UNIVERSITY OF CALIFORNIA SAN DIEGO

A Stabilized Reproducing Kernel Formulation for Shock Modeling in Fluids and Fluid-Structure Interactive Systems

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

in

Structural Engineering

by

Tsung-Hui Huang

Committee in charge:

Professor Jiun-Shyan Chen, Chair
Professor Randolph E. Bank
Professor Yuri Bazilevs
Professor Veronica Eliasson
Professor Kenneth Loh

2020

The Dissertation of Tsung-Hui Huang is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____

_____

Chair

University of California San Diego

2020

DEDICATION

To my family and friends.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENT

I would like to express my sincere appreciation to my advisor, Professor J. S. Chen, for his careful guidance and constant encouragement throughout my Ph. D. studies. It is my honor to work with him, where his extensive knowledge and strong enthusiasm for pursuing novel ideas in research have inspired me and my career.

I acknowledge the committee members of my Ph.D. dissertation defense and candidacy exam: Professors Randolph Bank, Yuri Bazilevs, Veronica Eliasson, Kenneth Koh. I am very appreciative of their helpful comments and suggestions.

I also want to thank my friends at UCSD. In particular, to all my colleagues in the computational mechanic's group: Jonghyuk Baek, Dr. Frank Beckwith, Dr. Qizhi He, Xiaolong He, Shigeki Kaneko, Dr. Jacob Koester, Dr. Marco Pasetto, Karan Taneja, Dr. Haoyan Wei, Dr. Guohua Zhou, and many others. Their accompany and support are necessary pieces in my Ph. D. career. In addition, I would like to thank Dr. Michael Hillman for helpful discussions and comments during the conference.

Chapter 2, the literature review, in part, is a reprint of the material as it appears in following publications:

- "TH. Huang, H. Wei, JS. Chen and M. Hillman. RKPM2D: An Open-Source Implementation of Nodally Integrated Reproducing Kernel Particle Method for Solving Partial Differential Equations. Computational Particle Mechanics 7.2 (2020): 393-433. DOI: https://doi.org/10.1007/s40571-019-00272-x,",
- "TH. Huang, JS. Chen, H. Wei, MJ. Roth, JA. Sherburn, J. Bishop, M. Tupek, and E. Fang. A MUSCL-SCNI Approach for Meshfree Modeling of Shock Waves in Fluids.

Computational Particle Mechanics 7.2 (2020): 329-350. DOI: https://doi.org/10.1007/s40571-019-00248-x.", "TH. Huang, JS. Chen, and MR. Tupek. An Immersed Variational Multiscale RKPM Formulation. Part I - Heterogeneous Materials. In Preparation,"

- "TH. Huang, JS. Chen, and MR. Tupek. An Immersed Variational Multiscale RKPM Formulation. Part II – Fluid-Structure Interaction. In Preparation."

The dissertation author was the primary investigator and author of this paper.

Chapter 3, in part, is a reprint of the material as it appears in "TH. Huang, H. Wei, JS. Chen and M. Hillman. RKPM2D: An Open-Source Implementation of Nodally Integrated Reproducing Kernel Particle Method for Solving Partial Differential Equations. Computational Particle Mechanics 7.2 (2020): 393-433. DOI: https://doi.org/10.1007/s40571-019-00272-x" The dissertation author was the primary investigator and author of this paper.

Chapter 4, in part, is a reprint of the material as it appears in "TH. Huang, JS. Chen, H. Wei, MJ. Roth, JA. Sherburn, J. Bishop, M. Tupek, and E. Fang. A MUSCL-SCNI Approach for Meshfree Modeling of Shock Waves in Fluids. Computational Particle Mechanics 7.2 (2020): 329-350. DOI: https://doi.org/10.1007/s40571-019-00248-x.". The dissertation author was the primary investigator and author of this paper.

Chapter 5, in part, is being prepared for submission for publication of the material as it may appear in "TH. Huang, JS. Chen and MR. Tupek. An Immersed Variational Multiscale RKPM Formulation. Part I - Heterogeneous Materials. 2020". The dissertation author was the primary investigator and author of this paper.

Chapter 6 and 7, in part, is being prepared for submitted for publication of the material as it may appear in "TH. Huang, JS. Chen and MR. Tupek. An Immersed Variational Multiscale RKPM

Formulation. Part II - Fluid-Structural Interaction. 2020". The dissertation author was the primary investigator and author of this paper.

Tsung-Hui Huang

La Jolla, CA

June 2020

VITA

2012 Bachelor in Mechanical Engineering, National Taiwan University, Taiwan

2014 Master in Mechanical Engineering, University of Minnesota Twin Cities, USA

2020 Ph. D. in Structural Engineering, University of California San Diego, USA

Journal Publications

1) TH. Huang, JS. Chen, and MR. Tupek. "An Immersed Variational Multiscale RKPM Formulation. Part II – Fluid-Structure Interaction." In Preparation.

2) TH. Huang, JS. Chen, and MR. Tupek. "An Immersed Variational Multiscale RKPM Formulation. Part I - Heterogeneous Materials." In Preparation.

3) A. Neofytou, R. Picelli, TH. Huang, JS. Chen, and H. A. Kim. "Level Set topology Optimization for Design-Dependent Pressure Loads Using the Reproducing Kernel Particle Method." Structural and Multidisciplinary Optimization (2020). In press.

4) TH. Huang, H. Wei, JS. Chen, and M. Hillman. "RKPM2D: An Open-Source Implementation of Nodally Integrated Reproducing Kernel Particle Method for Solving Partial Differential Equations." Computational Particle Mechanics 7.2 (2020): 393-433. DOI: https://doi.org/10.1007/s40571-019-00272-x.

5) TH. Huang, JS. Chen, H. Wei, MJ. Roth, JA. Sherburn, J. Bishop, M. Tupek, and E. Fang. "A MUSCL-SCNI Approach for Meshfree Modeling of Shock Waves in Fluids." Computational Particle Mechanics 7.2 (2020): 329-350. DOI: https://doi.org/10.1007/s40571-019-00248-x.

6) TH. Huang, TH. Huang, YS. Lin, CH. Chang, SW. Chang, CS. Chen. " A Time Integration Method for Phase-Field Modeling." Multiscale Science and Engineering. (2019): 1-14.

7) TH. Huang, CS. Chen, and SW. Chang. "Microcrack Patterns Control the Mechanical Strength in the Biocomposites." Materials & Design. 140 (2018): 505-515

8) TH. Huang, TH. Huang, YS. Lin, CH. Chang, PY. Chen, SW. Chang, CS. Chen. "Phase-Field Modeling of Microstructural Evolution by Freeze Casting Process." Advanced Engineering Materials. 20.3 (2017): 1700343

Conference Presentations

1) TH. Huang, H. Wei, JS. Chen. RKPM Shock Algorithms for Modeling Structural Fragmentation in Air Blast. Abstract and Oral Presentation. Asian Pacific Congress on Computational Mechanics, Taipei, Taiwan, December 17th-21st, 2019.

2) TH. Huang, H. Wei, JS. Chen. An RKPM-based Shock Algorithm for Modeling Immersed Fluid-Structure Interactions in Blast Events. Abstract and Oral Presentation. ASME 2019 International Mechanical Engineering Congress & Exposition Chicago, IL, November 10th-14th, 2019.

3) TH. Huang, H. Wei, JS. Chen. Eulerian Reproducing Particle Method for Shock Modeling. Oral Presentation. Finite Elements in Flow Problems, Chicago, IL, April 1st-3rd, 2019 (Travel Award Acceptant).

4) TH. Huang, JS. Chen. Eulerian Reproducing Particle Method for Shock Modeling. Abstract and Oral Presentation. Meshfree and Particle Methods: Applications and Theory, Santa Fe, NM, September 9-12th, 2018 (Travel Award Acceptant).

5) TH. Huang, JS. Chen. Eulerian Reproducing Particle Method for Shock Modeling. Abstract and Oral Presentation. The 13th World Congress in Computational Mechanics, New York City, CA, July 22-27th, 2018.

6) TH. Huang, T-H. Huang, and C-S. Chen. FFT-based Method for Characterization and Analysis of Microstructure from Freeze-Casting Process. Abstract and Oral Presentation. The 12th World Congress on Computational Mechanics, Seoul, Korea, July 24-29th, 2016.

7) TH. Huang, T-H. Huang, and C-S. Chen. Phase-field Model for Dendritic Solidification in Freeze Casting. Abstract and Oral Presentation. The 39th National Conference on Theoretical and Applied Mechanics, Taipei, Taiwan, November 20-21th, 2015.

8) TH. Huang, T-H. Huang, and C-S. Chen. Stability and Accuracy of Differential-Algebraic Phase-Field Equations. Abstract and Oral Presentation. First Computational Mechanics Conference in Taiwan, Taipei, Taiwan, October 22-23th, 2015.

ABSTRACT OF THE DISSERTATION

A Stabilized Reproducing Kernel Formulation for Shock Modeling in Fluids and Fluid-Structure Interactive Systems

by

Tsung-Hui Huang

Doctor of Philosophy in Structural Engineering
University of California San Diego, 2020
Professor Jiun-Shyan Chen, Chair

In the extreme event such as air-blast or explosion, strong shocks lead to severe damage and fragmentation in structures. Despite the considerable effort made in recent years, reliable numerical prediction of fragmentation processes in materials and solids under blast loading or shock wave remains challenging. The conventional mesh-based methods (e.g., finite element method (FEM)) are ineffective due to large deformation-induced mesh distortion issues and exhibit non-convergent solutions in fracture problems. The meshfree method, such as reproducing kernel particle method (RKPM), naturally avoids computational difficulties associated with low-quality meshes, allows efficient adaptive refinement, and provides flexible control of smoothness and locality in numerical approximations.

The objective of this work is to develop a computational framework for effective modeling of shock dynamics in fluids and fluid-structure interactive systems. In this work, a stabilized RKPM framework for modeling shock waves in fluids is first developed. To capture essential shock physics and to avoid numerical oscillations, a Riemann-enriched smoothed flux divergence with an oscillation limiter is introduced under the stabilized conforming nodal integration (SCNI) framework. Besides, a flux splitting approach is employed to avoid advection-induced instabilities in fluid modeling, and the Monotonic Upstream-Centered Scheme for Conservation Laws (MUSCL)-type oscillation limiter is employed to avoid over and undershooting of the numerical solution at shock front and to capture moving discontinuities with minimal diffusion. The proposed methods, termed MUSCL-SCNI, have been applied to the shock tube problem, compressible flow with vortex, and explosive detonation.

Next, an immersed RKPM formulation is developed for an effective body-unfitted spatial discretization of subdomains in heterogeneous materials and fluid structure interaction (FSI) problems involving complex geometries. RKPM naturally avoids computational challenges associated with low-quality meshes, allows efficient adaptive refinement, and provides flexible control of continuity and locality in the numerical approximations. A variational multiscale immersed method (VMIM) is proposed, where the solution fields are decoupled into coarse- and fine-scales, and the fine-scale solution represents the residual of the coarse-scale equations. Under VMIM, the coupling between different subdomains is done through a volumetric constraint, and the embedment of the fine-scale solution into coarse-scale equations yields a stabilized Galerkin formulation with enhanced stability and accuracy. The proposed method is first applied to modeling heterogeneous materials. It is then further extended to shock wave modeling in the FSI

systems, where the meshfree algorithm based on MUSCL-SCNI is employed for ensured stability. Finally, the proposed VMIM is applied to air-blast events simulations.

Chapter 1 Introduction

In this chapter, the motivation for the present research work is given in Chapter 1.1, the objectives of the research are introduced in Chapter 1.2, and the outline of this dissertation is given in Chapter 1.3.

1.1 Motivation

In extreme events such as air blast, impact and explosive welding [1], the structure under impact loading [2] and high explosive dynamics, the structure usually involves large deformation, results in failure, damage, and fragmentation due to strong hydrodynamical effect. In this type of extreme event research, experiments are often limited due to the significant deformation response, high rate, short-time scale that are inherently present in these problems. On the other hand, the computations enable us to extend the capabilities beyond experiments. However, the strong shock waves lead to strong gradients and jump in the velocity, pressure, and mass density fields, which are challenging for numerical methods to capture accurately. The satisfaction of essential shock physics is important and yet difficult to the conventional numerical method. In addition to that, numerical methods suited for modeling these events must also be able to deal with rough solutions involving fragmentation, sever material deformation and material instability, and multi-body

contacts due to countless post-shock debris. Traditional numerical methods, like finite element method (FEM), are regarded as not suitable since FEM runs into problems due to mesh distortion and entanglement [3, 4, 5] and corresponding poor solution accuracy, stability, and solution divergence. Techniques such as element erosion are employed to avoid mesh distortion in fragmentation problems, but they lead to the violation of mass conservation and artificial material degradation which are inconsistent with the physics.

On the other hand, meshfree methods, such as reproducing kernel particle method (RKPM) [6, 7, 8] have shown promising features in modeling these events, as they naturally avoid computational difficulties associated with low-quality meshes, allow efficient local *hp*-adaptive refinement for solutions with high gradients, and provide flexible control of smoothness and locality in numerical approximations. In the past 20 years, the applications of meshfree methods on extreme events have drawn success in recent years, including modeling shock wave propagation in solids and structure. Nevertheless, current research on the development of robust and accurate meshfree methods for shock modeling in fluid and further fluid-structure interaction is still rare and deserves investigation.

Furthermore, although structure failure subject to strong air shock waves or explosive waves has been of interest to the engineering community for a long time, it remains tedious to capture a broad set of evolving structural fractures and debris using conventional approaches. Explicit tracking and modeling of fluid-structure interfaces for fragmented structures result in non-trivial methods, such as modification of conforming mesh topology, neighbor list updating, removal, or addition to the extra degrees of freedom and others. These methods increase the computational difficulties and efficiency, especially when countless fragmented bodies associated with fluid flows need to be considered. On the other hand, implicit representation of fluid-structure

interface enables a flexible set of discretizations that avoids the aforementioned issues, where the solid is embedded or immersed in the fluid, and the compatibility condition can be imposed through different methods such as penalty method, Lagrange multiplier method. Nonetheless, these often face numerical issues such as degraded accuracy due to Gibbs phenomenon near material interfaces, violation of Ladyzhenskaya–Babuška–Brezzi (LBB) condition, and issues of stabilization parameter sensitivity. Therefore, the development of an efficient and robust numerical modeling approach for immersed simulation of the fluid-structure coupled problem is essential.

1.2 Objective

The objective of this work is to develop a meshfree method for the effective numerical simulation of shock wave modeling in fluid and fluid-structure interaction problems. The specific developments are summarized as follows:

1. Develop a stable reproducing kernel shock algorithm in modeling fluid problems. In order to capture the essential shock physics (including the Rankine-Hugoniot jump conditions and the entropy condition) in fluids, local Riemann enrichment is introduced under the stabilized conforming nodal integration (SCNI) framework. Meanwhile, numerical instabilities associated with the advection flux are eliminated by adopting a modified upwind scheme. To further enhance accuracy, a MUSCL-type method is introduced in conjunction with an oscillation limiter to avoid the Gibbs phenomenon and ensure monotonic piecewise linear reconstruction in the smooth region. The present meshfree formulation is free from tunable artificial parameters and is capable of capturing shock and rarefaction waves without over/undershoots. The effectiveness of the proposed meshfree method is demonstrated by solving several hydro-mechanically coupled problems. Several numerical examples are

presented to examine the effectiveness of the proposed method for shock wave modeling in fluids.

2. Develop an effective immersed formulation to model the heterogeneous material problem that enables flexible non-body-fitted discretizations and quadrature rules for each material subdomains. The compatibility condition at the material interface is imposed through a volumetric constraint, where methods such as penalty methods or Lagrange multiplier methods can be employed. However, the volumetric constraint may results in low order accuracy near the interface. In order to address these issues, a novel immersed method based on a variational multiscale approach (VMS) [9] is developed, where the introduced fine-scale solution is solved analytically, playing a role of representing the residual (error) from the coarse-scale equations. The fine-scale equations can be solved by the residual free bubble method or an approximation via the averaging of the fine-scale Green's function. The derived method, called the variational multiscale immersed method (VMIM), not only enhances the accuracy due to the introduction of fine-scale features, but also leads to a residual-based stabilization consistent with the theory shown in literature [10]. The VMIM precludes the numerical oscillation and enhance the accuracy compared to conventional immersed approaches, under comparable efficiency. Its capability is demonstrated in solving several numerical examples in linear elasticity, diffusion problem, and microstructure material modeling.

3. Develop an immersed method for modeling shock wave in fluid-structure interaction problems, where the background fluid domain is modeled by the Eulerian RKPM formulation,

and the foreground solid domain employs the Lagrangian or semi-Lagrangian RKPM formulation. To couple the fluid-structure interaction, the aforementioned VMIM formulation is employed, which enables convenient nodal integration schemes in the solid and fluid domain. The fine-scale solution represents the residuals from the fluid momentum equation. The MUSCL-type enhancement, along with SCNI formulation, is employed to model the fluid flow. The setting of Lagrangian or semi-Lagrangian RKPM with stabilized conforming or non-conforming nodal integration (SCNI/SNNI) in solid can model solid structure under large deformation and fragmentation due to shock waves through air straightforwardly. The derived method employs nodal integration for both fluid and solid domains, that is suitable for various types of meshfree particle method. The effectiveness of the proposed method is demonstrated in solving several numerical examples and applying to a blast event modeling.

1.3 Outline

The organization of this dissertation is given as follows: In chapter 2, an overview of mesh-based and meshfree methods for modeling shock wave propagation as well as fluid-structure interaction is given, with discussions on the advantages and limitations of conventional and current computational methods. In Chapter 3, the basic concept and equations are introduced, including the reproducing kernel approximation; Lagrangian and Eulerian kernel function; quadrature rules for meshfree method; Riemann problems and different Riemann solvers. In Chapter 4, the Eulerian RKPM formulation for modeling shocks in fluids is presented. In Chapter 5, the variational multiscale immersed method (VMIM) is developed for modeling heterogeneous material problems. Furthermore, in Chapter 6, the proposed variational multiscale immerse formulation is derived for

a coupled fluid-structure interaction framework, and the MUSCL-type shock enhancement in Chapter 4 is employed to model shock wave in fluid-structure interaction problems. The application of the developed method in modeling a blast event, concrete under TNT explosion, is given in Chapter 7. Finally, conclusions and discussions on future research directions are given in Chapter 8.

Chapter 2 Literature Review

In this chapter, the essential literatures are reviewed including the essential equations developed to model shock wave in fluids and fluid-structure interaction. This chapter is organized as follows. The overview of numerical methods for blast event, shock wave modeling by mesh-based and meshfree methods are reviewed in Chapter 2.1. The numerical methods for modeling heterogeneous material problems are reviewed in Chapter 2.2. The methods for modeling fluid-structure interaction are reviewed in Chapter 2.3.

2.1 Overview of Numerical Methods for Shock Modeling

A shock wave is a propagating disturbance that moves faster than the local speed of sound in the medium. Like an ordinary wave, a shock wave carries energy and can propagate through a medium, but it is characterized by an abrupt change (or called moving discontinuities) in pressure, temperature, and density of the medium. The most straightforward ways for numerical methods to handle the moving discontinuities is to employ the von Neumann-Richtmyer artificial viscosity [11, 12], or well-known Lax-Friedrichs scheme. However, the drawbacks are resulting strong diffusion or numerical parameters relying in empirical experience, and in general they cannot

always control the Gibbs oscillation properly. Therefore, advanced shock capturing algorithm in mesh-based or meshfree methods has been developed and is introduced in the following.

2.1.1 Mesh-Based Methods

Traditional mesh-based methods, such as finite volume (FV) and finite element (FE) methods, have been developed for years in shock modeling. Godunov's scheme [13] with various Riemann-solvers [14, 15, 16, 17] is popular in FV, where a piecewise constant volume integration is employed and local Riemann problems are solved on the volume boundary, so that the jump and entropy conditions are imposed. To address the low-order accuracy issue of the Godunov's scheme, van Leer [18, 19] introduced the Monotonic Upwind Scheme for Conservation Laws (MUSCL), which allows the use of high-order approximations with flux/slope limiters to achieve the total variation diminishing (TVD) feature. Accordingly, various flux and slope limiters were developed [20, 21] such that a piecewise constant approximation is employed only at the jump region. Different from MUSCL-type enhancement, Harten and Osher [22], Harten et al. [23], and Shu and Osher [24] developed the essentially non-oscillatory (ENO) and weighted essential non-oscillatory (WENO) schemes with TVD feature, where high-order approximations are constructed by adaptively weighting low-order candidate stencils. The weight associated with each candidate stencil is chosen to achieve high-order approximations in smooth regions, while the stencils covering discontinuities are assigned nearly zero weights. In the field of FE, Christie et al. [25] provided an upwind FE scheme, where a Petrov-Galerkin formulation using biased weighting functions was introduced in conjunction with linear and quadratic, C^0 continuous basis functions. It is shown that, with the proper selection of biasing, an oscillation-free solution is obtained in the shock problem. Although the upwinding techniques reduced solution oscillations, a major

disadvantage was found to be the excessive crosswind diffusion in multiple dimensional problems [26]. Hughes et al. [26, 27, 28] developed the streamline upwinded Petrov-Galerkin (SUPG) formulation to correct this cross-wind diffusion effect, such that the additive diffusion (introduced to the variational equation by a modified test function) is only active in the direction of the streamline along the flow direction. Even though SUPG is effective in reducing oscillations associated with the cross-wind diffusion, it does not preclude overshooting and undershooting in sharp layers [29]. Therefore, methods with stronger stability properties (e.g., 'monotonicity') is required. With that in mind, the discontinuity capture scheme was developed by Hughes et al. [29], Hughes and Tezduyar [30], and Tezduyar et al. [31, 32], where an additional "discontinuity capturing term" was introduced to the weight function enrichment. This additional term acts only in the solution gradient direction to enhance the resolution at strong shocks.

2.1.2 Meshfree-Based Methods

Comparing to abovementioned shock modeling in mesh-based method, meshfree methods [6, 7, 8] have shown promising features, as they naturally avoid computational difficulties associated with low-quality meshes, allow efficient local *hp*-adaptive refinement for solutions with high gradients, and provide flexible control of smoothness and locality in numerical approximations. In the early development of meshfree methods for shock modeling, the von Neumann-Richtmyer artificial viscosity [11, 12] approach was employed in the Smoothed Particle Hydrodynamics (SPH) formulation by Monaghan [33, 34], where the viscosity term was modified to follow the characteristic speed of the flow. However, the use of artificial viscosity requires tunable parameters to control the amount of numerical dissipation, which usually over-smears shock fronts, and thus the shock physics cannot be correctly represented. Later, Godunov's scheme [13] was

implemented in SPH [35] and other meshfree methods, such as finite volume particle method [36] and finite point method [37], where the gradient and Laplacian operators were modified to derive an edge-based or pair-wise local Riemann-solver. In these methods, the jump and entropy conditions of shock physics are captured by solving local Riemann problems using piecewise constant approximations. The Reproducing Kernel Particle Method (RKPM) [6], proposed as a correction to SPH to satisfy arbitrary order consistency in the numerical approximations, has also been developed recently for shock modeling. The Riemann-SCNI framework [38, 39] has been proposed by Roth et al. for modeling shock waves in nonlinear solids under the framework of stabilized conforming nodal integration (SCNI) [40], where a smoothed flux divergence operator for the local Riemann solutions was introduced at the nodal gradient smoothing cell boundaries, which naturally captured shock physics and avoids numerical instabilities. Recently, a non-ordinary state-based peridynamics formulation [41] is developed for modeling strong shocks in solids, where the Godunov scheme is introduced by embedding the Riemann solution into the force state. However, the employment of Godunov scheme results in first-order accuracy due to the piecewise constant approximation. Despite of Riemann-Enrichment-type approach, a residual-based shock capturing technique based on immersed meshfree framework [42, 43, 44] was developed which maintains the desired shock structure without disturbing smooth regions of the solution. However, a variety of related research questions remain open for this approach.

Regarding the low-order accuracy issue associated with the Godunov's scheme, van Leer [19, 18] introduced the Monotonic Upwind Scheme for Conservation Laws (MUSCL), which allows the use of high-order approximations with flux/slope limiters to achieve the total variation diminishing (TVD) feature. Accordingly, various flux and slope limiters were developed [21, 20] such that the piecewise constant approximation is introduced only at the jump region. Different

from MUSCL-type enhancement, Harten and Osher [22], Harten et al. [23], and Shu and Osher [24] developed the essentially non-oscillatory (ENO) and weighted essential non-oscillatory (WENO) schemes with TVD feature, where high-order approximations are constructed by adaptively weighting low-order candidate stencils. The weight associated with each candidate stencil is chosen to achieve high-order approximations in smooth regions, while the stencils covering discontinuities are assigned nearly zero weights. For meshfree methods, the WENO-type approach has been introduced in SPH [45] and RKPM [46, 47] to reconstruct high-order approximation through a moving least square procedure. However, for multi-dimensional applications, the selection of candidate stencils in these methods becomes tedious. On the other hand, when applied to SPH [48] and FPM [49], the MUSCL-type enhancement demonstrates good performance, with lower computational cost compared to the WENO-type schemes [50].

To ensure numerical stability in modeling shock waves in fluids, proper discretization of the advection term with velocity jumps needs to be considered. In the conventional Godunov's scheme, the Riemann solver is applied to the total flux consisting of both pressure and advection parts. As a result, the shock fronts are over-smeared when certain Riemann solvers with high dissipations are adopted, while less dissipative Riemann solvers, such as Roe solver [15], can produce spurious instabilities [51, 52] in modeling high Mach number flow. These numerical instabilities can be triggered by perturbations in initial conditions or nodal discretizations and can be rapidly amplified in the shock region, leading to severe numerical oscillations, known as the carbuncle effect [52]. To address these issues, the flux vector splitting (FVS) approach was proposed [53] to split the total flux into advection and pressure parts, which enables different schemes to be applied to each part, respectively. Depending on how the total flux is split, different FVS methods have been proposed. Liou and Steffen [54] split the pressure in the momentum equation and leave all the

other terms to the advection part. The Z-B splitting proposed by Zha and Bilgen [55] splits the total flux of all conservation equations (mass, momentum, and energy) into pressure and advection parts. The method proposed by Toro and Vazquez [56] is similar to the Z-B splitting, except that the internal energy term is considered in the pressure part. It is noted that, in the Z-B splitting, the advection part's characteristic wave speed is equal to the flow velocity, and the pressure part's characteristic wave speed is proportional to the acoustic speed [55], which is an attractive feature of this scheme as it agrees well with shock wave physics, i.e., the shock waves propagate with the sound speed while the advection of the material follows the flow velocity.

2.2 Overview of Numerical Methods in Immersed and Embedded Approach for Heterogeneous Material Problem

Heterogeneous material problem, such as the linear elastic analysis of composite solid [57], bi-material heat conduction problem [58], bone structure in biological systems [59] can be modeled by partial differential equations (PDEs) containing weak discontinuity at material interface [60], where the primary solution field (i.e. displacement, temperature, … etc) is C^0 smoothness near material interface while its gradient field is discontinuous (C^{-1}). Satisfaction of interfacial Dirichlet conditions is crucial yet non-trivial in numerical method in mesh-based and meshfree method shown hereafter.

2.2.1 Mesh-Based Methods

The convention finite element method (FEM) employs a geometric fitted (or called body-fitted) mesh across the material interface with standard FE shape functions to approximate solution at the different material domain. Hence, the material with distinct material properties naturally lies

in elements at each side of the material interface, and therefore the basis function is independent of interfacial Dirichlet condition. However, the construction of such fitted meshes is time-consuming, and non-trivial when material involves complex geometry in the multidimensional case. As a result, methods can handle unfitted meshes is attractive in heterogeneous material analysis. The body-unfitted meshes method by employing boundary integral on the material interface for interfacial Dirichlet condition (termed interfacial constraint hereafter) under a different embedded domain framework, have been studied in the field of FEM. Mortar FEM [61] coupled unfitted meshes across the interface by enforcing interfacial constraint with Lagrange multipliers. Unfortunately, such an approach is suffered from the violation of the LBB (Ladyzhenskaya–Babuška–Brezzi, or called inf-sup) conditions [62, 63] if the function space of Lagrange multipliers is not properly selected. The violation of the LBB condition may lead to instabilities and oscillations in the interfacial traction fields [64]. Nitsche's method by Stenberg [65] and Hansbo [66] was therefore introduced where the Lagrange multipliers are replaced by the tractions on the interface, and a least-square type stabilization is added controlled by a penalty parameter based on material properties. However, the determination of the penalty parameter is non-trivial, where a local eigenvalue analysis is required to identify the stability constant, which is not an efficient strategy [67]. In order to address this issue, bubble stabilized formulation under the framework of extended finite element method (XFEM) is applied [68, 69] by Mourad and Dolbow et al., where the displacement field is decomposed into coarse- and fine-scale field following the variational multiscale framework [9, 70], and the fine-scale field is enriched by element bubble function, similar to the residual-free bubble approach [71]. The static condensation of fine-scale degrees of freedom (DOF) yielded a form similar to standard

Nitsche's formulation, and the least-square type stabilization term is controlled by the given enrichment function. The interfacial constraint imposed by bubble-stabilized XFEM [68] and Nitsche's method [66] has been shown to preserve optimal convergence in L2 and energy norms for linear elasticity problems, although it relied on high order quadrature over boundary integral at the material interface, thereby coming with a high computational cost, or resulting in difficulty when the interface geometry is complex or material under crack/fragmentation. In addition to that, the employment of the interfacial constraint under the embedded framework usually causes an ill-conditioned system due to the near-zero internal energy at the overlapping domain (or called fictitious domain in some works of literature), in which additional removal of degrees of freedom (DOF) is necessary.

The alternative to interfacial constraint is to employ the volumetric constraint (enforcement of volumetric Dirichlet jump condition on the overlapping domain), where such approach has been exploited under different versions of immersed frameworks, especially in the field of fluid-structure interaction. Similar to the interfacial constraint approach, penalty method [72] and Lagrange multiplier method [73] were implemented to enforce such constraint. Glowinski et al. [74] developed the distributed Lagrange multiplier/fictitious domain method for particulate flow, where the kinematics of rigid body motion of particles follows background fluid flow through a perturbed Lagrange multiplier. Zhang et al. [75, 76] developed the immersed finite element method (IFEM) for fluid-structure interaction problems, where the Dirichlet condition of solid/fluid velocity is imposed weakly through a reproduced function by reproducing kernel (RK) method. The volumetric constraint enforcement not only avoids issues of ill-conditioning but also avoids tedious boundary integral along material interfacial, although it may still have the shortcomings of the low order accuracy near the interface. Due to the scarce of literature on the application of

volumetric constraint for solid heterogeneous material problems, the accuracy and robustness of immersed approach with volumetric constraint for solids need to be investigated.

2.2.2 Meshfree-Based Methods

Comparing to FEM and the other mesh-based methods, the meshfree method shows more promising numerical features for engineering problems. The approximation in meshfree methods, such as reproducing kernel (RK) [3, 6], is constructed based on a set of scattered points without any mesh connectivity, and thus the strong tie between the quality of the discretization is relaxed and therefore mesh distortion or mesh entanglement issues are avoided [3, 4, 5]. Also, the flexibility of control of local smoothness and consistency can be easily achieved, which is suitable for a heterogeneous material problem requiring different smoothness in each subdomain. The application of the meshfree method for the solid heterogeneous material problem had also been studied extensively, such as element free Galerkin (EFG) [77, 78, 79], meshless local Petrov-Galerkin (MLPG) [58], moving least-square (MLS) [80, 81, 82], reproducing kernel particle method (RKPM) [83, 84, 85, 86, 87], and Smoothed Particle Galerkin (SPG) [88], to name a few. Due to lack of Kronecker delta properties, it is not trivial for meshfree method to impose the boundary or interfacial constraint, and therefore different techniques were developed to enforce the constraint strongly or weakly. Meshfree approximation with local enrichment across the material interface [78, 81, 82, 87] was constructed to enable the direct impose of Dirichlet condition, but it requires additional DOF employed on the interface for the enrichment. Wang et al. [84] used a discontinuous Galerkin formulation based on an incompatible patch-based RK approximation, such that the additional DOF is not required since the continuities of

displacement were imposed weakly in the variational equation. Wu et al. [85, 88] introduced a novel meshfree discretization strategy for a given composite material in an immersed setting and employing the singular kernel method [89] for approximation such that the point-wise continuity across the material interface was naturally built under the proposed unique discretization setting. Koester et al. [86] employ conforming RK shape function constructed on the boundary-fitted local triangulation to avoid meshfree modeling involving complex geometry, and the Kronecker delta was recovered, allowing the direct enforcement of interfacial constraint. Although meshfree methods have demonstrated stable and accurate results in modeling solid heterogeneous material problems, most of the aforementioned works focused on using an embedded framework (employing interfacial constraint), and therefore the meshfree method under immersed framework (employing volumetric constraint) is attractive to avoid the drawbacks in embedded approach.

2.3 Challenges in Shock Wave Modeling with Fluid-Structure Interaction

2.3.1 Overview of Numerical Methods in FSI

In modelling solid structure under blast loadings, fluid-structure interaction (FSI) severs as an important role in capturing the shock wave propagation from the air to the solid, as well as the solid fragments moving in the fluid such as air. To accommodate the complicated motion of fluid–structure (F-S) interface and satisfy the kinematic/traction equilibrium on the F-S interface (named F-S constraint hereafter), adaptive re-meshing and the arbitrary Lagrangian Eulerian (ALE) techniques [90] are commonly adopted. Nevertheless, mesh update or remeshing algorithms are time consuming, as discussed by Johnson and Tezduyar [91].

In contrast to re-meshing and ALE methods, the immersed type approach is more suitable in modeling blast events due to the potential material fracture and fragmentation. In the immersed approach, the meshes of the solid and fluid domain are not required to be conforming to the material interface, and the structure model is immersed in the surrounding fluid mesh. As such, it is easy for immersed methods to handle topological changes arising from structural contacts to fragmentation. In a variety of immersed type methods, Glowinski et al. [92, 74] first developed the fictitious domain method for fluid-particle interaction, where the Lagrange multiplier was employed to represent the additional body force per unit volume needed to maintain the F-S interfacial condition along the particle boundary. Similarly, Baaijens [93] employed the mortar element method for the imposition of F-S interface condition, which circumvented the judicious choice of base functions of the Lagrange multiplier. Wang and Belytschko [94, 95] enforced the F-S constraint by Nitches' method in a discontinuous Galerkin formulation. In this method, the jumps in velocity and traction on F-S interface can be naturally captured, and the fictious domain is combined with the background fluid through a level-set function. Different from the fictitious domain method which imposes the F-S constraint through boundary integration along the F-S interface, Peskin [96] introduced the immersed boundary method for fluid-elastic fiber interaction by using a delta function to interpolate and exert the FSI coupling force. Following Peskin's approach, Wang and Liu, [97] , Liu et al. [98] and Zhang et al. [99, 75] developed the immersed finite element method (IFEM), where the F-S interfacial constraints were extended from the interface to the interior of the solid domain (or namely fictitious domain) and the RKPM function was employed to replace the delta function to interpolate the FSI force. Because of the higher-order smoothness in the RKPM function, the accuracy was increased in the coupling procedures between fluid and solid domains [97]. In IFEM, since the F-S constraint was performed within the

solid domain instead of on F-S interface, it avoided the tedious F-S interface boundary integral under the situation of countless new interfaces due to structure failure, Therefore IFEM, and other similar methods such as the material point method by Sulsky et al [100], and the direct forcing method by Uhlmann [101], are suitable for modeling blast events. Recently, Bazilevs et al. [43, 44] employ the idea of IFEM to couple isogeometric approach for fluid and meshfree RKPM for solid to model structure failure under air blast, which is promising in modeling structures under large inelastic deformations and fragmentation scenarios due to air-blast or high explosive.

2.3.2 Failure Modeling for Solid Structures under Shock Wave or Blast Loading

Regarding the issue of material fragmentation due to strong shocks, traditional finite element methods employ element erosion [102] or element death algorithm to remove elements undergoing very large deformations, which violates the mass conservation during the simulation. On the other hand, meshfree methods are suitable for modeling solid structures under blast loading with large inelastic deformations, excessively contact, and material fragment, as the mesh distortion issues are avoided.

Fragment-impact problems have been modeled by using meshfree methods such as SPH by Randels et al. [103] and Rabczuk and Eibl [104], RKPM by Chen et al. [105, 106, 107], and Peridynamics by Macek and Silling [108], Lai et al. [109] and Ren et al. [110]. The multibody contact problem for fragmented structure can be managed by various contact algorithms. In extreme deformation problems with material separation, contact surfaces are unknown, as they are parts of the solution. Hence, the conventional contact algorithms (such as master-slave contact algorithm for SPH/FE coupling by Attaway et al. [111] and Jonhson [112]), are ineffective because

these methods require all possible contact surfaces be defined a priori. Kernel contact (KC) algorithms by Chi et al. [106] and Guan et al. [113] approximate the contact conditions without relying on predefined contact surfaces. The overlap between the meshfree shape function supports induces internal forces between nodal particles, ensuring the impenetrability between different bodies. With the abovementioned advantages, the meshfree methods appear to be promising for modeling fragment-impact problems. However, the development of robust meshfree methods for modeling air-blast and air-blast induced material fragmentation needs further development.

2.4 Summary of Literature Review

The literatures related to this research are reviewed in this chapter, and the essential developments of shock-capturing techniques and immersed formulations are investigated. The challenges of modeling material fragmentation due to blast loading are also discussed. In the following chapters, the development of novel numerical methods for modeling shock waves in fluids, modeling heterogeneous materials with immersed formulation, and modeling air-blast induced fluid-structure interaction problems are developed.

Part of this chapter is a reprint of the material as it appears in the following papers:

1. “TH. Huang, H. Wei, JS. Chen, and M. Hillman. RKPM2D: An Open-Source Implementation of Nodally Integrated Reproducing Kernel Particle Method for Solving Partial Differential Equations. Computational Particle Mechanics 7.2 (2020): 393-433. DOI: https://doi.org/10.1007/s40571-019-00272-x”,
2. “TH. Huang, JS. Chen, H. Wei, MJ. Roth, JA. Sherburn, J. Bishop, M. Tupek, and E. Fang. A MUSCL-SCNI Approach for Meshfree Modeling of Shock Waves in Fluids.

Computational Particle Mechanics 7.2 (2020): 329-350. DOI: https://doi.org/10.1007/s40571-019-00248-x."

3. "TH. Huang, JS. Chen, and MR. Tupek. An Immersed Variational Multiscale RKPM Formulation. Part I - Heterogeneous Materials. In Preparation."
4. "TH. Huang, JS. Chen, and MR. Tupek. An Immersed Variational Multiscale RKPM Formulation. Part II – Fluid-Structure Interaction. In Preparation."

The dissertation author was the primary investigator and author of these papers.

Chapter 3 Basic Equations

In chapter 3, the basic equations employed in this study are expressed. The introduction of reproducing kernel particle method (RKPM) is given in Chapter 3.1. The concept of essential shock physics associated with Riemann problems and solvers is discussed in Chapter 3.2.

3.1 Reproducing Kernel Particle Method

In recent years, the Reproducing Kernel Particle Method (RKPM) [3, 6] has been recognized as an effective numerical method for solving partial differential equations (PDEs). Compared to conventional mesh-based numerical methods such as the Finite Element Method (FEM), the reproducing kernel (RK) approximation in RKPM is constructed based on a set of scattered points without any mesh connectivity, and thus the strong tie between the quality of the discretization and the quality of approximation in conventional mesh-based methods is relaxed. This “meshfree” feature makes RKPM well-suited for solving large deformation and multiphysics problems where FEM suffers from mesh distortion or mesh entanglement [3, 4, 5]. In addition, RKPM provides controllable orders of continuity and completeness, independent from one another, which enables effective solutions of PDEs involving high-order smoothness or discontinuities, and accordingly, implementation of h- and p-adaptive refinement becomes straightforward [114, 115, 116, 117].

Furthermore, the wavelet-like multi-resolution properties can be obtained in the RK approximation, making it suitable for multi-resolution and multi-scale modeling [114, 115]. Recently, accelerated and convergent RKPM formulations have been developed with the employment of variationally consistent and stabilized nodal integration techniques [118, 119]. With abovementioned advantages, RKPM has been successfully applied to a number of challenging engineering problems, including thin shell structural mechanics [120, 121], manufacturing processes [122, 123, 124], image-based biomechanics [125], geomechanics and natural disasters [126, 127], fracture/damage mechanics [128, 129, 130], shock dynamics [38, 131, 39] and penetration/fragmentation phenomena [113, 132, 106], to name a few. Interested readers can refer to [8] for a comprehensive review of RKPM and its applications.

3.1.1 Reproducing Kernel Approximation

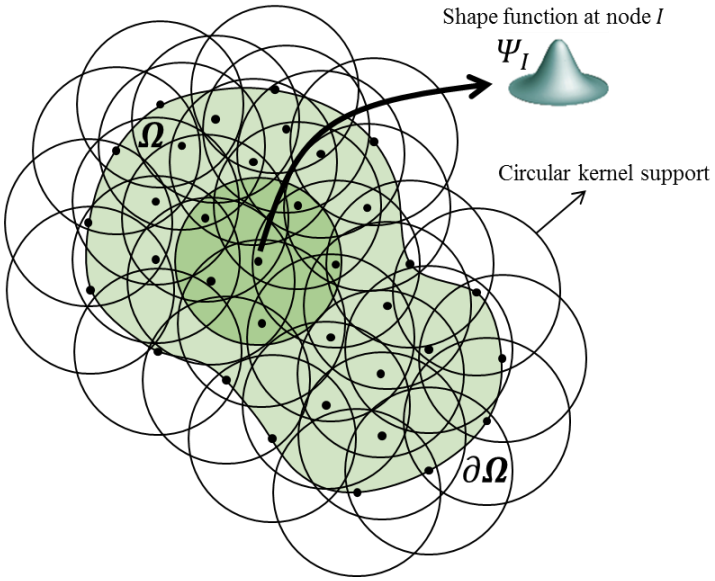


Figure 1. Illustration of a 2D RK discretization: support coverage and nodal shape function with circular kernel.

In RKPM, the numerical approximation is constructed based upon a set of scattered points or nodes [133]. The domain Ω is discretized by a set of nodes $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots \boldsymbol{x}_{NP}\}$ as shown in Figure 1, where $\boldsymbol{x}_I$ is the position vector of node I, and NP is the total number of nodes. The RK approximation of a function $u(\boldsymbol{x})$ is expressed as

$$u(\boldsymbol{x}) \approx u^h(\boldsymbol{x}) = \sum_{I \in G_x} \Psi_I(\boldsymbol{x}) u_I \tag{1}$$

where $\boldsymbol{x}$ is the spatial coordinates, u_I is the associated nodal coefficient to be determined, and $\Psi_I(\boldsymbol{x})$ is the reproducing kernel (RK) shape function of node I expressed as:

$$\Psi_I(\boldsymbol{x}) = \boldsymbol{H}^T(\boldsymbol{0}) \boldsymbol{M}^{-1}(\boldsymbol{x}) \boldsymbol{H}(\boldsymbol{x} - \boldsymbol{x}_I) \Phi_a(\boldsymbol{x} - \boldsymbol{x}_I) \tag{2}$$

where the basis vector $\boldsymbol{H}(\boldsymbol{x} - \boldsymbol{x}_I)$ is defined as

$$\boldsymbol{H}^T(\boldsymbol{x} - \boldsymbol{x}_I) = [1, x_1 - x_{1I}, x_2 - x_{2I}, x_3 - x_{3I}, (x_1 - x_{1I})^2, \ldots, (x_3 - x_{3I})^n] \tag{3}$$

and $\boldsymbol{M}(\boldsymbol{x})$ is the so-called moment matrix:

$$\boldsymbol{M}(x) = \sum_{I = G_x} \boldsymbol{H}(\boldsymbol{x} - \boldsymbol{x}_I) \boldsymbol{H}^T(\boldsymbol{x} - \boldsymbol{x}_I) \Phi_a(\boldsymbol{x} - \boldsymbol{x}_I) \tag{4}$$

The set $G_x = \{I | \Phi_a(\boldsymbol{x} - \boldsymbol{x}_I) \neq 0\}$ shown in Eq. (1) and (4) contains the nodal indexes of point $\boldsymbol{x}'s$ neighbors, and $\Phi_a(\boldsymbol{x} - \boldsymbol{x}_I)$ is the kernel function centered at $\boldsymbol{x}_I$ with compact support size a_I defined as

$$a_I = \tilde{c} h_I \tag{5}$$

In the above equation, $\tilde{c}$ is the normalized support size, and h_I is the nodal spacing associated with nodal point $\boldsymbol{x}_I$ defined as:

$$h_I = max\left(\left\|\boldsymbol{x}_J - \boldsymbol{x}_I\right\|\right), \ \forall \boldsymbol{x}_J \in B_I \tag{6}$$

in which the set B_I contains the four nodes that are closest to point $\boldsymbol{x}_I$ for 2D problems. The kernel function controls the smoothness of the approximation as shown in Figure 2, where the C^0 tent kernel function is compared with the following C^2 cubic B-spline kernel function:

$$\Phi_a(\boldsymbol{x}-\boldsymbol{x}_I)=\begin{cases} 2/3-4z_I^2+4z_I^3 & \text{for} \quad 0\le z_I\le 1/2\,, \\ 4/3-4z_I+4z_I^2-4/3\,z_I^3 & \text{for} \quad 1/2\le z_I\le 1, \\ 0 & \text{for} \quad z_I>1, \end{cases} \tag{7}$$

in which z_I is defined as $z_I=\frac{\|\boldsymbol{x}-\boldsymbol{x}_I\|}{a_I}$. In addition, shape functions with different normalized support sizes are plotted in Figure 3, which clearly illustrates that the locality of the approximation is controlled by the kernel support size.

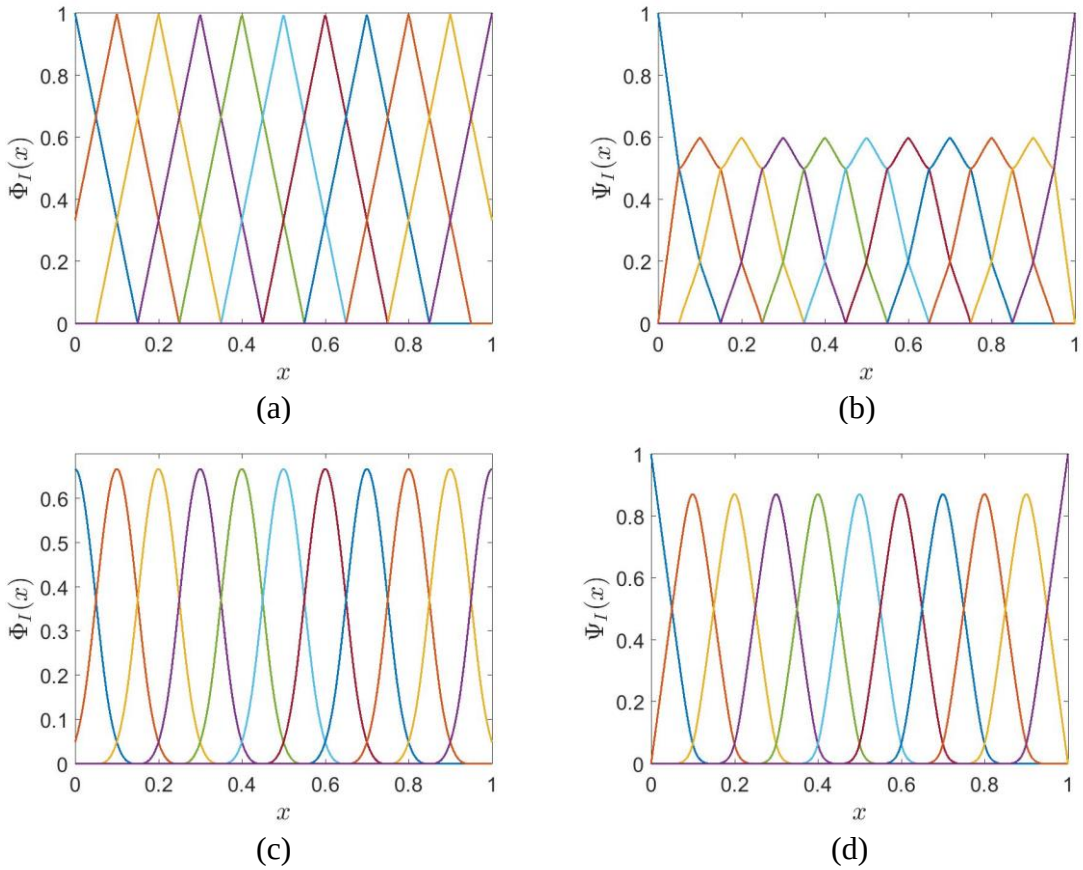
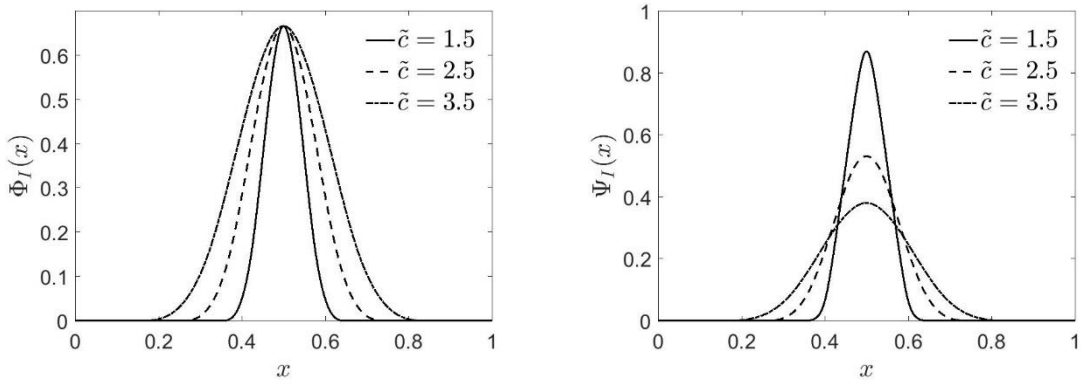


Figure 2. Kernel function and corresponding RK shape function with linear basis and normalized support size is 1.5: (a) tent kernel and (b) corresponding RK shape function. (c) cubic B-Spline kernel and (d) corresponding RK shape function.



(a) (b)

Figure 3. (a) Cubic B-spline kernel and (b) corresponding RK shape function (right) with linear basis and normalized support size equals 1.5, 2.5, 3.5

By construction, the RK shape functions satisfy the following n^{th} order reproducing conditions:

$$\sum_{I \in G_{\boldsymbol{x}}} \Psi_I(\boldsymbol{x}) x_{1I}^i x_{2I}^j x_{3I}^k = x_1^i x_2^j x_3^k, \qquad 0 \le i + j + k \le n \tag{8}$$

where n is the specified order of completeness, which determines the order of consistency in the solution of PDEs. When linear basis is employed, both the zero-th and first-order reproducing conditions are satisfied for uniform and arbitrary point distribution as shown in Figure 4.

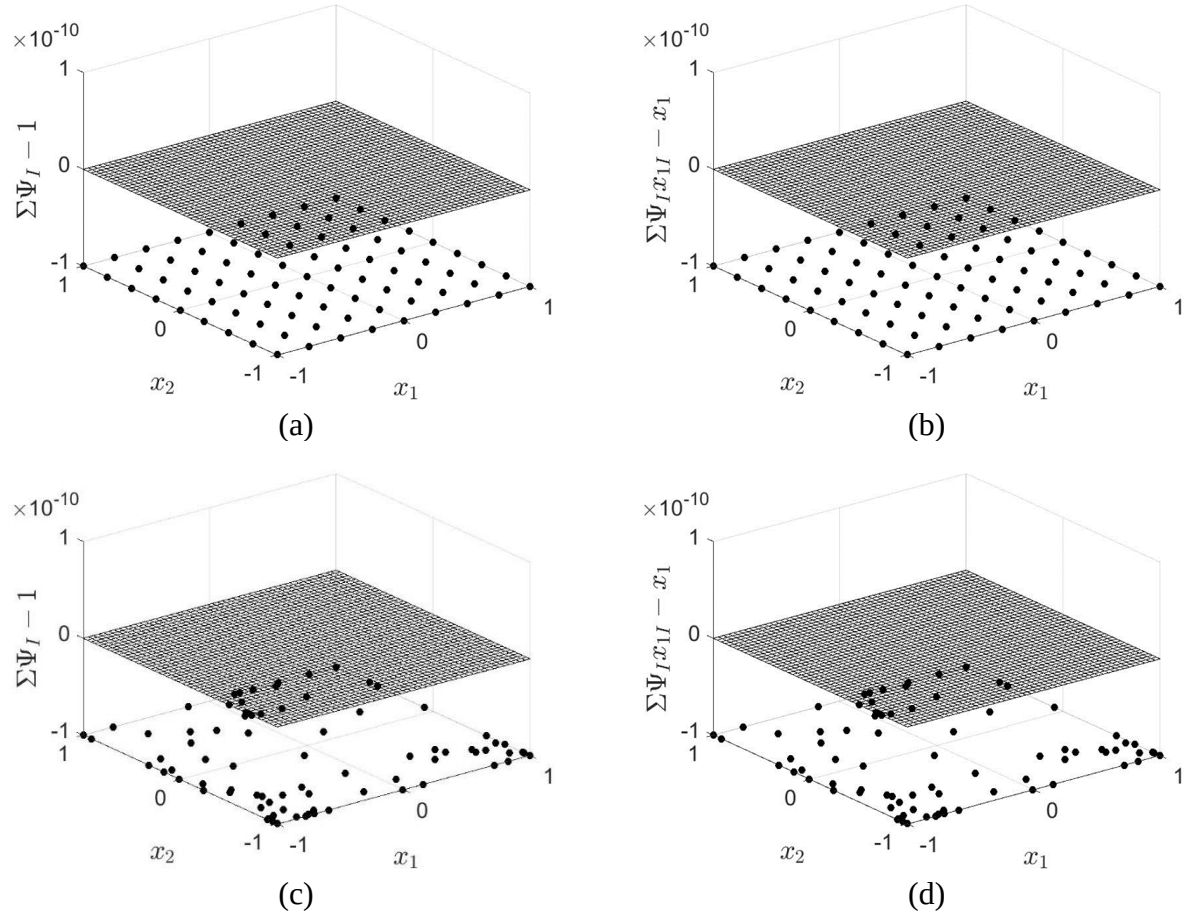


Figure 4. Errors in the zero-th and first order reproducing conditions for the RK shape function with linear basis and normalized support size is 1.5, where a uniform point distribution are used in (a) and (b) and an arbitrary point distribution is used in (c) and (d)

3.1.2 Lagrangian and Eulerian Kernel Functions

In the Lagrangian computation, the RKPM points are located at the material particles throughout deformation, and under this framework, the kinematic and kinetic variables can be referenced to the original configuration. Therefore, the kernel function is constructed in the original configuration, referred to as Lagrangian kernel function, expressed as

$$\Phi_a^{\boldsymbol{X}}(\boldsymbol{X}-\boldsymbol{X}_I)=\frac{1}{a}\Phi_a\left(\frac{\|\boldsymbol{X}-\boldsymbol{X}_I\|}{a}\right) \tag{9}$$

where $\boldsymbol{X}$ is the initial configuration coordinates, and $\Phi_a^{\boldsymbol{X}}(\boldsymbol{X}-\boldsymbol{X}_I)$ is the Lagrangian kernel function with the fixed support size a in the initial configuration. The kernel deforms following the material deformation. When Lagrangian formulation references to the current configuration, the kernel function is constructed by mapping the current position vector $\boldsymbol{x}$ to $\boldsymbol{X}=\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)$ with the mapping function $\boldsymbol{\varphi}$

$$\Phi_a^{\boldsymbol{X}}\left(\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)-\boldsymbol{\varphi}^{-1}(\boldsymbol{x}_I,t)\right)=\frac{1}{a}\Phi_a\left(\frac{\|\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)-\boldsymbol{\varphi}^{-1}(\boldsymbol{x}_I,t)\|}{a}\right) \tag{10}$$

An example of Lagrangian kernel under unidirectional tension deformation can be seen in Figure 5, where the support size of the kernel expands as the material is stretched.

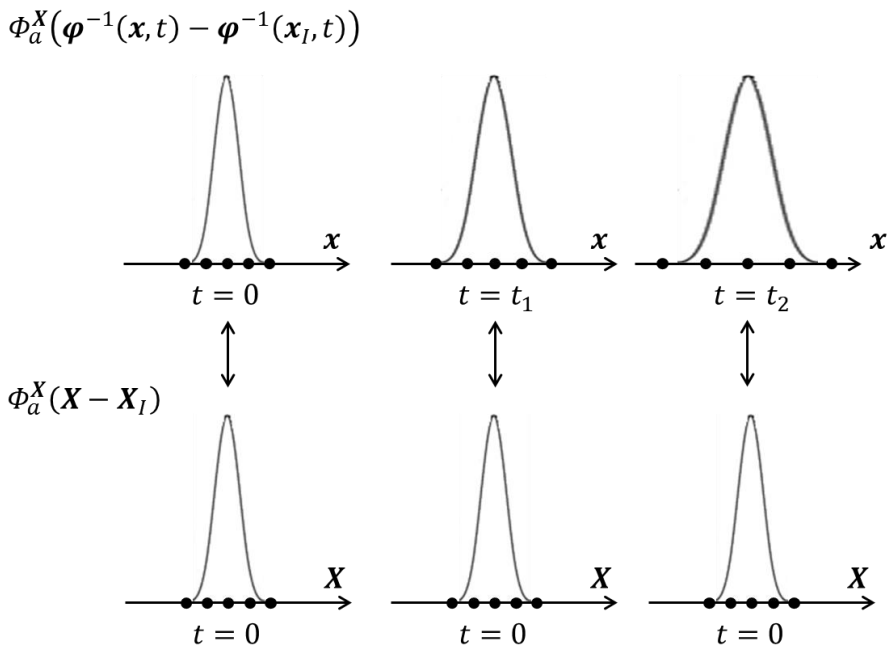


Figure 5. Lagrangian kernel function when material is under uniaxial stretching

One may also employ the Eulerian kernel function in the current configuration as follows

$$\Phi_a^{\boldsymbol{x}}(\boldsymbol{x}-\boldsymbol{x}_I)=\frac{1}{a}\Phi_a\left(\frac{\|\boldsymbol{x}-\boldsymbol{x}_I\|}{a}\right) \tag{11}$$

and similarly the Eulerian kernel function can also be expressed in the initial configuration by the mapping function $\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X}, t)$

$$\Phi_a^{\boldsymbol{x}}\big(\boldsymbol{\varphi}(\boldsymbol{X},t) - \boldsymbol{\varphi}(\boldsymbol{X}_I,t)\big) = \frac{1}{a}\Phi_a\left(\frac{\|\boldsymbol{\varphi}(\boldsymbol{X},t) - \boldsymbol{\varphi}(\boldsymbol{X}_I,t)\|}{a}\right) \tag{12}$$

Eqns. (11) and (12) indicate the Eulerian kernel functions $\Phi_a^{\boldsymbol{x}}(\boldsymbol{x} - \boldsymbol{x}_I)$ has a fixed support size in the current configuration, and has a deformation dependent support size when expressed in the initial configuration. An example of Eulerian kernel under unidirectional tension deformation can be seen in Figure 6, where it can be seen that the support size of the kernel is fixed as the material is stretched.

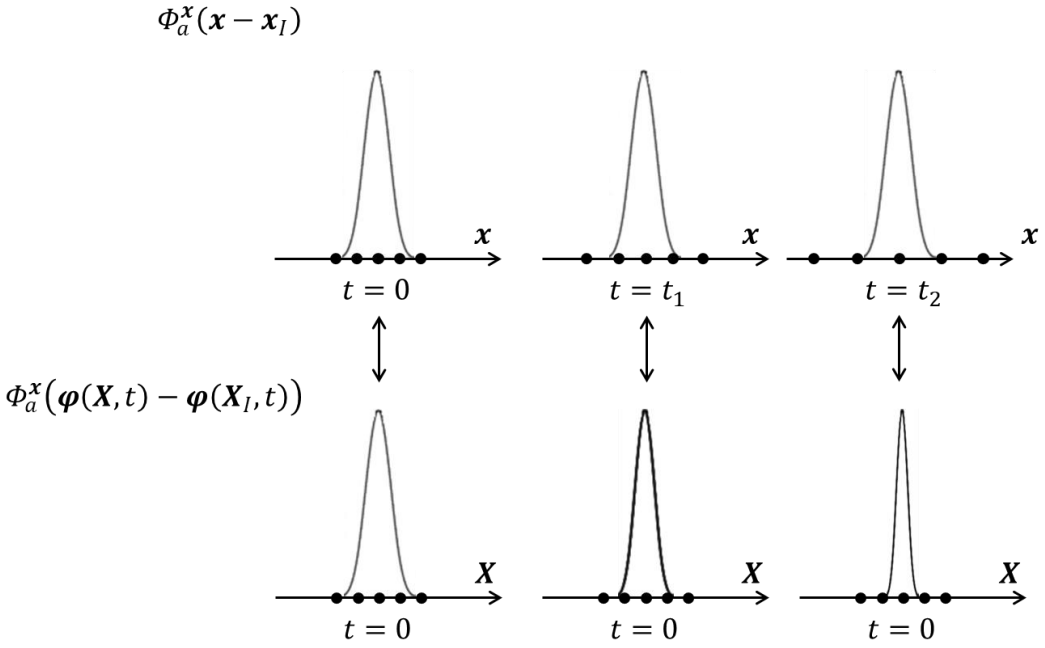


Figure 6. Eulerian kernel function when material is under uniaxial stretching

As can be seen, Eulerian kernel function encounters numerical instability if no re-adjustment in the support size is performed, whereas the Lagrangian kernel function avoids this difficulty. Therefore, Lagrangian kernel is adequate for structural problems where the Eulerian kernel

function with a fixed support size naturally fits the Eulerian formulation, like fluid problems, in which case the RKPM particles are fixed in space.

3.1.3 Nodal Quadrature Rules for Meshfree Methods

In meshfree methods, the quadrature rules paying an important role in domain integration. As shown in Chapter 3.1.1, the RK basis functions are constructed directly in the physical domain, without the requirement of mesh and corresponding iso-parametric coordinates. This feature makes RKPM suitable for modeling large deformation and fragmentation of the material, but also leads to challenges to the associated domain integration required in the weak form. When Gauss quadrature is adopted, quadrature points are generated based upon background cells [134, 135], where only the quadrature points inside the physical domain are considered for domain integration. Since RK shape functions are rational functions and their supports overlap with each other, the misalignment of Gauss integration cells and shape function supports lead to large quadrature errors unless high-order integration schemes are adopted, as shown in [118, 134]. Also employing the Gauss integration requires a background mesh, which in some aspect is against the uniqueness of "meshfree" features. Therefore, nodal integration is preferred in the meshfree method community and the development of different nodal quadrature rules is introduced as follows.

Direct Nodal Integration (DNI)

The simplest nodal integration method is Direct Nodal Integration (DNI), where shape functions and their derivatives are evaluated directly at nodes. The domain and boundary integrations can be computed as follows:

$$\int_{\Omega} \boldsymbol{P}(\boldsymbol{x}) d\Omega \approx \sum_{N=1}^{NP} \boldsymbol{P}(\boldsymbol{x}_N) A_N \tag{13}$$

$$\int_{\partial\Omega_g} \boldsymbol{Q}(\boldsymbol{x}) d\Gamma \approx \sum_{N=1}^{NPg} \boldsymbol{Q}(\hat{\boldsymbol{x}}_N) L_N \tag{14}$$

where $\boldsymbol{x}_N$, V_N and NP are the RK node locations, nodal representative domain area (volume in three-dimensional case) and the number of RK nodes, respectively, and $\hat{\boldsymbol{x}}_N$, L_N and NPg are essential boundary RK nodes, length (area in three-dimensional case) of the nodal representative domain, and the number of RK nodes on the essential boundary, respectively. The same integration rules are used for the natural boundary integration.

Stabilized Conforming Nodal Integration (SCNI)

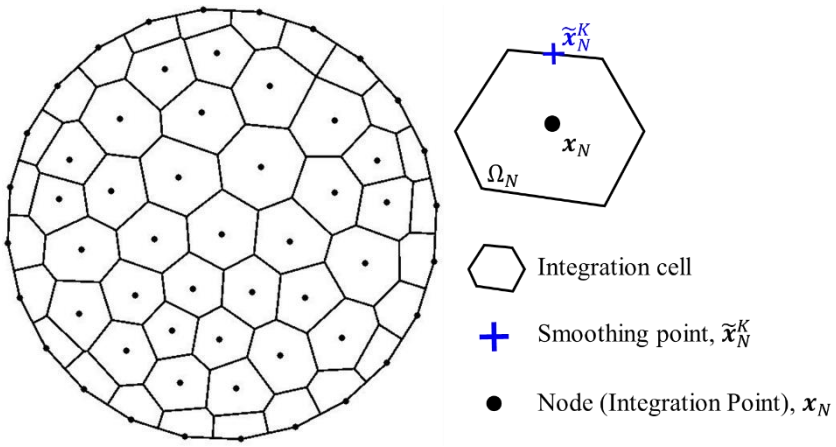


Figure 7. Voronoi cell diagram in two-dimensional domain Ω.

DNI is notorious for spurious zero-energy modes and non-convergent numerical solutions. To ensure linear variational consistency, i.e., the ability of numerical methods to pass the linear

patch test, Chen et al. [40] showed that the quadrature rules need to meet the following first order integration constraint for the shape function gradient:

$$\int_{\Omega}^{\hat{}} \Psi_{I,i} d\Omega = \int_{\partial\Omega}^{\hat{}} \Psi_I n_i d\Gamma \tag{15}$$

In (15), ^ over the integral symbols denotes numerical integration. For nodal integration as the quadrature rule for the domain integration on the left hand side of Eq. (15), Chen et al. [40] introduced the following nodally smoothed gradient $\widetilde{\Psi}_{I,i}$ at the nodal point $\boldsymbol{x}_N$:

$$\widetilde{\Psi}_{I,i}(\boldsymbol{x}_N) = \frac{1}{A_N} \int_{\Omega_N} \Psi_{I,i}(\boldsymbol{x}) d\Omega = \frac{1}{A_N} \int_{\partial\Omega_N} \Psi_I(\boldsymbol{x}) n_i(\boldsymbol{x}) d\Gamma \tag{16}$$

where A_N denotes the area of the nodal representative domain Ω_N associated with node N, and n_i denotes the i^{th} component of the outward unit normal vector to the smoothing domain boundary as shown in Figure 7. It was shown in [40] that integrating Eq. (15) with nodal integration with the smoothed gradient of shape function in Eq. (16), the first order integration constraint in Eq. (15) is exactly satisfied as long as the same boundary integral quadrature rules are used for the right hand side of both Eqns. (15) and (16). As discussed in [118], in order to maintain linear consistency of the smoothed gradient of a linearly consistent shape function, a simple one-point Gauss integration rule can be used for the contour integral in Eq. (16):

$$\widetilde{\Psi}_{I,i}(\boldsymbol{x}_N) \approx \frac{1}{A_N} \sum_{K \in S_N} \Psi_I(\widetilde{\boldsymbol{x}}_N^K) n_i(\widetilde{\boldsymbol{x}}_N^K) L_K \tag{17}$$

where $S_N = \{K | \widetilde{\boldsymbol{x}}_N^K \in \partial\Omega_N\}$ contains all center points of each boundary segment associated with node $\boldsymbol{x}_N$, and the integration weight L_K is the length of the K^{th} segment of the smoothing cell boundary. Take linear elasticity as an example, by employing smoothed shape function gradients, the stiffness matrix as well as the force vectors are re-formulated as follows:

$$\boldsymbol{K}_{IJ}^{c} = \int_{\Omega} \boldsymbol{B}_{I}^{T}(\boldsymbol{x})\boldsymbol{C}\boldsymbol{B}_{J}(\boldsymbol{x})\, d\Omega \approx \sum_{N=1}^{NP} \widetilde{\boldsymbol{B}}_{I}^{T}(\boldsymbol{x}_{N})\boldsymbol{C}\widetilde{\boldsymbol{B}}_{J}(\boldsymbol{x}_{N})A_{N} \tag{18}$$

where $\boldsymbol{K}_{IJ}^{c}$ is the stiffness matrix, $\boldsymbol{C}$ is the elastic tensor following Voigt notation, and $\widetilde{\boldsymbol{B}}_{I}(\boldsymbol{x}_{L})$ for two dimensional case is defined as

$$\widetilde{\boldsymbol{B}}_{I}(\boldsymbol{x}_{N}) = \begin{bmatrix} \widetilde{\Psi}_{I,1}(\boldsymbol{x}_{N}) & 0 \\ 0 & \widetilde{\Psi}_{I,2}(\boldsymbol{x}_{N}) \\ \widetilde{\Psi}_{I,2}(\boldsymbol{x}_{N}) & \widetilde{\Psi}_{I,1}(\boldsymbol{x}_{N}) \end{bmatrix} \tag{19}$$

Remark: For the smoothed gradients in Eq. (16), conforming nodal representative domains as shown in Figure 7 are employed here by following the SCNI approach [40]. On the other hand, non-conforming nodal representative domains can also be employed, which leads to the so-called stabilized non-conforming nodal integration (SNNI) approach [136]. While SNNI violates the first-order integration constraint (Eq. (15)), it can be corrected by the variational consistent integration (VCI) [118] to recover the optimal rates of convergence. VCI can also be used to achieve higher rates of convergence in RKPM with higher order bases in the RK approximation.

