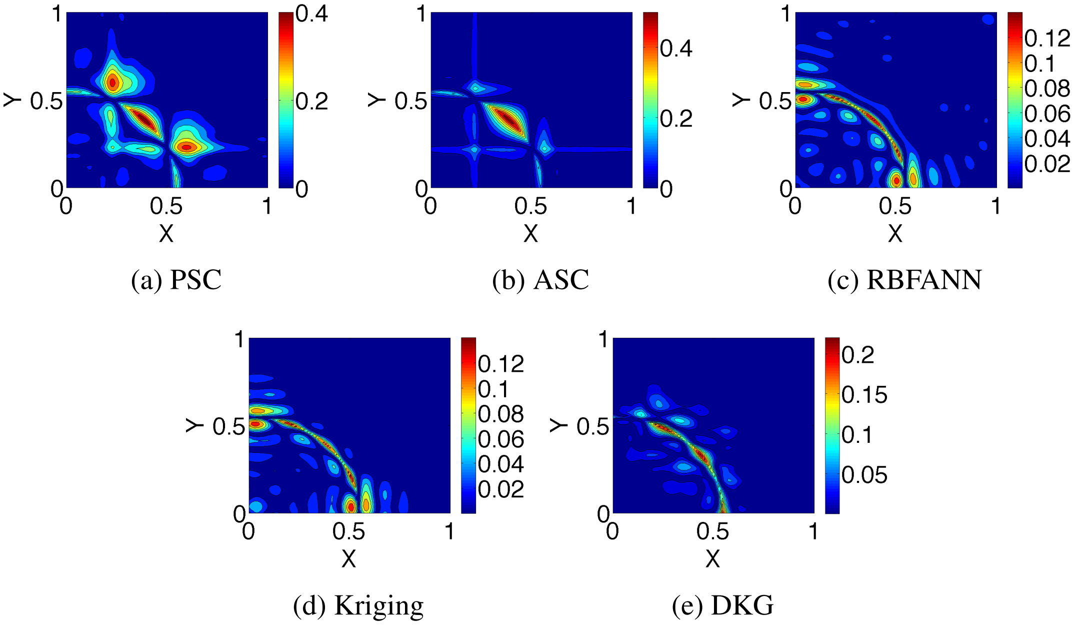
(c) ASC

(d) RBFANN

(e) Kriging

(f) DKG

Figure 5.9: The (a) exact contours and representative contour plots for the test function from (5.14) using (b) a PSC method, (c) an ASC method, (d) an RBF ANN, (e) a Kriging method and (f) the DKG method.



(a) PSC

(b) ASC

(c) RBFANN

(d) Kriging

(e) DKG

Figure 5.10: Local error plots for the approximation of the test function from (5.14) using (a) a PSC method, (b) an ASC method, (c) an RBF ANN, (d) a Kriging method, and (e) the DKG method.

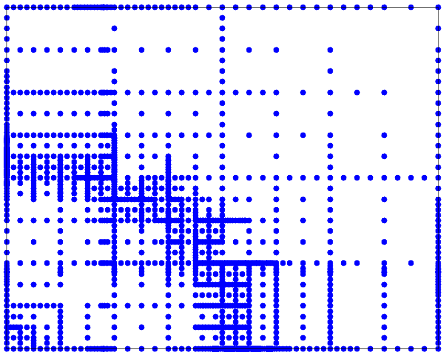


Figure 5.11: Locations of nodes in a 12th level Smolyak sparse grid, refined adaptively using hierarchical surpluses with a maximum error of $\varepsilon = 0.01$, interpolating the function in (5.14).

5.5 Analysis of Convergence Behavior of the Metamodeling Techniques for Empirical Drag Models

In addition to specific analytical functions designed to quantify the metamodels' approximation error on smooth harmonic functions and steep interior gradients, empirical drag models are considered. The hypersurfaces of these functions are expected to be similar to those of the closure models that the metamodeling techniques would be required to approximate in a multiscale modeling framework.

5.5.1 Boiko's Model for Drag on a Particle in a Shocked Flow

The first drag model considered is a model proposed by Boiko et. al. [15], and is given by

$$C_D = \left(0.38 + \frac{24}{Re_p} + \frac{4}{{Re_p}^{1/2}}\right)\left(1 + e^{-\frac{0.43}{M_p^{4.67}}}\right), \tag{5.15}$$

This function includes the effects of particle Reynolds number, $Re_p = |v_f - v_p|d_p/\nu$, and relative Mach number, $M_p = |v_f - v_p|/\sqrt{T_f}$, in the drag coefficient equation. The model is limited to relative Mach numbers of $M_p \leq 1.2$ and Reynolds numbers of $Re_p \leq 1 \times 10^4$. However, for the present study, the model is considered to apply for $0.1 \leq M_p \leq 3$ and $100 \leq Re_p \leq 10000$. The macro-scale EL code in [94, 34, 33] uses this empirical function to compute the particle drag coefficient.

The drag predicted by this model has two sharp zones of transition: at transonic Mach numbers ranging from $0.5 \lesssim M_p \lesssim 1.5$ and at $Re_p \lesssim 200$ when the flow changes from viscous Stokes' flow to an inertia dominated flow. The contour of the hypersurface of this figure is shown in Figure 5.14a.

The rate of convergence of the error of the metamodels with respect to the number of input points is shown in Figure 5.12. The function is most accurately represented by the SC methods. This is because in the SC methods collocation points are more concentrated close to the boundary (Figure 5.2). The sharpest transition region in the hypersurface in 5.14a lies along the low Re_p boundary. Because the concentration of nodes in the SC methods coincide with the highly non-linear regions of the hypersurface, the SC methods most accurately represent the function. It can be further observed from Figure 5.12 that among the SC methods, when the number of input points is below 400, the PSC method best approximates the hypersurface. This is because the high order global basis functions in the PSC method capture the flatter portions and smooth transition regions more accurately than the lower order local basis functions used by the ASC. If the number of input points is increased beyond 400, the ASC method is more accurate (Figure 5.12). This is because of the adaptive refinement based sampling strategy in the ASC method, which results in higher number of input points in the two transition zones and the ASC method converges exponentially.

As shown in Figure 5.12, the magnitude of the error of approximation by the RBFANN, Kriging and DKG methods is higher than the SC methods. In the RBFANN and the Kriging method, the predicted hypersurface shows spurious oscillations (from Figure 5.15). The magnitude of local error is highest for the Kriging method and the RBFANN method, while the magnitude of local error is higher for the DKG method than the SC methods. An important observation in Figure 5.15 that unlike the SC methods, the hypersurface obtained from the other metamodels is most errorneous along the transition region from low Re to high Re. This implies that unlike the SC methods, the number of input points for the other metamodeling techniques in these regions is not adequate to represent the sharp transition from low Re to high Re. Figure 5.13 shows the training points of the DKG method in. The number of training points is uniformly distributed throughout the domain, unlike the SC methods. Similarly, the RBFANN methods and the

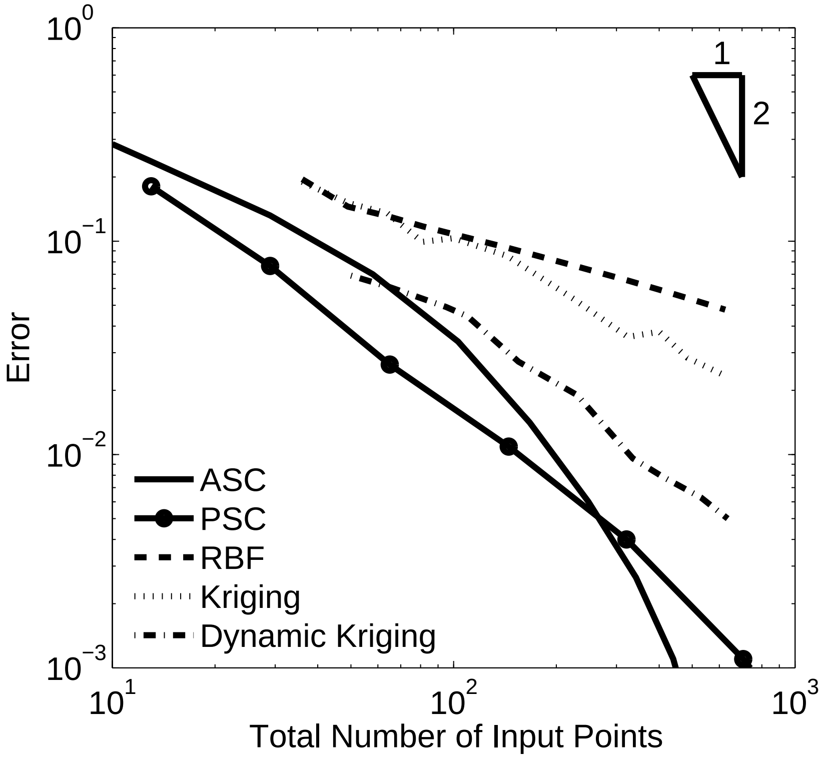


Figure 5.12: Error plot showing the convergence rates on approximating the drag model of Boiko et. al. [15] using a PSC method, an ASC method, an RBFANN, a Kriging method and the DKG method.

DKG methods are also trained using regularly spaced training points and hence the error of approximation of the RBFANN method, the Kriging method and the DKG method is higher than the SC methods, (Figure 5.12 and Figure 5.15).

5.5.2 Loth's Model for Drag on a Particle in a Shocked Flow

To further investigate the ability to capture steep gradients in the interior of the domain, consider the drag model proposed by Loth et. al. [88]. This model also corrects for high particle Mach and Reynolds numbers but over a wider range, $Re_p \leq 1 \times 10^5$ and

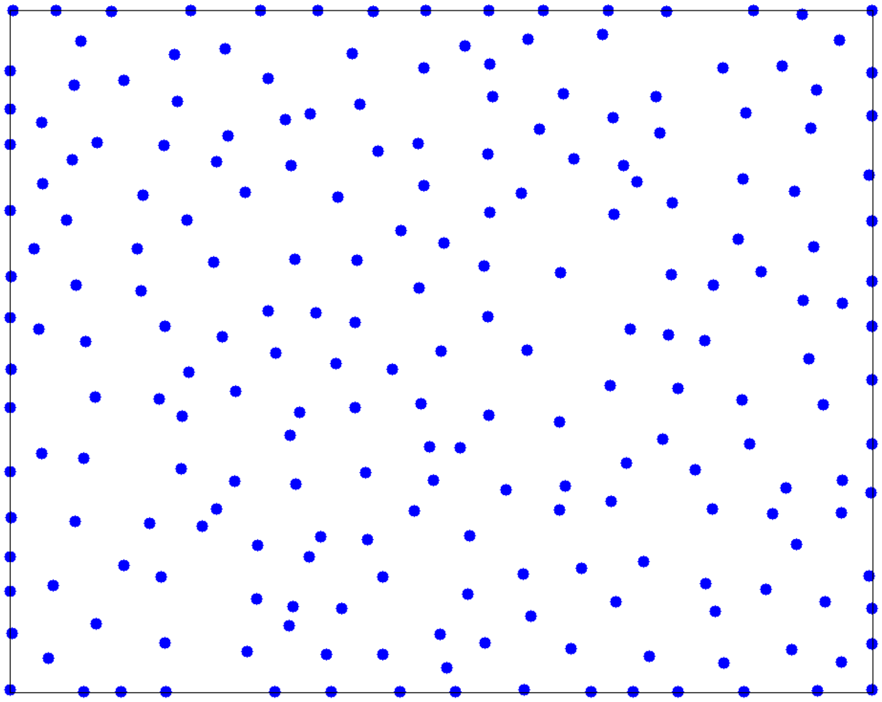


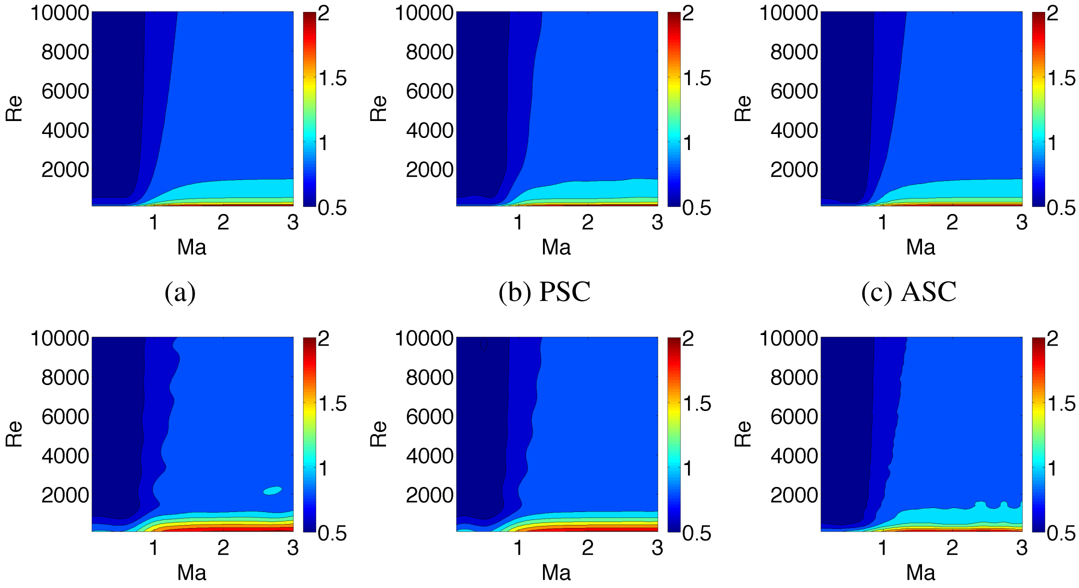
Figure 5.13: The distribution of training points in the DKG method

$M_p \leq 5$,

$$C_D = \frac{24}{Re_p}\left(1+0.25Re_p^{0.687}\right)\left(1.0-\frac{0.258C}{1.0+514G}\right)+\frac{0.42C}{1+4.25\times10^4 Re_p^{-1.16}G}, \quad (5.16)$$

where C and G are defined as,

$$C = \begin{cases} \frac{5}{3}\tanh(3.0\log(M_p+0.1)) & \text{if } M_p \leq 1.45, \\ 2.044+0.2e^{-1.8\left(\log\left(\frac{M_p}{2}\right)\right)^2} & \text{if } M_p > 1.45, \end{cases} \quad (5.17)$$



(a)

(b) PSC

(c) ASC

(d) RBFANN

(e) Kriging

(f) DKG

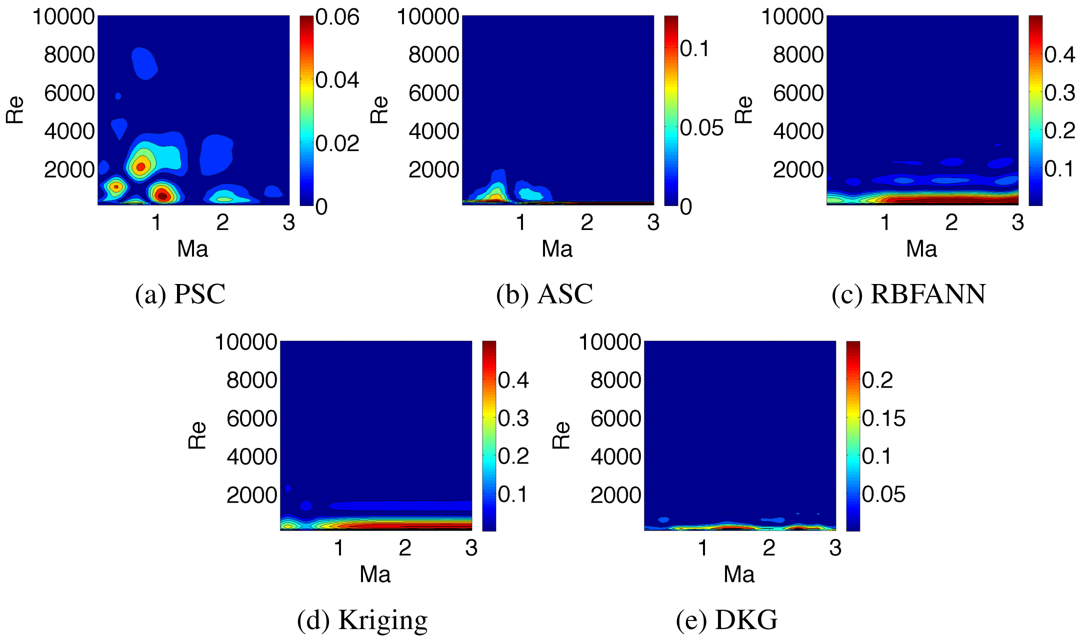
Figure 5.14: The (a) exact contours and representative contour plots for the drag coefficient of Boiko et. al.[15] using (b) a PSC method, (c) an ASC method, (d) an RBF ANN, (e) a Kriging method and (f) the DKG method

$$G = \begin{cases} 1.0 - 1.525 M_p^4 & \text{if } M_p \leq 0.89, \\ 0.0002 + 0.0008 \tanh\left(12.77\left(M_p - 2.02\right)\right) & \text{if } M_p > 0.89, \end{cases} \tag{5.18}$$

The function produces very steep gradients arranged in a series of steps in the interior of the domain. The contour of the hypersurface of Equation (5.16) is shown in Figure 5.17a.

As shown in Figure 5.16, most of the metamodels show a first order convergence rate in the normalized sum-of-squares error, but the ASC method converges exponentially. Similar to the drag model discussed in the previous section, because the adaptive refinement algorithm in the ASC adds additional training nodes at highly non-linear regions in the hypersurface, the ASC method converges exponentially.

As opposed to the function given by Equation (5.15), many of the complex characteristics of Loth's model lie in the interior of the domain as can be seen in 5.17a.



(a) PSC

(b) ASC

(c) RBFANN

(d) Kriging

(e) DKG

Figure 5.15: Local error plots for the approximation of drag model of Boiko et. al. [15] using (a) a PSC method, (b) an ASC method, (c) an RBF ANN, (d) a Kriging method and (e) the DKG method.

But from Figure 5.2, it can be seen that the number of nodes of the PSC method are scarce in the interior of the domain. This lack of training data along with the requirement that the approximation be exact on the training nodes causes spurious oscillations in regions of high gradient (Figure 5.17). Because the Kriging and DKG methods also interpolate the hypersurface through the available inputs, spurious oscillations can also be seen in the hypersurface predicted by the Kriging method and the DKG method, and is shown in Figure 5.17. The oscillations in the DKG method is more localized than the Kriging method. To investigate this, the correletion model used in the DKG method is studied, and it is found that for any given number of input points, the DKG method approximated the hypersurface of Equation (5.16) using a General Exponential correletion model. The correlation model used in the Kriging method is a Gaussian model, but a General Exponential correlation model is used in the DKG method. A General Exponential model

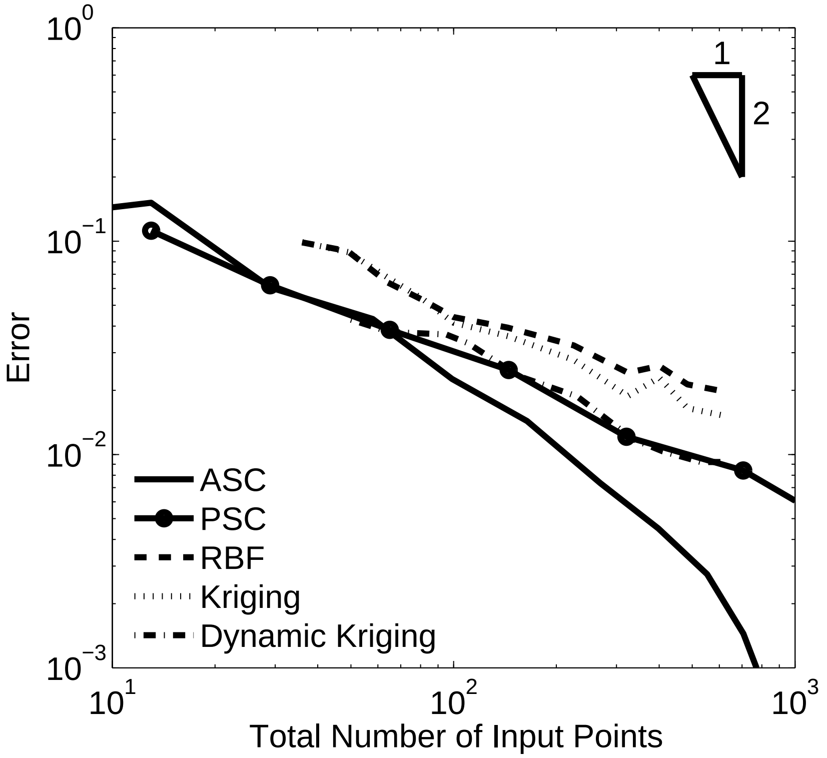
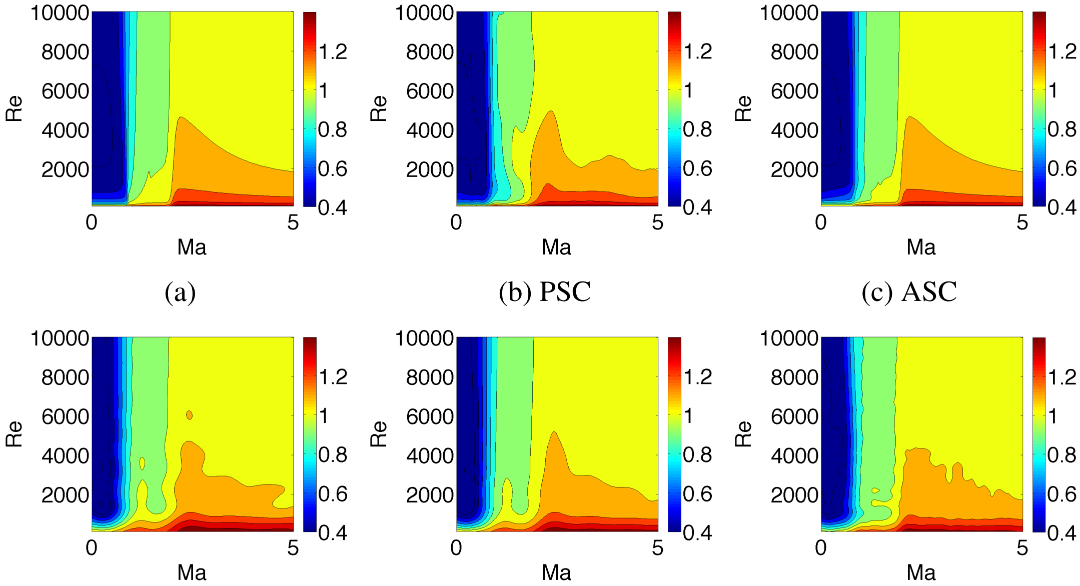


Figure 5.16: Error plot showing the convergence rates on approximating the drag model of Loth [88] using a PSC method, an ASC method, an RBFANN, a Kriging method and the DKG method.

is more localized than a Gaussian and the use of the General Exponential correlation model approximates the localized features of the highly non-linear portions of the hypersurface. Because the RBFANN method also uses non-compact Gaussian basis functions, spurious oscillations can also be seen in the hypersurface approximated by the RBFANN in Figure 5.17. Furthermore, the ASC model also uses local basis functions and adaptively places additional nodes near the higher gradient regions of the input domain thus eliminating these oscillations in Figure 5.17.



(a)

(b) PSC

(c) ASC

(d) RBFANN

(e) Kriging

(f) DKG

Figure 5.17: The (a) exact contours and representative contour plots for the drag coefficient of Loth [88] using (b) a PSC method, (c) an ASC method, (d) an RBFANN method, (e) a Kriging method and (f) the DKG method.

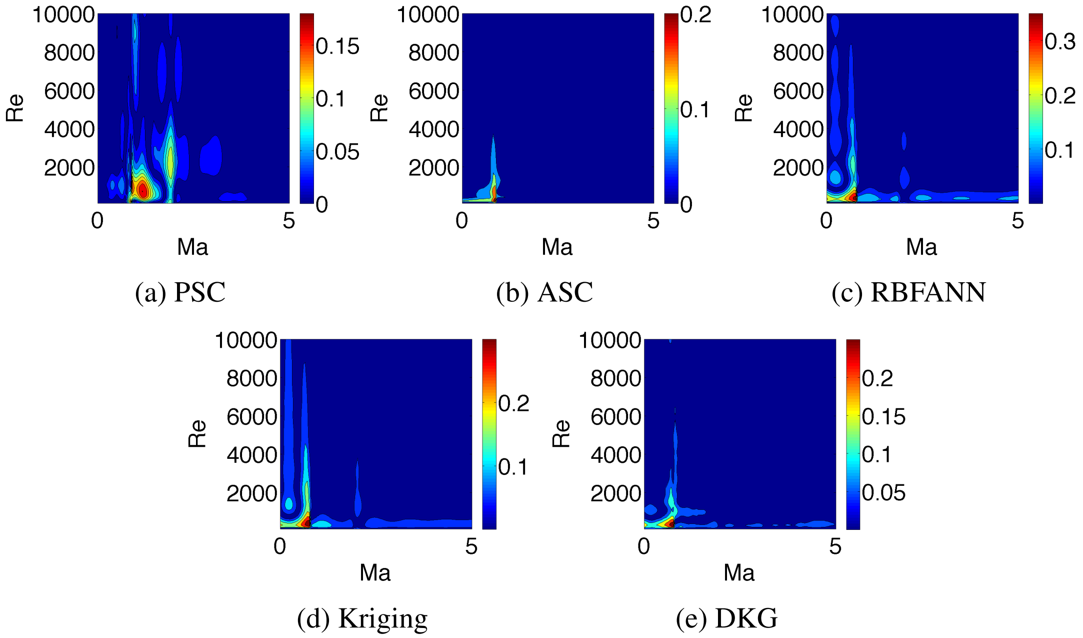
5.5.3 Tong's Model for Drag on a Particle in a Shocked Flow

Tong et. al. [143] have extended Loth's model [88] to include variations of particle shape and particle volume fraction, α,

$$C_D = C_{da} + 0.5048\alpha\left(1.0 + \frac{34.8}{Re_p^{0.5707}}\right)^4 + 0.9858\alpha\left(1.0 + \frac{34.8}{Re_p^{0.5707}}\right), \qquad (5.19)$$

where C_{da} is the drag coefficient calculated using (5.16). This model illustrates the complex dependence of the particle drag coefficient on many different parameters, including the Knudsen number (set equal to 10 here), M_p (set to 1 in here), Re_p (varied) and α (varied). The hypersurface is shown in Figure 5.20a.

The convergence of the metamodels with respect to the number of input points is shown in Figure 5.19. The PSC method and the DKG method approximate the



(a) PSC (b) ASC (c) RBFANN

(d) Kriging (e) DKG

Figure 5.18: Local error plots for the approximation of drag model of Loth [88] using (a) a PSC method, (b) an ASC method, (c) an RBFANN method, (d) a Kriging method and (e)the DKG method.

hypersurface most accurately. Despite the higher concentration of input nodes in the low Re_p, high α boundary, the local basis functions in the ASC method do not approximate the function as well. The RBFANN method, employing Gaussians as basis-functions, over-fits the nearly linear variation of the drag coefficient at higher volume fraction and lower Reynolds number. Therefore, as is shown in the contour plots of the reconstructed hypersurfaces in Figure 5.20, oscillations arise when the RBFANN method is used to build the approximation. Similar to the case of the harmonic function in Section 5.4, the Kriging model constructed using the DACE code does not converge monotonically. The value of the shape parameter as calculated using the Kriging method and the DKG method are compared against the number of input points in Table 5.3. The value of the shape parameter, θ, in the Kriging method is equal to 0.76655 for less than or equal to 400 input points but increases sharply to $\theta = 20$ past 400, where a non-monotonic jump

Table 5.3: Correlation parameters estimated by the Kriging method and the DKG method to approximate the hypersurface given by 5.19

Number of Inputs	θ(Kriging)	θ(DKG)
36	0.76655	0.9913
64	0.76655	1.3370
81	0.76655	2.7159
100	0.76655	2.5089
144	0.76655	2.8116
225	0.76655	3.5402
324	0.76655	5.8234
400	20	6.6144
529	20	8.3136
625	20	9.0773

is seen in the convergence plot in Figure 5.19. For the DKG method, the value of θ monotonically increases, indicating that the correlating model becomes more localized with an increased number of input points. This numerical example also illustrates the advantage of the GPS algorithm [133] for determining the optimum value of θ over the modified Hooke and Jeeves algorithm use in DACE.

5.6 Conclusions

The performance of five metamodeling techniques, the PSC, ASC, RBFANN, Kriging and DKG methods, is compared for use as the coupling algorithm or a metamodel in a multi-scale solver. The magnitude and the rates of the representation error of each of these methods has been characterized by their sum-of-squares error from Equation (5.10) and the local errors (5.11).

For a large number of training points, the SC methods generally approximate most of the hypersurfaces most acurately. In particular, the adaptive refinement of the ASC method around steep gradients on the interior of the input domain captures the

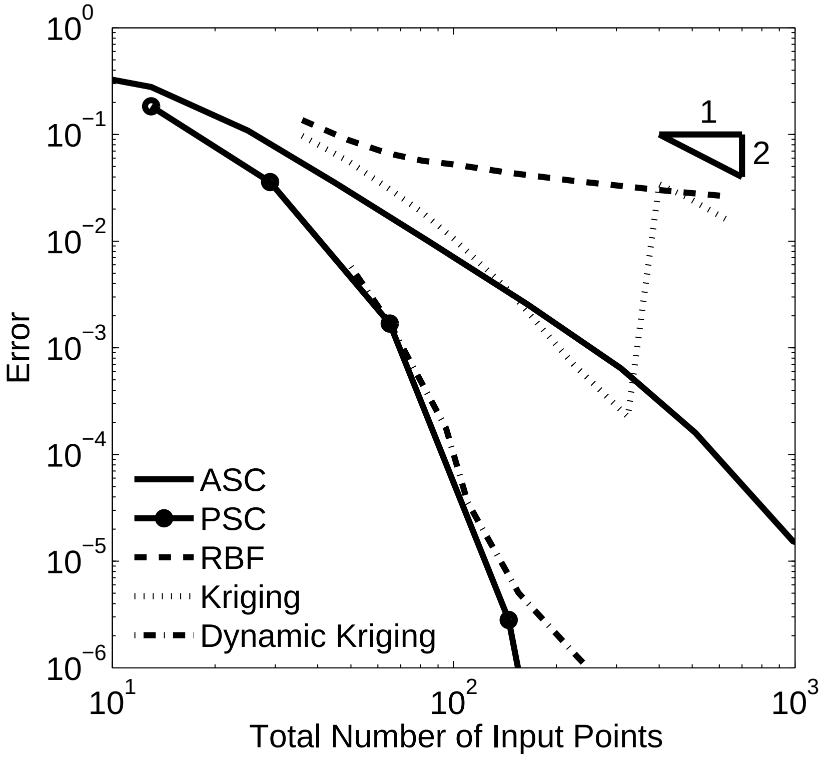
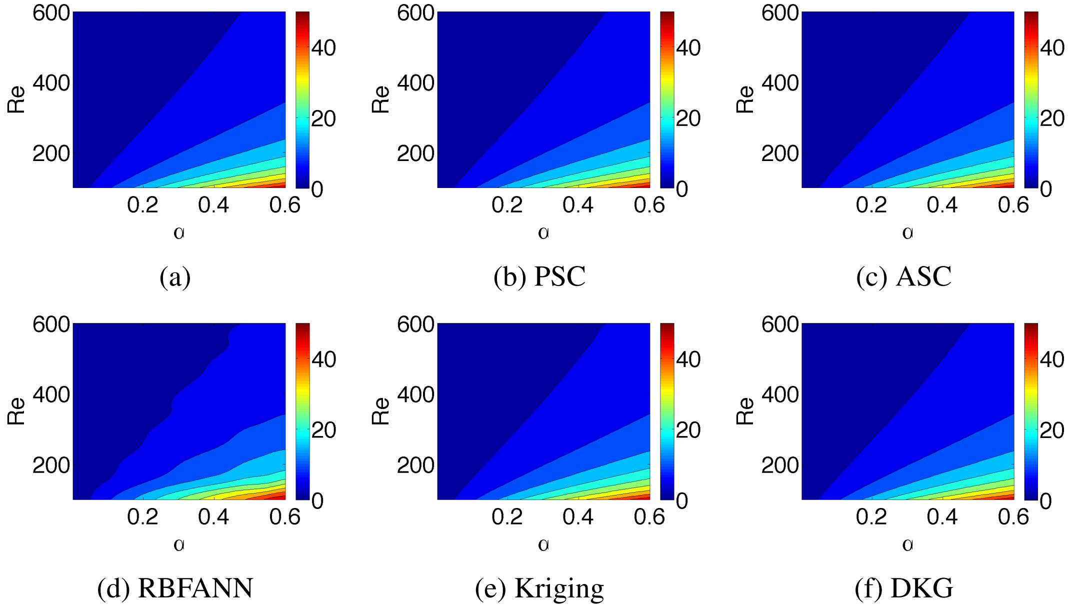


Figure 5.19: Error plot showing the convergence rates on approximating the drag model of Tong et. al. [143] using a PSC method, an ASC method, an RBFANN, a Kriging method and the DKG method.

complex regions of high gradient in the hypersurfaces of the empirical drag functions tested. But the number of input points required to accurately predict a hypersurface using the SC methods is roughly equal to or more than 100 for most of the hypersurfaces. Because in a multiscale modeling framework, input points correspond to high resolution mesoscale computations, generation of such a high number of input points is expensive. Additionally, both the PSC and ASC methods are constructed using a strict predetermined nodal architecture and lack the flexibility of the Kriging and the RBFANN methods with respect to placement of input data. For example, with the SC-based methods, expanding the parameter space would entail discarding the input from a previous set of data or



(a)

(b) PSC

(c) ASC

(d) RBFANN

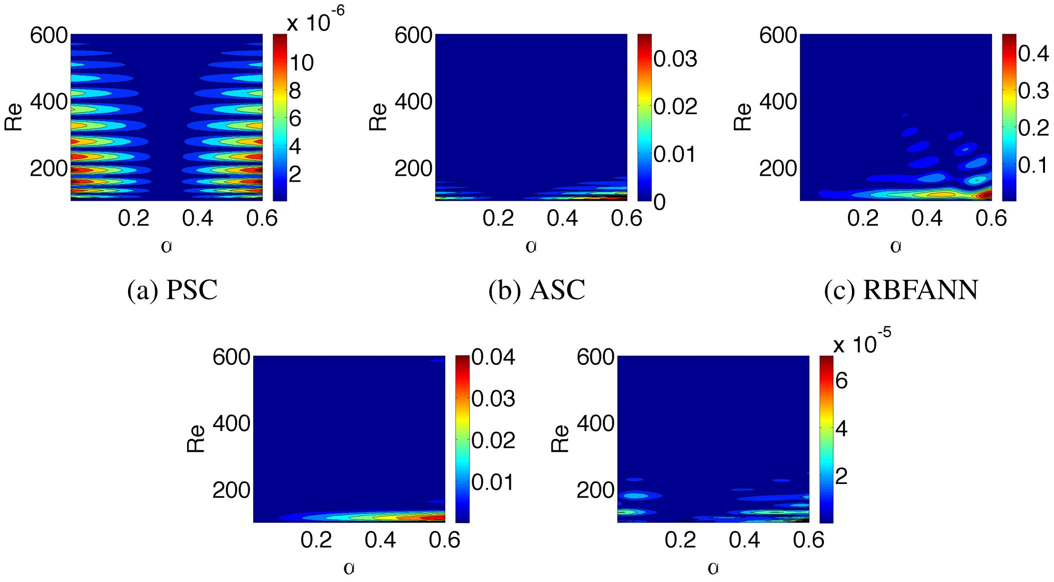
(e) Kriging

(f) DKG

Figure 5.20: The (a) exact contours and representative contour plots for the drag coefficient of Tong [143] using (b) a PSC method, (c) an ASC method, (d) an RBFANN, (e) a Kriging method and (f) the DKG method.

introducing additional interpolation errors. This would result in waste of computational time and resources when an expanded parameter space is required.

The input points of the RBFANN and the Kriging methods can be randomly placed throughout the domain with little or no effect on the convergence of the metamodel, as seen in Section 5.4.3. Because of this flexibility, the parameter space can be expanded to include a larger domain of approximation while continuing to utilize previous data. However, the RBFANN and Kriging methods have the highest sum-of-squares error in approximating most of the functions tested and do not converge at as high of rates as the SC methods. Additionally, the Kriging method using the DACE code does not converge monotonically in some cases. The parameter estimation technique integrated within the DACE code (i.e. the use of modified Hooke and Jeeves algorithm) leads to the selection of a local extremum value of the shape parameter θ as the global extremum in the maximum likelihood estimation process.



(a) PSC

(b) ASC

(c) RBFANN

(d) Kriging

(e) DKG

Figure 5.21: Local error plots for the approximation of drag model of Tong et. al. [143] using (a) a PSC method, (b) an ASC method, (c) an RBF ANN, (d) a Kriging method and (e) the DKG method.

The non-monotonic convergence of the Kriging method is circumvented in the DKG method by a Global Pattern Search (GPS) algorithm using a maximum likelihood estimator with a penalty function and by the use of dynamic selection of correlation models and mean structure. The DKG method is not only monotonically convergent for all the functions considered in the current work, but at roughly 100 input nodes, has either the lowest sum-of-squares error or is close to the lowest (i.e. relative to the SC methods). Therefore, metamodels may be built using less than 100 training points using the DKG method. Thus, for the functions approximated in the current work, the DKG method is the optimal choice to serve as the coupling algorithm for the multi-scale solver.

Sections 5.1 to 5.6, in full, is a reprint of the material as it appears in Evaluation of Convergence Behavior of Metamodeling Techniques for Bridging Scales in Multi-scale Multimaterial Simulation 2015. Sen, Oishik; Davis, Sean; Jacobs, Gustaaf; Udaykumar, H.S. Elsevier Press, 2015. The dissertation/thesis author was a primary investigator and author of this material.

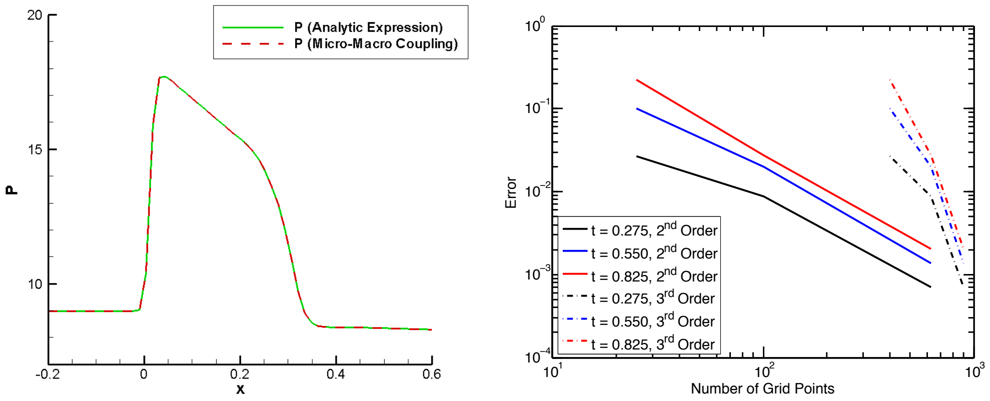
5.7 Analysis of 1D Shock-Particle Cloud Interaction

To validate the micro-macro coupling methods, we study the 1D case of a normal right running shock interacting with a cloud of particles in an initially quiescent flow found in Ref. [15, 68]. We compute the shock tube problem in the domain $x \in [-5, 6]$ with the initial pre-shock flow state $[\rho_R, u_R, p_R] = [1, 0, 1]$. The post-shock flow state is computed using the Rankine-Hugoniot relations for Mach number $M_s = 2.8$ with the shock initially at $X_s = 0$. Reflective boundary conditions are used at both ends of the domain. The grid spacings were taken to be $\Delta x = 0.055$

The initially stationary particle cloud is seeded evenly on the interval $[0, 0.2981]$ with a volume concentration of 3%. The density and particle response time are $\rho_p = 1200$ and $\tau_p = 3.9296 \times 10^3$, respectively. We take the Reynolds number as $Re_f = 1.7638 \times 10^6$ to compute the particle traces according to the experiment in Ref [15].

We wish to isolate the online step of calculating the individual particle drag values from the offline step of building the database because the DNS computations needed to build the database are computationally expensive. Using input values of 25 Mach numbers ranging from 0.1 to 2.75 and 25 Reynolds numbers ranging from 100 to 10,000, we computed a database of particle drag values with Eq. 5.15. We then calculated each corresponding particle drag coefficient from the database using the method described above with a second-order Lagrange interpolation.

Figure 5.22a shows the fluid pressure profile evolution at time $t = 0.275$ using both the analytic equation directly and the micro-macro coupling scheme. We see that the two profiles are nearly collinear, showing excellent agreement between the two methods. Convergence rates for various interpolation orders and particle drag grid sizes are shown in Fig. 5.22b as a validation of the macro-scale implementation of the multi-scale algorithm.



(a) (b)

Figure 5.22: One dimensional shock particle interaction validation example showing (a) a snapshot of the pressure profile at $t = 0.275$ using the empirical drag model from Eq. 5.15 and a 2nd order Lagrange interpolation on a 25x25 grid of particle drag data and (b) convergence rates of second and third order interpolations on varying sized grids.

Chapter 6

Normal Instabilities

6.1 Introduction

Sections 6.1 to 6.5 are to be submitted to the AIAA Journal under the title, "Carbuncles in high resolution Eulerian-Lagrangian simulations of Shock-Particle-laden flow."

High-speed flows are important in many technological environments, including supersonic and hypersonic aircraft. In these environments, particles interact with shocks and turbulence. Liquid or solid fuel particles in scramjet combustors and rocket engines, for example, are injected into a high-speed chemically reacting fluid causing shock waves to form. Many physical and numerical instabilities that arise in the analysis of particle-laden shocked flows must be resolved. Modeling the complex, multi-material particle-fluid interactions poses high demands on models and methods.

Shock waves, in particular, challenge numerical methods because they are difficult to capture in a stable, non-oscillatory manner. Traditionally, low order methods have beens employed that provide a robust regularization of discontinuities but do so with excessive numerical diffusion. To accurately model small scale flow structures, such as turbulent eddies, over a long period of time, higher order/resolution (greater than third order) shock capturing methods have been shown to reduce dissipation and dispersion errors [6, 68]. High-order methods on the other hand are plagued by numerical Gibbs oscillations, which arise near discontinuities.

Weighted Essentially Non-Oscillatory (WENO) schemes combine the benefits of both high-order and low-order methods by adding numerical dissipation only near the shock [18, 21] thereby mitigating the effects of Gibbs oscillations. The nonlinear weighting employed by the WENO method increases numerical dissipation only in areas near strong gradients, capturing discontinuities while maintaining a high order of accuracy in smoother areas of the flow. This type of method has proven very effective for the simulation of the fine scale and delicate structures of turbulent flows involving shocks.

Computing the entire flow over each particle, tracing individual solid or liquid complex particle boundaries along their paths and tracking shock waves in the moving framework of particle-laden shocked flow makes direct numerical simulation too computationally expensive to be feasible for process scale problems. In order to solve realistic problems, approximations to the full equations must be employed. Eulerian Lagrangian (EL) methods have been successfully implemented to solve particle-laden flows [10, 28, 68]. In particular, EL methods which model particles as points such as the Particle-Source-in-Cell (PSIC) method [30] have been shown to efficiently and accurately resolve particle-fluid interactions. In PSIC, the carrier gas is solved in the Eulerian frame on a stationary mesh, while individual particles are modeled as point sources and traced along their paths in a Lagrangian frame. By interpolating the fluid velocity and temperature onto the Lagrangian frame and using a deposition function to place the particle's drag influence onto the Eulerian frame, the two phases are coupled and can be solved simultaneously for large engineering applications.

While using WENO methods essentially eliminate spurious Gibbs oscillations near discontinuities, capturing stronger shocks in high Mach number flows creates another numerical instability when using high-order methods. In situations where a very strong, slow moving shock is aligned with the numerical grid, anomalous bumps and oscillations can occur in the shock profile which have earned the nickname *carbuncle* [111]. These phenomena have been studied in the case of solid boundaries immersed in high-speed flows [112, 111, 152]. Particles have a dissipative effect on the fluid phase and may mitigate the development of carbuncle instabilities. It is not clear whether carbuncles are found in the computation of shocked particle-laden flows and what effect, if any, do these instabilities have on the particle phase.

Experiments by Holden [60] have revealed that carbuncle-like phenomenon may arise naturally by the injection of dust particles along the stagnation line upstream of a blunt cylinder in a very high speed flow. Through a linear stability analysis of the motion of a discontinuity in an inviscid fluid, Robinet et. al. [114] derived a new mode, which could develop into a carbuncle as the instability becomes nonlinear. However, aside from these fabricated cases, experiments have not shown these instabilities to occur naturally. While some researchers have hypothesized that carbuncles are incurable, [38], we feel it safe to assume that carbuncles are an artificial numerical phenomenon that needs to be mitigated. Many “cures” have been proposed to eliminate carbuncles from numerical solvers [108, 120]. These methods generally add excessive dissipation, limiting the accurate resolution of small scale structures. While the Holden experiments used particles to trigger carbuncle-like instabilities [60], the effect of carbuncles on the development of particle-laden flows has not been analyzed.

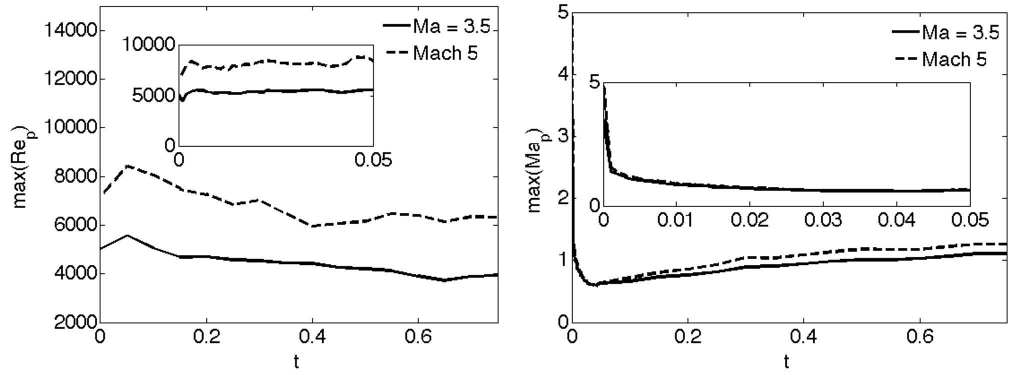
In this paper we report on the impact of carbuncles on the particle-laden flow developments of a cloud of particles interacting with a high-speed uniform flow using our in-house high fidelity EL solver [68]. We demonstrate, for the first time, the appearance of carbuncle instabilities in particle-laden flows and investigate their impact on both the particle and fluid phases. We then examine a method to mitigate these instabilities by adjusting the power parameter of the nonlinear weights in a WENO-Z code [35].

We present a brief description of the fifth order WENO scheme used in our computations. In Section 6.3, we discuss the formation of carbuncles in high-speed flows interacting with clouds of particles as well as the effect that they have on the carrier flow and particle phase. We then demonstrate in Sect. 6.4 how to mitigate the effect of carbuncles on the carrier flow by adjusting the power parameters of the WENO-Z nonlinear weights. Conclusions are given in Section 6.5.

6.2 Problem Setup

In our experiments, a 4% volume fraction cloud of 3.9×10^9 real particles is approximated using $39,762$ computational particles. The particle response time and density are, $\tau_p = 100.0$ and $\rho_p = 7.42 \times 10^4$, respectively. The particles are initially evenly distributed with zero velocity in a rectangular cloud stretching from 0.175 to 0.352 in x and -0.044 to 0.044 in y with 282×141 particles.

We consider a WENO mesh with 500×500 points in the x and y directions respectively. The computational domain spans an area of 0 to 0.5 in the x-direction and -0.25 to 0.25 in the y-direction. A uniform inflow boundary condition is specified at $x = 0$ and outflow boundary conditions are specified at all other boundaries. We use a Reynolds number around the particles of $Re_f = 3.387 \times 10^7$ to compute the drag force according to Davis et. al. [33]. Experiments are performed with initially uniform flows where the fluid Mach numbers ranges from 3.5 to 5. The maximum particle Mach and Reynolds numbers, M_p and Re_p respectively, shown in Fig. 6.1, stay within the physical limits of the drag correction factor, f_1, specified by Boiko et. al. [15]. While an initial acceleration in M_p occurs and exceeds the bounds of f_1, M_p decays rapidly as the particles accelerate and does not effect the particle physics in the computation.



(a) (b)

Figure 6.1: Maximum particle (a) Reynolds number and (b) Mach number vs. time for a uniform flow interacting with an initially rectangular cloud of particles. The embedded figures show the maxima for $t = 0$ to $t = 0.05$.

6.3 The Formation and Impact of the Carbuncle Instability

Carbuncles appear as anomalous oscillations in the profile of a strong shock, as seen by the spike in the pressure contour plot of a uniform flow of Mach number 5 interacting with an initially stationary cloud of particles at $t = 0.5$ shown in Fig. 6.2. Carbuncles form only along the grid-aligned portion of the resultant bow shock because they are damped by the added numerical dissipation when the shock and numerical grid are misaligned.

The instability "bleeds" [64] downstream behind the shock, causing ripples in the post-shock flow profile. This downstream influence can be seen in the Mach number contours in Fig. 6.3, where the sharp lines and spikes are artifacts of the carbuncles that appear in the bow shock. The carbuncle instabilities cause alternating high and low

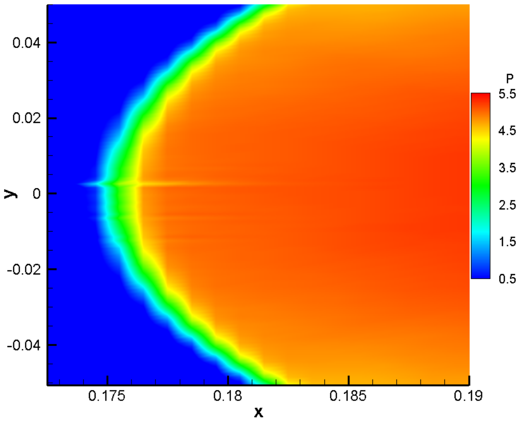


Figure 6.2: Pressure contours of the bow shock at $t = 0.5$ for a cloud of initially stationary particles interacting with a uniform Mach 5 flow. The spike seen at the stagnation line is the carbuncle instability.

velocity streaks to flow downstream (Fig. 6.3).

The lower velocity streaks from carbuncles in the bow shock cause instabilities, in turn, in the particle phase to form. Just above the centerline in Fig. 6.4a, a small grouping of particles protrudes upstream of the otherwise sharp leading edge of the cloud. These outcroppings have a lower downstream acceleration as compared to the bulk of the cloud because of the interaction with a lower velocity fluid. The combination of the lower velocity fluid and the finger shaped particle phase protrusions cause strong instabilities in the fluid density gradients just downstream of the particle phase instability (Fig. 6.4b).

6.4 Mitigation of the Carbuncle Instability

To reduce or eliminate the numerical errors caused by carbuncle instabilities, dissipation is generally added either directly (via numerical viscosity) or by rotating the numerical grid [108, 120]. Increased numerical dissipation over the entire domain limits the ability of the method to accurately capture small-scale turbulent structures. However,

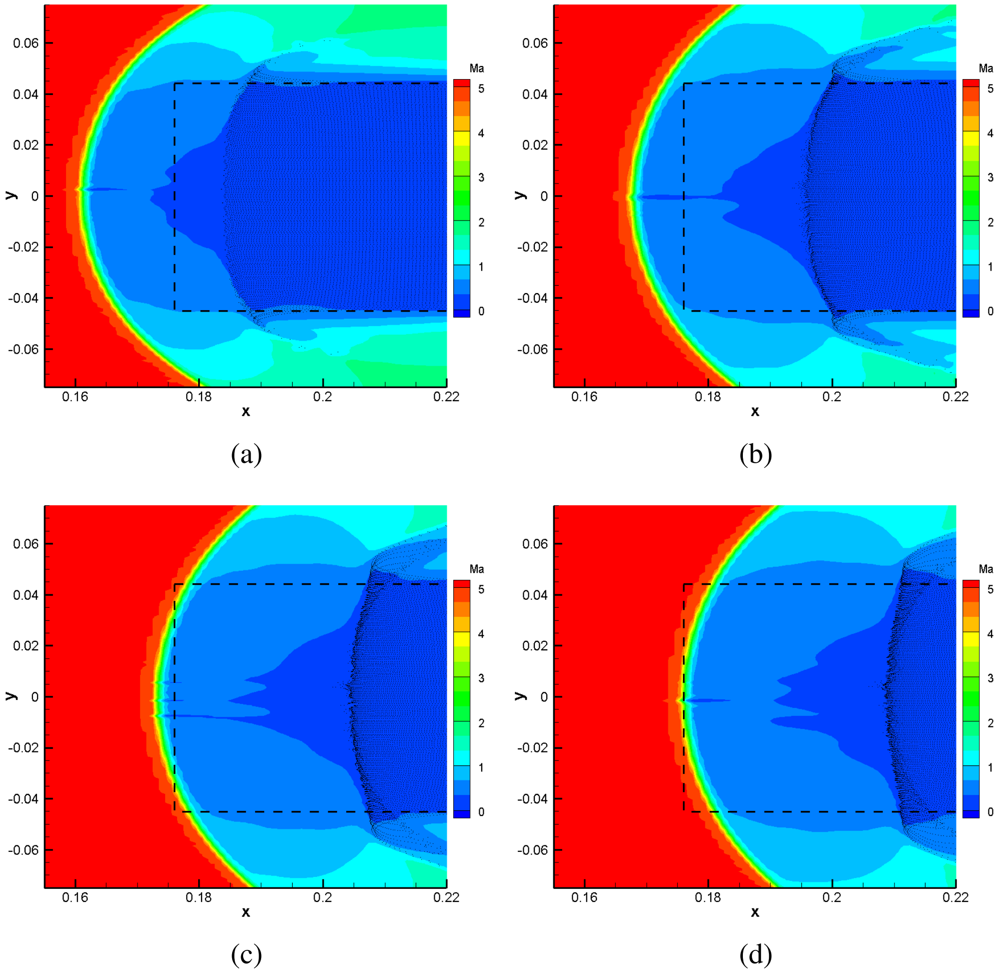
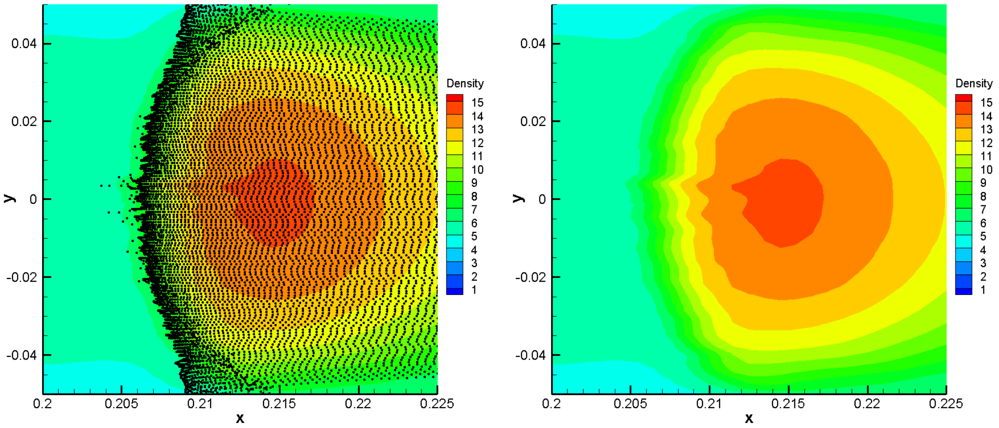


Figure 6.3: Mach number contours at (a) $t = 0.23$, (b) $t = 0.38$, (c) $t = 0.48$ and (d) $t = 0.53$ for a cloud of initially stationary particles interacting with a uniform Mach 5 flow. The spikes behind the bow shock show the bleeding of the carbuncle downstream. The the particle dispersion pattern is shown downstream of the bow shock. The dotted line shows the initial location of the particle cloud at $t = 0$.