Stabilized Non-Conforming Nodal Integration (SNNI)

For large-deformation, high strain-rate problems, it may be required for SCNI to recalculate the conforming domain cells at each time step, which is computationally costly. For problem involving fragmentation, the reconstruction of the conforming cell may not be physical. Hence, stabilized non-conforming nodal integration (SNNI) was devised [137, 138], where the conforming requirement is relaxed, and a simple circular or rectangular smoothing domain is used as expressed in Figure 8 below.

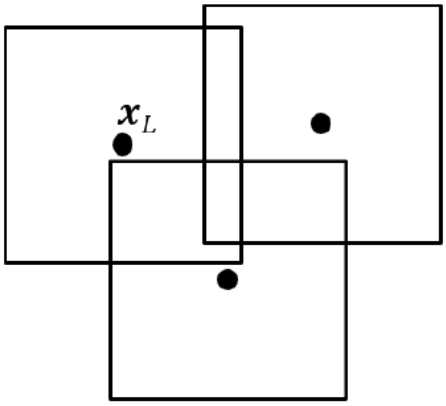


Figure 8. Nonconforming Smoothing Cells

To further suppress non-zero energy modes in problems where surface to volume ratio is low, additional stabilization for SNNI can be introduced to [113]. Due to the lack of the domain conformality, SNNI fails to pass the linear patch test but may still exhibits a convergence rate close to the theoretical one [113].

Variationally Consistent Integration (VCI)

To remedy the issue of the reduction in accuracy and convergence rates in using SNNI, variationally consistent integration (VCI) [139] can be applied. VCI corrects the smoothed test strains $\delta\tilde{\varepsilon}_{ij}$ yielding corrected test strains $\delta\bar{\varepsilon}_{ij}$ in the Galerkin formulation without correcting the smoothed trial strains $\tilde{\varepsilon}_{ij}$:

$$\delta\bar{\varepsilon}_{ij}(\boldsymbol{x}_L) = \frac{1}{2}\left(\delta\bar{u}_{i,j}(\boldsymbol{x}_L) + \delta\bar{u}_{j,i}(\boldsymbol{x}_L)\right) \tag{20}$$

To satisfy the integration constraint, a correction function $R_I(\boldsymbol{x}_L)$ is introduced into the variational gradients $\delta\bar{u}^h_{i,j}$:

$$\delta\bar{u}^h_{i,j}(\boldsymbol{x}_L) = \sum_{I=1}^{NP}\left(\widetilde{\Psi}_{I,j}(\boldsymbol{x}_L) + R_I(\boldsymbol{x}_L)\zeta_{jI}\right)\delta u_{Ii} \tag{21}$$

where $\widetilde{\Psi}_{I,j}$ is the SNNI smoothed RKPM shape function gradient, and $R_I(\boldsymbol{x}_L)$ is:

$$R_I(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{x} \in supp\big(\Psi_I(\boldsymbol{x})\big) \\ 0 & \text{otherwise} \end{cases} \tag{22}$$

The coefficients ζ_{jI} in Eq.(21) are solved so as to satisfy integration constraints:

$$\zeta_{jI} = -\frac{\left(\int_\Omega \widetilde{\Psi}_{I,j}\, d\Omega - \int_{\partial\Omega} \Psi_I n_j\, d\Gamma\right)}{\int_\Omega R_I\, d\Omega} \tag{23}$$

where "^" denotes numerical integration. SNNI with the VCI correction (VC-SNNI) results in a numerical accuracy close to SCNI as well as an optimal convergence rate [139].

Naturally Stabilized Nodal Integration

Spurious oscillatory modes can be triggered in nodal integration methods. Therefore, additional stabilization techniques are needed to eliminate these low-energy modes, which one can employ the *naturally stabilized nodal integration* (NSNI) proposed in [119], where an implicit gradient expansion of the strain field is introduced as:

$$\boldsymbol{\varepsilon}\Big(\boldsymbol{u}^h(\boldsymbol{x})\Big) \approx \boldsymbol{\varepsilon}\Big(\boldsymbol{u}^h(\boldsymbol{x}_L)\Big) + \sum_{i=1}^{d} (x_i - x_{Li})\, \boldsymbol{\varepsilon}\left(\widehat{\boldsymbol{u}}^h_{,i}(\boldsymbol{x}_L)\right) \tag{24}$$

where $\widehat{\boldsymbol{u}}^h_{,i}(\boldsymbol{x}_L) = \sum_I \Psi^{\nabla}_{Ii}(\boldsymbol{x}_L)\boldsymbol{u}_I$ is the implicit gradient of the displacement with Ψ^{∇}_{Ii} the implicit gradient of the RK shape function [129]:

$$\boldsymbol{\Psi}^{\nabla}_{Ii} = \boldsymbol{D}_i^T \boldsymbol{M}^{-1}(\boldsymbol{x})\boldsymbol{H}(\boldsymbol{x} - \boldsymbol{x}_I)\Phi_a(\boldsymbol{x} - \boldsymbol{x}_I) \tag{25}$$

where the vector $\boldsymbol{D_i}$ takes on the following values:

$$\boldsymbol{D}_i = -[0 \quad \delta_{i1} \quad \delta_{i2} \quad \delta_{i3} \quad 0 \quad \cdots \quad 0]^T \tag{26}$$

Introducing the gradient expansion terms (24) into the variational equations, the stiffness matrix is obtained as

$$\boldsymbol{K}_{IJ}^{c} \approx \boldsymbol{K}_{IJ}^{c-SCNI} + \boldsymbol{K}_{IJ}^{c-NSNI}$$
$$= \sum_{L=1}^{NP} \left(\widetilde{\boldsymbol{B}}_{I}^{T}(\boldsymbol{x}_L) \boldsymbol{C} \widetilde{\boldsymbol{B}}_{J}(\boldsymbol{x}_L) A_L \right) + \sum_{L=1}^{NP} \sum_{i=1}^{d} \boldsymbol{B}_{Ii}^{-\nabla T}(\boldsymbol{x}_L) \boldsymbol{C}^{-} \boldsymbol{B}_{Ji}^{-\nabla}(\boldsymbol{x}_L) M_{iL} \tag{27}$$

where $\boldsymbol{K}_{IJ}^{c-SCNI}$ is the stiffness matrix constructed through SCNI, $\boldsymbol{K}_{IJ}^{c-NSNI}$ is the additional stabilization matrix constructed by NSNI, and $\boldsymbol{B}_{Ii}^{\nabla}(\boldsymbol{x}_N)$ for two-dimensional case is defined as follows:

$$\boldsymbol{B}_{I1}^{\nabla}(\boldsymbol{x}_N) = \begin{bmatrix} \Psi_{Ii,1}^{\nabla}(\boldsymbol{x}_N) & 0 \\ 0 & \Psi_{Ii,2}^{\nabla}(\boldsymbol{x}_N) \\ \Psi_{Ii,2}^{\nabla}(\boldsymbol{x}_N) & \Psi_{Ii,1}^{\nabla}(\boldsymbol{x}_N) \end{bmatrix} \tag{28}$$

Here, the derivatives of the RK implicit gradients $\Psi_{Ii,j}^{\nabla}$ in Eq. (28) are obtained by the direct differentiation of the first-order implicit gradient Ψ_{Ii}^{∇} in Eq. (25) with respect to x_j [119]. The terms $M_{iL} = M_i(\boldsymbol{x}_L)$ in Eq. (28) are the second moments of inertia in each nodal integration domain:

$$M_i(\boldsymbol{x}_L) = \int_{\Omega_L} (x_i - x_{Li})^2 d\Omega \tag{29}$$

From Eqns. (27) - (29), no subdivision of integration cells is required in the stabilization.

From Chapter 3.1, a summary for the state-of-the-art reproducing kernel particle method is introduced, which includes the advanced nodal integration as well as Lagrangian and Eulerianl kernel description, that are ready for different solid and fluid problems. In the next section, a

special technique to solve partial differential equations involving shocks, called Riemann solver, is introduced.

3.2 Riemann Problem, Godunov Scheme and Riemann Solvers for Shock Modeling

A Riemann problem is a special initial boundary value problem having a conservation equation associated with piecewise constant initial data, which has a single discontinuity in the domain of interest. The Riemann problem is helpful for the understanding of essential physics of conservation equations because all properties, such as shocks and rarefaction waves, appear as characteristics in the solution. In certain scenarios, such as one dimensional adiabatic and inviscid compressible flow (a.k.a Euler equation), it also gives an exact solution termed Riemann solutions. For more complex physics that Riemann solutions cannot be derived analytically, approximated solvers or iterative solvers can be used. These solvers are called (approximated) Riemann solvers, and they are widely employed in the computational fluid mechanics involving shock wave dynamics.

For the illustration of how Riemann solver works, a one-dimensional Burgers equation is employed as a simplified Riemann problem:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad f(u) = \frac{1}{2}u^2, \quad \forall x \in]-L/2, L/2[\tag{30}$$

where L is the domain size. u is the scalar variable and $f(u)$ is the flux. The problem is subjected to the initial condition specifying a discontinuity at $x = 0$:

$$\begin{aligned} u &= u_L, \quad \forall x \leq 0 \\ u &= u_R, \quad \forall x > 0 \end{aligned} \tag{31}$$

with the boundary condition:

$$\begin{aligned} u &= u_L, \quad \forall x = -\frac{L}{2} \\ u &= u_R, \quad \forall x = \frac{L}{2} \end{aligned} \tag{32}$$

where the subscript L and R indicates the state at left-hand side and right-hand side of $x = 0$, respectively. Eq. (30) can be rewritten in the following form

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial u}\frac{\partial u}{\partial x} = \frac{\partial u}{\partial t} + \lambda \frac{\partial u}{\partial x} = 0 \tag{33}$$

and $\lambda = f_{,u}(u) = u$ gives the characteristic speed of the systems. Depending on the characteristic speed due to initial condition (31), rarefaction or shock wave may develop. The shock wave solution states that the shock wave speed can be obtained from the jump condition given by

$$S = \frac{[\![f(u)]\!]}{[\![u]\!]} = \frac{f(u_R) - f(u_L)}{u_R - u_L} = \frac{1}{2}\frac{u_R^2 - u_L^2}{u_R - u_L} = \frac{u_R + u_L}{2} \tag{34}$$

where S is the shock wave speed and $[\![\cdot]\!]$ is the jump operator. Eq. (34) represents the jump condition for conservation law, as known as Rankine-Hugoniot jump condition. From Eq. (34), it can be seen that both $u_R > u_L$ or $u_L > u_R$ may both develop shock wave solution. Therefore, a requirement for a unique single-valued solution is that the solution should satisfy the Lax entropy condition in Eq. (35)

$$\text{(downstream)}\ \ f_{,u}(u_R) = u_R \leq S \leq u_L = f_{,u}(u_L)\ \ \text{(upstream)} \tag{35}$$

where the shock wave speed should be bounded by the characteristic speeds at upstream and downstream side. For physically real applications this means that the solution should satisfy the entropy condition. From Eq. (35), it can be easily informed that $u_L > u_R$ gives the shock wave solution. On the other hand, the rarefaction wave solutions exist when $\lambda(x < 0) = u_L < \lambda(x > 0) = u_R$ since such characteristic wave solution creates a wave fan at $x = 0$. Therefore, the solution of the Riemann problem reads:

If $u_L > u_R$, the solution has a shock wave of speed S and the solution of the Riemann problem is given by

$$u(x,t) = \begin{cases} u_L, & \frac{x}{t} < S \\ u_R, & \frac{x}{t} > S \end{cases} \tag{36}$$

If $u_L < u_R$, then the solution has a rarefaction wave and the solution is given by

$$u(x,t) = \begin{cases} u_L, & \frac{x}{t} < f_{,u}(u_L) \\ f_{,u}^{-1}(x/t), & f_{,u}(u_R) > \frac{x}{t} > f_{,u}(u_L) \\ u_R, & \frac{x}{t} > f_{,u}(u_R) \end{cases} \tag{37}$$

Although this is a simple example, it still shows the basic properties of solving a Riemann problem with given initial condition, where the characteristics decompose the solution into different circumstances (shock wave, rarefaction, or more complicated wave pattern). For the Riemann solution of more complex physics, readers may refer to [140].

3.2.1 Godunov's scheme with Riemann Solvers

Godunov's scheme [13] employs the idea of using the Riemann solution for the flux calculation at cell boundary in the classical finite volume method framework. An one-dimensional conservation equation in a hyperbolic form is considered:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}(\boldsymbol{U})}{\partial x} = \boldsymbol{0} \tag{38}$$

where $\boldsymbol{U}$ is the vector composed of the conserved variables and $\boldsymbol{F}(\boldsymbol{U})$ is the associated flux. The solution of Eq. (38) by the Godunov's method in the discrete form can be written as:

$$\boldsymbol{U}_i^{n+1} = \boldsymbol{U}_i^n - \frac{\Delta t}{\Delta x}\left(\boldsymbol{F}_{i+1/2}^{*n} - \boldsymbol{F}_{i-1/2}^{*n}\right) \tag{39}$$

where $\boldsymbol{U}_i^{n+1} = \boldsymbol{U}(x_i, t_{n+1})$ is the solution at discrete nodal position $\boldsymbol{x}_i$ and time step t_{n+1}. $\boldsymbol{F}_{i+1/2}^{*n} = \boldsymbol{F}\left(\boldsymbol{U}^*\left(x_{i+1/2}, t_n\right)\right)$ is the flux evaluated by the Riemann solution $\boldsymbol{U}^*\left(x_{i+1/2}, t_n\right)$ from the following Riemann problem at finite volume cell interface $x_{i+1/2} = (x_i + x_{i+1})/2$ at time step t_n:

$$\begin{aligned} &\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}(\boldsymbol{U})}{\partial x} = \boldsymbol{0}, \qquad x \in \left]x_i, x_{i+1}\right[\\ &\boldsymbol{U} = \boldsymbol{U}_i, \qquad \forall x \le x_{i+1/2} \\ &\boldsymbol{U} = \boldsymbol{U}_{i+1}, \qquad \forall x > x_{i+1/2} \end{aligned} \tag{40}$$

It can be seen that the Godunov's scheme requires an exact, or approximated Riemann solver to solve the Riemann problem at every cell interface. The later one is preferred due to lower computational cost, and often the exact Riemann solution may not exist for complex physical problems. In general, these solvers usually gives an expression or approximation of the flux terms $\boldsymbol{F}_{i+1/2}^*$ with given conservation variable $\boldsymbol{U}$ and flux $\boldsymbol{F}$ at both side of $x_{i+1/2}$, instead of

obtaining the expression of $\boldsymbol{U}^*$ at $x_{i+1/2}$, for the convenience of evaluation of flux in Godunov's scheme in Eq. (39). Here, some well-known approximated Riemann solvers are listed below.

3.2.2 Common Riemann Solvers

For given initial value problems in Eq. (40), the flux $\boldsymbol{F}^*_{i+1/2}$ evaluated at $x_{i+1/2}$ can be obtained by the following Riemann solvers:

Rusanov solver (also called local Lax-Friedrich solver) [14]

The flux $\boldsymbol{F}^*_{i+1/2}$ can be evaluated by

$$\boldsymbol{F}^*_{i+1/2} = \frac{1}{2}[\boldsymbol{F}_i + \boldsymbol{F}_{i+1} - \eta(\boldsymbol{U}_{i+1} - \boldsymbol{U}_i)] \tag{41}$$

where $\eta = \max\big(\lambda(\boldsymbol{x}_i), \lambda(\boldsymbol{x}_{i+1})\big)$, where λ here is the maximum characteristic speed from the eigenvalues of $\boldsymbol{A} = \partial \boldsymbol{F}/\partial \boldsymbol{U}$.

Harten-Lax-van Leer (HLL) solver [16]

The flux $\boldsymbol{F}^*_{i+1/2}$ can be evaluated by

$$\boldsymbol{F}^*_{i+1/2} = \begin{cases} \boldsymbol{F}_i & if \;\; 0 \le S_- \\ \boldsymbol{F}^* & if \;\; S_- < 0 < S_+ \\ \boldsymbol{F}_{i+1} & if \;\; S_+ \le 0 \end{cases} \tag{42}$$

where $\boldsymbol{F}^*$ is computed by the following equation

$$\boldsymbol{F}^* = \frac{S_+ \boldsymbol{F}_i - S_- \boldsymbol{F}_{i+1} + S_+ S_- (\boldsymbol{U}_{i+1} - \boldsymbol{U}_i)}{(S_+ - S_-)} \tag{43}$$

and S_+ and S_- are respectively upper bound and lower bound of characteristic wave speed from the eigenvalues of $\boldsymbol{A} = \partial \boldsymbol{F}/\partial \boldsymbol{U}$. The reference [16] gives an rough estimate of S_+ and S_- as $S_+ = \max\big(0, \lambda(\boldsymbol{x}_i), \lambda(\boldsymbol{x}_{i+1})\big)$ and $S_- = \min\big(0, \lambda(\boldsymbol{x}_i), \lambda(\boldsymbol{x}_{i+1})\big)$ for all $\lambda = eig(\boldsymbol{A})$.

Dukowicz solver [17]

The Dukowicz solver is a special solver for the Lagrangian form of one-dimensional mass, momentum and energy conservation equation

$$\begin{aligned} \frac{d\rho}{dt} + \rho\frac{\partial u}{\partial x} &= 0 \\ \rho\frac{du}{dt} + \frac{\partial p}{\partial x} &= 0 \\ \rho\frac{dE}{dt} + \frac{\partial pu}{\partial x} &= 0 \end{aligned} \tag{44}$$

$\forall x \in]x_i, x_{i+1}[$. The variable ρ, u, p, E are respectively density, velocity, pressure and total energy density. The governing equation is subject to specific equation of state $p = p(\rho, u, E)$

$$\begin{aligned} (\rho, u, p, E) &= (\rho_i, u_i, p_i, E_i), \qquad \forall x \le x_{i+1/2} \\ (\rho, u, p, E) &= (\rho_{i+1}, u_{i+1}, p_{i+1}, E_{i+1}), \qquad \forall x > x_{i+1/2} \end{aligned} \tag{45}$$

By using the Dukowicz Riemann solver [17], the velocity $u^*_{i+1/2}$ at can be obtained by solving the following semi-quadratic equation

$$\begin{aligned} \rho_i B_i |u^* - u^*_{min}|\big(u^*_{i+1/2} - u^*_{min}\big) + \rho_{i+1} B_{i+1} |u^* - u^*_{max}|\big(u^*_{i+1/2} - u^*_{max}\big) + p^*_i - p^*_{i+1} \\ = 0 \end{aligned} \tag{46}$$

where

$$\begin{aligned} u^*_{min} &= u_i - 0.5\frac{c_i}{B_i}, \qquad u^*_{max} = u_{i+1} + 0.5\frac{c_{i+1}}{B_{i+1}} \\ p^*_i &= p_i - 0.25\frac{\rho_i c_i^2}{B_i}, \qquad p^*_{i+1} = p_{i+1} - 0.25\frac{\rho_{i+1} c_{i+1}^2}{B_{i+1}}, \end{aligned} \tag{47}$$

Here c is the sound speed, and B is a parameter directly related to the shock density ratio in the limit of strong shocks. For ideal gas $B = (\gamma + 1)/2$, for other materials, one may selecting B by the method given in [17]. Once the velocity $u^*_{i+1/2}$ is obtained by Eq. (55), the pressure $p^*_{i+1/2}$ at $x_{i+1/2}$ can be obtained by the following equation

$$p^*_{i+1/2} = \frac{1}{2}(p^*_{i+1} + p^*_i) + \frac{1}{2}\rho_i B_i |u^* - u^*_{min}|\left(u^*_{i+1/2} - u^*_{min}\right) - \frac{1}{2}\rho_{i+1} B_{i+1} |u^* - u^*_{max}|\left(u^*_{i+1/2} - u^*_{max}\right) \tag{48}$$

The velocity $u^*_{i+1/2}$ and pressure $p^*_{i+1/2}$ can then be used to formulate the flux $\boldsymbol{F}^*_{i+1/2}$ directly.

Part of this chapter is a reprint of the material as it appears in the paper "TH. Huang, H. Wei, JS. Chen, and M. Hillman. RKPM2D: An Open-Source Implementation of Nodally Integrated Reproducing Kernel Particle Method for Solving Partial Differential Equations. Computational Particle Mechanics 7.2 (2020): 393-433. DOI: https://doi.org/10.1007/s40571-019-00272-x". The dissertation author was the primary investigator and author of this paper.

Chapter 4 Reproducing Kernel Particle Method for Modeling Shocks in Fluids

In this chapter, the overview of proposed shock algorithm, MUSCL-SCNI, for meshfree shock modeling in fluid problems is reviewed, where the fluid model is mentioned in Chapter 4.1. the numerical formulation of MUSCL-SCNI is developed and formulated in Chapter 4.2. The numerical results to benchmark the formulation are given in Chapter 4.3.

4.1 Fluid Model

4.1.1 Strong Form

Consider compressible flow in the domain Ω, the Navier-Stokes equations in conservation form consists of balance equations of mass, momentum and energy, as follows:

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho\boldsymbol{u}\\ \rho E\end{pmatrix}+\boldsymbol{\nabla}\cdot\begin{pmatrix}\rho\boldsymbol{u}\\ \rho\boldsymbol{u}\otimes\boldsymbol{u}\\ \rho E\boldsymbol{u}\end{pmatrix}+\boldsymbol{\nabla}\cdot\begin{pmatrix}0\\ p\boldsymbol{I}\\ p\boldsymbol{u}\end{pmatrix}-\boldsymbol{\nabla}\cdot\begin{pmatrix}0\\ \boldsymbol{\tau}\\ \boldsymbol{\tau}\cdot\boldsymbol{u}\end{pmatrix}=\begin{pmatrix}0\\ \rho\boldsymbol{b}\\ \rho\boldsymbol{b}\cdot\boldsymbol{u}+\rho r\end{pmatrix}\ \text{on}\ \Omega \qquad (49)$$

where $\boldsymbol{u}$ consists of the flow velocity components in Cartesian coordinates, ρ is mass density, E is the total energy density, p is the thermodynamic pressure, $\boldsymbol{\tau}$ is the viscous stress, $\boldsymbol{b}$ is the

body force, and r is the energy source. The whole system is assumed to be adiabatic, so no external heat flux is supplied. Initial conditions for velocity, mass density and energy density are defined as follows

$$\begin{aligned} \boldsymbol{u}(\boldsymbol{x}, t=0) &= \boldsymbol{u}_0 \\ \rho(\boldsymbol{x}, t=0) &= \rho_0 \\ E(\boldsymbol{x}, t=0) &= E_0 \end{aligned} \tag{50}$$

The essential boundary conditions are prescribed on $\partial\Omega_g$ as:

$$\begin{aligned} \boldsymbol{u}(\boldsymbol{x}, t) &= \boldsymbol{u}_g(\boldsymbol{x}, t), \qquad \text{on } \partial\Omega_g^u \\ \rho(\boldsymbol{x}, t) &= \rho_g(\boldsymbol{x}, t), \qquad \text{on } \partial\Omega_g^\rho \\ E(\boldsymbol{x}, t) &= E_g(\boldsymbol{x}, t), \qquad \text{on } \partial\Omega_g^E \end{aligned} \tag{51}$$

On the natural boundary $\partial\Omega_h$, the following conditions are imposed:

$$\boldsymbol{\sigma}(\boldsymbol{x}, t) \cdot \boldsymbol{n}(\boldsymbol{x}, t) = \boldsymbol{h}(\boldsymbol{x}, t), \qquad \text{on } \partial\Omega_h^u \tag{52}$$

where $\boldsymbol{n}$ is the outward unit normal vector on the boundary, $\boldsymbol{h}$ is the traction acting on the boundary, and $\boldsymbol{\sigma}$ is the Cauchy stress:

$$\boldsymbol{\sigma} = -p\boldsymbol{I} + \boldsymbol{\tau} \tag{53}$$

The material response is described by the following constitutive equations:

$$p = p(\boldsymbol{u}, \rho, E) \tag{54}$$

$$\boldsymbol{\tau} = \lambda(\boldsymbol{\nabla} \cdot \boldsymbol{u})\boldsymbol{I} + \mu(\boldsymbol{\nabla}\boldsymbol{u} + (\boldsymbol{\nabla}\boldsymbol{u})^T) \tag{55}$$

where λ is the bulk viscosity and μ is the dynamic viscosity. Eq. (54) is often referred to as the Equation of State (EOS). For example, the EOS of ideal gas is expressed as

$$p = (\gamma - 1)\rho e \tag{56}$$

where $e = E - \frac{\|\boldsymbol{u}\|}{2}$ is the specific internal energy and γ is the ideal gas constant.

4.1.2 Essential Shock Physics

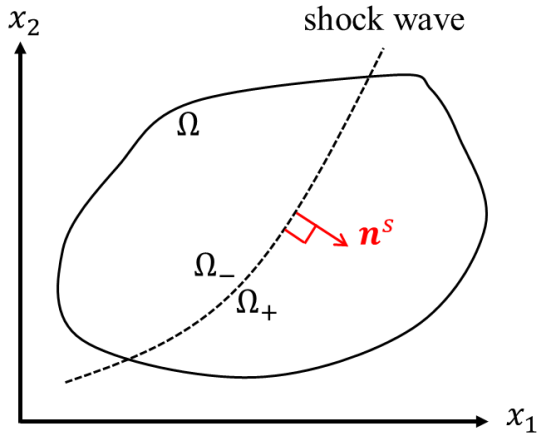


Figure 9. A two-dimensional domain subject to a shock wave.

For describing shock wave propagation, it is convenient to introduce a local coordinate axis defined by $x^s = (\boldsymbol{x}^+ - \boldsymbol{x}^-) \cdot \boldsymbol{n}^s$, where $\boldsymbol{x}^+$ and $\boldsymbol{x}^-$ denote the locations ahead and behind the shock wave front, and $\boldsymbol{n}^s$ is the unit normal vector along the shock wave propagating direction. The projection of the flux $\boldsymbol{F}_i = \rho u_i(1, u_1, u_2, u_3, E)^T + p(0, \delta_{1i}, \delta_{2i}, \delta_{3i}, u_i)^T$, and the velocity $\boldsymbol{u}$ can then be defined as $F_i^s = \boldsymbol{F}_i \cdot \boldsymbol{n}^s$ and $u^s = \boldsymbol{u} \cdot \boldsymbol{n}^s$, respectively. When shock wave propagates, as illustrated in Figure 9, the solution to the conservation equations (49) contains discontinuities in velocity, mass density, and total energy density along the direction of $\boldsymbol{n}^s$. The shock wave speed S is governed by the Rankine-Hugoniot (R-H) jump conditions for the flux and conservation variables:

$$\begin{aligned} S &= \frac{[\![\rho u^s]\!]}{[\![\rho]\!]} \\ S &= \frac{[\![\rho (u^s)^2 + p]\!]}{[\![\rho u^s]\!]} \\ S &= \frac{\left[\!\left[\rho u^s \left(E + \frac{p}{\rho}\right)\right]\!\right]}{[\![\rho E]\!]} \end{aligned} \tag{57}$$

where $[\![\cdot]\!]$ represents the jump operator, i.e., $[\![u^s]\!] = u^{s+} - u^{s-}$, "+" implies ahead of the wave, "−" implies behind the wave, and S is the shock speed. In addition, the shock speed is bounded

by the characteristic speed $F^s_{,u}(u)$ on the upstream and the downstream of the shock front, as described by the entropy condition [141]:

$$F^s_{,u}(u^{s+}) < S < F^s_{,u}(u^{s-}), \qquad \text{on } x^s \tag{58}$$

where the characteristic speed $F^s_{,u}(u)$ is the eigenvalue of the Jacobian matrix $\frac{\partial \boldsymbol{F}_i}{\partial \boldsymbol{U}}$, where $\boldsymbol{U}$ is the conservation variable, $\boldsymbol{U} = \rho(1, u_1, u_2, u_3, E)^T$. The above equation indicates that the discontinuity moves faster than the characteristic speed $F^s_{,u}(u^{s+})$ in front of the shock front, and slower than the characteristic speed $F^s_{,u}(u^{s-})$ trailing it. The R-H jump condition in (57) and entropy conditions in (58) represent essential shock physics, which need to be satisfied in the numerical modeling.

4.1.3 Weak Form and Galerkin Equation

The weak form for Eq. (49) can be written as

$$\int_\Omega \begin{pmatrix} w_\rho \\ \boldsymbol{w}_u \\ w_E \end{pmatrix} \cdot \left[\begin{pmatrix} \rho \\ \rho \boldsymbol{u} \\ \rho E \end{pmatrix}_{,t} + \boldsymbol{\nabla} \cdot \begin{pmatrix} \rho \boldsymbol{u} \\ \rho \boldsymbol{u} \otimes \boldsymbol{u} \\ \rho E \boldsymbol{u} \end{pmatrix} + \boldsymbol{\nabla} \cdot \begin{pmatrix} 0 \\ p\boldsymbol{I} \\ p\boldsymbol{u} \end{pmatrix} - \boldsymbol{\nabla} \cdot \begin{pmatrix} 0 \\ \boldsymbol{\tau} \\ \boldsymbol{\tau} \cdot \boldsymbol{u} \end{pmatrix} - \begin{pmatrix} 0 \\ \rho \boldsymbol{b} \\ \rho \boldsymbol{b} \cdot \boldsymbol{u} + \rho r \end{pmatrix} \right] \mathrm{d}\Omega = \begin{pmatrix} 0 \\ \boldsymbol{0} \\ 0 \end{pmatrix} \tag{59}$$

where the trial functions $\boldsymbol{u} \in \left[H^1_g(\Omega)\right]^d$, $\rho \in H^1_g(\Omega)$, $E \in H^1_g(\Omega)$, and test functions $\boldsymbol{w}_u \in [H^1_0(\Omega)]^d$, $w_\rho \in H^1_0(\Omega)$, $w_E \in H^1_0(\Omega)$. Applying integration by part to the viscosity term and considering the boundary conditions (52) and (53), the weak form is re-written as:

$$\int_\Omega \begin{pmatrix} w_\rho \\ \boldsymbol{w}_u \\ w_E \end{pmatrix} \cdot \left[\begin{pmatrix} \rho \\ \rho\boldsymbol{u} \\ \rho E \end{pmatrix}_{,t} + \boldsymbol{\nabla} \cdot \begin{pmatrix} \rho\boldsymbol{u} \\ \rho\boldsymbol{u}\otimes\boldsymbol{u} \\ \rho E\boldsymbol{u} \end{pmatrix} + \boldsymbol{\nabla} \cdot \begin{pmatrix} \boldsymbol{0} \\ p\boldsymbol{I} \\ p\boldsymbol{u} \end{pmatrix} \right] \mathrm{d}\Omega + \int_\Omega \begin{pmatrix} \boldsymbol{\nabla} w_\rho \\ \boldsymbol{\nabla} \boldsymbol{w}_u \\ \boldsymbol{\nabla} w_E \end{pmatrix} : \begin{pmatrix} 0 \\ \boldsymbol{\tau} \\ \boldsymbol{\tau} \cdot \boldsymbol{u} \end{pmatrix} \mathrm{d}\Omega = \int_\Omega \begin{pmatrix} w_\rho \\ \boldsymbol{w}_u \\ w_E \end{pmatrix} \cdot \begin{pmatrix} 0 \\ \rho\boldsymbol{b} \\ \rho\boldsymbol{b} \cdot \boldsymbol{u} + \rho r \end{pmatrix} \mathrm{d}\Omega + \int_{\partial\Omega_h^u} \begin{pmatrix} w_\rho \\ \boldsymbol{w}_u \\ w_E \end{pmatrix} \cdot \begin{pmatrix} 0 \\ \boldsymbol{h} + p\boldsymbol{n} \\ 0 \end{pmatrix} d\Gamma \tag{60}$$

For shock waves in compressible flow, the effects of bulk viscosity λ and dynamic viscosity μ are usually negligible [142]. Hence, the above equations become:

$$\int_\Omega \begin{pmatrix} w_\rho \\ \boldsymbol{w}_u \\ w_E \end{pmatrix} \cdot \left[\begin{pmatrix} \rho \\ \rho\boldsymbol{u} \\ \rho E \end{pmatrix}_{,t} + \boldsymbol{\nabla} \cdot \begin{pmatrix} \rho\boldsymbol{u} \\ \rho\boldsymbol{u}\otimes\boldsymbol{u} \\ \rho E\boldsymbol{u} \end{pmatrix} + \boldsymbol{\nabla} \cdot \begin{pmatrix} \boldsymbol{0} \\ p\boldsymbol{I} \\ p\boldsymbol{u} \end{pmatrix} \right] \mathrm{d}\Omega = \int_\Omega \begin{pmatrix} w_\rho \\ \boldsymbol{w}_u \\ w_E \end{pmatrix} \cdot \begin{pmatrix} 0 \\ \rho\boldsymbol{b} \\ \rho\boldsymbol{b} \cdot \boldsymbol{u} + \rho r \end{pmatrix} \mathrm{d}\Omega + \int_{\partial\Omega_h^u} \begin{pmatrix} w_\rho \\ \boldsymbol{w}_u \\ w_E \end{pmatrix} \cdot \begin{pmatrix} 0 \\ \boldsymbol{h} + p\boldsymbol{n} \\ 0 \end{pmatrix} d\Gamma \tag{61}$$

To construct Galerkin equations for Eq. (61), the trial functions: $\boldsymbol{u}^h \in \left[H_g^1(\Omega)\right]^d$, $\rho^h \in H_g^1(\Omega), E^h \in H_g^1(\Omega)$, and test functions $\boldsymbol{w}_u^h \in [H_0^1(\Omega)]^d$, $w_\rho^h \in H_0^1(\Omega), w_E^h \in H_0^1(\Omega)$ are introduced by applying the reproducing kernel (RK) approximation [6, 3] for the primary variables as follows

$$\begin{aligned} \boldsymbol{u}^h(\boldsymbol{x},t) &= \sum_{I=1}^{NP} \Psi_I(\boldsymbol{x})\boldsymbol{u}_I(t), \quad \boldsymbol{w}_u^h(\boldsymbol{x}_I,t) = \sum_{I=1}^{NP} \Psi_I(\boldsymbol{x})\boldsymbol{w}_I^u(t) \\ \rho^h(\boldsymbol{x},t) &= \sum_{I=1}^{NP} \Psi_I(\boldsymbol{x})\rho_I(t), \quad w_\rho^h(\boldsymbol{x}_I,t) = \sum_{I=1}^{NP} \Psi_I(\boldsymbol{x})w_I^\rho(t) \\ E^h(\boldsymbol{x},t) &= \sum_{I=1}^{NP} \Psi_I(\boldsymbol{x})E_I(t), \quad w_E^h(\boldsymbol{x}_I,t) = \sum_{I=1}^{NP} \Psi_I(\boldsymbol{x})w_I^E(t) \end{aligned} \tag{62}$$

where $\Psi_I(\boldsymbol{x})$ is the RK shape function of node I evaluated at point $\boldsymbol{x}$, and $\boldsymbol{u}_I(t), \rho_I(t), E_I(t)$, $\boldsymbol{w}_I^u(t), w_I^\rho(t), w_I^E(t)$ are the nodal coefficients. The conservation variables $\boldsymbol{U}^h$ and $\boldsymbol{W}^h$, pressure flux $\boldsymbol{F}_p^h$, advection flux $\boldsymbol{F}_{adv}^h$, body force $\boldsymbol{F}_b^h$ and traction $\boldsymbol{F}_t^h$ are defined as follows

$$\boldsymbol{U}^h = (\rho^h, \rho^h \boldsymbol{u}^h, \rho^h E^h)^T \tag{63}$$

$$\boldsymbol{W}^h = \left(w_\rho^h, \boldsymbol{w}_u^h, w_E^h\right)^T \tag{64}$$

$$\boldsymbol{F}_p^h = (\boldsymbol{0}, p^h \boldsymbol{I}, p^h \boldsymbol{u}^h)^T \tag{65}$$

$$\boldsymbol{F}_{adv}^h = (\rho^h \boldsymbol{u}^h, \rho^h \boldsymbol{u}^h \otimes \boldsymbol{u}^h, \rho^h E^h \boldsymbol{u}^h)^T \tag{66}$$

$$\boldsymbol{F}_b^h = (0, \rho^h \boldsymbol{b}, \rho^h \boldsymbol{b} \cdot \boldsymbol{u}^h + \rho^h r)^T \tag{67}$$

$$\boldsymbol{F}_t^h = (0, \boldsymbol{h} + p^h \boldsymbol{n}, 0)^T \tag{68}$$

in which the pressure p^h can be obtained from EOS (54) as

$$p^h = p(\boldsymbol{u}^h, \rho^h, E^h) \tag{69}$$

Accordingly, the RK discretization of Eq. (61) is written as:

$$\int_\Omega \boldsymbol{W}^h \cdot \left[\boldsymbol{U}_{,t}^h + \boldsymbol{\nabla} \cdot \boldsymbol{F}_p^h + \boldsymbol{\nabla} \cdot \boldsymbol{F}_{adv}^h\right] d\Omega = \int_\Omega \boldsymbol{W}^h \cdot \boldsymbol{F}_b^h d\Omega + \int_{\partial\Omega_h^u} \boldsymbol{W}^h \cdot \boldsymbol{F}_t^h d\Gamma \tag{70}$$

4.2 Shock-Enhanced Meshfree Formulation for Fluid Problems

In this section, the shock-enhanced algorithm is discussed to solve the Galerkin equations mentioned in the previous section.

4.2.1 Riemann-SCNI with Flux Splitting

The domain integration of Eq. (70) is performed by employing the stabilized conforming nodal integration (SCNI) approach [38, 39, 40]:

$$\begin{aligned} &\sum_{I=1}^{NP} \boldsymbol{W}^h(\boldsymbol{x}_I) \cdot \left[\boldsymbol{U}_{,t}^h(\boldsymbol{x}_I) + \widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^h(\boldsymbol{x}_I) + \widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^h(\boldsymbol{x}_I)\right] V_I \\ &\qquad = \sum_{I=1}^{NP} \boldsymbol{W}^h(\boldsymbol{x}_I) \cdot \boldsymbol{F}_b^h(\boldsymbol{x}_I) V_I + \sum_{I \in \partial\Omega_h^u} \boldsymbol{W}^h(\boldsymbol{x}_I) \cdot \boldsymbol{F}_t^h(\boldsymbol{x}_I) A_I \end{aligned} \tag{71}$$

where V_I denotes the volume of the nodal representative domain associated with node I, and A_I denotes the boundary surface area associated with node I, and the smooth flux divergence (SFD) operators are computed by boundary integrals as follows:

$$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^h(\boldsymbol{x}_I) \approx \frac{1}{V_I} \sum_{K \in S_I} \left. \begin{pmatrix} \mathbf{0} \\ p^h \boldsymbol{I} \\ p^h \boldsymbol{u}^h \end{pmatrix} \right|_{(\widetilde{\boldsymbol{x}}_I^K)} \cdot \boldsymbol{n}_I^K A_I^K \tag{72}$$

$$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^h(\boldsymbol{x}_I) \approx \frac{1}{V_I} \sum_{K \in S_I} \left. \begin{pmatrix} \rho^h \boldsymbol{u}^h \\ \rho^h \boldsymbol{u}^h \otimes \boldsymbol{u}^h \\ \rho^h E^h \boldsymbol{u}^h \end{pmatrix} \right|_{(\widetilde{\boldsymbol{x}}_I^K)} \cdot \boldsymbol{n}_I^K A_I^K \tag{73}$$

in which $S_I = \{K | \widetilde{\boldsymbol{x}}_I^K \in \partial\Omega_I\}$ contains all boundary smoothing points $\widetilde{\boldsymbol{x}}_I^K$ of node I, $\boldsymbol{n}_I^K$ denotes the outward unit normal vector of the K^{th} boundary segment of the I^{th} nodal representative domain, and A_I^K is the boundary segment area in three dimensions or segment length in two dimensions (see Figure 10 for an illustration of the two-dimensional nodal representative domain).

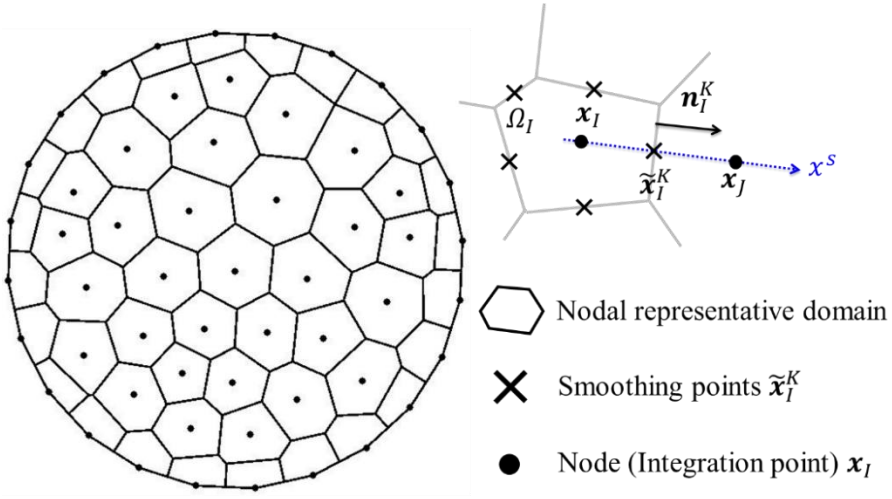


Figure 10. Illustration of nodal representative domain.

In the boundary evaluation of the smooth flux divergence operators shown in Figure 10, the velocity $\boldsymbol{u}^h(\widetilde{\boldsymbol{x}}_I^K)$ and pressure $p^h(\widetilde{\boldsymbol{x}}_I^K)$ in Eqns. (72) and (73) are computed using the Riemann-enriched velocity $\boldsymbol{u}^*(\widetilde{\boldsymbol{x}}_I^K)$ and pressure $p^*(\widetilde{\boldsymbol{x}}_I^K)$ from solving local Riemann problems along the

local coordinate x^s, in which the superscript "$*$" indicates Riemann solutions. Analogous to the Godunov's scheme [13], initial conditions of velocity, mass density, total energy density, and pressure for the local Riemann problem are constructed from cell-averages from two sides of each cell boundary:

$$\bar{\boldsymbol{u}}_I = \frac{1}{V_I}\int_{\Omega_I} \boldsymbol{u}^h(\boldsymbol{x})d\Omega\,, \qquad \bar{\boldsymbol{u}}_J = \frac{1}{V_J}\int_{\Omega_J} \boldsymbol{u}^h(\boldsymbol{x})d\Omega \tag{74}$$

$$\bar{\rho}_I = \frac{1}{V_I}\int_{\Omega_I} \rho^h(\boldsymbol{x})d\Omega\,, \qquad \bar{\rho}_J = \frac{1}{V_J}\int_{\Omega_J} \rho^h(\boldsymbol{x})d\Omega \tag{75}$$

$$\bar{E}_I = \frac{1}{V_I}\int_{\Omega_I} E^h(\boldsymbol{x})d\Omega\,, \qquad \bar{E}_J = \frac{1}{V_J}\int_{\Omega_J} E^h(\boldsymbol{x})d\Omega \tag{76}$$

$$\bar{p}_I = \frac{1}{V_I}\int_{\Omega_I} p^h(\boldsymbol{x})d\Omega\,, \qquad \bar{p}_J = \frac{1}{V_J}\int_{\Omega_J} p^h(\boldsymbol{x})d\Omega \tag{77}$$

Based on (74), projections of velocity fields of cell I and J onto the local coordinate axis x^s are expressed as:

$$\bar{u}_I = \bar{\boldsymbol{u}}_I \cdot \boldsymbol{n}_I^K, \qquad \bar{u}_J = \bar{\boldsymbol{u}}_J \cdot \boldsymbol{n}_I^K \tag{78}$$

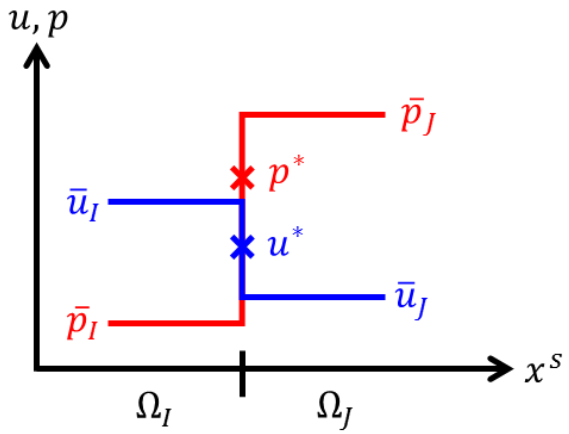


Figure 11. The local Riemann problem along x^s direction with piecewise constant approximations of pressure and velocity, where (p^*, u^*) are the Riemann solutions.

As illustrated in Figure 11, the jump conditions (57) for the pressure flux [143] are imposed across cell boundaries:

$$\begin{aligned}(p^* - \bar{p}_I) &= \bar{\rho}_I S(u^* - \bar{u}_I) \\ (p^* - \bar{p}_J) &= \bar{\rho}_J S(u^* - \bar{u}_J)\end{aligned} \tag{79}$$

where p^*, u^* are the corrected pressure and velocity fields from the Riemann solver, and the shock speed S can be approximated from the shock Hugoniot curve [143] as:

$$S = c\text{sign}([\![\bar{u}]\!]) + A_s[\![\bar{u}]\!] \tag{80}$$

where c is the sound speed and A_s is the parameter that correlates the shock speed S with the flow velocity $\boldsymbol{u}$, obtained from experimental tests. Eqns. (79) and (80) with the entropy condition (58) can be solved together by using the Newton-Raphson method, but in practice, less expensive approximate solutions are often sought by adopting Riemann solvers, such as Rusanov solver [14], Roe solver [15], Harten-Lax-van Leer (HLL) solver [16], and Dukowicz solver [17], etc. In this study, the Dukowicz solver is employed, due to its high accuracy. Once the velocity u^* is obtained from the Riemann solver, the global corrected velocity at cell boundary can be reconstructed:

$$\boldsymbol{u}^*(\tilde{\boldsymbol{x}}_I^K) = u^*(\tilde{\boldsymbol{x}}_I^K)\boldsymbol{n}_I^K \tag{81}$$

Note that, if the smoothing point $\tilde{\boldsymbol{x}}_I^K$ is located at the domain boundary $\partial\Omega$, then chosing $(p^* = \bar{p}_I, u^* = \bar{u}_I)$, instead of applying Riemann solvers. The corrected velocity $\boldsymbol{u}^*(\tilde{\boldsymbol{x}}_I^K)$ and pressure $p^*(\tilde{\boldsymbol{x}}_I^K)$ at smoothing points are then employed to calculate the Riemann-enriched smoothed divergence of the pressure flux as a correction to the flux divergence $\tilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^h(\boldsymbol{x}_I)$ in Eq. (72) as:

$$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^*(\boldsymbol{x}_I) \approx \frac{1}{V_I} \sum_{K \in S_I} \boldsymbol{F}_p^*(\widetilde{\boldsymbol{x}}_I^K) \cdot \boldsymbol{n}_I^K A_I^K \tag{82}$$

where $\boldsymbol{F}_p^*(\widetilde{\boldsymbol{x}}_I^K)$ is the Riemann-enriched pressure flux:

$$\boldsymbol{F}_p^*(\widetilde{\boldsymbol{x}}_I^K) = \begin{pmatrix} \boldsymbol{0} \\ p^*(\widetilde{\boldsymbol{x}}_I^K)\boldsymbol{I} \\ p^*(\widetilde{\boldsymbol{x}}_I^K)\boldsymbol{u}^*(\widetilde{\boldsymbol{x}}_I^K) \end{pmatrix} \tag{83}$$

where the Riemann-enriched pressure and velocity naturally introduces the shock physics within the nodal integration framework. In addition, as the piecewise constant data $\bar{\boldsymbol{u}}_I, \bar{\rho}_I$ and $\bar{p}_I$ are used as input for the Riemann solvers, new extrema are avoided, and the corresponding smoothed flux divergence in Eq. (82) is non-oscillatory.

To correct the advection flux divergence $\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^h(\boldsymbol{x}_I)$ of Eq. (73), a natural approach is to define the following Riemann-enriched smoothed divergence with an upwind scheme:

$$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^{\mathrm{up}*}(\boldsymbol{x}_I) \approx \frac{1}{V_I} \sum_{K \in S_I} \boldsymbol{F}_{adv}^{\mathrm{up}*}(\widetilde{\boldsymbol{x}}_I^K) \cdot \boldsymbol{n}_I^K A_I^K \tag{84}$$

in which an upwind scheme is formulated to take into account the velocity jump across shock fronts:

$$\boldsymbol{F}_{adv}^{\mathrm{up}*}(\widetilde{\boldsymbol{x}}_I^K) = \begin{cases} \begin{pmatrix} \bar{\rho}_I \boldsymbol{u}^* \\ \bar{\rho}_I \boldsymbol{u}^* \otimes \bar{\boldsymbol{u}}_I \\ \bar{\rho}_I \bar{E}_I \boldsymbol{u}^* \end{pmatrix} & \text{if } u^* \geq 0 \\ \begin{pmatrix} \bar{\rho}_J \boldsymbol{u}^* \\ \bar{\rho}_J \boldsymbol{u}^* \otimes \bar{\boldsymbol{u}}_J \\ \bar{\rho}_J \bar{E}_J \boldsymbol{u}^* \end{pmatrix} & \text{if } u^* < 0 \end{cases} \tag{85}$$

where $\boldsymbol{u}^*$ is the corrected velocity defined in Eq. (81). However, this conventional upwind scheme in Eq. (85) causes strong diffusion in modeling shock waves and over-smears shock fronts. As shown in the modified wave number analysis [144], the magnitude of numerical diffusion in the conventional upwind scheme is $O(u^*)$; so excessive diffusion is introduced when the corrected velocity u^* is high. To address this issue, the modified upwind scheme [55, 145] is adopted here.

Instead of Eqns. (84) and (41), the following modified smoothed divergence for the advection flux is introduced:

$$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^{\mathrm{mup}*}(\boldsymbol{x}_I) \approx \frac{1}{V_I} \sum_{K \in S_I} \boldsymbol{F}_{adv}^{\mathrm{mup}*}(\widetilde{\boldsymbol{x}}_I^K) \cdot \boldsymbol{n}_I^K A_I^K \tag{86}$$

and

$$\boldsymbol{F}_{adv}^{\mathrm{mup}*}(\widetilde{\boldsymbol{x}}_I^K) = \begin{cases} \begin{pmatrix} \bar{\rho}_I \boldsymbol{B}_I \\ \bar{\rho}_I \boldsymbol{B}_I \otimes \bar{\boldsymbol{u}}_I \\ \bar{\rho}_I \bar{E}_I \boldsymbol{B}_I \end{pmatrix} & \text{if } u^* \geq 0 \\ \begin{pmatrix} \bar{\rho}_J \boldsymbol{B}_J \\ \bar{\rho}_J \boldsymbol{B}_J \otimes \bar{\boldsymbol{u}}_J \\ \bar{\rho}_J \bar{E}_J \boldsymbol{B}_J \end{pmatrix} & \text{if } u^* < 0 \end{cases} \tag{87}$$

where $\boldsymbol{B}_I$ and $\boldsymbol{B}_J$ denote the local characteristic wave speeds:

$$\boldsymbol{B}_I = B_I \boldsymbol{n}_I^K, \ \boldsymbol{B}_J = B_J \boldsymbol{n}_I^K \tag{88}$$

in which $B_{L=I,J}$ is defined as:

$$B_L = M_L c_L \tag{89}$$

where $L = I, J$, and M_L denotes the local Mach number

$$M_L = \frac{u^*}{u^* - s_L} \tag{90}$$

s_L denotes the local wave speed estimate suggested by Davis [146]:

$$s_L = \begin{cases} \min(0, \bar{u}_L - c_L, u^* - c^*), & \text{if } L = I \\ \max(0, \bar{u}_L + c_L, u^* + c^*), & \text{if } L = J \end{cases} \tag{91}$$

and c_L is sound speed:

$$c_L = \bar{u}_L - s_L \tag{92}$$

By introducing the Riemann-enriched smoothed flux divergence defined in Eq. (82) and the advection flux divergence defined in Eq. (86), Eq. (71) becomes:

$$\begin{aligned} &\sum_{I=1}^{NP} \boldsymbol{W}^h(\boldsymbol{x}_I) \cdot \left[\boldsymbol{U}_{,t}^h(\boldsymbol{x}_I) + \widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^*(\boldsymbol{x}_I) + \widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^{\mathrm{mup}*}(\boldsymbol{x}_I) \right] V_I \\ &\qquad = \sum_{I=1}^{NP} \boldsymbol{W}^h(\boldsymbol{x}_I) \cdot \boldsymbol{F}_b^h(\boldsymbol{x}_I) V_I + \sum_{I \in \partial\Omega_h^u} \boldsymbol{W}^h(\boldsymbol{x}_I) \cdot \boldsymbol{F}_t^h(\boldsymbol{x}_I) A_I \end{aligned} \tag{93}$$

4.2.2 MUSCL-type Enhancement and MUSCL-SCNI Formulation

Piecewise constant (PC) approximations used to reconstruct local Riemann solutions can guarantee the non-oscillatory behavior. However, it also leads to low-order accuracy. It is desirable to employ low-order approximation at discontinuities [38, 39, 130, 126], while maintaining high-order approximations within smooth regions. To this end, the MUSCL scheme [18] is introduced into the Riemann-SCNI formulation as an enhancement with the construction of a monotonic piecewise linear (PL) approximation:

$$\hat{p}_I(\boldsymbol{x}) \approx \bar{p}_I + \varphi_I^p \widetilde{\boldsymbol{\nabla}} p^h(\boldsymbol{x}_I) \cdot (\boldsymbol{x} - \boldsymbol{x}_I), \quad \forall \boldsymbol{x} \in \Omega_I \tag{94}$$

which represents the limited pressure field in the *I*-th nodal representative domain, and $\widetilde{\boldsymbol{\nabla}}$ is the smoothed gradient operator:

$$\widetilde{\boldsymbol{\nabla}} p^h(\boldsymbol{x}_I) = \frac{1}{V_I} \int_{\Omega_I} \boldsymbol{\nabla} p^h(\boldsymbol{x}) d\Omega = \frac{1}{V_I} \int_{\partial\Omega_I} p^h(\boldsymbol{x}) \boldsymbol{n}(\boldsymbol{x}) d\Gamma \approx \frac{1}{V_I} \sum_{K \in S_I} p^h(\tilde{\boldsymbol{x}}_I^K) \boldsymbol{n}_I^K A_I^K \tag{95}$$

in Eq. (94), φ_I^p is the limiting parameter for pressure field associated with the *I*-th smoothing cell, and its absolute value is bounded by one, which controls the magnitude of the local slope $\widetilde{\nabla} p^h(\boldsymbol{x}_I)$. In the limiting procedure, the Barth-Jespersen (BJ) limiter [147] is employed to construct the limiting parameter φ_I^p by measuring the local field's variation. The limiter φ_I^p is defined as:

$$\varphi_I^p = \min_{J \in G_I} \begin{cases} \min\left(1, r_J^{max}\right) & \text{if } \widetilde{\boldsymbol{\nabla}} p^h(\boldsymbol{x}_I) \cdot \left(\boldsymbol{x}_J - \boldsymbol{x}_I\right) > 0 \\ \min\left(1, r_J^{min}\right) & \text{if } \widetilde{\boldsymbol{\nabla}} p^h(\boldsymbol{x}_I) \cdot \left(\boldsymbol{x}_J - \boldsymbol{x}_I\right) < 0 \\ 1 & \text{otherwise} \end{cases} \tag{96}$$

where $G_I = \left\{J | \partial\Omega_I \cap \partial\Omega_J \neq \emptyset, \ \forall I \neq J\right\}$ contains the list of all neighbor cells of the I^{th} gradient smoothing cell, and r_J^{min} and r_J^{max} are the minimum and maximum thresholds for the allowable

variation at the J^{th} neighbor cell of the gradient smoothing cell I, along the direction of $\overrightarrow{\boldsymbol{x}_I\boldsymbol{x}_J}$, defined as:

$$r_J^{min} = \frac{\min\limits_{J\in G_I}\left[p^h(\boldsymbol{x}_J)\right] - p^h(\boldsymbol{x}_I)}{\widetilde{\boldsymbol{\nabla}}p^h(\boldsymbol{x}_I)\cdot(\boldsymbol{x}_J - \boldsymbol{x}_I)},\ r_J^{max} = \frac{\max\limits_{J\in G_I}\left[p^h(\boldsymbol{x}_J)\right] - p^h(\boldsymbol{x}_I)}{\widetilde{\boldsymbol{\nabla}}p^h(\boldsymbol{x}_I)\cdot(\boldsymbol{x}_J - \boldsymbol{x}_I)} \tag{97}$$

As seen from Eqns. (94) - (97), at the shock region the limiting parameter is close to zero which recovers the PC approximation, whereas at non-shock region limiting parameter is close to one, which recovers the high-order approximation. The limiting procedure is also applied to the velocity field, the mass density field and the total energy density. For velocity field $\hat{\boldsymbol{u}}$, the PL reconstructed velocity $\hat{\boldsymbol{u}}$ is obtained by applying limiting process to each velocity component as follows:

$$\hat{u}_{iI}(\boldsymbol{x}) \approx \bar{u}_{iI} + \varphi_{iI}^u \widetilde{\boldsymbol{\nabla}} u_i^h(\boldsymbol{x}_I)\cdot(\boldsymbol{x} - \boldsymbol{x}_I) \tag{98}$$

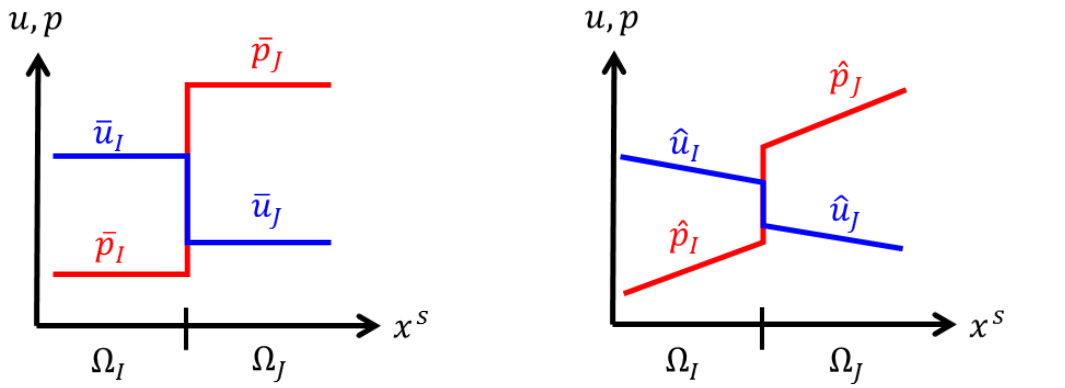
where $\hat{\boldsymbol{u}}$ and $\bar{\boldsymbol{u}}$ are PL and PC reconstructed *i*-th component and $\boldsymbol{\varphi_{iI}^u}$ is the corresponding limiting parameter. The PL reconstructed mass density field and the total energy density are expressed as follows:

$$\hat{\rho}_I(\boldsymbol{x}) \approx \bar{\rho}_I + \varphi_I^\rho \widetilde{\boldsymbol{\nabla}}\rho^h(\boldsymbol{x}_I)\cdot(\boldsymbol{x} - \boldsymbol{x}_I) \tag{99}$$

$$\hat{E}_I(\boldsymbol{x}) \approx \bar{E}_I + \varphi_I^E \widetilde{\boldsymbol{\nabla}} E^h(\boldsymbol{x}_I)\cdot(\boldsymbol{x} - \boldsymbol{x}_I) \tag{100}$$

where φ_I^ρ and φ_I^E are the limiters for mass density and total energy density correspondingly. Given u_i^h, ρ^h, E^h and their gradient $\widetilde{\boldsymbol{\nabla}}u_i^h$, $\widetilde{\boldsymbol{\nabla}}\rho^h$, $\widetilde{\boldsymbol{\nabla}}E^h$, the limiter φ_{iI}^u, φ_I^ρ, and φ_I^E in Eqns (98) - (100) can be calculated according to Eq. (96) and (97). Local velocity projections for the limited velocity are also applied along the local coordinate axis x^s as $\hat{u}_I = \hat{\boldsymbol{u}}_I \cdot \boldsymbol{n}_I^K$ and $\hat{u}_J = \hat{\boldsymbol{u}}_J \cdot \boldsymbol{n}_I^K$. Replacing the PC approximation in Eqns. (74) - (77) with the monotonic PL

reconstruction in Eqns. (94), and (98) - (100), an accuracy-enhanced formulation is obtained, which is referred as MUSCL-SCNI hereafter. Riemann solutions (p^*, u^*) computed at the smoothing point $\tilde{\boldsymbol{x}}_I^K$ by using PC reconstruction $(\bar{p}, \bar{u})$ and PL reconstruction $(\hat{p}, \hat{u})$ are illustrated in Figure 12 (a) and (b), respectively.



(a) Piecewise constant approximation

(b) Piecewise linear approximation

Figure 12. Piecewise constant (PC) and piecewise linear (PL) reconstruction.

To demonstrate the accuracy enhancement of the monotonic PL reconstruction, the approximation error of a smooth function is analyzed:

$$u(x) = -\cos(\pi x), \quad -1 \leq x \leq 1 \tag{101}$$

and a non-smooth function:

$$u(x) = \begin{cases} 2.5 - \cos(\pi x), & -1 \leq x \leq 0 \\ -0.5 - \cos(2\pi x), & 0 < x \leq 1 \end{cases} \tag{102}$$

The L_2 error norm is defined as follows:

$$\|u - u^h\|_{L_2} = \left(\int_{-1}^{1} \left(u^{exact}(x) - u^h(x) \right)^2 dx \right)^{\frac{1}{2}} \tag{103}$$

in which the superscripts 'h' and 'exact' denote numerical and exact solutions, respectively. Different approaches, including PL reconstruction in Eqns. (94), (98) - (100), PC reconstruction in Eqns. (74) - (77) , and the standard RK approximation in Eq. (62), are considered. The RK approximation is constructed using linear basis, cubic B-spline kernel, and normalized support size $\tilde{c} = 1.5$, under both uniform and non-uniform nodal discretizations, as shown in Figure 13. From Figure 13 (a) and (c), both standard RK approximation and piecewise linear (PL) reconstruction capture the smooth function (101) very well, while the piecewise constant (PC) reconstruction forms discontinuities. For the non-smooth function (102), as shown in Figure 13 (b) and (d), both PC and PL reconstructions capture the discontinuity, whereas the standard RK approximation shows numerical smearing effects. Although PC reconstruction captures the discontinuity, it shows poor approximation accuracy for the smooth region away from the discontinuity due to the low order (zero-th order) approximation adopted in PC reconstruction. The PL method, on the other hand, works well for both smooth and non-smooth functions. The convergence behaviors of the RK approximation with PC and PL reconstructions are shown in Figure 14, where h is the average nodal spacing of the discrete RK nodes, and numbers in the legends denote convergence rates. It can be seen only the PL reconstruction shows optimal convergence rates in all cases.