(a)

(b)

Figure 6.4: Density contours of the flow at $t = 0.5$ for a cloud of initially stationary particles interacting with a uniform Mach 5 flow. The anomalous bumps in the density contour at the front edge of the particle cloud, in (b), show the downstream effects of the carbuncle instability on the particle phase.

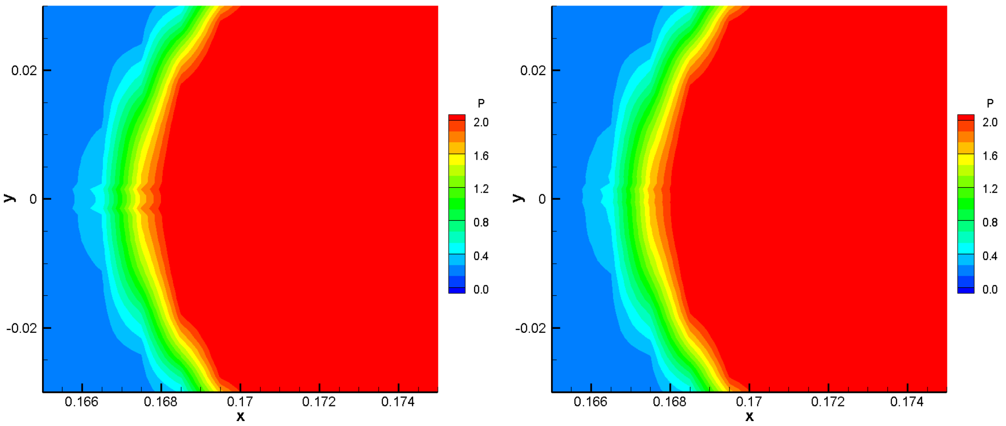
by properly adjusting the WENO-Z nonlinear weights from (3.7), the method retains its high fidelity in smooth areas of the domain while damping out carbuncle instabilities near strong discontinuities.

In stencils containing a discontinuity, the substencils, S_k, are categorized as either smooth (β_k is $O(\Delta x^2)$) or non-smooth (β_k is O(1)) [35]. Because τ_5 from (3.6) is of the same order as the largest β_k, the factor,

$$\left(\frac{\tau_5}{\beta_k + \varepsilon}\right)^p, \tag{6.1}$$

from (3.7) is $O(1)$ in a non-smooth substencil and $O(\Delta x^{-2p})$ in a smooth substencil [35]. The larger the power parameter, p, is, the greater the scale separation in the associated weight between smooth and non-smooth substencils. To increase the damping of instabilities near discontinuities, we adjust the nonlinear weights, ω_k^z, by increasing p. The contribution of the flux from the non-smooth substencils becomes vanishingly small [35]. With a decreased weight for the discontinuous substencil, the WENO method increasingly resembles a third order upwinding ENO scheme near shocks while in smooth areas of the flow, the term inside the brackets from (6.1) is $\ll 1$ for all substencils and p does not greatly effect the nonlinear weighting.

To demonstrate the effects of changing p, consider the interaction of an initially stationary cloud of particles and a uniform Mach 3.5 flow shown in Fig. 6.5 at $t = 0.75$. The added numerical dissipation long the leading edge of the bow shock reduces the amplitude of carbuncles and mitigates their downstream effects, as seen in comparing the pressure contours in Fig. 6.5a where $p = 2$ with Fig. 6.5b where $p = 3$. With less scale separation in the weighting term between the smooth and discontinuous substencils in Fig. 6.5a, the carbuncles persist downstream of the shock. While carbuncles still arise along the leading contours of the shock with $p = 3$, the instability is damped out on the downstream edge in Fig. 6.5b because of the increased dissipation near the strong shock.



(a) (b)

Figure 6.5: Magnified view of the pressure contours at the front of the bow shock at time $t = 0.75$ when an initially stationary cloud of particles is placed in a Mach 3.5 uniform flow. Carbuncles can be seen in (a) using a power parameter $p = 2$ whereas the bow shock pressure contour is much more stable when the power parameter is $p = 3$, as shown in (b).

6.5 Conclusions

The formation of carbuncles in a particle-laden flow is presented using a high-order EL solver. The effects of the carbuncle bleeding downstream in alternating high and low velocity regions furthermore causes an instability to form in the particle phase, which manifests as a spike of particles protruding from an otherwise well-defined leading edge. The strong two-way coupling of the particle and fluid phases amplifies the effects of the carbuncle, causing a strong instability in the density gradient of the fluid just downstream of the particle instability. The effects of the carbuncle instability can be mitigated by increasing the power parameter in the nonlinear weighting of the WENO-Z scheme, thereby adding dissipation to areas of the flow containing steep gradients without compromising the high order accuracy of the scheme in smoother regions of the flow.

Sections 6.1 to 6.5, in full, are to be submitted to the AIAA Journal under the title, Carbuncles in high resolution Eulerian-Lagrangian simulations of Shock-Particle-laden flow. Davis, Sean; Don, W.S.; Jacobs, Gustaaf. The dissertation/thesis author was the primary investigator and author of this material.

6.6 Gibbs Oscillations Occurring within a Multidimensional ENO Stencil

The ENO method, outlined in Section 3.1 is employed to essentially eliminate oscillations in the interpolation of carrier phase variables to the particle locations. However, in the case of a shock entering the host cell of a particle at an angle, the method of lines is ineffective at determining the proper sub-stencil and causes instabilities. Figure 6.6a shows the sub-stencil chosen by the algorithm if only those divided differences along the x and y grid-lines from the particle host cell location are used.

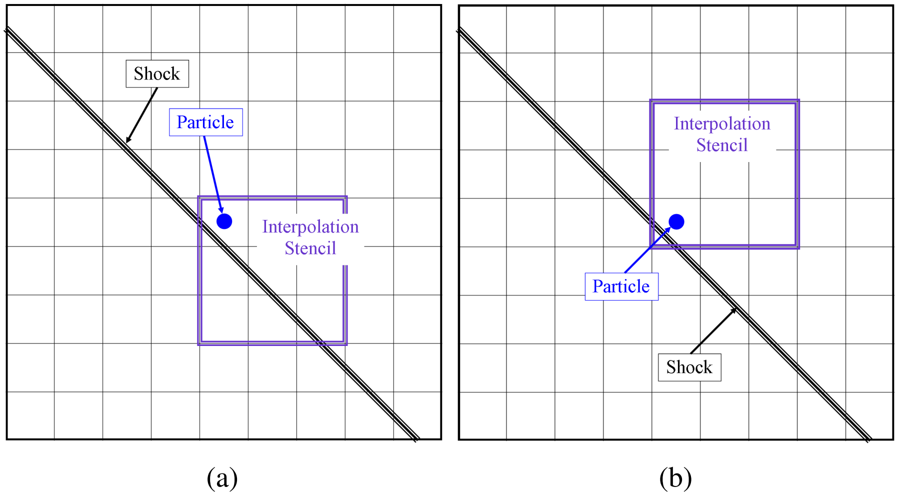


Figure 6.6: Schematic of the ENO stencil chosen by comparing the divided differences over (a) the grid lines running in the $x-$ and $y-$directions from the particle's host node (method of lines) and (b) the summation of the divided differences in each possible 3^2 sub-stencil surrounding the particle. The particle is shown in blue and the shock is shown as a triple line in black.

A more robust algorithm for finding the proper ENO sub-stencil must account for the divided differences over more than just the orthogonal lines stretching from the particle host node. Instead of looking at only the smoothness along these lines, the divided differences on all nodes within the subdomain created by these lines are can be summed to account for angled discontinuities with respect to the uniform grid. However, adding all of the divided differences for larger order (9^{th}, for example) in multiple dimensions adds an unnecessary computational strain. If the shock is far from the particle, the method of lines accurately predicts the most stable sub-stencil. By only summing all of the divided differences within a 3^n box, where n is the number of dimensions, and adding it to the method of lines approach, a more accurate and still efficient sub-stencil is found, as shown in the schematic in Figure 6.6b.

Chapter 7

Wake Instabilities

Sections 7.1 to 7.4 are taken from a manuscript published in the Journal of Applied Mechanics and Technical Physics in 2013 [33].

7.1 Introduction

Particle-laden and droplet-laden flows play an important role in high-speed technologies such as solid rocket propulsion systems and high-speed liquid-fuel combustors. Shock waves occur in scramjet combustors and interact with fuel particles in the resultant supersonic flow. Accurate tracing of fuel droplet trajectories would allow for more precise ignition and combustion, boosting overall engine efficiency. The physical complexity of simultaneously resolving particle-turbulence, shock-turbulence, and shock-particle interactions along with the large range of spatial and temporal scales pose high demands on both experimental and computational analysis. Experimental analysis is limited to large scale observation due to the high velocities, while first principle computations are expensive and models are not optimal.

Significant effort has gone toward the improvement of empirical governing models for the particle phase through investigation of the acceleration of an individual particle behind a shockwave. Boiko et al. [13] determined the drag of a droplet behind a shock by comparing the known relaxation times of a hard sphere to their experimentally measured droplet relaxation times. Sun et al. [139] numerically studied the dynamic drag coefficient of a spherical particle behind a shock wave. A reflected bow shock was observed in front of the spherical particle, and, as the shock wave traversed over the sphere, a Mach reflection formed. The Mach reflection proceeded to the rear center of the sphere before converging with the Mach reflection from the other side. This caused shock focusing to occur and a region of very high pressure at the rear of the spherical particle. This region of high pressure resulted in a brief negative drag. Their numerical data matched experimental data within 10%. Loth [88] investigated the effect of compressibility and rarefaction on a spherical particle. Boiko et al. [14] also studied different shaped particles. In a comparison of a cubical and a spherical particle they found that the drag is predominantly a function of the frontal area of the particle. Therefore the relative bluntness of the shapes did not significantly affect the particle dynamics.

A limited number of studies have been performed on the dynamics of a large number of particles in high-speed flow. Olim et al. [106] studied the attenuation of a normal shock wave in a homogeneous gas particle mixture. Kiselev et al. [78] compared simulations based on Boiko's empirical particle models to shock tube experiments on the dispersion of a cloud of plexiglas and bronze particles in the accelerated flow behind a moving shock [78, 15]. Not only did they visualize the particle dynamics and dispersion, they also matched some of their results quantitatively to the experimental dispersions. In Jacobs et al. [65, 66], a high-order/high-resolution Eulerian-Lagrangian scheme was developed that was based on the same empirical physical governing model as proposed by Kiselev et al. Extensive high-order/high-resolution results showed good comparison with experimental results, while small scale turbulent structures were resolved with improved efficiency. In Jacobs et al. [65], we studied the effect of initial shape of the cloud's geometry on the dispersion of particles. It was shown that the aerodynamics of the shape significantly alter the cross-stream dispersion of particles.

This paper extends the investigation of the effect of the geometry of the cloud on the dispersion of particles in the accelerated flow behind a moving shock. We study the effect of aspect ratio and rotation of the cloud and present detailed statistics of the particle dispersion in terms of the averaged particle coordinates. This investigation is part of our ongoing effort to thoroughly validate Eulerian-Lagrangian methods against shock tube experiments for shocked, particle-laden flow.

In Section 7.2 we present the physical model and the governing equations, and briefly review the high-order/high-resolution Eulerian-Lagrangian method. In Section 7.3 we first briefly review the effect of particle cloud shape on the dispersion of particles, followed by a discussion of the effect of aspect ratios and rotation of the cloud shapes on the particle-laden flow development. Concluding remarks and future directions are given in Section 7.4.

7.2 The Physical Model and Governing Equations

In the particle-source-in-cell (PSIC) method, the Eulerian continuum equations are solved for the carrier flow in the Eulerian frame, while particles are traced along in the Lagrangian frame. In the following, we present the coupled system of Euler equations that govern the gas flow and kinematic equations that govern the particle motion. We shall denote the subscript p for the particle variables and f for the gas variables at the particle position. Variables without subscript refer to the gas variables unless specified otherwise. For more detailed discussion of the physical model and governing equations, readers are referred to Ref. [65].

7.2.1 Euler equation in the Eulerian frame

The governing equations for the carrier flow are the two-dimensional Euler equations in Cartesian coordinates given by:

$$\mathbf{Q}_t + \mathbf{F}_x + \mathbf{G}_y = \mathbf{S}, \tag{7.1}$$

where

$$\begin{aligned} \mathbf{Q} &= (\rho, \rho u, \rho v, E)^T, \\ \mathbf{F} &= \left(\rho u, \rho u^2 + P, \rho uv, (E+P)u\right)^T, \\ \mathbf{G} &= \left(\rho v, \rho uv, \rho v^2 + P, (E+P)v\right)^T, \end{aligned} \tag{7.2}$$

and

$$P = (\gamma - 1)\left(E - \frac{1}{2}\rho\left(u^2 + v^2\right)\right), \qquad \gamma = 1.4. \tag{7.3}$$

The equation of state closes the system of equations

$$T = \frac{\gamma P M^2}{\rho}, \tag{7.4}$$

where $M = U/\sqrt{\gamma RT}$ is a reference Mach number determined with the reference velocity, U and reference temperature, T. The source term, $\mathbf{S}$, accounts for the effect of the particles on the carrier gas and will be discussed in more detail below.

7.2.2 Particle equation in the Lagrangian frame

Particles are tracked individually in the Lagrangian frame. The kinematic equation describing the particle's position $\vec{x}_p$, is given as

$$\frac{d\mathbf{x_p}}{dt} = \mathbf{v_p}, \tag{7.5}$$

where $\mathbf{v_p}$ is the particle velocity vector.

The particles' acceleration is governed by Newton's second law forced by the drag on the particle. With particles assumed spherical, we take the drag as a combination of the Stokes drag corrected for high Reynolds and Mach number and the pressure drag leading to the following equations governing the particle velocity

$$\frac{d\mathbf{v_p}}{dt} = f_1 \left(\frac{\mathbf{v_f} - \mathbf{v_p}}{\tau_p} \right) - \frac{1}{\rho_p} \nabla P|_f, \tag{7.6}$$

where $\mathbf{v_f}$ is the velocity of the gas at the particle position, and ρ_p the particle density. The first term on the right hand side of equation (7.6) describes the particle acceleration resulting from the velocity difference between the particle and the gas. The second term in the right hand side of equation (7.6) represents the particle acceleration induced by the pressure gradient in the carrier flow at the particle position. The particle time constant $\tau_p = Re_f d_p^2 \rho_p / 18$, where d_p is the particle diameter, is a measure for the reaction time of the particle to the changes in the carrier gas. $Re_f = UL/\nu$ is the Reynolds number of the carrier gas flow with L a reference length and ν the dynamic viscosity. In this study, we assume Re_f large and we therefore do not model viscous effects in the governing Eulerian equations for the gas flow (7.1).

The term f_1 is an empirical correction factor that yields an accurate determination within 10% of measured particle acceleration for higher relative particle Reynolds number up to $Re_p = |\mathbf{v_f} - \mathbf{v_p}|d_p/\nu = 1 \times 10^4$ and relative particle Mach number up to $M_f = |\mathbf{v_f} - \mathbf{v_p}|/\sqrt{T_f} = 1.2$ and is given by

$$f_1 = \frac{3}{4}\left(24 + 0.38Re_p + 4\sqrt{Re_p}\right)\left(1 + \exp\left[\frac{-0.43}{M_f^{4.67}}\right]\right). \tag{7.7}$$

From the first law of thermodynamics and Fourier's law for heat transfer, the equation for temperature is derived as,

$$\frac{dT_p}{dt} = \frac{1}{3}\frac{Nu}{Pr}\left(\frac{T_f - T_p}{\tau_p}\right), \tag{7.8}$$

where $Pr = 1.4$ is the Prandtl number, taken as its typical value for air in this paper. $Nu = 2 + \sqrt{Re_p}Pr^{0.33}$ is the Nusselt number corrected for high Reynolds number.

7.2.3 Source term S for the Euler equation

Each particle generates a momentum and energy that affects the carrier flow. The volume averaged summation of all these contributions gives a continuum source contribution on the momentum and energy equation in (7.1) as:

$$\mathbf{S_m}(\mathbf{x}) = \sum_{i=1}^{N_p} \mathbf{K}(\mathbf{x_p}, \mathbf{x})\mathbf{W_m}, \tag{7.9}$$

$$\mathbf{S_e}(\mathbf{x}) = \sum_{i=1}^{N_p} \mathbf{K}(\mathbf{x_p}, \mathbf{x})(\mathbf{W_m} \cdot \mathbf{v_p} + \mathbf{W_e}), \tag{7.10}$$

where $\mathbf{K}(x,y) = \mathbf{K}(|x-y|)/V$ is a normalized weighing function that distributes the influence of each particle onto the carrier flow. $\mathbf{W_m} = m_p f_1(\mathbf{v_f} - \mathbf{v_p})/\tau_p$ and $\mathbf{W_e} = m_p(Nu/(3Pr))\left(T - T_p\right)/\tau_p$ are weigh functions describing the momentum and energy contribution of one particle, respectively. The term m_p is the mass of one spherical particle which can be derived from τ_p. N_p is the total number of particles in a finite volume V.

7.2.4 Flow and Particle Solver

The governing equations are discretized and solved using the WENO-Z based PSIC scheme proposed by Jacobs et al. [65, 66]. The nonlinear nature of the hyperbolic Euler equations admits finite time singularities in the solution even when the initial condition is smooth. It is important that the numerical methods employed avoid non-physical oscillations, also known as the Gibbs phenomenon, when the solution becomes discontinuous. Among many high order shock capturing schemes, the classical weighted essentially non-oscillatory conservative finite difference schemes (WENO) for conservation laws by Shu et al. [130] has been very successfully employed for the simulation of the fine scale and delicate structures of the physical phenomena related to shock-turbulence interactions [6, 92, 162, 49].

The improved version of classical WENO conservative finite difference scheme, namely, WENO-Z scheme, [18, 18] is ideally suited for computing a shock wave interacting with a cloud of particles due to the complicated shock structures of the problem, and the importance of preserving high order resolution to resolve the small scale interactions present in particle-laden shocked flow.

The essence of the WENO scheme is the use of adaptive stencils. The method creates a stencil over the computational domain, in which a smoothness indicator is employed to estimate within which substencils the shocks lie. The method then assigns an essentially zero weight to low order local interpolation polynomials of the flux based on the values from substencils that contain high gradients and/or shocks to prevent Gibbs oscillations.

In computational cells where no shock is present, a high order upwinded central finite difference scheme is used to calculate the flow properties. But in shocked regions, a centered interpolation will produce undesirable Gibbs oscillations. With an ENO interpolation, these oscillations are essentially removed. The ENO interpolation is only necessary in WENO-domains identified by the smoothness indicator. The WENO scheme ensures that the formal order of accuracy/resolution in the smooth regions of the simulation remains intact as best as one can do.

The flow solver calculates the parameters of the flow by solving the Euler equations with a particle source term weighed onto the flow at grid points within the domain. The particle variables are then calculated using a high order ENO interpolation with a spline function of the same order of accuracy as the flow solver to avoid excessive computational errors.

7.3 Results and Discussion

We perform simulations based on the fifth order WENO-Z based PSIC algorithm for flow evolution when a shock runs through a cloud of particles. To summarize the algorithm, we approximate the system of hyperbolic Euler equations (7.1) in the Eulerian frame,

$$\frac{d\mathbf{Q}}{dt} = -\nabla \cdot \vec{F}(\mathbf{Q}) + \mathbf{S}(\mathbf{x_p} - \vec{x}), \tag{7.11}$$

on each grid point. A N_p number of particles are individually traced in the Lagrangian frame with

$$\begin{aligned} \frac{d\mathbf{x_p}}{dt} &= \mathbf{v_p}, && (7.12) \\ \frac{d\mathbf{v_p}}{dt} &= f_1 \left(\frac{\mathbf{v_f} - \mathbf{v_p}}{\tau_p} \right) - \frac{1}{\rho_p} \boldsymbol{\nabla} P|_f, && (7.13) \\ \frac{dT_p}{dt} &= \frac{1}{3} \frac{Nu}{Pr} \left(\frac{T_f - T_p}{\tau_p} \right). && (7.14) \end{aligned}$$

Interpolation determines $\mathbf{v_f}$, and T_f, while weighing determines $\mathbf{S}(\mathbf{x_p} - \vec{x})$.

We employ the third order Total Variation Diminishing Runge-Kutta scheme (RK-TVD):

$$\begin{aligned}
\vec{U}^1 &= \vec{U}^n + \Delta t L(\vec{U}^n) \\
\vec{U}^2 &= \frac{1}{4}\left(3\vec{U}^n + \vec{U}^1 + \Delta t L(\vec{U}^1)\right), \\
\vec{U}^{n+1} &= \frac{1}{3}\left(\vec{U}^n + 2\vec{U}^2 + 2\Delta t L(\vec{U}^2)\right)
\end{aligned} \tag{7.15}$$

where L is the spatial operator as in the right side of (7.11) or the right hand side of (7.14). The CFL number is set to 0.4.

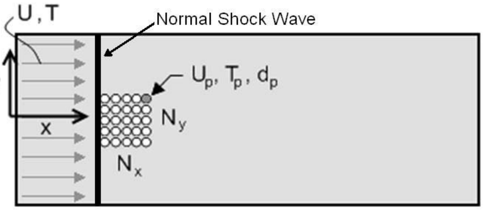


Figure 7.1: Initial setup of the shock-particle laden flow.

We initialize a right running Mach three shock $M_s = 3$ at $x_s = 0.175$ in a rectangular domain $[0,3] \times [-0.611, 0.611]$. Inflow and outflow boundary conditions are specified, respectively, in the x direction. Periodic boundary conditions are imposed in the y direction. A uniformly distributed bronze particle cloud is seeded in a rectangular shape, $[0.175, 0.352] \times [-0.044, 0.044]$, with zero initial velocity. The volume concentration of the particles in the cloud is 4%. The particle response time and density are, $\tau_p = 51.69$ and $\rho_p = 7.42 \times 10^4$, respectively. We take the Reynolds number needed to compute the particle traces according to the experiment at $Re_f = 3.387 \times 10^7$. In our discussion of the results below, we shall refer to the collection of the particles as *shape*.

In the following computations, the number of grid points used to solve the Euler equation in the Eulerian frame is 1500 × 500 in the x and y directions respectively. Clouds of 3.9×10^9 real particles were approximated using 40,000 ± 100 computational particles depending upon the cloud shape. The dimensions of the different clouds were determined based on the shape and aspect ratio desired and particles were seeded accordingly to provide for an even distribution. As an example, the square shape shown in Fig. 7.1 was arranged with 200 × 200 particles.

We briefly summarize the effect of initial cloud shape that was studied in Jacobs et. al. [65] to set the stage for the analysis of the effect of aspect ratio and rotation of the cloud. In Jacobs et. al. [65], we compared rectangular, circular and triangular shaped clouds with the same particle volume concentrations. The clouds cover the same geometric area and are initialized so that the cloud's location is directly adjacent to the downstream side of the normal shock. At early times the normal shock wave propagates through the particle cloud, and a reflected bow shock forms upstream of the particle cloud (see the vorticity contours in Fig. 7.2). Whereas the bow shock development is comparable for the circular and rectangular cases (Figs. 7.2a and 7.2b), showing a strong detached bow shock, the more aerodynamic triangular case (Fig. 7.2c) induces a much weaker bow shock.

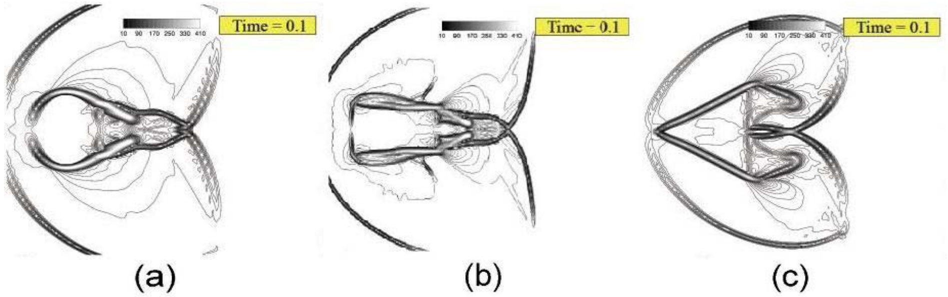


Figure 7.2: Vorticity contour of the accelerated gas flow behind a moving shock over (a) circular, (b) rectangular and (c) triangular shaped clouds at $t = 0.1$.

The gas flow separates at the sharp front corners of the rectangular shape and at the apex of the circular shape. The flow over the triangular shape separates only at the two rear corners. Figs. 7.3a, 7.3b and 7.3c show that the particle trajectories closely follow the separated shear layers initially and form distinct material lines. The accelerated flow stagnates at the front of the blunt rectangular and circular shapes compressing these clouds. The particles at the front end move downstream and increase the particle density within the cloud. The leading area of the triangle yields a much lesser compression and the average location of the particles within the cloud do not move to the right as far as the other two cases. For a more detailed discussion of the effects of shape on this particle-laden flow we refer the reader to Jacobs et. al. [65].

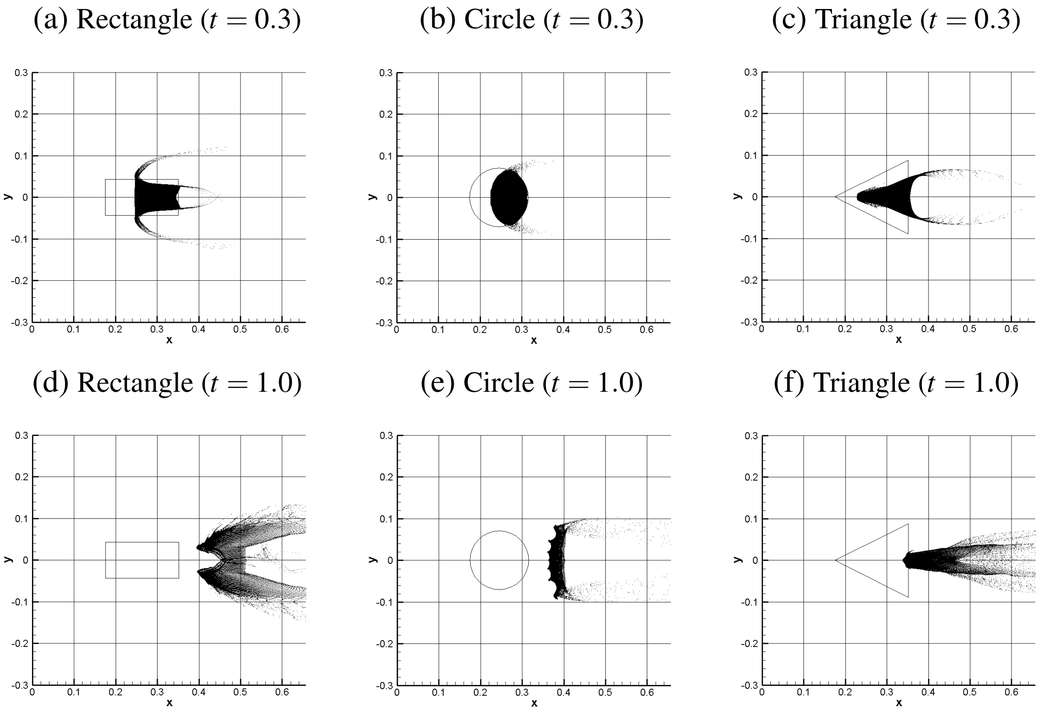

(a) Rectangle ($t = 0.3$)
(b) Circle ($t = 0.3$)
(c) Triangle ($t = 0.3$)
(d) Rectangle ($t = 1.0$)
(e) Circle ($t = 1.0$)
(f) Triangle ($t = 1.0$)

Figure 7.3: Dispersion pattern of 40K bronze particles in the accelerated gas flow behind a moving shock over (a, d) rectangular, (b, e) circular and (c, f) triangular cloud shapes at (top row) an early time, $t = 0.3$, and (bottom row) a late time, $t = 1.0$.

To quantify the motion and dispersion of the particles we determine the averaged and normalized root-mean-square global statistics on the particle cloud. The mean x-displacement of the particles is determined by comparing the particles' average location at a given time t, $\bar{x}(t)$, to their average location at the initial time $t = 0$, $\bar{x}(0)$:

$$x_{disp}(t) = \bar{x}(t) - \bar{x}(0), \quad \text{where} \quad \bar{x} = \frac{1}{N_p}\sum_{i=1}^{N_p} x_i(t). \tag{7.16}$$

is the averaged x-location of N_p number of particles.

Similarly, we quantify the dispersion in the cross-stream y-direction through the normalized root-mean-square deviation from the average y-location of the cloud at time t, $\bar{y}(t)$, as

$$y_{disp} = \frac{y_{rms}(t) - y_{rms}(0)}{y_{rms}(0)}, \quad \text{where} \quad y_{rms}(t) = \sqrt{\frac{1}{N_p}\sum_{i=1}^{N_p} (y_i(t))^2}. \tag{7.17}$$

is the root-mean-square particle y-location at time t.

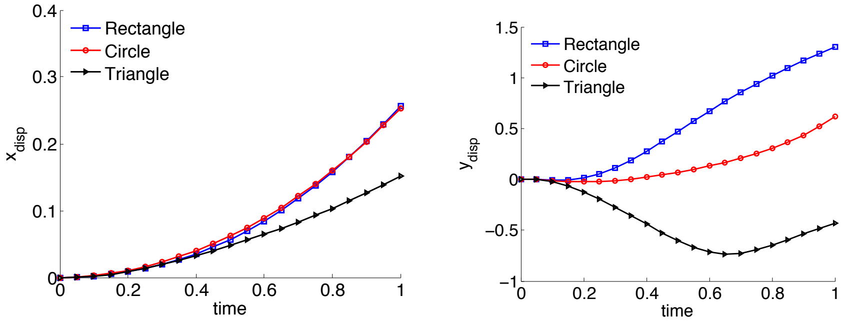


Figure 7.4: (a) Temporal history of the averaged x-location of the circular, rectangular and triangular shaped clouds, $x_{disp}(t)$. (b) Temporal history of the normalized root-mean-square y-location deviation of the circular, rectangular and triangular shaped clouds, $y_{disp}(t)$.

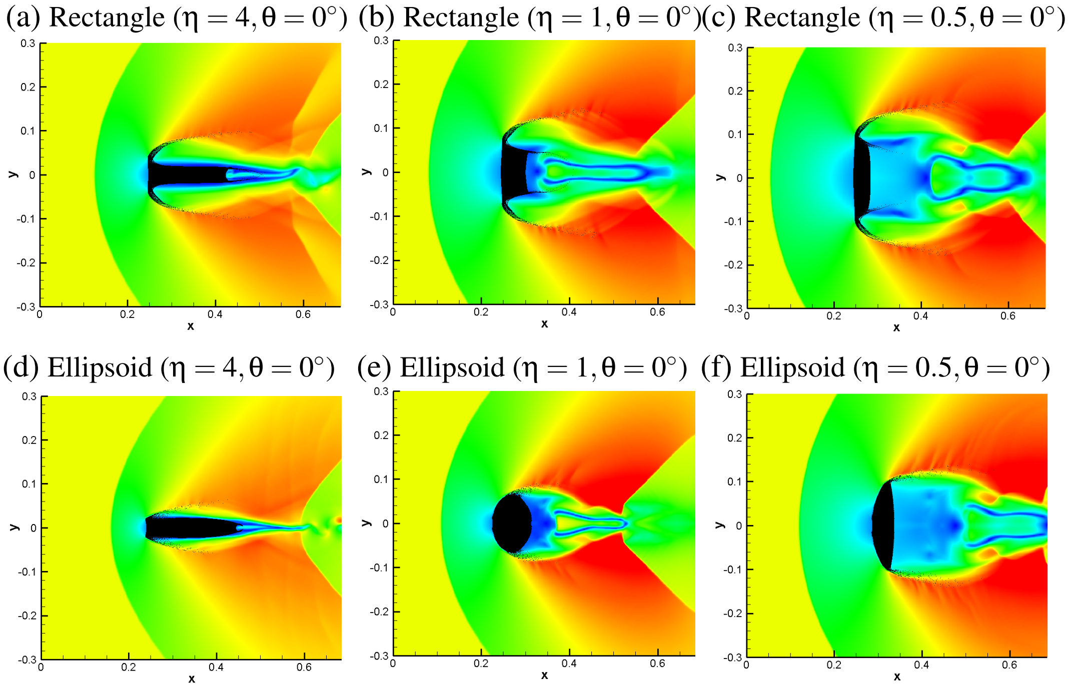
In Fig. 7.4a, the x_{disp} statistic shows that the circular and the rectangular shaped clouds have comparable x-direction convections, while the triangle's motion is 40% less at time $t = 1.0$. This difference is attributed to the smaller force between the gas and particle phase in the x-direction induced by the more aerodynamic triangle. A smaller force translates to a lesser acceleration of the shape in x-direction according to Newton's second law.

Fig. 7.4b, which plots the temporal evolution of y_{disp}, shows that the particle transport along separated shear layers at the front corners of the rectangle induces the greatest dispersion of particles in the cross-stream as compared to the other shapes. The initial dispersion of the cloud leads to greater movement in the cross-stream direction. However, at around $t = 0.4$, for the rectangle, the dispersion curve moves from concave up to concave down indicating a leveling off of the dispersion over time. As the particles spread away from the bulk cloud shape, micro-scale flow-particle interactions become the dominant force and the particles convect purely downstream at very late times. The relative cross-stream motion of particles in the triangular shape in time is greater than those of the circle and the rectangle. However, the compression of the triangular shape in time, as shown in Figs. 7.3c and 7.3f, causes y_{disp} to be negative and hence the dispersion of the cloud is significantly reduced for the triangle as compared to the other shapes.

7.3.1 Effect of Aspect Ratio η

Changing the cloud's aspect ratio, η, defined as the ratio of the length of the initial cloud shape in the x-direction to the width of the initial cloud shape in the y-direction, does not change the qualitative behavior of the particle-laden flow dynamics for a given type of shape at early (< 0.4) and late ($t > 0.7$) times .

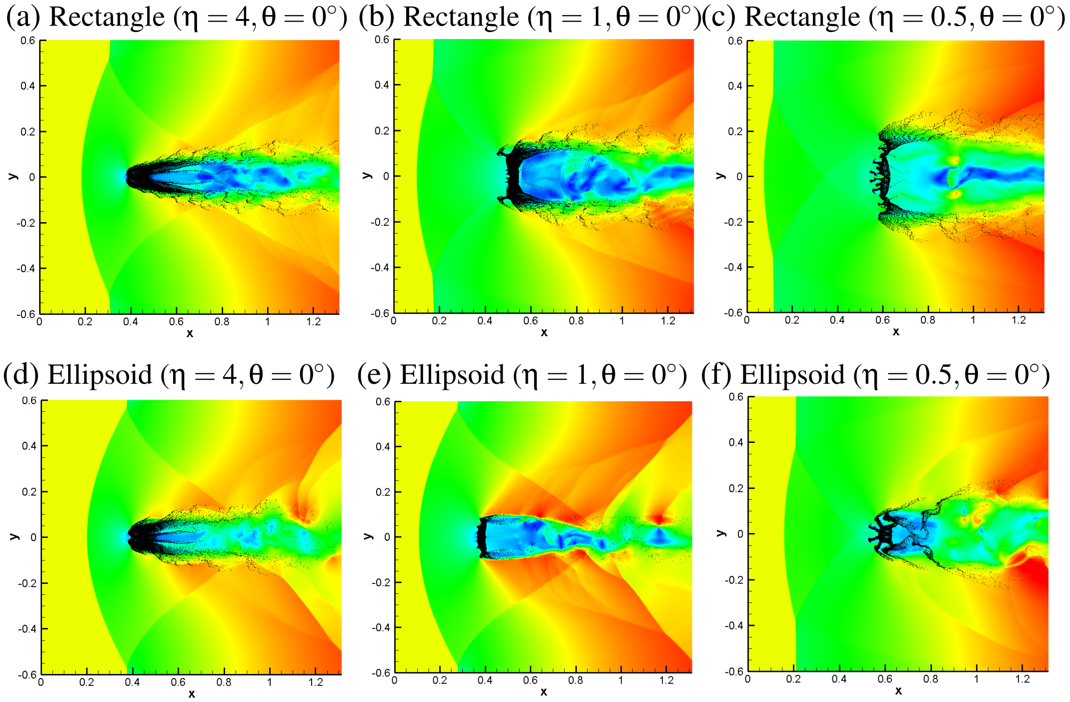
The snapshots of velocity magnitude contours and particle dispersion patterns in Fig. 7.5 at t=0.3 and Fig. 7.6 at t=1.0 show that the particles in the rectangular shaped cloud are transported along the separated shear layers at the front and rear corners of the shape at early times. At later times, the arms which initially formed at the front corners shield the flow extending the width of the wake. The reduced velocity in the wake prevents further transport of the particles into the arms at the rear corners of the shape.



(a) Rectangle ($\eta = 4, \theta = 0°$) (b) Rectangle ($\eta = 1, \theta = 0°$)(c) Rectangle ($\eta = 0.5, \theta = 0°$)

(d) Ellipsoid ($\eta = 4, \theta = 0°$) (e) Ellipsoid ($\eta = 1, \theta = 0°$) (f) Ellipsoid ($\eta = 0.5, \theta = 0°$)

Figure 7.5: Velocity magnitude contours and particle dispersion patterns at an early time $t = 0.3$ for (a) a rectangle shaped cloud with aspect ratios $\eta = 4$, (b) $\eta = 1$, (c) $\eta = 0.5$ and (d) an ellipsoid shaped cloud with aspect ratios $\eta = 4$, (e) $\eta = 1$, (f) $\eta = 0.5$, and an angle of attack $\theta = 0°$.



(a) Rectangle ($\eta = 4, \theta = 0°$) (b) Rectangle ($\eta = 1, \theta = 0°$)(c) Rectangle ($\eta = 0.5, \theta = 0°$)

(d) Ellipsoid ($\eta = 4, \theta = 0°$) (e) Ellipsoid ($\eta = 1, \theta = 0°$) (f) Ellipsoid ($\eta = 0.5, \theta = 0°$)

Figure 7.6: Same as caption 7.5 except at a late time $t = 1.0$.

The flow separation location changes with time along the smooth surface of the ellipsoidal shapes. As with the rectangular shaped cloud, particles are transported along the separated shear layers. Since the separation location is moving in time as opposed to the fixed separation at the front corners of the rectangular cloud shapes, the particle arms are less defined for the ellipsoidal shapes at early times.

Both circular and rectangular shaped clouds with $\eta > 1$ are compressed relatively more in the x-direction in time than clouds with $\eta < 1$. Clouds with $\eta < 1$ experience more compression along the longer sides in y-direction as compared to clouds with $\eta > 1$ that have relatively small lengths along the sides.

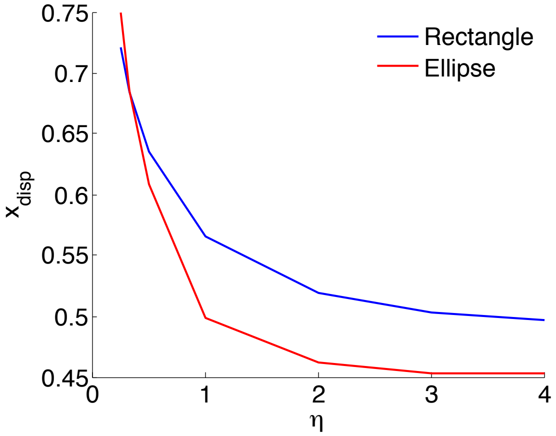


Figure 7.7: $x_{disp}(t)$ versus aspect ratio η, for initially rectangular and ellipsoidal particle shaped clouds at $t = 1.0$.

With an increase of the cloud's aspect ratio, the cloud's geometry is more slender and the cloud blocks the flow less. For higher aspect ratio η, the wake zones and low velocity stagnation areas in front of the shape decrease due to this smaller obstruction. The slender geometry induces a smaller forcing between the gas and particle phase, which results in the reduced convection of the cloud in x-direction with increased η, as shown in Fig. 7.7, for both the ellipsoidal and rectangular shaped clouds at the late time $t = 1$. The reduced convection of the ellipsoids as compared to the rectangles, shows that the ellipsoid induces a smaller mutual forcing and is more aerodynamic.

7.3.2 Effect of Rotation θ

To rotate the shape, we change the coordinates of particles in the rectangle and the ellipsoid shaped clouds as follows

$$[x_p, y_p]_{new} = \mathbf{R}[x_p, y_p]_{old}, \tag{7.18}$$

where

$$\mathbf{R} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}, \tag{7.19}$$

is the rotation matrix with a given angle of rotation θ. The leading point of the rotated particle cloud is initially positioned directly behind the right moving shock at $x = 0.175$ ensuring that the shock wave moves through the cloud at the same time for all rotation angles.

Snapshots of the velocity magnitudes and particle dispersion patterns at early times (Figs. 7.8a and 7.8d) show that the particle-laden flow developments are qualitatively similar at moderate angles of attack as compared to the unrotated cases. Particles separate into arms along the front corner of the rectangle and shield the downstream portion of the cloud more in the rectangular shaped cloud than the ellipsoidal one. However, the flow strength, flow separation and amount of arm shielding changes due to the rotation as the upstream corner of the cloud becomes a focal point for the flow stagnation. A larger, more dense cloud region forms from this focus and can be seen at later times as a leading arm protruding from the rest of the cloud most evidently in Figs. 7.9a but also in Fig. 7.9d. Because of the rotation, the top surface faces the oncoming flow directly, shielding the bottom downstream portion of the cloud at early times causing a larger arm to form on the downstream portion of the cloud at later times.

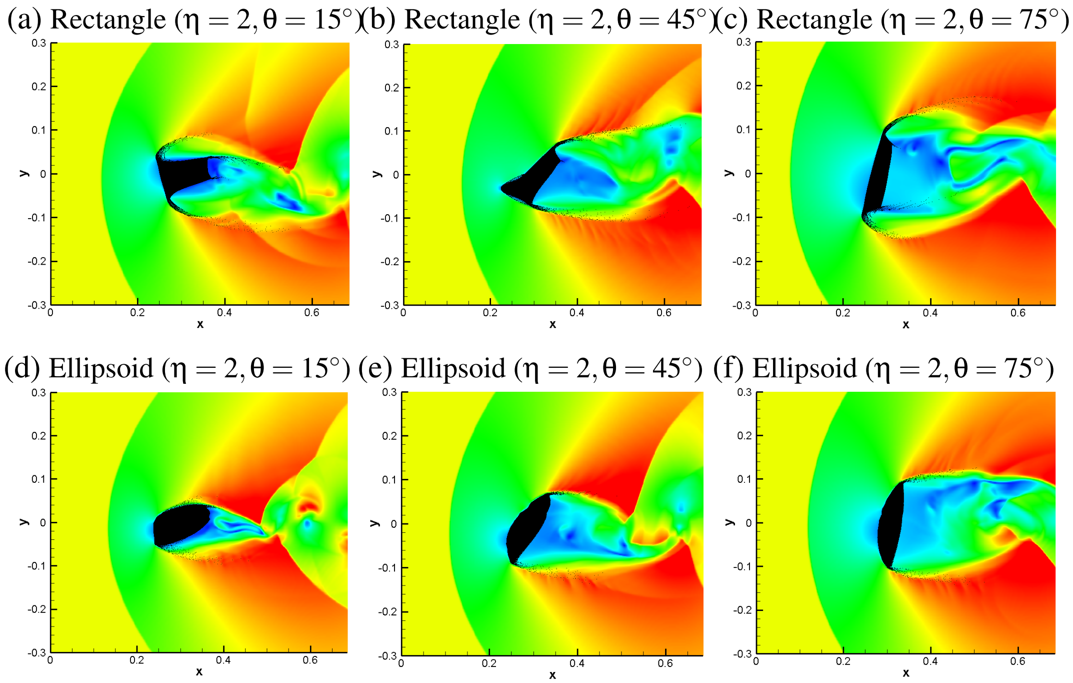

(a) Rectangle ($\eta = 2, \theta = 15^\circ$)
(b) Rectangle ($\eta = 2, \theta = 45^\circ$)
(c) Rectangle ($\eta = 2, \theta = 75^\circ$)
(d) Ellipsoid ($\eta = 2, \theta = 15^\circ$)
(e) Ellipsoid ($\eta = 2, \theta = 45^\circ$)
(f) Ellipsoid ($\eta = 2, \theta = 75^\circ$)

Figure 7.8: Velocity magnitude contours and particle dispersion patterns at an early time $t = 0.3$ for a rectangle shaped cloud with angles of attack (a) $\theta = 15^\circ$, (b) $\theta = 45^\circ$, (c) $\theta = 75^\circ$ and an ellipsoid shaped cloud with angles of attack (d) $\theta = 15^\circ$, (e) $\theta = 45^\circ$, (f) $\theta = 75^\circ$, and a fixed aspect raio $\eta = 2$.

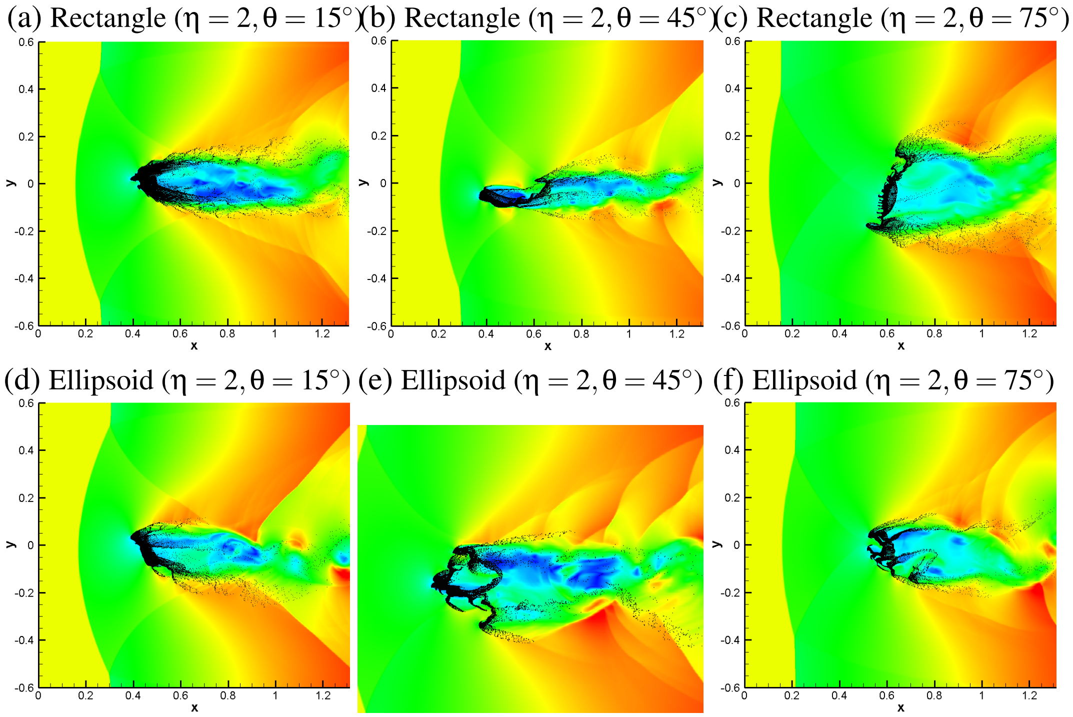

(a) Rectangle ($\eta = 2, \theta = 15^\circ$)
(b) Rectangle ($\eta = 2, \theta = 45^\circ$)
(c) Rectangle ($\eta = 2, \theta = 75^\circ$)
(d) Ellipsoid ($\eta = 2, \theta = 15^\circ$)
(e) Ellipsoid ($\eta = 2, \theta = 45^\circ$)
(f) Ellipsoid ($\eta = 2, \theta = 75^\circ$)

Figure 7.9: Same as caption 7.8 except at a late time $t = 1.0$.

As the angle of rotation θ further increases, the particle-laden flow development dramatically changes both qualitatively and quantitatively. At medium angles of rotation the flow development of the initially rectangular shaped cloud shows similar features to the flow development of the initially triangular particle cloud (Figs. 7.8b and 7.10a). One of the front corners of the shape now protrudes into the flow and the flow does not separate from this corner. Since the flow is attached along the sides, particles no longer leave from the shape at this corner. At early times most particles are transported out of the shape at the two corners that are downstream of the leading corner of the rectangle comparable to particle dynamics of the triangular cloud. At later times the rectangular shaped cloud has been compressed towards the symmetry line (Fig. 7.9b), also comparable to the triangle case (Fig. 7.10c). At early times, the ellipsoidal case is very similar to the rectangular case at medium angle of rotation (Fig. 7.8e), but the smoother geometry leads to material arms and flow separations that are less distinct. One additional feature of the ellipsoid shaped cloud is the cavernous region which forms from the flow separation point near the upper half of the cloud. In Fig. 7.8e, a small divot can be seen which, at later times, grows until it bifurcates the cloud. This split lasts only briefly before the spanwise dispersion of the two new clouds overlap.

We note that the cases of a rectangle shaped cloud and an ellipsoid shaped cloud with $\eta > 1$ and large angle of rotation (Figs. 7.8c and 7.8f as well as 7.9c and 7.9f) are geometrically the same as a shape with $1/\eta$ and a small angle of rotation. As discussed previously, the particle-laden flow developments of shapes with small rotation angle are comparable to the flow developments of the same shape with zero angle of rotation.

To underscore that the flow dynamics of a rectangle shaped cloud at medium angles are comparable to that of a triangle, we place a square ($\eta = 1$) under angle of rotation $\theta = 45^\circ$. The front half of the diamond shape is now geometrically exactly a triangle. From the snapshots at early times (Fig. 7.10a and b) and late times (Fig. 7.10c and d), we can see that the dynamics of this front half are indeed the same as that of the triangle described above. The trailing half of the diamond does not move significantly and since this half is shielded from the oncoming flow it does not significantly affect the gas flow at early times. At later times, the wider trailing half of the diamond is exposed to the oncoming flow yielding a slightly wider cloud and wake as compared to the triangle case.

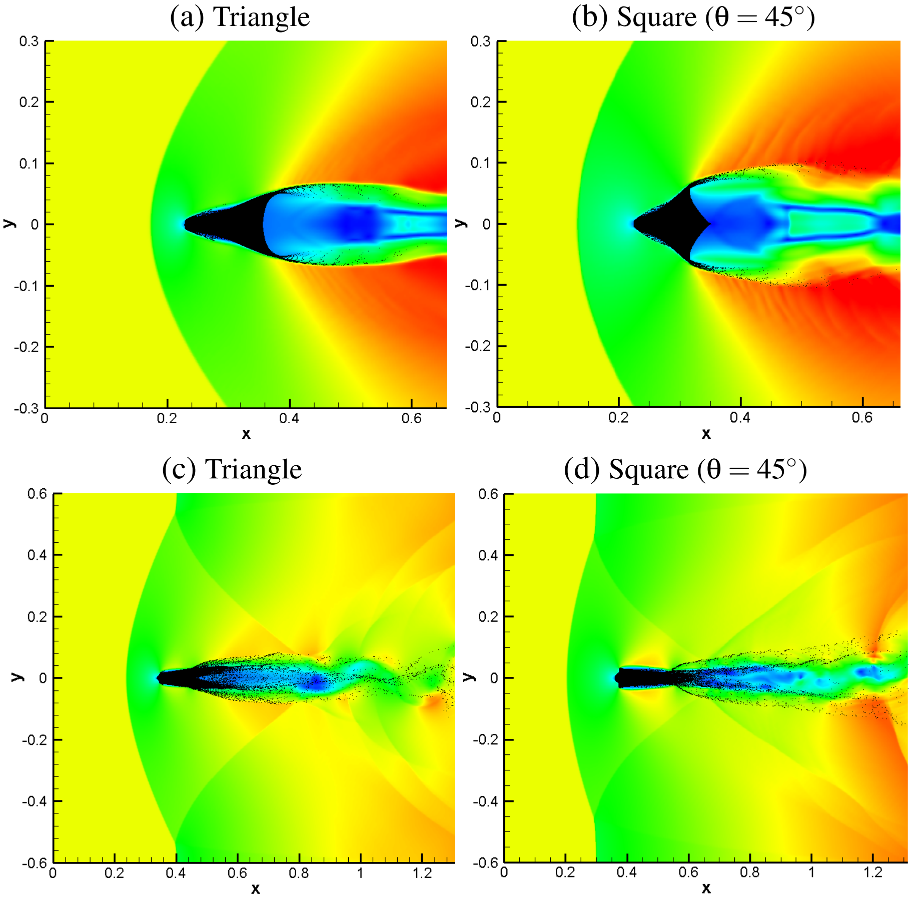


Figure 7.10: Velocity magnitude contours and particle dispersion patterns for a triangle shaped cloud and a square shaped cloud rotated $\theta = 45°$, at t=0.3 (top row) and $t = 1.0$ (bottom row).