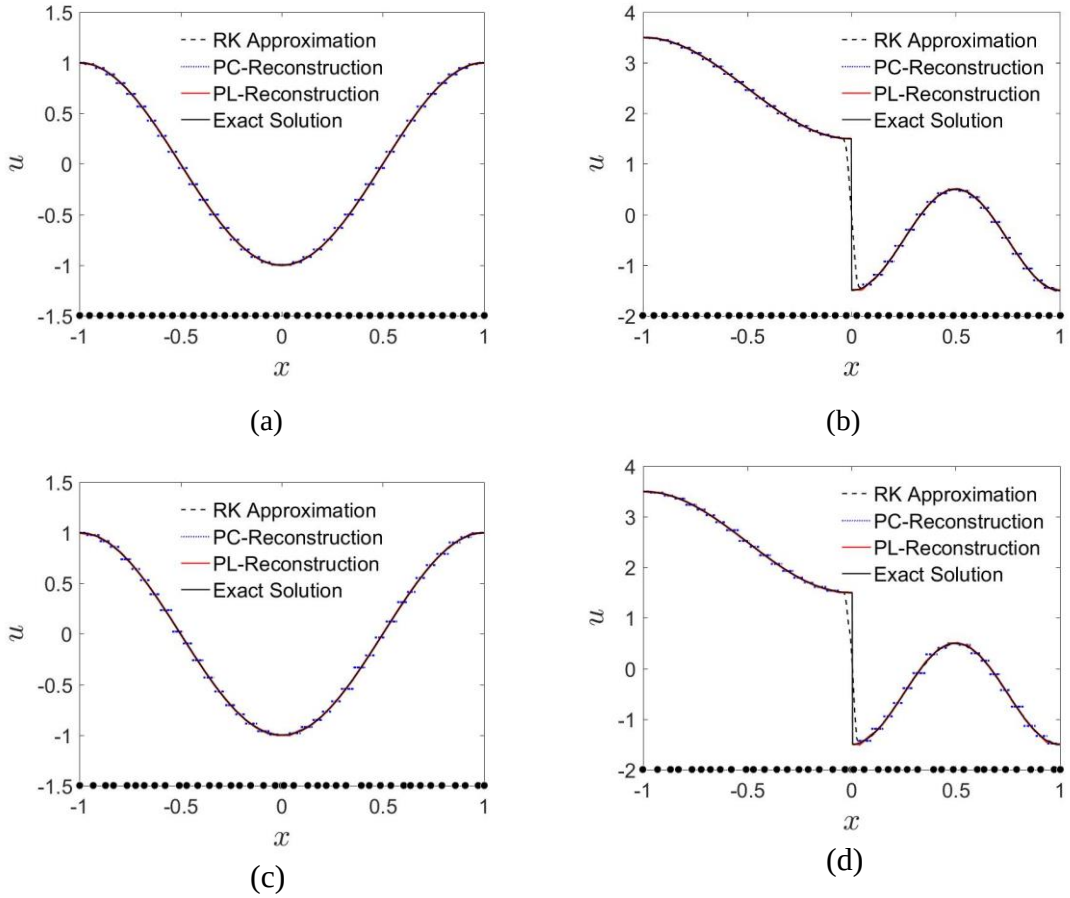
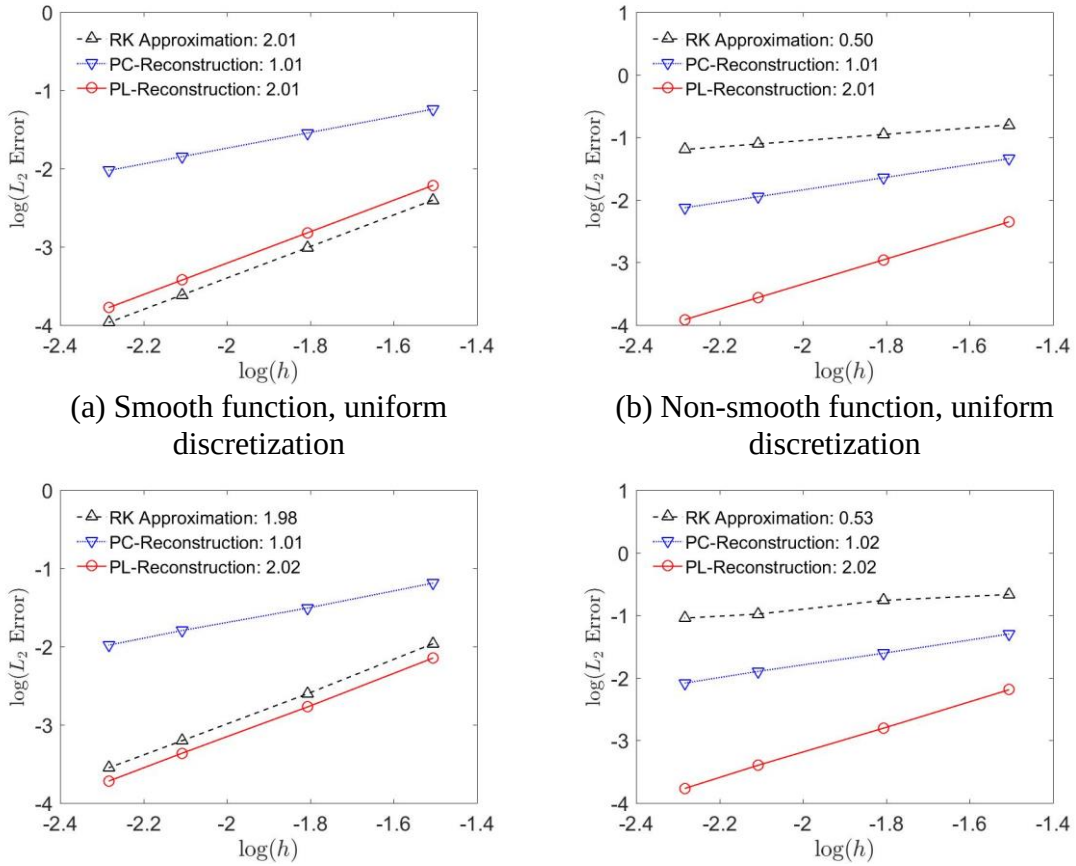


Figure 13. Approximate solution from RK approximation with PC and PL reconstruction comparing to exact solution. (a) Smooth function, uniform discretization, (b) Non-smooth function, uniform discretization, (c) Smooth function, non-uniform discretization, (d) Non-smooth function, non-uniform discretization.



(a) Smooth function, uniform discretization

(b) Non-smooth function, uniform discretization

(c) Smooth function, non-uniform discretization

(d) Non-smooth function, non-uniform discretization

Figure 14. Accuracy of RK approximation with PC and PL reconstruction. The numbers in the legends denote convergence rates.

4.2.3 Numerical Procedures of MUSCL-SCNI

Introducing the Riemann-enriched pressure flux divergence in Eqns. (82), (83) and the advection flux divergence in Eqns. (86) and (87) into Eq. (93), and considering the arbitrariness of test functions, the following semi-discrete equations are obtained at time step $n+1$:

$$\boldsymbol{M}_{IJ}\left(\dot{\boldsymbol{U}}_J\right)_{n+1} = \left(\boldsymbol{F}_I^{ext}\right)_{n+1} - \left(\boldsymbol{F}_I^{int}\right)_{n+1} \tag{104}$$

where $\boldsymbol{M}_{IJ}$ is the mass matrix, $\boldsymbol{F}_I^{ext}$ is the external force and $\boldsymbol{F}_I^{int}$ is the internal force. By further applying the central difference discretization:

$$\boldsymbol{U}_{n+1} = \boldsymbol{U}_n + 0.5\Delta t\dot{\boldsymbol{U}}_n + 0.5\Delta t\dot{\boldsymbol{U}}_{n+1} \tag{105}$$

where Δt is the time step size, the following predictor-corrector form of the explicit time stepping procedure is obtained:

$$\boldsymbol{U}_{n+1}^{predict} = \boldsymbol{U}_n + 0.5\Delta t\dot{\boldsymbol{U}}_n \tag{106}$$

$$\left(\dot{\boldsymbol{U}}_I\right)_{n+1} = \boldsymbol{M}_{II}^{-1}\left((\boldsymbol{F}_I^{ext})_{n+1}^{predict} - \left(\boldsymbol{F}_I^{int}\right)_{n+1}^{predict}\right) \tag{107}$$

$$\boldsymbol{U}_{n+1} = \boldsymbol{U}_{n+1}^{predict} + 0.5\Delta t\dot{\boldsymbol{U}}_{n+1} \tag{108}$$

where $\boldsymbol{U}_{n+1}^{predict}$ consists of the velocity, mass density, and total energy density $\left(\boldsymbol{u}_{n+1}^{predict}, \rho_{n+1}^{predict}, E_{n+1}^{predict}\right)$ at the predictor stage, and the predictor of pressure can be obtained by EOS (69), i.e., $p_{n+1}^{predict} = p\left(\boldsymbol{u}_{n+1}^{predict}, \rho_{n+1}^{predict}, E_{n+1}^{predict}\right)$. The internal force $\left(\boldsymbol{F}_I^{int}\right)_{n+1}^{predict}$ in Eq. (107) is expressed as

$$\begin{aligned}\left(\boldsymbol{F}_I^{int}\right)_{n+1}^{predict} = &\sum_{N=1}^{NP} \boldsymbol{\Psi}_I^T(\boldsymbol{x}_N) \\ &\cdot \sum_{K\in S_N}\left(\left(\boldsymbol{F}_p^*(\tilde{\boldsymbol{x}}_N^K)\right)_{n+1}^{predict} + \left(\boldsymbol{F}_{adv}^{\mathrm{mup}*}(\tilde{\boldsymbol{x}}_N^K)\right)_{n+1}^{predict}\right)\cdot \boldsymbol{n}_N^K A_N^K\end{aligned} \tag{109}$$

where $\left(\boldsymbol{F}_p^*(\tilde{\boldsymbol{x}}_N^K)\right)_{n+1}^{predict}$ and $\left(\boldsymbol{F}_{adv}^{\mathrm{mup}*}(\tilde{\boldsymbol{x}}_N^K)\right)_{n+1}^{predict}$ are the Riemann-enriched pressure flux in Eq. (83) and the advection flux in Eq. (87), respectively, evaluated based on the predictor $\left(\boldsymbol{u}_{n+1}^{predict}, \rho_{n+1}^{predict}, E_{n+1}^{predict}, p_{n+1}^{predict}\right)$. The external force $\left(\boldsymbol{F}_I^{int}\right)_{n+1}^{predict}$ in Eq. (107) is:

$$(\boldsymbol{F}_I^{ext})_{n+1}^{predict} = \sum_{N=1}^{NP} \boldsymbol{\Psi}_I^T(\boldsymbol{x}_N)\cdot \boldsymbol{F}_b^h(\boldsymbol{x}_N)V_N + \sum_{N=1}^{NS} \boldsymbol{\Psi}_I^T(\boldsymbol{x}_N)\cdot\left(\boldsymbol{F}_t^h(\boldsymbol{x}_N)\right)_{n+1}^{predict} A_N \tag{110}$$

where NS denotes the number of evaluation points on the natural boundary, and $\boldsymbol{M}_{IJ}$ is the lumped generalized mass matrix:

$$\boldsymbol{M}_{IJ} = \begin{cases} \mathbf{0}, & \text{if } I \neq J \\ \sum_{N=1}^{NP} \boldsymbol{\Psi}_I^T(\boldsymbol{x}_N)\boldsymbol{I}V_N, & \text{if } I = J \end{cases} \tag{111}$$

A computational flow chart of MUSCL-SCNI in Eqns. (106) - (108) is given in **Algorithm 1**.

Algorithm 1: Numerical procedure for MUSCL-SCNI

1. {%} Given initial conditions for velocity, density and total energy density: $\boldsymbol{u}_0, \rho_0, E_0$
2. Calculate initial pressure p_0 by EOS (Eq. (69))
3. Construct and store RK shape functions Ψ_L
4. Construct lumped mass matrix $\boldsymbol{M}$ from Eq. (111)
5. **for** $n = 1{:}\,nt$ **do** {%} Time marching
6. **Predictor**:
   $\boldsymbol{U}_{n+1}^{predict} = \boldsymbol{U}_n + 0.5\Delta t\dot{\boldsymbol{U}}_n$ (Eq. (106))
7. **for** $I = 1{:}\,NP$ **do** {%} Nodal integration loop over gradient smoothing cells
8. Obtain $p_I^h = p(\boldsymbol{u}_I^h, \rho_I^h, E_I^h)$ by EOS (Eq. (69))
9. Obtain PC-reconstruction $\bar{\boldsymbol{u}}_I, \bar{\rho}_I, \bar{p}_I, \bar{E}_I$ (Eqns. (74) - (77))
10. Obtain PL-reconstruction $\hat{\boldsymbol{u}}_I, \hat{\rho}_I, \hat{p}_I, \hat{E}_I$ based on $\bar{\boldsymbol{u}}_I, \bar{\rho}_I, \bar{p}_I, \bar{E}_I$ and $\widetilde{\boldsymbol{\nabla}}u_i^h, \widetilde{\boldsymbol{\nabla}}\rho^h, \widetilde{\boldsymbol{\nabla}}p^h, \widetilde{\boldsymbol{\nabla}}E^h$ (Eqns. (94), (98) - (100))
11. Obtain p^*, u^* at gradient smoothing points from Riemann solvers
12. Obtain smoothed flux divergence $\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^*, \widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^{\text{mup}*}$ (Eqns. (82), (86))
13. Obtain internal and external forces $(\boldsymbol{F}^{int})_{n+1}^{predict}$, $(\boldsymbol{F}^{ext})_{n+1}^{predict}$ (Eqns. (109) - (110))
14. **end for**
15. **Solve:** $\dot{\boldsymbol{U}}_{n+1} = \boldsymbol{M}^{-1}\left((\boldsymbol{F}^{ext})_{n+1}^{predict} - (\boldsymbol{F}^{int})_{n+1}^{predict}\right)$ by Eq. (107)
16. **Corrector**:
    $\boldsymbol{U}_{n+1} = \boldsymbol{U}_{n+1}^{predict} + 0.5\Delta t\dot{\boldsymbol{U}}_{n+1}$ (Eq. (108))
    $p_{n+1}^h = p(\boldsymbol{u}_{n+1}^h, \rho_{n+1}^h, E_{n+1}^h)$ by EOS (Eq. (69))
17. **Next time step**
18. **end for**

4.3 Numerical Examples

Several numerical examples are analyzed to examine the performance of the proposed MUSCL-SCNI approach for shock modeling. In all the tests, RK approximation is constructed using linear basis, cubic B-spline rectangular kernel, with normalized support size $\tilde{a} = 1.5$. Body forces and heat sources are neglected.

4.3.1 Sod Shock Tube Problem

The one-dimensional Sod shock tube problem is first tested to verify the performance of the MUSCL-SCNI scheme on modeling shock and rarefaction waves. As shown in Figure 15, a tube is filled with ideal gas that contains discontinuities in the pressure and density fields. The gas constant $\gamma = 1.4$, and the initial conditions are given in Eq. (112):

$$(u, \rho, p) = \begin{cases} (0.0, 1.0, 1.0) & 0.0 \le x < 0.5 \\ (0.0, 0.125, 0.1) & 0.5 \le x \le 1.0 \end{cases} \tag{112}$$

Zero traction boundary conditions are applied at both ends of the tube.

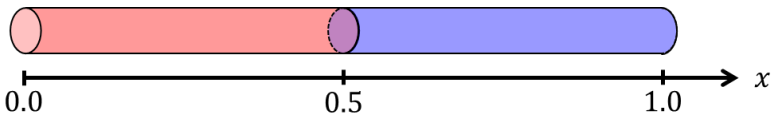


Figure 15. Illustration for Sod shock tube problem.

The tube is discretized into 201 nodes along the *x*-direction. The non-uniform discretization is generated by introducing random perturbations to nodal positions between $\pm 0.5h$ into the uniform discretization, where h is the nodal distance of the uniform discretizations. The final time for the simulation is $t = 2.0$ (dimensionless unit is used here), and the time step size is $\Delta t = 0.005$.

From Figure 16, it can be seen that the numerical prediction of MUSCL-SCNI agrees very well with the exact solutions, and it avoids the over-smearing of shock and rarefaction observed in Riemann-SCNI solutions. On the other hand, if the flux limiting algorithm described in Eqns. (96) and (97) is not adopted, then spurious overshoot and undershoot behaviors are observed in the solutions (denoted as "MUSCL-SCNI, w/o Limiting" in Figure 16).

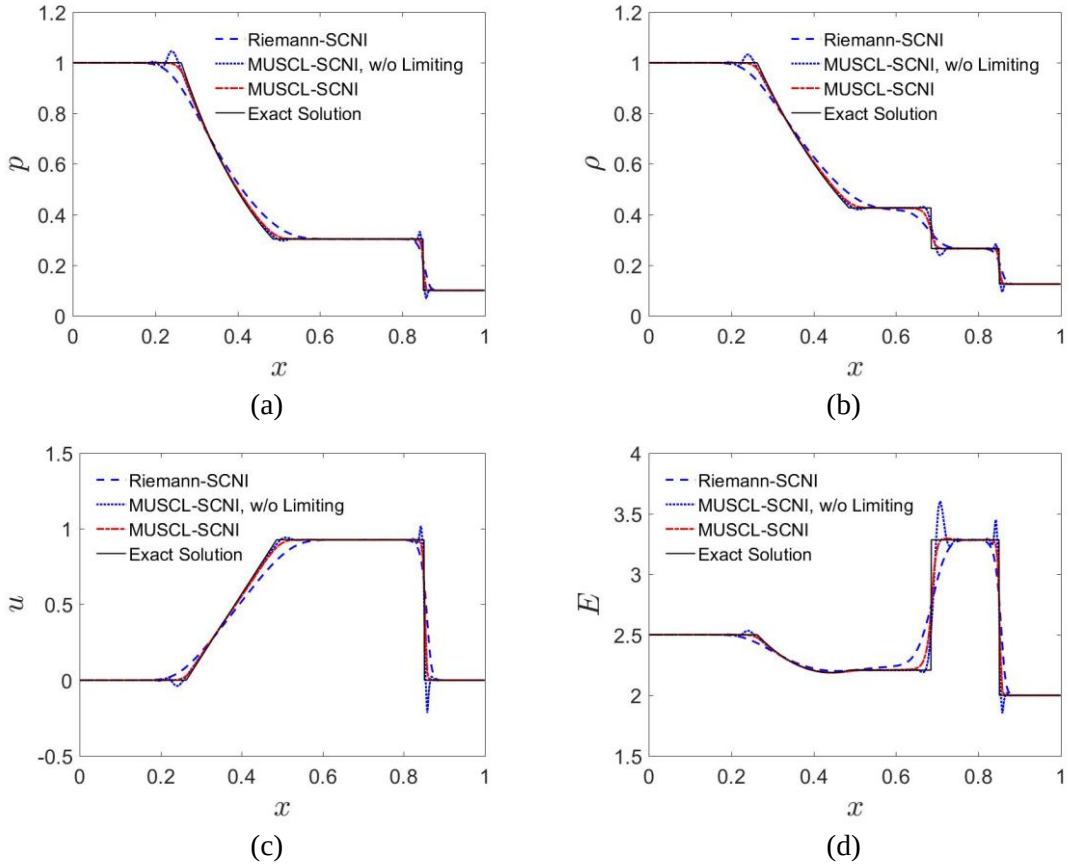


Figure 16. Numerical results at $t = 2.0$ of Sod problem in uniform discretization: (a) pressure. (b) density. (c) velocity. (d) total energy density.

The superior performance of MUSCL-SCNI is also demonstrated in the numerical results shown in Figure 17, where non-uniform discretizations are employed. Numerical results from WENO-LF 5th scheme (denoted as WENO5) [50] are also compared. The WENO5 scheme is implemented in the finite difference method with a total-variational-diminishing Runge-Kutta 3rd-order (TVD-

RK3) time integration method [148]. It can be seen that MUSCL-SCNI under non-uniform discretizations is very accurate in both rarefaction wave and shock regions, and it achieves a similar accuracy as WENO-LF 5th, but WENO-LF 5th is reported to be quite expensive [50].

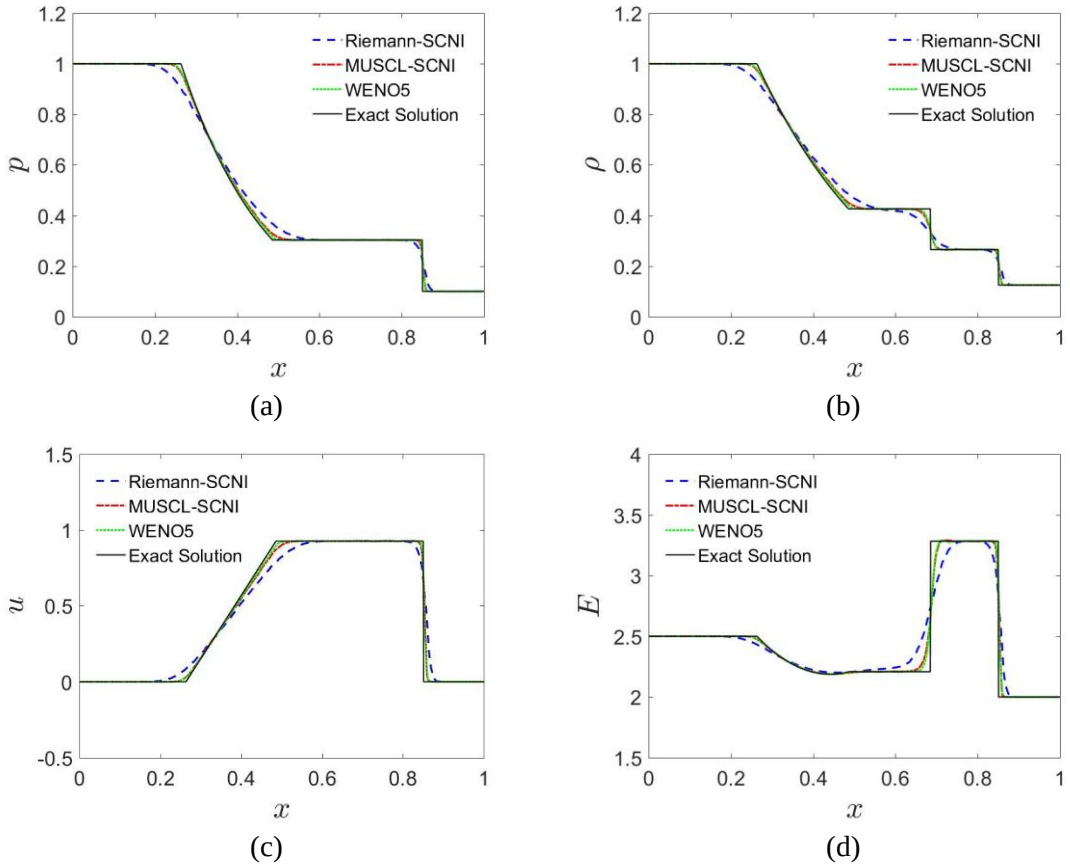
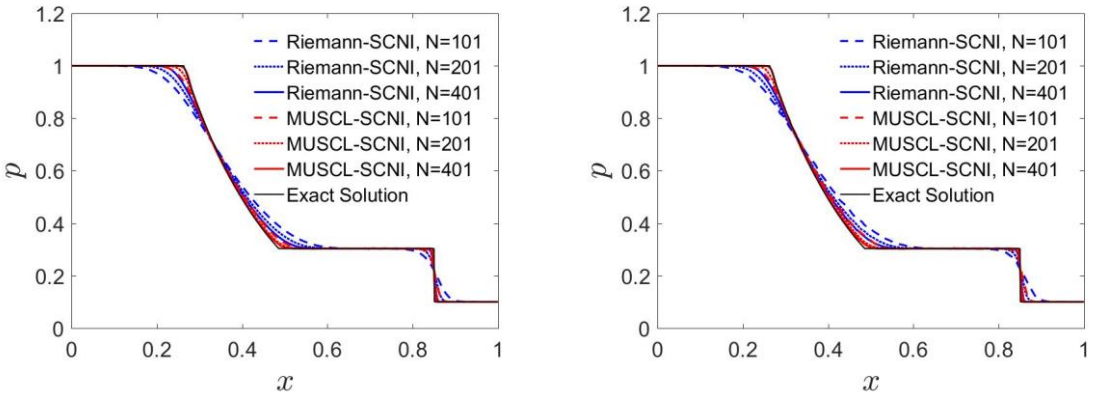


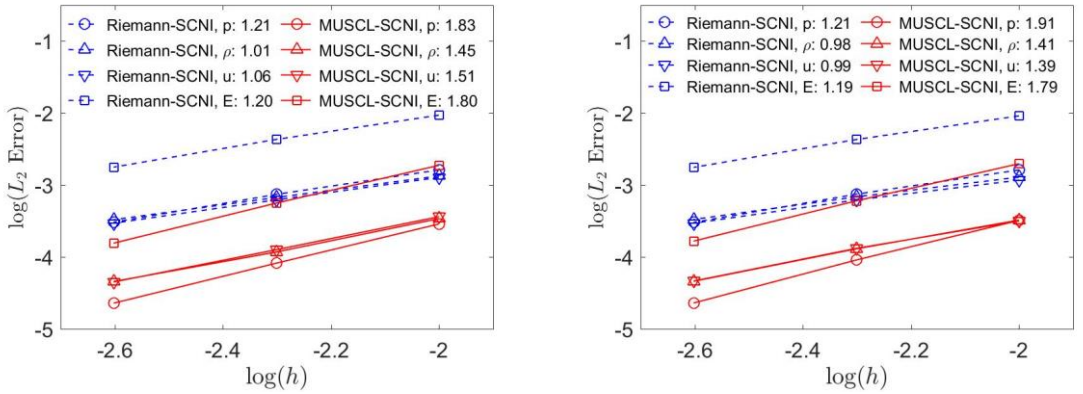
Figure 17. Numerical results at $t = 2.0$ of Sod problem in non-uniform discretization: (a) pressure. (b) density. (c) velocity. (d) total energy density

Lastly, convergence study is performed with three levels of model refinements. The results are shown in Figure 18 and Figure 19, where it clearly shows that MUSCL-SCNI achieves better convergence rates and accuracy in all the primary solution fields (i.e., velocity, mass density, total energy density and pressure) than Riemann-SCNI under both uniform and non-uniform discretizations.



(a) Uniform discretization (b) Non-uniform discretization

Figure 18. Refinement study of Riemann-SCNI and MUSCL-SCNI for the Sod problem at t = *2.0*, under (a) uniform disrcretizations, (b) non-uniform discretizations. The numbers in the legends denote convergence rates.



(a) Uniform discretization (b) Non-uniform discretization

Figure 19. Convergence of Riemann-SCNI and MUSCL-SCNI for the Sod problem at t = *2.0*, under (a) uniform disrcretizations, (b) non-uniform discretizations. The numbers in the legends denote convergence rates.

4.3.2 Odd-Even Decoupling Problem

The odd-even decoupling problem [52] is modeled to test the sensitivity of the proposed formulation with respect to nodal distributions. As illustrated in Figure 20, a shock wave propagates along the negative x_1 direction in an idea gas (ideal gas constant $\gamma = 1.4$) in hypersonic flow with Mach number, $M = 6.0$. The final time $t = 0.4$ (dimensionless unit is used

here) and the time step size $\Delta t = 2.0 \times 10^{-6}$ are considered, and the initial conditions are given in Eq. (113):

$$(u_1, u_2, \rho, p) = \begin{cases} (1.0, 0.0, 6.0, 1.0) & 0.0 \le x_1 < 1.5 \\ (5.25,\ 0.0,\ 0.656,\ 40.64) & 1.5 \le x_1 \le 3.0 \end{cases} \tag{113}$$

Zero traction boundary conditions are applied at the boundaries.

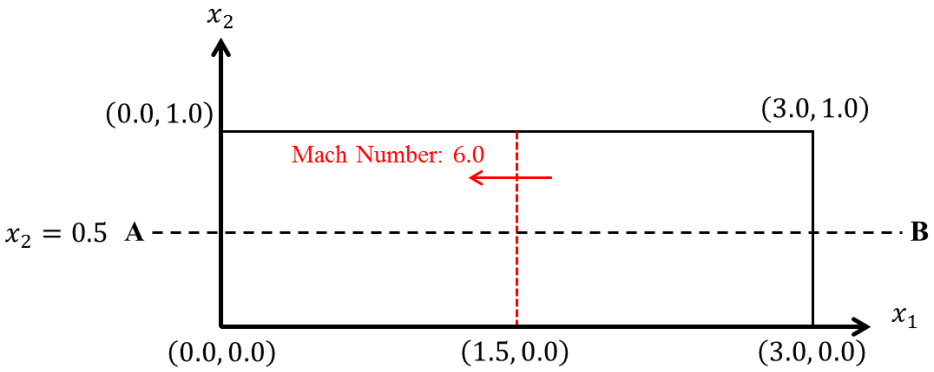


Figure 20. Illustration for the odd-even problem.

The domain is first discretized uniformly with nodal spacing $h_2 = 0.0125$ and $h_1 = 4h_2$, and then the x_2 coordinate of nodes located on the symmetric axis line A-B ($x_2 = 0.5$) are perturbed:

$$x_2 = \begin{cases} 0.5 + 1.0 \times 10^{-4} & \text{even nodes in } x_{1I} \\ 0.5 - 1.0 \times 10^{-4} & \text{odd nodes in } x_{1I} \end{cases} \tag{114}$$

This perturbation often causes numerical instabilities in shock modeling, known as the carbuncle effect [52].

Table 1. Numerical schemes based on different treatments for the pressure and advection flux divergence

	Divergence of Pressure Flux	Divergence of Advection Flux
Scheme 1	$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^h$	$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^h$
Scheme 2	$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^*$	$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^h$
Scheme 3	$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^*$	$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^{\mathrm{up}*}$
Scheme 4	$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^h$	$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^{\mathrm{mup}*}$
Scheme 5 (proposed method)	$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^*$	$\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^{\mathrm{mup}*}$

In the present study, 5 numerical schemes with different treatments for the pressure/advection flux divergence are studied and compared to examine the influence of the adopted Riemann-enriched smoothed flux divergence. These schemes are listed in Table 1, where

(1) $\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^h$ denotes the pressure flux divergence in Eq. (65) without Riemann-enrichment;

(2) $\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_p^*$ denotes the Riemann-enriched pressure flux divergence in Eqns. (82) - (83);

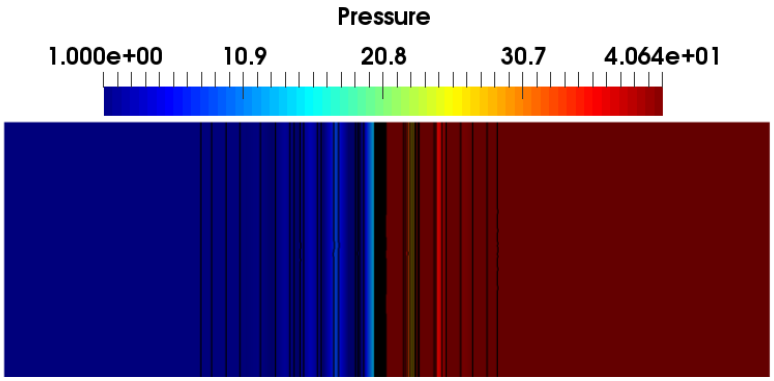
(3) $\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^h$ denotes the advection flux divergence in Eq. (66) without Riemann-enrichment;

(4) $\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^{\mathrm{up}*}$ denotes the Riemann-enriched advection flux divergence using the conventional upwind approximation in Eqns. (84) - (85);

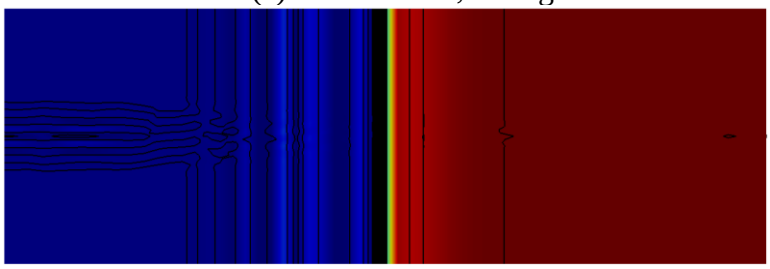
(5) $\widetilde{\boldsymbol{\nabla}} \cdot \boldsymbol{F}_{adv}^{\mathrm{mup}*}$ denotes the Riemann-enriched advection flux divergence using the modified upwind approximation in Eqns. (86) - (87).

The numerical results of all 5 schemes at time $t = 0.08$ are shown in Figure 21. As can be seen, Scheme 1 (without any Riemann-enrichment) and Scheme 2 (with Riemann-enrichment for the pressure flux only) both suffer from remarkable oscillations, and as a result, their solutions

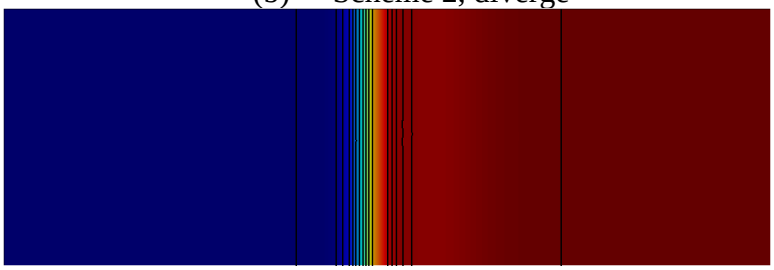
diverge after $t = 0.08$. Scheme 3 is numerically stable, but the shock front is over-smeared due to the strong dissipations of the conventional upwind scheme, as shown in Figure 21 (c). The modified upwind scheme provides less smeared results for Scheme 4 shown in Figure 21 (d), but strong oscillations at post-shock wave area show up as Riemann enrichment is not applied to the pressure flux. The proposed MUSCL-SCNI formulation (denoted as Scheme 5 here), on the other hand, is stable and captures the sharp shock front very well, as shown in Figure 21 (e). Similar behaviors are observed at final time of $t = 0.20$ in Figure 22 (a) to (c), without the carbuncle effect noted in [52]. Figure 23 compares pressure distributions obtained from different schemes along the line $x_2 = 0.5$, which clearly shows the accuracy and stability performance of the proposed Scheme 5. The above comparison verifies that the Riemann-enriched flux divergence with modified upwind scheme is key to achieve stable solutions without over-smearing the shock fronts.



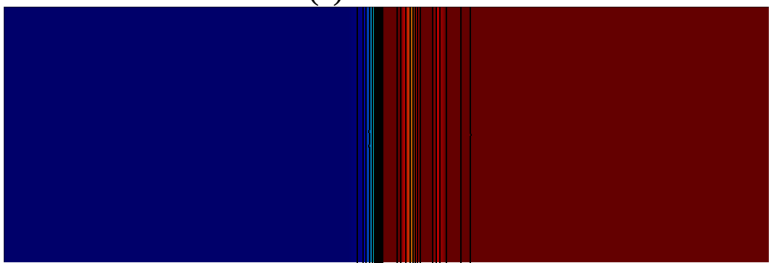
(a) Scheme 1, diverge



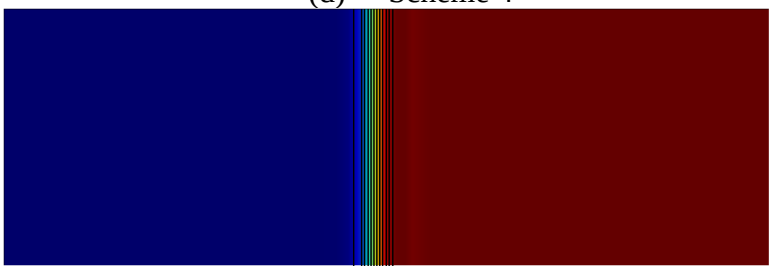
(b) Scheme 2, diverge



(c) Scheme 3

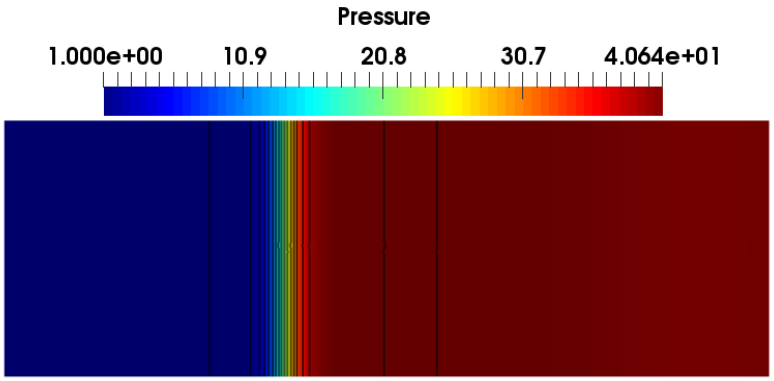


(d) Scheme 4

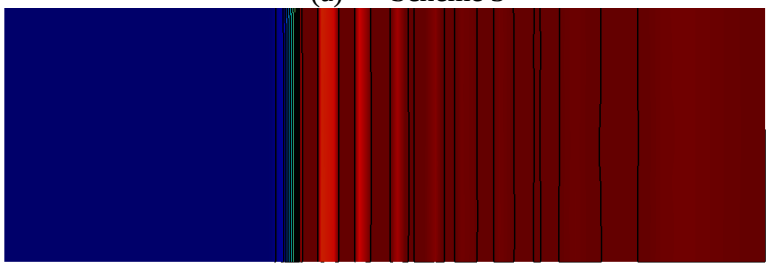


(e) Scheme 5

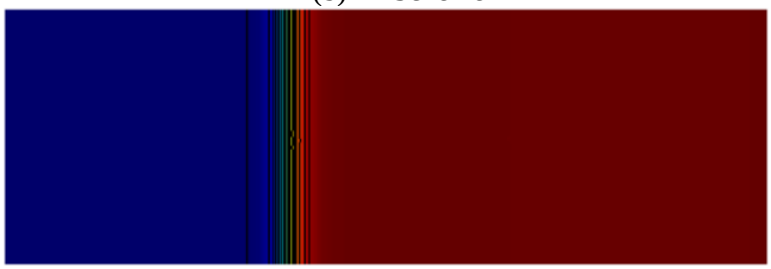
Figure 21. Shock wave pattern at $t = 0.08$ in the odd-even decoupling problem using 5 different schemes (Table 1). The dark curves denote the mass density contours ranging from 1.00 to 5.25.



(a) Scheme 3

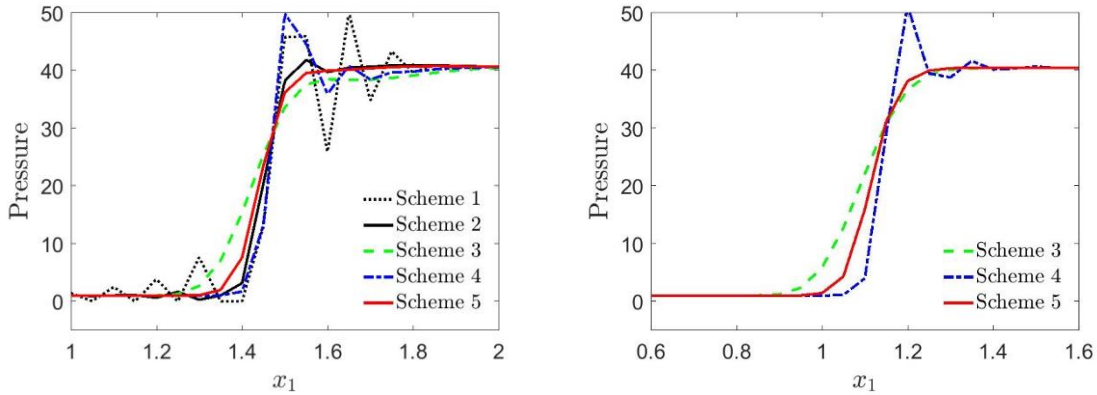


(b) Scheme 4



(c) Scheme 5

Figure 22. Shock wave pattern at $t = 0.20$ in the odd-even decoupling problem using scheme 3, scheme 4, and scheme 5 (Table 1). The dark curves denote the mass density contours ranging from 1.00 to 5.25.



(a) $t = 0.08$ (b) $t = 0.20$

Figure 23. Pressure distribution using 5 different schemes (Table 1) along the line $x_2 = 0.5$.

4.3.3 Schardin's Problem

As illustrated in Figure 24 (a), the Schardin's problem [149] considers a planar shock moving along x_1 direction which impinges on an equilateral triangle rigid wedge and generates scattered shocks and vortices. The domain is discretized into 37136 nodes as shown in Figure 24 (b). The ideal gas EOS is considered for the flow with ideal gas constant $\gamma = 1.4$. The initial conditions are given in Eq. (115) as

$$(u_1, u_2, \rho, p) = \begin{cases} \left(202\frac{\mathrm{m}}{s}, 0\frac{\mathrm{m}}{s}, 1.942\frac{\mathrm{kg}}{\mathrm{m}^3}, 0.096\mathrm{MPa}\right) & 0 \le x_1 \le 0.025\ \mathrm{m} \\ \left(0\frac{\mathrm{m}}{s}, 0\frac{\mathrm{m}}{s}, 1.225\frac{\mathrm{kg}}{\mathrm{m}^3}, 0.050\mathrm{MPa}\right) & 0.025\ \mathrm{m} < x_1 \end{cases} \tag{115}$$

The zero traction boundary conditions are applied on the outer boundaries of the fluid domain, and the reflecting boundary condition is applied at the boundaries of triangle wedge (Figure 24 (a)) by solving the Riemann problem with reflecting velocity condition. To do so, the reflecting velocity $\boldsymbol{u}_{\mathrm{reflect}}$ based on the approximated flow velocity $\boldsymbol{u}^h(\boldsymbol{x}_I, t)$ is defined on the reflecting boundary as:

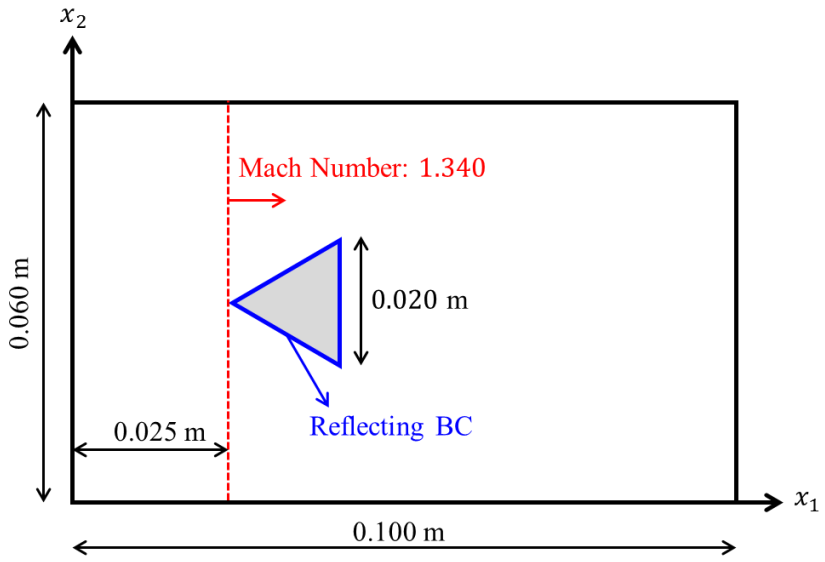
$$\begin{aligned} u_{\mathrm{reflect}} &= u^h(x_I, t) - 2\Big(u^h(x_I, t) \cdot n(x_I)\Big)n(x_I), \\ u_{\mathrm{reflect}} &= u_{\mathrm{reflect}} \cdot n(x_I), \qquad \forall x_I \in \partial\Omega_{\mathrm{reflecting}} \end{aligned} \tag{116}$$

Then, the pressure jump condition is reformulated at the reflecting boundary as:

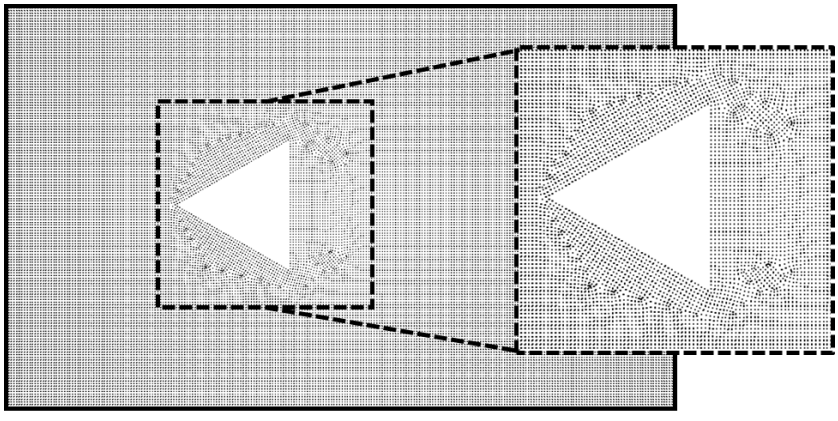
$$\begin{aligned} (p^* - \bar{p}_I) &= \bar{\rho}_I S(u^* - \bar{u}_I) \\ (p^* - \bar{p}_I) &= \bar{\rho}_I S(u^* - u_{\mathrm{reflect}}) \end{aligned} \tag{117}$$

The Riemann solutions (p^*, u^*) by solving jump conditions (117), shock Hugoniot curve (80) and entropy condition (58) are used for calculating the pressure and advection flux in (82), and (84) at the wedge boundaries. The final time and the time step size are chosen as $180\ \mu s$ and $\Delta t = 0.02\ \mu s$, respectively.

The numerical prediction and experimental observation [150] of the pressure fields are shown in Figure 25. As can be seen, the shock wave is diffracted at the right vertices of the triangular wedge at $t = 53\ \mu s$. Then, vortices are generated near back vertices of the wedge at $t = 102\ \mu s$ and $130\ \mu s$. At $t = 172\ \mu s$, the discrete vortex-lets formed along the slip layer of the main vortex are observed. MUSCL-SCNI effectively captures these complex physical processes of shock diffraction, vortex generation and shock-vortex interaction, and the simulation results agree quite well with the experimental observation



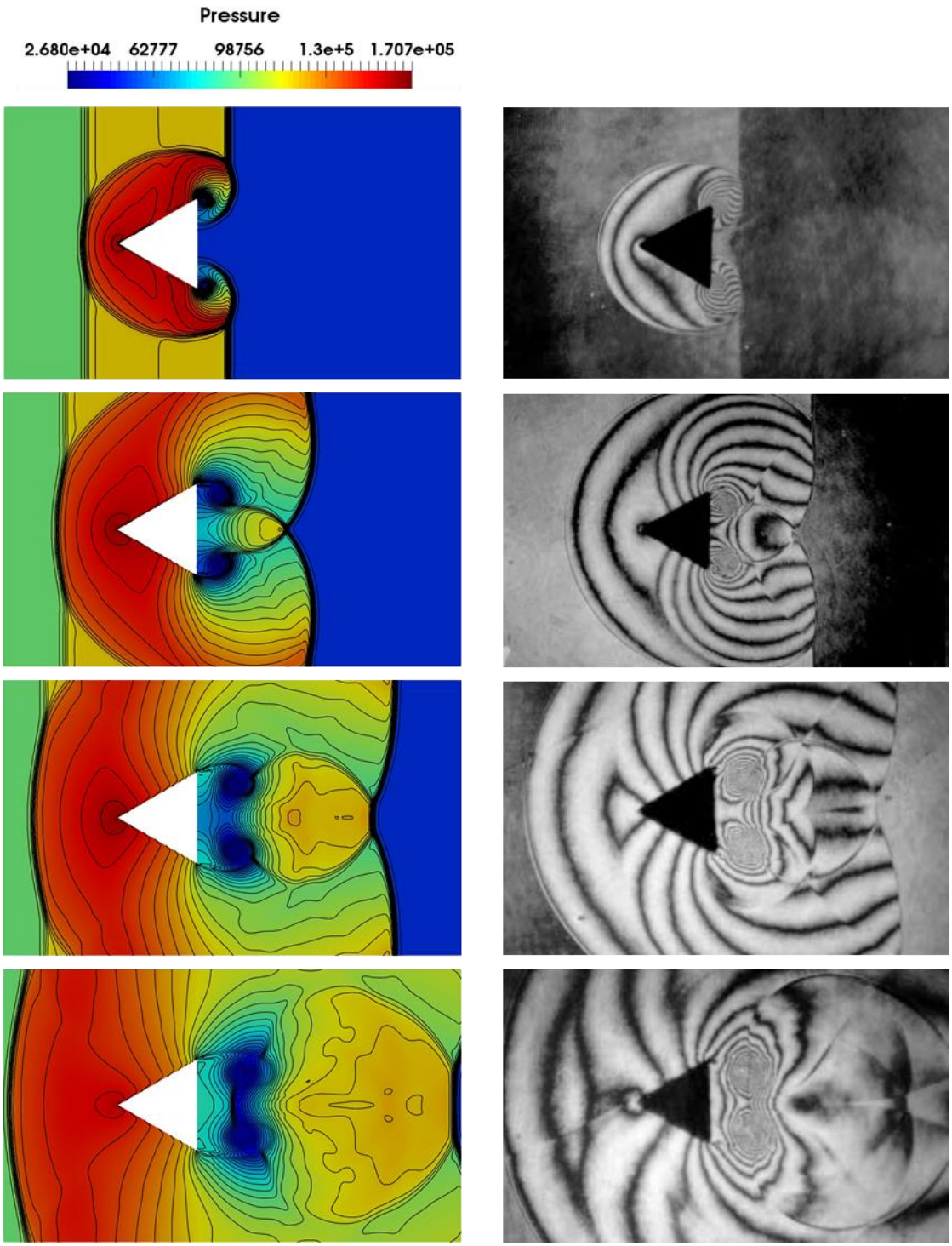
(a)



Domain Discretization

(b)

Figure 24. (a) Problem illustration and (b) domain discretization for Schardin's problem.



(a) Numerical results (b) Experimental results

Figure 25. The numerical and experimental results of the pressure field for Schardin's problem. Top-down: results at $53\ \mu s$, $102\ \mu s$, $130\ \mu s$, and $172\ \mu s$. In numerical results, the dark curves are the mass density contours ranging from 0.08 kg/m^3 to 0.28 kg/m^3. The experiment results are snapshots of density field by interferograms, obtained from [150].

4.3.4 Detonation Problem

To further examine the performance of the present method for modeling shock waves, 1D and 2D high energy explosion problems are tested. The Jones–Wilkins–Lee (JWL) EOS [151] for high energy explosive gas product is employed:

$$p(\rho, e) = \rho\omega e + \sum_{i=1}^{2} A_i \left(1 - \frac{\omega\rho}{R_i\rho_0}\right) \exp\left(-\frac{R_i\rho_0}{\rho}\right), \qquad e = E - \frac{1}{2}\|\boldsymbol{u}\| \tag{118}$$

and ω, A_1, A_2, R_1, R_2 are material constants, and ρ_0 is the bulk density of the solid explosive. The Chapman–Jouguet (CJ) model [152, 153] for the detonation process is adopted, which assumes that the reacted gas product from high explosive reaches CJ point of the phase diagram (as shown in Figure 26). The chemical reaction in high explosive is neglected in this study.

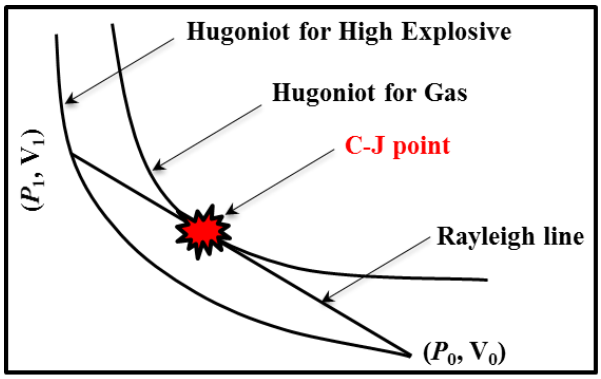


Figure 26. The phase diagram of high explosive detonation process.

The detonation process is assumed to be steady-state with a constant detonation speed, D. Once the high explosive is reacted, the reacted gas velocity, density, pressure and sound speed reaches CJ point, denoted as u_{CJ}, ρ_{CJ}, p_{CJ}, and c_{CJ}. The CJ velocity u_{CJ} and density ρ_{CJ} can be approximated by the ideal gas expansion process:

$$\gamma = \frac{\rho_0 D^2}{p_{\mathrm{CJ}}} - 1, \ \ \rho_{\mathrm{CJ}} = \frac{\gamma + 1}{\gamma}\rho_0, \ \ u_{\mathrm{CJ}} = D - c_{\mathrm{CJ}} \tag{119}$$

where γ denotes the approximated gas constant, and ρ_0 is the bulk explosive mass density. The detonation speed D and CJ pressure p_{CJ} are obtained from the experimental data. The CJ model parameters for TNT and PBX9404 explosives with the JWL EOS are listed in Table 2 [154]:

Table 2. Typical parameters of explosives for JWL EOS.

Material	ρ_0 (kg/m^3)	D (m/s)	p_{CJ} (GPa)	A_1 (GPa)	A_2 (GPa)	R_1	R_2	ω
TNT	1630	6930	21.0	373.8	3.747	4.15	0.90	0.35
PBX9404	1842	8795	37.0	852.4	18.020	4.60	1.30	0.38

A one-dimensional detonation problem, as illustrated in Figure 27, is first tested, where the left-hand-side wall is the ignition point for TNT. The domain is discretized into 501 nodes. The final time and the time step size are chosen as $14\ \mu s$ and $\Delta t = 2.5 \times 10^{-3}\ \mu s$, respectively. The initial velocity of the TNT is zero. The zero traction boundary conditions are applied on the right end of TNT and the reflecting boundary condition is applied to the left end of the TNT. The numerical results of MUSCL-SCNI are shown in Figure 28, where the detonation process is effectively simulated. The results agree well with the analytical solutions from Beltman and Shepherd [155].

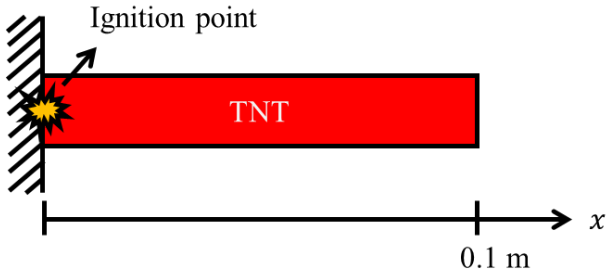


Figure 27. Illustration for the one-dimensional TNT detonation problem.

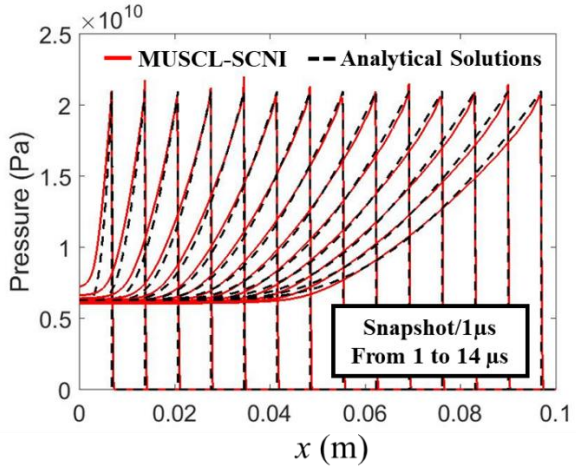


Figure 28. Numerical result of one-dimensional TNT detonation problem.

Next, the double ignition of a square high explosive PBX9404 [156] is modelled. As illustrated in Figure 29, the domain is discretized into 13455 nodes. The final time and the time step size are $16.8\,\mu s$ and $\Delta t = 2 \times 10^{-3}\,\mu s$, respectively. The initial velocity of the PBX9404 is zero and the zero traction boundary conditions are applied for all domain boundaries.

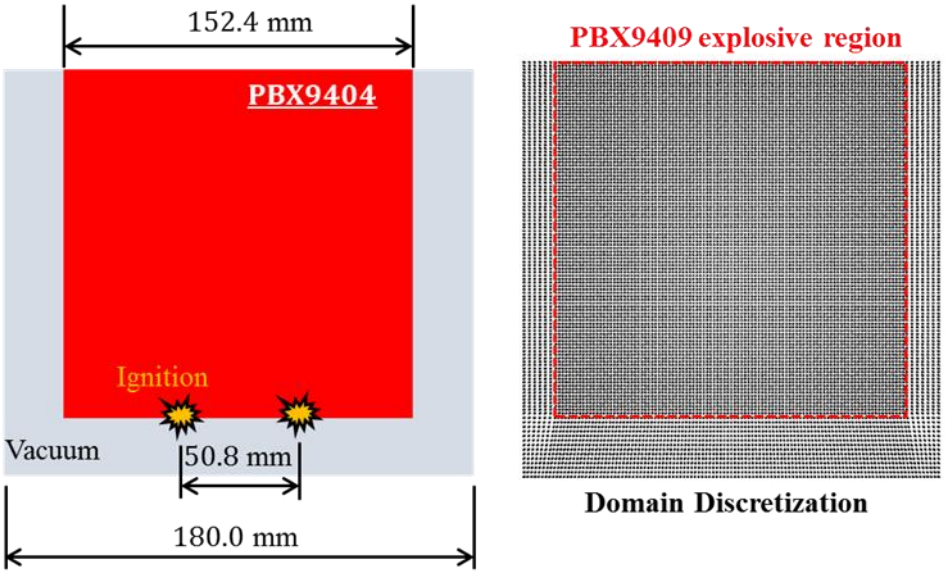
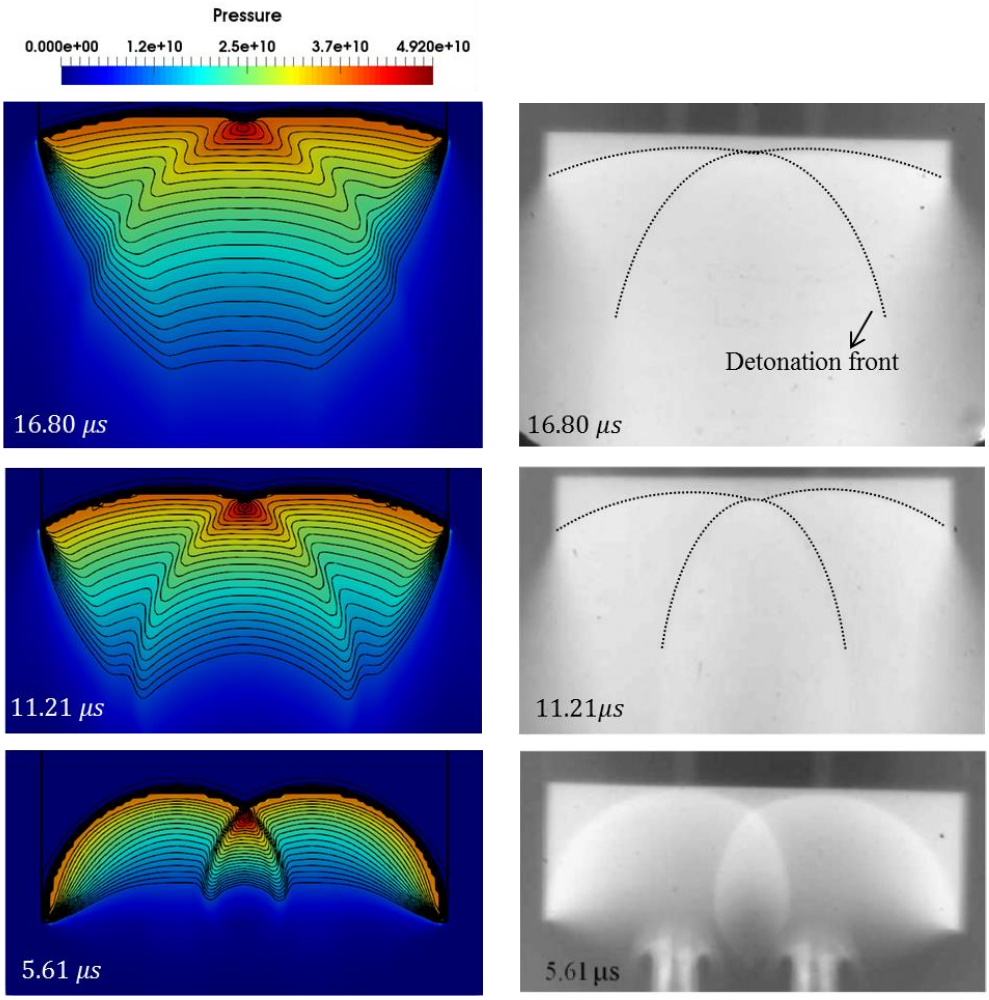


Figure 29. Problem settings and domain discretization for PBX9404 detonation problem.

As shown in Figure 30, two uniform detonation waves propagate radially and form a seagull-like detonation front. Later, the detonation front becomes planar, and the pressure pulse at the

overlapped detonation front is amplified, leading to high pressure of approximately 50 GPa at the overlapped area. The pressure amplification phenomenon in the cross-detonation area and the seagull-like detonation fronts are well captured [156] in the numerical simulation by the proposed MUSCL-SCNI formulation.



(a) (b)

Figure 30. Pressure distribution of PBX9404 detonation problem. (a) Numerical results by MUSCL-SCNI. The dark curves are mass density contours ranging from 1600 kg/m^3 to 2800 kg/m^3. (b) Experimental results from [156].

4.4 Summary of the MUSCL-SCNI for Shock Wave Modeling in Fluid Problems

In this study, a robust MUSCL-SCNI approach is developed for RKPM modeling of shock waves in compressible flow. The conservation equations are discretized by RK approximation, which naturally avoid numerical difficulties associated with low-quality meshes. The essential physics of shocks in fluids, including the Rankine-Hugoniot jump conditions and the entropy condition, are captured through embedding local Riemann enrichment under the stabilized conforming nodal integration (SCNI) framework. In addition, a flux splitting scheme is introduced to allow proper numerical treatments for the pressure and advection fluxes. Corrected pressure and velocities from Riemann solvers are adopted to construct the smoothed pressure flux divergence, and a modified upwind scheme is further employed for the discretization of the advection flux to ensure proper numerical dissipations. To further enhance accuracy, the MUSCL approach is introduced in conjunction with an oscillation limiter, which avoids Gibbs phenomenon and at the same time ensures monotonic piecewise linear reconstruction in the smooth region.

The effectiveness of the proposed meshfree framework is verified by solving several benchmark numerical examples. As shown, the SCNI-based nodally integrated RKPM formulation does not show numerical instabilities in shock modeling when perturbations in nodal positions are present (known as the carbuncle effect). Compared to conventional RKPM shock formulations based on piecewise constant approximations, the present formulation achieves better accuracy in all primary solution fields (i.e., velocity, mass density, total energy density and pressure) under both uniform and non-uniform discretizations. The MUSCL-SCNI approach shows similar accuracy as WENO-LF 5^{th} scheme, but at a lower computational cost and is more straightforward to implement in multi-dimensional RKPM formulations. The numerical tests also verify that the

Riemann-enriched smoothed flux divergence with modified upwind scheme is a key factor for achieving stable solutions without over-smearing shock fronts. Overall, the proposed formulation is effective in modeling shock phenomena without over/undershoots, and it is free from tunable artificial parameters.

This chapter is a in part, is a reprint of the material as it appears in "TH. Huang, JS. Chen, H. Wei, MJ. Roth, JA. Sherburn, J. Bishop, M. Tupek, and E. Fang. A MUSCL-SCNI Approach for Meshfree Modeling of Shock Waves in Fluids. Computational Particle Mechanics 7.2 (2020): 329-350. DOI: https://doi.org/10.1007/s40571-019-00248-x.". The dissertation author was the primary investigator and author of this paper.

Chapter 5 A Variational Multiscale Immersed Meshfree Method for Heterogeneous Materials

This chapter introduces the new Variational Multiscale Immersed Method (VMIM) for modeling heterogeneous materials. The immersed method for the linear elasticity problem with volumetric constraint is introduced in Chapter 5.1. The variational multiscale immersed method is presented in Chapter 5.2. The numerical examples are tested and demonstrated in Chapter 5.3. Finally, a brief summary of the proposed method is given in Chapter 5.4. The application of VMIM for heterogeneous material diffusion problem is given in Appendix A.

5.1 Immersed Method for Heterogeneous Material Problem with Volumetric Constraint

5.1.1 Strong Form

For demonstration of concept, a two-dimensional solid with heterogeneous materials occupying a bounded domain $\Omega \subset \mathbb{R}^{d=2}, \Omega = \Omega^1 \cup \Omega^2$, is considered as shown in Figure 31, where $\partial\Omega_N^2$ and $\partial\Omega_D^2$ denotes the Neumann and Dirichlet boundary on boundary $\partial\Omega^2$, respectively, $\partial\Omega_N^2 \cup \partial\Omega_D^2 = \partial\Omega^2$ and $\partial\Omega_N^2 \cap \partial\Omega_D^2 = \emptyset$. The solid carries two perfectly bounded materials occupying subdomains Ω^1 and Ω^2 with volumeless interface Γ. The compatibility and equilibrium of the solid are characterized by the continuity of displacement and stress equilibrium across the material interface Γ.

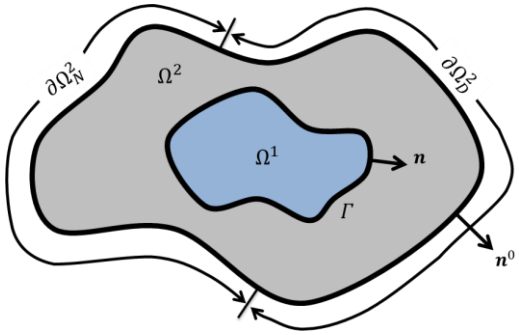


Figure 31. Illustration for two-dimensional solid with heterogeneous materials.

It can be easily seen that the conventional weak form of setting in Figure 31 requires a geometric conforming domain discretization and numerical integration for subdomains Ω^1 and Ω^2, which is difficult for problem involving complex geometry or evolving interfaces. Hence, an immersed setting of the Figure 31 can be achieved by employing the superposition:

the composite domain can be decoupled into a foreground domain $\Omega^+ = \Omega^1$ on top of a background domain $\Omega^- = \Omega^1 \cup \Omega^2$.

The strong form of the problem can be described by the following elliptic boundary value problem (BVP) for whole domain Ω

$$-\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(\boldsymbol{x}) = \boldsymbol{\mathcal{L}}(\boldsymbol{u}) = \boldsymbol{b}, \quad \forall \boldsymbol{x} \in \Omega \tag{120}$$

$$\boldsymbol{\sigma}(\boldsymbol{x}) \cdot \boldsymbol{n}^0 = \boldsymbol{t}, \quad \forall \boldsymbol{x} \in \partial\Omega_N \tag{121}$$

$$\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{g}, \quad \forall \boldsymbol{x} \in \partial\Omega_D \tag{122}$$

$$\boldsymbol{\sigma}(\boldsymbol{x}) = \boldsymbol{C}(\boldsymbol{x}) \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}) \tag{123}$$

where $\boldsymbol{u}(\boldsymbol{x})$ is the displacement fields, $\boldsymbol{\varepsilon}(\boldsymbol{u})$ is the strain tensor operator following the Voigt notation, $\boldsymbol{n}^0$ is the surface normal vectors corresponding to boundary $\partial\Omega$, $\boldsymbol{b}$ is the body force, $\boldsymbol{t}$ and $\boldsymbol{g}$ denote the prescribed traction and displacement on $\partial\Omega_N$ and $\partial\Omega_D$, respectively, $\boldsymbol{\sigma}$ is the Cauchy stress tensor. The body force $\boldsymbol{b}(\boldsymbol{x})$ and material elastic tensor $\boldsymbol{C}(\boldsymbol{x})$ can exhibit discontinuities across adjacent subdomains, but have smooth distribution in each subdomain Ω^+ and $\Omega^- \backslash \Omega^+$ respectively, given as

$$\boldsymbol{b} = \begin{cases} \boldsymbol{b}^+ & \text{in} \quad \Omega^+ \\ \boldsymbol{b}^- & \text{in} \quad \Omega^- \backslash \Omega^+ \end{cases}, \qquad \boldsymbol{C} = \begin{cases} \boldsymbol{C}^+ & \text{in} \quad \Omega^+ \\ \boldsymbol{C}^- & \text{in} \quad \Omega^- \backslash \Omega^+ \end{cases} \tag{124}$$

In this study, $\boldsymbol{C}^+$ and $\boldsymbol{C}^-$ are considered to be isotropic linear elasticity tensor. Due to the discontinuities across adjacent subdomains, the homogeneous Dirichlet and Neumann jump conditions are applied on the interface Γ:

$$[\![\boldsymbol{u}]\!] = \boldsymbol{0}, \quad \forall \boldsymbol{x} \in \Gamma \tag{125}$$

$$[\![\boldsymbol{\sigma} \cdot \boldsymbol{n}]\!] = \boldsymbol{0}, \quad \forall \boldsymbol{x} \in \Gamma \tag{126}$$

$$[\![\boldsymbol{q}]\!](\boldsymbol{x}) = \boldsymbol{q}^{+}(\boldsymbol{x}) - \boldsymbol{q}^{-}(\boldsymbol{x}) \tag{127}$$

where $[\![\cdot]\!]$ is defined as the jump operator and $\boldsymbol{n}$ is the surface normal vectors corresponding to interface Γ.