The square shaped cloud is convected further downstream in x-direction at zero angle of rotation $\theta = 0°$ than at $\theta = 45°$ of rotation (when it behaves like a triangle), since the force between gas and particle phase is larger for the blunt square shape at zero rotation as compared to the more aerodynamic rotated shape. The averaged x-location of the cloud, x_{disp}, is hence larger at small angle of rotation as compared to medium angles of rotation (Fig. 7.11a). Because of the symmetry of the square, the x_{disp} curve is symmetric versus angle of rotation θ.

Noting again that a shape with $\eta > 1$ at a ninety degree rotation ($\theta = 90°$) is geometrically identical to that same shape with one over η ($\frac{1}{\eta}$) at $\theta = 0°$, it follows that x_{disp} is smaller for $\eta > 1$ at $\theta = 0°$ as compared to that shape at $\theta = 90°$, similar to the trends of x_{disp} versus η in Fig. 7.7. We observe that curves for different aspect ratio in Fig. 7.11a cross through a single point, indicating a correlation between rectangular shaped clouds at a rotation angle $\theta \approx 30°$.

At medium angles of rotation, rectangles behave like triangles and hence they show a minimum y_{disp} for all η (Fig. 7.11b). The minimum is naturally at $\theta = 45°$ for the symmetric square and the compression of the cloud reduces with increasing η.

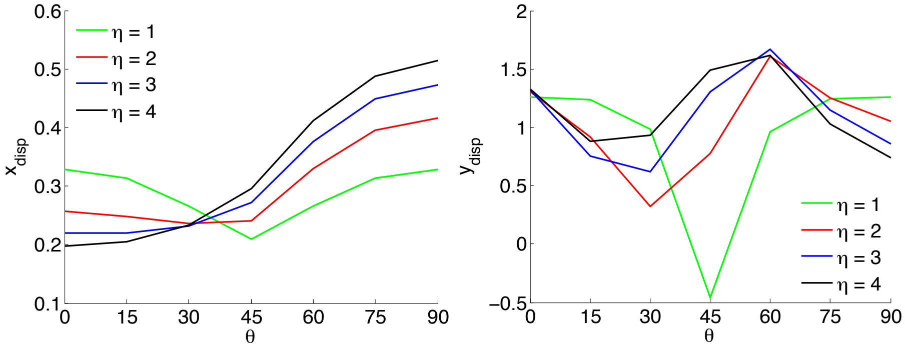


Figure 7.11: (a) $x_{disp}(t)$ and (b) $y_{disp}(t)$ versus angle of rotation, θ, for initially rectangular particle clouds with aspect ratio $\eta = 1, 2, 3, 4$.

A comparison of x_{disp} for the ellipsoid shaped cloud in Fig. 7.12a with x_{disp} for the rectangular shaped cloud in Fig. 7.11a confirms the similarity between the two shapes. The trends with rotation and aspect ratio are the same and Fig. 7.12a shows the same crossing at a single point of the curves with different aspect ratio, only at a slightly smaller rotation angle of $\theta = 15°$. Note that since a circle with $\eta = 1$ does not change

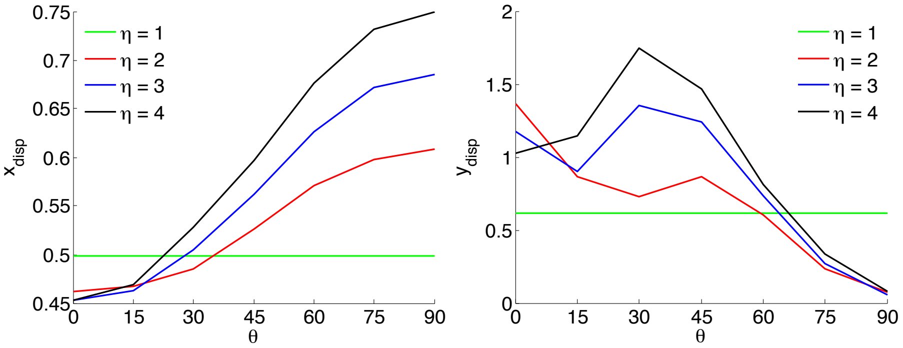


Figure 7.12: (a) $x_{disp}(t)$ and (b) $y_{disp}(t)$ versus angle of rotation θ, for initially ellipsoidal particle clouds with aspect ratio $\eta = 1,2,3,4$.

geometrically with rotation, x_{disp} of an ellipsoid shaped cloud is not affected by rotation at $\eta = 1$. For shapes with $\eta > 1$, instabilities play a major role in the dispersion of the particle clouds at late times, which causes a disparity in the initial spanwise dispersion for an angle of rotation of zero. Early flow dynamics dominate the y_{disp} trends for clouds rotated past $\theta = 45°$, which are geometrically similar to clouds with $\eta < 1$, with greater spanwise dispersion as the cloud becomes longer in the y-direction.

7.4 Conclusions and Future Developments

A numerical study of the effect of initial shape, aspect ratio and rotation of a cloud of particles on the particle-laden flow development in the accelerated flow behind a moving shock is conducted using a high-order/high-resolution Eulerian-Lagrangian method.

A change of shape dramatically changes the particle-laden flow development. In the case of an initially rectangular shaped cloud, particles separate mostly along shear layers into distinct arms that emanate from the front corners. These arms shield the incoming flow from the rear part of the cloud. The flow separation location changes in time along the smooth circular shaped cloud surface, and hence the particle arms are less distinct as compared to the initially rectangular shaped cloud. The flow remains attached along the sides of the more aerodynamically shaped triangular cloud and particles are transported along shear layers that emanate from the rear corners.

The averaged triangular shaped cloud location is convected 40% less downstream as compared to the blunt rectangular and the circular shaped clouds at time $t = 1.0$. This is attributed to a reduced forcing between the gas and particle phase and hence a lesser particle acceleration for the aerodynamic triangle shape. The attached gas flow along the sides of the triangular shape compresses the particles towards the symmetry line, leading to a reduction of the root-mean-square cross-stream location of the cloud. The particles that are pulled out of the rectangular and circular shapes by separated shear layers move away from the symmetry line and increase the root-mean-square cross-stream cloud location.

A change of aspect ratio does not change the particle-laden flow qualitatively. However, a slender, aerodynamic high aspect ratio shape is convected less in the streamwise direction as compared to low aspect ratio shapes. Low aspect ratio shapes are relatively more compressed in the streamwise direction, whereas high aspect ratio shapes are relatively more compressed in the cross-stream direction.

Except for small wake and dispersion asymmetries, at low angles of rotation, qualitative flow developments are similar to the case of zero angle of rotation. At medium angles of rotation, the flow characteristics of the cases with rectangular and ellipsoidal shaped clouds are comparable to the flow features of the initially triangular shaped cloud case.

The current investigation is part of a larger effort to validate our high-order/high-resolution Eulerian-Lagrangian solver against shock-tube experiments detailing the dispersion of a cloud of particles in the flow behind a moving shock. Results are databased, while the particle-laden shocked flow developments are elucidated and documented. We have extended our codes to three-dimensions recently and are conducting investigations into the three-dimensional effects on these configurations.

Sections 7.1 to 7.4, in full, are a reprint of the material as it appears in Dispersion of a cloud of particles by a moving shock: Effects of the shape, angle of rotation, and aspect ratio. Davis, Sean; Dittman, Thomas; Jacobs, Gustaaf; Don, Wai Sun, Springer Publishing, 2013. The dissertation/thesis author was the primary investigator and author of this material.

Chapter 8

Shear Instabilities

Sections 8.1 to 8.7 are taken from a manuscript published in Physics of Fluids in 2015 [125].

8.1 Introduction

The mixing of two layers with different velocity and particle loading profiles is commonly observed in natural and industrial processes such as the shear layer formed by the confluence of two rivers or the injection of coal particles in a coal-fired power plant. The mixing efficiency of these flow systems depends on intrinsic hydrodynamic instabilities. These instabilities not only determine the size of the mixing zone, but also give rise to large- and small-scale flow structures, which homogenize the mass, momentum and energy balance.

In linear stability analysis of particle-laden flows one typically studies the growth rate of a linear disturbance governed by a reduced, linearized model of the full non-linear Eulerian-Eulerian (EE) model. In these EE models, both the carrier phase and the particle phase are modeled in an Eulerian frame. Like the carrier phase, the particle phase is governed by conservation laws.

Whereas the modeling assumptions of first-principle Eulerian carrier-phase models, such as the Navier-Stokes equations, typically do not give rise to significant inaccuracy within modeling bounds, Eulerian particle models suffer from a “closure problem”. While a range of Eulerian particle phase models exist that address the closure problem [63, 69, 121, 150, 109, 127, 128], it remains an open research area.

Saffman [118] presented the first analytical stability analysis of a viscous, incompressible mixing layer with uniform loading in 1961. In the nineties and the early 2000s, a range of studies [29, 144, 142, 103] appeared on the temporal linear stability analysis of a mixing layer, which covered viscous [144, 103] and inviscid [29, 142] particle-laden mixing layers.

In all of these studies, a non-linear first-moment (averaged) particle model that governs averaged particle phase variables is reduced to either one Rayleigh equation or the Orr-Sommerfeld equation for inviscid and viscous flow, respectively. This equation governs the growth of stable and unstable modes. Further, it is assumed in all such studies that the mean flow is a particle-free single-phase mixing layer. The particle and gas phases are initially in dynamic equilibrium before being slightly perturbed [161]. As compared to a single phase analysis, the particle-laden flow stability analysis involves at least two additional parameters including the bulk mass loading and the particle Stokes number (St).

The primary investigations on linear stability analysis of viscous, particle-laden mixing layers [118, 144, 103] focused on incompressible flow and uniform bulk mass loading. In particular, Saffman [118] found that particles with fast response to the fluid (i.e. small St) destabilize a viscous flow, because an increase in the bulk average mixture density of the gas leads to an increase in the effective Reynolds number. The flow is therefore less stable because the effective critical Reynolds number decreases (velocity, density and reference length kept constant) with respect to the single phase flow. Alternatively, particles with a large response time dissipate disturbances in the gas and stabilize the flow.

Through numerical experiments, Tong and Wang [144] confirmed that the addition of small-inertia particles can destabilize the flow and increase the effective interphase momentum transfer. Particles with a relative medium- to large-inertia dissipate flow instabilities, with the most added dissipation at St = 1. The qualitatively different behavior of small-, medium- and large-inertia particles is shown to affect the direction of interphase energy transfer.

Prior to this study, as compared to the single-phase configuration, growth rate computations in the temporal stability analysis of inviscid particle-laden mixing layers [142, 155, 156, 142] have not shown a destabilizing effect induced by the presence of the particles. Thevand and Daniel [142] studied the temporal development of inviscid particle-laden mixing layers with a given uniform bulk mass loading and different disturbance wave numbers. As opposed to the results obtained by Tong and Wang [144] for viscous mixing layers, the maximum growth rate inevitably decreases when particles are present in the flow. Consistent with the results of Tong and Wang, the stabilization effect depends greatly on the particle inertia and is a maximum for particles with Stokes number around unity. For inviscid, uniformly laden shear layers, the lowest growth rate attenuation is computed for fast-responding particles, i.e. low Stokes number.

In this paper, we analyze the effect of non-uniform bulk mass loading on the linear, temporal development of an incompressible, inviscid particle-laden shear flow when the mass loading interface is much thinner than, and displaced with respect to, the velocity interface. For the first time in stability analysis of inviscid flows, we show that particles with small St may destabilize the temporal mixing layer development with respect to pure-gas flow. The destabilizing effect is triggered by the non-uniformity of the bulk mass loading.

Theoretical results predicted by the proposed Rayleigh equation are analytically derived and then verified numerically. The physical explanation of the destabilizing effect induced by quick-responding particles is acquired by applying the interface energy transfer budget proposed by Tong and Wang [144] to this particular case. We show that an energy transfer that flows globally from the particle phase to the gas phase causes the destabilization. The verification of the growth rate computations is achieved through simulations based on a fully non-linear Eulerian-Lagrangian (EL) model [67]. In the EL model, individual particles are modeled as points and traced along their Lagrangian path. Both the energy budget and growth rates are in very good comparison with theoretical results of Thevand and Daniel's approach [142] and we then extend beyond their analysis to the case of non-uniform bulk mass loading.

We note that, in the nineties, the investigation of the temporal development of inviscid, non-uniformly laden, particle-laden shear flows was tackled by Wen and Evans [155, 156]. In their first article [155], the amplification rates for different disturbance wavenumbers are computed for a "frozen" particle phase configuration, i.e. a simplified situation in which the particles do not respond to the fluctuation of the gas phase but rather the particle response time is much greater than the time scale of the perturbations. They found that two opposite traveling unstable modes govern the particle-laden shear layer instability: the first unstable mode (Kelvin-Helmholtz mode) has the largest growth rate and the second unstable mode is analogous to the Holmboe instability in a density-stratified mixing layer flow [61, 81]. In this paper, we confirm this behavior for particles with high Stokes numbers.

When Wen and Evans attempted to extend their analysis to the broader scenario of finite particle inertia in a follow-up article [156], the computed growth rates showed significant lack of consistency with the results presented by Thevand and Daniel [142] for a uniformly laden inviscid, nearly incompressible shear layer. Quite surprisingly, the growth rate of the dominant unstable mode in Wen and Evans' study[156] weakens with decreasing St.

Especially for the case of slow-responding particles, we believe that Wen and Evans' model [155] suffers from the omission of the convective terms in the momentum equation for the particle phase. To correct this, our model starts with the stochastic Eulerian model proposed by Pandya and Mashayek [109] and Shotorban [127] for large-eddy simulation and direct numerical simulation, respectively. This EE model is derived based on a combination of the Liouville theorem applied to the system of dynamical particle equations and a method of moments. This approach does include convective terms and, because in linear stability analysis the evolution of small perturbations is determined, we address the "closure" of second order moments in the momentum equation for the particle phase with a first order approximation (zeroing the second order cumulant). Due to the difference in the proposed Rayleigh equations, the boundary conditions to be imposed at the bulk mass loading interface also formally differ from the ones proposed by Wen and Evans [155].

In Section 8.2 we discuss the particle-laden shear layer flow to be studied. Following, in Section 8.3 the first-order Eulerian-Eulerian model is derived. Then, the Rayleigh equation and the numerical solution of the evolution of a linear particle-laden gas perturbation are presented in Section 8.4. We then discuss the stability behavior of non-uniform shear layers in Section 8.5. In Section 8.6 we verify the results of the stability analysis with EL simulations. Conclusions are reserved for the last Section, 8.7.

8.2 Particle-laden shear layers with non-uniformly laden bulk mass Loading

We consider a two-dimensional (2D) shear layer as shown in 8.1 with a normalized velocity profile given by

$$U(z) = \frac{U_1 + U_2}{2} + \frac{U_2 - U_1}{2}\tanh(2z - \varepsilon) \ , \tag{8.1}$$

where ε is a measure of the relative offset between the x-direction and the location of maximum shear. The velocity profile is normalized with respect to the reference velocity $\Delta U^* = U_2^* - U_1^*$, where the asterisk superscript denotes a dimensional variable. The coordinate system is selected such that the streamwise flow is in the x-direction. Following Yang et. al. [161], the velocity profile for the carrier-phase and particle phase are the same, i.e. the slip velocity between the phases is zero. The reference length scale is taken as the shear layer's vorticity thickness [19], δ_ω^*, as defined by

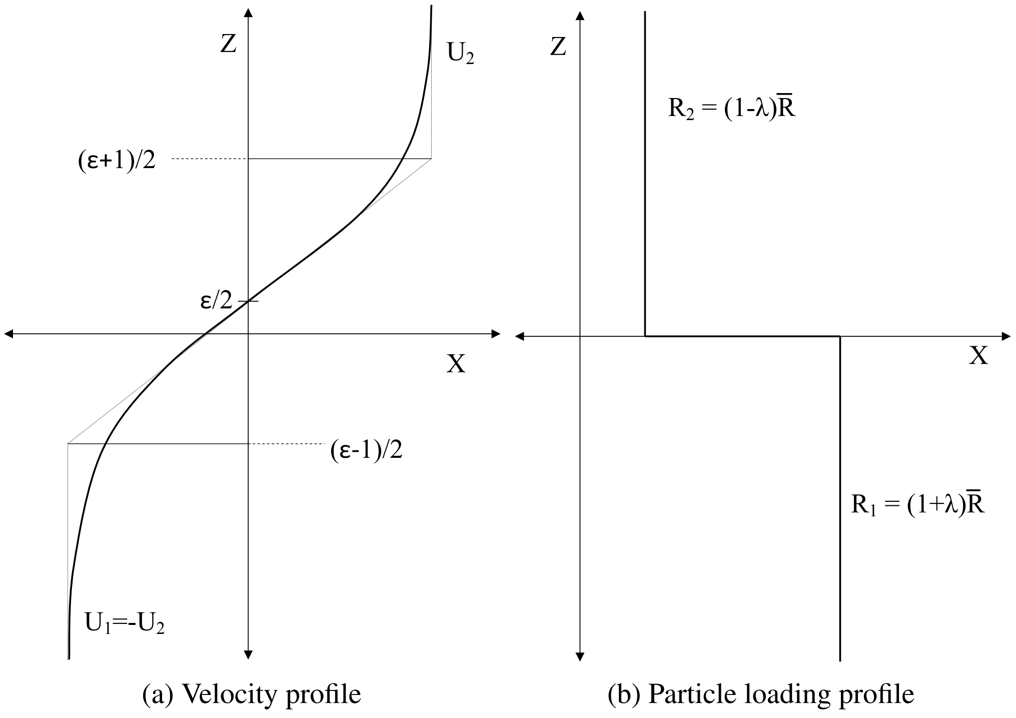
$$\delta_\omega^* = \frac{\Delta U^*}{\left(\frac{dU^*}{dz^*}\right)_{max}} \ , \tag{8.2}$$

where the denominator represents the maximum gradient in the shear in the z-direction.

Following Wen and Evans [155], the particle phase mass loading is discontinuous (8.1b) and is initialized with a particle to gas mass loading ratio, R, defined by:

$$R(z) = \frac{m_p^* n_p^*(z)}{\rho^*} = \begin{cases} R_1 = (1+\lambda)\overline{R} \ , & z \leq 0 \ , \\ R_2 = (1-\lambda)\overline{R} \ , & z > 0 \ , \end{cases} \tag{8.3}$$

where ρ^* is the fluid density, $-1 \leq \lambda \leq 1$ is the relative offset parameter of the particle loading profile with respect to the average value $\overline{R}$; m_p^* and n_p^* are particle's mass and



(a) Velocity profile

(b) Particle loading profile

Figure 8.1: (a) Base (unperturbed) velocity profile U versus z and (b) particle mass loading R versus z, in the shear layer.

particle phase number density, respectively.

8.3 First-order Stochastic Eulerian-Eulerian model

8.3.1 Eulerian fluid phase

Following Yang *et al.* [161], we neglect viscous stresses in the carrier phase momentum equation. As reported by Thevand and Daniel [142], this simplification is physically justified by the inviscid nature of the Kelvin-Helmoltz instability and has been validated for single phase flows in the viscous stability calculations of Ragab and Wu [113] where results were shown to be independent of Reynolds number for Re$= \rho^*\Delta U^*\delta^*_\omega/\mu^* > 1000$. We retain, instead, the momentum exchanged with the dispersed phase because the typical particle Reynolds number, Re$_p = \rho^*|\boldsymbol{u}^* - \boldsymbol{V}^*|D^*/\mu^*$, is usually small for fine particles [155]. In the definition of Re_p, $\boldsymbol{u}^*$ is the fluid velocity and D^* and $\boldsymbol{V}^*$ are the particle diameter and Lagrangian velocity, respectively. The *non-dimensional*, 2D, incompressible flow equations in the Eulerian framework are given by

$$\nabla \cdot \boldsymbol{u} = 0 \,, \tag{8.4a}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \frac{1}{\rho} \nabla p = \boldsymbol{S} \,, \tag{8.4b}$$

where $\rho = \rho^*/\rho^* = 1$ is the fluid density and $p = 2p^*/\rho^* \Delta U^{*2}$ is the fluid pressure. The source term $\boldsymbol{S}$ on the right hand side accounts for the momentum transfer between the gas and particle phase. In the limit of small particle Reynolds number and particle Mach number [95], $\mathrm{Ma}_p = |\boldsymbol{u}^* - \boldsymbol{V}^*|/\sqrt{kp^*/\rho^*}$ (k being the fluid specific heat ratio), the source term for an Eulerian particle phase reference frame is

$$\boldsymbol{S} = \frac{1}{\mathrm{St}} \, \widetilde{r}(\widetilde{\boldsymbol{u}} - \boldsymbol{u}) \,, \tag{8.5}$$

where St=$\rho_p \mathrm{Re} D^2/18$ is the Stokes number[28], with $D = D^*/\delta^*_\omega$ and $\rho_p = \rho^*_p/\rho^*$ the non-dimensional particle diameter and density, respectively. In (8.5), $\widetilde{\boldsymbol{u}} = (\widetilde{u}, \widetilde{w})$ is the Eulerian particle phase velocity and $\widetilde{r} = m^*_p \widetilde{\phi}^*/\rho^*$ represents the Eulerian particle to gas mass loading ratio, where $\widetilde{\phi}^*$ is the Eulerian particle number density. These variables are rigorously defined in Section 8.3.3. For a complete derivation of the source term, see Pandya and Mashayek [109] and Shotorban [127].

8.3.2 Lagrangian particle phase

Finite sized particles are modeled as singular points according to the point particle model [28]. In the Lagrangian frame, the non-dimensional kinematic equation,

$$\frac{d\boldsymbol{X}}{dt} = \boldsymbol{V} \ , \tag{8.6}$$

governs the position vector of the particle, $\boldsymbol{X}$. In gas-particle flows with high particle-to-gas density ratios, low particle volume fractions, $\rho_p = \rho_p^*/\rho^* > O(10^3)$, and small particle Reynolds number and Mach number, the velocity vector $\boldsymbol{V}$ of an isolated non-rotating spherical particle in a uniform flow is governed by Stokes' law [28]

$$\frac{d\boldsymbol{V}}{dt} = \frac{1}{\mathrm{St}}\big(\boldsymbol{u}(\boldsymbol{X},t) - \boldsymbol{V}\big) \ . \tag{8.7}$$

8.3.3 Eulerian particle phase

Following Pandya and Mashayek [109], Shotorban [127], and Shotorban et. al. [128], we derive an Eulerian model for the particle phase from the Lagrangian model by using the Liouville equation in combination with a method of moments.

The fine-grained phase-space density [109], $W(\boldsymbol{x},\boldsymbol{v},t)$, defined by

$$W(\boldsymbol{x},\boldsymbol{v},t) = \delta(\boldsymbol{x}-\boldsymbol{X})\delta(\boldsymbol{v}-\boldsymbol{V}) \ , \tag{8.8}$$

is the probability of finding a particle in phase-space near $(\boldsymbol{x},\boldsymbol{v})$ at time t. More precisely, $W(\boldsymbol{x},\boldsymbol{v},t)$ is proportional to the fraction of particles in the ensemble having a phase-space coordinate within the volume defined by $\boldsymbol{x}$ to $\boldsymbol{x}+d\boldsymbol{x}$ and $\boldsymbol{v}$ to $\boldsymbol{v}+d\boldsymbol{v}$. The fine-grained phase-space density satisfies the Liouville equation,

$$\frac{\partial W}{\partial t} + \nabla_{\boldsymbol{x}} \cdot (\boldsymbol{v} W) - \nabla_{\boldsymbol{v}} \cdot \left[\frac{1}{\mathrm{St}} (\boldsymbol{v} - \boldsymbol{u}) W \right] = 0 \; , \tag{8.9}$$

where $\nabla_{\boldsymbol{x}} = \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2} \right)$, $\nabla_{\boldsymbol{v}} = \left(\frac{\partial}{\partial v_1}, \frac{\partial}{\partial v_2} \right)$ and $\boldsymbol{x}$ and $\boldsymbol{v}$ are the phase-space vectors corresponding to $\boldsymbol{X}$ and $\boldsymbol{V}$, respectively. With a filter on the size of the smallest scale of the fluid field, the fine-grained phase-space equation is coarse grained to

$$\frac{\partial \bar{W}}{\partial t} + \nabla_{\boldsymbol{x}} \cdot (\boldsymbol{v} \bar{W}) - \nabla_{\boldsymbol{v}} \cdot \left[\frac{1}{\mathrm{St}} (\boldsymbol{v} - \boldsymbol{u}) \bar{W} \right] = 0 \; , \tag{8.10}$$

where $\bar{W}$ is the coarse grained density function [127]. Moments of the coarse grained density function lead to the Eulerian number density, $\widetilde{\phi}$, and the velocity of the particle phase, $\widetilde{\boldsymbol{u}}$, given by

$$\widetilde{\phi}(\boldsymbol{x}, t) = \int \bar{W}(\boldsymbol{x}, \boldsymbol{v}, t) \; d\boldsymbol{v} \; , \tag{8.11a}$$

$$\widetilde{\boldsymbol{u}}(\boldsymbol{x}, t) = \frac{1}{\widetilde{\phi}} \int \boldsymbol{v} \bar{W} \; d\boldsymbol{v} \; , \tag{8.11b}$$

respectively.

By taking zero and first moments of (8.10) for the two-dimensional case, governing equations for the number density and particle velocity are derived as

$$\frac{\partial \widetilde{\phi}}{\partial t} + \nabla \cdot (\widetilde{\phi} \widetilde{\boldsymbol{u}}) = 0 \; , \tag{8.12a}$$

$$\frac{\partial (\widetilde{\phi} \widetilde{\boldsymbol{u}})}{\partial t} + \nabla (\widetilde{\phi} \widetilde{\boldsymbol{u}\boldsymbol{u}}) = \frac{1}{\mathrm{St}} \widetilde{\phi} (\boldsymbol{u} - \widetilde{\boldsymbol{u}}). \tag{8.12b}$$

Here, the convective terms are included in the particle phase momentum governing equation unlike in the study performed by Wen and Evans[156], where the convective terms were neglected. The closure problem arises with the second order moment, $\widetilde{\boldsymbol{u}\boldsymbol{u}}$, that, taking $\widetilde{u_1} = \widetilde{u}$ and $\widetilde{u_2} = \widetilde{w}$, is defined as

$$\widetilde{[\boldsymbol{uu}]}_{ij} = \widetilde{u_i u_j} = \frac{1}{\tilde{\tilde{\phi}}} \int v_i v_j \bar{W} d\boldsymbol{v}, \tag{8.13}$$

for $i = [1,2]$ and $j = [1,2]$. While a transport equation can be derived for the second order moments $\widetilde{[\boldsymbol{uu}]}_{ij}$ by taking the second moment of 8.9, they give rise to a new closure problem on the third order moments. Pandya and Mashayek [109] and Shotorban [127] closed the problem by setting the third cumulants to zero, so that the third-order moments can be expressed in terms of the lower-order moments.

Rather than closing on the third order moments, since we are interested in the growth of small, linear pertubations in stability analysis and second order effects may be assumed small, we will close the equation on the second order moments following the same approach, i.e. by setting the second cumulants equal to zero,

$$\int (v_i - \widetilde{u}_i)(v_j - \widetilde{u}_j)\bar{W} \, d\boldsymbol{v} \cong 0 \quad \Rightarrow \quad [\widetilde{\boldsymbol{uu}}]_{ij} \cong \widetilde{u}_i \widetilde{u}_j \, . \tag{8.14}$$

8.4 Linear stability analysis in non-uniformly laden shear layers

8.4.1 Modified Rayleigh's equation

To derive the Rayleigh equation for small disturbances, we follow the standard linear stability analysis approach by dividing the instantaneous flow property state

$$q = \{p, u, \widetilde{u}, w, \widetilde{w}, \widetilde{r}\} \tag{8.15}$$

into a base flow component, Q, described by the shear layer in Fig. 8.1, and a small perturbation component, q', such that

$$q = Q + q' = Q + \hat{q}e^{i\alpha(x-ct)} \ . \tag{8.16}$$

Here α and c are the wave number and phase speed of an infinitesimal perturbation. Substituting into Eqs. (8.4) and (8.12) and collecting the first order terms, the system of governing equations is algebraically reduced into a modified Rayleigh's equation for the transverse component of the gas perturbation velocity, $\hat{w}$. In the case of a *piecewise constant* mass particle loading profile, R, as in Eq. (8.3), we find:

$$\frac{d^2\hat{w}}{dz^2} = \left[\alpha^2 + \frac{\frac{d^2U}{dz^2} + R\frac{d}{dz}(M^2\frac{dU}{dz})}{U - c - iR(1-M)/\alpha\mathrm{St}}\right]\hat{w} \ , \tag{8.17}$$

where the damping factor M describes the averaged frequency response of the particle phase to fluctuations in gas velocity. It is defined by

$$M(z) = \frac{1}{1 + i\alpha\mathrm{St}\big(U(z) - c\big)} \ . \tag{8.18}$$

In the far-field the perturbation goes to zero, i.e.

$$\lim_{z\to\mp\infty} \hat{w} \to 0 \quad . \tag{8.19}$$

Proper matching conditions are derived at the interface $z = 0$. A dynamic condition is obtained by matching the pressures:

$$i\alpha\Delta\hat{p} = \Delta\left[\left((U-c) - \frac{i}{\alpha\mathrm{St}}R(1-M)\right)\frac{d\hat{w}}{dz} - \frac{dU}{dz}(1 + RM^2)\widehat{w}\right] = 0 \ , \tag{8.20}$$

where the symbol Δ represents the difference across the interface. Directly from (8.20), a kinematic condition is derived by observing that

$$\frac{i\alpha\hat{p}}{(U-c-iR(1-M)/\alpha\mathrm{St})^2}=\frac{d}{dz}\left[\frac{\hat{w}}{U-c-iR(1-M)/\alpha\mathrm{St}}\right]. \qquad (8.21)$$

Therefore, integrating across the interface, i.e. between $z\mp\beta$, in the limit of β tending to zero, another condition is found as

$$\lim_{\beta\to 0}\int_{-\beta}^{\beta}\frac{i\alpha\hat{p}}{(U-c-iR(1-M)/\alpha\mathrm{St})^2}\,dz=\Delta\left[\frac{\hat{w}}{U-c-iR(1-M)/\alpha\mathrm{St}}\right]=0\,. \qquad (8.22)$$

For the temporal stability analysis conducted in this investigation, disturbances grow in time but not in space and thus it follows that α is real while c is complex. The imaginary part, c_i, multiplied by the wavenumber α denotes the amplification rate of the disturbance, ω_i, while the real part, c_r, represents its phase speed. Eq. (8.17), together with Eqs. (8.19), (8.20) and (8.22), poses an eigenvalue problem of a second order differential operator with c as the complex eigenvalue. The Rayleigh equation for a particle-free mixing layer is obtained by setting $R=0$ in Eq. (8.17). The Rayleigh equation has real coefficients and hence any complex eigenvalues will appear in complex conjugate pairs [123]. From Eq. (8.17), we conclude that for the more complex case of particle-laden shear layers, the eigenvalues are no longer necessarily conjugate pairs because the coefficients are, in general, complex. Within the (countable) set of eigenvalues that can be computed, we are looking for eigenvalues, if any, with a positive imaginary part representing the modes of an unstable disturbance growth rate. In stability analysis of non-uniformly laden shear flows [61, 81, 17, 155], two unstable modes may coexist for certain values of the wavenumber α; the principal (or first) mode is the one which corresponds to the highest amplification rate ω_i. As shown in Section 8.5 and first pointed out by Wen and Evans [155], this is the case for a non-uniformly laden shear layer laden with particles whose response time is much greater than the perturbation time scale.

8.4.2 Numerical solution of the eigenvalue problem

The eigenvalue problem in Eq. (8.17), together with Eqs. (8.19), (8.20) and (8.22), is solved by means of a shooting technique. In far fields, the behavior of the eigenfunctions doesn't depend on the unknown phase speed c. Therefore, given an initial estimate, $c^{(0)}$, for the eigenvalue c, the candidate solution is propagated in each domain starting with the prescribed far field behavior. The *shoot* for c is adjusted to satisfy the matching conditions from Eqs. (8.20) and (8.22) within the desired accuracy.

To solve the eigenvalue problem in the infinite domain $z = (-\infty, +\infty)$, we first truncate the domain at a sufficiently large distance $z_{max} = 6z^*/\delta_\omega^*$ from the interface $z = 0$. Second, the two subdomains $\mathbf{I}_1 = [-z_{max}, 0]$ and $\mathbf{I}_2 = [0, z_{max}]$ are mapped onto a unity computational interval $\xi \in [0,1]$ by affine mappings:

$$\xi = \frac{z}{z_{max}} + 1, \quad z \in [-z_{max}, 0], \quad \xi \in [0,1] \ ; \tag{8.23a}$$

$$\xi = 1 - \frac{z}{z_{max}}, \quad z \in [0, +z_{max}], \quad \xi \in [0,1] \ . \tag{8.23b}$$

In this way integrations are conducted from $\xi = 0$ to $\xi = 1$ in both subdomains $\mathbf{I}_{1,2}$.

A fourth order variable step size Runge-Kutta method (the MATLAB subroutine `ode45`), with a relative tolerance of 10^{-9} and the absolute tolerance equal to 10^{-12} is used for integration.

To determine the boundary conditions in the far field, we first note that in the limit of large z, the velocity profile approaches the constant free stream velocities, $U_{1,2}$. Since the damping factor M depends on the velocity profile, $U(z)$, it will in turn approach a constant value in the limit of large z and Eq. (8.17) thus simplifies to:

$$\frac{d^2\hat{w}}{dz^2} \cong \alpha^2\hat{w}, \quad \text{as } z \rightarrow \mp\infty \,. \tag{8.24}$$

By requiring that the velocity perturbations vanish in the far field, $\hat{w} \xrightarrow{z\rightarrow\mp\infty} 0$, the asymptotic boundary conditions in the far-field are then given by

$$\hat{w}_1 \xrightarrow{z\rightarrow-\infty} \sigma e^{\alpha z}, \tag{8.25a}$$

$$\hat{w}_2 \xrightarrow{z\rightarrow+\infty} \gamma e^{-\alpha z}, \tag{8.25b}$$

where $\hat{w}_{1,2}$ refers to the eigenfunction (i.e. solution of Eqs. (8.17), (8.19), (8.20) and (8.22)) defined in subdomains $\mathbf{I}_{1,2}$, respectively, and σ and γ are undetermined multiplicative factors, but linearly dependent through matching conditions (8.20) and (8.22).

Because we are interested in unstable flow perturbations, we concern ourselves only with solutions to the eigenvalue problem in Eq. (8.17), c, with a positive imaginary part. For wave numbers with multiple unstable modes, the choice of the initial estimate $c^{(0)}$ is key since it needs to be ”close” to the target in order for the algorithm to successfully converge. Because of this sensitivity, a range of $c^{(0)}$ values must be tested before determining the type of instability arising from the initial perturbation. The candidate solutions, $\hat{w}_{1,2}(\xi, c^{(0)})$, are computed by integrating Eq. (8.17) from $\xi = 0$ to $\xi = 1$ starting with the far field behaviors (8.25a) and (8.25b). Without loss of generality, multiplicative factors σ and γ are set to unity.

At the interface between the subdomains $\mathbf{I}_1$ and $\mathbf{I}_2$, a matching condition is applied. This matching condition is derived as follows. Because of the linearity of the spectral problem

$$\hat{w}_1(\xi, c^{(0)}) = \sigma \left[\hat{w}_1(\xi, c^{(0)})\right]_{\sigma=1}, \tag{8.26a}$$

$$\hat{w}_2(\xi, c^{(0)}) = \gamma \left[\hat{w}_2(\xi, c^{(0)})\right]_{\gamma=1}, \tag{8.26b}$$

and by imposing (8.20) and (8.22), the second order linear system

$$MC(\xi = 1, c^{(0)}) \begin{pmatrix} \sigma \\ \gamma \end{pmatrix} = 0 \tag{8.27}$$

is obtained, where MC is the matching condition matrix.

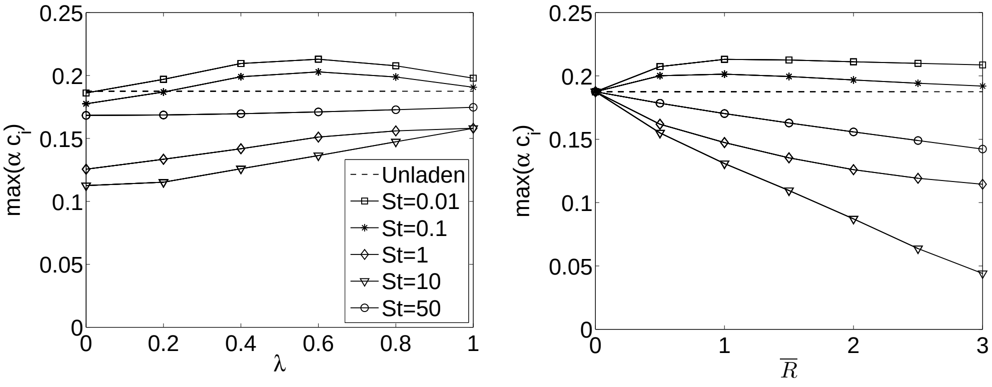
Eq. (8.27) admits the solution $\sigma = \gamma = 0$, but non trivial solutions may exist if $c^{(0)}$ is an eigenvalue of the spectral problem. This condition would zero the determinant of the matching condition matrix:

$$F(c^{(0)}) = \det\left[MC(c^{(0)})\right]_{\xi=1}. \tag{8.28}$$

If $F(c^{(0)})$ is not sufficiently close to zero (i.e. if the initial guess doesn't approximate the eigenvalue c within the desired accuracy), the shoot $c^{(0)}$ is adjusted using a Newton-Krylov solver (described by Kelley [76]) and replaced by $c^{(1)}$. Starting at the far field with the prescribed behavior, Eq. (8.17) is integrated again and the new candidate solutions are computed:

$$\hat{w}_1(\xi, c^{(1)}) = \sigma \left[\hat{w}_1(\xi, c^{(1)})\right]_{\sigma=1}, \tag{8.29a}$$

$$\hat{w}_2(\xi, c^{(1)}) = \gamma \left[\hat{w}_2(\xi, c^{(1)})\right]_{\gamma=1}, \tag{8.29b}$$



(a) Maximum growth rates for $\overline{R} = 1$ (b) Maximum growth rates for $\lambda = 0.5$

Figure 8.2: Maximum perturbation growth rates for an antisymmetric configuration, $\varepsilon = 0$, of a two-phase shear layer with differing particle St. Figure (a) shows the maximum growth rates for an increasing jump in particle concentration between the two layers, λ, but fixed mean mass loading, $\overline{R} = 1$, while in (b), a fixed $\lambda = 0.5$ but variable $\overline{R}$ is studied. The unladen case is shown as a reference in each of the figures as well.

together with

$$F(c^{(1)}) = \det\left[MC(c^{(1)})\right]_{\xi=1} . \tag{8.30}$$

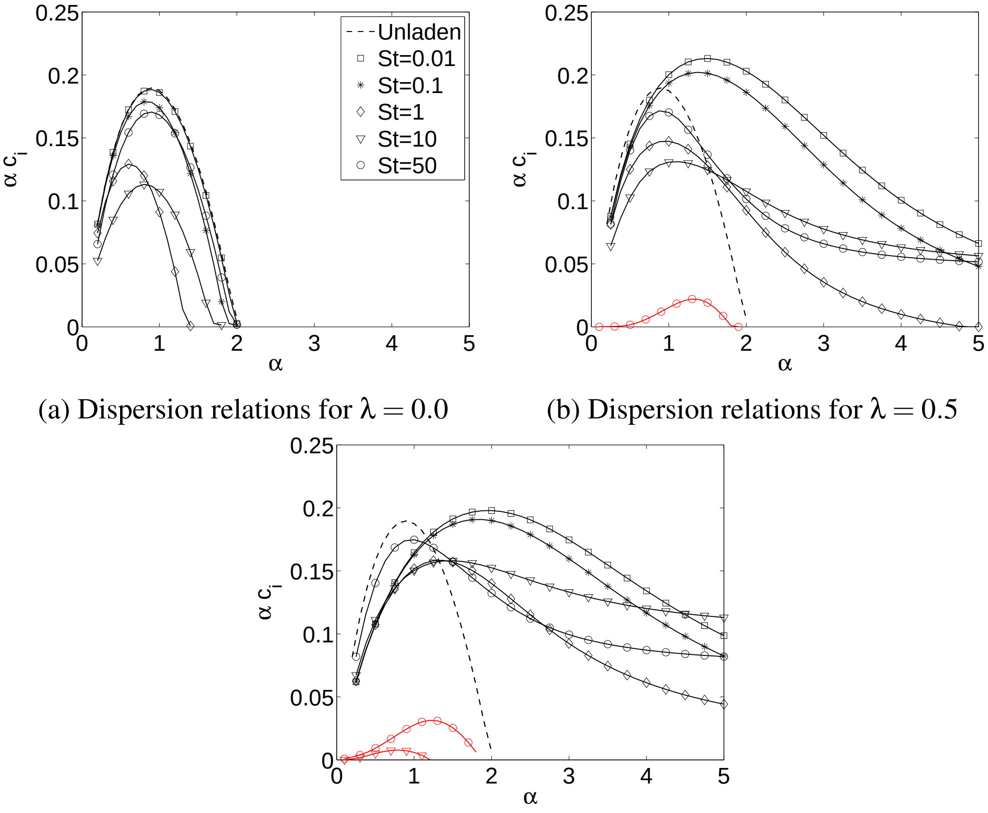
The procedure stops at the n^{th} iteration, when:

$$F(c^{(n)}) - \underbrace{F(c)}_{=0} = F(c^{(n)}) \leq 10^{-9}. \tag{8.31}$$

8.5 Disturbance Growth Rates from Stability Analysis

8.5.1 Antisymmetric mixing layer

In an antisymmetric mixing layer, $\varepsilon = 0$, both the fluid velocity direction and the particle mass loading ratio are reflected across the $z = 0$ axis. Consistently with previous linear stability analysis of uniformly loaded shear layers [118, 144, 142], we show that the shear layer's instability, eventually characterized by the maximum value over the different wavenumbers, α, of the amplification rate, $\omega_i = \alpha c_i$, depends strongly on the



(a) Dispersion relations for $\lambda = 0.0$

(b) Dispersion relations for $\lambda = 0.5$

(c) Dispersion relations for $\lambda = 1.0$

Figure 8.3: Dispersion relations for a fixed mass loading, $\overline{R} = 1$, alongside a reference unladen case where $\overline{R} = 0$ with variable jump in particle concentration between the shear layers (a) $\lambda = 0.0$ (b) $\lambda = 0.5$ (c) $\lambda = 1.0$. The second mode is shown in red.

inertial properties of the particles.

For an increasing jump in the particle concentration, $0 \leq \lambda \leq 1$, between the two layers, particles with St < 1 may destabilize the fluid flow as compared to the unladen flow configuration. The destabilizing effects are highest for $\lambda \cong 0.5$ as can be seen in 8.2a. For this value of λ, with increasing mean mass loading $\overline{R}$ (see 8.2b), the destabilizing effect of the low St laden configuration is reduced for $\overline{R} > 1$. While the strength of the stabilization depends on the value of the offset parameter, λ, intermediate St ($1 <$ St $<$ 10) particles have a stabilizing effect for all λ.

As is evident in our results for λ=0 in 8.2a, we confirm Thevand and Daniel's

[142] conclusion that the addition of particles is always stabilizing for a *uniformly* laden inviscid, nearly incompressible shear layer. The destabilizing effect, hence, is clearly induced by the jump in the particle mass loading at the interface ($\lambda > 0$) contrary to the findings of Wen and Evans [156].

For a bulk mass loading $\overline{R} = 1$, the dispersion relation in 8.3a for a uniformally laden ($\lambda = 0$) shear layer shows that there are no unstable modes for $\alpha \geq 2$. However, in the case of discontinuous mass loading, the instability in the first mode is extended for greater streamwise wavenumbers, see 8.3b and 8.3c. As first pointed out by Wen and Evans [155], for the case of a piecewise linear velocity profile laden with stationary particles (not responding to fluid oscillations), a second weaker unstable mode arises for ”sufficiently” high St, depending on the value of the differential parameter λ. In particular, with λ increasing, the second mode appears at lower St, while the most unstable wave numbers are shifted to modes with higher frequency, α.

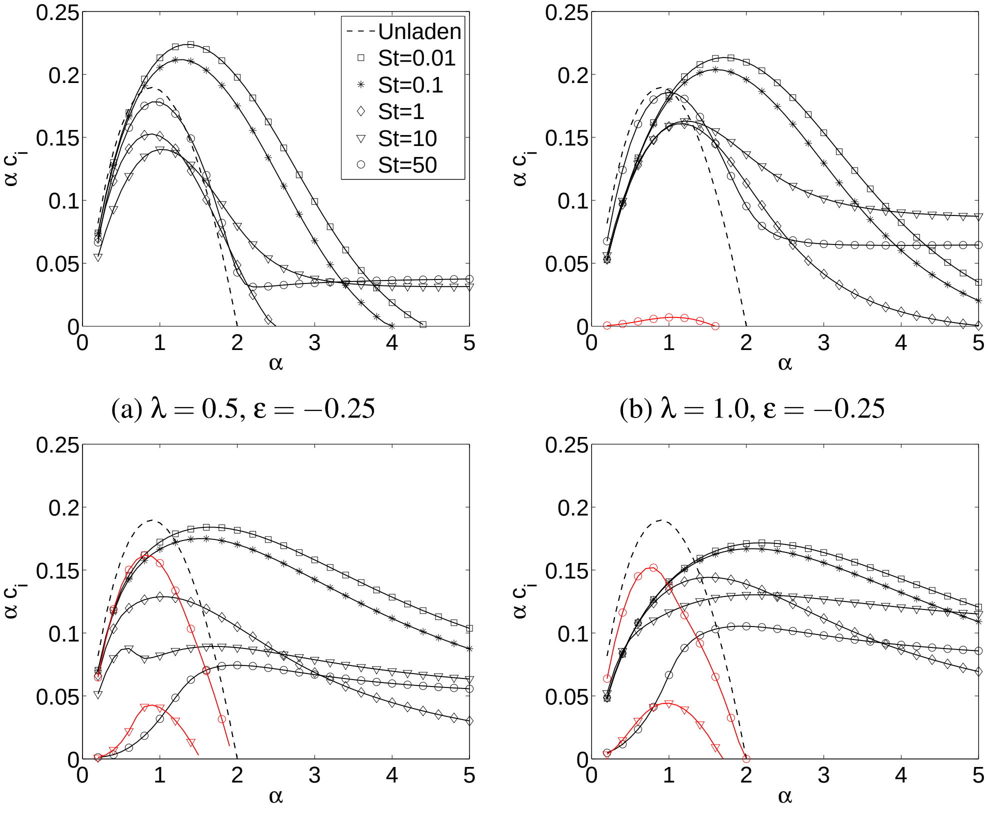
The two unstable modes can be characterized by looking at their phase speeds, qualitatively similar to those found by Wen and Evans [155]. In summary, while Kelvin-Helmholtz modes are standing waves, the first and second unstable modes computed here travel in opposite directions. The first mode, usually the dominant one, is defined as a dispersive wave traveling in the negative direction. In the long wave range, the phase speed of the first mode grows in magnitude for smaller St. The second mode resembles the Holmboe instability in density-stratified mixing layer flow. It only exists in non-uniformly laden flows, where the density between layers differs because of the particle loading. The second mode is only unstable when the Stokes number is sufficiently high and only in the long and medium wave range. It becomes stable for a high wavenumber (short wave range), approaching the value computed for the particle-free mixing layer.

8.5.2 Effect of ε

To further investigate if a jump in the particle mass loading may induce a destabilizing effect on the temporal development of the mixing layer, we analyze the impact of moving the fluid shear layer interface in the z-direction. 8.4 shows the dispersion relations for the case of an asymmetric loading configuration, with a shift in the z-direction of $\varepsilon = \pm 0.25$.

For the case of $\varepsilon < 0$, the majority of the boundary layer resides in the region of higher mass loading, $R_1 > 1$, causing the flow to be more stable at higher wave numbers independent of St. A comparison of Figs. 8.3b, 8.4a for $\lambda = 0.5$ and Figs. 8.3c, 8.4b for $\lambda = 1.0$ shows that the maximum growth rate is slightly higher and occurs at a lower wave number for the $\varepsilon = -0.25$ case as well. In particular, for the first mode, the imaginary part of the phase speed, c_i, is negative (and hence stable) at lower wave numbers, α, for St<1. The second mode is damped with $\varepsilon < 0$. As can be seen by the red lines (or lack thereof) in 8.4a and 8.4b, the second mode does not grow for the case of $\lambda = 0.5$ for any St up to and including 50 but for $\lambda = 1.0$, a weak, second mode does appear at low wave numbers.

A comparison of Figs. 8.3b, 8.4c for $\lambda = 0.5$ and Figs. 8.3c, 8.4d for $\lambda = 1.0$ shows that a shear layer with an asymmetry of $\varepsilon = 0.25$ stabilizes the maximum growth rate of the flow perturbations. In the case of a positive ε, most of the boundary layer resides in the region of lowest particle mass loading. The maximum eigenvalue locations shift towards higher wave numbers, α, with increasing ε, hence the asymptotic growth rate in the short wave range (large α) is strongly enhanced. With positive asymmetry, the second mode becomes more predominant. In fact, the second mode is the dominant mode for St=50. Since the maximum eigenvalue of the second mode occurs at a smaller α, a longer wave type instability characterizes the linear growth when the second mode dominates the first one.



(a) $\lambda = 0.5$, $\varepsilon = -0.25$

(b) $\lambda = 1.0$, $\varepsilon = -0.25$

(c) $\lambda = 0.5$, $\varepsilon = 0.25$

(d) $\lambda = 1.0$, $\varepsilon = 0.25$

Figure 8.4: The dispersion relations for (a) $\lambda = 0.5$ and $\varepsilon = -0.25$, (b) $\lambda = 1.0$ and $\varepsilon = -0.25$, (c) $\lambda = 0.5$ and $\varepsilon = 0.25$ and (d) $\lambda = 1.0$ and $\varepsilon = 0.25$ with varying St and an average mass loading, $\overline{R} = 1$, for the first modes are shown in black. Dispersion relations for the second modes are represented in red.

8.5.3 Stability and energy transfer between phases

To elucidate the underlying physical mechanisms in the increased instability induced by fast-responding particles, we analyze the energy transfer between the carrier and dispersed phases. Because of viscous drag between the two phases, the amount of energy leaving the fluid phase does not equal the energy transferred to the particle phase. The energy transfer budget, firstly introduced by Tong and Wang [144], has never been applied before to a case with non-uniformly laden particle mass loading. The interphase energy transfer budget consists of three components: the energy rate transferred to the particle phase, the energy rate departing from the continuous phase and the viscous dissipation rate induced by the presence of the particles (Stokes drag).

Because of relative motion between the phases, energy is transferred from one phase to the other. First, the energy rate (per unit volume) transferred to the particle phase, P^*, is determined by

$$P^*(x,z) = \frac{\rho^*(\Delta U^*)^3}{\delta^*_w} \underbrace{\frac{1}{\mathrm{St}}\widetilde{r}(\boldsymbol{u}-\widetilde{\boldsymbol{u}})\cdot\widetilde{\boldsymbol{u}}}_{P(x,z)} \,, \tag{8.32}$$

where ρ^*, ΔU^*, δ^*_w are the dimensional reference quantities. In this analysis, the volume is computed by considering a unit reference length in the transversal y-direction (i.e. $y^* = \delta^*_w$). The *non-dimensional* energy rate (per unit volume), P, is defined as the product between the momentum transferred from the gas to the particle phase (see also 8.5) and the Eulerian particle phase velocity.

Analogously, the energy rate (per unit volume) departing from the continuous phase, Q^*, can be defined as

$$Q^*(x,z) = \frac{\rho^*(\Delta U^*)^3}{\delta^*_w} \underbrace{\frac{1}{\mathrm{St}}\widetilde{r}(\boldsymbol{u}-\widetilde{\boldsymbol{u}})\cdot\boldsymbol{u}}_{Q(x,z)} \,, \tag{8.33}$$

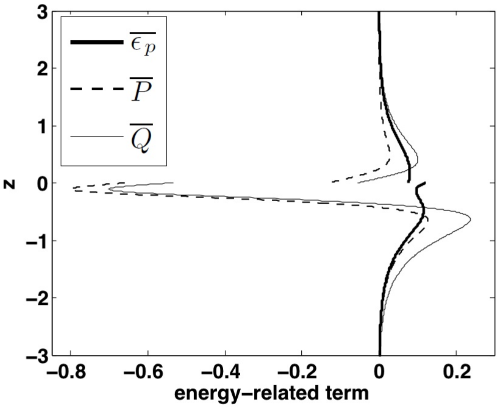


Figure 8.5: Streamwise average of energy-related terms with a discontinuous mass loading, $\lambda = 0.5$, *St*=1, α=1 and $\overline{R}$=1.

Both Q and P can be either positive or negative, since they depend on the inertial response of both the fluid and the particles.