5.1.2 Weak Form

With Eqns. (120) to (127), the weak form of this problem can be constructed as follows:

$$\begin{aligned}\left(\delta\boldsymbol{u}, \boldsymbol{\mathcal{L}}(\boldsymbol{u})\right)_{\Omega} &= \left(\delta\boldsymbol{u}^{-}, \boldsymbol{\mathcal{L}}^{-}(\boldsymbol{u}^{-})\right)_{\Omega^{-}\setminus\Omega^{+}} + \left(\delta\boldsymbol{u}^{+}, \boldsymbol{\mathcal{L}}^{-}(\boldsymbol{u}^{+})\right)_{\Omega^{+}} \\ &= \left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\boldsymbol{u}^{-})\right)_{\Omega^{-}\setminus\Omega^{+}} + \left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^{+}), \boldsymbol{C}^{+}\cdot\boldsymbol{\varepsilon}(\boldsymbol{u}^{+})\right)_{\Omega^{+}} \\ &\quad - (\delta\boldsymbol{u}^{-}, \boldsymbol{t})_{\partial\Omega_{N}^{-}} - (\delta\boldsymbol{u}^{-}, [\![\boldsymbol{\sigma}\cdot\boldsymbol{n}]\!])_{\Gamma} \\ &= (\delta\boldsymbol{u}^{-}, \boldsymbol{b}^{-})_{\Omega^{-}\setminus\Omega^{+}} + (\delta\boldsymbol{u}^{+}, \boldsymbol{b}^{+})_{\Omega^{+}}\end{aligned} \tag{128}$$

where $(\cdot,\cdot)$ denotes the L_2 inner product. $\boldsymbol{u}^{+}$ and $\boldsymbol{u}^{-}$ are the displacement fields on the foreground and background domains, respectively. $\boldsymbol{\mathcal{L}}^{-}(\cdot) = -\boldsymbol{\nabla}\cdot\left(\boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\cdot)\right)$, $\boldsymbol{\mathcal{L}}^{+}(\cdot) = -\boldsymbol{\nabla}\cdot\left(\boldsymbol{C}^{+}\cdot\boldsymbol{\varepsilon}(\cdot)\right)$ are stress divergence operator acting on the foreground and background subdomains, respectively. The application of the Dirichlet jump condition (126) on interface Γ to the Eq. (128) yields the weak form: find $(\boldsymbol{u}^{+}, \boldsymbol{u}^{-}) \in \boldsymbol{\mathcal{U}}^{+}\times\boldsymbol{\mathcal{U}}^{-}, \boldsymbol{u}^{+} = \boldsymbol{u}^{-}$ on Γ, such that

$$\begin{aligned}&\left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\boldsymbol{u}^{-})\right)_{\Omega^{-}\setminus\Omega^{+}} + \left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^{+}), \boldsymbol{C}^{+}\cdot\boldsymbol{\varepsilon}(\boldsymbol{u}^{+})\right)_{\Omega^{+}} \\ &\qquad = (\delta\boldsymbol{u}^{-}, \boldsymbol{b}^{-})_{\Omega^{-}\setminus\Omega^{+}} + (\delta\boldsymbol{u}^{+}, \boldsymbol{b}^{+})_{\Omega^{+}} + (\delta\boldsymbol{u}^{-}, \boldsymbol{t})_{\partial\Omega_{N}^{-}}\end{aligned} \tag{129}$$

$\forall(\delta\boldsymbol{u}^{+}, \delta\boldsymbol{u}^{-}) \in \boldsymbol{\mathcal{V}}^{+}\times\boldsymbol{\mathcal{V}}^{-}$. The function spaces $\boldsymbol{\mathcal{U}}^{+}$, $\boldsymbol{\mathcal{U}}^{-}$, $\boldsymbol{\mathcal{V}}^{+}$, and $\boldsymbol{\mathcal{V}}^{-}$ are given in Eq. (130)

$$\begin{aligned}
\boldsymbol{\mathcal{U}}^+ &\equiv \{\boldsymbol{u}|\boldsymbol{u} \in [H^1(\Omega^+)]^d\} \\
\boldsymbol{\mathcal{V}}^+ &\equiv \{\boldsymbol{u}|\boldsymbol{u} \in [H^1(\Omega^+)]^d\} \\
\boldsymbol{\mathcal{U}}^- &\equiv \{\boldsymbol{u}|\boldsymbol{u} \in [H^1(\Omega^-)]^d, \boldsymbol{u} = \boldsymbol{g} \text{ on } \partial\Omega_D^-\} \\
\boldsymbol{\mathcal{V}}^- &\equiv \{\boldsymbol{u}|\boldsymbol{u} \in [H^1(\Omega^-)]^d, \boldsymbol{u} = \boldsymbol{0} \text{ on } \partial\Omega_D^-\}
\end{aligned} \tag{130}$$

The weak form in Eq. (129) is usually referred to as weak form for "embedded method", which employs the interfacial constraint "$\boldsymbol{u}^+ = \boldsymbol{u}^-$ on Γ" in solving the composite material system. However, the employment of interfacial constraint yields non-uniqueness of solution at the overlapping domain $\boldsymbol{u}^-(\boldsymbol{x} \in \Omega^+)$ [73], as well as the ill-conditioning issue [73]. To remedy this issue, a volumetric constraint approach can be introduced [76] as given in the next section.

5.1.3 Volumetric Constraint

The volumetric constraint for the overlapping domain is employed in this study and is defined as

$$[\![\boldsymbol{u}]\!] = \boldsymbol{u}^+ - \boldsymbol{u}^- = \boldsymbol{0} \text{ on } \Omega^+ \tag{131}$$

The employment of the volumetric constraint (131) for the Eq. (129) not only avoids the ill-conditioning issue, but also bypasses front tracking and contour integral along with material interfaces, where front tracking is difficult for problems involving complex geometry or evolving interfaces. By imposing $\boldsymbol{u}^+ = \boldsymbol{u}^-$ and $\delta\boldsymbol{u}^+ = \delta\boldsymbol{u}^-$ on Ω^+ strongly, the internal energy and body forces terms in Eq. (129) can be rewritten as

$$\begin{aligned}
&\left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^-), \boldsymbol{C}^- \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^-)\right)_{\Omega^- \setminus \Omega^+} + \left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^+), \boldsymbol{C}^+ \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^+)\right)_{\Omega^+} \\
&\quad = \left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^-), \boldsymbol{C}^- \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^-)\right)_{\Omega^-} + \left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^+), [\![\boldsymbol{C}]\!] \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^+)\right)_{\Omega^+}
\end{aligned} \tag{132}$$

$$(\delta \boldsymbol{u}^-, \boldsymbol{b}^-)_{\Omega^- \setminus \Omega^+} + (\delta \boldsymbol{u}^+, \boldsymbol{b}^+)_{\Omega^+} = (\delta \boldsymbol{u}^-, \boldsymbol{b}^-)_{\Omega^-} + (\delta \boldsymbol{u}^+, [\![\boldsymbol{b}]\!])_{\Omega^-} \tag{133}$$

From Eqns. (132) and (133), the new weak form reads: find $(\boldsymbol{u}^+, \boldsymbol{u}^-) \in \boldsymbol{\mathcal{U}}^+ \times \boldsymbol{\mathcal{U}}^-, \boldsymbol{u}^+ = \boldsymbol{u}^-$ on Ω^+, such that

$$\begin{aligned} &\left(\boldsymbol{\varepsilon}(\delta \boldsymbol{u}^-), \boldsymbol{C}^- \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^-)\right)_{\Omega^-} + \left(\boldsymbol{\varepsilon}(\delta \boldsymbol{u}^+), [\![\boldsymbol{C}]\!] \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^+)\right)_{\Omega^+} \\ &\qquad = (\delta \boldsymbol{u}^-, \boldsymbol{b}^-)_{\Omega^-} + (\delta \boldsymbol{u}^+, [\![\boldsymbol{b}]\!])_{\Omega^+} + (\delta \boldsymbol{u}^-, \boldsymbol{t})_{\partial \Omega_N^-} \end{aligned} \tag{134}$$

$\forall (\delta \boldsymbol{u}^+, \delta \boldsymbol{u}^-) \in \boldsymbol{\mathcal{V}}^+ \times \boldsymbol{\mathcal{V}}^-$. Comparing to Eq. (129) (embedded approach), Eq. (134) is referred to as "immersed weak form" hereafter. The imposition of the volumetric constraint in Eq. (131) can be achieved by various methods. Here three common approaches are introduced to impose the volumetric constraint by Penalty method, Lagrange multiplier method, and Nitsche's method.

5.1.3.1 Penalty Method

The penalty method can be performed by introducing a least-square penalty term for the volumetric constraint (131) to the immersed weak form (134), reading: find $(\boldsymbol{u}^+, \boldsymbol{u}^-) \in \boldsymbol{\mathcal{U}}^+ \times \boldsymbol{\mathcal{U}}^-$ such that

$$\begin{aligned} &\left(\boldsymbol{\varepsilon}(\delta \boldsymbol{u}^-), \boldsymbol{C}^- \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^-)\right)_{\Omega^-} + \left(\boldsymbol{\varepsilon}(\delta \boldsymbol{u}^+), [\![\boldsymbol{C}]\!] \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^+)\right)_{\Omega^+} + \beta([\![\delta \boldsymbol{u}]\!], [\![\boldsymbol{u}]\!])_{\Omega^+} \\ &\qquad = (\delta \boldsymbol{u}^-, \boldsymbol{b}^-)_{\Omega^-} + (\delta \boldsymbol{u}^+, [\![\boldsymbol{b}]\!])_{\Omega^+} + (\delta \boldsymbol{u}^-, \boldsymbol{t})_{\partial \Omega_N^-} \end{aligned} \tag{135}$$

for all $(\delta \boldsymbol{u}^+, \delta \boldsymbol{u}^-,) \in \boldsymbol{\mathcal{V}}^+ \times \boldsymbol{\mathcal{V}}^-$. The penalty parameter β is defined as

$$\beta = \beta_{nor} \frac{[\![E]\!]}{h^2} \tag{136}$$

where β_{nor} is the normalized penalty parameter. E is the Young's modulus, and h is the nodal spacing.

5.1.3.2 Lagrange Multiplier Method

The Lagrange multiplier method can be performed by introducing an additional constraint force variable $\boldsymbol{\lambda} \in \boldsymbol{\mathcal{W}} \equiv [L^2(\Omega^+)]^d$ with its corresponding test function $\delta\boldsymbol{\lambda} \in \boldsymbol{\mathcal{W}}$ for Eqns. (134) and (131), stating as: find $(\boldsymbol{u}^+, \boldsymbol{u}^-, \boldsymbol{\lambda}) \in \boldsymbol{\mathcal{U}}^+ \times \boldsymbol{\mathcal{U}}^- \times \boldsymbol{\mathcal{W}}$ such that

$$\begin{aligned}\left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^-), \boldsymbol{C}^- \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^-)\right)_{\Omega^-} + \left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^+), [\![\boldsymbol{C}]\!] \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^+)\right)_{\Omega^+} + ([\![\delta\boldsymbol{u}]\!], \boldsymbol{\lambda})_{\Omega^+} \\ + (\delta\boldsymbol{\lambda}, [\![\boldsymbol{u}]\!])_{\Omega^+} = (\delta\boldsymbol{u}^-, \boldsymbol{b}^-)_{\Omega^-} + (\delta\boldsymbol{u}^+, [\![\boldsymbol{b}]\!])_{\Omega^+} + (\delta\boldsymbol{u}^-, \boldsymbol{t})_{\partial\Omega_N^-}\end{aligned} \tag{137}$$

for all $(\delta\boldsymbol{u}^+, \delta\boldsymbol{u}^-, \delta\boldsymbol{\lambda}) \in \boldsymbol{\mathcal{V}}^+ \times \boldsymbol{\mathcal{V}}^- \times \boldsymbol{\mathcal{W}}$.

5.1.3.2 Nitsche's Method

A standard Nitsche's formulation [157] can be constructed by selecting the Lagrange multipliers $\boldsymbol{\lambda} = [\![\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(\boldsymbol{u})]\!]$ as the energy conjugate terms corresponding to the jump of the test function $[\![\boldsymbol{u}]\!]$. Also, a least-square type stabilization is added to ensure the coercivity of the system. The Nitsche's formulation for the weak form (134) then reads: find $(\boldsymbol{u}^+, \boldsymbol{u}^-) \in \widetilde{\boldsymbol{\mathcal{U}}}^+ \times \widetilde{\boldsymbol{\mathcal{U}}}^-$, such that

$$\begin{aligned}\left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^-), \boldsymbol{C}^- \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^-)\right)_{\Omega^-} + \left(\boldsymbol{\varepsilon}(\delta\boldsymbol{u}^+), [\![\boldsymbol{C}]\!] \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^+)\right)_{\Omega^+} + ([\![\delta\boldsymbol{u}]\!], [\![\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(\boldsymbol{u})]\!])_{\Omega^+} \\ + ([\![\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(\delta\boldsymbol{u})]\!], [\![\boldsymbol{u}]\!])_{\Omega^+} + \beta([\![\delta\boldsymbol{u}]\!], [\![\boldsymbol{u}]\!])_{\Omega^+} \\ = (\delta\boldsymbol{u}^-, \boldsymbol{b}^-)_{\Omega^-} + (\delta\boldsymbol{u}^+, [\![\boldsymbol{b}]\!])_{\Omega^+} + (\delta\boldsymbol{u}^-, \boldsymbol{t})_{\partial\Omega_N^-}\end{aligned} \tag{138}$$

for all $(\delta\boldsymbol{u}^+, \delta\boldsymbol{u}^-) \in \widetilde{\boldsymbol{\mathcal{V}}}^+ \times \widetilde{\boldsymbol{\mathcal{V}}}^-$. Here $\beta = \beta_{nor}[\![E]\!]h^{-2}$ works as the stabilization to ensure the coercivity of the system. Due to the stress divergence operator $\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}$ employed in the weak form, the function spaces $\widetilde{\boldsymbol{\mathcal{U}}}^+$, $\widetilde{\boldsymbol{\mathcal{U}}}^-$, $\widetilde{\boldsymbol{\mathcal{V}}}^+$ and $\widetilde{\boldsymbol{\mathcal{V}}}^-$ are taken as

$$\begin{aligned}
\widetilde{\boldsymbol{\mathcal{U}}}^{+} &\equiv \{\boldsymbol{u}|\boldsymbol{u} \in [H^2(\Omega^{+})]^d\} \\
\widetilde{\boldsymbol{\mathcal{V}}}^{+} &\equiv \{\boldsymbol{w}|\boldsymbol{w} \in [H^2(\Omega^{+})]^d\} \\
\widetilde{\boldsymbol{\mathcal{U}}}^{-} &\equiv \{\boldsymbol{u}|\boldsymbol{u} \in [H^2(\Omega^{-})]^d, \boldsymbol{u} = \boldsymbol{g} \text{ on } \partial\Omega_D^{-}\} \\
\widetilde{\boldsymbol{\mathcal{V}}}^{-} &\equiv \{\boldsymbol{w}|\boldsymbol{w} \in [H^2(\Omega^{-})]^d, \boldsymbol{w} = \boldsymbol{0} \text{ on } \partial\Omega_D^{-}\}
\end{aligned} \tag{139}$$

5.2 Variational Multiscale Immersed Method for Modeling Heterogeneous Materials with Volumetric Constraint

As discussed in the Chapter 2, the employment of conventional interfacial constraint enforcement may result reduced accuracy or the need of front tracking of the evolving interfaces. Here, the variational multiscale method (VMS) originally proposed by Hughes et al. [9, 70] is employed. From VMS, the displacement field at background $\boldsymbol{u}^{-}(\boldsymbol{x})$ is decomposed into a coarse-scale displacement field $\bar{\boldsymbol{u}}^{-}(\boldsymbol{x})$ and a fine-scale displacement field $\hat{\boldsymbol{u}}^{-}(\boldsymbol{x})$:

$$\begin{aligned}
\boldsymbol{u}^{-}(\boldsymbol{x}) &= \bar{\boldsymbol{u}}^{-}(\boldsymbol{x}) + \hat{\boldsymbol{u}}^{-}(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \Omega^{-} \\
\boldsymbol{u}^{+}(\boldsymbol{x}) &= \bar{\boldsymbol{u}}^{+}(\boldsymbol{x}) \quad \forall \boldsymbol{x} \in \Omega^{+}
\end{aligned} \tag{140}$$

and the foreground solution remains in coarse-scale. Note that the scale decomposition in the foreground or background domain can be selected freely based on the material composition in the heterogeneous materials. The approximation fields for the coarse-scale and fine-scale are constructed to be linearly independent. Define the function space $\boldsymbol{\mathcal{U}}^{-} = \bar{\boldsymbol{\mathcal{U}}}^{-} \oplus \hat{\boldsymbol{\mathcal{U}}}^{-}$ where $\bar{\boldsymbol{\mathcal{U}}}^{-}$ and $\hat{\boldsymbol{\mathcal{U}}}^{-}$ are the function spaces for the coarse-scale and fine-scale, respectively. For the foreground domain field, the function space $\boldsymbol{\mathcal{U}}^{+} = \bar{\boldsymbol{\mathcal{U}}}^{+}$ remains in coarse-scale and $\bar{\boldsymbol{u}}^{+} \in \bar{\boldsymbol{\mathcal{U}}}^{+}$. The following kinematic conditions are imposed for the coarse-scale and fine-scale solution:

$$\bar{\boldsymbol{u}}^- = \boldsymbol{g} \text{ on } \partial\Omega_D^-, \ \ \forall \bar{\boldsymbol{u}}^- \in \bar{\boldsymbol{U}}^- \tag{141}$$

$$\hat{\boldsymbol{u}}^- = \boldsymbol{0} \text{ on } \partial\Omega^-, \ \ \forall \hat{\boldsymbol{u}}^- \in \hat{\boldsymbol{U}}^- \tag{142}$$

$$\delta\bar{\boldsymbol{u}}^- = \boldsymbol{0} \text{ on } \partial\Omega_D^-, \ \ \forall \delta\bar{\boldsymbol{u}}^- \in \bar{\boldsymbol{V}}^- \tag{143}$$

$$\delta\hat{\boldsymbol{u}}^- = \boldsymbol{0} \text{ on } \partial\Omega^-, \ \ \forall \delta\hat{\boldsymbol{u}}^- \in \hat{\boldsymbol{V}}^- \tag{144}$$

which states the fine-scale homogeneous boundary conditions. One can treat the fine scale solution as the error of the course-scale numerical solution.

Remark 5.1.1: The fine-scale field can also be introduced to the foreground domain Ω^+ or both foreground and background domains. However, since the foreground inclusion domain is "immersed" in the background domain via the volumetric constraint, the fine-scale features shall influence the solution at the overlapping region and associated foreground domain. This effect is demonstrated in the numerical evidence.

5.2.1 Multiscale Problems and Variational Multiscale Immersed Method

To derived the variational multiscale immersed method, the multiscale decomposition in the Eq. (140) is introduced into the immersed weak form with Lagrange multiplier (137). The variational multiscale immersed weak form can then be expressed as:

$$\begin{aligned} &\left(\boldsymbol{\varepsilon}(\delta\bar{\boldsymbol{u}}^- + \delta\hat{\boldsymbol{u}}^-), \boldsymbol{C}^- \cdot \boldsymbol{\varepsilon}(\bar{\boldsymbol{u}}^- + \hat{\boldsymbol{u}}^-)\right)_{\Omega^-} + \left(\boldsymbol{\varepsilon}(\delta\bar{\boldsymbol{u}}^+), [\![\boldsymbol{C}]\!] \cdot \boldsymbol{\varepsilon}(\bar{\boldsymbol{u}}^+)\right)_{\Omega^+} \\ &\quad + ([\![\delta\bar{\boldsymbol{u}}]\!] - \delta\hat{\boldsymbol{u}}^-, \boldsymbol{\lambda})_{\Omega^+} + (\delta\boldsymbol{\lambda}, [\![\bar{\boldsymbol{u}}]\!] - \hat{\boldsymbol{u}}^-)_{\Omega^+} \\ &\quad = (\delta\bar{\boldsymbol{u}}^- + \delta\hat{\boldsymbol{u}}^-, \boldsymbol{b}^-)_{\Omega^-} + (\delta\bar{\boldsymbol{u}}^+, [\![\boldsymbol{b}]\!])_{\Omega^-} + (\delta\bar{\boldsymbol{u}}^-, \boldsymbol{t})_{\partial\Omega_N^-} \end{aligned} \tag{145}$$

By the virtue of the linear independence of test functions at different scale $\delta\bar{\boldsymbol{u}}^-$ and $\delta\hat{\boldsymbol{u}}^-$, Eq. (145) can be decoupled into the coarse-scale equations:

$$\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{-}+\widehat{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} - (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{\lambda})_{\Omega^{+}} = (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{b}^{-})_{\Omega^{-}} + (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{t})_{\partial\Omega_N^{-}} \tag{146}$$

$$\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{+}), [\![\boldsymbol{C}]\!]\cdot\boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{+})\right)_{\Omega^{+}} + (\delta\overline{\boldsymbol{u}}^{+}, \boldsymbol{\lambda})_{\Omega^{+}} = (\delta\overline{\boldsymbol{u}}^{+}, [\![\boldsymbol{b}]\!])_{\Omega^{+}} \tag{147}$$

$$(\delta\boldsymbol{\lambda}, \overline{\boldsymbol{u}}^{+})_{\Omega^{+}} - (\delta\boldsymbol{\lambda}, \overline{\boldsymbol{u}}^{-})_{\Omega^{+}} - (\delta\boldsymbol{\lambda}, \widehat{\boldsymbol{u}}^{-})_{\Omega^{+}} = 0 \tag{148}$$

and the fine-scale equation:

$$\left(\boldsymbol{\varepsilon}(\delta\widehat{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\widehat{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} = (\delta\widehat{\boldsymbol{u}}^{-}, \boldsymbol{b}^{-})_{\Omega^{-}} + (\delta\widehat{\boldsymbol{u}}^{-}, \boldsymbol{\lambda})_{\Omega^{+}} - \left(\boldsymbol{\varepsilon}(\delta\widehat{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} \tag{149}$$

The fine-scale solution serves as a correction to error of the coarse-scale solution, and the fine-scale equation (149) can also be represented in a different form by using the integration-by-part with the homogeneous fine-scale boundary condition (142):

$$\begin{aligned}\left(\boldsymbol{\varepsilon}(\delta\widehat{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\widehat{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} &= \left(\delta\widehat{\boldsymbol{u}}^{-}, \boldsymbol{\mathcal{L}}^{-}(\widehat{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} \\ &= (\delta\widehat{\boldsymbol{u}}^{-}, \boldsymbol{b}^{-})_{\Omega^{-}} + (\delta\widehat{\boldsymbol{u}}^{-}, \boldsymbol{\lambda})_{\Omega^{+}} - \left(\boldsymbol{\varepsilon}(\delta\widehat{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\widehat{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} \\ &= (\delta\widehat{\boldsymbol{u}}^{-}, \boldsymbol{b}^{-})_{\Omega^{-}} + (\delta\widehat{\boldsymbol{u}}^{-}, \boldsymbol{\lambda})_{\Omega^{+}} - \left(\delta\widehat{\boldsymbol{u}}^{-}, \boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}}\end{aligned} \tag{150}$$

The algebraic manipulation of Eq. (150) results in Eq. (151):

$$\left(\delta\widehat{\boldsymbol{u}}^{-}, \boldsymbol{\mathcal{L}}^{-}(\widehat{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} = (\delta\widehat{\boldsymbol{u}}^{-}, -\boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-}) + \boldsymbol{b}^{-})_{\Omega^{-}} + (\delta\widehat{\boldsymbol{u}}^{-}, \boldsymbol{\lambda})_{\Omega^{+}} \tag{151}$$

Since the domain Ω^{+} is immersed in Ω^{-}, the problem of Eq. (151) can be rewritten as

$$\begin{aligned}\widehat{\prod}\boldsymbol{\mathcal{L}}^{-}(\widehat{\boldsymbol{u}}^{-}) &= \widehat{\prod}(-\boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-}) + \boldsymbol{b}^{-} + \boldsymbol{\lambda}), \quad \text{in } \Omega^{-} \\ \widehat{\boldsymbol{u}}^{-} &= \boldsymbol{0}, \quad \text{on } \partial\Omega^{-}\end{aligned} \tag{152}$$

where $\widehat{\prod}$ is the L_2-projection onto $\widehat{\boldsymbol{\mathcal{U}}}^{-}$. The solution of Eq. (152) can be solved with the aid of Green's function [70]. Instead, the numerical approach can be employed to solve Eq. (152) or equivalently Eq. (151) or (149). By the integration-by-part with fine-scale homogeneous boundary condition (142), the coarse-scale equation at the background (146) leads to:

$$\begin{aligned}
&\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} + \left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\widehat{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} - (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{\lambda})_{\Omega^{+}} \\
&\qquad = \left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} - (\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \widehat{\boldsymbol{u}}^{-})_{\Omega^{-}} - (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{\lambda})_{\Omega^{+}} \\
&\qquad = (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{b}^{-})_{\Omega^{-}} + (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{t})_{\partial\Omega_{N}^{-}}
\end{aligned} \tag{153}$$

where $\boldsymbol{\mathcal{L}}^{-}(\cdot) = -\boldsymbol{\nabla}\cdot\left(\boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\cdot)\right)$ is the stress divergence operator. From Eq. (153), the fine-scale solution $\widehat{\boldsymbol{u}}^{-}$ can be directly employed without any differentiation performed on it.

Remark 5.2.1.1. The variational multiscale immersed formulation can be derived by solving the fine-scale equation (152) analytically, numerically, or approximately for $\widehat{\boldsymbol{u}}^{-}$. **T**he coarse-scale equations (146) or (153) are solved by embedding the fine-scale solution through a static condenstation. Inherently, Eq. (149) or (151) can both be used. In this study, the coarse-scale Eq. (146) and the fine-scale Eq. (149) are solved by introducing the bubble function [71] for the fine-scale solution. The coarse-scale Eq. (153) and fine-scale Eq. (151) are solved by the collocation method for fine-scale equation with an approximation of the local Green's function.

Remark 5.2.1.2. In Eq. (151) and (153), that $\boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-})$ requires a second order gradient performed on $\overline{\boldsymbol{u}}^{-}$, that can be achieved by method with higher continuity such as RKPM.

5.2.2 Method 1: Residual-Free Bubble Method

The first method employed here is to solve the fine-scale equation (149) by the aid of residual-free bubble method [71], where the $\widehat{\boldsymbol{u}}^-$ can be approximated by an enrichment function [68] or so-called bubble function [71]. The bubble basis can be computed from the integral of the local Green's function [9]. To start with, the bounded domain Ω can be discretized into subdomains Ω_L with their boundaries Γ_L such that

$$\widehat{\Omega}^- = \bigcup_L (int)\Omega_L^- \ \text{(domain interiors)} \tag{154}$$

$$\widehat{\Gamma}^- = \bigcup_L \Gamma_L^- \ \text{(domain boundaries)} \tag{155}$$

An assumption is made for employing the fine-scale bubble function (solution) [71]:

$$\widehat{\boldsymbol{u}}^- = \delta\widehat{\boldsymbol{u}}^- = \mathbf{0}, \ \text{on } \widehat{\Gamma}^- \tag{156}$$

where $\widehat{\Gamma}^-$ is defined in (155), such that the fine-scale fields can be approximated by bubble function for each nodal domain Ω_L^-:

$$\widehat{\boldsymbol{u}}^-(\boldsymbol{x}) \approx \widehat{\boldsymbol{u}}^{-h}(\boldsymbol{x}) = \sum_{L\in S^-} \widehat{\boldsymbol{N}}_L^-(\boldsymbol{x})\widehat{\boldsymbol{u}}_L^-, \ \ \widehat{\boldsymbol{N}}_L^- = \begin{bmatrix} \widehat{N}_L^- & 0 & 0 \\ 0 & \widehat{N}_L^- & 0 \\ 0 & 0 & \widehat{N}_L^- \end{bmatrix} \tag{157}$$

where $\widehat{N}_L^-$ is the nodal bubble basis function for domain Ω_L^- and $\widehat{\boldsymbol{u}}_L^-$ are the coefficients. $S^- = \{I|\boldsymbol{x}_I \in \Omega^-\}$ is the node set containing all nodes within domain Ω^-. From the assumption (156) that $\widehat{\boldsymbol{u}}^-$ vanishes on nodal domain boundaries, the bubble function satisfies the following condition:

$$\widehat{N}_L^-(\boldsymbol{x}) = 0, \ \ \forall \boldsymbol{x} \in \widehat{\Gamma}^- \tag{158}$$

The strain tensor operator $\boldsymbol{\varepsilon}$ for $\widehat{\boldsymbol{u}}^{-}$ can then be written as

$$\boldsymbol{\varepsilon}\left(\widehat{\boldsymbol{u}}^{-h}(\boldsymbol{x})\right) = \sum_{L\in S^{-}} \widehat{\boldsymbol{B}}_{L}^{-}(\boldsymbol{x})\widehat{\boldsymbol{u}}_{L}^{-} \tag{159}$$

where $\widehat{\boldsymbol{B}}_{L}^{-}$ following the Voigt notation is expressed as

$$\widehat{\boldsymbol{B}}_{L}^{-} = \begin{bmatrix} \widehat{N}_{L,1}^{-} & 0 & 0 \\ 0 & \widehat{N}_{L,2}^{-} & 0 \\ 0 & 0 & \widehat{N}_{L,3}^{-} \\ \widehat{N}_{L,2}^{-} & \widehat{N}_{L,1}^{-} & 0 \\ \widehat{N}_{L,3}^{-} & 0 & \widehat{N}_{L,1}^{-} \\ 0 & \widehat{N}_{L,3}^{-} & \widehat{N}_{L,2}^{-} \end{bmatrix} \tag{160}$$

Remark 5.2.2.1: The construction of nodal bubble function for conforming nodal represented domain (such as Voronoi cell) is usually tedious, especially for multidimensional case. Therefore, the non-conforming nodal domain [136] is employed in this study to construct the bubble function.

By substituting fine-scale solution with the bubble function in Eqns. (157) to (160) into Eq. (149), and revoke the arbitrariness of test fine-scale coefficients, the fine-scale problem can be rewritten in Eq. (161):

$$\sum_{J}\int_{\Omega^{-}} \widehat{\boldsymbol{B}}_{I}^{-T}\boldsymbol{C}^{-}\widehat{\boldsymbol{B}}_{J}d\Omega\,\widehat{\boldsymbol{u}}_{J}^{-} = \int_{\Omega^{-}} \widehat{N}_{I}^{-}\boldsymbol{b}^{-}d\Omega + \int_{\Omega^{+}} \widehat{N}_{I}^{-}\boldsymbol{\lambda}d\Omega - \int_{\Omega^{-}} \widehat{\boldsymbol{B}}_{I}^{-T}\boldsymbol{C}^{-}\overline{\boldsymbol{B}}_{J}d\Omega\overline{\boldsymbol{u}}_{J}^{-}\,,\forall I \tag{161}$$

Then, the coarse-scale approximation for coarse-scale variables $\overline{\boldsymbol{u}}^{-}$, $\overline{\boldsymbol{u}}^{+}$ and $\boldsymbol{\lambda}$ are introduced as follows:

$$\overline{\boldsymbol{u}}^{-}(\boldsymbol{x}) \approx \overline{\boldsymbol{u}}^{-h}(\boldsymbol{x}) = \sum_{I\in S^{-}} \overline{\boldsymbol{N}}_{I}^{-}(\boldsymbol{x})\overline{\boldsymbol{u}}_{I}^{-} \tag{162}$$

$$\bar{\boldsymbol{u}}^{+}(\boldsymbol{x}) \approx \bar{\boldsymbol{u}}^{+h}(\boldsymbol{x}) = \sum_{I \in S^{+}} \bar{\boldsymbol{N}}_I^{+}(\boldsymbol{x}) \bar{\boldsymbol{u}}_I^{+} \tag{163}$$

$$\boldsymbol{\lambda}(\boldsymbol{x}) \approx \boldsymbol{\lambda}^{h}(\boldsymbol{x}) = \sum_{I \in S^{+}} \boldsymbol{N}_I^{\lambda}(\boldsymbol{x}) \boldsymbol{\lambda}_I \tag{164}$$

$$\bar{\boldsymbol{N}}_I^{-} = \begin{bmatrix} \bar{N}_I^{-} & 0 & 0 \\ 0 & \bar{N}_I^{-} & 0 \\ 0 & 0 & \bar{N}_I^{-} \end{bmatrix}, \quad \bar{\boldsymbol{N}}_I^{+} = \begin{bmatrix} \bar{N}_I^{+} & 0 & 0 \\ 0 & \bar{N}_I^{+} & 0 \\ 0 & 0 & \bar{N}_I^{+} \end{bmatrix}, \quad \boldsymbol{N}_I^{\lambda} = \begin{bmatrix} N_I^{\lambda} & 0 & 0 \\ 0 & N_I^{\lambda} & 0 \\ 0 & 0 & N_I^{\lambda} \end{bmatrix} \tag{165}$$

where $\bar{\boldsymbol{N}}_I^{-}$, $\bar{\boldsymbol{N}}_I^{+}$ and $\boldsymbol{N}_I^{\lambda}$ are shape function for $\bar{\boldsymbol{u}}^{-}$, $\bar{\boldsymbol{u}}^{+}$ and $\boldsymbol{\lambda}$, respectively. $\bar{\boldsymbol{B}}_I^{-}$ and $\bar{\boldsymbol{B}}_I^{+}$ are gradient matrix for $\bar{\boldsymbol{N}}_I^{-}$ and $\bar{\boldsymbol{N}}_I^{+}$, respectively. $S^{+} = \{I | \boldsymbol{x}_I \in \Omega^{+}\}$ is the node set containing all nodes within domain Ω^{+}.

By the introduction of coarse-scale approximation shown in Eqns. (162) to (165), the fine-scale equation (161) can be solved in the following matrix form

$$\sum_J \hat{\bar{\boldsymbol{K}}}_{IJ} \hat{\boldsymbol{u}}_J^{-} = \hat{\boldsymbol{F}}_I^{-} + \sum_J \left(\hat{\boldsymbol{G}}_{IJ}^{-\lambda} \boldsymbol{\lambda}_J - \hat{\bar{\boldsymbol{K}}}_{IJ}^{-} \bar{\boldsymbol{u}}_J^{-} \right), \forall I \tag{166}$$

The matrix and vector in Eq. (166) are expressed in Eqns. (167) to (170):

$$\hat{\bar{\boldsymbol{K}}}_{IJ}^{-} = \int_{\Omega^{-}} \hat{\boldsymbol{B}}_I^{-T} \boldsymbol{C}^{-} \hat{\boldsymbol{B}}_J^{-} d\Omega \tag{167}$$

$$\hat{\boldsymbol{G}}_{IJ}^{-\lambda} = \int_{\Omega^{+}} \hat{\boldsymbol{N}}_I^{-T} \boldsymbol{N}_J^{\lambda} d\Omega \tag{168}$$

$$\hat{\bar{\boldsymbol{K}}}_{IJ}^{-} = \int_{\Omega^{-}} \hat{\boldsymbol{B}}_I^{-T} \boldsymbol{C}^{-} \bar{\boldsymbol{B}}_J^{-} d\Omega \tag{169}$$

$$\hat{\boldsymbol{F}}_I^{-} = \int_{\Omega^{-}} \hat{\boldsymbol{N}}_I^{-T} \boldsymbol{b}^{-} d\Omega \tag{170}$$

Once the fine-scale solution $\hat{\boldsymbol{u}}^{-}$ is solved, the static condensation of fine-scale solution into the coarse-scale system results in the VMIM weak form: find $(\bar{\boldsymbol{u}}^{+}, \bar{\boldsymbol{u}}^{-}, \boldsymbol{\lambda}) \in \bar{\boldsymbol{\mathcal{U}}}^{+} \times \bar{\boldsymbol{\mathcal{U}}}^{-} \times \boldsymbol{\mathcal{W}}$, such that

$$\begin{aligned}\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{-}),\boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}}+\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{-}),\boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\widehat{\boldsymbol{u}}^{-})\right)_{\Omega^{-}}-(\delta\overline{\boldsymbol{u}}^{-},\boldsymbol{\lambda})_{\Omega^{+}}\\ =(\delta\overline{\boldsymbol{u}}^{-},\boldsymbol{b}^{-})_{\Omega^{-}}+(\delta\overline{\boldsymbol{u}}^{-},\boldsymbol{t})_{\partial\Omega_{N}^{-}}\end{aligned}\tag{171}$$

$$\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{+}),[\![\boldsymbol{C}]\!]\cdot\boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{+})\right)_{\Omega^{+}}+(\delta\overline{\boldsymbol{u}}^{+},\boldsymbol{\lambda})_{\Omega^{+}}=(\delta\overline{\boldsymbol{u}}^{+},[\![\boldsymbol{b}]\!])_{\Omega^{+}}\tag{172}$$

$$(\delta\boldsymbol{\lambda},\overline{\boldsymbol{u}}^{+})_{\Omega^{+}}-(\delta\boldsymbol{\lambda},\overline{\boldsymbol{u}}^{-})_{\Omega^{+}}-(\delta\boldsymbol{\lambda},\widehat{\boldsymbol{u}}^{-})_{\Omega^{+}}=0\tag{173}$$

$\forall(\delta\overline{\boldsymbol{u}}^{+},\delta\overline{\boldsymbol{u}}^{-},\delta\boldsymbol{\lambda})\in\overline{\boldsymbol{\mathcal{V}}}^{+}\times\overline{\boldsymbol{\mathcal{V}}}^{-}\times\boldsymbol{\mathcal{W}}$. The function spaces $\overline{\boldsymbol{\mathcal{U}}}^{+}$, $\overline{\boldsymbol{\mathcal{U}}}^{-}$, $\overline{\boldsymbol{\mathcal{V}}}^{+}$, and $\overline{\boldsymbol{\mathcal{V}}}^{-}$ here are given as

$$\begin{aligned}\overline{\boldsymbol{\mathcal{U}}}^{+}&\equiv\{\boldsymbol{u}|\boldsymbol{u}\in[H^{1}(\Omega^{+})]^{d}\}\\ \overline{\boldsymbol{\mathcal{V}}}^{+}&\equiv\{\boldsymbol{w}|\boldsymbol{w}\in[H^{1}(\Omega^{+})]^{d}\}\\ \overline{\boldsymbol{\mathcal{U}}}^{-}&\equiv\{\boldsymbol{u}|\boldsymbol{u}\in[H^{1}(\Omega^{-})]^{d},\boldsymbol{u}=\boldsymbol{g}\text{ on }\partial\Omega_{D}^{-}\}\\ \overline{\boldsymbol{\mathcal{V}}}^{-}&\equiv\{\boldsymbol{w}|\boldsymbol{w}\in[H^{1}(\Omega^{-})]^{d},\boldsymbol{w}=\boldsymbol{0}\text{ on }\partial\Omega_{D}^{-}\}\end{aligned}\tag{174}$$

Then, the matrix form of coarse-scale equations (171) to (173) can be derived by introducing the aforementioned approximations for coarse-scale fields, given in Eqns. (175) to (177):

$$\sum_{J}\left(\overline{\boldsymbol{K}}_{IJ}^{-}\overline{\boldsymbol{u}}_{J}^{-}+\overline{\widehat{\boldsymbol{K}}}_{IJ}^{-}\widehat{\boldsymbol{u}}_{J}^{-}-\overline{\boldsymbol{G}}_{IJ}^{-\lambda}\boldsymbol{\lambda}_{J}\right)=\overline{\boldsymbol{F}}_{I}^{-}\tag{175}$$

$$\sum_{J}\left(\overline{\boldsymbol{K}}_{IJ}^{+}\overline{\boldsymbol{u}}_{J}^{+}+\overline{\boldsymbol{G}}_{IJ}^{+\lambda}\boldsymbol{\lambda}_{J}\right)=\overline{\boldsymbol{F}}_{I}^{+}\tag{176}$$

$$\sum_{J}\left(\overline{\boldsymbol{G}}_{IJ}^{\lambda+}\overline{\boldsymbol{u}}_{J}^{+}-\overline{\boldsymbol{G}}_{IJ}^{\lambda-}\overline{\boldsymbol{u}}_{J}^{-}-\widehat{\boldsymbol{G}}_{IJ}^{\lambda-}\widehat{\boldsymbol{u}}_{J}^{-}\right)=\boldsymbol{0}\tag{177}$$

for all I. The matrices and vectors shown in Eqns. (175) to (177) are computed by the following domain integration:

$$\overline{\boldsymbol{K}}_{IJ}^{-}=\int_{\Omega^{-}}\overline{\boldsymbol{B}}_{I}^{-T}\boldsymbol{C}^{-}\overline{\boldsymbol{B}}_{J}^{-}\,d\Omega\tag{178}$$

$$\overline{\boldsymbol{K}}_{IJ}^{+} = \int_{\Omega^+} \overline{\boldsymbol{B}}_I^{+T} [\![\boldsymbol{C}]\!] \overline{\boldsymbol{B}}_J^{+} d\Omega \tag{179}$$

$$\overline{\widehat{\boldsymbol{K}}}_{IJ}^{-} = \int_{\Omega^-} \overline{\boldsymbol{B}}_I^{-T} \boldsymbol{C}^{-} \widehat{\boldsymbol{B}}_J^{-} d\Omega = \overline{\widehat{\boldsymbol{K}}}_{IJ}^{-T} \tag{180}$$

$$\overline{\boldsymbol{G}}_{IJ}^{-\lambda} = \int_{\Omega^+} \overline{\boldsymbol{N}}_I^{-T} \boldsymbol{N}_J^{\lambda} d\Omega = \overline{\boldsymbol{G}}_{IJ}^{\lambda -T} \tag{181}$$

$$\overline{\boldsymbol{G}}_{IJ}^{+\lambda} = \int_{\Omega^+} \overline{\boldsymbol{N}}_I^{+T} \boldsymbol{N}_J^{\lambda} d\Omega = \overline{\boldsymbol{G}}_{IJ}^{\lambda +T} \tag{182}$$

$$\widehat{\boldsymbol{G}}_{IJ}^{\lambda -} = \int_{\Omega^+} \boldsymbol{N}_I^{\lambda T} \widehat{\boldsymbol{N}}_J^{-} d\Omega \tag{183}$$

$$\overline{\boldsymbol{F}}_I^{-} = \int_{\Omega^-} \overline{\boldsymbol{N}}_I^{-T} \boldsymbol{b}^{-} d\Omega + \int_{\partial\Omega_N^-} \overline{\boldsymbol{N}}_I^{-T} \boldsymbol{t} d\Gamma \tag{184}$$

$$\overline{\boldsymbol{F}}_I^{+} = \int_{\Omega^+} \overline{\boldsymbol{N}}_I^{+T} [\![\boldsymbol{b}]\!] d\Omega \tag{185}$$

With the static condensation of the fine-scale solution from Eq. (166) into the Eqns. (175) to (177), the final coarse-scale matrix equations to be solved can be written in the form shown in Eqns. (186) to (188):

$$\sum_J \left(\overline{\boldsymbol{K}}_{IJ}^{-*} \overline{\boldsymbol{u}}_J^{-} + \overline{\boldsymbol{G}}_{IJ}^{-\lambda *} \boldsymbol{\lambda}_J \right) = \overline{\boldsymbol{F}}_I^{-*} \tag{186}$$

$$\sum_J \left(\overline{\boldsymbol{K}}_{IJ}^{+} \overline{\boldsymbol{u}}_J^{+} + \overline{\boldsymbol{G}}_{IJ}^{+\lambda} \boldsymbol{\lambda}_J \right) = \overline{\boldsymbol{F}}_I^{+} \tag{187}$$

$$\sum_J \left(\overline{\boldsymbol{G}}_{IJ}^{+\lambda T} \overline{\boldsymbol{u}}_J^{+} + \boldsymbol{G}_{IJ}^{-\lambda * T} \overline{\boldsymbol{u}}_J^{-} + \boldsymbol{G}_{IJ}^{\lambda\lambda *} \boldsymbol{\lambda}_J \right) = \boldsymbol{F}_I^{\lambda *} \tag{188}$$

for all I, where the matrices and vector in Eqns. (186) to (188) are expressed as

$$\overline{\boldsymbol{K}}^{-*} = \overline{\boldsymbol{K}}^{-} - \overline{\widehat{\boldsymbol{K}}}^{-T} \left(\overline{\widehat{\boldsymbol{K}}}^{-} \right)^{-1} \overline{\widehat{\boldsymbol{K}}}^{-} \tag{189}$$

$$\overline{\boldsymbol{G}}^{-\lambda*} = \widehat{\overline{\boldsymbol{K}}}^{-T}\left(\widehat{\overline{\boldsymbol{K}}}^{-}\right)^{-1}\widehat{\boldsymbol{G}}^{-\lambda} - \overline{\boldsymbol{G}}^{-\lambda} \tag{190}$$

$$\overline{\boldsymbol{F}}^{-*} = \overline{\boldsymbol{F}}^{-} - \widehat{\overline{\boldsymbol{K}}}^{-T}\left(\widehat{\overline{\boldsymbol{K}}}^{-}\right)^{-1}\widehat{\boldsymbol{F}}^{-} \tag{191}$$

$$\boldsymbol{G}^{\lambda\lambda*} = \widehat{\boldsymbol{G}}^{-\lambda T}\left(\widehat{\overline{\boldsymbol{K}}}^{-}\right)^{-1}\widehat{\boldsymbol{G}}^{-\lambda} \tag{192}$$

$$\boldsymbol{F}^{\lambda*} = \widehat{\boldsymbol{G}}^{-\lambda T}\left(\widehat{\overline{\boldsymbol{K}}}^{-}\right)^{-1}\widehat{\boldsymbol{F}}^{-} \tag{193}$$

Eqns. (186) to (188) can be rearranged into a final matrix form as shown in Eq. (194)

$$\sum_{J}\begin{bmatrix} \boldsymbol{K}_{IJ}^{-*} & \boldsymbol{0} & \boldsymbol{G}_{IJ}^{-\lambda*} \\ \boldsymbol{0} & \overline{\boldsymbol{K}}_{IJ}^{+} & \overline{\boldsymbol{G}}_{IJ}^{+\lambda} \\ \boldsymbol{G}_{IJ}^{-\lambda*T} & \overline{\boldsymbol{G}}_{IJ}^{+\lambda T} & \boldsymbol{G}_{IJ}^{\lambda\lambda*} \end{bmatrix}\begin{Bmatrix} \overline{\boldsymbol{u}}_{J}^{-} \\ \overline{\boldsymbol{u}}_{J}^{+} \\ \boldsymbol{\lambda}_{J} \end{Bmatrix} = \begin{bmatrix} \overline{\boldsymbol{F}}_{I}^{-*} \\ \overline{\boldsymbol{F}}_{I}^{+} \\ \boldsymbol{F}_{I}^{\lambda*} \end{bmatrix}, \qquad \forall I \tag{194}$$

Eq. (194) is the final matrix form to be solved.

Remark 5.2.2.2. A direct comparison of VMIM in Eq. (194) with the conventional immersed approach can be made by analyzing the matrix form from the standard immersed formulation with Lagrange multipliers, Eq. (137), as shown in Eq. (195)

$$\sum_{J}\begin{bmatrix} \overline{\boldsymbol{K}}_{IJ}^{-} & \boldsymbol{0} & -\overline{\boldsymbol{G}}_{IJ}^{-\lambda} \\ \boldsymbol{0} & \overline{\boldsymbol{K}}_{IJ}^{+} & \overline{\boldsymbol{G}}_{IJ}^{+\lambda} \\ -\overline{\boldsymbol{G}}_{IJ}^{-\lambda T} & \overline{\boldsymbol{G}}_{IJ}^{+\lambda T} & \boldsymbol{0} \end{bmatrix}\begin{Bmatrix} \overline{\boldsymbol{u}}_{J}^{-} \\ \overline{\boldsymbol{u}}_{J}^{+} \\ \boldsymbol{\lambda}_{J} \end{Bmatrix} = \begin{bmatrix} \overline{\boldsymbol{F}}_{I}^{-} \\ \overline{\boldsymbol{F}}_{I}^{+} \\ \boldsymbol{0} \end{bmatrix}, \qquad \forall I \tag{195}$$

It can be observed that Eq. (195) is the degeneration of Eq. (194) with the elimination of the effect of fine-scale. As mentioned earlier, Eq. (195) required to satisfy the LBB condition to yield a stable and convergent solution. Eq. (194) can be viewed as a modified or augmented Lagrange multiplier formulation for Eq. (195) where the term $\widehat{\overline{\boldsymbol{K}}}^{-1}$ plays the role of residual-based stabilization. In the conventional augmented Lagrange multiplier formulation, such stabilization is introduced through an additional penalization. It is important to note that in the present method the fine-scale component naturally leads to the residual-based stabilization.

Remark 5.2.2.3: The left hand side matrix in Eq. (194) is a symmetric matrix which is expressed entirely in terms of the coarse scale solution. The fine-scale solution is solved and calculated separately from Eq. (166) with the solution from the coarse-scale.

5.2.3 Method 2: Approximated Fine-Scale Solution

The second method derived here follows the concept of stabilized Galerkin method [70] such as stabilized upwind Petrov Galerkin (SUPG) [27], where the fine-scale solution can be derived approximately. Since the goal is to improve the accuracy and numerical stability of the original immersed method, the collocation method with an approximation to the solution of Eq. (151) can be employed as shown in Eq. (196)

$$\widehat{\boldsymbol{u}}^{-} = -\hat{\tau}^{-}(-\boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-}) + \boldsymbol{b}^{-} + \boldsymbol{\lambda}) \tag{196}$$

where $\hat{\tau}^{-}$ is a scalar, that can be interpreted as the nodal domain average of fine-scale Green functions from Eq. (152). The value of $\hat{\tau}^{-}$ is derived later. By substituting Eq. (196) into Eq. (153), we have

$$\begin{aligned}\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} + \left(\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \hat{\tau}^{-}(-\boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-}) + \boldsymbol{b}^{-} + \boldsymbol{\lambda})\right)_{\Omega^{-}} - (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{\lambda})_{\Omega^{+}} \\ = (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{b}^{-})_{\Omega^{-}} + (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{t})_{\partial\Omega_{N}^{-}}\end{aligned} \tag{197}$$

The rearrangement of Eq. (197) gives the form:

$$\begin{aligned}\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} - \left(\hat{\tau}^{-}\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} + (\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \hat{\tau}^{-}\boldsymbol{\lambda})_{\Omega^{-}} \\ - (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{\lambda})_{\Omega^{+}} \\ = (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{b}^{-})_{\Omega^{-}} + (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{t})_{\partial\Omega_{\mathrm{N}}^{-}} - (\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \hat{\tau}^{-}\boldsymbol{b}^{-})_{\Omega^{-}}\end{aligned} \tag{198}$$

Similar to the method 1 (bubble method), the substitution of fine-scale solution into the constraint equation (148) gives the following equation

$$(\delta\boldsymbol{\lambda}, \overline{\boldsymbol{u}}^{+})_{\Omega^{+}} - (\delta\boldsymbol{\lambda}, \overline{\boldsymbol{u}}^{-})_{\Omega^{+}} + \left(\delta\boldsymbol{\lambda}, \hat{\tau}^{-}(-\boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-}) + \boldsymbol{b}^{-} + \boldsymbol{\lambda})\right)_{\Omega^{+}} = 0 \tag{199}$$

and the rearrangement of Eq. (199) gives

$$(\delta\boldsymbol{\lambda}, \overline{\boldsymbol{u}}^{+})_{\Omega^{+}} - \left(\delta\boldsymbol{\lambda}, \overline{\boldsymbol{u}}^{-} + \hat{\tau}^{-}\boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{+}} + (\delta\boldsymbol{\lambda}, \hat{\tau}^{-}\boldsymbol{\lambda})_{\Omega^{+}} = (\delta\boldsymbol{\lambda}, -\hat{\tau}^{-}\boldsymbol{b}^{-})_{\Omega^{+}} \tag{200}$$

Then, the weak form of VMIM via approximated fine-scale solution reads: find $(\overline{\boldsymbol{u}}^{+}, \overline{\boldsymbol{u}}^{-}, \boldsymbol{\lambda}) \in \overline{\boldsymbol{\mathcal{U}}}^{+} \times \overline{\boldsymbol{\mathcal{U}}}^{-} \times \boldsymbol{\mathcal{W}}$, such that

$$\begin{aligned} &\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{C}^{-} \cdot \boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} - \left(\hat{\tau}^{-}\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} + (\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \hat{\tau}^{-}\boldsymbol{\lambda})_{\Omega^{-}} \\ &\quad - (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{\lambda})_{\Omega^{+}} \\ &\quad = (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{b}^{-})_{\Omega^{-}} + (\delta\overline{\boldsymbol{u}}^{-}, \boldsymbol{t})_{\partial\Omega_{\mathrm{N}}^{-}} - (\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \hat{\tau}^{-}\boldsymbol{b}^{-})_{\Omega^{-}} \end{aligned} \tag{201}$$

$$\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{+}), [\![\boldsymbol{C}]\!] \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}^{+})\right)_{\Omega^{+}} + (\delta\overline{\boldsymbol{u}}^{+}, \boldsymbol{\lambda})_{\Omega^{+}} = (\delta\overline{\boldsymbol{u}}^{+}, [\![\boldsymbol{b}]\!])_{\Omega^{+}} \tag{202}$$

$$(\delta\boldsymbol{\lambda}, \overline{\boldsymbol{u}}^{+})_{\Omega^{+}} - \left(\delta\boldsymbol{\lambda}, \overline{\boldsymbol{u}}^{-} + \hat{\tau}^{-}\boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{+}} + (\delta\boldsymbol{\lambda}, \hat{\tau}^{-}\boldsymbol{\lambda})_{\Omega^{+}} = (\delta\boldsymbol{\lambda}, -\hat{\tau}^{-}\boldsymbol{b}^{-})_{\Omega^{+}} \tag{203}$$

$\forall(\delta\overline{\boldsymbol{u}}^{+}, \delta\overline{\boldsymbol{u}}^{-}, \delta\boldsymbol{\lambda}) \in \overline{\boldsymbol{\mathcal{V}}}^{+} \times \overline{\boldsymbol{\mathcal{V}}}^{-} \times \boldsymbol{\mathcal{W}}$. The function spaces $\overline{\boldsymbol{\mathcal{U}}}^{+}$, $\overline{\boldsymbol{\mathcal{U}}}^{-}$, $\overline{\boldsymbol{\mathcal{V}}}^{+}$, and $\overline{\boldsymbol{\mathcal{V}}}^{-}$ are given as

$$\begin{aligned} \overline{\boldsymbol{\mathcal{U}}}^{+} &\equiv \{\boldsymbol{u} | \boldsymbol{u} \in [H^{1}(\Omega^{+})]^{d}\} \\ \overline{\boldsymbol{\mathcal{V}}}^{+} &\equiv \{\boldsymbol{w} | \boldsymbol{w} \in [H^{1}(\Omega^{+})]^{d}\} \\ \overline{\boldsymbol{\mathcal{U}}}^{-} &\equiv \{\boldsymbol{u} | \boldsymbol{u} \in [H^{2}(\Omega^{-})]^{d}, \boldsymbol{u} = \boldsymbol{g} \text{ on } \partial\Omega_{D}^{-}\} \\ \overline{\boldsymbol{\mathcal{V}}}^{-} &\equiv \{\boldsymbol{w} | \boldsymbol{w} \in [H^{2}(\Omega^{-})]^{d}, \boldsymbol{w} = \boldsymbol{0} \text{ on } \partial\Omega_{D}^{-}\} \end{aligned} \tag{204}$$

The stress divergence operator $\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}$ for the coarse-scale solution $\overline{\boldsymbol{u}}^{-}$ be derived as shown in Eq. (205),

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(\overline{\boldsymbol{u}}^{-}) = \sum_{I \in S^{-}} \boldsymbol{\eta}_{i} \boldsymbol{C}^{-} \overline{\boldsymbol{B}}_{I,i}^{-} \overline{\boldsymbol{u}}_{I}^{-}, \quad (\text{sum on } i) \tag{205}$$

where $\boldsymbol{\eta}_{i=1,2,3}$ are geometric tensors expressed as

$$\boldsymbol{\eta}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\boldsymbol{\eta}_2 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{206}$$

$$\boldsymbol{\eta}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

and $\overline{\boldsymbol{B}}_{I,i}^{-}$ is the derivative of $\overline{\boldsymbol{B}}_{I}^{-}$ with respect to x_i:

$$\overline{\boldsymbol{B}}_{I,i}^{-} = \frac{\partial}{\partial x_i} \begin{bmatrix} \overline{N}_{I,1}^{-} & 0 & 0 \\ 0 & \overline{N}_{I,2}^{-} & 0 \\ 0 & 0 & \overline{N}_{I,3}^{-} \\ \overline{N}_{I,2}^{-} & \overline{N}_{I,1}^{-} & 0 \\ \overline{N}_{I,3}^{-} & 0 & \overline{N}_{I,1}^{-} \\ 0 & \overline{N}_{I,3}^{-} & \overline{N}_{I,2}^{-} \end{bmatrix} \tag{207}$$

In this study, the implicit gradient technique [129] is employed for calculating $\overline{N}_{I,ij}^{-}$ by the differentiation of the first-order implicit gradient $\overline{N}_{I,i}^{-}$ with respect to $\boldsymbol{x}_j$.

Finally, by employing the coarse-scale approximation in Eqns. (162) to (164) into the weak form, and using the arbitrariness of test function, the matrix form of VMIM via approximated fine-scale solution can be obtained as

$$\sum_{J} \begin{bmatrix} \overline{\boldsymbol{K}}_{IJ}^{-} - \widehat{\boldsymbol{K}}_{IJ}^{-\tau} & \boldsymbol{0} & -\widehat{\boldsymbol{G}}_{IJ}^{-\tau\lambda} - \overline{\boldsymbol{G}}_{IJ}^{-\lambda} \\ \boldsymbol{0} & \overline{\boldsymbol{K}}_{IJ}^{+} & \overline{\boldsymbol{G}}_{IJ}^{+\lambda} \\ \widehat{\boldsymbol{G}}_{IJ}^{-\tau\lambda^T} - \overline{\boldsymbol{G}}_{IJ}^{-\lambda^T} & \overline{\boldsymbol{G}}_{IJ}^{+\lambda^T} & \boldsymbol{G}_{IJ}^{\lambda\tau\lambda} \end{bmatrix} \begin{Bmatrix} \overline{\boldsymbol{u}}_J^{-} \\ \overline{\boldsymbol{u}}_J^{-} \\ \boldsymbol{\lambda}_J \end{Bmatrix} = \begin{bmatrix} \overline{\boldsymbol{F}}_I^{-} + \widehat{\boldsymbol{F}}_I^{\tau} \\ \overline{\boldsymbol{F}}_I^{+} \\ -\boldsymbol{F}_I^{\tau\lambda} \end{bmatrix}, \qquad \forall I \tag{208}$$

where $\overline{\boldsymbol{K}}_{IJ}^{-}$, $\overline{\boldsymbol{K}}_{IJ}^{+}$, $\overline{\boldsymbol{G}}_{IJ}^{-\lambda}$, $\overline{\boldsymbol{G}}_{IJ}^{+\lambda}$, $\overline{\boldsymbol{F}}_I^{-}$, and $\overline{\boldsymbol{F}}_I^{+}$ are given in Eqns. (178) to (185) and $\widehat{\boldsymbol{G}}_{IJ}^{-\tau}$, $\widehat{\boldsymbol{G}}_{IJ}^{-\tau\lambda}$, $\boldsymbol{G}_{IJ}^{\lambda\tau\lambda}$, $\widehat{\boldsymbol{F}}_I^{\tau}$and $\boldsymbol{F}_I^{\tau\lambda}$ are given as follows:

$$\widehat{\boldsymbol{K}}_{IJ}^{-\tau} = \int_{\Omega^-} \sum_{i=1}^{d} \left(\boldsymbol{\eta}_i \boldsymbol{C}^- \overline{\boldsymbol{B}}_{I,i}^-\right)^T \hat{\tau}^- \sum_{j=1}^{d} \left(\boldsymbol{\eta}_j \boldsymbol{C}^- \overline{\boldsymbol{B}}_{J,j}^-\right) d\Omega \tag{209}$$

$$\widehat{\boldsymbol{G}}_{IJ}^{-\tau\lambda} = \int_{\Omega^-} \sum_{i=1}^{d} \left(\boldsymbol{\eta}_i \boldsymbol{C}^- \overline{\boldsymbol{B}}_{I,i}^-\right)^T \hat{\tau}^- \boldsymbol{N}_J^{\lambda} d\Omega \tag{210}$$

$$\boldsymbol{G}_{IJ}^{\lambda\tau\lambda} = \int_{\Omega^+} \boldsymbol{N}_I^{\lambda T} \hat{\tau}^- \boldsymbol{N}_J^{\lambda} d\Omega \tag{211}$$

$$\widehat{\boldsymbol{F}}_I^{\tau} = \int_{\Omega^+} \sum_{i=1}^{d} \left(\boldsymbol{\eta}_i \boldsymbol{C}^- \overline{\boldsymbol{B}}_{I,i}^-\right)^T \hat{\tau}^- \boldsymbol{b}^- d\Omega \tag{212}$$

$$\boldsymbol{F}_I^{\tau\lambda} = \int_{\Omega^+} \hat{\tau}^- \boldsymbol{N}_J^{\lambda T} \boldsymbol{b}^- d\Omega \tag{213}$$

The parameter $\hat{\tau}^-$ is suggested to be written in the following form [161]:

$$\hat{\tau}^- = c_\tau \frac{h^2}{E^-} \tag{214}$$

where h is the characteristic nodal distance, E^- is the Young's modulus of background matrix media, and c_τ is a constant determined by the user, usually taken as $c_\tau \in [0,1]$. The parameter $\hat{\tau}^-$ is computed from the nodal domain average of Green's function [162]:

$$\hat{\tau}^- = \frac{1}{meas(\Omega_L)} \int_{\Omega_L} \int_{\Omega_L} g_L(x,x') d\Omega_x \, d\Omega_{x'} = \frac{1}{meas(\Omega_L)} \int_{\Omega_L} B(x') d\Omega_{x'} \tag{215}$$

for a given nodal domain Ω_L where $g_L(x,x')$ is the nodal Green's function and $B(x') = \int_{\Omega_L} g_L(x,x') d\Omega_x$ is the fine-scale basis, satisfying the following local problem:

$$\begin{gathered} \mathcal{L}^-(B) = 1 \text{ on } \Omega_L \\ B = 0 \text{ on } \Gamma_L \end{gathered} \tag{216}$$

A one-dimensional version of Eq. (216) is considered here:

$$-E^{-}B_{,xx} = 1, \quad \forall x \in]0,h[$$
$$B(0) = B(h) = 0 \tag{217}$$

where h is the size of the nodal domain and the solution of Eq. (217) can be derived:

$$B(x) = -\frac{1}{2E^{-}}x^2 + \frac{h}{2E^{-}}x \tag{218}$$

and from Eq. (215), $\hat{\tau}^{-}$ can be obtained

$$\hat{\tau}^{-} = \frac{1}{h}\int_0^h B(x)dx = \frac{h^2}{12E^{-}} \tag{219}$$

By comparing Eq. (219) to Eq. (214), $c_\tau = \frac{1}{12}$ is employed.

Remark 5.2.1: The term $\hat{\tau}^{-}$ represents an approximation to the fine-scale Green's function as shown in Eq. (215). From Eq. (196), the residual (or error) of the coarse-scale equations, namely $\boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-}) - \boldsymbol{b}^{-} - \boldsymbol{\lambda}$, is normalized by $\hat{\tau}^{-}$.

Remark 5.2.2: By employing the integration-by-part for the coarse-scale internal energy terms, the summation of Eqns. (198), (200), and (147) gives the following form in terms of the coarse-scale fields only:

$$\begin{aligned}\left(\delta\overline{\boldsymbol{u}}^{-} - \hat{\tau}^{-}\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}} - (\delta\overline{\boldsymbol{u}}^{-} - \hat{\tau}^{-}\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{\lambda})_{\Omega^{+}} \\ + \left(\delta\overline{\boldsymbol{u}}^{+}, \boldsymbol{\mathcal{L}}^{+}(\overline{\boldsymbol{u}}^{+})\right)_{\Omega^{-}} + (\delta\overline{\boldsymbol{u}}^{+}, \boldsymbol{\lambda})_{\Omega^{+}} \\ = (\delta\overline{\boldsymbol{u}}^{-} - \hat{\tau}^{-}\boldsymbol{\mathcal{L}}^{-}(\delta\overline{\boldsymbol{u}}^{-}), \boldsymbol{b}^{-})_{\Omega^{-}} + (\delta\overline{\boldsymbol{u}}^{+}, [\![\boldsymbol{b}]\!])_{\Omega^{+}}\end{aligned} \tag{220}$$

which is a Petrov-Galerkin formulation for the given immersed setting.

Remark 5.2.3: From the expression of Eq. (197), the derived VMIM via approximated fine-scale solution naturally leads to a stabilized Galerkin formulation [9] in the form of Eq. (221):

$$\begin{aligned}\left(\boldsymbol{\varepsilon}(\delta\overline{\boldsymbol{u}}^{-}),\boldsymbol{C}^{-}\cdot\boldsymbol{\varepsilon}(\overline{\boldsymbol{u}}^{-})\right)_{\Omega^{-}}+\left(\hat{\tau}^{-}\mathbb{L}(\delta\overline{\boldsymbol{u}}^{-}),(-\boldsymbol{\mathcal{L}}^{-}(\overline{\boldsymbol{u}}^{-})+\boldsymbol{b}^{-}+\boldsymbol{\lambda})\right)_{\Omega^{-}}-(\delta\overline{\boldsymbol{u}}^{-},\boldsymbol{\lambda})_{\Omega^{+}}\\ =(\delta\overline{\boldsymbol{u}}^{-},\boldsymbol{b}^{-})_{\Omega^{-}}+(\delta\overline{\boldsymbol{u}}^{-},\boldsymbol{t})_{\partial\Omega_{N}^{-}}\end{aligned} \tag{221}$$

with the operator $\mathbb{L}=\boldsymbol{\mathcal{L}}^{-}$ chosen to be the Galerkin/lease-square (GLS) operator [10]. This validates the stabilization effect in the proposed formulation.

5.2.4 Quadrature Rule in Immersed Framework

A stabilized nodal integration scheme is employed for the domain integration in the conventional immersed method and variational multiscale immersed method, where the stabilized conforming nodal integration (SCNI) [40] with naturally stabilized nodal integration (NSNI) stabilization [119] are employed for the coarse-scale equations. The stiffness matrix in (178) and (179).

$$\bar{\boldsymbol{K}}_{IJ}^{-} = \bar{\boldsymbol{K}}_{IJ}^{-SCNI} + \bar{\boldsymbol{K}}_{IJ}^{-NSNI} \tag{222}$$

$$\bar{\boldsymbol{K}}_{IJ}^{+} = \bar{\boldsymbol{K}}_{IJ}^{+SCNI} + \bar{\boldsymbol{K}}_{IJ}^{+NSNI} \tag{223}$$

and $\bar{\boldsymbol{K}}_{IJ}^{-SCNI}$ and $\bar{\boldsymbol{K}}_{IJ}^{+SCNI}$ are stiffness matrix integrated by the following nodal quadrature

$$\begin{aligned} \bar{\boldsymbol{K}}_{IJ}^{-SCNI} &= \int_{\Omega^-} \bar{\boldsymbol{B}}_I^{-T} \boldsymbol{C}^- \bar{\boldsymbol{B}}_J^- d\Omega = \sum_{L \in S^-} \int_{\Omega_L^-} \bar{\boldsymbol{B}}_I^{-T} \boldsymbol{C}^- \bar{\boldsymbol{B}}_J^- d\Omega \\ &\approx \sum_{L \in S^-} \tilde{\bar{\boldsymbol{B}}}_I^{-T}(\boldsymbol{x}_L) \boldsymbol{C}^-(\boldsymbol{x}_L) \tilde{\bar{\boldsymbol{B}}}_J^-(\boldsymbol{x}_L) V_L \end{aligned} \tag{224}$$

$$\begin{aligned} \bar{\boldsymbol{K}}_{IJ}^{+SCNI} &= \int_{\Omega^+} \bar{\boldsymbol{B}}_I^{+T} [\![\boldsymbol{C}]\!] \bar{\boldsymbol{B}}_J^+ d\Omega = \sum_{L \in S^+} \int_{\Omega_L^+} \bar{\boldsymbol{B}}_I^{+T} [\![\boldsymbol{C}]\!] \bar{\boldsymbol{B}}_J^+ d\Omega \\ &\approx \sum_{L \in S^+} \tilde{\bar{\boldsymbol{B}}}_I^{+T}(\boldsymbol{x}_L) [\![\boldsymbol{C}]\!](\boldsymbol{x}_L) \tilde{\bar{\boldsymbol{B}}}_J^+(\boldsymbol{x}_L) V_L \end{aligned} \tag{225}$$

where V_L is the volume for L-th nodal representative domain. The smoothed gradient $\tilde{\bar{\boldsymbol{B}}}_I^-(\boldsymbol{x}_L)$ is calculated by Eq. (16) given in Chapter 3:

$$\tilde{\bar{\boldsymbol{B}}}_I^-(\boldsymbol{x}_L) = \begin{bmatrix} \tilde{\bar{\Psi}}_{I,1}^-(\boldsymbol{x}_L) & 0 & 0 \\ 0 & \tilde{\bar{\Psi}}_{I,2}^-(\boldsymbol{x}_L) & 0 \\ 0 & 0 & \tilde{\bar{\Psi}}_{I,3}^-(\boldsymbol{x}_L) \\ \tilde{\bar{\Psi}}_{I,2}^-(\boldsymbol{x}_L) & \tilde{\bar{\Psi}}_{I,1}^-(\boldsymbol{x}_L) & 0 \\ \tilde{\bar{\Psi}}_{I,3}^-(\boldsymbol{x}_L) & 0 & \tilde{\bar{\Psi}}_{I,1}^-(\boldsymbol{x}_L) \\ 0 & \tilde{\bar{\Psi}}_{I,3}^-(\boldsymbol{x}_L) & \tilde{\bar{\Psi}}_{I,2}^-(\boldsymbol{x}_L) \end{bmatrix} \tag{226}$$

where $\widetilde{\bar{\Psi}}_{I,i}^{-}(\boldsymbol{x}_L)$ is the smoothed gradient at the nodal point $\boldsymbol{x}_L$ calculated following Chen et al. [40]:

$$\widetilde{\bar{\Psi}}_{I,i}^{-}(\boldsymbol{x}_L) = \frac{1}{V_L}\int_{\Omega_L} \bar{\Psi}_{I,i}^{-}(\boldsymbol{x})d\Omega = \frac{1}{V_L}\int_{\partial\Omega_L} \bar{\Psi}_{I}^{-}(\boldsymbol{x})n_i(\boldsymbol{x})d\Gamma \tag{227}$$

where n_i denotes the i^{th} component of the outward unit normal vector to the smoothing domain boundary. Same computation is applied for $\widetilde{\bar{\boldsymbol{B}}}_I^{+}$.

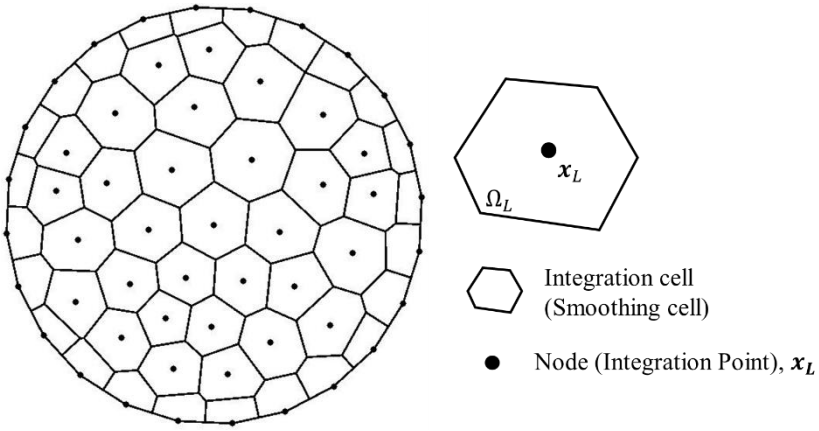


Figure 32. Voronoi cell diagram in two-dimensional domain Ω.

It was shown in [40] that with the smoothed gradient of shape function in Eq. (16), the first order integration constraint is exactly satisfied. As discussed in [118], in order to maintain linear consistency of the smoothed gradient of a linearly consistent shape function, a simple one-point Gauss integration rule can be used for the contour integral in Eq. (227). In this study, the Voronoi tessellation is employed to generate such integration cell diagram, as shown in Figure 32.

Spurious oscillatory modes can be triggered in nodal integration methods, and therefore the naturally stabilized nodal integration (NSNI) [119] is employed, resulting stabilization terms $\bar{\boldsymbol{K}}_{IJ}^{-NSNI}$ and $\bar{\boldsymbol{K}}_{IJ}^{+NSNI}$ in Eqns. (222) and (223), expressed as

$$\bar{\boldsymbol{K}}_{IJ}^{-NSNI} = \sum_{L\in S^-}\sum_{i=1}^{d} \bar{\boldsymbol{B}}_{I,i}^{-\nabla^T}(\boldsymbol{x}_L)\boldsymbol{C}^-\bar{\boldsymbol{B}}_{J,i}^{-\nabla}(\boldsymbol{x}_L)M_{iL} \tag{228}$$

$$\bar{\boldsymbol{K}}_{IJ}^{+NSNI} = \sum_{L\in S^-}\sum_{i=1}^{d} \bar{\boldsymbol{B}}_{I,i}^{+\nabla^T}(\boldsymbol{x}_L)\boldsymbol{C}^-\bar{\boldsymbol{B}}_{J,i}^{+\nabla}(\boldsymbol{x}_L)M_{iL} \tag{229}$$

where $\bar{\boldsymbol{B}}_{I,i}^{-\nabla}$ is defined as follows:

$$\bar{\boldsymbol{B}}_{Ii}^{-\nabla} = \begin{bmatrix} \bar{\Psi}_{Ii,1}^{-\nabla} & 0 & 0 \\ 0 & \bar{\Psi}_{Ii,2}^{-\nabla} & 0 \\ 0 & 0 & \bar{\Psi}_{Ii,3}^{-\nabla} \\ \bar{\Psi}_{Ii,2}^{-\nabla} & \bar{\Psi}_{Ii,1}^{-\nabla} & 0 \\ \bar{\Psi}_{Ii,3}^{-\nabla} & 0 & \bar{\Psi}_{Ii,1}^{-\nabla} \\ 0 & \bar{\Psi}_{Ii,3}^{-\nabla} & \bar{\Psi}_{Ii,2}^{-\nabla} \end{bmatrix} \tag{230}$$

where $\bar{\Psi}_{Ii,j}^{-\nabla}$ is calculated by the direct differentiation of the first-order implicit gradient $\bar{\Psi}_{Ii}^{-\nabla}$ with respect to x_j [119]. The terms M_{iL} in Eqns. (228) and (229) is the second moments of inertia in each nodal integration domain, calculated from Eq. (231) (same as the one shown in Chapter 3)

$$M_{iL} = M_i(\boldsymbol{x}_L) = \int_{\Omega_L} (x_i - x_{Li})^2 d\Omega \tag{231}$$

From Eqns. (228) to (231) , no subdivision of integration cells is required in the stabilization. The matrix $\bar{\boldsymbol{K}}_{IJ}^{-NSNI}$ and $\bar{\boldsymbol{K}}_{IJ}^{+NSNI}$ are introduced to maintain coercivity of the system due to reduced nodal integration.

The higher order SNNI [163] (the non-conforming version of higher order SCNI (H-SCNI) [164]) is employed for the domain integration for computing $\widehat{\bar{\boldsymbol{K}}}_{IJ}$ in Eq. (167) since the local bubble function is constructed based on non-conforming domain as explained in *Remark 5.2.2.1*.

$$\hat{\hat{\boldsymbol{K}}}_{IJ}^{-} = \int_{\Omega^-} \hat{\boldsymbol{B}}_I^{-T} \boldsymbol{C}^- \hat{\boldsymbol{B}}_J^- d\Omega \approx \sum_{L \in S^-} \sum_{M \in S_L^-} \hat{\boldsymbol{B}}_I^{-T}(\boldsymbol{x}_L^M) \boldsymbol{C}^-(\boldsymbol{x}_L^M) \hat{\boldsymbol{B}}_J^-(\boldsymbol{x}_L^M) V_L^M \tag{232}$$

where $S_L^- = \{M | \boldsymbol{x}_L^M \in \Omega_L^{M-}\}$ is the node set containing the subdomain of non-conforming nodal domain as shown in Figure 33. V_L^M is the volume of subdomain Ω_L^{M-}. In this study, the size of the non-conforming nodal domain is chosen to be a circular domain (for two-dimensional case) where its volume is equal to the nodal volume V_L.

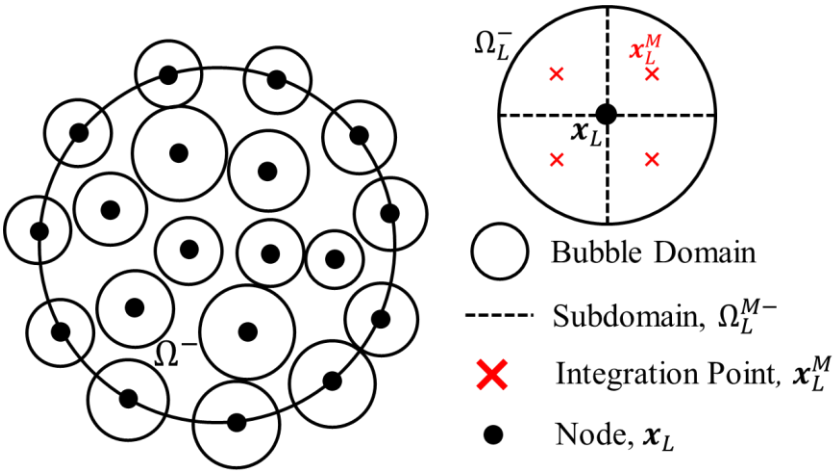


Figure 33. Non-conforming nodal diagram in two-dimensional domain Ω.

For all other domain integration terms shown in Chapter 5.3, direct nodal integration is employed:

$$\int_{\Omega^-} \boldsymbol{P}(\boldsymbol{x}) d\Omega \approx \sum_{L \in S^-} \boldsymbol{P}(\boldsymbol{x}_L) V_L \tag{233}$$

$$\int_{\partial\Omega_D^-} \boldsymbol{Q}(\boldsymbol{x}) d\Gamma \approx \sum_{L | \boldsymbol{x}_L \in \partial\Omega^-} \boldsymbol{Q}(\boldsymbol{x}_L) A_L \tag{234}$$

where A_L is the boundary surface area.

5.3 Numerical Examples

In this study, benchmark numerical examples are solved to examine the performance of the proposed variational multiscale immersed method (VMIM). The reproducing kernel approximation with a linear basis and cubic B-spline kernel are adopted for the approximation of displacement fields and Lagrange multipliers shown in Eqns. (162) to (164), for which circular support with a normalized support size equaling 2.0 is employed. The Nitsche's formulation with normalized penalty parameter of 100 is employed to impose the boundary condition on $\partial\Omega_D^-$.

The following methods with their abbreviation are given and tested:

(1). **IM** + **Penalty**: Immersed Method with Penalty Method, Eq. (135)

(2). **IM** + **LM**: Immersed Method with Lagrangian Multiplier Method, Eq. (138)

(3). **IM** + **Nitsches**: Immersed Method with Nitsche's Method, Eq. (139)

(4). **VMIM(bubble) + LM**: Variational Multiscale Immersed Method by residual-free bubble method with Lagrangian Multiplier Method, Eq. (194)

(5). **VMIM(App)** + **LM**: Variational Multiscale Immersed Method by approximated fine-scale solution with Lagrangian Multiplier Method, Eq. (208)

To access the accuracy of different numerical schemes, the following normalized displacement and energy norms are employed:

$$\left\|\boldsymbol{u}-\boldsymbol{u}^h\right\|_{L_2}=\sqrt{\frac{\left(\boldsymbol{u}^{-h}-\boldsymbol{u}^{exact},\boldsymbol{u}^{-h}-\boldsymbol{u}^{exact}\right)_{\Omega^-\setminus\Omega^+}+\left(\boldsymbol{u}^{+h}-\boldsymbol{u}^{exact},\boldsymbol{u}^{+h}-\boldsymbol{u}^{exact}\right)_{\Omega^+}}{\left(\boldsymbol{u}^{exact},\boldsymbol{u}^{exact}\right)_{\Omega}}} \tag{235}$$

$$\left\|\boldsymbol{u}-\boldsymbol{u}^h\right\|_{E}=\sqrt{\frac{\left(\boldsymbol{\varepsilon}^{-h}-\boldsymbol{\varepsilon}^{exact},\boldsymbol{\sigma}^{-h}-\boldsymbol{\sigma}^{exact}\right)_{\Omega^-\setminus\Omega^+}+\left(\boldsymbol{\varepsilon}^{+h}-\boldsymbol{\varepsilon}^{exact},\boldsymbol{\sigma}^{+h}-\boldsymbol{\sigma}^{exact}\right)_{\Omega^+}}{\left(\boldsymbol{\varepsilon}^{exact},\boldsymbol{\sigma}^{exact}\right)_{\Omega}}} \tag{236}$$

in which the superscript "exact" denotes the exact solutions. $\boldsymbol{u}^{-h}$, $\boldsymbol{\varepsilon}^{-h}$, $\boldsymbol{\sigma}^{-h}$ are numerical displacement, strain, and stress from the background domain, respectively and $\boldsymbol{u}^{+h}$, $\boldsymbol{\varepsilon}^{+h}$, $\boldsymbol{\sigma}^{+h}$ are numerical displacement, strain, and stress from the foreground domain, respectively. For domain integration involved in error calculation (235) and (236), a high-order Gauss integration is employed (10×10 Gauss points per background integration cell) is used.

5.3.1 Patch Test

In the first example, the linear patch test is analyzed to verify the accuracy using linear basis in the RK approximation under given nodal integration for the derived variational multiscale immersed formulation. An elasticity equation is considered with the exact solution defined as a linear polynomial function:

$$\boldsymbol{u}^{exact} = \begin{bmatrix} 0.1 + 0.1x_1 + 0.2x_2 \\ 0.05 + 0.15x_1 + 0.1x_2 \end{bmatrix} \tag{237}$$

where the problem contains a square foreground inclusion domain immersed in a square background matrix domain as shown in Figure 34.

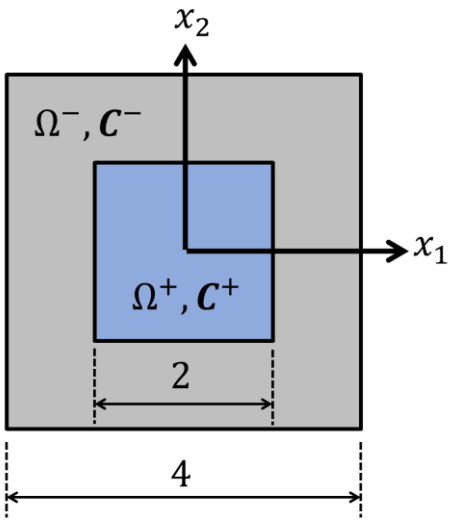


Figure 34. Computational domain for the patch test: a foreground square domain Ω^+ is immersed in a background square domain Ω^-.

For the patch test, the elastic moduli at foreground and background are set to be homogeneous $\boldsymbol{C}^{+} = \boldsymbol{C}^{-}$, with Young's modulus $E^{+} = E^{-} = 1 \times 10^{3}$ and Poisson's ratio $\nu^{+} = \nu^{-} = 0.3$, $\boldsymbol{\varepsilon}^{exact}$ is the exact strain with $\boldsymbol{\varepsilon}^{exact} = [0.1,\ 0.1,\ 0.175]^{T}$; $\boldsymbol{g} = \boldsymbol{u}^{exact}$ is enforced on $\partial\Omega_{g}: (x_1, x_2) \in \partial\Omega^{-}$, and the body force is set to $\boldsymbol{b} = \boldsymbol{0}$. Next, the solution accuracy with model discretization refinement under both uniform and non-uniform discretizations is examined. As shown in Figure 35, non-uniform discretizations are generated by introducing random numbers between $\pm 0.5h$ into the uniform discretizations, where h is the nodal distance in the uniform discretizations of a unit square.

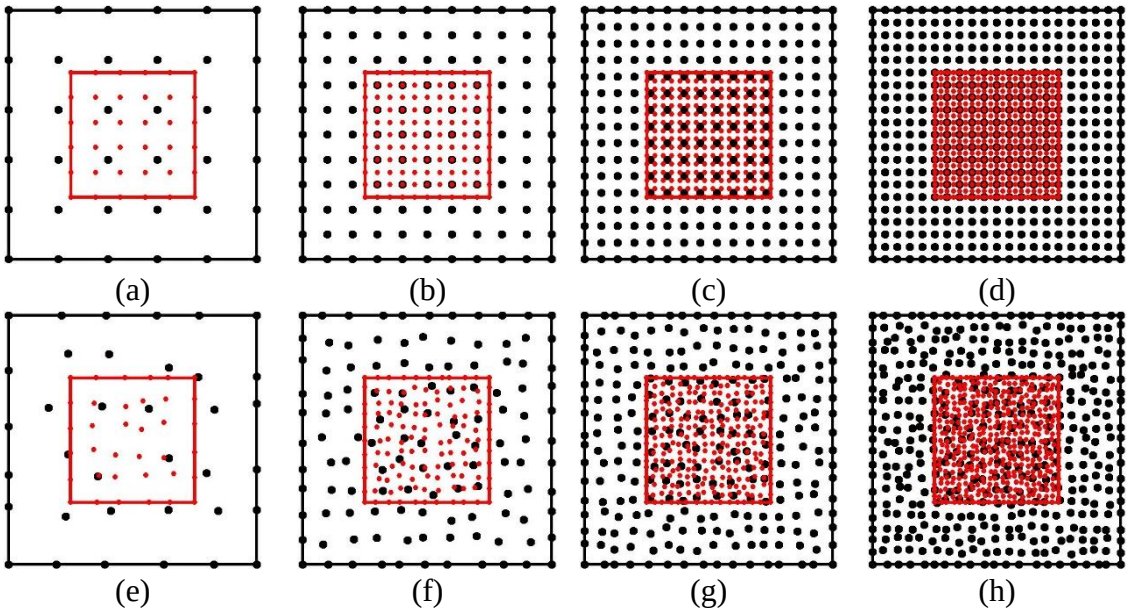


Figure 35. Uniform and non-uniform discretization of a square domain with (a) 6×6, (b) 11×11, (c) 16×16, and (d) 21×21 nodes for both background and foreground domain, where the non-uniform discretizations from (e) to (h) are consisted of randomized nodal distribuitons that corespond to the uniform nodal distribuitons from (a) to (d), respectively.

As shown in Table 3 to Table 6, the L2 and energy norm in uniform and nonuniform discretizations show that all tested methods pass the linear patch test, which shows the quadrature

scheme for both the conventional immersed approach or variational immersed approach all satisfied the integration constraint.

Table 3. L_2 error norm in linear patch tests (uniform discretization)

Quadrature Scheme	Refinement Level			
	6×6	11×11	16×16	21×21
IM + Penalty	0.8E-15	0.4E-15	0.8E-15	1.2E-15
IM + LM	1.0E-15	0.3E-15	0.6E-15	1.2E-15
IM + Nitsches	1.1E-15	0.4E-15	0.6E-15	1.2E-15
VMIM(Bubble) + LM	1.1E-15	0.4E-15	0.6E-15	1.1E-15
VMIM(App) + LM	0.6E-15	0.6E-15	0.9E-15	1.3E-15

Table 4. Energy error norm in linear patch tests (uniform discretization)

Quadrature Scheme	Refinement Level			
	6×6	11×11	16×16	21×21
IM + Penalty	1.9E-15	1.0E-15	0.8E-15	1.2E-15
IM + LM	1.3E-15	0.6E-15	0.7E-15	1.1E-15
IM + Nitsches	1.3E-15	0.5E-15	0.8E-15	1.2E-15
VMIM(Bubble) + LM	1.4E-15	0.5E-15	0.8E-15	1.2E-15
VMIM(App) + LM	1.8E-15	0.9E-15	1.3E-15	1.2E-15

Table 5. L_2 error norm in linear patch tests (nonuniform discretization)

Quadrature Scheme	Refinement Level			
	6×6	11×11	16×16	21×21
IM + Penalty	1.6E-15	0.8E-15	1.7E-15	2.3E-15
IM + LM	2.1E-15	0.7E-15	1.2E-15	2.5E-15
IM + Nitsches	2.2E-15	0.8E-15	1.2E-15	2.5E-15
VMIM(Bubble) + LM	2.2E-15	0.7E-15	1.2E-15	2.4E-15
VMIM(App) + LM	1.3E-15	1.2E-15	1.8E-15	2.6E-15

Table 6. Energy error norm in linear patch tests (nonuniform discretization)

Quadrature Scheme	Refinement Level			
	6×6	11×11	16×16	21×21
IM + Penalty	3.8E-15	1.2E-15	1.7E-15	2.3E-15
IM + LM	2.7E-15	1.2E-15	1.5E-15	2.5E-15
IM + Nitsches	2.7E-15	1.1E-15	1.6E-15	2.4E-15
VMIM(Bubble) + LM	2.8E-15	1.1E-15	1.5E-15	2.4E-15
VMIM(App) + LM	3.9E-15	1.9E-15	2.6E-15	2.6E-15

5.3.2 Heterogeneous Material Diffusion Problems

In the second numerical example, the heterogeneous material diffusion problem shown in [165] is tested, where the strong form of the heterogeneous material diffusion problem is given in Appendix A (Eqns. (286) to (289)). The derivation of VMIM for the heterogeneous material diffusion problem are given in Appendix. The diffusivity is set to be $k^+ = 100$ and $k^- = 1$ and the problem is subjected to a source term $s = -4$. The problem domain and the discretization are demonstrated in Figure 36.

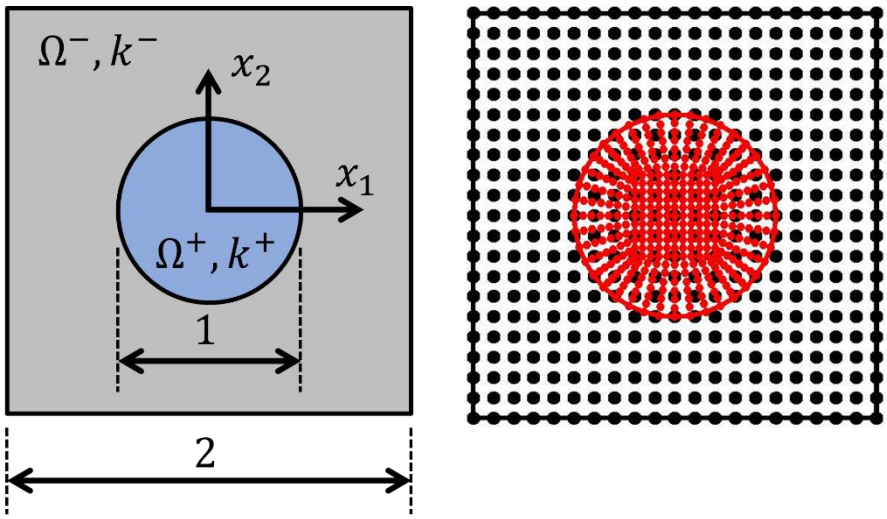


Figure 36. Illustration of heterogeneous material diffusion problem and the numerical discretization. Foreground is discretized into 401 nodes and background is discretized into 441 nodes (regular discretization).

The exact solution of the problem reads:

$$u^{exact}(\boldsymbol{x}) = \frac{x_1^2 + x_2^2}{k^+}, \quad \forall \boldsymbol{x} \in \Omega^+$$
$$u^{exact}(\boldsymbol{x}) = \frac{x_1^2 + x_2^2}{k^-} + r_0^2\left(\frac{1}{k^+} - \frac{1}{k^-}\right), \quad \forall \boldsymbol{x} \in \Omega^- \tag{238}$$

where $r_0 = 0.5$ is the radius of the circular inclusion. The exact solution $g = u^{exact}$ is employed as the essential boundary conditions at boundary $\partial\Omega_g: (x_1, x_2) \in \partial\Omega^-$. The

performance of conventional immersed formulation with different constraint method is investigated. As shown in Figure 37, remarkable error can be found out in the scalar variable inside the region Ω^+ for immersed method with all three constraint enforcement methods. Also, it can be seen from out Figure 38 that flux field exhibits strong oscillation. The numerical investigation shows the conventional immersed approach yields unstable and inaccurate solution with different constraint enforcement method.

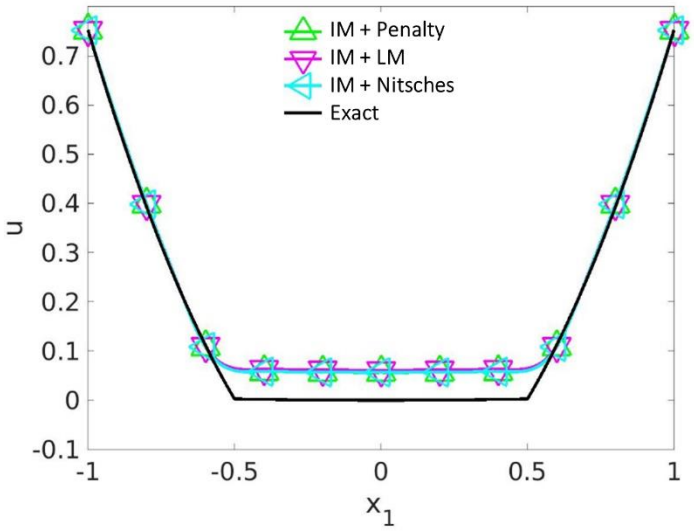
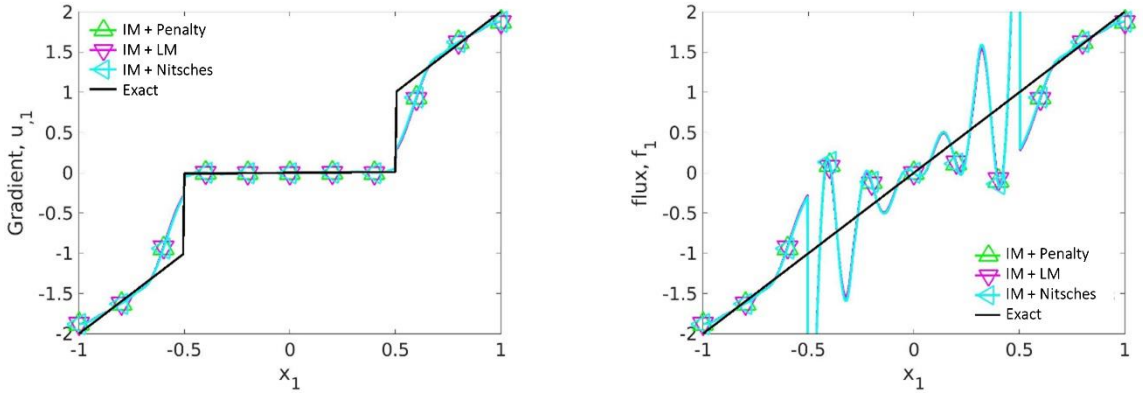


Figure 37. Numerical solution of scalar variable field along $x_2 = 0$ by immersed method with penalty, Lagrange multiplier, and Nitsche's method, compared with exact solution.



(a) Gradient field (b) flux field

Figure 38. Numerical solution of gradient and flux field along $x_2 = 0$ by by immersed method with penalty, Lagrange multiplier, and Nitsche's method, compared with exact solution.

On the other hand, from Figure 39, by employing the variational multiscale immersed method, the solution becomes much more accurate in the scalar variable field. Also, the solution in the gradient field becomes much more accurate close to the interface; oscillation in the flux field is largely reduced, as shown in Figure 40, where only small perturbation near the material interface is found. The introduction of fine-scale features not only increases the accuracy of the solution but also enhances the numerical stability.

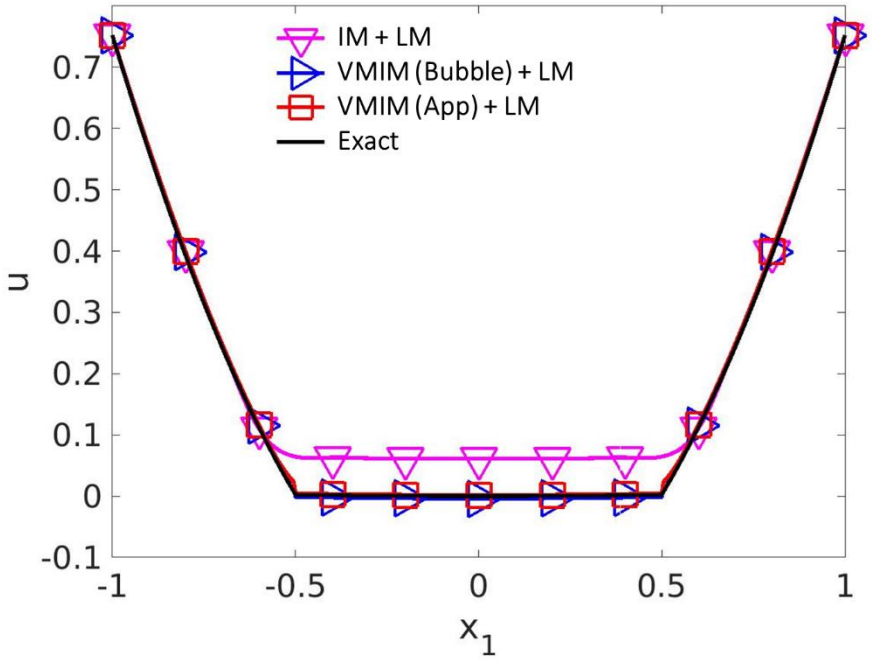
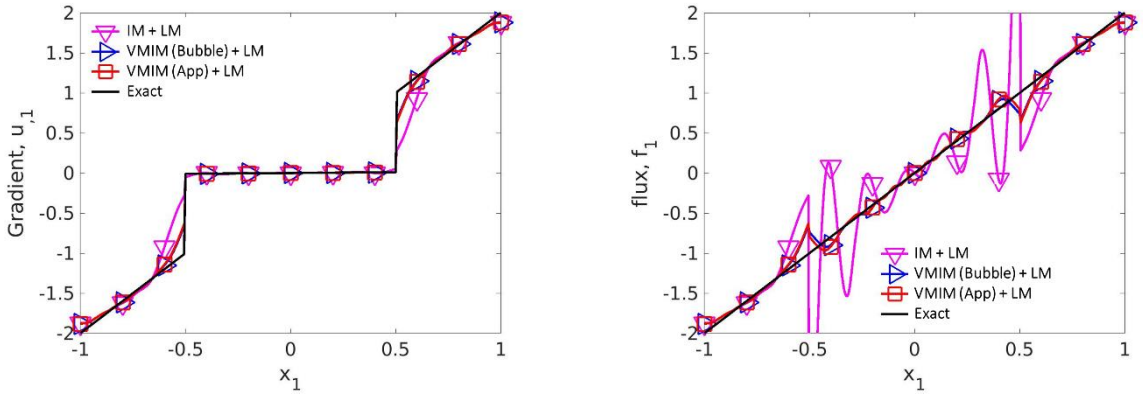


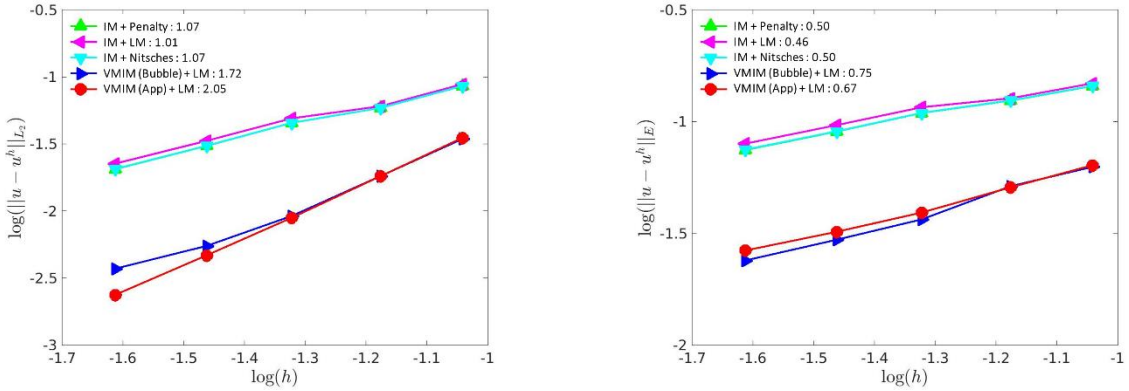
Figure 39. Numerical solution of scalar variable field along $x_2 = 0$ by immersed method and variational multiscale immersed method, compared with exact solution.



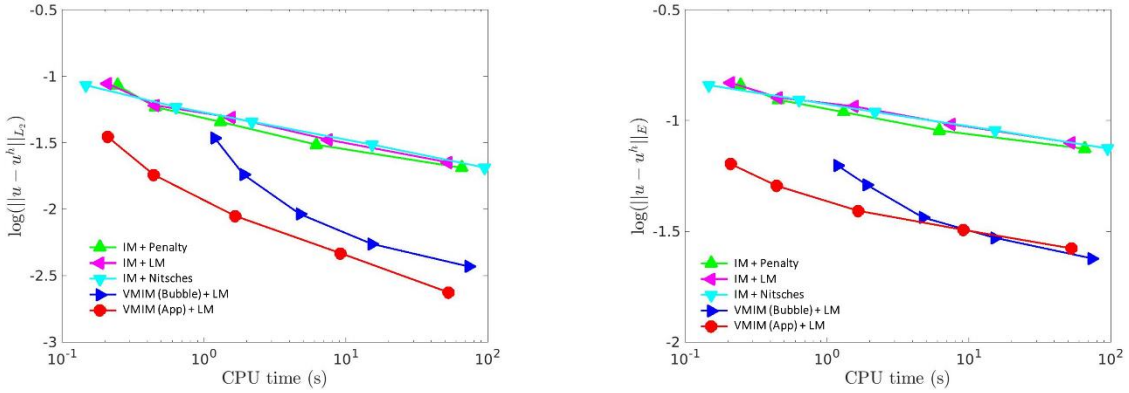
(a) Gradient field (b) flux field

Figure 40. Numerical solution of radient and flux field along $x_2 = 0$ by immersed method and variational multiscale immersed method, compared with exact solution.

The convergence rate and computational costs for all tested methods are given in Figure 41. From the convergence study, the VMIMs with Lagrange multipliers show optimal convergence rate comparing to conventional IMs for both L2 and energy norm. From the computational cost in Figure 41 (b), it can be found out the VMIMs result in similar computational costs compared to IMs. The computational cost of VMIM with the bubble method (method 1) is higher than the approximated approach (method 2) due to the domain subdivision required in numerical integration for the fine-scale equation.



(a) Convergence Rate



(b) Computational Cost

Figure 41. Convergence rate and computational cost of all tested methods. The vales in the legends indicates the average convergence rate.

Finally, the distribution of scalar variable, gradient and flux field are demonstrated in Figure 42, where it can be seen clearly seen that the flux field from IM + LM results in oscillatory results, while the results from VMIMs are much more tsable. Small perturbation are found in VMIMs, but the overall performance is statisfied. The reason of perturbation near the material interface is unclear at the current stage, which shall be investigated further in the future work.

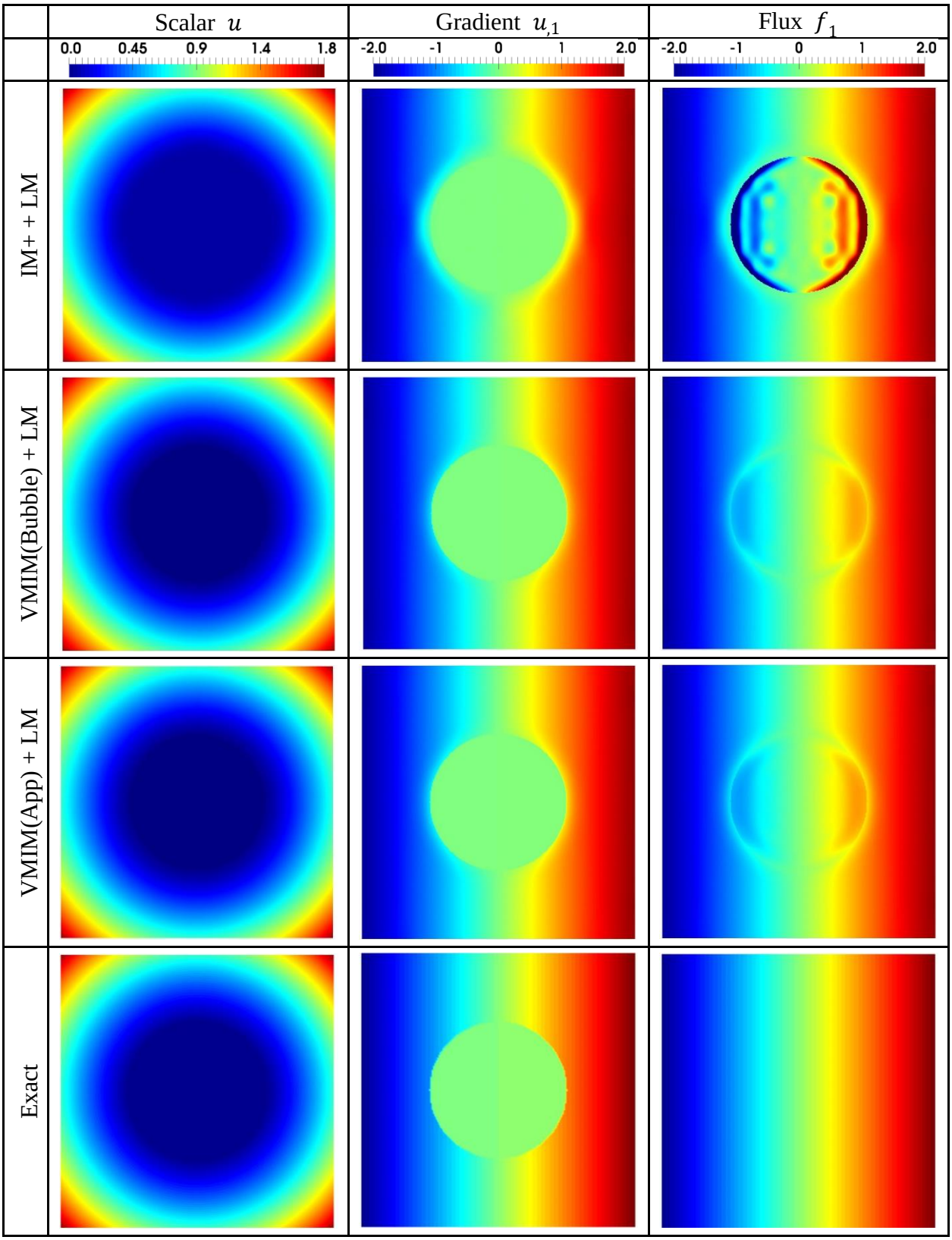
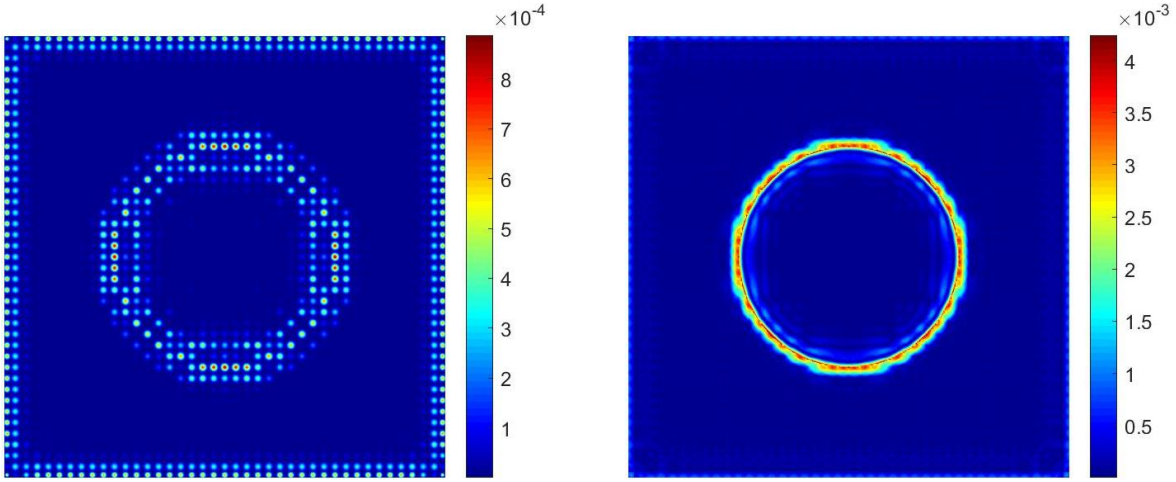


Figure 42. Computational domain for the patch test: a foreground square domain Ω^+ is immersed in a background square domain Ω^-.

The fine-scale solution from the results of VMIMs can be found out in Figure 43, where the peak value lies near the material interfaces. As mentioned in Chapter 5.2, the fine-scale solution plays a rule in indicating the error between the coarse-scale solution and true solution. Here, it can be verified that errors near the material interface are larger due to the low continuity (C^0) in true solution near the interface, and therefore the fine-scale solution can be used as *a-posterior* error indicator for the heterogeneous material problem via the immersed method, which is consistent with the theory of VMS [9].



(a) Fine-scale solution, VMIM(Bubble) (b) Fine-scale solution, VMIM(App)

Figure 43. Fine-scale solution $|\hat{u}^-|$ by (a) residual-free bubble method and (b) approximated fine-scale solution.

5.3.3 Circular inclusion in an infinite plate subject to far-field traction

In the third numerical example, the linear elasticity problem in heterogeneous material [166] is tested, where a circular inclusion with radius R in an infinite plate is under a far-field traction P as shown in Figure 44. The computational domain is a finite square domain with length L. The Young's modulus $E^+ = 1 \times 10^5$ is used for the inclusion and $E^- = 1 \times 10^3$ is used for the background matrix media. Poisson ratio $v^+ = v^- = 0.3$ is used for both domains.

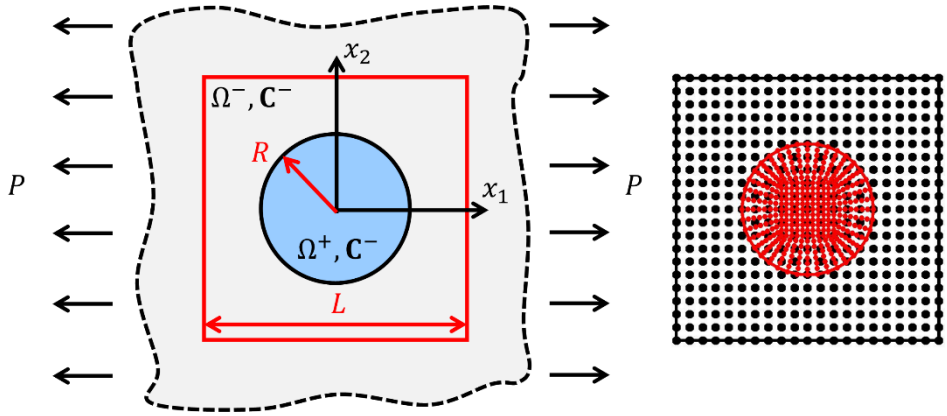


Figure 44. Illustration of circular inclusion problem and the numerical discretization. Foreground is discretized into 401 nodes and background is discretized into 441 nodes (regular discretization).

The plane stress condition is assumed and the exact solution in the background matrix domain reads

$$u_r^+ = \frac{PR}{8\mu^+}\left\{\left[\frac{r(\kappa^+ - 1)}{R} + 2\gamma_1\frac{R}{r}\right] + \left[\frac{2r}{R} + \beta_1(\kappa^+ + 1)\frac{R}{r} + 2\delta_1\frac{R^3}{r^3}\right]\cos(2\theta)\right\}$$
$$u_\theta^+ = \frac{PR}{8\mu^+}\left[\left(-2\frac{r}{R} - \beta_1(\kappa^+ - 1)\frac{R}{r} + 2\delta_1\frac{R^3}{r^3}\right)\right]\sin(2\theta) \quad (239)$$

$$\sigma_{rr}^{exact} = \frac{P}{2}\left[1 - \gamma_1 \frac{R^2}{r^2} + \left(1 - 2\beta_1 \frac{R^2}{r^2} - 3\delta_1 \frac{R^4}{r^4}\right)\cos(2\theta)\right]$$

$$\sigma_{\theta\theta}^{exact} = \frac{P}{2}\left[1 + \gamma_1 \frac{R^2}{r^2} - \left(1 - 3\delta_1 \frac{R^4}{r^4}\right)\cos(2\theta)\right] \tag{240}$$

$$\sigma_{r\theta}^{exact} = -\frac{P}{2}\left(1 + \beta_1 \frac{R^2}{r^2} + 3\delta_1 \frac{R^4}{r^4}\right)\sin(2\theta)$$

where (r, θ) are polar coordinates and β_1, γ_1 and δ_1 are expressed as

$$\beta_1 = \frac{\mu^+(\kappa^- - 1) - \mu^-(\kappa^+ - 1)}{2\mu^- + \mu^+(\kappa^- - 1)}$$

$$\gamma_1 = \frac{\mu^+(\kappa^- - 1) - \mu^-(\kappa^+ - 1)}{2\mu^- + \mu^+(\kappa^- - 1)} \tag{241}$$

$$\delta_1 = \frac{\mu^- - \mu^+}{\mu^+ - \mu^-\kappa^+}$$

$$\mu^- = \frac{E^-}{1 + v^-}, \quad \mu^+ = \frac{E^+}{1 + v^+}$$

$$\kappa^- = \frac{3 - v^-}{1 + v^-}, \quad \kappa^+ = \frac{3 - v^+}{1 + v^+} \tag{242}$$

Then, the exact solution inside the inclusion are given by the following equations

$$u_r^{exact} = \frac{Pr}{8\mu^-}\{\beta_2(\kappa_2 - 1) + 2\delta_2 \cos(2\theta)\}$$

$$u_\theta^{exact} = -\frac{Pr\delta_2}{8\mu^-}\sin(2\theta) \tag{243}$$

$$\sigma_{rr}^{exact} = \frac{P}{2}[\beta_2 + \delta_2 \cos(2\theta)]$$

$$\sigma_{\theta\theta}^{exact} = \frac{P}{2}[\beta_2 - \delta_2 \cos(2\theta)] \tag{244}$$

$$\sigma_{r\theta}^{exact} = -\frac{P\delta_2}{2\mu^-}\sin(2\theta)$$

where β_2 and δ_2 are expressed as

$$\beta_2 = \frac{\mu^-(\kappa^+ + 1)}{2\mu^- + \mu^+(\kappa^- - 1)}$$

$$\delta_2 = \frac{\mu^-(\kappa^+ + 1)}{\mu^+ - \mu^-\kappa^+} \tag{245}$$

The exact solution is employed as the essential boundary conditions for boundary $\partial\Omega_g: (x_1, x_2) \in \partial\Omega^-$. As shown in Figure 45 and Figure 46, the error can be found out in the displacement field for the immersed method with all three constraint enforcement methods. Also, it can be seen from Figure 46 (b) that the stress field exhibit strong oscillation, similar to the case observed in the heterogeneous material diffusion problem. It can be concluded that the conventional immersed approach yields an unstable and inaccurate solution with the three common constraint enforcement methods.

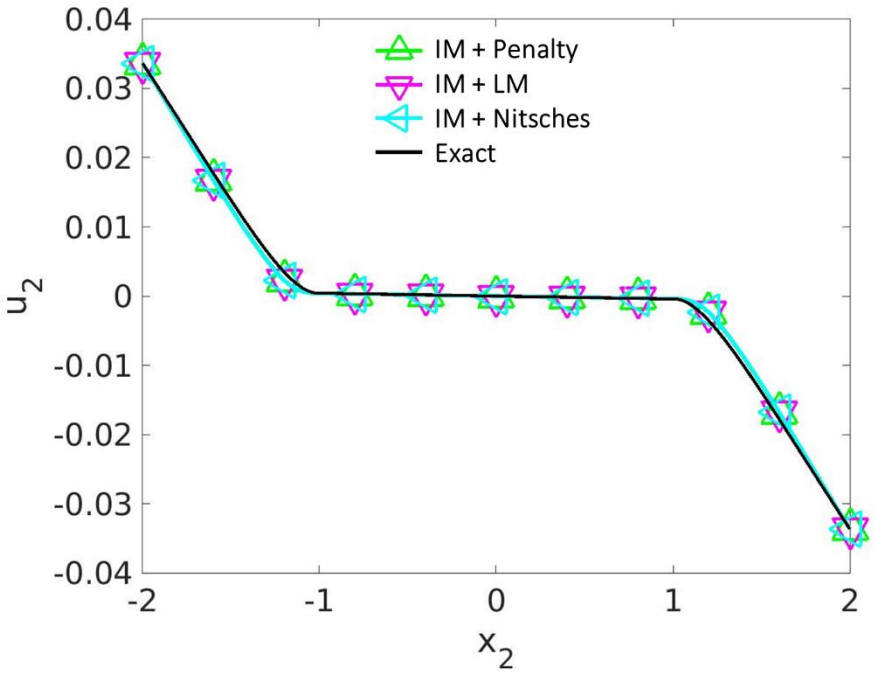
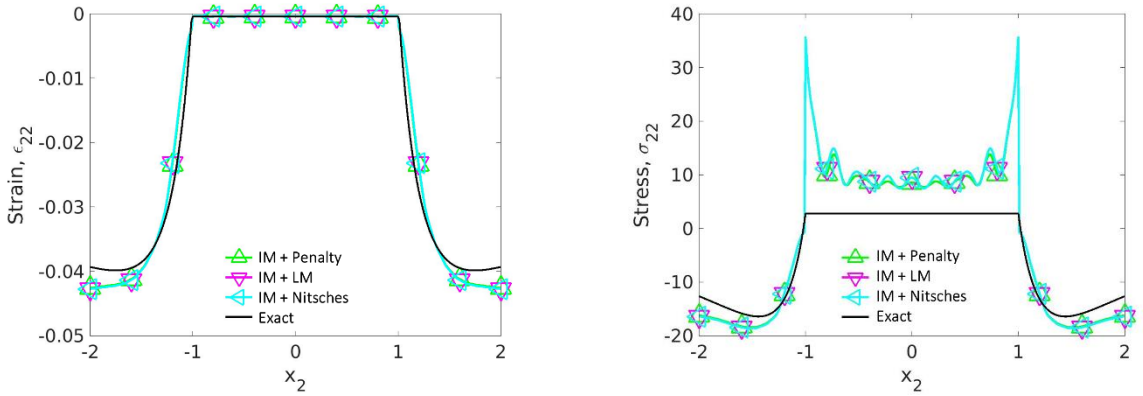


Figure 45. Numerical solution of x-direction displacement along $x_1 = 0$ by immersed method with the penalty, Lagrange multiplier, and Nitsche's method, compared with exact solution.



(a) strain field (b) stress field

Figure 46. Numerical solution of strian and stress field along $x_1 = 0$ by immersed method with the penalty, Lagrange multiplier, and Nitsche's method, compared with exact solution.

On the other hand, from Figure 47 and Figure 48, by employing the variational multiscale immersed method via bubble method or approximation method, the solution becomes much more accurate in all fields. Also, the oscillation in the stress field is largely reduced by VMIMs. The

introduction of fine-scale features not only increases the accuracy of solution but also reduces the numerical instability.

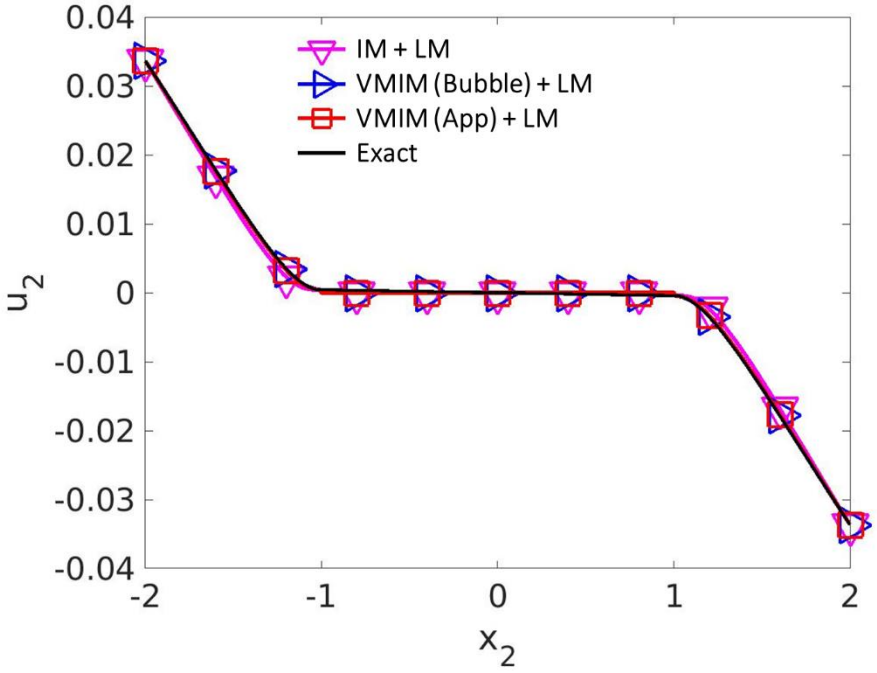
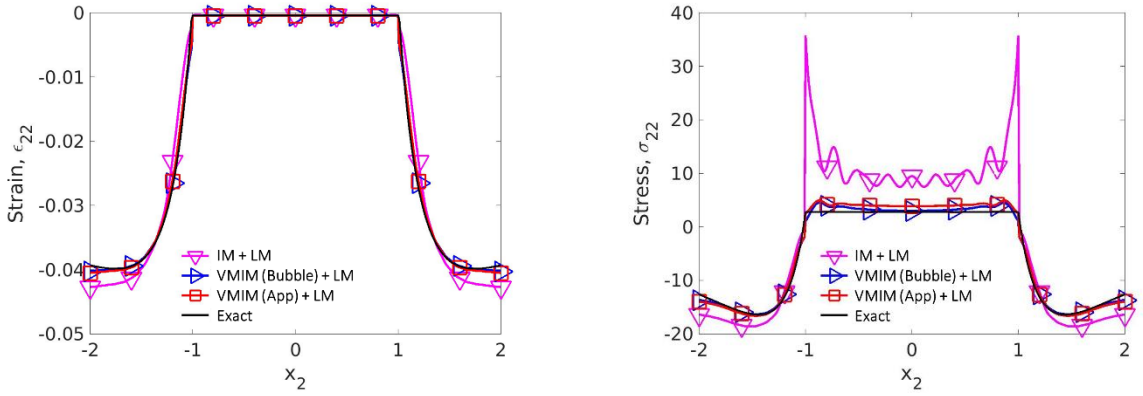


Figure 47. Numerical solution of x-direction displacement along $x_1 = 0$ by immersed method and variational multiscale immersed method, compared with exact solution.

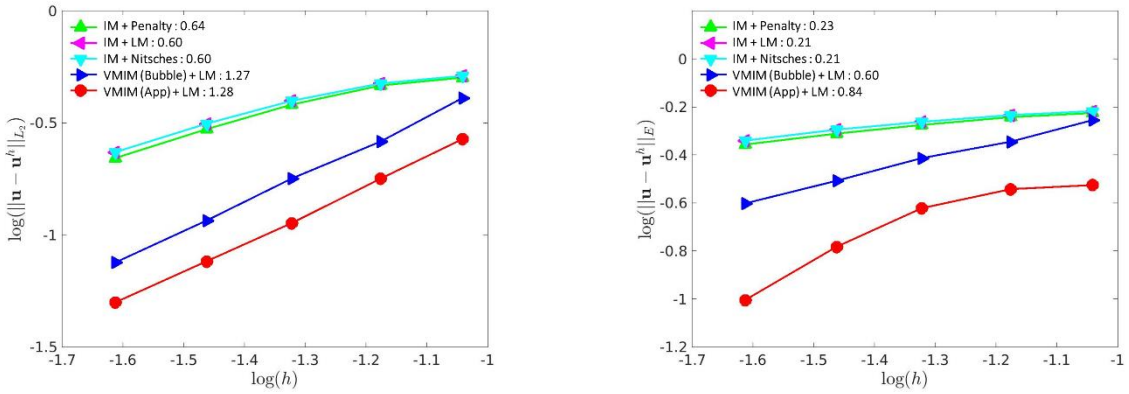


(a) Gradient field (b) flux field

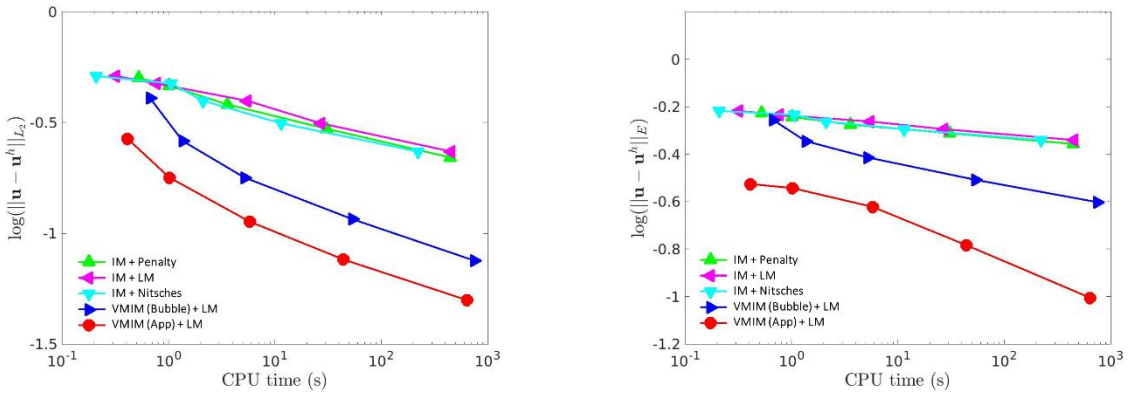
Figure 48. Numerical solution of strian and stress field along $x_1 = 0$ by immersed method and variational multiscale immersed method, compared with exact solution.

The convergence rates and computational costs for all methods are given in Figure 49. From the convergence rate study, the VMIMs with Lagrange multipliers show optimal convergence rate

comparing to conventional IMs for both L2 and energy norm. From the computational cost in Figure 49 (b), it can be found out the cost of VMIMs is slightly higher than that of IMs, but with much better accuracy.



(a) Convergence Rate



(b) Computational Cost

Figure 49. Convergence rate and computational cost of all tested methods. The vales in the legends indicates the average convergence rate.

Finally, the distribution of displacement, strain, and stress field are demonstrated in Figure 50 and Figure 51, where it can be seen clearly seen that the stress field from IM + LM results in oscillatory results, while the results from VMIMs are much more stable. Small perturbation can still be found in VMIM(Bubble) + LM, but the overall performance is satisfied.

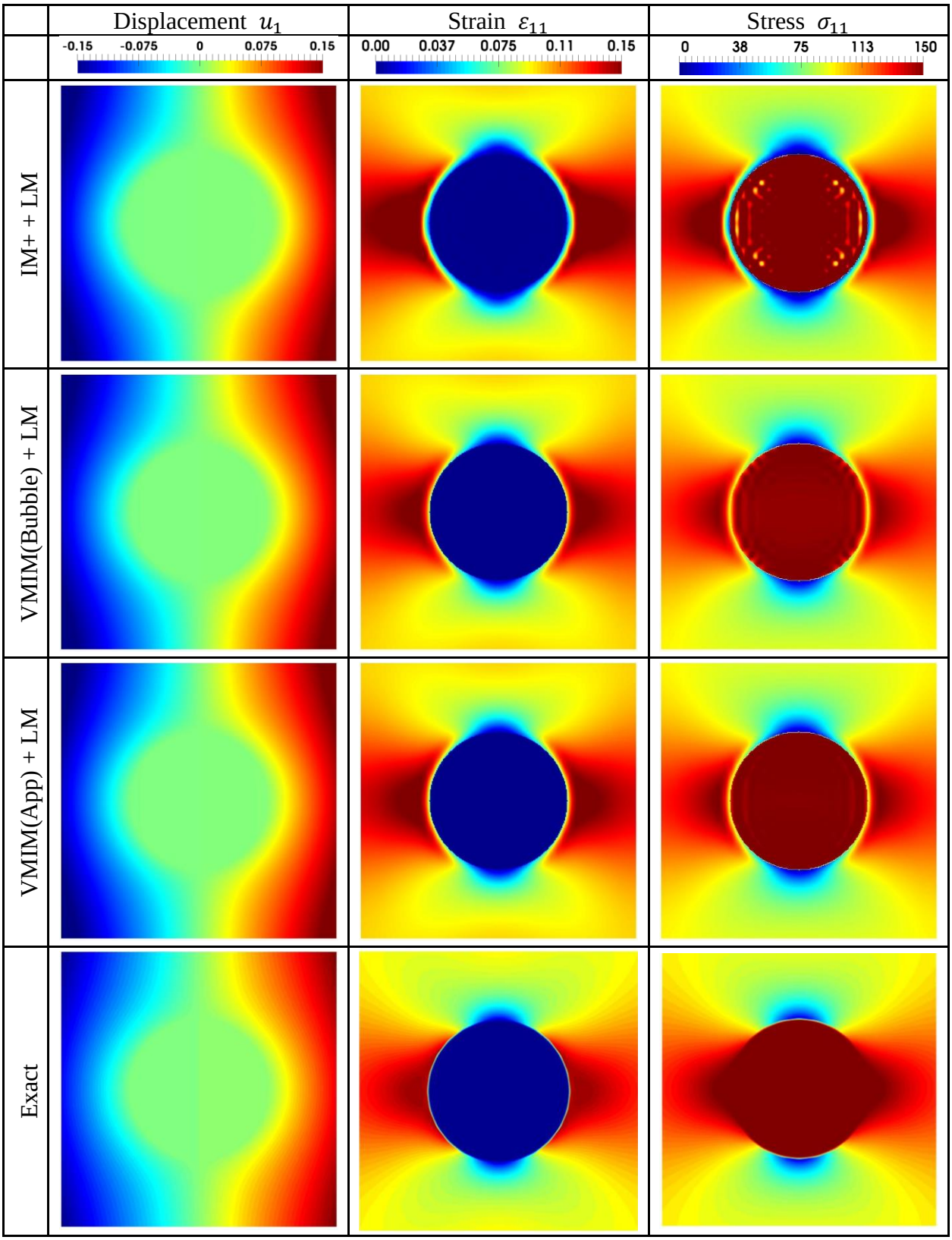


Figure 50. Numerical results of elastic inclusion under far-field tension: displacement u_1, Strain ε_{11}, and stress σ_{11}

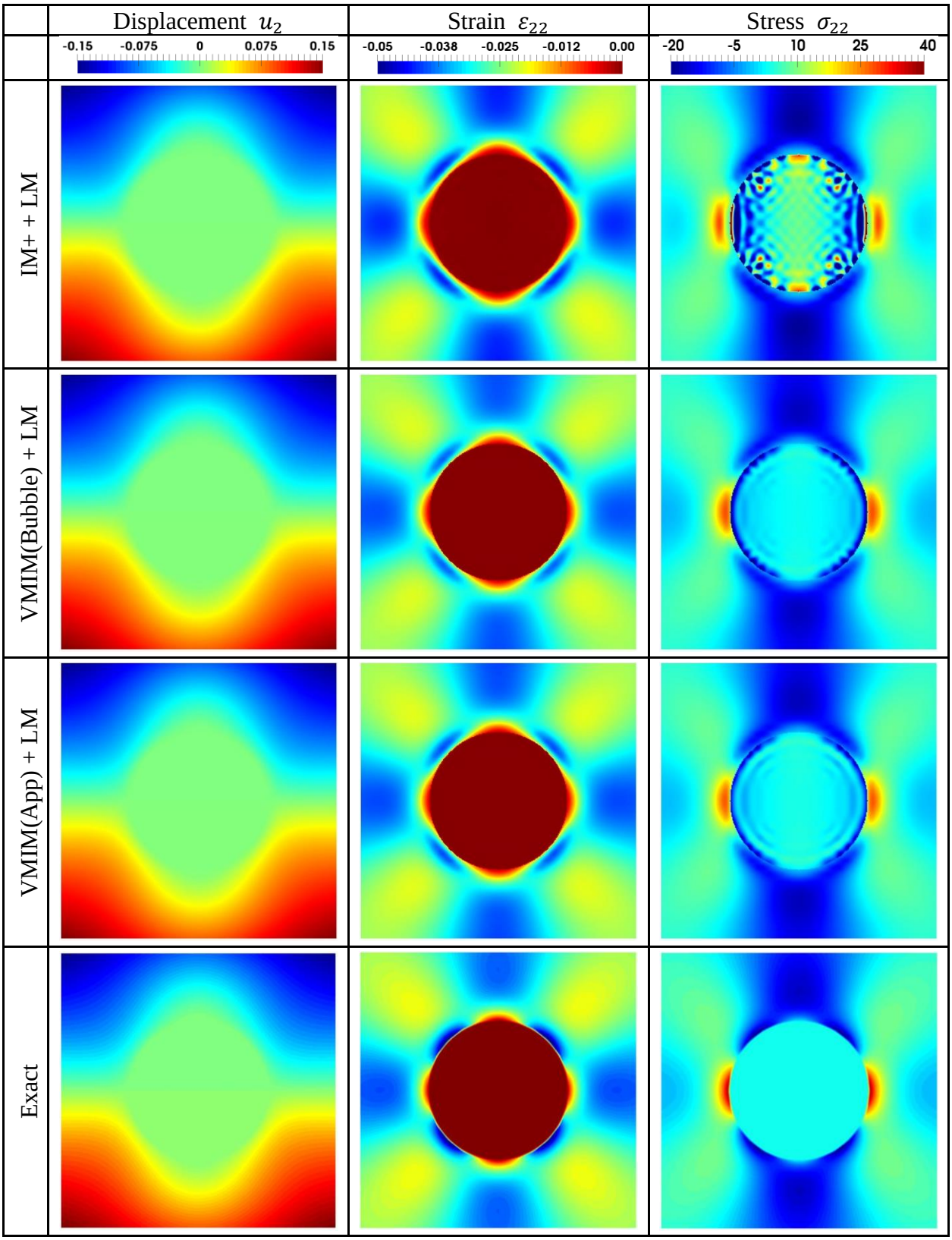
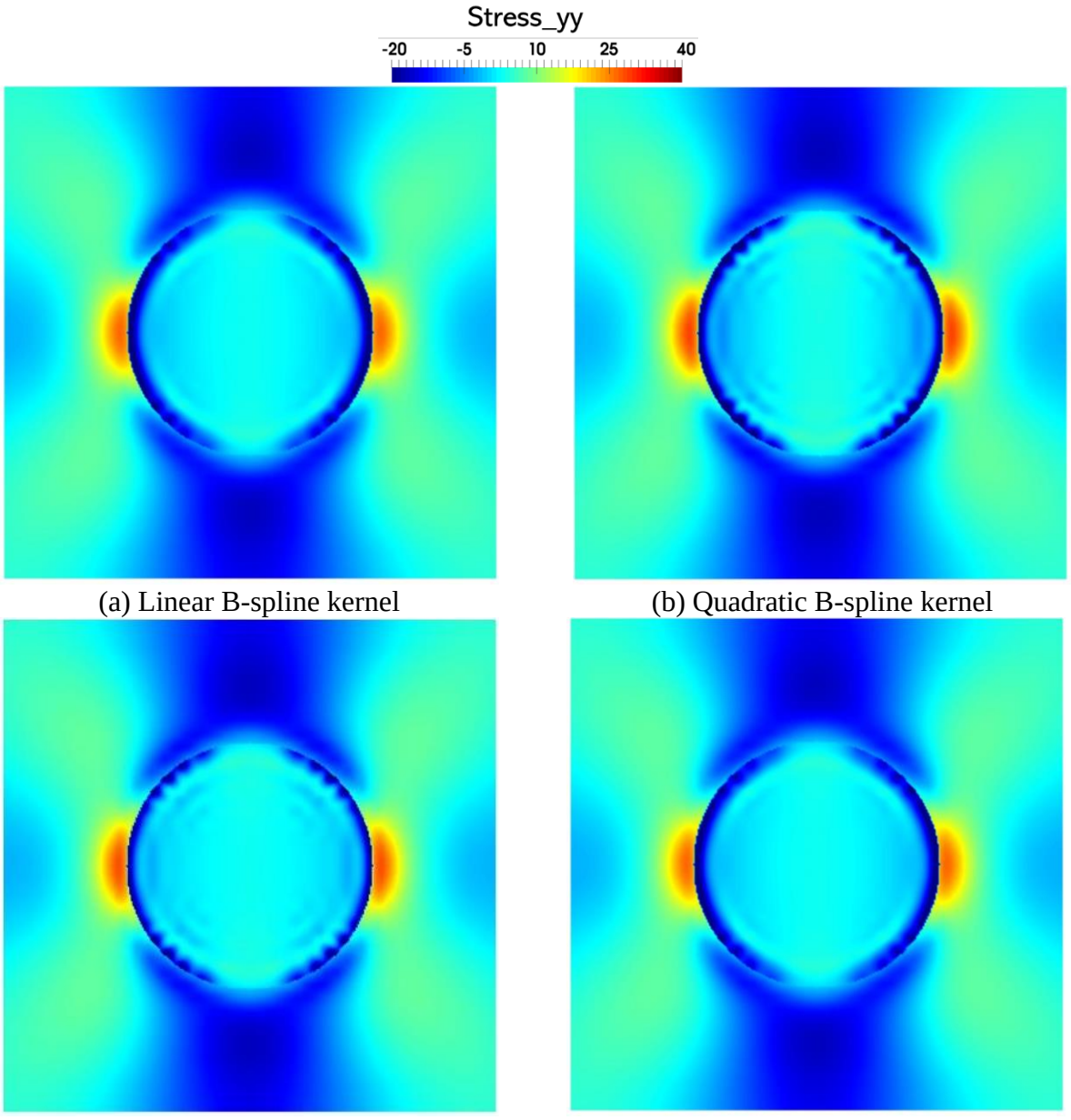


Figure 51. Numerical results of elastic inclusion under far-field tension: displacement u_2, Strain ε_{22}, and stress σ_{22}

Also, the comparison of the numerical results of VMIM(Bubble) + Lagrange Multiplier with different bubble function are expressed in Figure 52. It can be seen that the results by bubble function with different continuity results in almost identical results, where it is found out the numerical oscillation are less remarkable in the case of linear B-spline function and power kernel function.