Finally, because of viscous drag forces, particles dissipate energy from the carrier phase (and vice versa). The local dissipation rate (per unit volume), ε_p, represents the difference between Q and P:

$$\varepsilon_p = Q - P = \frac{1}{\mathrm{St}}\widetilde{r}(\boldsymbol{u} - \widetilde{\boldsymbol{u}})^2 \, . \tag{8.34}$$

To determine if the particle phase is stabilizing or destablizing the flow with respect to the single phase case, we consider the bulk energy-related budgets [144]. Starting from the definitions above, the bulk energy-related budgets $\langle\overline{P}\rangle$, $\langle\overline{Q}\rangle$, $\langle\overline{\varepsilon_p}\rangle$ are computed by first taking a streamwise average over one wavelength (indicated by the overbar):

$$\overline{P}(z) = \frac{\alpha}{2\pi}\int_0^{2\pi/\alpha} P(x,z)dx, \tag{8.35a}$$

$$\overline{Q}(z) = \frac{\alpha}{2\pi}\int_0^{2\pi/\alpha} Q(x,z)dx, \tag{8.35b}$$

$$\overline{\varepsilon_p}(z) = \frac{\alpha}{2\pi}\int_0^{2\pi/\alpha} \varepsilon_p(x,z)dx, \tag{8.35c}$$

and then integrating over z (indicated by angle brackets):

$$\left\langle\overline{P}\right\rangle = \int_{-\infty}^{+\infty}\overline{P}(z)dz \cong \int_{-z_{max}}^{+z_{max}}\overline{P}(z)dz, \tag{8.36a}$$

$$\left\langle\overline{Q}\right\rangle = \int_{-\infty}^{+\infty}\overline{Q}(z)dz \cong \int_{-z_{max}}^{+z_{max}}\overline{Q}(z)dz, \tag{8.36b}$$

$$\left\langle\overline{\varepsilon_p}\right\rangle = \int_{-\infty}^{+\infty}\overline{\varepsilon_p}(z)dz \cong \int_{-z_{max}}^{+z_{max}}\overline{\varepsilon_p}(z)dz. \tag{8.36c}$$

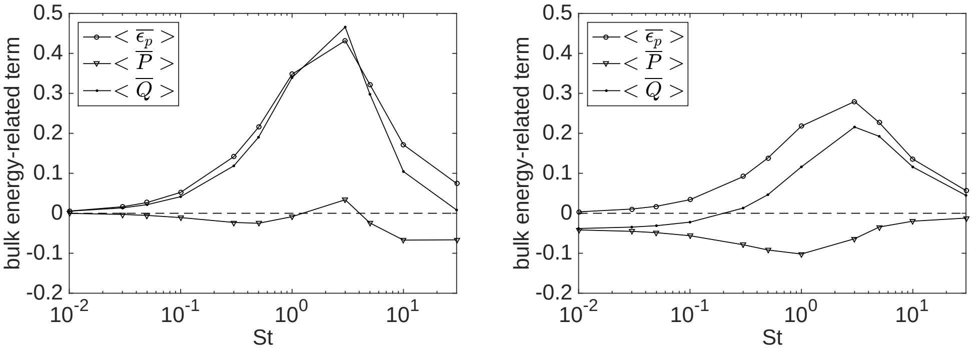
Whether the flow is more stable or not relative to the single phase case depends on the sign of $\left\langle\overline{Q}\right\rangle$, that is zero in the case of single phase flow. For a negative $\left\langle\overline{Q}\right\rangle$, a net energy is *globally* transferred from the particle phase to the carrier phase and the work exerted by the particles on the carrier phase is larger than the dissipation rate itself. Particles then destabilize the flow.

The bulk energy-related budgets are computed within the framework of the linear instability. In particular, referring to (8.16) for the decomposition of the flow property state into a basic component and a perturbation component (indicated by the apostrophe), the streamwise averaging of energies leads to:

$$\overline{P}(z) = \frac{1}{\mathrm{St}}\left(R\overline{\tilde{\boldsymbol{u}}'\cdot(\boldsymbol{u}'-\tilde{\boldsymbol{u}}')} + U\overline{\tilde{r}'(u'_x-\tilde{u_x}')}\right), \tag{8.37a}$$

$$\overline{Q}(z) = \frac{1}{\mathrm{St}}\left(R\overline{\boldsymbol{u}'\cdot(\boldsymbol{u}'-\tilde{\boldsymbol{u}}')} + U\overline{\tilde{r}'(u'_x-\tilde{u_x}')}\right), \tag{8.37b}$$

$$\overline{\varepsilon_p}(z) = \frac{R}{\mathrm{St}}\overline{\left(\boldsymbol{u}'-\tilde{\boldsymbol{u}}'\right)^2}. \tag{8.37c}$$



(a) $\lambda = 0.0$

(b) $\lambda = 0.5$

Figure 8.6: Bulk energy-related budget as a function of St ($\alpha = 1$, $\overline{R} = 1$) for (a) $\lambda = 0.0$ and (b) $\lambda = 0.5$.

In 8.5 the streamwise averaged energy contributions are plotted versus the cross stream direction, z, for a representative particle-laden, non-uniformly laden shear flow with $\overline{R} = 1$, $\alpha = 1$, St$= 1$, $\lambda = 0.5$ and ε=0. $\overline{P}$ and $\overline{Q}$ are negative near the center of the mixing layer. At that location, kinetic energy is transferred from the particle phase to the carrier phase. Away from the interface, the signs on $\overline{P}$ and $\overline{Q}$ are positive and energy flows from the fluid to the particle phase.

The bulk energy budgets versus St plotted in 8.6a, for a uniform mass loading, and 8.6b, for a discontinuous mass loading, show a transition from a consistently stabilizing energy transfer with uniform loading to a net destabilizing energy transfer for small values of St when the mass loading is discontinuous. With increasing St in 8.6b, a significantly increasing (always positive) viscous dissipation changes the sign of $\langle \overline{Q} \rangle$ and yields a more stable flow with respect the single phase configuration at intermediate and large St. The transition from a destabilizing to a stabilizing effect from the particles occurs at approximately $St \cong$0.2 for a mixing layer with $\bar{R}$=1 and λ=0.5. Note further that for St between 1 and 10, the dissipation rate and the rate of work departing from the continuous phase have a peak, in agreement with the fact that the maximum stabilizing effect is obtained for this range of St.

8.6 Eulerian-Lagrangian Validation

8.6.1 Eulerian-Lagrangian model

To assess the validity of the theoretical results from the linear stability analysis, based on a lower-order stochastic Eulerian particle model, we compute the disturbance growth rate of the mixing layer with differential loading using an extensively tested and validated fully non-linear, high-order resolution, EL model described by Jacobs and Don [68]. This EL model is based on closure point particles, which are traced in the Lagrangian frame, while the compressible Euler equations are solved with a higher-order resolution weighted essentially non-oscillatory (WENO) finite difference scheme [18] in an Eulerian frame. The source term on the right-hand side of the Euler equations, modeling the drag force from the individual computational particles, is defined as

$$\boldsymbol{S} = \sum_{i=1}^{N_c} \mathbf{K}\mathbf{W_m} \,. \tag{8.38}$$

In (8.38), **K** is a normalized weighing function, which distributes the influence of each computational particle onto the carrier flow using a high-order smooth spline deposition function and $\mathbf{W_m}$ is a function that describes the momentum of each particle. A high-order essentially non-oscillatory (ENO) algorithm interpolates the carrier phase velocity and temperature onto the Lagrangian particle locations so the particle phase momentum and energy equations can be solved directly. These methods are shown to preserve high-order resolution[68] and hence capture small-scale flow structures that lead to the growth of the instabilities in the shear layer with low dissipation and dispersion errors. This numerical scheme ensures a high-fidelity resolution of this validation computation with two-way coupling between the particle and fluid phases. For a detailed derivation and validation of the Eulerian-Lagrangian model used in the computations, see for example

Jacobs and Don [68], Jacobs et. al.[66] and Davis et. al. [33].

8.6.2 Flow initializations

We compute four cases that are summarized in Table 8.1 to verify the theoretical results including an unladen case and the three most unstable modes for discontinuously laden cases (St=0.01, 0.1 and 1). The carrier and particle flow velocities are initialized with the hyperbolic-tangent streamwise velocity profile defined in (8.1), perturbed with the real part of the eigenfunctions obtained from the linear stability analysis,

$$u(x,z,t_0) = U(z) + \underbrace{\Theta\Re\{\hat{u}(z)e^{i\alpha(x-ct_0)}\}}_{u'} \ , \tag{8.39}$$

$$w(x,z,t_0) = \underbrace{\Theta\Re\{\hat{w}(z)e^{i\alpha(x-ct_0)}\}}_{w'} \ . \tag{8.40}$$

Here, t_0 is an arbitrary initial time that we set to t_0=5, i.e. the instant at which the computation starts. The small coefficient $\Theta = |\Delta U|/2 \times 10^{-3}$ ensures linear growth of the perturbation. For one wavelength ($2\pi/\alpha$), the contour plot for the z-directed flow velocity $w = w'(x,z,t_0)$ at time $t = t_0$ is shown in 8.7a and 8.7b for particles of St=0.1 and St=1, respectively. The contour plot for the case of St=0.01 is indistinguishable from 8.7a and is therefore not shown.

Since our EL method solves the compressible Euler equations and we have performed an incompressible stability analysis, we set a low free-stream Mach number, Ma_∞, equal to 0.15, which minimizes density variations. While a lower Mach number would be more preferable, the Euler solver becomes excessively computationally demanding because of restrictions on the stable, explicit time step.

The characteristic fluid Reynolds number, Re, has been set to 1.80×10^6 in order to compute the particle drag. It is not used in the inviscid fluid governing equations[68].

Table 8.1: Initialization parameter settings depending on the value of St.

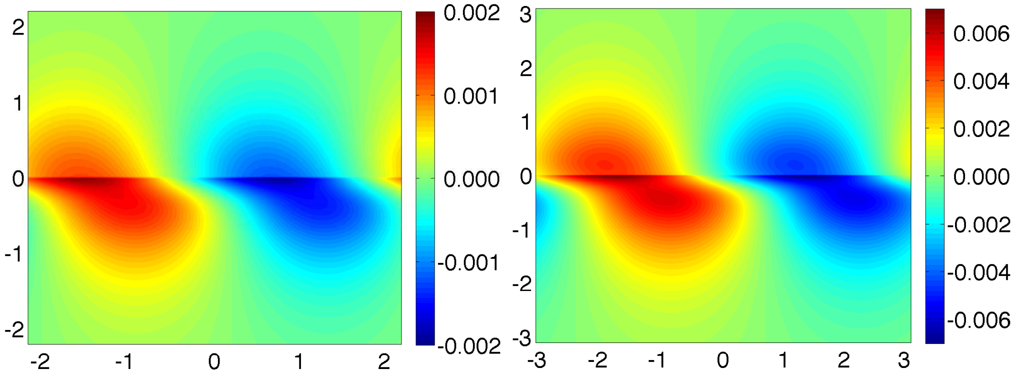
St	diameter (D)	N_p	α	L_x	L_z	grid pts	N_c	$\overline{S}_f$	$\overline{R}$
	$\left(\sqrt{\frac{18\text{St}}{\rho_p\text{Re}}}\right)$			$(4\pi/\alpha)$		$(N_x\text{x}N_z)$		(N_p/N_c)	$\left(\frac{\pi N_p\rho_p D^3}{6\rho V}\right)$
unladen	-	-	0.9	13.96	12	465x400	-	-	-
0.01	1.00E-05	8.00E+13	1.4	8.98	12	299x400	1.91E+06	4.18E+07	0.39
0.1	3.16E-05	6.50E+12	1.4	8.98	12	299x400	1.91E+06	3.39E+06	1.00
1	1.00E-04	2.88E+11	1.0	12.57	12	419x400	2.68E+06	1.07E+05	1.00

Given the (non-dimensional) density of the particle, $\rho_p = 10^3$, and particle's St, the particle diameter, D, is computed and reported in Table 8.1.

The 2D computational domain spans from $z = -6$ to $z = 6$, so that the perturbation at the z-boundaries is negligible, which has been tested with a domain convergence study. The size of the domain in the x-direction, L_x, fits two waves ($4\pi/\alpha$, the domain represented in 8.7a has been truncated to show only one wave), with the wavenumber of the most unstable modes, α, in Tab. 8.1. The boundary conditions are periodic in x, while free-stream flow conditions are specified on the z-boundaries. Computations are performed on a structured grid with uniform and equal grid spacing in the x- and z-direction. A fifth-order WENO scheme approximates the spatial derivatives and a third order total variation diminishing (TVD) Runge-Kutta scheme advances the solver temporally. We have performed resolution studies and find that with $N_z = 400$ grid points in the z-direction, the solution is more than sufficiently resolved. The number of grid points in the x-direction varies with L_x from case to case in order to ensure square cells.

The particles are initially uniformly spaced in the computational domain and are initialized with the fluid velocity. The number of particles is fixed for a given mass loading, $\overline{R}$, and vice versa, see Table 8.1, where $V = 1 \times L_x \times L_z$ is the volume. For St=0.1 and St=1, $\overline{R} = 1.0$ is imposed and N_p is consequently obtained, whereas when St=0.01 it was preferred to lower $\overline{R}$ to 0.39 (i.e. a volume fraction of 0.00039) to reduce the computational costs.

However, since the number of physical particles is by far too large to fully resolve



(a) St=0.1

(b) St=1

Figure 8.7: Level curves of the initial z-directed flow perturbation velocity $w = w'$ defined in Eq. (8.40) for different values of the Stokes number, see Tab. 8.1. The figures are magnified to focus on one wave in the domain.

with limited computational resources, we consider a reduced number of *computational* particles, N_c, in our Eulerian-Lagrangian simulations [68] so that each computational particle represents a cloud of physical particles. The ratio of the number of computational particles to the number of physical particles, $\overline{S}_f = N_p/N_c$, is tabulated in Table 8.1. The discontinuity in the mass loading is specified by weighting the ratio, $\overline{S}_f$, above and below the bulk mass loading interface as follows:

$$S_{f_j} = \begin{cases} \overline{S}_f(1-\lambda), & \text{if } Z_j(t_0) \geq 0\,, \\ \overline{S}_f(1+\lambda), & \text{if } Z_j(t_0) < 0\,, \end{cases} \tag{8.41}$$

with the particle counter $j = 1, 2, \ldots, N_c$ and Z_j is the location of the particle in the z-direction. To ensure a statistically meaningful description of the two phase coupling source terms, sixteen computational particles are initially uniformly located in every fluid grid cell. The boundary conditions are periodic in x-direction for each particle, while particles leave the domain freely in the z-direction.

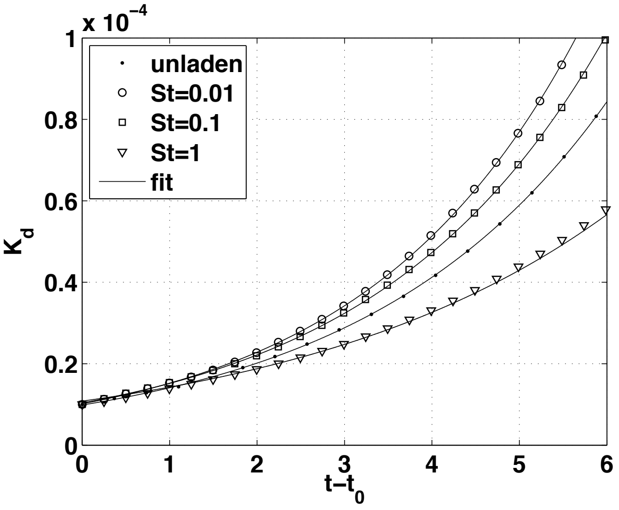


Figure 8.8: Time evolution of integrated disturbance fluid kinetic energies, K_d, for different values of St. Marks correspond to results obtained from the simulation, while continuous lines refer to the exponential fitting.

8.6.3 Growth rate computations from numerical experiments

To validate the results from the linear stability analysis, we compare growth rates of the initial perturbation in the EL simulation with the growth rates determined from the stability analysis. As a measure of the magnitude of the perturbation at a given time, we integrate the disturbance kinetic energy of the fluid phase over the computational domain

$$K_d = \frac{1}{2}\int_D \rho\left(\underbrace{(u-U)^2}_{=u'^2}+\underbrace{w^2}_{=w'^2}\right)dxdz\ . \tag{8.42}$$

Disturbance kinetic energy transient plots for the case of particle-free and non-uniformly laden (St=0.01, 0.1, 1) mixing layer are reported in 8.8. The destabilizing effect induced by quick-responding particles is significant. In particular, the growth rate is computed as

$$\omega_i = \frac{1}{2}\frac{d}{dt}\ln(K_d) \tag{8.43}$$

from an exponential fit to the integrated disturbance kinetic energy, K_d, versus time, t.

Table 8.2: Comparison of growth rates obtained using the Eulerian-Lagrangin (EL) method to those obtained from linear stability analysis (LSA).

St	Growth Rate		Difference
	EL	LSA	%
unladen	0.1789	0.1897	-5.7
0.01	0.2026	0.2040	-0.7
0.1	0.1896	0.2021	-6.1
1	0.1379	0.1474	-6.4

In Table 8.2, a comparison of the growth rates obtained from the EL methods and the linear stability analysis (LSA) shows that the EL computations and LSA predict the same trends and are in very good agreement. The slightly smaller growth rates of the EL computations as compared to the LSA growth rates are attributed to small compressibility effects that reduce the disturbance growth.

The EL computations confirm that the growth rate of the mixing layer is strongly dependent on the Stokes number of the particle phase. In the EL computations, the growth rate of the mixing layer laden with particles with small St is larger than for an unladen shear layer. Hence, like in the LSA analysis, the EL computations confirm that particles with small St have a destabilizing effect as compared to the unladen flow.

8.7 Conclusions

For the first time, we have shown that a non-uniformly laden particle mass loading may trigger a destabilizing effect on the mixing layer development of particle-laden shear layers as compared to the pure-gas flow configuration. This destabilization occurs when the gas is laden with small-inertia particles (St $<$ 0.2).

The application of the interface energy transfer budget [144] to the case of small-inertia particles shows a net energy rate flowing globally from the particulate phase to the gas phase. This energy transfer explains why the destabilization effect is seen with low St particles. For intermediate particle's St (1<St<10), maximum stabilizing effect is computed, while at larger St two unstable modes may coexist.

Furthermore, the growth rate computations from the linear stability analysis are verified numerically through simulations based on a fully non-linear Eulerian-Lagrangian (EL) model [68]. The growth rates found in numerical experiments using the EL method are in good agreement with growth rates from the linear stability analysis. In particular, the destabilizing effect induced by the presence of particles with low St is validated by the EL computations.

Previous attempts to tackle the linear stability problem of a non-uniformly laden particle-laden shear layer [155, 156] suffered from the omission of the convective terms in the momentum equation for the particle phase and didn't preserve consistency with several studies on *uniformly* laden shear flows [118, 144, 142]. The stochastic Eulerian-Eulerian (EE) model proposed here, obtained from a combination of the Liouville theorem applied to the system of dynamical particle equations and a method of moments [109, 127], does include the convective terms. In particular, zeroing the second order cumulant of the second velocity moments not only preserves consistency, but also allows for the computation of results in good agreement with full-scale EL simulations.

Sections 8.1 to 8.7, in full, are a reprint of the material as it appears in The Effect of Non-Uniform Mass Loading on the Linear, Temporal Development of Particle-Laden Shear Layers. Senatore, Giacomo; Davis, Sean; Jacobs, Gustaaf AIP Publishing, 2015. The dissertation/thesis author was a primary investigator and author of this material.

Chapter 9

Conclusions

Numerical methods and modeling tools have been developed to enable the improved study of high-speed particle-laden flows and related instabilities. A high-order Weighted Essentially Non-Oscillatory (WENO) Eulerian Lagrangian (EL) method is implemented. A multi-scale framework for the closure of the macro-scale modeling of averaged particle cloud dynamics and drag forcing with micro-scale statistics has been introduced. Normal, wake and shear instabilities are studied.

9.1 Numerical Methods/Modeling

EL Solver: A parallel Lagrangian particle tracking algorithm is two-way coupled with a high-fidelity WENO-Z scheme to solve shocked particle-laden flows. A high-order Essentially Non-Oscillatory (ENO) interpolation ensures a stable interpolation of the fluid velocity and temperature to the particle locations. An ENO sub stencil determined using the method of lines can be unstable if the shock is misaligned with the tensor grid in multiple dimensions. The sub stencil resulting from a summation of the divided differences within an area around the particle's host cell removes these numerical instabilities.

Sub Scale Modeling: Sub-scale particle dynamics are modeled to a single computational particle by a method of averaging. The so-called Subgrid Particle Averaged Reynolds Stress Equivalent (SPARSE) model can reduce the number of computational particles by a factor on the order of 10^3. The dispersion of a cloud of 27,000 physical particles in 3D isotropic turbulence is modeled using a single computational particle with both the current state-of-the-art method and the SPARSE model. Near zero errors in the computation of the particle velocity are found using the SPARSE model while the first order model produced errors on the order of 10%.

Multi-Scale Closure: Metamodeling techniques that build closure models from micro-scale statistics for macro-scale computations, are compared. Because mesoscale experiments are computationally expensive, the number of support points must be minimized. For the low number of inputs desired, a dynamic kriging method builds the most accurate representation of the data needed to close the particle drag and SPARSE models.

9.2 Instabilities

Normal Instabilities: The formation of numerical oscillations on grid-aligned, strong shocks, called carbuncles, is presented for the first time in a particle-laden flow. The effects of the alternating high and low velocity regions behind the carbuncle, which bleed downstream, cause an instability in the particle phase. The particle phase instability manifests as a spike of particles protruding upstream of an otherwise well-defined leading edge. The strong two-way coupling of the particle and fluid phases amplifies the effects of the carbuncle, causing an instability in the density gradient of the fluid just downstream of the particle instability. By increasing the power parameter in the non-linear WENO-Z weighting, the formation and downstream effects of the carbuncle instabilities can be mitigated. Increasing the WENO-Z power parameter adds dissipation to areas of the flow containing steep gradients while not effecting the high order accuracy of the scheme in smoother regions of the flow.

Wake Instabilities: A study on the effect of initial shape, aspect ratio and rotation of a cloud of particles on the particle dispersion and flow development behind a moving shock is conducted. Changing the initial shape of even a low volume fraction (4% in this study) particle cloud changes the particle-laden flow dynamics. For initially aerodynamically shaped clouds, such as a triangle, the flow remains attached along the sides and the particle cloud is compressed towards the symmetry line. Because the cloud is compressed rather than dispersing individual particles into the flow, the overall Stokes number of the cloud is increased resulting in a reduction of the downstream convection as compared to more blunt shapes. In the case of initially blunter rectangular and circular shaped clouds, particles separate along strong shear layers. The particles entrained in the fluid shear layers form arms, shielding the downstream portion of the particle cloud. Because of the streamwise particle dispersion, the averaged particle cloud accelerates more quickly than the aerodynamically shaped cloud.

Shear Instabilities: To better understand the growth rate in the shear layers formed by the interaction of a high-speed flow and a particle cloud, a fundamental analysis of the linear growth rate of particle-laden shear layers was conducted. The growth rates computed with a linear stability analysis of a non-uniformly particle-laden shear layer compare well with growth rates found in numerical experiments with the EL method. Non-uniform mass loading of small inertia particles ($\mathrm{St} < 0.2$) creates a net energy transfer from the particle phase to the carrier phase, causing a destabilization of the flow. For intermediate particle response times ($1<\mathrm{St}<10$), a maximum stabilizing effect is computed. At large St, two unstable modes may coexist and the addition of particles always stabilizes the flow.

9.3 Future Work

Extension of this work should be directed at numerical improvement, fundamental physical understanding and physical application. Micro-scale DNS results of shock-particle-turbulence interaction is needed to produce the relevant statistics for closure in both the SPARSE and the particle drag modeling of the macro-scale EL code. Models for particle-particle collisions should be implemented for the study of larger volume fraction flows. The linear stability analysis of particle-laden shear layers should be further extended to include compressibility effects and higher order moments. The addition of methods to resolve evaporation and reaction in the flow would provide more detailed results on the dispersion of fuel in scramjet combustors.

References

[1] Y. Abdelhamid and U. El Shamy. “Multiscale Modeling of Flood-Induced Scour in a Particle Bed”. In: *Bridges* 10 (2014), pp. 740–749.

[2] H. Abe, S. Natsuhiko, and R. Itatani. “High-order spline interpolations in the particle simulation”. In: *J. Comp. Phys.* 63 (1986), pp. 247–267.

[3] M Anderson et al. “An experimental and numerical study of shock interaction with a gas column seeded with droplets”. In: *Shock Waves* 25.2 (2015), pp. 107–125.

[4] Vincenzo Armenio and Sutanu Sarkar. “An investigation of stably stratified turbulent channel flow using large-eddy simulation”. In: *Journal of Fluid Mechanics* 459 (May 2002), pp. 1–42.

[5] D. Gottlieb B. Costa W. S. Don and R. Sendersky. “Two-Dimensional Multi-Domain Hybrid Spectral-WENO Methods for Conservation Laws”. In: *Comm. in Comp. Phys.* 1 (2006), pp. 550–577.

[6] D Balsara and C W Shu. “Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy”. In: *J. Comp. Phys.* 160 (2000), pp. 405–452.

[7] Volker Barthelmann, Erich Novak, and Klaus Ritter. “High dimensional polynomial interpolation on sparse grids”. In: *Advances in Computational Mathematics* 12 (2000), pp. 273–288.

[8] J. B. Bdzil et al. “Two-phase modeling of deflagration-to-detonation transition in granular materials: A critical examination of modeling issues”. In: *Physics of Fluids* 11.2 (Feb. 1999), p. 378.

[9] Nabil Benoudjit and Michel Verleysen. “On the Kernel Widths in Radial-Basis Function Networks”. In: *Neural Processing Letters* 18 (2003), pp. 139–154.

[10] C.K. Birdsall and A.B. Langdon. *Plasma physics via computer simulation*. McGraw-Hill, Inc., 1985.

[11] G. A. Blaisdell. “Numerical simulation of compressible homogeneous turbulence”. PhD thesis. Stanford Univ., CA., 1991.

[12] G. A. Blaisdell, N. N. Mansour, and W. C. Reynolds. “Compressibility effects on the growth and structure of homogeneous turbulent shear flow”. In: *Journal of Fluid Mechanics* 256 (Nov. 1993), pp. 443–485.

[13] V. Boiko and S. Poplavski. “Particle and Drop Dynamics in the Flow Behind a Shock Wave”. In: *Fluid Dynamics* 42 (2007), pp. 441–443.

[14] V. Boiko and S. Poplavsky. “Dynamics of Irregularly Shaped Bodies in a Flow Behind a Shock Wave”. In: *C. R. Mecanique* 332 (2004), pp. 181–187.

[15] V Boiko et al. “Shock wave interaction with a cloud of particles”. In: *Shock Waves* 7 (1997), pp. 275–285.

[16] V. M. Boiko and S. V. Poplavskii. “Drag of nonspherical particles in a flow behind a shock wave”. In: *Combustion, Explosion, and Shock Waves* 41.1 (Jan. 2005), pp. 71–77.

[17] P.A.M.. Boomkamp et al. “A Chebyshev collocation method for solving two-phase flow stability problems”. In: *J. Comp. Phys.* 132 (1997), pp. 191–200.

[18] R Borges et al. “An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws”. In: *J. Comp. Phys.* 227 (2008), pp. 3191–3211.

[19] G.L. Brown and A. Roshko. “On density effects and large structures in turbulent mixing layers”. In: *J. Fluid Mech* 64 (1974), pp. 775–781.

[20] J. Burkardt et al. *User Manual and Supporting Information for Library of Codes for Centroidal Voronoi Placement and Associated Zeroth, First, and Second Moment Determination*. Tech. rep. 2002.

[21] B. Castro, B. Costa, and W. S. Don. “High Order Weighted Essentially Non-Oscillatory WENO-Z schemes for Hyperbolic Conservation Laws”. In: *J. Comp. Phys.* 230 (2011), pp. 1766–1792.

[22] C. P. Chen and P. E. Wood. “A turbulence closure model for dilute gas-particle flows”. In: *The Canadian Journal of Chemical Engineering* 63.3 (1985), pp. 349–360.

[23] S Chen, C F M Cowan, and P M Grant. “Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks”. In: *IEEE Transactions on Neural Networks* 2.2 (1991), pp. 302–309.

[24] S Chen, P M Grant, and C F N Cowan. “Orthogonal least-squares algorithm for training multioutput radial basis function networks”. In: *IEEE Proceedings-F* 139.6 (1992), pp. 378–384.

[25] Victoria CP Chen et al. “A review on design, modeling and applications of computer experiments”. In: *IIE transactions* 38.4 (2006), pp. 273–291.

[26] R Clift, J R Grace, and M E Weber. *Bubbles, Drops, and Particles*. Academic Press, 1978.

[27] Noel Cressie. “Statistics for spatial data”. In: *Terra Nova* 4.5 (1992), pp. 613–617.

[28] C. Crowe, M. Sommerfeld, and Y. Tsuji. *Multiphase flows with droplets and particles*. CRC Press, 1998.

[29] C. T. Crowe, J.N. Chung, and T.R. Troutt. “Particle mixing in free shear flows”. In: *Prog. Energy Combust. Sci.* 14.3 (1988), pp. 171–194.

[30] C T Crowe, M P Sharma, and D E Stock. “The Particle-Source in Cell (PSI-Cell) model for gas-droplet flows”. In: *J. Fluids Eng.* 6 (1977), pp. 325–332.

[31] Chooleewan Dachapak et al. “Orthogonal Least Squares for Radial Basis Function Network in Reproducing Kernel Hilbert Space”. In: *IFAC Workshop on Adaptation and Learning in Control and Signal Processing and IFAC Workshop on Periodic Control Systems*. 2004, pp. 847–852.

[32] S. Davis, G. Jacobs, and W. S. Don. “Carbuncles in high resolution Eulerian-Lagrangian simulations of particle-laden flow”. In: *AIAA Journal, Submitted* (2015).

[33] S. Davis et al. “Dispersion of a cloud of particles by a moving shock: Effects of the shape, angle of rotation, and aspect ratio”. In: *Journal of Applied Mechanics and Technical Physics* 54.6 (2013), pp. 900–912.

[34] S Davis et al. “High-Fidelity Eulerian-Lagrangian Methods for Simulation of Three Dimensional, Unsteady, High-Speed, Two-Phase Flows in High-Speed Combustors”. In: *47th AIAA/ASME/SAE/ASEE Joint Propulsion Conference*. 2011.

[35] W. S. Don and R. Borges. “Accuracy of the Improved Weighted Essentially Non-Oscillatory WENO-Z Conservative Finite Difference Schemes”. In: *J. Comput. Phys.* (2013).

[36] K.L. Du and M. N. S. Swamy. *Neural Networks in a Soft Computing Framework*. Springer, 2006, pp. 183–200.

[37] Weinan E, Bjorn Engquist, and Zhongyi Huang. "Heterogeneous multiscale method: A general methodology for multiscale modeling". In: *Physical Review B* 67.9 (Mar. 2003), pp. 1–4.

[38] Volker Elling. "The carbuncle phenomenon is incurable". In: *Acta Mathematica Scientia* 29.6 (2009), pp. 1647 –1656.

[39] Gordon Erlebacher et al. "The analysis and simulation of compressible turbulence". In: *Theoretical and Computational Fluid Dynamics* 2.2 (1990), pp. 73–95.

[40] Hongbing Fang and Mark F Horstemeyer. "Global response approximation with radial basis functions". In: *Engineering Optimization* 38.04 (2006), pp. 407–424.

[41] Gregory E Fasshauer and Michael J McCourt. "Stable evaluation of Gaussian radial basis function interpolants". In: *SIAM Journal on Scientific Computing* 34.2 (2012), A737–A762.

[42] A. V. Fedorov, Yu. V. Kharlamova, and T. A. Khmel'. "Reflection of a shock wave in a dusty cloud". In: *Combustion, Explosion, and Shock Waves* 43.1 (Jan. 2007), pp. 104–113.

[43] Zhi-Gang Feng and Efstathios E Michaelides. "Drag Coefficients of Viscous Spheres at Intermediate and High Reynolds Numbers". In: *Journal of Fluids Engineering* 123.4 (2001), pp. 841–849.

[44] Stefano Ferrari, Mauro Maggioni, and N Alberto Borghese. "Multiscale Approximation with Hierarchial Radial Basis Function Networks". In: *IEEE Transactions on Neural Networks* 15.1 (2004), pp. 178–188.

[45] B. Fornberg et al. "Observations on the behavior of radial basis function approximations near boundaries". In: *Computers & Mathematics with Applications* 43.3–5 (Feb. 2002), pp. 473–490.

[46] Bengt Fornberg and Cécile Piret. "A stable algorithm for flat radial basis functions on a sphere". In: *SIAM Journal on Scientific Computing* 30.1 (2007), pp. 60–80.

[47] D. L. Frost et al. "Particle jet formation during explosive dispersal of solid particles". In: *Physics of Fluids* 24.9 (Sept. 2012), p. 091109.

[48] D. L. Frost et al. "Particle momentum effects from the detonation of heterogeneous explosives". In: *Journal of Applied Physics* 101.11, 113529 (2007), pp. –.

[49] Z. Gao, W. S. Don, and Z. Li. “High Order Weighted Essentially Non-Oscillation Schemes for Two-Dimensional Detonation Wave Simulations”. In: *J. Sci. Comput.* (2012).

[50] Sergei Konstantinovich Godunov. “A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics”. In: *Matematicheskii Sbornik* 89.3 (1959), pp. 271–306.

[51] T. Goel et al. “Pitfalls of Using a Single Criterion for Selecting Experimental Designs”. In: *International Journal of Numerical Methods in Engineering* 75 (2 2008), pp. 127 –155.

[52] Ridha Hambli. “Apparent damage accumulation in cancellous bone using neural networks.” In: *Journal of the mechanical behavior of biomedical materials* 4 (Aug. 2011), pp. 868–78.

[53] Ridha Hambli. “Multiscale prediction of crack density and crack length accumulation in trabecular bone based on neural networks and finite element simulation”. In: *International Journal of Numerical Methods in Biomedical Engineering* 27 (2011), pp. 461–475.

[54] Ridha Hambli. “Numerical procedure for multiscale bone adaptation prediction based on neural networks and finite element simulation”. In: *Finite Elements in Analysis and Design* 47.7 (July 2011), pp. 835–842.

[55] Ridha Hambli, Houda Katerchi, and Claude-Laurent Benhamou. “Multiscale methodology for bone remodelling simulation using coupled finite element and neural network computation.” In: *Biomechanics and modeling in mechanobiology* 10 (Feb. 2011), pp. 133–145.

[56] Ami Harten et al. “Uniformly high order accurate essentially non-oscillatory schemes, {III}”. In: *Journal of Computational Physics* 71.2 (1987), pp. 231 –303.

[57] Yosuke Hasegawa and Nobuhide Kasagi. “Hybrid DNS/LES of high Schmidt number mass transfer across turbulent air water interface”. In: *International Journal of Heat and Mass Transfer* 52.3-4 (2009), pp. 1012–1022.

[58] Simon Haykin. “Neural networks: a comprehensive foundation, 1994”. In: *Mc Millan, New Jersey* (2010).

[59] M.A. van der Hoef et al. “Numerical Simulation of Dense Gas-Solid Fluidized Beds: A Multiscale Modeling Strategy”. In: *Annual Review of Fluid Mechanics* 40.1 (Jan. 2008), pp. 47–70.

[60] M Holden. *Studies of boundary layer transition and surface roughness effects in hypersonic flow*. Tech. rep. 6430-A-5. CALSPAN, Oct. 1983.

[61] J. Holmboe. “On the behavior of symmetric waves in stratified shear layers”. In: *Geofys. Publ.* 24 (1962), pp. 67–113.

[62] Guang-Bin Huang, P. Saratchandran, and N. Sundararajan. “A generalized growing and pruning RBF (GGAP-RBF) neural network for function approximation”. In: *IEEE Transactions on Neural Networks* 16.1 (Jan. 2005), pp. 57–67.

[63] M. Ishii. “One-dimensional drift-flux model and constitutive equations for relative motion between phases in various two-phase flow regimes”. In: *Argonne National Laboratory Report* 77 (1977), p. 47.

[64] F. Ismail, P. Roe, and H. Nishikawa. “A Proposed Cure to the Carbuncle Phenomenon”. In: *Comp. Fluid Dyn.* (2009), pp. 149–154.

[65] G. Jacobs and W. S. Don. “A High-Order WENO-Z Finite Difference based Particle-Source-in-Cell Method for Computation of Particle-Laden Flows with Shocks”. In: *Journal of Computational Physics* 5 (2009), pp. 1365–1379.

[66] G. B. Jacobs, W. S. Don, and T. Dittmann. “High-Order resolution Eulerian-Lagrangian simulations of particle dispersion in the accelerated flow behind a moving shock”. In: *Ther. Comput. Fluid Dyn.* 26.1-4 (2012), pp. 37–50.

[67] G.B. Jacobs and J.S. Hesthaven. “High-order nodal discontinuous Galerkin particle-in-cell method on unstructured grids”. In: *J. Comp. Phys.* 214 (2006), pp. 96–121.

[68] Gustaaf B. Jacobs and Wai-Sun Don. “A high-order WENO-Z finite difference based particle-source-in-cell method for computation of particle-laden flows with shocks”. In: *Journal of Computational Physics* 228.5 (2009), pp. 1365 –1379.

[69] J.T. Jenkins and S.B. Savage. “A theory for rapid flow of identical, smooth, nearly elastic spherical particles”. In: *J. Fluid Mech* 130 (1983), pp. 187–202.

[70] G S Jiang and C W Shu. “Efficient Implementation of Weighted ENO Schemes”. In: *J. Comp. Phys.* 126 (1996), pp. 202–228.

[71] Ruichen Jin, Wei Chen, and Timothy W Simpson. “Comparative studies of metamodelling techniques under multiple modelling criteria”. In: *Structural and Multidisciplinary Optimization* 23.1 (2001), pp. 1–13.

[72] Ruichen Jin, Xiaoping Du, and Wei Chen. “The use of metamodeling techniques for optimization under uncertainty”. In: *Structural and Multidisciplinary Optimization* 25.2 (2003), pp. 99–116.

[73] Yaochu Jin. “A comprehensive survey of fitness approximation in evolutionary computation”. In: *Soft computing* 9.1 (2005), pp. 3–12.

[74] A. Kapahi and H. S. Udaykumar. “Dynamics of void collapse in shocked energetic materials: physics of void–void interactions”. In: *Shock Waves* 23.6 (Mar. 2013), pp. 537–558.

[75] A. K. Kapila et al. “Two-phase modeling of deflagration-to-detonation transition in granular materials: Reduced equations”. In: *Physics of Fluids* 13.10 (Oct. 2001), p. 3002.

[76] C.T. Kelley. *Solving nonlinear equations with Newton’s method*. SIAM, 2003.

[77] Shigeo Kida and StevenA. Orszag. “Energy and spectral dynamics in decaying compressible turbulence”. In: *Journal of Scientific Computing* 7.1 (1992), pp. 1–34.

[78] V. Kiselev, S. Kiselev, and V. Fomin. “Interaction of a Shock Wave with a Cloud of Particles of Finite Dimensions”. In: *J. Appl. Mech. and Tech. Phys.* 35 (1994), pp. 183–192.

[79] Jack PC Kleijnen. *Statistical tools for simulation practitioners*. Marcel Dekker, Inc., 1986.

[80] B.E. Launder and D.B. Spalding. “The numerical computation of turbulent flows”. In: *Computer Methods in Applied Mechanics and Engineering* 3.2 (1974), pp. 269 –289.

[81] G.A. Lawrence, F.K. Browand, and L.G. Redekopp. “The stability of a sheared density interface”. In: *Phys. Fluids* A3 (1991), p. 2360.

[82] J.A. Leonard, M.A. Kramer, and L.H. Ungar. “Using radial basis functions to approximate a function and its error bounds”. In: *IEEE Transactions on Neural Networks* 3.4 (July 1992), pp. 624–627.

[83] Randall J LeVeque. *Finite volume methods for hyperbolic problems*. Vol. 31. Cambridge university press, 2002.

[84] Hua Li and Shantanu S Mulay. *Meshless methods and their numerical properties*. CRC Press, 2013.

[85] Xu-Dong Liu, Stanley Osher, and Tony Chan. “Weighted Essentially Non-oscillatory Schemes”. In: *Journal of Computational Physics* 115.1 (1994), pp. 200 –212.

[86] Søren N Lophaven, Jacob Sø ndergaard, and Hans Bruun Nielsen. *Kriging Toolbox*. Tech. rep. 2002, pp. 1–28.

[87] Søren Nymand Lophaven, Hans Bruun Nielsen, and Jacob Søndergaard. *Aspects of the matlab toolbox DACE*. Tech. rep. Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2002.

[88] E Loth. “Compressibility and Rarefaction Effects on Drag of a Spherical Particle”. In: *AIAA Journal* 46.9 (2008), pp. 2219–2228.

[89] E. Loth. “Numerical approaches for motion of dispersed particles, droplets and bubbles”. In: *Progress in Energy and Combustion Science* 26.3 (2000), pp. 161 –223.

[90] C Lu et al. “Multi-scale modeling of shock interaction with a cloud of particles using an artificial neural network for model representation”. In: *Procedia IUTAM* 3 (2012), pp. 25–52.

[91] Christopher Lu. “ARTIFICIAL NEURAL NETWORK FOR BEHAVIOR LEARNING FROM MESO-SCALE SIMULATIONS, APPLICATION TO MULTI-SCALE MULTIMATERIAL FLOWS”. PhD thesis. The University of Iowa, 2010, pp. 1–111.

[92] O. Schilling M. Latini and W. S. Don. “Effects of order of WENO flux reconstruction and spatial resolution on reshocked two-dimensional Richtmyer-Meshkov instability”. In: *Journal of Computational Physics* 221 (2007), pp. 805–836.

[93] Xiang Ma and Nicholas Zabaras. “An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations”. In: *Journal of Computational Physics* 228.8 (2009), pp. 3084–3113.

[94] J Maijerink, G B Jacobs, and W S Don. “A WENO-Z based Eulerican-Lagrangian code for simulation of shocked flows laden with evaporating droplets”. In: 48th AIAA Aerospace Sciences Meeting, 2010.

[95] F. Mashayek and R.V.R. Pandya. “Analytical description of particle/droplet-laden turbulent flows”. In: *Progress in Energy and Combustion Science* 29 (2003), p. 329.

[96] G. Matheron. “Principles of Geostatics”. In: *Economic Geology* 58 (1963), pp. 1246–1266.

[97] Martin R. Maxey and James J. Riley. “Equation of motion for a small rigid sphere in a nonuniform flow”. In: *Physics of Fluids (1958-1988)* 26.4 (1983), pp. 883–889.

[98] R. Menikoff. “Hot spot formation from shock reflections”. In: *Shock Waves* 21.2 (Feb. 2011), pp. 141–148.

[99] F. R. Menter. “Two-equation eddy-viscosity turbulence models for engineering applications”. In: *AIAA Journal* 32.8 (1994), pp. 1598–1605.

[100] John Moody and Christian J Darken. “Fast Learning in Networks of Locally-Tuned Processing Units”. In: *Neural Computation* 1 (1989), pp. 281–294.

[101] Bryan S Morse et al. “Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions”. In: *ACM SIGGRAPH 2005 Courses*. ACM. 2005, p. 78.

[102] A.A. Mostafa and H.C. Mongia. “On the interaction of particles and turbulent fluid flow”. In: *International Journal of Heat and Mass Transfer* 31.10 (1988), pp. 2063 –2075.

[103] C. Narayanan, D. Lakehal, and G. Yadigaroglu. “Linear stability analysis of particle-laden mixing layers using Lagrangian particle tracking”. In: *Powder Tech.* 125 (2002), pp. 122–130.

[104] Roman Neruda and Petra Vidnerova. “Learning Errors by Radial Basis Function Neural Networks and Regularization Networks”. In: *International Journal of Grid and Distributed Computing* 1.2 (2009), pp. 49–58.

[105] Fuat Odar and Wallis S. Hamilton. “Forces on a sphere accelerating in a viscous fluid”. In: *Journal of Fluid Mechanics* 18 (02 Feb. 1964), pp. 302–314.

[106] M. Olim et al. “A General Attenuation Law of Moderate Planar Waves Propagating into Dusty Gases with Relatively High Loading Ratios of Solid Particles”. In: *Fluid Dynamics Research* 6 (1990), pp. 185–199.

[107] Wenxia Pan et al. “Predicting Dynamics and Rheology of Blood Flow: A Comparitive Study of Multiscale and Low-Dimensional Models of Red Blood Cells”. In: *Microvascular Research* 82 (2011).

[108] M Pandolfi and D D’Ambrosio. “Numerical Instabilities in Upwind Methods: Analysis and Cures for the ”Carbuncle” Phenomenon”. In: *J. Comput. Phys.* 166 (2001), pp. 271–301.

[109] R.V.R. Pandya and F. Mashayek. “Two-fluid large-eddy simulation approach for particle-laden turbulent flows”. In: *Int. J. Heat Mass Transfer* 6 (2002), p. 4753.

[110] J Park and I.W. Sandberg. “Universal Approximation using Radial-Basis-Function Networks”. In: *Neural Computation* 3 (1991), pp. 246–257.

[111] K M Peery and S T Imlay. “Blunt-body flow simulations”. In: *AIAA* 88.2904 (1988).

[112] J J Quirk. “A contribution to the great Riemann solver debate”. In: *Int. J. Numer. Methods Fluids* 18.6 (1994), pp. 555–574.

[113] S.A. Ragab and J.L. Wu. “Linear instabilities in two-dimensional compressible mixing layer”. In: *Physics of Fluids* A 1 (1989), p. 957.

[114] Jean-Christophe Robinet et al. “Shock wave instability and the carbuncle phenomenon: same intrinsic origin?” In: *Journal of Fluid Mechanics* 417 (2000), pp. 237–263.

[115] P L Roe. “Approximate Riemann solvers, parameter vectors, and difference schemes”. In: *J. Comput. Phys.* 43.2 (1981), pp. 357–372.

[116] Robert Sugden Rogallo. “Numerical experiments in homogeneous turbulence”. In: *NASA Tech. Mem. 81315* (1981).

[117] P. G. Saffman. “The lift on a small sphere in a slow shear flow”. In: *Journal of Fluid Mechanics* 22 (02 June 1965), pp. 385–400.

[118] P.G. Saffman. “On the stability of laminar flow of a dusty gas”. In: *J. Fluid Mech* 13 (1961), pp. 120–128.

[119] S K Sambasivan and H S UdayKumar. “Sharp interface simulations with Local Mesh Refinement for multi-material dynamics in strongly shocked flows”. In: *Computers and Fluids* 39 (2010), pp. 1456–1479.

[120] R Sanders, E Morano, and M Druguet. “Multidimensional Dissipation for Upwind Schemes: Stability and Applications to Gas Dynamics”. In: *J. Comput. Phys.* 145 (1998), pp. 511–537.

[121] S.B. Savage and D.J. Jeffrey. “The stress tensor in a granular flow at high shear rates”. In: *J. Fluid Mech* 110 (1981), pp. 255–272.

[122] L. Schiller and A. Naumann. “A drag coefficient correlation”. In: *VDI Zeitschrift* 77 (1933), pp. 318–320.

[123] P. J. Schmid and D.S. Henningson. *Stability and transition in shear flows*. Springer, 2001.

[124] O. Sen et al. “Evaluating Convergence and Accuracy of Techniques to Bridge Scales in a Multi-Scale Solver”. In: *J. of Comp. Phys., In Press* (2015).

[125] Giacomo Senatore, Sean Davis, and Gustaaf Jacobs. “The effect of non-uniform mass loading on the linear, temporal development of particle-laden shear layers”. In: *Physics of Fluids* 27.3, 033302 (2015), pp. –.

[126] Mirko Sgarbi, Valentina Colla, and Leonardo Maria Reyneri. “A comparison between weighted radial basis functions and wavelet networks.” In: *ESANN*. Citeseer. 1998, pp. 13–20.

[127] B. Shotorban. “Preliminary assessment of two-fluid model for direct numerical simulation of particle-laden flows”. In: *AIAA Journal* 49 (2011), pp. 438–443.

[128] Babak Shotorban et al. “An Eulerian model for particles nonisothermally carried by a compressible fluid”. In: *International Journal of Heat and Mass Transfer* 65.0 (2013), pp. 845 –854.

[129] C W Shu. *Essentially non-oscillatory and weighted essentially non-osillatory schemes for hyperbolic conservation laws*. ICASE Report 97-65. Hampton, VA: NASA Langley Research Center, 1997.

[130] C. W. Shu. “High order weighted essentially non-oscillatory schemes for convection dominated problems”. In: *SIAM Rev.* 51 (2009), pp. 82–126.

[131] T W Simpson et al. “Metamodels for Computer-based Engineering Design : Survey and recommendations”. In: *Engineering with Computers* 17 (2001), pp. 129–150.

[132] S A Smolyak. “Quadrature and interpolation formulas for tensor products of certain classes of functions”. In: *Dokl. Akad. Nauk SSSR* 4 (1963), pp. 240–243.

[133] Hyeongjin Song, K K Choi, and David Lamb. “A Study on Improving the Accuracy of Kriging Models by Using Correlation Model/Mean Structure Selection and Penalized Log-Likelihood Function”. In: *10th World Conference on Structural and Multidisciplinary Optimization*. 2013, pp. 1–10.

[134] Charles G Speziale, Sutanu Sarkar, and Thomas B Gatski. “Modelling the pressure–strain correlation of turbulence: an invariant dynamical systems approach”. In: *Journal of Fluid Mechanics* 227 (1991), pp. 245–272.

[135] M. Stein. “Large Sample Properties of Simulations Using Latin Hypercube Sampling”. In: *Technometrics* 29 (2 1987), pp. 143–151.

[136] G G Stokes. “On the effect of the internal friction of fluids on the motion of pendulums.” In: *Transactions of the Cambridge Philosophical Society* 9 (1851).

[137] J. P. Suarez, G. B. Jacobs, and W. S. Don. “A High-Order Dirac-Delta Regularization with Optimal Scaling in the Spectral Solution of One-Dimensional Singular Hyperbolic Conservation Laws”. In: *SIAM Journal of Scientific Computing* 36.4 (2014), A1831–A1849.

[138] C T Sun and R S Vaidya. “Prediction of Composite Properties from a Representative Volume Element”. In: *Composites Science and Technology* 56 (1996), pp. 171–179.

[139] M. Sun et al. “Unsteady Drag on a Sphere by Shock Wave Loading”. In: *Shock Waves* 14 (2004), pp. 3–9.

[140] K. Mike Tao. “A closer look at the radial basis function (RBF) networks”. In: *1993 Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers* (1993), pp. 401–405.

[141] G Tedeschi, H Gouin, and M Elena. “Motion of tracer particles in supersonic flows”. In: *Experiments in Fluids* 26.4 (1999), pp. 288–296.

[142] N. Thevand and E. Daniel. “Linear instability of dilute particle-laden compressible mixing layers”. In: *Phys. Fluids* 14 (2002), p. 392.

[143] Xiaoling Tong et al. “Multiphase Simulations of Blast-Soil Interactions”. In: *Proceedings of the ASME 2013 Fluids Engineering Division Summer Meeting*. FEDSM2013 16549. ASME. 2013.

[144] X.L. Tong and L.P. Wang. “Two-way coupled particle-laden mixing layer. Part 1: linear instability”. In: *International Journal of Multiphase Flow* 25 (1999), pp. 575–598.

[145] F Trochu. “A Contouring Program Based on Dual Kriging Interpolation”. In: *Engineering with Computers* 9 (1993), pp. 160–177.

[146] J.Y. Tu and C.A.J. Fletcher. “An improved model for particulate turbulence modulation in confined two-phase flows”. In: *International Communications in Heat and Mass Transfer* 21.6 (1994), pp. 775 –783.

[147] J. F. Unger and C. Könke. “Coupling of scales in a multiscale simulation using neural networks”. In: *Computers and Structures* 86.21 (2008), pp. 1994–2003.

[148] Jorg F. Unger and Stefan Eckardt. “Multiscale Modeling of Concrete”. In: *Archives of Computational Methods in Engineering* 18.3 (July 2011), pp. 341–393.

[149] Bram Van Leer. “Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov’s method”. In: *Journal of computational Physics* 32.1 (1979), pp. 101–136.

[150] V. Vand. “Viscosity of solutions and suspensions”. In: *J. Phy. Colloid Chem.* 52 (1948), pp. 277–321.

[151] Peter Vorobieff et al. “Vortex Formation in a Shock-Accelerated Gas Induced by Particle Seeding”. In: *Phys. Rev. Lett.* 106 (18 2011), p. 184503.

[152] Y. Wada and M. S. Liou. *A Flux Splitting Scheme with High-Resolution and Robustness for Discontinuities*. T. M. 106452. NASA, 1994.

[153] G Gary Wang and S Shan. “Review of metamodeling techniques in support of engineering design optimization”. In: *Journal of Mechanical Design* 129.4 (2007), pp. 370–380.

[154] Wei Wang and Jinghai Li. “Simulation of gas solid two-phase flow by a multi-scale CFD approach-of the EMMS model to the sub-grid level”. In: *Chemical Engineering Science* 62.1-2 (2007), pp. 208–231.

[155] F. Wen and J. Evans. “Linear instability of a two-layer flow with differential particle loading”. In: *Phys. Fluids* 6 (1994), p. 3893.

[156] Feng Wen and John Evans. “Effect of particle inertia on the instability of a particle-laden flow”. In: *Computers And Fluids* 25.7 (1996), pp. 667 –676.

[157] Holger Wendland. “Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree”. In: *Advances in Computational Mathematics* 4.1 (Dec. 1995), pp. 389–396.

[158] F.M. White. *Viscous Fluid Flow*. McGraw-Hill series in mechanical engineering. McGraw-Hill, 1991.

[159] Zongmin Wu. “Compactly supported positive definite radial functions”. In: *Advances in Computational Mathematics* 4.1 (1995), pp. 283–292.

[160] Dongbin Xiu and Jan S. Hesthaven. “High-Order Collocation Methods for Differential Equations with Random Inputs”. In: *Siam J. Sci. Comput.* 27.3 (2005), pp. 1118–1139.

[161] Y. Yang et al. “The influence of particles on the spatial stability of two-phase mixing layers”. In: *Phys. Fluids* A2 (1990), p. 1839.

[162] W. S. Don Z. Gao and Z. Li. “High Order Weighted Essentially Non-Oscillation Schemes for One-Dimensional Detonation Wave Simulations”. In: *Journal of Computational Mathematics* 29 (2011), pp. 623–638.

[163] F. Zhang et al. “Explosive dispersal of solid particles”. English. In: *Shock Waves* 10.6 (2001), pp. 431–443.

[164] Liang Zhao et al. “A Metamodeling Method Using Dynamic Kriging and Sequential Sampling”. In: *American Institute of Aeronautics and Astronautics*. 2010, pp. 1–18.