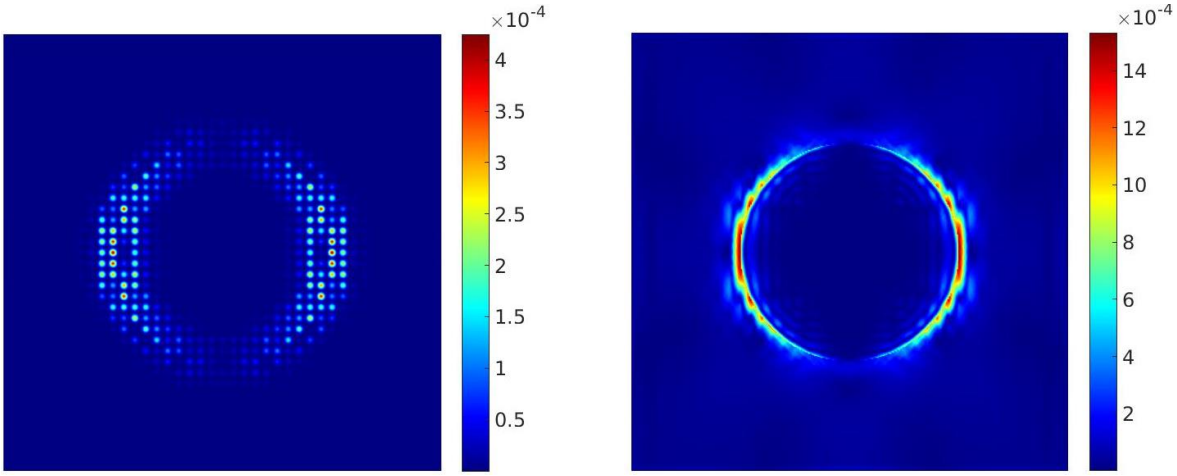


(a) Linear B-spline kernel (b) Quadratic B-spline kernel

(c) Cubic B-spline kernel (d) Power Kernel [39]

Figure 52. Stress distribution by the VMIM(Bubble) + Lagrange Multiplier method with different bubble function.

The fine-scale solution from the results of VMIMs for the heterogeneous material elastic problem can be found out in Figure 53, where the magnitude of fine-scale solution ($|\hat{\boldsymbol{u}}| = \sqrt{(\hat{u}_1^-)^2 + (\hat{u}_2^-)^2}$) have higher error near the two sides of the inclusion. Since the traction is applied in the x_1 direction, it can be expected that more errors are accumulated along x_1 direction at $x_2 = 0$.



(a) Fine-Scale Solution, VMIM(Bubble) (b) Fine-Scale Solution, VMIM(App)

Figure 53. Fine-scale solution $|\hat{\boldsymbol{u}}|$ by two VMIM approaches.

5.3.4 Modeling of Heterogeneous Microstructure

The heterogeneous microstructure usually involves complex geometry which leads to a low-quality mesh for FEM, unless very fine mesh is used, as shown in Figure 54. The proposed meshfree immersed framework can easily handle the discretization for a given microstructure, with accurate and stable numerical results.

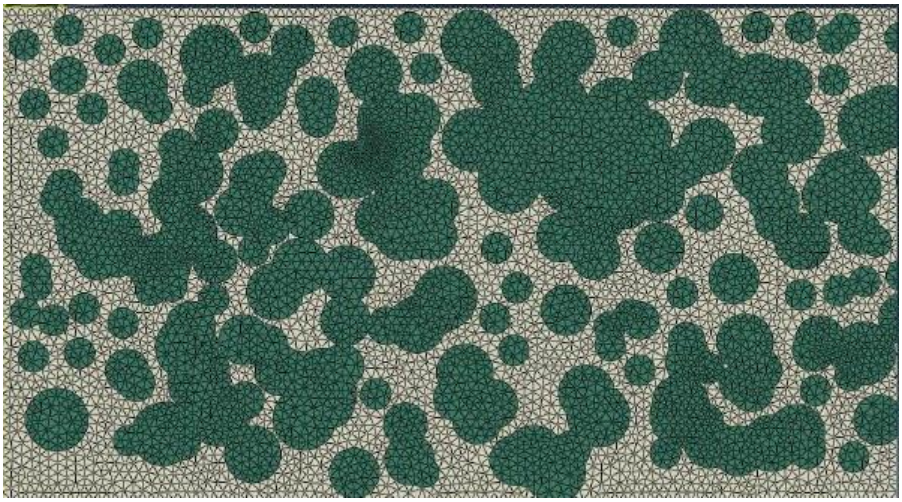


Figure 54. Finite element mesh of the duplex stainless-steel microstructures sample. Figure is obtained from the literature [167].

In this numerical example, a two-dimensional microstructure under tension and shear deformation are tested by prescribing a horizontal and vertical displacement, as shown in Figure 55. The geometry of the microstructure is generated by randomly distributing 35 circular inclusions inside a square domain with unit length. The radii of the inclusions are also randomly picked from 0.05 to 0.15 unit shown in Figure 55. The Young's modulus and Poisson ratio for the background material and foreground inclusions are set to be $(E^-, v^-) = (2, 0.4)$ and $(E^+, v^+) = (40, 0.1)$, respectively. The plane stress condition is assumed. The numerical discretization for the proposed immersed method is expressed in Figure 76, where 1,108 nodes are used for the inclusions, and 41 by 41 uniform discrete nodes are used for the background domain. In this test, only VMIM with an approximated fine-scale solution is employed due to the better efficiency and accuracy shown in the linear elasticity problem shown in the previous section. A mesh-refined conforming FEM result is employed as the reference solution. Note that in the numerical testing of FEM, the conforming mesh must be refined enough with minimum mesh size 0.005 (correspondingly 44,258 nodes in FEM) to avoid low-quality mesh in this case. In comparison, the

presented immersed formulation only requires 2,789 nodes, and there is no issue of low-quality discretization.

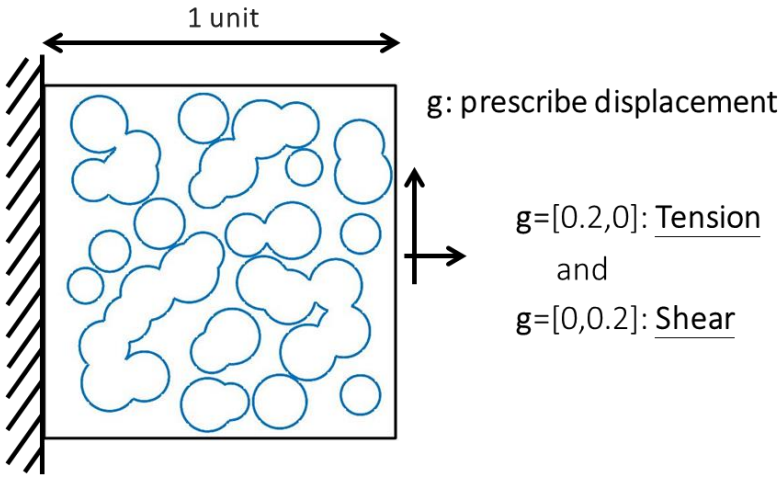


Figure 55. Tension and shear test of the heterogeneous microstructure.

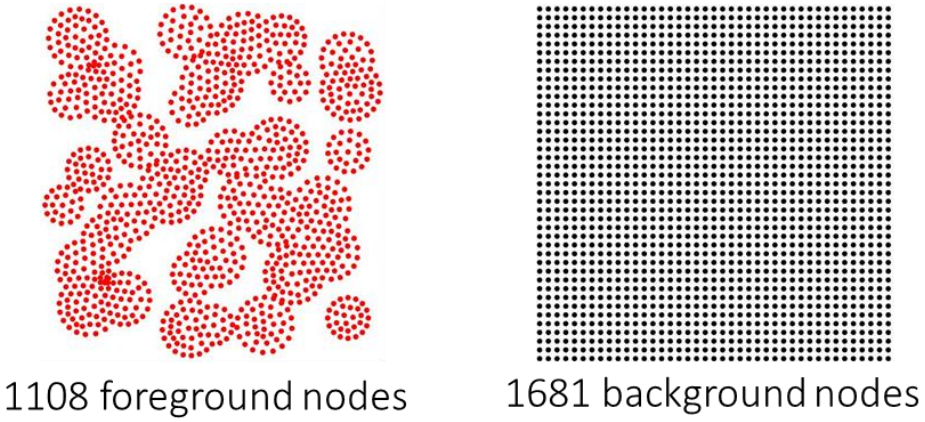


Figure 56. Numerical discretization employed for the given microstructure in Figure 55.

The results of the microstructure under tensile deformation can be found in Figure 57 and Figure 58, where it can be easily found out that the VMIM results in consistent displacement and stress distribution in the microstructure, compared to reference FEM solution. Also, from the stress distribution, VMIM produces non-oscillatory strain and stress solution, and the results is in good

agreement to the reference conforming FEM solutions, while FEM requires 20 times more nodes than the immersed framework.

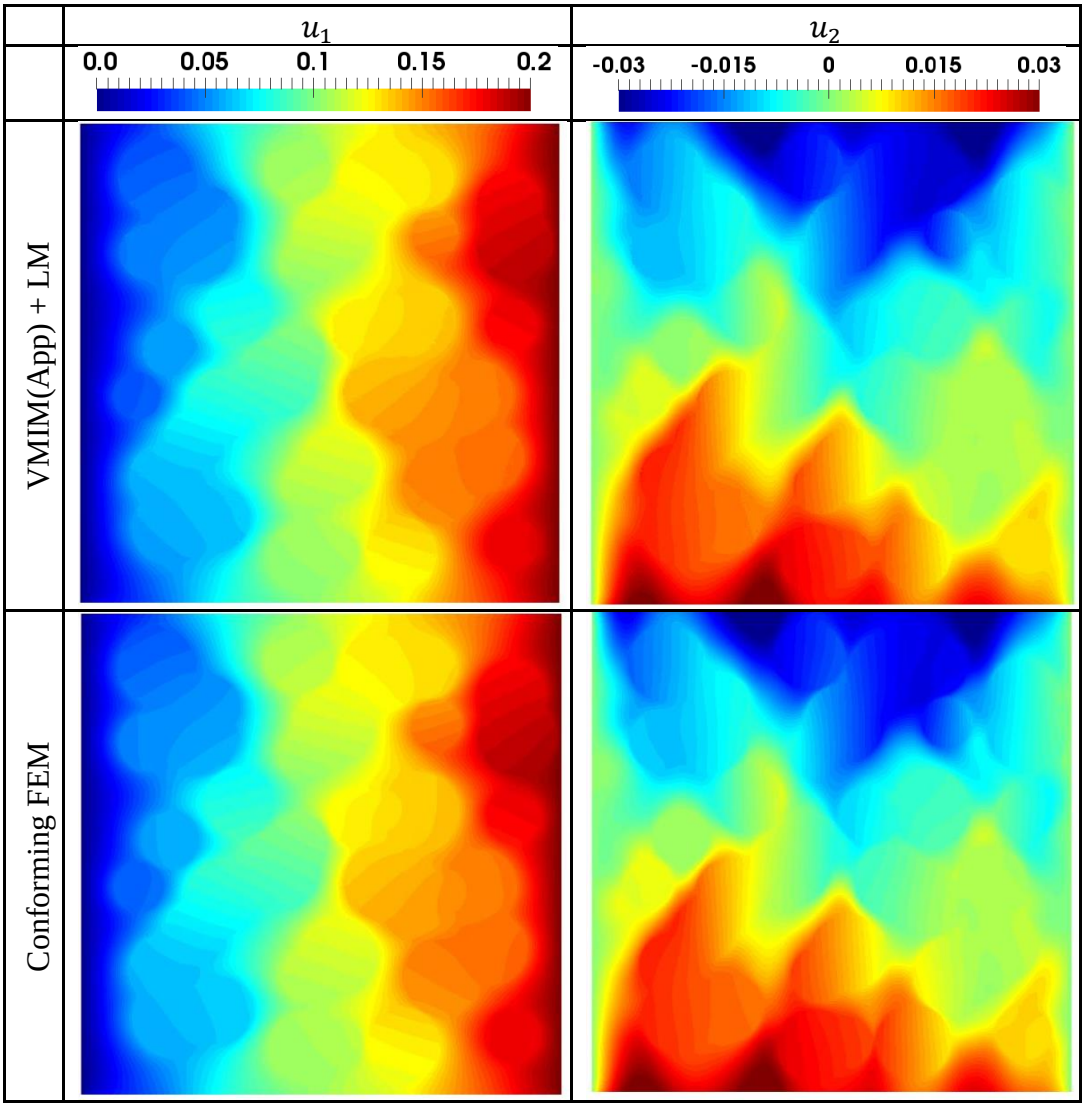


Figure 57. Displacement fields of the microstructure under the tensile deformation.

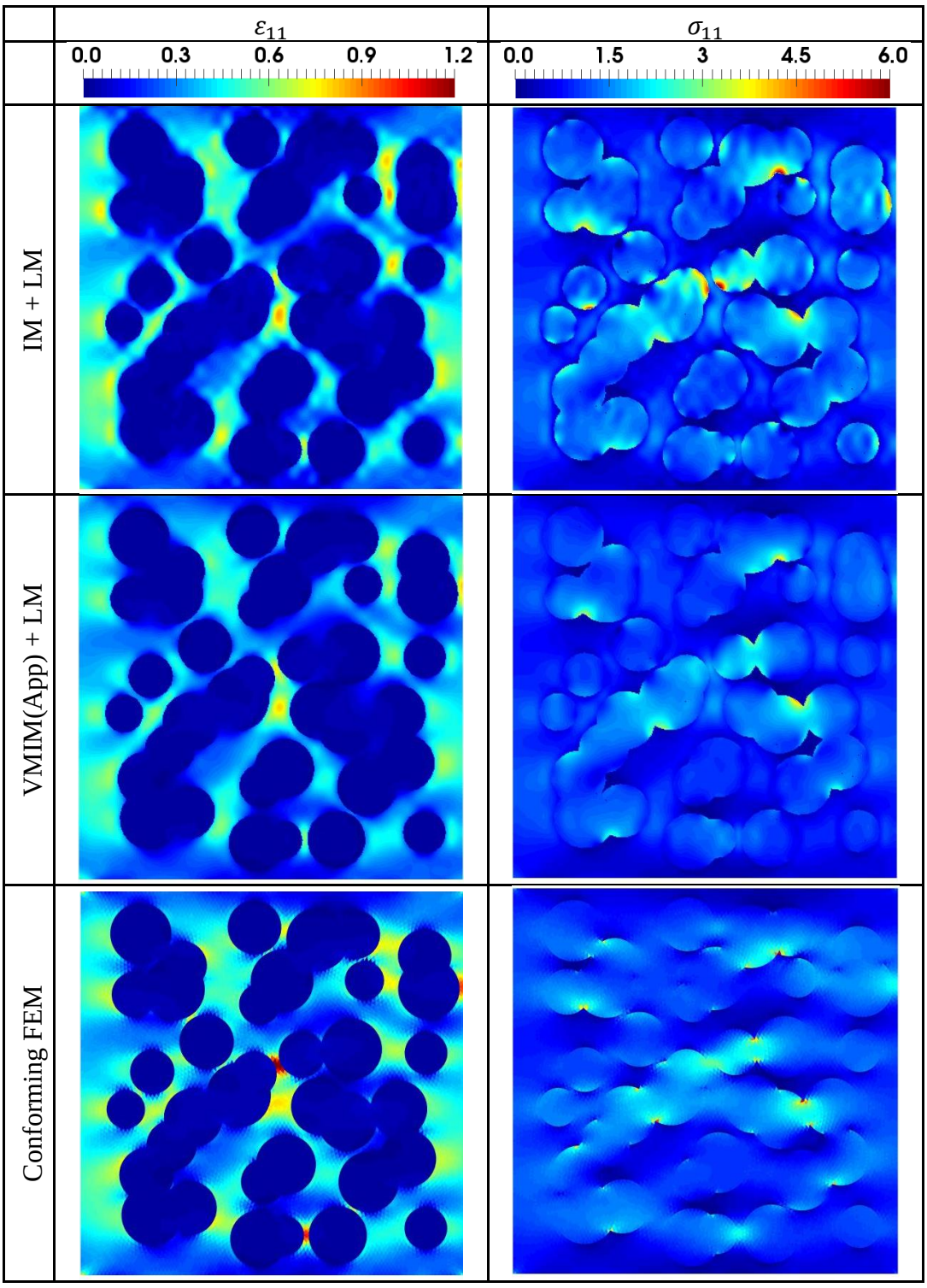


Figure 58. Strain and stress field of the microstructure under the tensile deformation.

The results of the microstructure under shear deformation can be found in Figure 59 and Figure 60. Similar to the tensile test, the VMIM produce smooth and comparable accuracy with a coarser discretization in comparison with the reference FEM solution.

It is noted for both tensile and shear test, the strain concentration for close-by inclusions are smoother in the case of VMIM. It may be due to the smooth features employed in RKPM, and the coarser discretization employed in the meshfree framework. Also, the concave geometry of the foreground inclusions may cause RK support overlapping at region with strong concavity, which shall lose some accuracy. The methodology avoiding such situation is under investigation, such as employing the FE shape function or conforming RK method [86] for foreground inclusion domain.

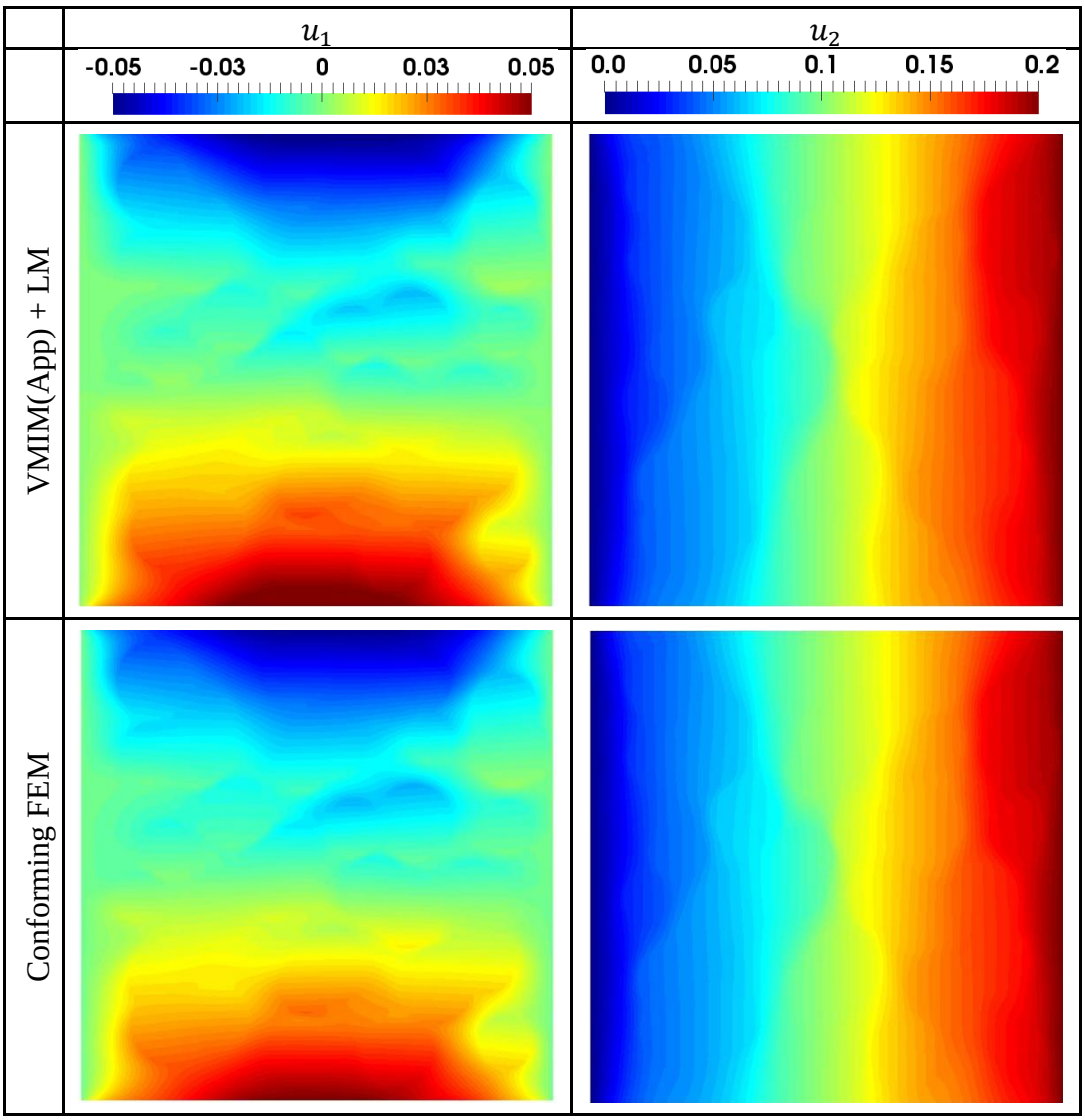


Figure 59. Displacement fields of the microstructure under the shear deformation.

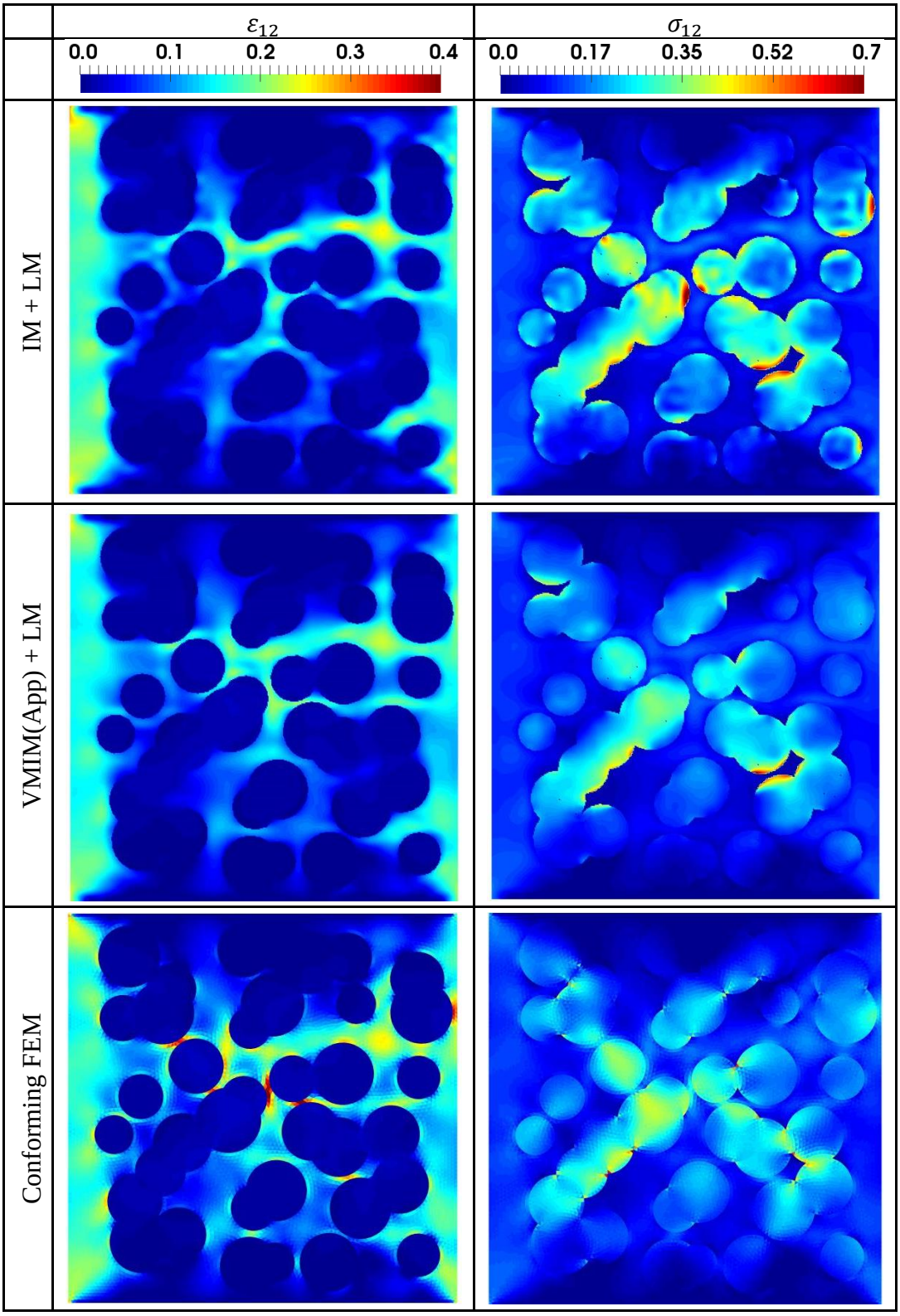


Figure 60. Strain and stress field of the microstructure under the shear deformation.

5.4 Summary of VMIM for Heterogeneous Material Problem

In this study, a robust immersed formulation is developed for the RKPM modeling of heterogeneous material problems. The problem domain is performed under an immersed setting with the volumetric constraint, which avoids numerical difficulties associated with low-quality discretization and issues with complex geometry. A variational multiscale immersed method (VMIM) is presented, where the multiscale decomposition is employed for the solution field. The fine-scale solution represents the error from the coarse-scale equations, and two strategies are used to solve the fine-scale equations: (1) residual-free bubble method, and (2) approximated fine-scale solution. In the first method, the fine-scale equation is solved numerically by the introduction of fine-scale approximation via nodal bubble functions. In the second method, the fine-scale solution is approximated by taking averaging of the fine-scale Green's function. The substitution of the fine-scale solution into the coarse-scale equations naturally gives a stabilized Galerkin formulation shown in the literature [10], but in an immersed setting. Due to the stress divergence operator required in the VMIM, numerical methods with higher-order continuity are preferred, such as RKPM.

The effectiveness of the proposed variational multiscale immersed meshfree framework is verified by solving several benchmark numerical examples. In the general heterogeneous material problems, VMIM shows enhanced accuracy and stability compared to the conventional immersed method, with comparable efficiency. Furthermore, the proposed immersed framework can effectively model material involving complex geometry, such as the material with complex microstructure. The accuracy of the presented method is comparable with refined conforming FEM, but a coarser discretization is needed. In sum, the proposed novel immersed method is suitable for various heterogeneous material problem that conventional approaches hard to dealing with.

This chapter, in part, is being prepared for submission for publication of the material as it may appear in “TH. Huang, JS. Chen and MR. Tupek. An Immersed Variational Multiscale RKPM Formulation. Part I - Heterogeneous Materials. 2020”. The dissertation author was the primary investigator and author of this paper.

Chapter 6 A Variational Multiscale Immersed RKPM Formulation for Fluid-Structural Interaction

This chapter focuses on developing an immersed RKPM formulation for fluid-structure interaction (FSI) problems. The strong form and weak form of the fluid, solid and coupled system are introduced in Chapter 6.1. The derivation of immersed FSI formulation is given in Chapter 6.2. The numerical examples are expressed in Chapter 6.3 to verify the proposed algorithm.

6.1 Model of Fluid-Structure Interaction

To model the fluid structure interaction by the immersed framework introduced in previous chapter, a solid problem lying in a bounded domain Ω^s is immersed in the background fluid domain Ω^f as shown in Figure 61 (a two-dimensional example).

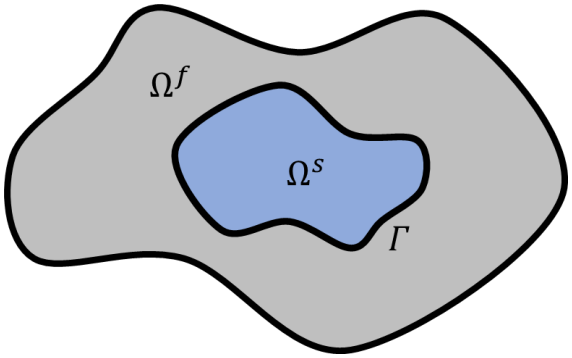


Figure 61. The solid subdomain Ω^s and the fluid subdomain Ω^f.

The solid problem is defined in the domain Ω^s and the fluid problem is defined in the domain Ω^f. The Dirichlet and Neumann jump conditions on the interface Γ are denoted as the fluid-structure interfacial condition. In this chapter, the governing equations of compressible flow and inelastic solid are considered for fluid and solid problem, respectively. Their strong and weak forms are presented in the following sections.

6.1.1 Solid Problem

In this study, the solid problem lying in the domain Ω^s is written in the updated Lagrangian form as

$$\rho^s - \rho_0^s \det(\boldsymbol{F}) = 0 \text{ on } \Omega^s \tag{246}$$

$$\rho^s \frac{d\boldsymbol{v}^s}{dt} - \nabla \cdot \boldsymbol{\sigma}^s = \rho^s \boldsymbol{b}^s \text{ on } \Omega^s \tag{247}$$

where ρ^s and ρ_0^s are the solid density in the current and reference configuration, respectively. $\boldsymbol{F} =$ is the deformation gradient of solid, $d(\cdot)/dt$ is the material derivative, $\boldsymbol{v}^s$ is the solid velocity, $\boldsymbol{\sigma}^s$ is the solid Cauchy stress, and $\boldsymbol{b}^s$ is the body force for the solid. Note that the superscript "s" denotes the variable for solid domain.

6.1.2 Fluid Problem

In the blast event, the compressible Navier-Stokes equations in the domain Ω^f is considered, where the conservation form consists of balance equations of mass, momentum and energy follows:

$$\frac{\partial \rho^f}{\partial t} + \boldsymbol{\nabla} \cdot \rho^f \boldsymbol{v}^f = 0 \text{ on } \Omega^f \tag{248}$$

$$\frac{\partial \rho^f \boldsymbol{v}^f}{\partial t} + \boldsymbol{\nabla} \cdot \rho^f \boldsymbol{v}^f \otimes \boldsymbol{v}^f - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}^f = \rho^f \boldsymbol{b}^f \text{ on } \Omega^f \tag{249}$$

$$\frac{\partial \rho^f E^f}{\partial t} + \boldsymbol{\nabla} \cdot \rho^f E^f \boldsymbol{v}^f - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}^f \cdot \boldsymbol{v}^f = \rho^f \boldsymbol{b}^f \cdot \boldsymbol{v}^f \text{ on } \Omega^f \tag{250}$$

where $\partial(\cdot)/\partial t$ is the spatial derivative. $\boldsymbol{v}^f$ consists of the flow velocity components in Cartesian coordinates, ρ^f is fluid mass density, E^f is the fluid total energy density, $\boldsymbol{\sigma}^f = \boldsymbol{\tau}^f - \rho^f \boldsymbol{I}$, is the fluid Cauchy stress, p^f is the fluid pressure, $\boldsymbol{\tau}^f = \lambda^f (\boldsymbol{\nabla} \cdot \boldsymbol{v}^f) \boldsymbol{I} + \mu^f (\boldsymbol{\nabla} \boldsymbol{v}^f + (\boldsymbol{\nabla} \boldsymbol{v}^f)^T)$ is the fluid viscous stress with bulk viscosity λ^f and dynamic viscosity μ^f, $\boldsymbol{b}^f$ is the body force for fluid. The whole system is assumed to be adiabatic, so no external heat flux is supplied. The detailed expression of fluid viscous stress, fluid pression, with total energy density E^f are listed in Chapter 3, Eqns. (53) to (56), respectively.

6.1.3 Fluid-Structure Interaction Problems

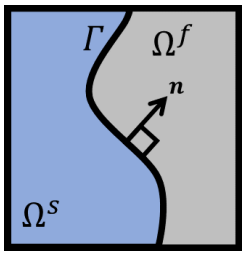


Figure 62. Interface of fluid and solid.

To couple the fluid and solid problem, the fluid-structure interaction condition is given in Figure 62, where the Dirichlet and Neumann jump condition (251) and (252) at interface Γ must hold:

$$\boldsymbol{\sigma}^s \cdot \boldsymbol{n} - \boldsymbol{\sigma}^f \cdot \boldsymbol{n} = [\![\boldsymbol{\sigma} \cdot \boldsymbol{n}]\!] = \mathbf{0}, \qquad \forall \boldsymbol{x} \in \Gamma \tag{251}$$

$$\boldsymbol{v}^s - \boldsymbol{v}^f = [\![\boldsymbol{v}]\!] = \mathbf{0}, \qquad \forall \boldsymbol{x} \in \Gamma \tag{252}$$

where $[\![\cdot]\!]$ is the jump operator. Eqns. (251) and (252) describes the equilibrium and no-slip behavior between the solid subdomain Ω^s and the fluid subdomain Ω^f. From Chapter 6.1.1 to 6.1.3, the strong form of fluid-structure interaction problems is given, and the weak form is derived by the immersed method introduced in Chapter 5.

6.1.4 Immersed Weak Form of Fluid-Structure Interaction Problems

By the immersed approach discussed in Chapter 5, the fluid-structure interaction problem can be viewed as the fluid response in the whole background fluid domains Ω^f, the solid response in the foreground solid domain Ω^s, and then subtract the effect of fluid response in the overlapping area Ω^s. In this way, the strong form of the fluid-structure coupled system can be written as:

$$\rho^s \frac{d\boldsymbol{v}^s}{dt} - \rho^f \frac{d\boldsymbol{v}^f}{dt} - (\nabla \cdot \boldsymbol{\sigma}^s - \nabla \cdot \boldsymbol{\sigma}^f) = \rho^s \boldsymbol{b}^s - \rho^f \boldsymbol{b}^f, \qquad \forall \boldsymbol{x} \in \Omega^s \tag{253}$$

$$\frac{\partial}{\partial t}\begin{pmatrix} \rho^f \\ \rho^f \boldsymbol{v}^f \\ \rho^f E^f \end{pmatrix} + \boldsymbol{\nabla} \cdot \begin{pmatrix} \rho^f \boldsymbol{v}^f \\ \rho^f \boldsymbol{v}^f \otimes \boldsymbol{v}^f \\ \rho^f E^f \boldsymbol{v}^f \end{pmatrix} - \boldsymbol{\nabla} \cdot \begin{pmatrix} 0 \\ \boldsymbol{\sigma}^f \\ \boldsymbol{\sigma}^f \cdot \boldsymbol{v}^f \end{pmatrix} = \begin{pmatrix} 0 \\ \rho^f \boldsymbol{b}^f \\ \rho^f \boldsymbol{b}^f \cdot \boldsymbol{v}^f \end{pmatrix}, \qquad \forall \boldsymbol{x} \in \Omega^f \tag{254}$$

Eq. (253) indicates the subtraction of fluid response from the solid system at the overlapping region. The strong form of the fluid problem (248) to (250) remains unchanged in the background domain as shown in Eq. (254). In the proposed immersed setting, the interfacial constraint (252) is extended into the interior of the solid domain, turning a the volumetric constraint. In this way, the fluid-structure interaction condition are rewritten as follows:

$$[\![\boldsymbol{\sigma} \cdot \boldsymbol{n}]\!] = \boldsymbol{0}, \qquad \forall \boldsymbol{x} \in \Gamma \tag{255}$$

$$[\![\boldsymbol{v}]\!] = \boldsymbol{0}, \qquad \forall \boldsymbol{x} \in \Omega^s \tag{256}$$

To derive the weak form of the immersed FSI problem, the Lagrange multiplier $\boldsymbol{\lambda}$ is introduced to impose the volumetric constraint. The test functions $\boldsymbol{W} \equiv (\boldsymbol{w}_{v^s}, \boldsymbol{w}_{v^f}, w_{\rho^f}, w_{E^f}, \boldsymbol{w}_{\boldsymbol{\lambda}})$ are introduced corresponding to the trail functions $\boldsymbol{U} \equiv (\boldsymbol{v}^s, \boldsymbol{v}^f, \rho^f, E^f, \boldsymbol{\lambda})$ respectively. The trail functions are members of $\boldsymbol{\mathcal{S}} \equiv \boldsymbol{\mathcal{S}}^{v^s} \times \boldsymbol{\mathcal{S}}^{v^f} \times \boldsymbol{\mathcal{S}}^{\rho^f} \times \boldsymbol{\mathcal{S}}^{E^f} \times \boldsymbol{\mathcal{S}}^{\boldsymbol{\lambda}}$ and the test functions are members of $\boldsymbol{\mathcal{V}} \equiv \boldsymbol{\mathcal{V}}^{v^s} \times \boldsymbol{\mathcal{V}}^{v^f} \times \boldsymbol{\mathcal{V}}^{\rho^f} \times \boldsymbol{\mathcal{V}}^{E^f} \times \boldsymbol{\mathcal{V}}^{\boldsymbol{\lambda}}$. $\boldsymbol{\mathcal{S}}$ and $\boldsymbol{\mathcal{V}}$ contains the function satisfying the essential and natural boundary conditions, and then the weak form can be read as: find $(\boldsymbol{v}^s, \boldsymbol{v}^f, \rho^f, E^f, \boldsymbol{\lambda}) \in \boldsymbol{\mathcal{S}}$, such that

$$\begin{aligned} \left(\boldsymbol{w}_{v^s}, \rho^s \frac{d\boldsymbol{v}^s}{dt} - \rho^f \frac{d\boldsymbol{v}^f}{dt}\right)_{\Omega^s} - (\boldsymbol{w}_{v^s}, \nabla \cdot \boldsymbol{\sigma}^s - \nabla \cdot \boldsymbol{\sigma}^f)_{\Omega^s} + (\boldsymbol{w}_{v^s}, \boldsymbol{\lambda})_{\Omega^f} \\ = (\boldsymbol{w}_{v^s}, \rho^s \boldsymbol{b}^s - \rho^f \boldsymbol{b}^f)_{\Omega^s} \end{aligned} \tag{257}$$

$$\left(\begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \frac{\partial}{\partial t}\begin{pmatrix} \rho^f \\ \rho^f \boldsymbol{v}^f \\ \rho^f E^f \end{pmatrix}\right)_{\Omega^f} + \left(\begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \boldsymbol{\nabla} \cdot \begin{pmatrix} \rho^f \boldsymbol{v}^f \\ \rho^f \boldsymbol{v}^f \otimes \boldsymbol{v}^f \\ \rho^f E^f \boldsymbol{v}^f \end{pmatrix}\right)_{\Omega^f}$$
$$- \left(\begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \boldsymbol{\nabla} \cdot \begin{pmatrix} 0 \\ \boldsymbol{\sigma}^f \\ \boldsymbol{\sigma}^f \cdot \boldsymbol{v}^f \end{pmatrix}\right)_{\Omega^f} - \left(\begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \begin{pmatrix} 0 \\ \boldsymbol{\lambda} \\ 0 \end{pmatrix}\right)_{\Omega^s} \tag{258}$$
$$= \left(\begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \begin{pmatrix} 0 \\ \rho^f \boldsymbol{b}^f \\ \rho^f \boldsymbol{b}^f \cdot \boldsymbol{v}^f \end{pmatrix}\right)_{\Omega^f}$$

$$(\boldsymbol{w}_{\boldsymbol{\lambda}}, \boldsymbol{v}^s - \boldsymbol{v}^f)_{\Omega^s} = \boldsymbol{0} \tag{259}$$

$\forall(\boldsymbol{w}_{\boldsymbol{v}^s}, \boldsymbol{w}_{\boldsymbol{v}^f}, w_{\rho^f}, w_{E^f}, \boldsymbol{w}_{\boldsymbol{\lambda}}) \in \boldsymbol{\mathcal{V}}$. $(\cdot,\cdot)_\omega$ denotes the inner product performed on ω. By employing the integration-by-part with the imposition of boundary conditions, the weak form can be re-written as: find $(\boldsymbol{v}^s, \boldsymbol{v}^f, \rho^f, E^f, \boldsymbol{\lambda}) \in \boldsymbol{\mathcal{S}}$, such that

$$\left(\boldsymbol{w}_{\boldsymbol{v}^s}, \rho^s \frac{d\boldsymbol{v}^s}{dt} - \rho^f \frac{d\boldsymbol{v}^f}{dt}\right)_{\Omega^s} + (\nabla \cdot \boldsymbol{w}_{\boldsymbol{v}^s}, \boldsymbol{\sigma}^s - \boldsymbol{\sigma}^f)_{\Omega^s} + (\boldsymbol{w}_{\boldsymbol{v}^s}, \boldsymbol{\lambda})_{\Omega^s}$$
$$= (\boldsymbol{w}_{\boldsymbol{v}^s}, \rho^s \boldsymbol{b}^s - \rho^f \boldsymbol{b}^f)_{\Omega^s} + (\boldsymbol{w}_{\boldsymbol{v}^s}, [\![\boldsymbol{\sigma} \cdot \boldsymbol{n}]\!])_\Gamma \tag{260}$$
$$= (\boldsymbol{w}_{\boldsymbol{v}^s}, \rho^s \boldsymbol{b}^s - \rho^f \boldsymbol{b}^f)_{\Omega^s}$$

$$\left(\begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \frac{\partial}{\partial t}\begin{pmatrix} \rho^f \\ \rho^f \boldsymbol{v}^f \\ \rho^f E^f \end{pmatrix}\right)_{\Omega^f} + \left(\begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \boldsymbol{\nabla} \cdot \begin{pmatrix} \rho^f \boldsymbol{v}^f \\ \rho^f \boldsymbol{v}^f \otimes \boldsymbol{v}^f \\ \rho^f E^f \boldsymbol{v}^f \end{pmatrix}\right)_{\Omega^f}$$
$$+ \left(\boldsymbol{\nabla} \cdot \begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \begin{pmatrix} 0 \\ \boldsymbol{\sigma}^f \\ \boldsymbol{\sigma}^f \cdot \boldsymbol{v}^f \end{pmatrix}\right)_{\Omega^f} - \left(\begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \begin{pmatrix} 0 \\ \boldsymbol{\lambda} \\ 0 \end{pmatrix}\right)_{\Omega^s} \tag{261}$$
$$= \left(\begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \begin{pmatrix} 0 \\ \rho^f \boldsymbol{b}^f \\ \rho^f \boldsymbol{b}^f \cdot \boldsymbol{v}^f \end{pmatrix}\right)_{\Omega^f} + \left(\begin{pmatrix} w_{\rho^f} \\ \boldsymbol{w}_{\boldsymbol{v}^f} \\ w_{E^f} \end{pmatrix}, \begin{pmatrix} 0 \\ \boldsymbol{h} \\ 0 \end{pmatrix}\right)_{\partial\Omega_h^f}$$

$$(\boldsymbol{w}_{\boldsymbol{\lambda}}, \boldsymbol{v}^s - \boldsymbol{v}^f)_{\Omega^s} = \mathbf{0} \tag{262}$$

$\forall \left(\boldsymbol{w}_{\boldsymbol{v}^s}, \boldsymbol{w}_{\boldsymbol{v}^f}, w_{\rho^f}, w_{E^f}, \boldsymbol{w}_{\boldsymbol{\lambda}}\right) \in \boldsymbol{\mathcal{V}}$, where $\boldsymbol{h}$ is the traction applied on the background fluid boundary $\partial\Omega_h^f$.

6.2 The Immersed FSI Formulation

In this section, the immersed formulation is derived to solve the FSI problem shown in Eqns. (260) to (262). The variational multiscale immersed formulation in Chapter 5 is employed as an enhancement for the accuracy and stability. Then, a mixed density approach is introduced such that the inertia effect in solid domain can be eliminated, enabling the fluid equations to be solved with the Lagrange multiplier calculated from the solid domain. The RK approximation with the MUSCL-SCNI formulation is employed to solve the fluid equations. The nodal integration method is applied to the background fluid and foreground solid, making it effective in modeling the structure under large deformation and fragmentation.

6.2.1 Variational Multiscale Immersed Method for Immersed FSI

To solve the immersed fluid-structure weak form, the variational multiscale immersed formulation (VMIM) is employed, where the fluid velocity is decoupled into coarse-scale and fine-scale, i.e. $\boldsymbol{v}^f(\boldsymbol{x}, t) = \overline{\boldsymbol{v}}^f(\boldsymbol{x}, t) + \widehat{\boldsymbol{v}}^f(\boldsymbol{x})$, where $\overline{(\cdot)}$ and $\widehat{(\cdot)}$ denotes the variables in coarse-scale and fine-scale respectively. Note that the fine-scale solution is assumed to be in steady-state [168] such that inertia effect (including advection) vanishes $d\widehat{\boldsymbol{v}}^f/dt = \mathbf{0}$ and $\partial\widehat{\boldsymbol{v}}^f/\partial t = \mathbf{0}$. According

to [168] , this treatment can avoid the numerical instability when the timestep size is very small. Further, the fine-scale solution is assumed to vanish on the domain boundary. Except for the background fluid velocity $\boldsymbol{v}^f$, all other test and trail functions are represented by the coarse scale functions. By employing the multiscale decomposition and using the linear independency of test functions, the FSI problems can be decoupled into coarse-scale equations:

$$\left(\overline{\boldsymbol{w}}_{\boldsymbol{v}^s}, \rho^s \frac{d\overline{\boldsymbol{v}}^s}{dt} - \rho^f \frac{d\overline{\boldsymbol{v}}^f}{dt}\right)_{\Omega^s} + (\nabla \cdot \overline{\boldsymbol{w}}_{\boldsymbol{v}^s}, \boldsymbol{\sigma}^s - \boldsymbol{\sigma}^f)_{\Omega^s} + \left(\overline{\boldsymbol{w}}_{\boldsymbol{v}^s}, \bar{\boldsymbol{\lambda}}\right)_{\Omega^s} = (\overline{\boldsymbol{w}}_{\boldsymbol{v}^s}, \bar{\rho}^s \boldsymbol{b}^s - \bar{\rho}^f \boldsymbol{b}^f)_{\Omega^s} \tag{263}$$

$$\begin{aligned} &\left(\begin{pmatrix} \overline{w}_{\rho^f} \\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f} \\ \overline{w}_{E^f} \end{pmatrix}, \frac{\partial}{\partial t}\begin{pmatrix} \bar{\rho}^f \\ \bar{\rho}^f \overline{\boldsymbol{v}}^f \\ \bar{\rho}^f \bar{E}^f \end{pmatrix}\right)_{\Omega^f} + \left(\begin{pmatrix} \overline{w}_{\rho^f} \\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f} \\ \overline{w}_{E^f} \end{pmatrix}, \boldsymbol{\nabla} \cdot \begin{pmatrix} \bar{\rho}^f \bar{\rho}^f \overline{\boldsymbol{v}}^f \\ \bar{\rho}^f \overline{\boldsymbol{v}}^f \otimes \overline{\boldsymbol{v}}^f \\ \bar{\rho}^f \bar{E}^f \overline{\boldsymbol{v}}^f \end{pmatrix}\right)_{\Omega^f} \\ &+ \left(\boldsymbol{\nabla} \cdot \begin{pmatrix} \overline{w}_{\rho^f} \\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f} \\ \overline{w}_{E^f} \end{pmatrix}, \begin{pmatrix} 0 \\ \boldsymbol{\sigma}^f \\ \boldsymbol{\sigma}^f \cdot \overline{\boldsymbol{v}}^f \end{pmatrix}\right)_{\Omega^f} - \left(\begin{pmatrix} \overline{w}_{\rho^f} \\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f} \\ \overline{w}_{E^f} \end{pmatrix}, \begin{pmatrix} 0 \\ \bar{\boldsymbol{\lambda}} \\ 0 \end{pmatrix}\right)_{\Omega^s} \\ &= \left(\begin{pmatrix} \overline{w}_{\rho^f} \\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f} \\ \overline{w}_{E^f} \end{pmatrix}, \begin{pmatrix} 0 \\ \bar{\rho}^f \boldsymbol{b}^f \\ \bar{\rho}^f \boldsymbol{b}^f \cdot \overline{\boldsymbol{v}}^f \end{pmatrix}\right)_{\Omega^f} + \left(\begin{pmatrix} \overline{w}_{\rho^f} \\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f} \\ \overline{w}_{E^f} \end{pmatrix}, \begin{pmatrix} 0 \\ \boldsymbol{h} \\ 0 \end{pmatrix}\right)_{\partial\Omega_h^f} \end{aligned} \tag{264}$$

$$(\overline{\boldsymbol{w}}_{\boldsymbol{\lambda}}, \overline{\boldsymbol{v}}^s - \overline{\boldsymbol{v}}^f)_{\Omega^s} = 0 \tag{265}$$

and the fine-scale equations on the momentum equation

$$\left(\widehat{\boldsymbol{w}}_{\boldsymbol{v}^f}, \frac{\partial \bar{\rho}^f \overline{\boldsymbol{v}}^f}{\partial t}\right)_{\Omega^f} + \left(\widehat{\boldsymbol{w}}_{\boldsymbol{v}^f}, \boldsymbol{\nabla} \cdot \bar{\rho}^f \overline{\boldsymbol{v}}^f \otimes \overline{\boldsymbol{v}}^f\right)_{\Omega^f} + \left(\boldsymbol{\nabla} \cdot \widehat{\boldsymbol{w}}_{\boldsymbol{v}^f}, \boldsymbol{\sigma}^f\right)_{\Omega^f} - \left(\widehat{\boldsymbol{w}}_{\boldsymbol{v}^f}, \bar{\boldsymbol{\lambda}}\right)_{\Omega^s} = \left(\widehat{\boldsymbol{w}}_{\boldsymbol{v}^f}, \bar{\rho}^f \boldsymbol{b}^f\right)_{\Omega^f} \tag{266}$$

With some algebraic manipulation, the fine-scale equation can be rewritten as

$$\left(\widehat{\boldsymbol{w}}_{\boldsymbol{v}^f}, -\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}^f(\bar{p}^f,\widehat{\boldsymbol{v}}^f)\right)_{\Omega^f}$$
$$= -\left(\widehat{\boldsymbol{w}}_{\boldsymbol{v}^f}, \frac{\partial\bar{\rho}^f\overline{\boldsymbol{v}}^f}{\partial t} + \boldsymbol{\nabla}\cdot\bar{\rho}^f\overline{\boldsymbol{v}}^f\otimes\overline{\boldsymbol{v}}^f + \boldsymbol{\nabla}\bar{p}^f - \boldsymbol{\nabla}\cdot\boldsymbol{\tau}^f(\overline{\boldsymbol{v}}^f)\right. \tag{267}$$
$$\left. - \bar{\rho}^f\boldsymbol{b}^f\right)_{\Omega^f} - \left(\widehat{\boldsymbol{w}}_{\boldsymbol{v}^f}, \bar{\boldsymbol{\lambda}}\right)_{\Omega^s}$$

From Eq. (267), the following approximation for the fine-scale solution is proposed, where the fine-scale solution can be written in terms of the fluid momentum balance residual [161]:

$$\widehat{\boldsymbol{v}}^f = -\hat{\tau}^f\boldsymbol{R}\left(\bar{\rho}^f,\overline{\boldsymbol{v}}^f,\bar{p}^f,\bar{\boldsymbol{\lambda}}\right)$$
$$= -\hat{\tau}^f\left(\frac{\partial\bar{\rho}^f\overline{\boldsymbol{v}}^f}{\partial t} + \boldsymbol{\nabla}\cdot\bar{\rho}^f\overline{\boldsymbol{v}}^f\otimes\overline{\boldsymbol{v}}^f + \boldsymbol{\nabla}\bar{p}^f - \boldsymbol{\nabla}\cdot\boldsymbol{\tau}^f(\overline{\boldsymbol{v}}^f) - \bar{\boldsymbol{\lambda}} - \bar{\rho}^f\boldsymbol{b}^f\right) \tag{268}$$

and the scalar operator $\hat{\tau}^f$ can be defined in Eq. (269), to mimic the effect of divergence operator on the left hand side of Eq. (267):

$$\hat{\tau}^f = c_\tau\frac{h^2}{K^f + 2\mu^f} \tag{269}$$

where K^f is the bulk modulus of the fluid (for compressible ideal gas, the bulk modulus is $K^f = \gamma p^f$, where γ is the ideal gas constant), h is the characteristic nodal distance, and c_τ is the a scalar selected from 0 to 1. In this study $c_\tau = 1/12$ is selected based on analysis shown in previous chapter. Finally, with the fine-scale solution in hand, the final variational multiscale immersed FSI formulation reads: find $\left(\overline{\boldsymbol{v}}^s,\overline{\boldsymbol{v}}^f,\bar{\rho}^f,\bar{E}^f,\bar{\boldsymbol{\lambda}}\right)\in\overline{\boldsymbol{\mathcal{S}}}$,

$$\left(\overline{\boldsymbol{w}}_{\boldsymbol{v}^s},\bar{\rho}^s\frac{d\overline{\boldsymbol{v}}^s}{dt} - \bar{\rho}^f\frac{d\overline{\boldsymbol{v}}^f}{dt}\right)_{\Omega^s} + \left(\nabla\cdot\overline{\boldsymbol{w}}_{\boldsymbol{v}^s},\boldsymbol{\sigma}^s - \boldsymbol{\sigma}^f\right)_{\Omega^s} + \left(\overline{\boldsymbol{w}}_{\boldsymbol{v}^s},\bar{\boldsymbol{\lambda}}\right)_{\Omega^s} \tag{270}$$
$$= \left(\overline{\boldsymbol{w}}_{\boldsymbol{v}^s},\bar{\rho}^s\boldsymbol{b}^s - \bar{\rho}^f\boldsymbol{b}^f\right)_{\Omega^s}$$

$$
\begin{aligned}
&\left(\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \frac{\partial}{\partial t}\begin{pmatrix}\bar{\rho}^f\\ \bar{\rho}^f\overline{\boldsymbol{v}}^f\\ \bar{\rho}^f\bar{E}^f\end{pmatrix}\right)_{\Omega^f} + \left(\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \boldsymbol{\nabla}\cdot\begin{pmatrix}\bar{\rho}^f\bar{\rho}^f\overline{\boldsymbol{v}}^f\\ \bar{\rho}^f\overline{\boldsymbol{v}}^f\otimes\overline{\boldsymbol{v}}^f\\ \bar{\rho}^f\bar{E}^f\overline{\boldsymbol{v}}^f\end{pmatrix}\right)_{\Omega^f}\\
&\quad+\left(\boldsymbol{\nabla}\cdot\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \begin{pmatrix}0\\ \boldsymbol{\sigma}^f(\overline{\boldsymbol{v}}^f)\\ \boldsymbol{\sigma}^f\cdot\overline{\boldsymbol{v}}^f\end{pmatrix}\right)_{\Omega^f}\\
&\quad-\left(\begin{pmatrix}0\\ \boldsymbol{\nabla}\cdot\boldsymbol{\sigma}^f(\overline{\boldsymbol{w}}_{\boldsymbol{v}^f})\\ 0\end{pmatrix}, -\hat{\tau}^f\begin{pmatrix}0\\ \boldsymbol{R}(\rho^f,\overline{\boldsymbol{v}}^f,\bar{p}^f,\bar{\boldsymbol{\lambda}})\\ 0\end{pmatrix}\right)_{\Omega^f}\\
&\quad-\left(\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \begin{pmatrix}0\\ \bar{\boldsymbol{\lambda}}\\ 0\end{pmatrix}\right)_{\Omega^s}\\
&\quad=\left(\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \begin{pmatrix}0\\ \bar{\rho}^f\boldsymbol{b}^f\\ \bar{\rho}^f\boldsymbol{b}^f\cdot\overline{\boldsymbol{v}}^f\end{pmatrix}\right)_{\Omega^f} + \left(\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \begin{pmatrix}0\\ \boldsymbol{h}\\ 0\end{pmatrix}\right)_{\partial\Omega_h^f}
\end{aligned}
\tag{271}
$$

$$
(\overline{\boldsymbol{w}}_{\boldsymbol{\lambda}}, \overline{\boldsymbol{v}}^s - \overline{\boldsymbol{v}}^f - \widehat{\boldsymbol{v}}^f)_{\Omega^s} = 0 \tag{272}
$$

$\forall\left(\overline{\boldsymbol{w}}_{\boldsymbol{v}^s}, \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}, \overline{w}_{\rho^f}, \overline{w}_{E^f}, \overline{\boldsymbol{w}}_{\boldsymbol{\lambda}}\right) \in \overline{\boldsymbol{\mathcal{V}}}$. $\overline{\boldsymbol{\mathcal{S}}} \subset \boldsymbol{\mathcal{S}}$ and $\overline{\boldsymbol{\mathcal{V}}} \subset \boldsymbol{\mathcal{V}}$ are collections of trial and test function spaces of the coarse-scale variables, respectively. From Eq. (271), it can be seen the derived formulation leads to a stabilized formulation with the numerical error represented by $\widehat{\boldsymbol{v}}^f$ using the residual of the momentum balance equations, and then substitute this information into the coarse-scale equations.

6.2.2 Mixed Density Field Approach

To solve the coupled equations (270) to (272), a mixed density approach [169] is employed. Because the fluid in the overlapping domain is fictitious, it can be assumed to have the same density as the solid without losing generality. A mixed density field $\tilde{\rho}(\boldsymbol{x})$ is defined by Eq. (273):

$$\tilde{\rho}(\boldsymbol{x}) = \bar{\rho}^f + (\bar{\rho}^s - \bar{\rho}^f)H(\boldsymbol{x}) \tag{273}$$

This is used to replace the original fluid density field, where $H(\boldsymbol{x})$ is a Heaviside function indicating the position of the solid, defined as

$$H(\boldsymbol{x}) = \begin{cases} 1, \ \forall \boldsymbol{x} \in \Omega^s \\ 0, \ \text{otherwise} \end{cases} \tag{274}$$

In addition, the body force (i.e. gravity) applied are generally identical in the solid and fluid, such that $\boldsymbol{b}^s(\boldsymbol{x}) = \boldsymbol{b}^f(\boldsymbol{x}), \ \forall \boldsymbol{x} \in \Omega^s$. In this way, the inertia terms and body force terms in Eq. (270) naturally vanishes and therefore Eq. (270) can be rewritten as

$$(\nabla \cdot \bar{\boldsymbol{w}}_{\boldsymbol{v}^s}, \boldsymbol{\sigma}^s - \boldsymbol{\sigma}^f)_{\Omega^s} = -\left(\bar{\boldsymbol{w}}_{\boldsymbol{v}^s}, \bar{\boldsymbol{\lambda}}\right)_{\Omega^s} \tag{275}$$

The corresponding fluid equations in Eq. (271) can be also be rewritten as

$$
\begin{aligned}
&\left(\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \frac{\partial}{\partial t}\begin{pmatrix}\tilde{\rho}\\ \tilde{\rho}\overline{\boldsymbol{v}}^f\\ \tilde{\rho}\bar{E}^f\end{pmatrix}\right)_{\Omega^f} + \left(\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \boldsymbol{\nabla}\cdot\begin{pmatrix}\tilde{\rho}\overline{\boldsymbol{v}}^f\\ \tilde{\rho}\overline{\boldsymbol{v}}^f\otimes\overline{\boldsymbol{v}}^f\\ \tilde{\rho}\bar{E}^f\overline{\boldsymbol{v}}^f\end{pmatrix}\right)_{\Omega^f}\\
&\quad + \left(\boldsymbol{\nabla}\cdot\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \begin{pmatrix}0\\ \boldsymbol{\sigma}^f(\overline{\boldsymbol{v}}^f)\\ \boldsymbol{\sigma}^f\cdot\overline{\boldsymbol{v}}^f\end{pmatrix}\right)_{\Omega^f}\\
&\quad - \left(\begin{pmatrix}0\\ \boldsymbol{\nabla}\cdot\boldsymbol{\sigma}^f(\overline{\boldsymbol{w}}_{\boldsymbol{v}^f})\\ 0\end{pmatrix}, -\hat{\tau}^f\begin{pmatrix}0\\ \boldsymbol{R}(\tilde{\rho},\overline{\boldsymbol{v}}^f,\bar{p}^f,\bar{\boldsymbol{\lambda}})\\ 0\end{pmatrix}\right)_{\Omega^f}\\
&\quad - \left(\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \begin{pmatrix}0\\ \bar{\boldsymbol{\lambda}}\\ 0\end{pmatrix}\right)_{\Omega^s}\\
&\quad = \left(\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \begin{pmatrix}0\\ \tilde{\rho}\boldsymbol{b}^f\\ \tilde{\rho}\boldsymbol{b}^f\cdot\overline{\boldsymbol{v}}^f\end{pmatrix}\right)_{\Omega^f} + \left(\begin{pmatrix}\overline{w}_{\rho^f}\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}\\ \overline{w}_{E^f}\end{pmatrix}, \begin{pmatrix}0\\ \boldsymbol{h}\\ 0\end{pmatrix}\right)_{\partial\Omega_h^f}
\end{aligned}
\tag{276}
$$

Eq. (276) is subjected to two algebraic equations (272) and (275). The Galerkin equation of (272), (275) and (276) can be formulated by employed the RK approximation for test and trial functions, reading: find $\left(\overline{\boldsymbol{v}}^{s^h}, \overline{\boldsymbol{v}}^{f^h}, \bar{\rho}^{f^h}, \bar{E}^{f^h}, \bar{\boldsymbol{\lambda}}^h\right) \in \overline{\boldsymbol{S}}^h \subset \overline{\boldsymbol{S}}$, such that

$$
\begin{aligned}
&\left(\begin{pmatrix}\overline{w}_{\rho^f}^h\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}^h\\ \overline{w}_{E^f}^h\end{pmatrix},\frac{\partial}{\partial t}\begin{pmatrix}\tilde{\rho}^h\\ \tilde{\rho}^h\overline{\boldsymbol{v}}^{f\,h}\\ \tilde{\rho}^h\overline{E}^{f\,h}\end{pmatrix}\right)_{\Omega^f}+\left(\begin{pmatrix}\overline{w}_{\rho^f}^h\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}^h\\ \overline{w}_{E^f}^h\end{pmatrix},\boldsymbol{\nabla}\cdot\begin{pmatrix}\tilde{\rho}^h\overline{\boldsymbol{v}}^{f\,h}\\ \tilde{\rho}^h\overline{\boldsymbol{v}}^{f\,h}\otimes\overline{\boldsymbol{v}}^{f\,h}\\ \tilde{\rho}^h\overline{E}^f\overline{\boldsymbol{v}}^f\end{pmatrix}\right)_{\Omega^f}\\
&\quad+\left(\boldsymbol{\nabla}\cdot\begin{pmatrix}\overline{w}_{\rho^f}^h\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}^h\\ \overline{w}_{E^f}^h\end{pmatrix},\begin{pmatrix}0\\ \boldsymbol{\sigma}^f\\ \boldsymbol{\sigma}^f\cdot\overline{\boldsymbol{v}}^{f\,h}\end{pmatrix}\right)_{\Omega^f}\\
&\quad=\left(\begin{pmatrix}\overline{w}_{\rho^f}^h\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}^h\\ \overline{w}_{E^f}^h\end{pmatrix},\begin{pmatrix}0\\ \tilde{\rho}^h\boldsymbol{b}^f\\ \tilde{\rho}^h\boldsymbol{b}^f\cdot\overline{\boldsymbol{v}}^f\end{pmatrix}\right)_{\Omega^f}+\left(\begin{pmatrix}\overline{w}_{\rho^f}^h\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}^h\\ \overline{w}_{E^f}^h\end{pmatrix},\begin{pmatrix}0\\ \boldsymbol{h}\\ 0\end{pmatrix}\right)_{\partial\Omega_h^f}\\
&\quad+\left(\begin{pmatrix}\overline{w}_{\rho^f}^h\\ \overline{\boldsymbol{w}}_{\boldsymbol{v}^f}^h\\ \overline{w}_{E^f}^h\end{pmatrix},\begin{pmatrix}0\\ \bar{\boldsymbol{\lambda}}^h\\ 0\end{pmatrix}\right)_{\Omega^s}\\
&\quad+\left(\begin{pmatrix}0\\ \boldsymbol{\nabla}\cdot\boldsymbol{\sigma}^f\left(\overline{\boldsymbol{w}}_{\boldsymbol{v}^f}^h\right)\\ 0\end{pmatrix},-\hat{\tau}^f\begin{pmatrix}0\\ \boldsymbol{R}\left(\tilde{\rho}^h,\overline{\boldsymbol{v}}^{f\,h},\bar{p}^{f\,h},\bar{\boldsymbol{\lambda}}^h\right)\\ 0\end{pmatrix}\right)_{\Omega^f}
\end{aligned}
\tag{277}
$$

$$
\left(\overline{\boldsymbol{w}}_{\boldsymbol{v}^s}^h,\bar{\boldsymbol{\lambda}}^h\right)_{\Omega^s}=-\left(\nabla\cdot\overline{\boldsymbol{w}}_{\boldsymbol{v}^s}^h,\boldsymbol{\sigma}^s-\boldsymbol{\sigma}^f\right)_{\Omega^s}
\tag{278}
$$

$$
\left(\overline{\boldsymbol{w}}_{\boldsymbol{\lambda}}^h,\overline{\boldsymbol{v}}^{s\,h}\right)_{\Omega^s}=\left(\overline{\boldsymbol{w}}_{\boldsymbol{\lambda}}^h,\overline{\boldsymbol{v}}^{f\,h}-\hat{\tau}^f\boldsymbol{R}\left(\tilde{\rho}^h,\overline{\boldsymbol{v}}^{f\,h},\bar{p}^{f\,h},\bar{\boldsymbol{\lambda}}^h\right)\right)_{\Omega^s}
\tag{279}
$$

$\forall\left(\overline{\boldsymbol{w}}_{\boldsymbol{v}^s}^h,\overline{\boldsymbol{w}}_{\boldsymbol{v}^f}^h,\overline{w}_{\rho^f}^h,\overline{w}_{E^f}^h,\overline{\boldsymbol{w}}_{\boldsymbol{\lambda}}^h\right)\in\overline{\boldsymbol{\mathcal{V}}}^h\subset\overline{\boldsymbol{\mathcal{V}}}$. The superscript $(\cdot)^h$ indicates the variables obtained by the proper RK approximation. The fine-scale related terms and Lagrange multiplier terms are regarded as unbalanced force on the right hand side of the Eq. (277). The mixed density $\tilde{\rho}^h$ is approximated by $\tilde{\rho}^h=\bar{\rho}^{f\,h}+\left(\bar{\rho}^{s\,h}-\bar{\rho}^{f\,h}\right)H(\boldsymbol{x})$. In this study, the Heaviside function $H(\boldsymbol{x})$ can be obtained by solving the following equations:

$$\int_{\Omega^{\mathrm{f}}} H(\boldsymbol{x})d\Omega = \int_{\Omega^{\mathrm{s}}} 1d\Omega \tag{280}$$

with $H(\boldsymbol{x})$ approximated by the background fluid RK shape function Ψ_I^f with the generalized coefficient H_I, as shown in Eq. (281):

$$H(\boldsymbol{x}) \approx H^h(\boldsymbol{x}) = \sum_{I,\boldsymbol{x}_I \in \Omega^f} \Psi_I^f(\boldsymbol{x}) H_I \tag{281}$$

Then the Heaviside function can be solved by the following equation:

$$\sum_J \int_{\Omega^{\mathrm{f}}} {\Psi_I^f}^T(\boldsymbol{x}) \Psi_J^f(\boldsymbol{x}) d\Omega \, H_J = \int_{\Omega^{\mathrm{s}}} \Psi_I^f(\boldsymbol{x}) d\Omega, \qquad \forall I \tag{282}$$

6.2.3 Numerical Procedure

An outline of the numerical algorithm for proposed immersed formulation within an explicit time integration solver is illustrated as follows:

(1). Given the structural configuration Ω^s and all variables $\overline{\boldsymbol{v}}^{s^h}_n, \overline{\boldsymbol{v}}^{f^h}_n, \bar{\rho}^{f^h}_n, \bar{E}^{f^h}_n$ at time step n.

(2). Obtain the current solid density $\bar{\rho}^{s^h}_n$ by (246). Obtain fluid pressure $\bar{p}^{f^h}_n$ by given equation of state.

(3). Calculate the Heaviside function $H(\boldsymbol{x})$ by Eq. (280) to (282), and compute the mixed density $\tilde{\rho}^h_n = \bar{\rho}^{f^h}_n + \left(\bar{\rho}^{s^h}_n - \bar{\rho}^{f^h}_n\right) H(\boldsymbol{x})$.

(4). Evaluate the Lagrange multiplier $\bar{\boldsymbol{\lambda}}^h_n$ by Eq. (278).

(5). Calculate the residual $\boldsymbol{R}\left(\tilde{\rho}^h_n, \overline{\boldsymbol{v}}^{f^h}_n, \bar{p}^{f^h}_n, \bar{\boldsymbol{\lambda}}^h_n\right)$.

(6). Apply the mixed density $\tilde{\rho}_n^h$, Lagrange multiplier $\bar{\boldsymbol{\lambda}}_n^h$, and momentum equation residual $\boldsymbol{R}\left(\tilde{\rho}_n^h, \overline{\boldsymbol{v}}^{f\,h}_{\ n}, \bar{p}^{f\,h}_{\ n}\right)$ onto the fluid equations (277), and solve the fluid variables $\overline{\boldsymbol{v}}^{f\,h}_{\ n+1}, \bar{\rho}^{f\,h}_{\ n+1}, \bar{E}^{f\,h}_{\ n+1}$ at time step $n+1$ in Eq. (277) by using the MUSCL-SCNI formulation (with central different method for time integration) shown in Chapter 4. Obtain the fluid pressure $\bar{p}^{f\,h}_{\ n+1}$ by the given equation of state.

(7). Compute the solid velocity $\overline{\boldsymbol{v}}^{s\,h}_{\ n+1}$ by Eq. (279).

(8). Update the positions of the solid domain using the velocity from step (7), then return to step 1 for the next time step.

6.3 Numerical Examples

In this study, benchmark numerical examples are analyzed to examine the performance of the proposed immersed FSI with respected to the structure under various shock wave impact condition. The reproducing kernel approximation with a linear basis and cubic B-spline kernel are adopted for the approximation of all variables. The normalized support size equaling 2.0 is employed for both fluid and solid. Eulerian RKPM with MUSCL-SCNI formulation is employed for background fluid and Lagrangian RKPM formulation is employed for the solid domain.

6.3.1 Free Fall Test

In the first example, a 2-D solid plate dropping in free fall in the air, is tested. As shown in Figure 63, the fluid domain is a rectangular domain with 2 cm in the horizontal direction and 4 cm in the vertical direction, with no-slip steady boundary conditions for all walls. The solid domain is a circular plate initially placed at 3 cm height from the bottom edge. The air is simulated with a

density of 0.001 g/cm^3, and initially at rest with the pressure of 1.0 dyn/cm^2. The solid plate is modeled by an elastic solid with the Young's modulus of 10,000 dyn/cm^2 and the Poisson ratio of 0.0. Plane strain condition is employed for this case. The gravity is set to be 980 cm/s^2, and only works on the solid domain. The fluid domain is discretized into 41 by 81 nodes, and the solid is discretized into 414 nodes.

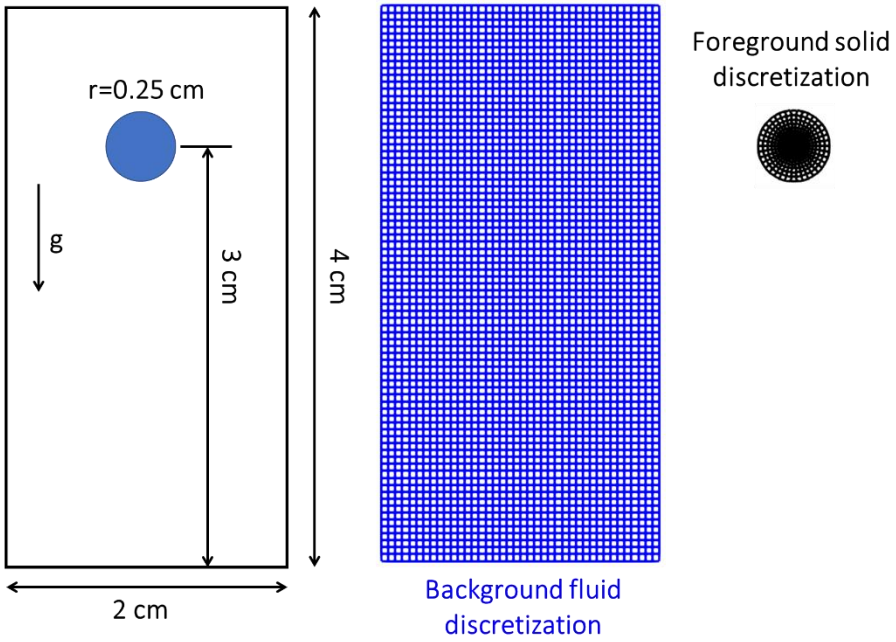


Figure 63. Solid plate free fall problem: geometry, setting, and discretization.

The snapshots of the simulation can be found in Figure 64, where it can be seen small vortices near the ball are clearly captured as it falls, and the fluid velocity field is increased based on the falling of the solid. Also, the boundary effect is limited in this case.

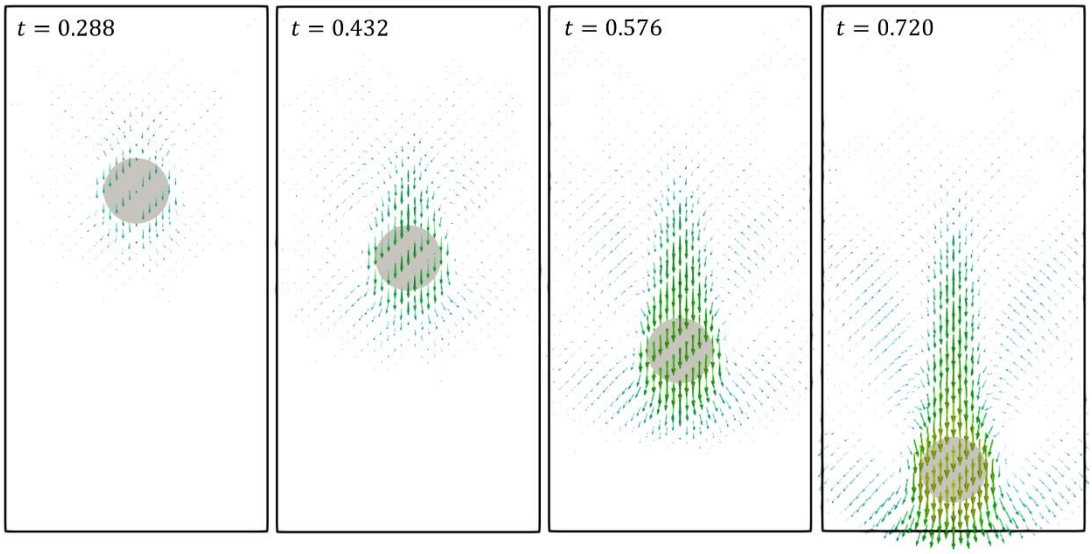


Figure 64. Solid disk dropping in a free fall in a channel at different time steps.

The comparison of the time history of solid position and velocity with the theoretical prediction of the rigid body falling in an open-spaced channel is shown in Figure 65 , where it can be seen that the simulation results agree with the theoretical prediction. Although the physics of the problem is not complicated, the purpose of this study is to demonstrate the capability of the developed immersed formulation, which can handle large displacement without any re-meshing or contour integral along with a fluid-solid interface when the fast-moving solid body.

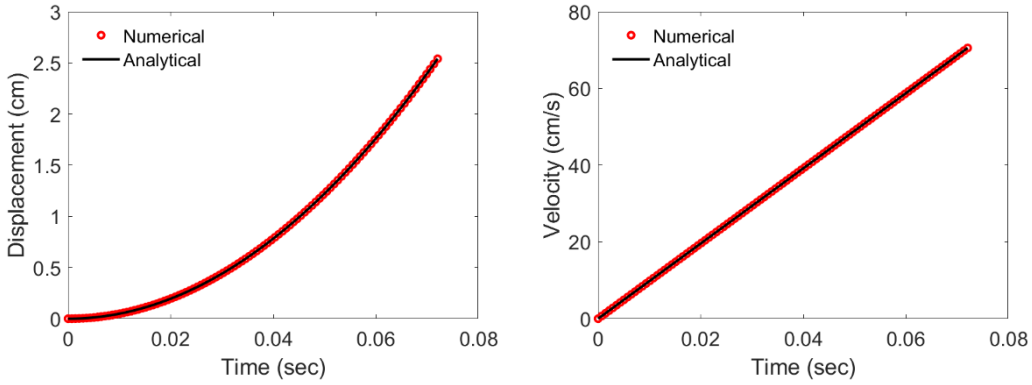


Figure 65. Comparison of numerical and analytical results in free fall problem.

6.3.2 Schardin's Problem

Schardin's problem [149] is tested here to verify the accuracy of the immersed formulation. All numerical setting for the problems is identical to those shown in Chapter 4.3.3. In the proposed immersed formulation simulation, the background fluid domain is discretized uniformly with nodal distance of $h = 4 \times 10^{-4}$ m, and the foreground solid domain is discretized into 7,651 nodes with an average nodal distance of $h = 4 \times 10^{-4}$ m.

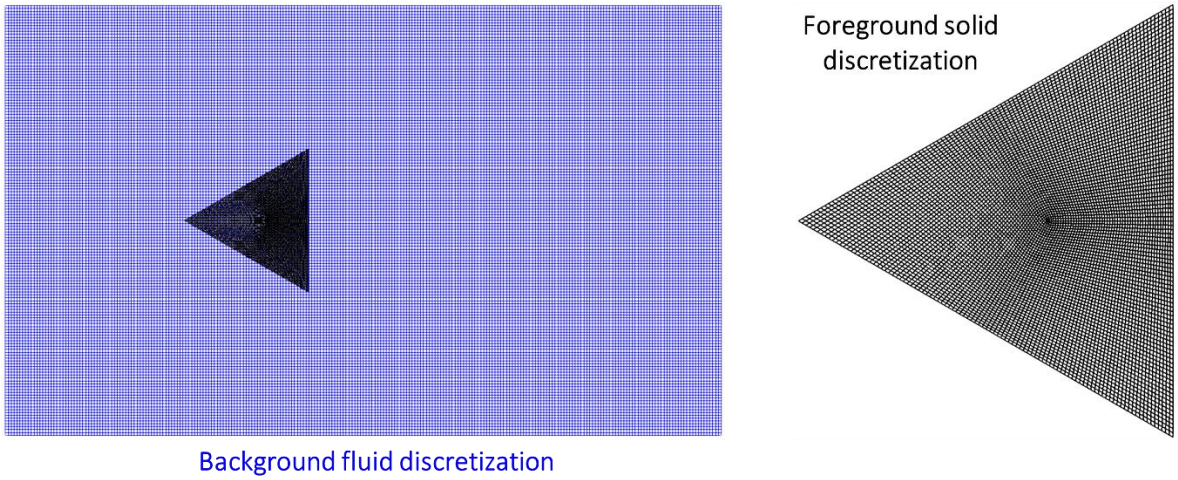
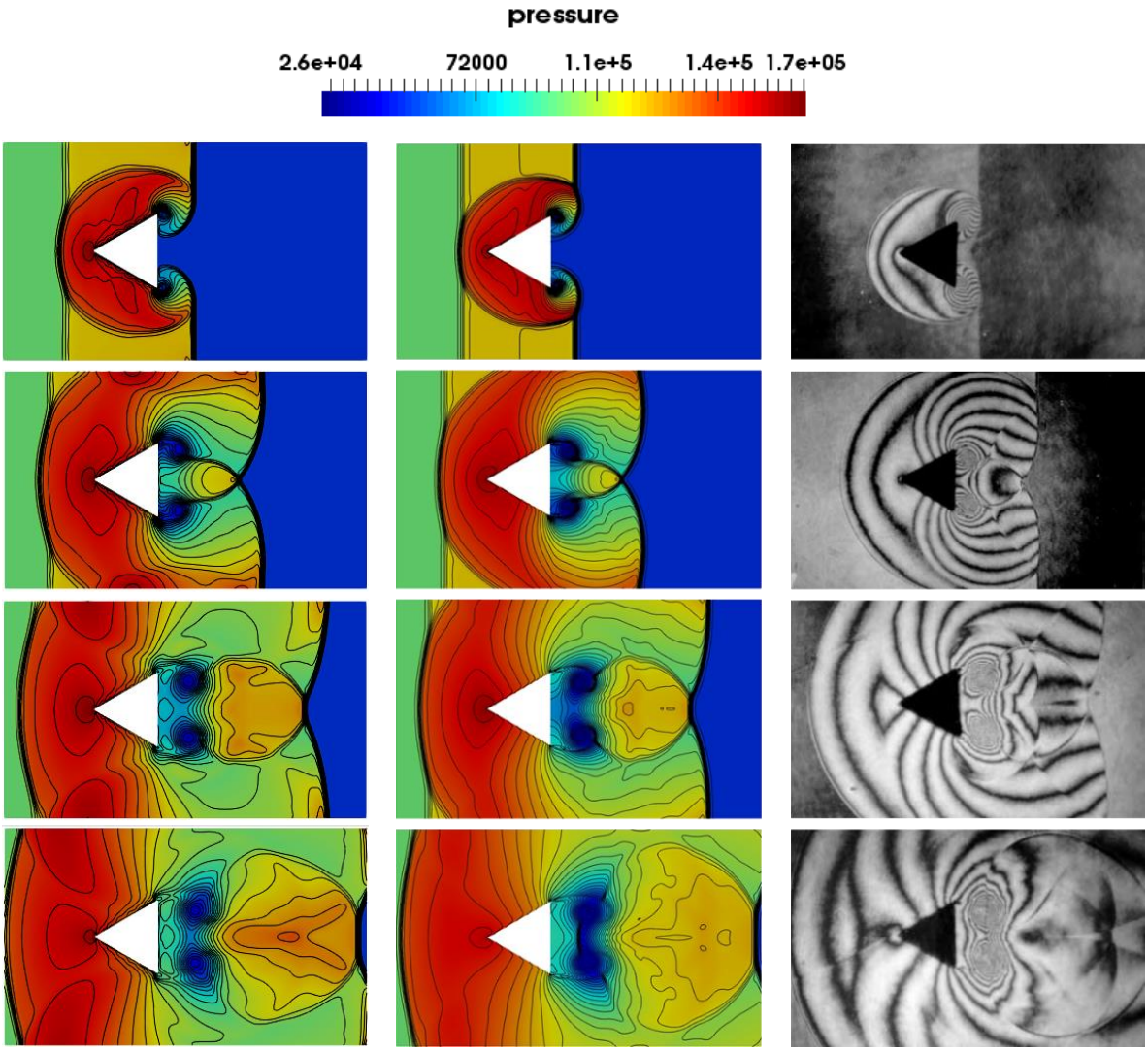


Figure 66. Schardin's problem: discretization of the foreground solid domain on top of background fluid domain.

The foreground solid domain is modeled as a rigid body with very large mass such that it is nearly immovable. The experimental results from [150] and the numerical results by the MUSCL-SCNI RKPM shown in Chapter 4.3.3, are employed as the reference solution. From the numerical results shown in Figure 67, the proposed immersed simulation demonstrates consistent results compared to the reference numerical solution. Both numerical results are in good agreement with the experimental observation [150]. As can be seen, the shock wave is diffracted at the vertices of the triangular wedge at $t = 53\ \mu s$. Then, vortices are generated near back vertices of the wedge at $t = 102\ \mu s$ and $130\ \mu s$. At $t = 172\ \mu s$, the discrete vortex-lets formed along the slip layer

of the primary vortex are observed. The developed immersed formulation effectively captures these complex physical processes of shock diffraction, vortex generation, and shock-vortex interaction, and the simulation results agree quite well with the experimental observation



(a) P (b) (c)

Figure 67. The pressure field for Schardin's problem by (a) proposed immersed simulation, (b) MUSCL-SCNI RKPM [131] with conforming discretization and (c) experimental results [150]. Top-down: results at $53\ \mu s$, $102\ \mu s$, $130\ \mu s$, and $172\ \mu s$.

6.3.3 Shock Traveling Down a Deformable Tube

The problem in [170] is tested to verify the capability of the proposed FSI algorithm on modeling a shock wave travels through a deformable two-dimensional tube with large deformation on the tube walls. As shown in Figure 68, the numerical set-up of the problem consists of a two-dimensional channel filled with air, and the tube wall are modeled by an elastic solid. The initial conditions of the air in pre-shock and post-shock conditions are also given in Figure 68. The solid is modeled by a linear elasticity model with a density of 3.1538 kg/m^3, Young's modulus of 10 MPa, and Poisson ratio of 0.3. The symmetry boundary condition is used so only the bottom half domain is employed for the simulation as shown in Figure 69. The background fluid domain covers the channel as well as the deformable body. The regular discretization is employed for both fluid and solid domain with equal nodal spacing 0.005 m and 0.01 m, respectively. The simulation time is set to be 0.6 msec, with the timestep size of 0.05 μsec.

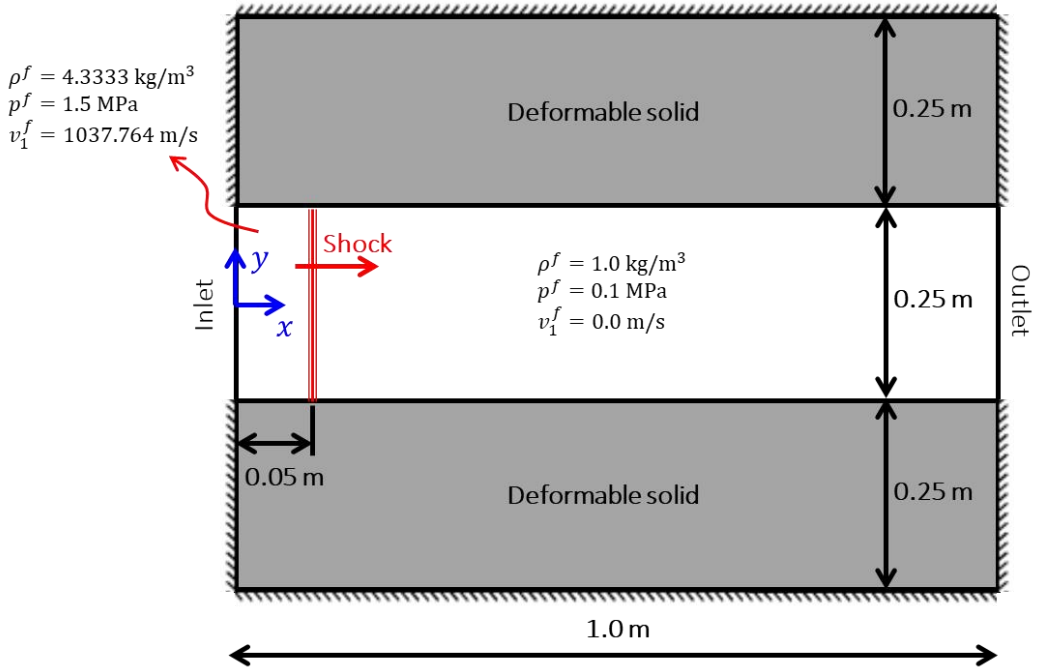


Figure 68. Shock traveling down a deformable tube: geometry and setup.

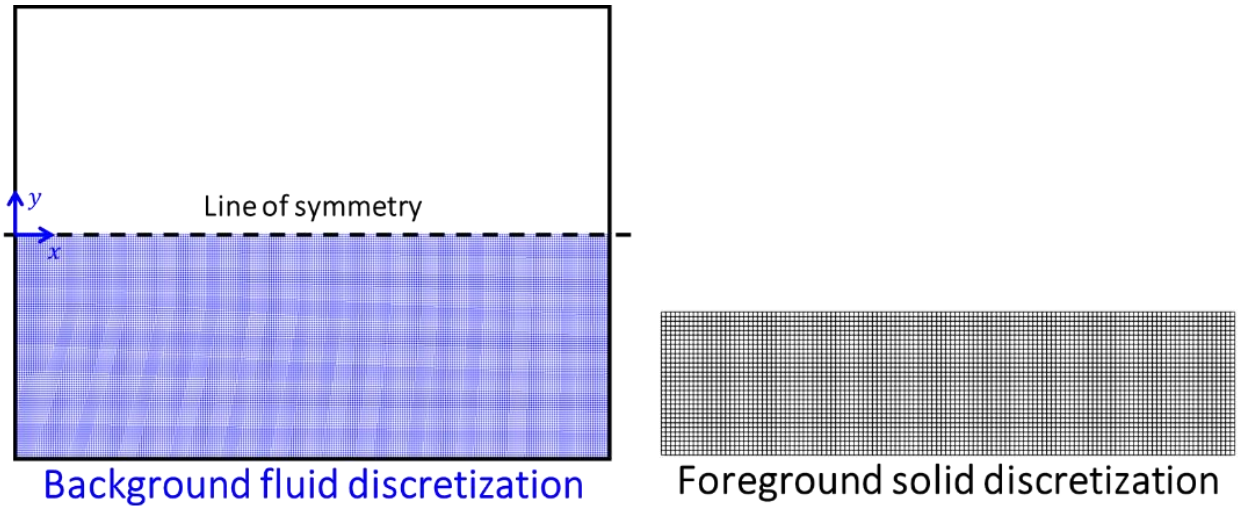


Figure 69. Shock traveling down a deformable tube: discretization.

From Figure 70, it can be seen the tube wall is compressed as the shock wave pass through. The planar shock front also becomes a curved shock front at the later timestep of simulation due to the fluid-structure interaction, which is consistent with the numerical observation shown in [170]. From Figure 71, the tube wall geometry is compared with the ALE simulation in [170], and it can be found the deformation history of the tube wall by the proposed immersed method is consistent with that reference solution.

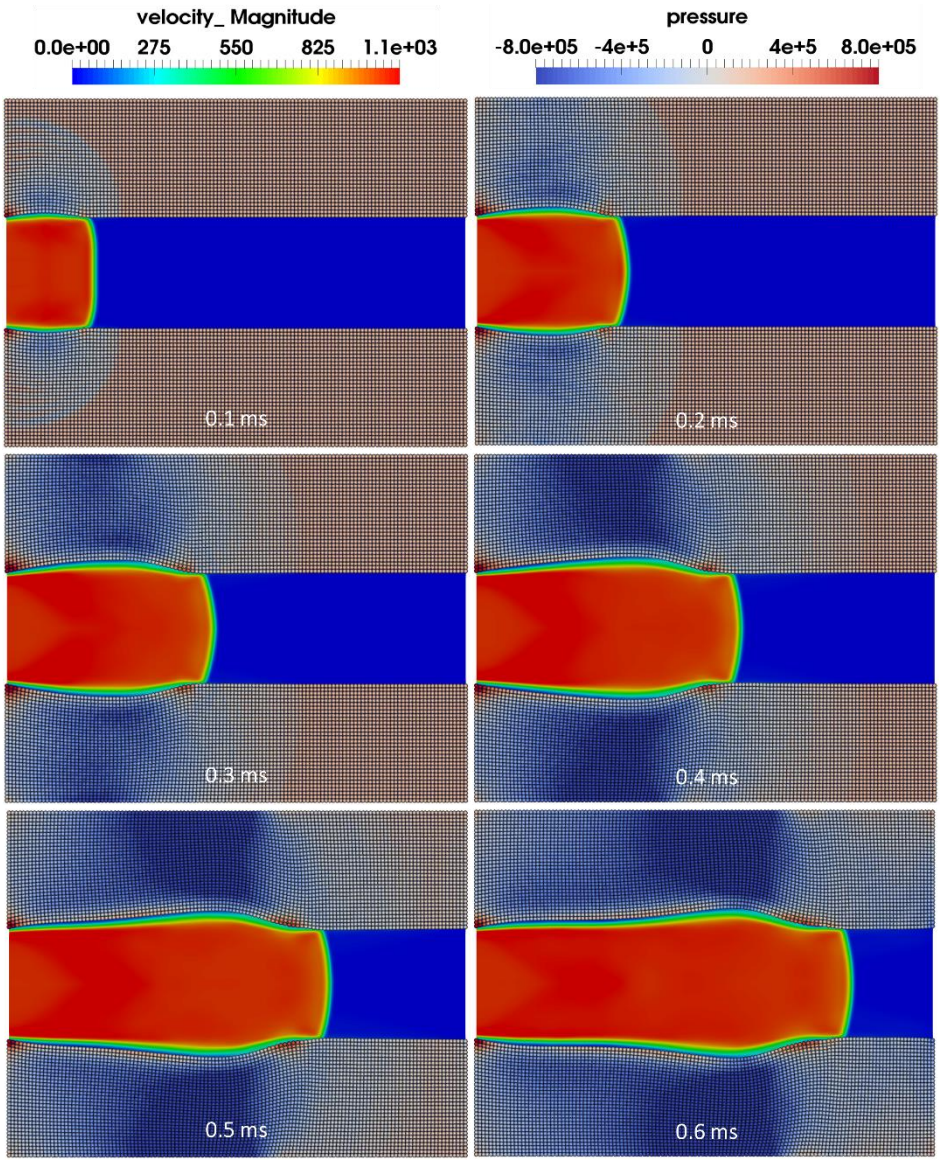


Figure 70. Time history of tube geometry during the shock wave pass through the tube. The velocity field is plotted for the fluid domain and the pressure field is plotted for the solid domain.

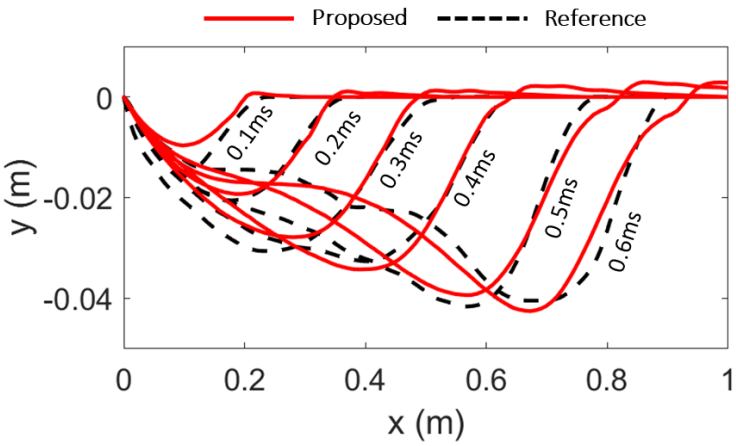


Figure 71. The tube wall geometry at different timestep of simulation by the proposed immersed framework. The reference solution is from the ALE simulation shown in [170].

6.3.4 Cylinder Lift-Off Problem

The cylinder lift-off problem, suggested by [171, 172], is employed to test the capability of FSI computation on capturing the light mass solid under shock loading. As shown in Figure 72, the numerical set-up of the problem consists of a two-dimensional channel filled with air, and a rigid light-weight cylinder is initially resting on the lower wall. The cylinder is driven and lifted upward by a shock wave from the left with a Mach number of 3.0. The initial conditions of the air in pre-shock and post-shock conditions are also given in Figure 72. The solid is modeled by a linear elasticity model with a density of 7.6 kg/m^3, Young's modulus of 10,000 Pa, and Poisson ratio of 0.0 to mimic the rigid body motion. The fluid domain is discretized into 401 by 81 nodes (equal nodal spacing 0.0025 m in both directions). The solid domain is discretized into 2,309 nodes with averaging nodal spacing 0.0020 m. The simulation time is set to be 0.25 seconds, with the timestep size of 0.025 msec.

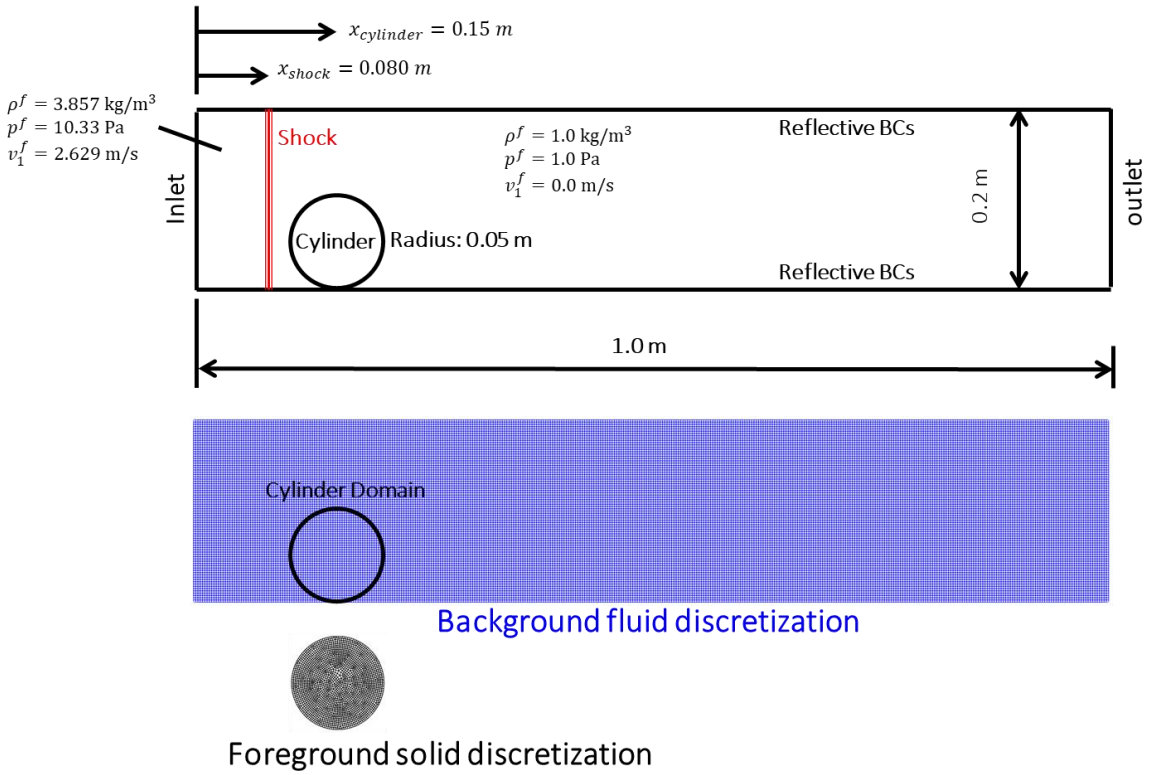


Figure 72. Chamber detonation problem: geometry, setup and discretization.

From Figure 74, the cylinder successfully "take off" from the lower wall. The reflective shock waves between the upper and lower walls are captured. From Figure 74, the position of the cylinder is similar to that from the reference solution at 0.14 seconds. However, the numerical solution at the final time (0.25 second) deviates from the reference solution. The possible reason could be that part of the kinetic energy of the cylinder becomes the internal energy, and therefore the cylinder cannot be lift-off high enough. More studies on the exact reason of accuracy degeneration at later timestep will be investigated in the future. Although the results do not fully agree with the reference solution, it still shows the capability of the immersed formulation to model solid response under shock wave impact.

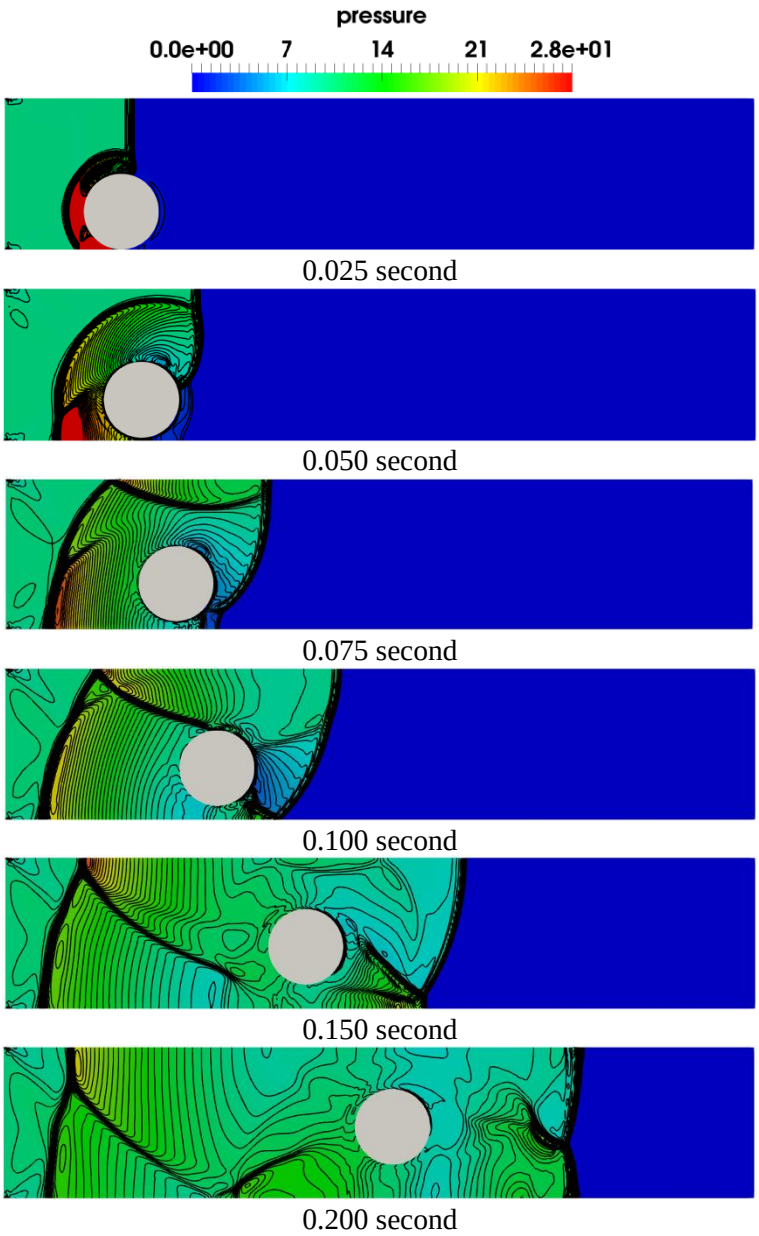


Figure 73. Cylinder position under the shock wave at different timestep.

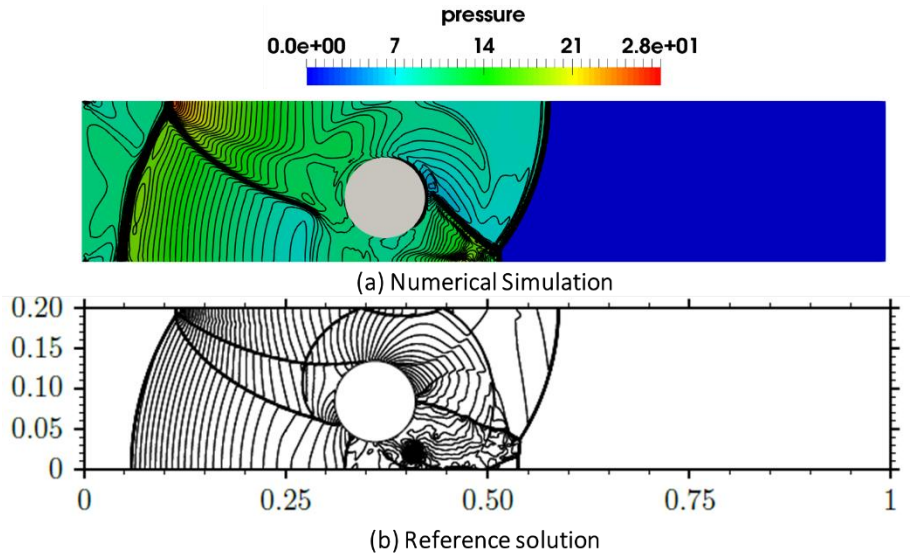


Figure 74. Cylinder position: comparison of numerical results with reference result [172], at time 0.14 second. The dark curves in the figures are pressure contour ranging from 0 to 28 Pa.

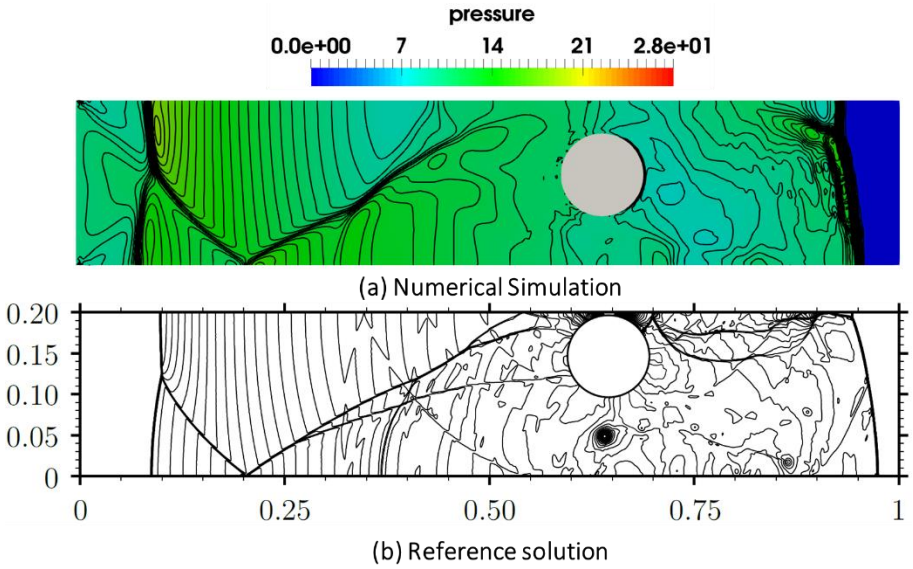


Figure 75. Cylinder position: comparison of numerical results with reference result [172] at time 0.25 second. The dark curves in the figures are pressure contour ranging from 0 to 28 Pa.

6.3.5 Chamber Detonation Problem

In this shock wave induced FSI example, a steel plate subjected to a detonation blast load [43] is modeled, where the problem setting is shown in Figure 76, where Ω_{det} is the region contains the detonation gas.

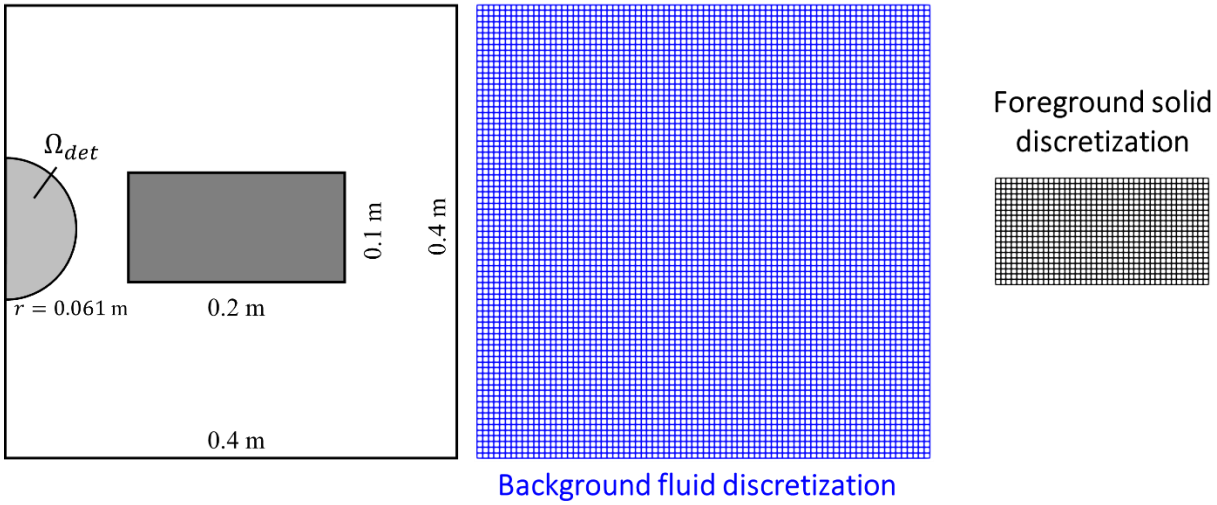


Figure 76. Chamber detonation problem: geometry, setup and discretization.

A two-dimensional steel plate with dimensions 0.2 m × 0.1 m is placed at the center of a closed chamber with dimensions 0.4 m × 0.4 m, and the plate is initially at rest. The steel plate is modeled by J2 plasticity with Young's modulus 200 GPa, density 7,870 kg/m^3, Poisson ratio 0.3, yield stress 0.4 GPa and plastic modulus 0.1 GPa. The initial condition of the air and detonation region are listed in Eq. (283):

$$\left(v_1^f, v_2^f, \rho^f, p^f\right) = \begin{cases} \left(0\ \text{m/s}, 0\ \text{m/s}, 16.042\ \text{kg/m}^3, 6.746\ \text{MPa}\right) & , \forall \boldsymbol{x} \in \Omega_{det} \\ \left(0\ \text{m/s}, 0\ \text{m/s},\ 1.201\ \text{kg/m}^3,\ 0.100\ \text{MPa}\right) & \text{otherwise} \end{cases} \tag{283}$$

Reflective boundary conditions are assumed at all chamber walls, same as [43]. Both foreground solid and background fluid domains are discretized uniformly with three nodal spacing

$h = 1.0 \times 10^{-2},\ 5.0 \times 10^{-3},\ 2.5 \times 10^{-3}$ m. The numerical results shown in the literature [43] is employed as reference solutions. From the numerical results shown in Figure 77, the steel plate under shock wave impact forms a mushroom-like geometry, and the repeated shock reflection and pressure concentration are consistent under three different discretization.

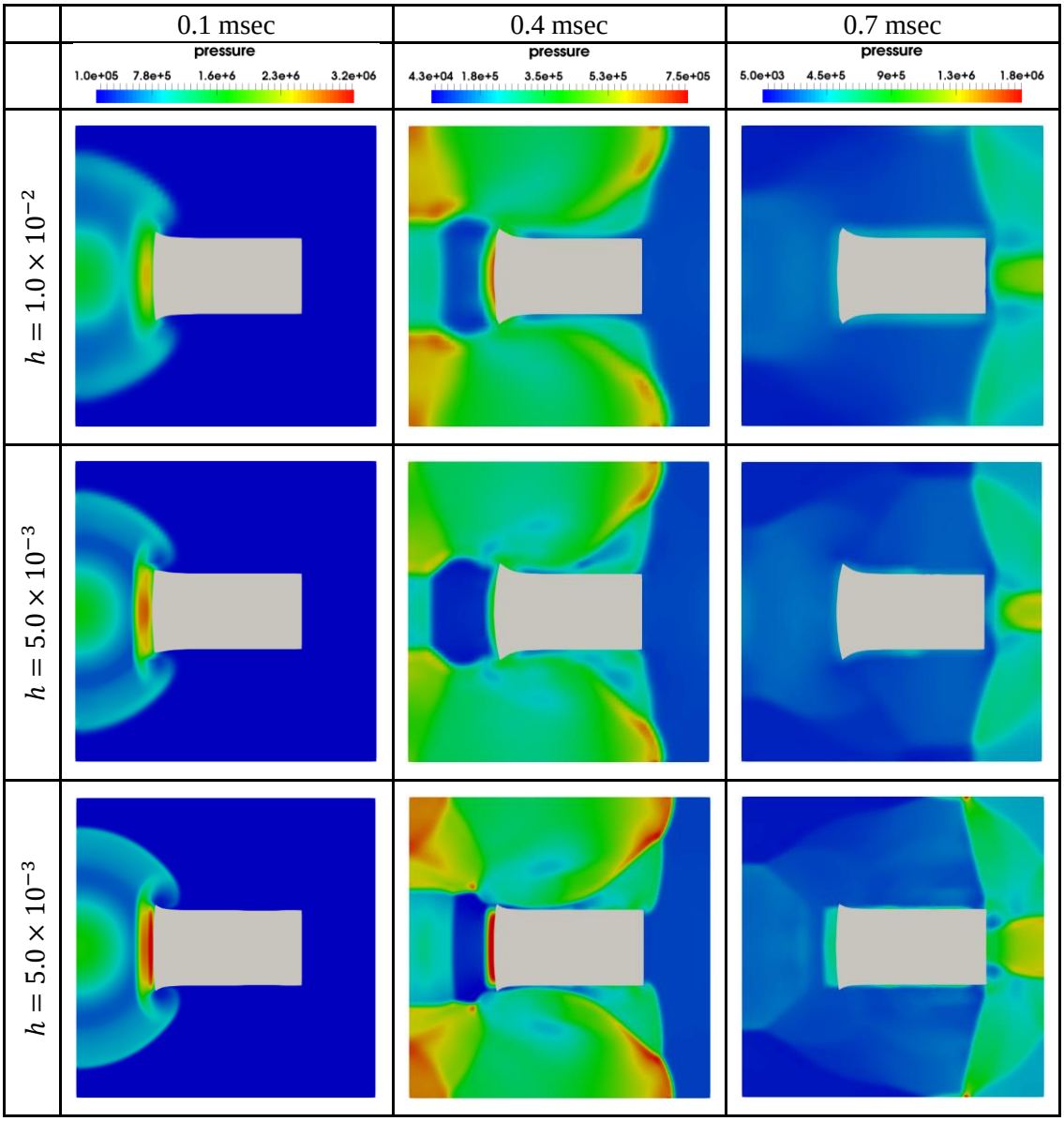
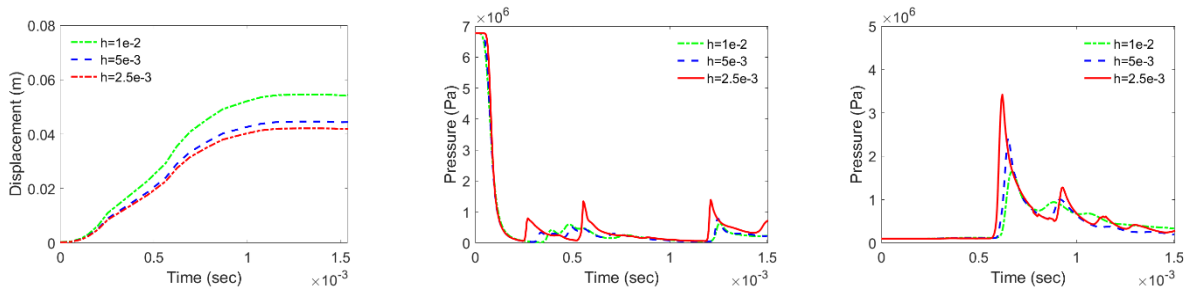


Figure 77. The pressure field and steel plate position at different timestep under different refinements at time of 0.1, 0.4, and 0.7 ms.

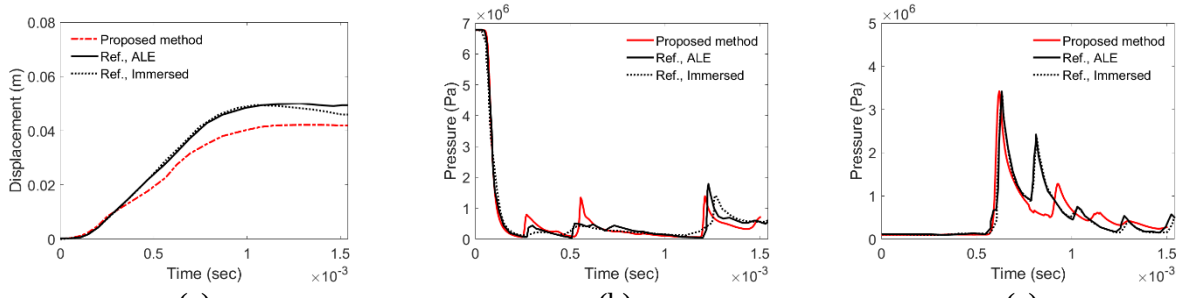
Figure 78 expresses the time history of horizontal displacement of the center of the bar, air pressure at the center of detonation, and air pressure at the center of the right wall. It can be verified that the proposed immersed solution converge as the discretization is refined.



(a) (b) (c)

Figure 78. Convergence study of the proposed immersed formulation under three refinements. (a) Horizontal displacement of the bar; (b) pressure at the center of detonation; (c) pressure at the center of the right wall.

Figure 79 shows the comparison of the proposed method with refinement level $h = 5.0 \times 10^{-3}$ with the reference solution. From the displacement history, the chamber trajectory of proposed method is like that of reference solution in ALE and immersed simulation, although the total displacement is smaller in our case. From pressure history in the Figure 79 (b) and (c), the response of pressure reflection are in good agreement with the reference solution. The pressure by proposed method at the detonation center is higher than the reference solution, while the pressure at the center of the right wall is lower than that of reference solution



(a) (b) (c)

Figure 79. Chamber detonation. Comparison of immersed simulation with reference solution [43]. “Ref.” indicates the reference solution, “ALE” indicates the Arbitrary Lagrangian Eulerian computation, and “immersed” is the reference immersed method proposed in [43]

6.3.6 Flexible Panel under Shock Wave

In this example, a panel of length 40 mm and thickness 1mm is subjected to a shock load in a tube [173], and the dimensions and set-up of the problem are expressed in Figure 80. The panel is assumed to be elastic with Young's modulus of 220 GPa, an initial density of 7600 kg/m^3, and the Poisson ratio of 0.33. The shock is initially placed 5mm ahead of the step. The panel is hit by a shock wave with the Mach number of 1.21, where the pre-shock and post-shock fluid initial conditions resemble airs can be found in Figure 80. The reflective boundary conditions are employed for all boundaries except for the inlet boundary, where the initial conditions are applied as the inlet boundary condition. The locally refined mesh is applied for the fluid domain, and the fluid mesh resolution around the panel is 0.4 mm, and the panel is with the uniform discretization of size 0.1 mm (see Figure 81). The pressure history from the pressure sensor (see Figure 80) is compared with the experimental data in [173] to validate our computational formulation.

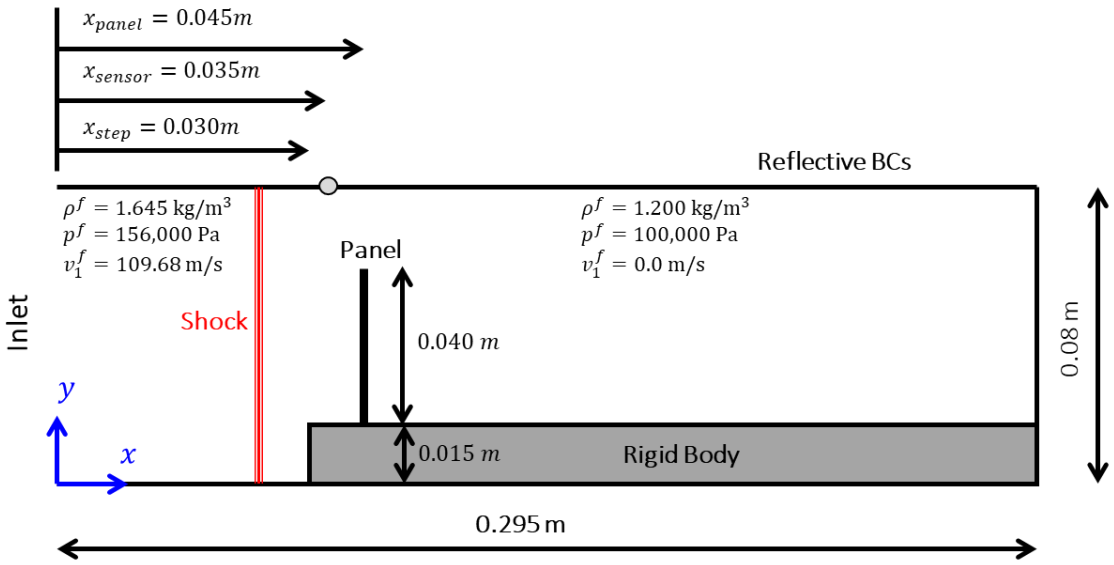


Figure 80. Flexible Panel under shock loading: geometry, setup.

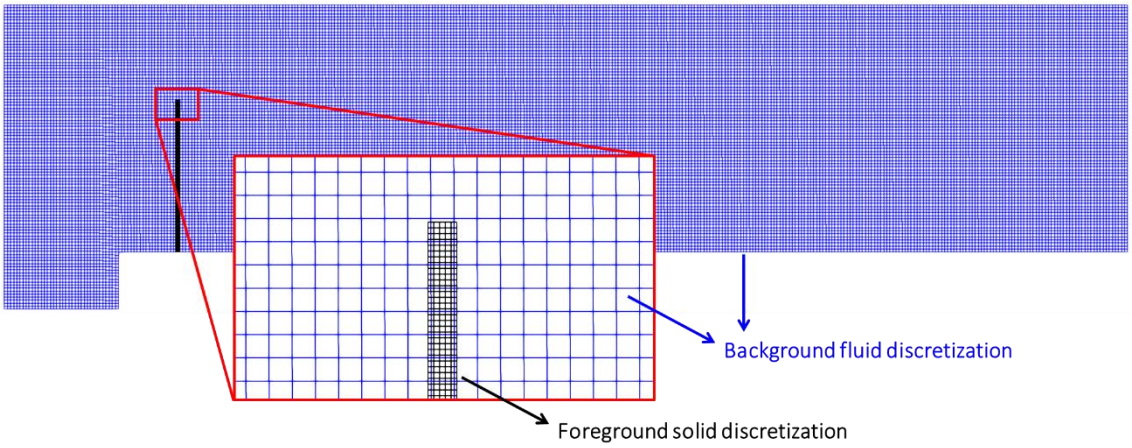


Figure 81. Discretization of the background fluid domain and foreground solid domain.

The pressure field at different time steps during the simulation can be found in Figure 82. The shock wave impacts the panel, causing the vibration of the panel, and the shock wave propagates reversely between the panel, step and tube boundary. The vortex generated near the panel can be observed as well. There qualitative observation is consistent with the experiment description [173].

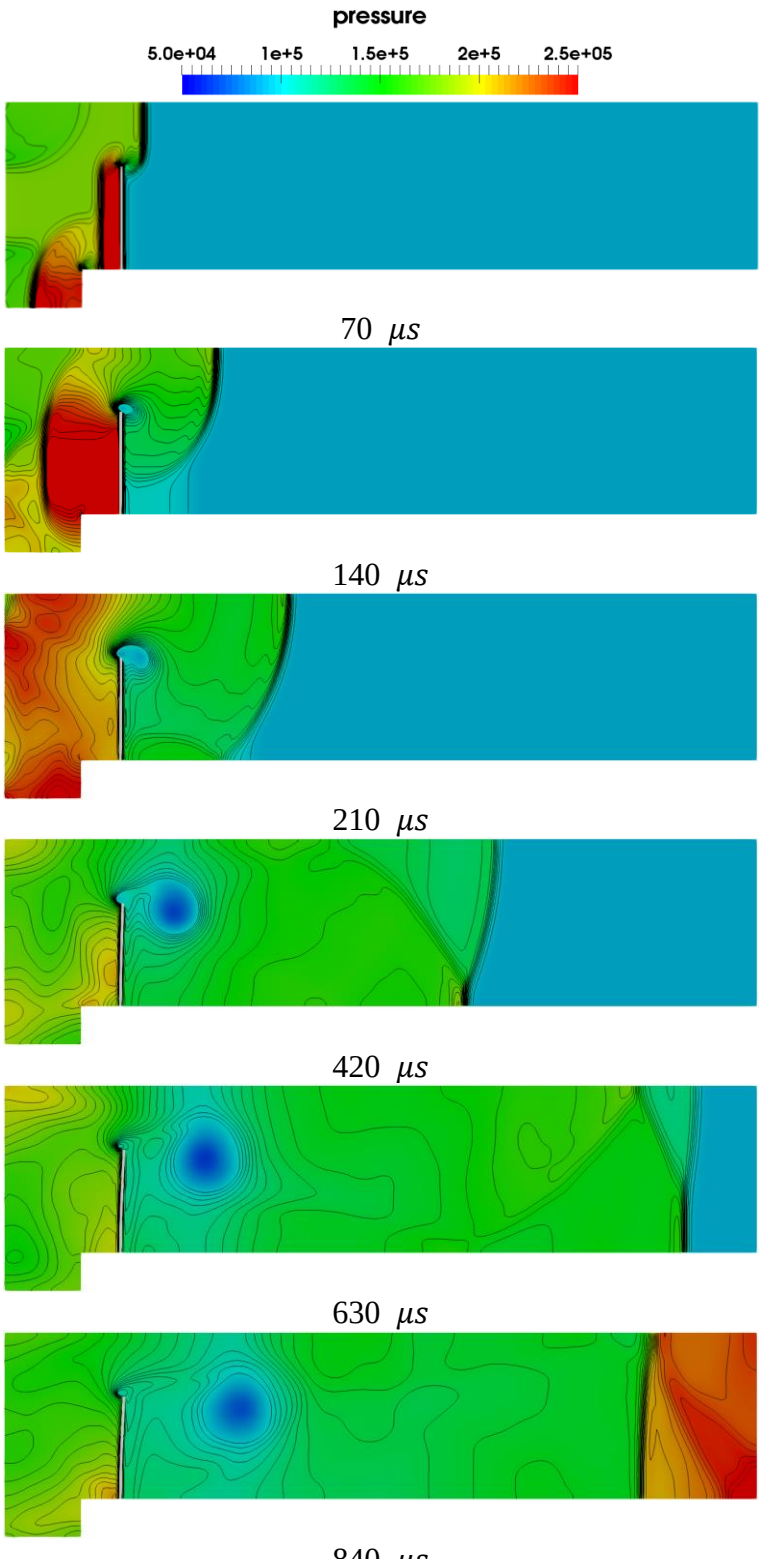
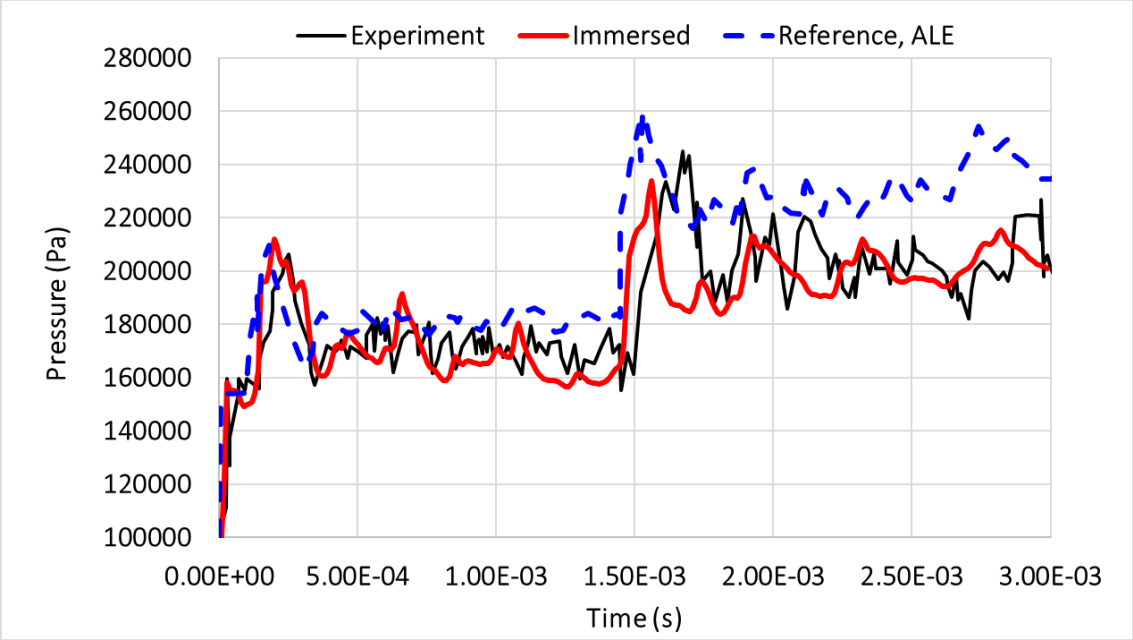
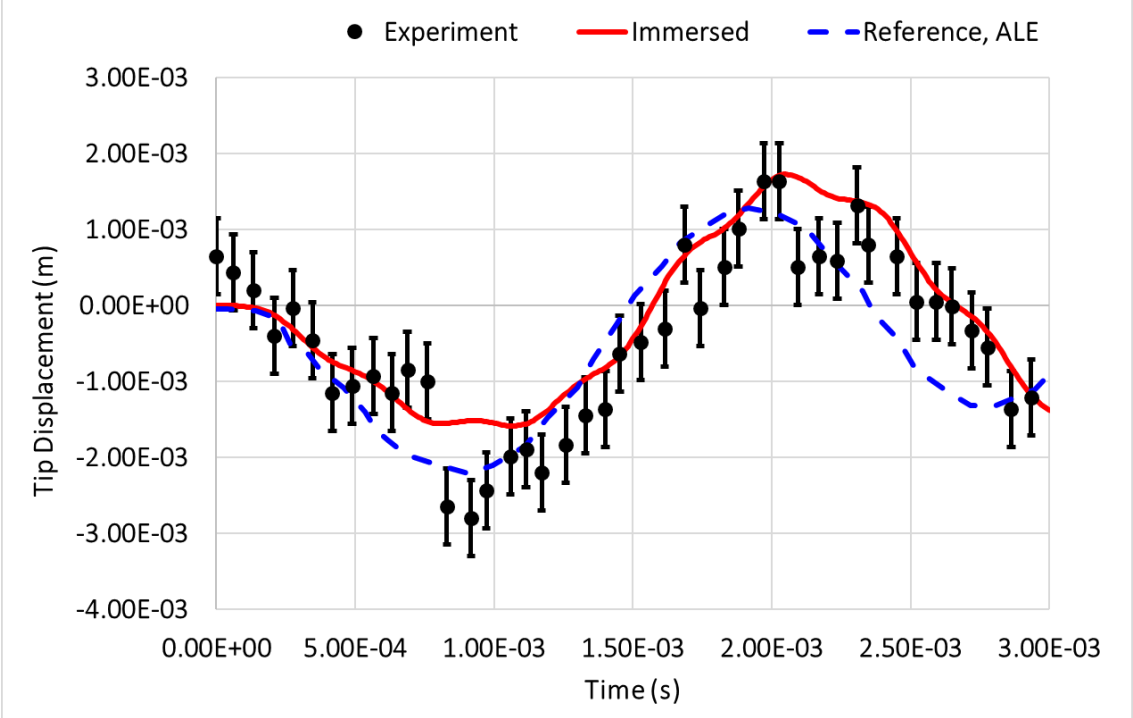


Figure 82. Flexible panel subjected to a shock load: Pressure distribution and panel deformation at different timesteps. The dark curves are density contour ranging from 1.225 kg/m^3 to 2 kg/m^3

The comparison of simulation results with experimental data for the pressure at the pressure sensor can be found in Figure 83. Compared to experimental data, the proposed immersed simulation captures the physical behavior of the shock wave reflection between the computational and experiments. The tip deflection of the panel shows good agreement with the experimental data, where the period of the vibration is consistent with the experimental result. The amplitude of the tip deflection is slightly smaller than the experimental results at the first peak. The pressure history from the proposed immersed method captures the pressure peak when reflected shock waves impact on the pressure sensor, showing a good agreement with experimental data, while the reference ALE simulation deviates after the second shock wave impact.



(a) Time history of the pressure at the pressure sensor location.



(b) Time history of the tip deflection of the flexible panel.

Figure 83. Flexible panel subjected to a shock load: comparison of numerical results, reference ALE simulation [173] and experimental data [173].

Besides the case of 40mm panel length, the 50 mm panel case is also tested, as shown in Figure 93. The vortex generation, shock front propagation and reflection, and vortex shedding are all well captured. The numerical simulations successfully reproduce the observation shown in the experiment in [173].

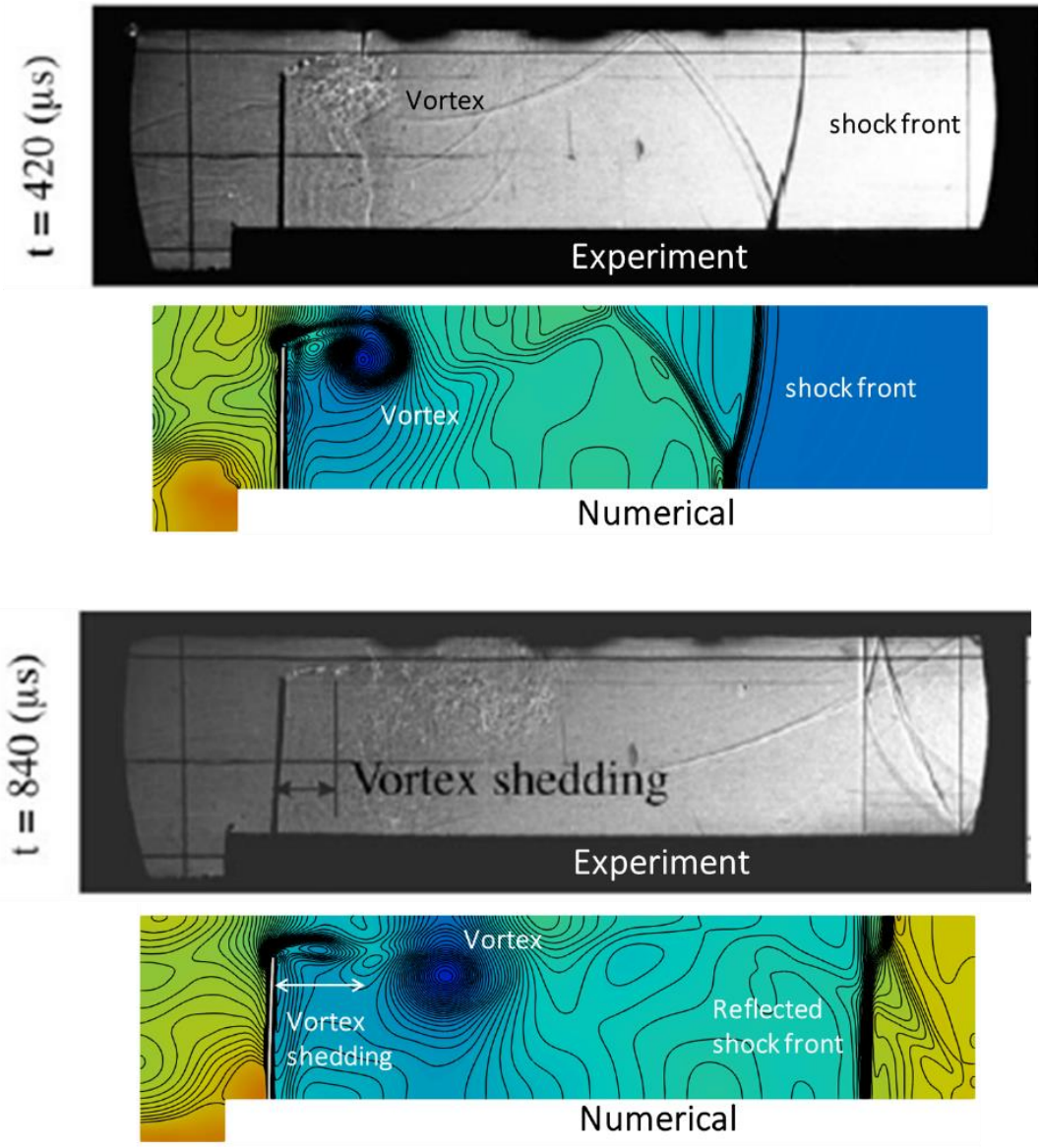


Figure 84. Flexible panel subjected to a shock load (50 mm case): comparison of numerical results and experiments at different timestep. Experimental results are extracted from [173]. The dark curves are density contour ranging from 0.9 to 2.0 kg/m^3

6.4 Summary of Proposed Immersed FSI formulation

In this study, an immersed meshfree formulation is developed for modeling shock waves in the FSI problem under variational multiscale immersed RKPM framework with the enhanced shock algorithm developed under the MUSCL-SCNI method. The fluid and solid are coupled through a volumetric constraint approach under an immersed setting. By the variational multiscale approach, the fluid velocity is decoupled into coarse- and fine-scale, and the fine-scale fluid velocity here represents the residual of the coarse-scale fluid momentum equations. A mixed density approach is used, such that the inertia effect at the foreground solid domain explicitly vanishes, and the fluid system can be solved by MUSCL-SCNI formulation and the solid velocity is updated based on the constraint equation. The MUSCL-SCNI formulation enables the control of the Gibbs phenomenon and enhances the accuracy in the FSI simulation involving shocks. The developed FSI formulation has been verified and validated against benchmark problems and demonstrate the method's capability in handling FSI under shock regime.

This chapter, in part, is being prepared for submitted for publication of the material as it may appear in "TH. Huang, JS. Chen and MR. Tupek. An Immersed Variational Multiscale RKPM Formulation. Part II - Fluid-Structural Interaction. 2020". The dissertation author was the primary investigator and author of this paper.

Chapter 7 Meshfree Modeling for the Blast Events: Concrete Slab Subjected to Blast Loading due to High Explosives

This chapter introduces the application of meshfree algorithm developed in this dissertation for the blast event modeling. A brief introduction of blast events and blast wave modeling is given in Chapter 7.1. Two example problems are given in Chapter 7.2 and Chapter 7.3, and the simulation results are also discussed. Brief conclusion is summarized in Chapter 7.4.

7.1 Blast Event Modeling

The blast events modeling describes the dynamical response of the solid structures or materials under very fast transient and high magnitude of peak loads, e.g., concrete under blast impact due to high explosive. The strong blast wave acting on brittle material usually results in

remarkable damage and fragmentation and often causes an explosion crater, as shown in Figure 85 , where an explosion crater of a concrete slab is formed due to the near-field TNT explosion. The crater is surrounded by cracks in radial and hoop direction.

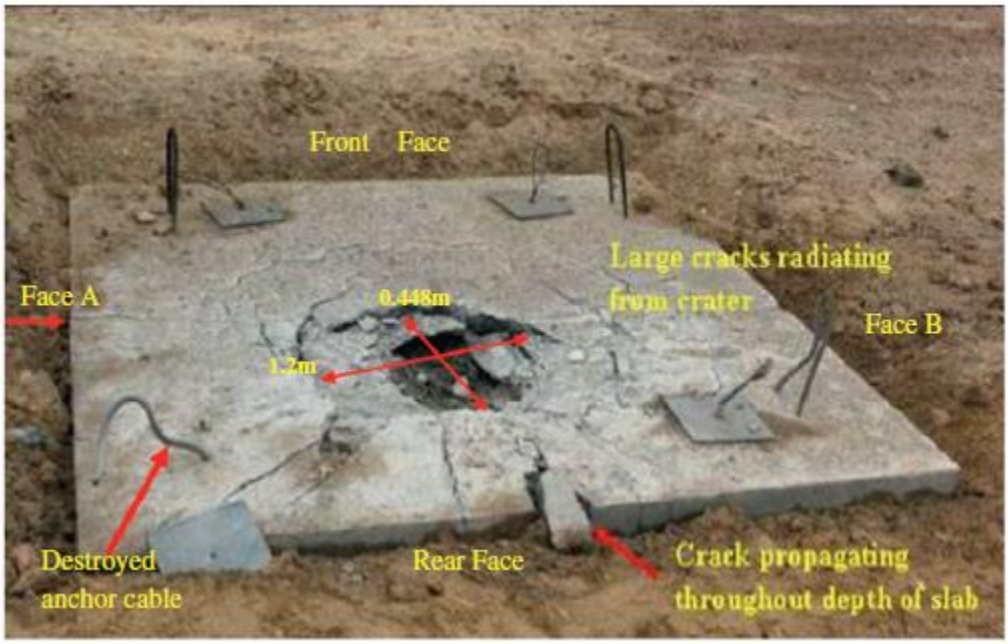


Figure 85. The crater of concrete pavement slab after blast loading [174].

During the cratering, the material is ejected from the surface of the main body by an explosive event at the surface. It is typically bowl-shaped. High-pressure gas and shock waves cause three processes responsible for the creation of the crater:

1. Plastic deformation of the material surface.
2. Ejection of material from the material surface by the explosion.
3. Spallation of the material surface.

Depending on the position of explosion occurs and the composition of the material, different types of crater geometry are formed. In the cratering process, the material experiences strong surface deformation and material ejection, which are all difficult for conventional mesh-based method with ALE-type FSI approach. Therefore, the proposed immersed meshfree and shock-

enhancement algorithm are employed to model the cratering process for the structure under blast loading due to high explosive. Two numerical examples are tested to benchmark the performance of the proposed method.

7.2 Example 1: Concrete under Explosive Loading

In this example, the concrete slab subjects to an explosive loading shown in [2]. As shown in Figure 86, a concrete slab is loaded with a plane wave generator with explosives which consist of an inner cone of TNT and an outer cone of Composition B.

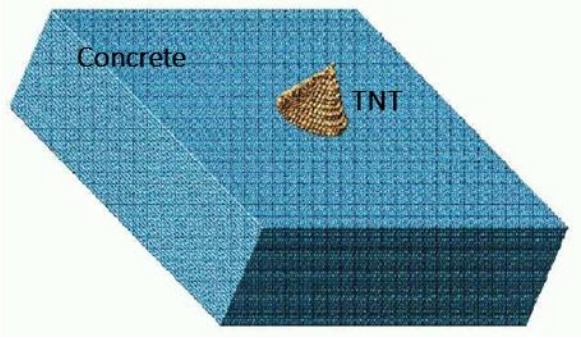


Figure 86. Concrete slab loaded with a plane wave generator. The figure is obtained from [2].

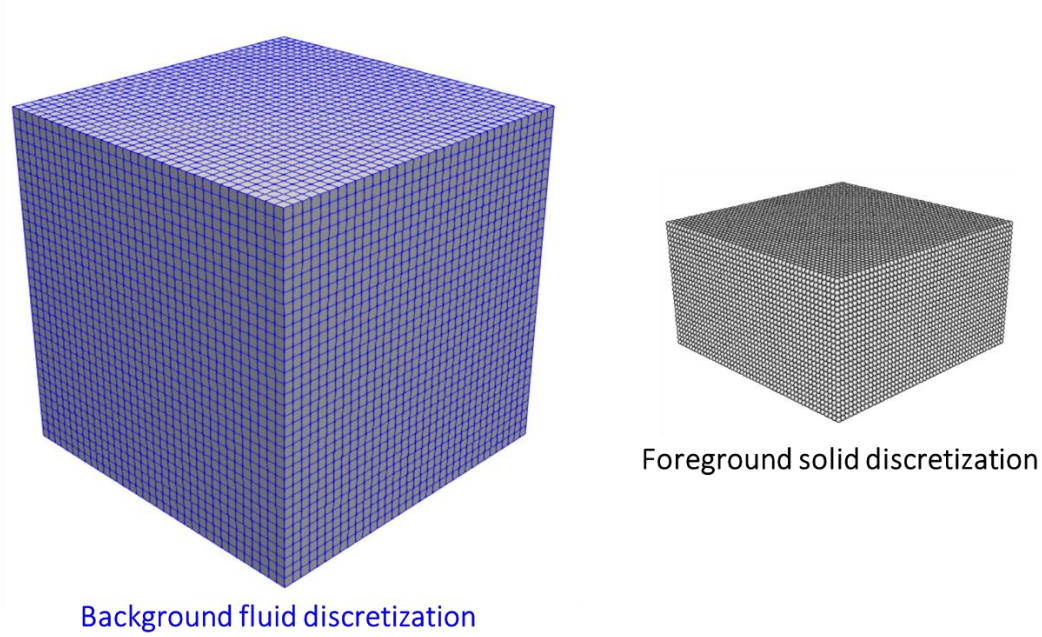


Figure 87. The discretization of the solid (concrete) domain and fluid (air) domain.

The dimension of the slab is 1.2 m × 1.2 m × 0.32 m. Due to the symmetry of the model, only a quarter of the domain is modeled. The quarter domain is discretized with ~ 45,000 nodes. The explosive is modeled in the background fluid as an explosive gas with a cone of the diameter of 0.103 m and a height of 0.075 m. The slab is immersed into a background mesh with dimensions 1.4 m × 1.4 m × 1.0 m, which is discretized with ~ 50,000 nodes as shown in Figure 91. The concrete is modeled by the Holmquist-Johnson-Cook model (HJC) [175] with compressive strength of the concrete is 48 MPa, and the rest of the parameters can be found in [2]. The concrete slab is assumed to be initially at room temperature under the atmosphere. The reproducing kernel approximation with a linear basis and cubic B-spline kernel is adopted for the approximation of all variables. The normalized support size equaling 2.0 are employed for both background air and solid.

The cross-section and 3D view of concrete slab damage evolution can be found out in Figure 88 and Figure 89. In the early stage of the simulation, it can be seen that the TNT explosion causes

surface damage, and the damage particles are ejected from the concrete surfaces. The damage profile at 0.06 msec shows the initiation of the spalling at the bottom side of the slab. After 0.1 msec, the damage spread out quickly to the edge surfaces of the concrete slab, as shown in the 3D view of the damage process. The crater starts to form after 0.4 msec, and also crack forms in the diagonal direction. The shock wave causes strong plastic deformation of the material surface, following by the material ejection along the tangential direction of the crater surface. The final damage pattern and the damage process of the concrete slab show a qualitative agreement between the current computation and the numerical reference simulation by the SPH method in [2].

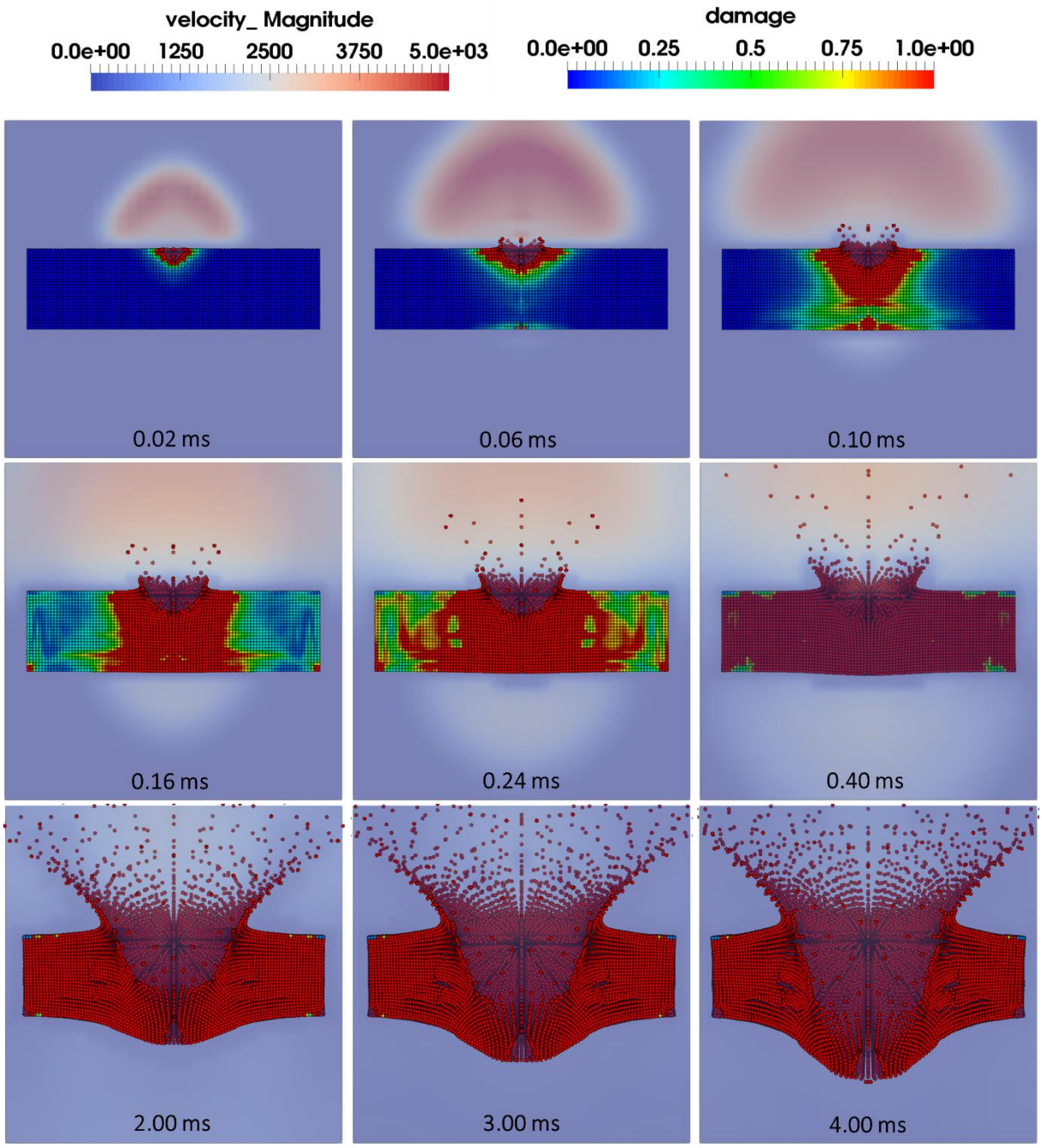


Figure 88. Cross-section view of concrete slab subjected to explosive loading: evolution of damage in concrete and air velocity distribution.

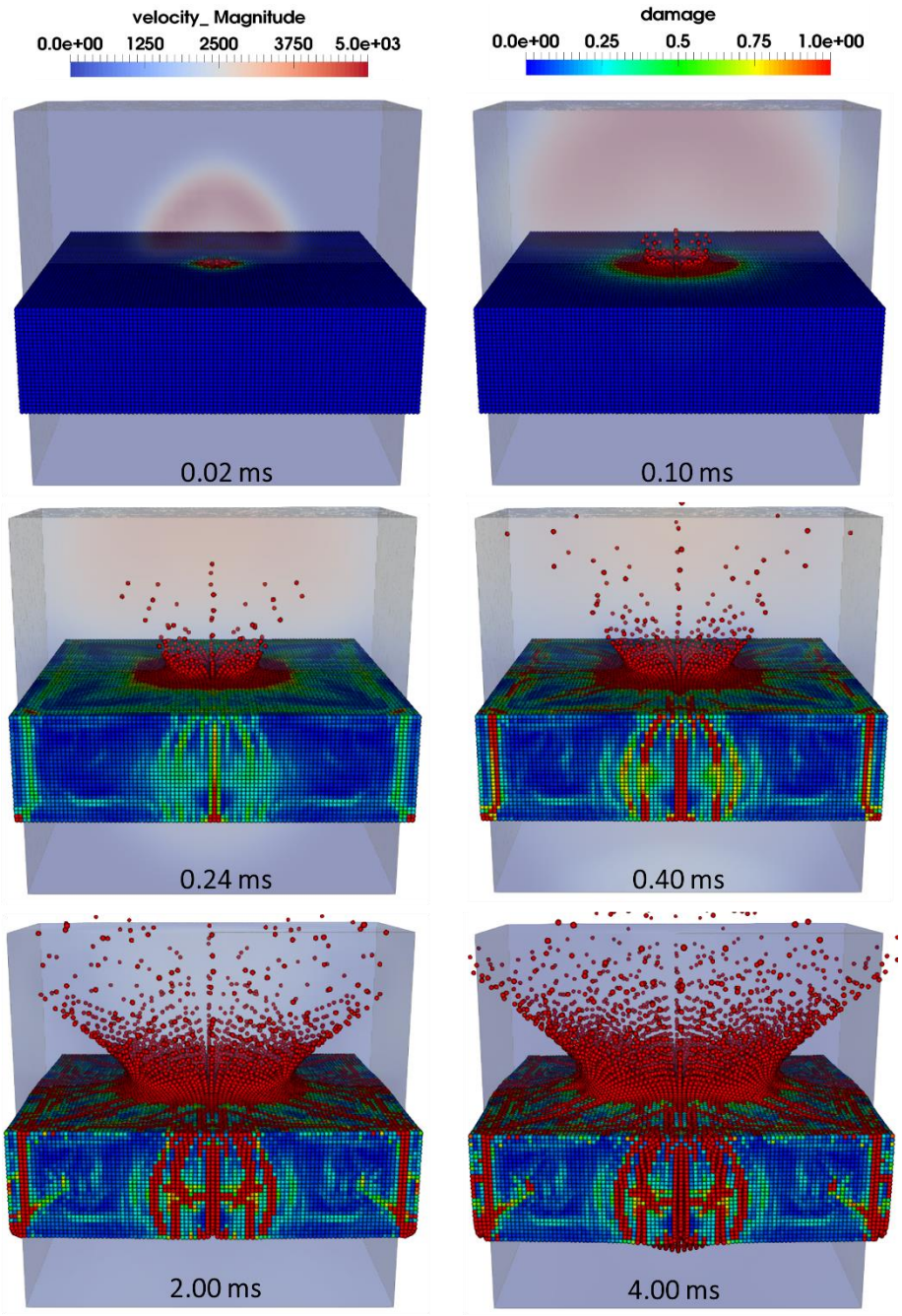
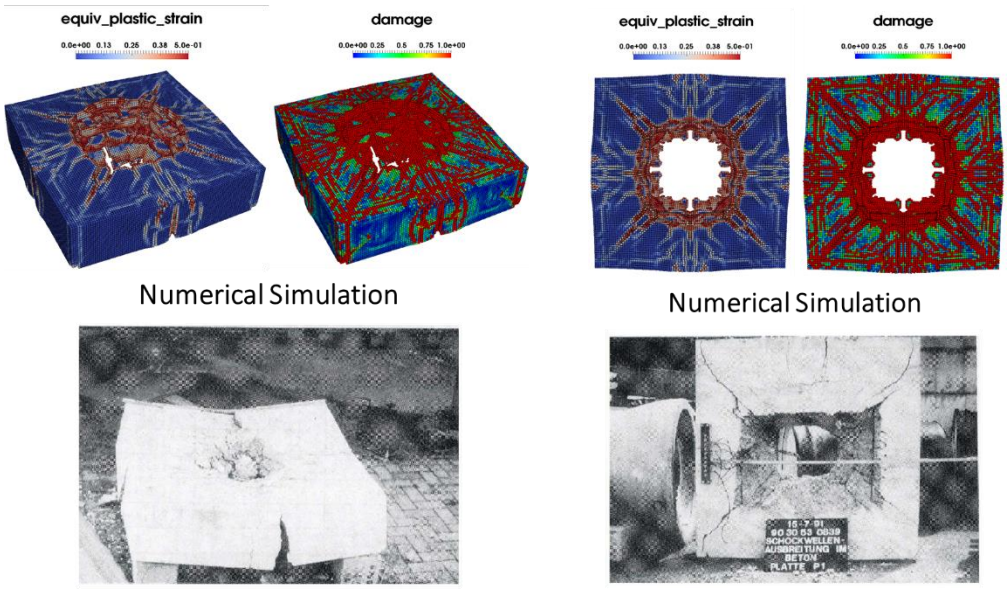


Figure 89. 3D view of concrete slab subjected to explosive loading: evolution of damage in concrete and air velocity distribution. The background domain is cut to half for visualization.

The comparison of the damaged concrete slab with the experimental results can be found in Figure 90. The perforation of the concrete slab is successfully model as observed in the experiment. The crack propagation at the side surfaces are also captured. The diagonal crack from the bottom

view can also be observed in the numerical simulation, which qualitatively agrees with experimental results.



(a) Top view (b) back view

Figure 90. Comparison of final concrete slab from top view and bottom view. In the simulation, the equivalent plastic strain over 1.0 is removed to exclude the particles flying away after explosion. The experiment result is obtained from [2].

Finally, the comparison of the crater diameter at the upper and lower side of the concrete slab is listed in Table 7. The proposed simulation shows good agreement in the lower crater diameter comparing to the experiment. The upper crater diameter is similar to the SPH simulation demonstrated in [2], but deviates from the experimental results. Due to the large scatter in the experimental data, it is pointed out in [2] only qualitative predictions can be made for the problem under consideration. Hence, the simulation results shown in this study capture the cratering process successfully.

Table 7. Crater size of the concrete.

	Upper crater diameter (cm)	Lower crater diameter (cm)
Proposed Study	73	50
Numerical, SPH [2]	75	50
Experiment	62	51

7.3 Example 2: Concrete Pavement Slab subjected to Blast Loading

In this study, the experiment of the concrete pavement slab subjected to the high explosive [174] is picked. The setting of the tests can be found in Figure 91, where the slab is subjected to one blast detonation. A bomb with equivalent 7.3 kg TNT charge weight was placed at the center of the concrete slab with the bomb mass center elevated at about 170 mm above the slab center surface. The concrete slab at the site is with a dimension of 2.8 m by 2.8 m and 0.275 m thick. It is reported in [174] that minimal reinforcement is installed because of the convenience of transportation, where the reinforcement is installed at the bottom side of the concrete (T12 bars in both directions at around 350 mm spacing). Four anchorages are installed to anchor the pavement slab to the ground, as shown in Figure 92, where the bottom and four side surfaces are attached to the soil ground.

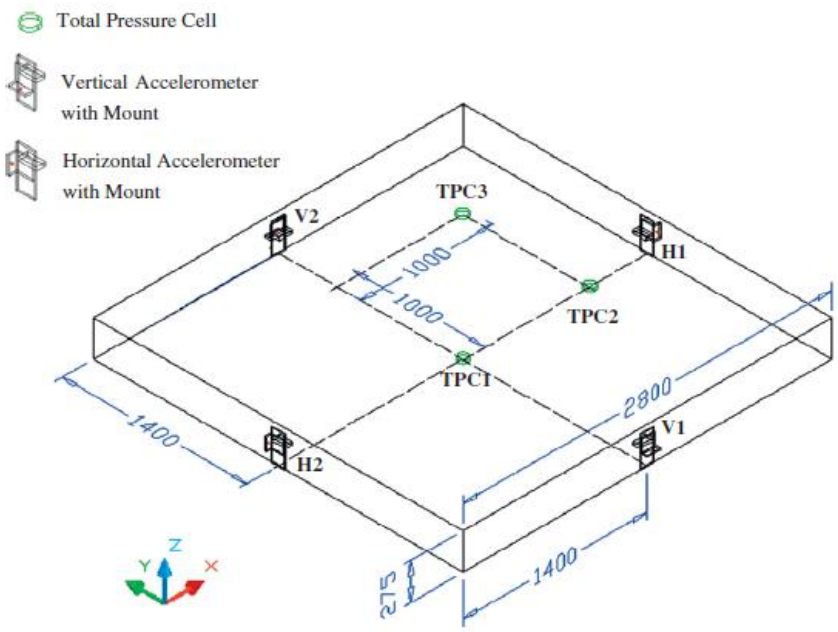


Figure 91. Layout of the concrete employed in the experiments [174]. The unit is in mm.

Figure 92. Concrete pavement slab before blast event [174].

Various measuring instruments were installed onto pavement slabs to measure the response of the pavement slab during the blast, as shown in Figure 91. Four accelerometers were installed in the middle of the side of the slab to measure the vertical and horizontal acceleration at four faces. The three total pressure cells (TPC1, TPC 2, and TPC3 in Figure 91) were buried in the soil just below the bottom of the slab to measure the pressure transferred from the pavement slab.

Due to the symmetry condition of the model, only a quarter domain with symmetric boundary conditions are employed, as shown in Figure 93. The concrete with dimension 1.4m by 1.4m by 275 mm and only the top surface is exposed to the air where the rest of the surfaces are attached to the soil. The rebar is installed at the bottom surfaces with span 350 mm according to the experiment. The soil model is 2.0-meter height in the z-direction. Strong damping is applied to the soil model to absorb the wave impact from the concrete so that the concrete is not influenced by the reflective wave from the bottom boundary. The background air only covers the domains from -0.5m to +0.5m in z-direction. The solid nodes outside of the background fluid is modeled by the

Lagrangian RKPM. The reproducing kernel approximation with a linear basis and cubic B-spline kernel is adopted for the approximation of all variables. The normalized support size equaling 2.0 is employed for both air and solid. The simulation time is 4.0 ms, with a timestep size of 0.1 μs.

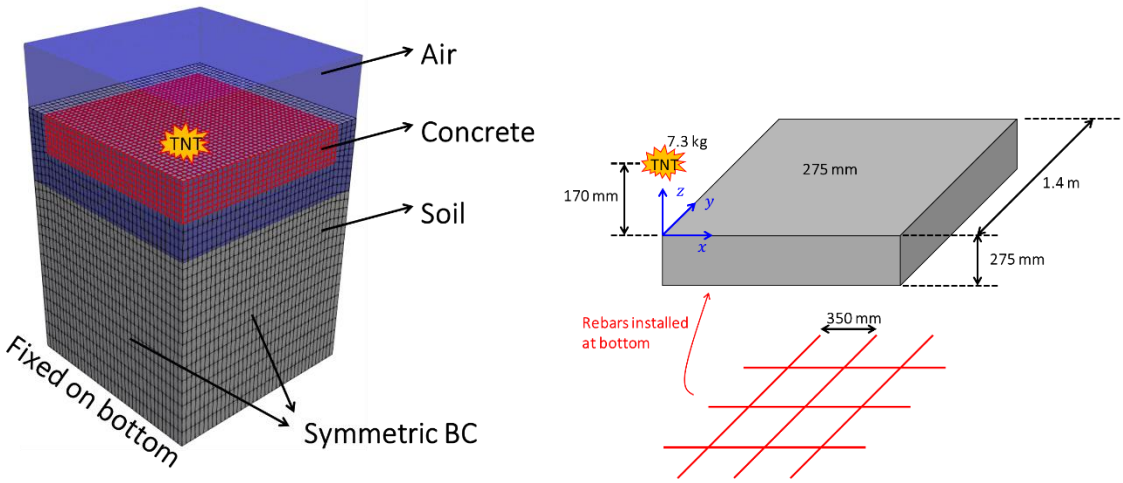


Figure 93. Numerical model for the concrete under TNT explosion. The grids in the left hand side indicates the discretization of the concrete and soil model.

In the numerical test, the material model and the corresponding material properties are summarized as follows

- The concrete is modeled by the Holmquist-Johnson-Cook model (HJC) [175]. The parameters are listed in Table 8.
- the steel rebars are modeled by the J2-plasticity model. The parameters are listed in Table 9.
- The soil is modeled by the Drucker-Prager model. The parameters are listed in Table 10.
- The TNT detonation is modeled by the Chapman-Jouquet Model with JWL equation of state. The parameters are listed in Table 11.
- The air with ideal gas equation of state at atmosphere under room temperature is modeled.

Table 8. Material properties of HJC model for Concrete.

Material	Density	Young's Modulus	Poisson ratio	Compressive Strength	Tensile Strength
Concrete	2400 kg/m^3	27 GPa	0.2	40 MPa	3.5 MPa

Table 9. Material properties of Drucker-Prager model for soil

Material	Density	Young's Modulus	Poisson ratio	Yield Stress	Friction Angle
Soil	2100 kg/m^3	700.0 MPa	0.3	4.60 MPa	26 degree

Table 10. Material properties of Drucker-Prager model for steel rebar

Material	Density	Young's Modulus	Poisson ratio	Yield's Stress
Steel Rebar	7850 kg/m^3	207 GPa	0.3	460 MPa

Table 11. Material properties of Chapman-Jouquet model with JWL equation of state for TNT

Material	Density (kg/m^3)	D (m/s)	p_{CJ} (GPa)	A_1 (GPa)	A_2 (GPa)	R_1	R_2	ω
TNT	1630	6930	21.0	373.8	3.747	4.15	0.90	0.35

As shown in Figure 94, the cratering process due to the impact of the TNT explosion is successfully modeled, where the cratering process is revealed. Bowl-shaped geometry is shown after the TNT explosion. The shock wave causes strong plastic deformation of the material surface, following by the material ejection along the tangential direction of the crater surface. The spallation of the material surface is also observed. Finally, the detonation wave penetrates through the concrete and shows the soil top surface, as shown in Figure 95.

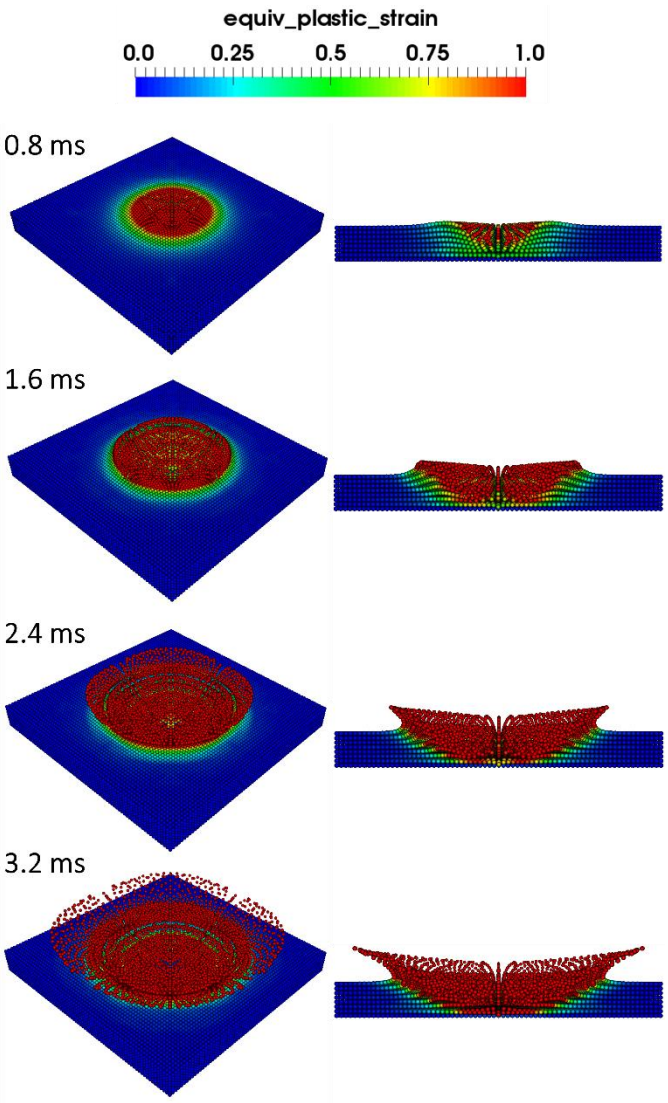


Figure 94. Dynamic response of concrete under TNT explosion: Top and cross-section view.

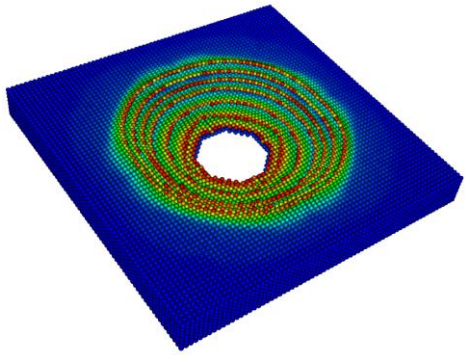


Figure 95. Final crater geometry of concrete. (particles with equivalent plastic strain greater than 1.0 are removed).

The final crater geometry is expressed in Figure 96, where it can be cleared to see that the TNT explosion penetrates through the concrete, which is consistent with the experimental observation. The diameter of the crater ranges from 0.57 m (bottom surface) to 1.80 m (top surface), which shows good agreement with the crater size measured in the experiment (0.448 m to 1.2 m).

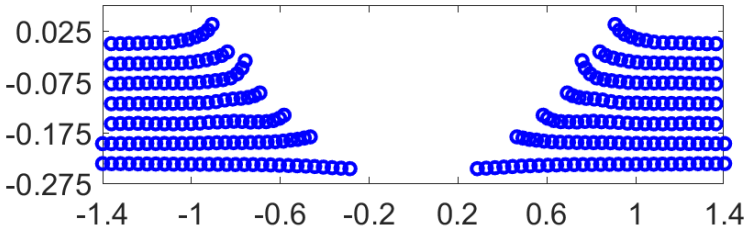


Figure 96. Cross-section of crater geometry at final step (particles with equivalent plastic strain greater than 1.0 are removed).

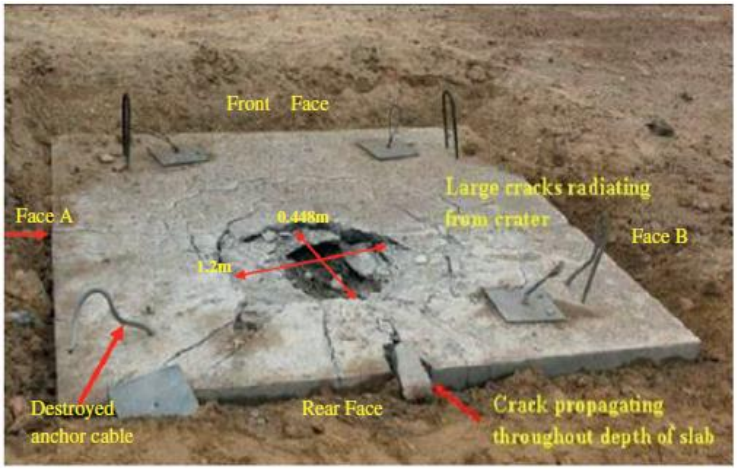


Figure 97. Final crater geometry in the experiment.

To confirm if the magnitude of the pressure loading on the concrete, the air pressure history close to the concrete center is measured and compared to an empirical formula called Friedlander equation [176]. The Friedlander equation describes the pressure at a spatial location experience a blast wave passing by, and the expression is expressed in Eq. (284):

$$P(t) = P_s e^{-\alpha \frac{t}{t^*}} \left(1 - \frac{t}{t^*}\right) + P_0 \tag{284}$$

where $P(t)$ is the blast wave pressure at time t, P_s is the peak pressure, α is the decay coefficient, t^* is the time at which the pressure decay to zero, and P_0 is the pressure at steady state. P_s can be estimated by the empirical formula for TNT [177]:

$$P_s = \frac{1772}{Z^3} - \frac{114}{Z^2} + \frac{108}{Z} \text{ (Unit: kPa)}, \qquad Z = \frac{R}{\sqrt[3]{W}} \tag{285}$$

where $P_s = 2.61 \times 10^{10}$ Pa from (285), which is close to the peak pressure, 2.55×10^{10} Pa, in the numerical simulation. The decay time and coefficient can be obtained from the empirical formula from [177], where $t^* \approx 1 \times 10^{-3}$ and $\alpha \approx 20$. As shown in Figure 98, the shock wave pattern is qualitatively following waveform similar to the Friedlander wave shown in Eq. (284) with the given value of P_s, t^*, α.

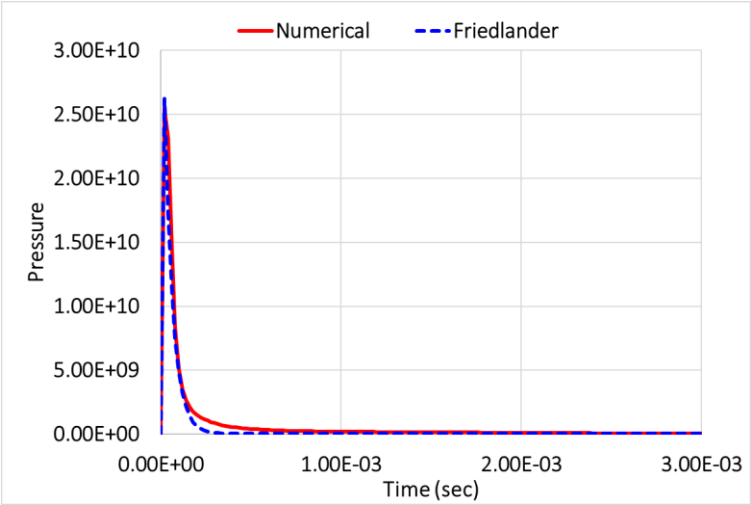


Figure 98. Dynamic response of concrete under TNT explosion.

The measured results of the accelerometers from the experiments and numerical simulation are summarized in Table 12. From Table 12, it is shown that the peak accelerometer recorded at

V1 and V2 were practically the same in the normal concrete pavement slab. However, it was indicated in the experiments [174] that the peak acceleration at H2 was much larger than that at H1 due to the unsymmetrical placement of the bomb above the center of the slab, where the bomb might be closer to H2 than H1. From the acceleration read in the experiment, it can be found out the peak horizontal acceleration is lying within the range of the experimental results. The horizontal acceleration recorded from the simulation is also close to that of experiments.

Table 12. Peak acceleration measured from the experiment and computation.

Accelerometer Position	Experiments (m/s^2)	Numerical (m/s^2)
V1 (vertical)	21,480	23,363
V2 (vertical)	22,820	
H1 (horizontal)	14,820	27,141
H2 (horizontal)	60,450	

In addition to acceleration measured, Table 13 reports the peak pressure recorded in the experiments and numerical simulation. TPC1 is reported destroyed by the explosion in the experiment. In the experiment, the peak reading of TPC 2 was higher than that of TPC 3, as TPC3 is far more from the center of the blast. In the numerical simulation, the peak pressure measured is close to that of experimental results.

Table 13. Peak pressure measured from the pressure cell.

Pressure Sensor	Experiments (kPa)	Numerical (kPa)
TPC1	Damaged	49,074
TPC2	178.00	182.26
TPC3	152.00	150.19

From the numerical results and quantitative comparison, the proposed immersed meshfree simulation provides a stable simulation and promising results for blast events modeling. To further investigate the influence of reinforcement on the concrete, a numerical test is performed where the reinforcements are installed at the mid-span of the concrete instead of the bottom side, as shown in Figure 99. In the previous case, the rebars are installed at the bottom side of the concrete, so the reinforcement effect is limited.

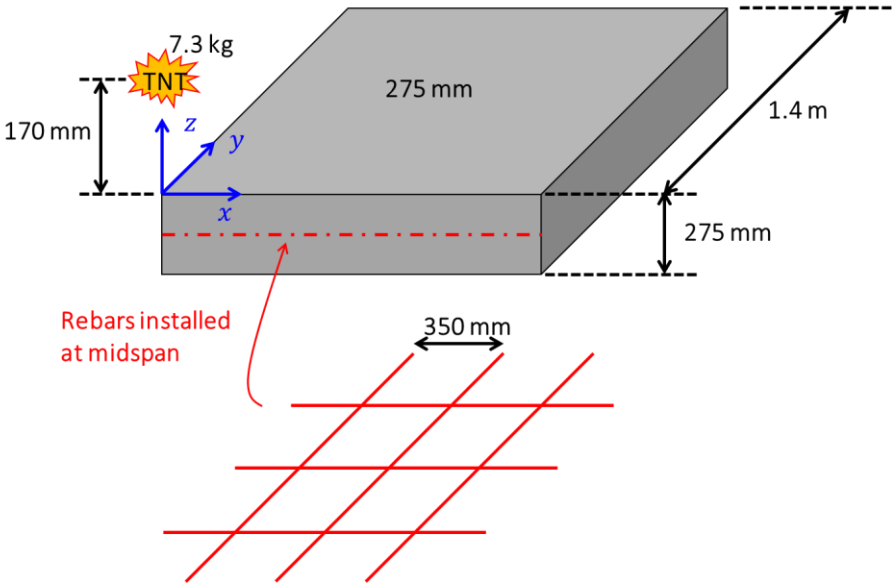


Figure 99. Numerical model for the midspan reinforced concrete under TNT explosion.

The comparison of dynamical responses of concrete under different reinforcement is expressed in Figure 100. With the reinforcement at midspan, the impact of shock wave does not penetrate all the way through the concrete. Also, the plastic deformation concentrates more on the top-half domain of the concrete than that in the case of the bottom reinforcement. From the final crater geometry shown in Figure 101, the crater is shallower but boarder in the case of midspan reinforcement.

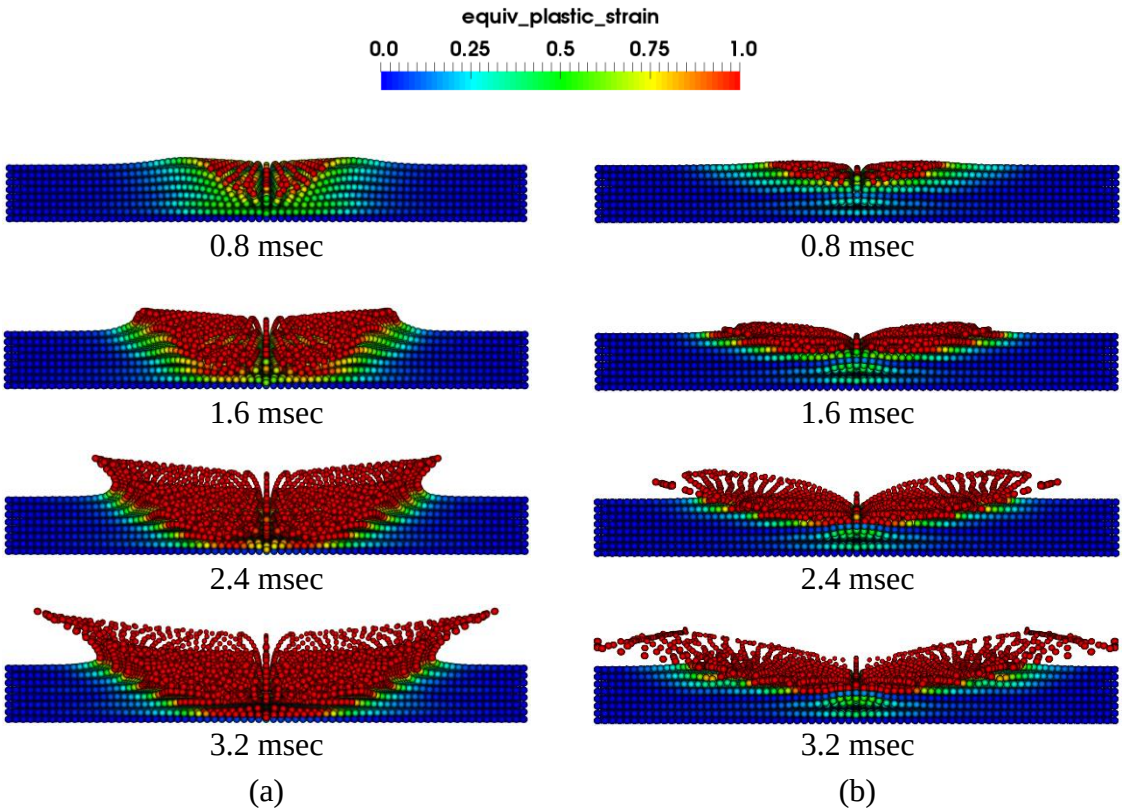


Figure 100. Cross-section of cratering process under different reinforcement installation: (a) reinforcement at bottom; (b) reinforcement at midspan

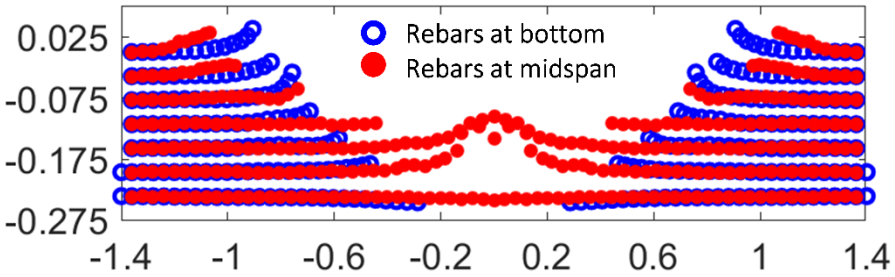


Figure 101. Comparison of final crater geometry under different location of the rebar reinforcement.

7.4 Summary of Blast Events Modeling

In this chapter, the developed shock enhanced immersed RKPM formulation is applied for the blast event. The cratering process in the application of concrete slab under the TNT explosion is successfully modeled. The plastic deformation and damaged particles scattering are successfully modeled. The quantitative and qualitative comparison of numerical simulation and experimental observation demonstrate the proposed method is promising in a modeling concrete damage process in the blast event. In the case study of the first numerical example, it is found the rebar reinforcement installed in the midspan prevents the shock wave penetration. The crater is shallower but broader in the case of midspan rebar. In the second example, the severe damage due to the blast loading causes the perforation of the concrete slab. It is expected the proposed method can be applied to other types of blast events in future work.

This chapter, in part, is being prepared for submitted for publication of the material as it may appear in "TH. Huang, JS. Chen and MR. Tupek. An Immersed Variational Multiscale RKPM Formulation. Part II - Fluid-Structural Interaction. 2020". The dissertation author was the primary investigator and author of this paper.

Chapter 8 Conclusion

This chapter provides a summary on the material presented in the dissertation and an outlook on the related research directions.

8.1 Conclusion

A stable and efficient meshfree method is proposed to solve the shock wave modeling in fluids and FSI problems. It has been shown that the Riemann-enrichment embedded in the smoothed flux divergence operator under the SCNI framework can control the Gibbs oscillation in the appearance of the shock wave. The MUSCL-type oscillation limiter successfully suppresses the overshoot and undershoot behavior when the higher-order approximation is employed. The flux splitting approach with the modified upwinding scheme for advection flux avoids the over-smearing or instability at the shock front. Several benchmark problems have been analyzed, and the results demonstrate the excellent performance of the stabilized meshfree formulation for modeling the shock waves in fluids.

To effective model FSI problems with potential fragmentation in the solids, an immersed formulation is developed. As the first step in this development, a novel immersed approach for modeling heterogeneous material under nodally integrated reproducing kernel discretization has been presented. A volumetric constraint approach is employed to avoid tedious contour integral along with material interface as well as the possible ill-conditioned system in a conventional immersed approach. A variational multiscale approach is proposed for the immersed framework, termed variational multiscale immersed method (VMIM). The fine-scale solution represents the residual of the coarse-scale equations, where two approaches, residual-free bubble method, and approximated fine-scale solution have been investigated for obtaining the fine-scale solution. The embedment of the fine-scale solution results in a stabilized Petrov-Galerkin formulation, leading to increased accuracy and enhanced stability in the coarse-scale solutions. The naturally stabilized conforming nodal integration has been employed to achieve accelerated, stable and convergent solution. The proposed VMIM exhibits a better convergence rate comparing to the conventional immersed method with comparable efficiency. The proposed immersed RKPM method allows an effective body-unfitted spatial discretization for the material subdomains, suitable for the problem involving complex geometry. For problems with complicated geometry or topology, such as material microstructure, cumbersome treatments (e.g., adaptive refinement, or contour integral along with the material interface, etc.) commonly encountered in conventional methods have been effectively avoided in the proposed approach.

The variational multiscale immersed framework, formulated under the developed MUSCL-SCNI formulation, is extended to fluid-structure interaction (FSI) problems without and with shocks. The fluid/solid is coupled through the volumetric constraint, and the fluid velocity is decoupled into coarse- and fine-scale. By the VMIM formulation, the fine-scale fluid velocity can

be represented from the residual of the coarse-scale fluid momentum equations. The substitution of fine-scale fluid velocity into the coarse-scale system results in an unbalanced force acting on the fluid system. In order to employ explicit time integration schemes, a mixed density approach is introduced, such that the inertia effect at the foreground solid domain vanishes, and the fluid system can be solved by the MUSCL-SCNI formulation. The developed FSI formulation has been verified and validated against several benchmark problems. In particular, the flexible panel under shock loading and the cylinder lift-off problems are modeled to demonstrate the method's capability in handling FSI under shock regime.

Furthermore, the capability of the shock-enhanced immersed framework in modeling FSI is applied to the blast event modeling. The blast wave due to the detonation of high explosives, i.e., TNT, are modeled by the Chapman–Jouguet (CJ) model. RKPM naturally avoids computational challenges associated with low-quality meshes, which overcomes the numerical issues in modeling the large deformation and fragmentation of structure under blast loading in the conventional mesh-based methods. The concrete block slab under the TNT explosion has been tested, and the satisfactory performance of the proposed method has been obtained. Good agreement has been achieved between the numerical results and experimental data. The crater geometry under blast loading has been successfully predicted under different installation condition of the reinforced rebars.

8.2 Recommendations for Future Research

The recommendations for future research are summarized as follows:

1) To better capture the fracturing and fragmentation of debris under explosion in the FSI process, the semi-Lagrangian RKPM can be employed, where the naturally stabilized variationally consistent nodal integration (VC-NSNI) [119] can also be employed for structure modeling.

2) The proposed shock-algorithm relies on the Riemann solvers that are derived from the physics or phenomenological model. For complex solid plasticity models or non-Newtonian fluid models, such a constitutive model becomes too complicated to acquire such solver. Therefore, a data-driven approach for solving Riemann problems by given states at each side of discontinuity can be an alternative for the method proposed in this dissertation.

3) In the variational multiscale immersed framework for the FSI problem, the dynamical response of fine-scale fluid velocity is neglected, where the fine-scale solution is assumed to be in steady-state only. Such an assumption simplifies the variational multiscale formulation but could yield the loss of accuracy when fluid experience strong turbulence of complicated wave propagation, where the high frequency is dominating in the time domain. A space-time approach for the fine-scale fluid solution may be considered for future work.

Appendix

Appendix A. VMIM for Heterogeneous Material Diffusion Problems

The variational multiscale immersed method (VMIM) can also be applied for other type of BVPs involving heterogeneous material, such as heterogeneous material diffusion problems where the strong form is given by following equations:

$$-k(\boldsymbol{x})\boldsymbol{\nabla}^2 u = \mathcal{L}(u) = s, \quad \forall \boldsymbol{x} \in \Omega \tag{286}$$

$$k\boldsymbol{\nabla} u \cdot \boldsymbol{n}^0 = q, \quad \forall \boldsymbol{x} \in \partial\Omega_N \tag{287}$$

$$u = g, \quad \forall \boldsymbol{x} \in \partial\Omega_D \tag{288}$$

$$\boldsymbol{f}(\boldsymbol{x}) = k\boldsymbol{\nabla} u \tag{289}$$

where u is the scalar variable field. $\boldsymbol{\nabla}^2 = \boldsymbol{\nabla} \cdot \boldsymbol{\nabla}$ is the divergence operator, s is the source term, and q and g denote the prescribed flux and essential boundary condition on $\partial\Omega_N$ and $\partial\Omega_D$, respectively. $\boldsymbol{f}$ is the flux vector. Similarly, the source term s and diffusivity k can exhibit discontinuities across different subdomains, but have smooth distribution in each subdomain Ω^+ and $\Omega^-\backslash\Omega^+$ respectively, given as

$$s = \begin{cases} s^+ & \text{in} \quad \Omega^+ \\ s^- & \text{in} \quad \Omega^-\backslash\Omega^+ \end{cases}, \qquad k = \begin{cases} k^+ & \text{in} \quad \Omega^+ \\ k^- & \text{in} \quad \Omega^-\backslash\Omega^+ \end{cases} \tag{290}$$

where k^+ and k^- are scalar diffusivity corresponding to different materials in Ω^+ and $\Omega^-\backslash\Omega^+$ respectively, and s^+ and s^- are source term corresponding to different materials in Ω^+ and $\Omega^-\backslash\Omega^+$ respectively. In this study, k^+ and k^- are taken as material constants. The following conditions apply on the interface Γ:

$$[\![u]\!] = 0, \quad \forall \boldsymbol{x} \in \Gamma \tag{291}$$

$$[\![\boldsymbol{f} \cdot \boldsymbol{n}]\!] = 0, \quad \forall \boldsymbol{x} \in \Gamma \tag{292}$$

and the divergence operator can also be defined for each material subdomains:

$$\mathcal{L}^-(\cdot) = -k^- \boldsymbol{\nabla}^2 u^-, \ \ \mathcal{L}^+(\cdot) = -k^+ \boldsymbol{\nabla}^2 u^+ \tag{293}$$

for the convenience of the derivation. The immersed weak form for the given heterogeneous material diffusion problem reads: find $(u^+, u^-) \in \mathcal{U}^+ \times \mathcal{U}^-, u^+ = u^-$ on Ω^+, such that:

$$\begin{aligned} &(\boldsymbol{\nabla}\delta u^-, k^- \boldsymbol{\nabla} \delta u^-)_{\Omega^-} + (\boldsymbol{\nabla}\delta u^+, [\![k]\!] \boldsymbol{\nabla} \delta u^+)_{\Omega^+} \\ &\qquad = (\delta u^-, s^-)_{\Omega^-} + (\delta u^+, [\![s]\!])_{\Omega^-} + (\delta u^-, q)_{\partial\Omega_N^-} \end{aligned} \tag{294}$$

$\forall (\delta u^+, \delta u^-) \in \mathcal{V}^+ \times \mathcal{V}^-$. The function space $\mathcal{U}^+$, $\mathcal{U}^-$, $\mathcal{V}^+$, and $\mathcal{V}^-$ are given as

$$\begin{aligned} \mathcal{U}^+ &\equiv \{u | u \in H^1(\Omega^+)\} \\ \mathcal{V}^+ &\equiv \{u | u \in H^1(\Omega^+)\} \\ \mathcal{U}^- &\equiv \{u | u \in H^1(\Omega^-), u = g \text{ on } \partial\Omega_D^-\} \\ \mathcal{V}^- &\equiv \{u | u \in H^1(\Omega^-), u = 0 \text{ on } \partial\Omega_D^-\} \end{aligned} \tag{295}$$

A.1 VMIM via Residual-Free Bubble Method

By employing the procedure shown in Chapter 5.2.2, the VMIM weak form with given bubble function for fine-scale solution $\hat{u}^-$ reads: find $(\bar{u}^+, \bar{u}^-, \lambda) \in \bar{\mathcal{U}}^+ \times \bar{\mathcal{U}}^- \times \mathcal{W}$, such that

$$\begin{aligned}(\boldsymbol{\nabla}\delta\bar{u}^-, k^-\boldsymbol{\nabla}\delta\bar{u}^-)_{\Omega^-} + (\boldsymbol{\nabla}\delta\bar{u}^-, k^-\boldsymbol{\nabla}\hat{u}^-)_{\Omega^-} - (\delta\bar{u}^-, \lambda)_{\Omega^+} \\ = (\delta\bar{u}^-, s^-)_{\Omega^-} + (\delta\bar{u}^-, t)_{\partial\Omega_N^-}\end{aligned} \tag{296}$$

$$(\boldsymbol{\nabla}\delta\bar{u}^+, [\![k]\!]\boldsymbol{\nabla}\delta\bar{u}^+)_{\Omega^+} + (\delta\bar{u}^+, \lambda)_{\Omega^+} = (\delta\bar{u}^+, [\![s]\!])_{\Omega^+} \tag{297}$$

$$(\delta\lambda, \bar{u}^+)_{\Omega^+} - (\delta\lambda, \bar{u}^-)_{\Omega^+} - (\delta\lambda, \hat{u}^-)_{\Omega^+} = 0 \tag{298}$$

$\forall(\delta\bar{u}^+, \delta\bar{u}^-, \delta\lambda) \in \bar{\mathcal{V}}^+ \times \bar{\mathcal{V}}^- \times \mathcal{W}$. The function space $\bar{\mathcal{U}}^+$, $\bar{\mathcal{U}}^-$, $\bar{\mathcal{V}}^+$, and $\bar{\mathcal{V}}^-$ are given as

$$\begin{aligned}\bar{\mathcal{U}}^+ &\equiv \{u | u \in [H^1(\Omega^+)]^d\} \\ \bar{\mathcal{V}}^+ &\equiv \{w | w \in [H^1(\Omega^+)]^d\} \\ \bar{\mathcal{U}}^- &\equiv \{u | u \in [H^1(\Omega^-)]^d, u = g \text{ on } \partial\Omega_D^-\} \\ \bar{\mathcal{V}}^- &\equiv \{w | w \in [H^1(\Omega^-)]^d, w = 0 \text{ on } \partial\Omega_D^-\}\end{aligned} \tag{299}$$

The matrix equations of Eqns. (296) to (298) are given as follows (including the static condensation of fine-scale matrix equation)

$$\sum_J \left(\bar{K}_{IJ}^{-*}\bar{u}_J^- + \bar{G}_{IJ}^{-\lambda*}\lambda_J\right) = \bar{F}_I^{-*} \tag{300}$$

$$\sum_J \left(\bar{K}_{IJ}^{+}\bar{u}_J^+ + \bar{G}_{IJ}^{+\lambda}\lambda_J\right) = \bar{F}_I^{+} \tag{301}$$

$$\sum_J \left(\bar{G}_{IJ}^{+\lambda^T}\bar{u}_J^+ + G_{IJ}^{-\lambda*^T}\bar{u}_J^- + G_{IJ}^{\lambda\lambda*}\lambda_J\right) = F_I^{\lambda*} \tag{302}$$

for all I, where the matrices and vector in Eqns. (300) to (302) are expressed as

$$\bar{\boldsymbol{K}}^{-*} = \bar{\boldsymbol{K}}^- - \hat{\bar{\boldsymbol{K}}}^{-^T}\left(\hat{\bar{\boldsymbol{K}}}^-\right)^{-1}\hat{\bar{\boldsymbol{K}}}^- \tag{303}$$

$$\bar{\boldsymbol{G}}^{-\lambda*} = \hat{\bar{\boldsymbol{G}}}^{-^T}\left(\hat{\bar{\boldsymbol{K}}}^-\right)^{-1}\hat{\boldsymbol{G}}^{-\lambda} - \bar{\boldsymbol{G}}^{-\lambda} \tag{304}$$

$$\bar{\boldsymbol{F}}^{-*} = \bar{\boldsymbol{F}}^- - \hat{\bar{\boldsymbol{K}}}^{-^T}\left(\hat{\bar{\boldsymbol{K}}}^-\right)^{-1}\hat{\boldsymbol{F}}^- \tag{305}$$

$$\boldsymbol{G}^{\lambda\lambda*} = \widehat{\boldsymbol{G}}^{-\lambda^T}\left(\widehat{\widehat{\boldsymbol{K}}}^{-}\right)^{-1}\widehat{\boldsymbol{G}}^{-\lambda} \tag{306}$$

$$\boldsymbol{F}^{\lambda*} = \widehat{\boldsymbol{G}}^{-\lambda^T}\left(\widehat{\widehat{\boldsymbol{K}}}^{-}\right)^{-1}\widehat{\boldsymbol{F}}^{-} \tag{307}$$

and the evaluation for each matrix and vector follows Eqns. (308) to (317).

$$\overline{K}_{IJ}^{-} = \int_{\Omega^-} \overline{\boldsymbol{B}}_I^{-T} k^{-} \overline{\boldsymbol{B}}_J^{-} d\Omega \tag{308}$$

$$\overline{K}_{IJ}^{+} = \int_{\Omega^+} \overline{\boldsymbol{B}}_I^{+T} [\![k]\!] \overline{\boldsymbol{B}}_J^{+} d\Omega \tag{309}$$

$$\bar{G}_{IJ}^{-\lambda} = \int_{\Omega^+} \overline{N}_I^{-T} N_J^{\lambda} d\Omega \tag{310}$$

$$\bar{G}_{IJ}^{+\lambda} = \int_{\Omega^+} \overline{N}_I^{+T} N_J^{\lambda} d\Omega \tag{311}$$

$$\overline{F}_I^{-} = \int_{\Omega^-} \overline{N}_I^{-T} s^{-} d\Omega + \int_{\partial\Omega_N^-} \overline{N}_I^{-T} q d\Gamma \tag{312}$$

$$\overline{F}_I^{+} = \int_{\Omega^+} \overline{N}_I^{+T} [\![s]\!] d\Omega \tag{313}$$

$$\widehat{\overline{K}}_{IJ}^{-} = \int_{\Omega^-} \widehat{\boldsymbol{B}}_I^{-T} k^{-} \widehat{\boldsymbol{B}}_J^{-} d\Omega \tag{314}$$

$$\hat{G}_{IJ}^{\lambda-} = \int_{\Omega^+} \widehat{N}_I^{-T} N_J^{\lambda} d\Omega \tag{315}$$

$$\widehat{\overline{K}}_{IJ}^{-} = \int_{\Omega^-} \widehat{\boldsymbol{B}}_J^{-T} k^{-} \overline{\boldsymbol{B}}_I^{-} d\Omega \tag{316}$$

$$\hat{F}_I^{-} = \int_{\Omega^-} \widehat{N}_I^{-T} s^{-} d\Omega \tag{317}$$

where $\bar{N}_I^-$, $\bar{N}_I^+$ and N_I^λ are I-th shape function for $\bar{u}^-$, $\bar{u}^+$ and λ, respectively. $\widehat{N}_I^-$ us the fine-scale bubble function for node I. $\bar{\boldsymbol{B}}_I^-$ and $\bar{\boldsymbol{B}}_I^+$ are the gradient matrices for $\bar{N}_I^-$ and $\bar{N}_I^+$, respectively. For example, $\bar{\boldsymbol{B}}_I^-$ and $\bar{\boldsymbol{B}}_I^+$ in the three-dimensional case are expressed as

$$\bar{\boldsymbol{B}}_I^- = \begin{bmatrix} \bar{N}_{I,1}^- \\ \bar{N}_{I,2}^- \\ \bar{N}_{I,3}^- \end{bmatrix}, \quad \bar{\boldsymbol{B}}_I^+ = \begin{bmatrix} \bar{N}_{I,1}^+ \\ \bar{N}_{I,2}^+ \\ \bar{N}_{I,3}^+ \end{bmatrix} \tag{318}$$

and $\bar{\boldsymbol{B}}_{I,i}^-$ is the i-th derivative of $\bar{\boldsymbol{B}}_I^-$

$$\bar{\boldsymbol{B}}_{I,i}^- = \frac{\partial}{\partial x_i} \begin{bmatrix} \bar{N}_{I,1}^- \\ \bar{N}_{I,2}^- \\ \bar{N}_{I,3}^- \end{bmatrix} \tag{319}$$

The geometric vector $\boldsymbol{\eta}_{i=1,2,3}$ for the diffusion problem is expressed as

$$\begin{aligned} \boldsymbol{\eta}_1 &= [1 \quad 0 \quad 0] \\ \boldsymbol{\eta}_2 &= [0 \quad 1 \quad 0] \\ \boldsymbol{\eta}_3 &= [0 \quad 0 \quad 1] \end{aligned} \tag{320}$$

A.2 VMIM via Approximated Fine-Scale Solution

By employing the derivation of approximated fine-scale solution shown in Chapter 5.2.3, the VMIM weak form for heterogeneous material diffusion problem reads: find $(\bar{u}^+, \bar{u}^-, \lambda) \in \bar{\mathcal{U}}^+ \times \bar{\mathcal{U}}^- \times \mathcal{W}$, such that

$$\begin{aligned} &(\boldsymbol{\nabla}\delta\bar{u}^-, k^-\boldsymbol{\nabla}\delta\bar{u}^-)_{\Omega^-} - \left(\hat{\tau}^-\mathcal{L}^-(\delta\bar{u}^-), \mathcal{L}^-(\bar{u}^-)\right)_{\Omega^-} + (\mathcal{L}^-(\delta\bar{u}^-), \hat{\tau}^-\lambda)_{\Omega^-} \\ &\qquad + (\delta\bar{u}^-, \lambda)_{\Omega^+} \\ &\qquad = (\delta\bar{u}^-, s^-)_{\Omega^-} + (\delta\bar{u}^-, q)_{\partial\Omega_{\mathrm{N}}^-} - (\mathcal{L}^-(\delta\bar{u}^-), \hat{\tau}^- s^-)_{\Omega^-} \end{aligned} \tag{321}$$

$$(\boldsymbol{\nabla}\delta\bar{u}^+, [\![k]\!]\boldsymbol{\nabla}\delta\bar{u}^+)_{\Omega^+} + (\delta\bar{u}^+, \lambda)_{\Omega^+} = (\delta\bar{u}^+, [\![s]\!])_{\Omega^+} \tag{322}$$

$$(\delta\lambda, [\![\bar{u}]\!])_{\Omega^+} - \left(\delta\lambda, \hat{\tau}^-\mathcal{L}^-(\bar{u}^-)\right)_{\Omega^+} + (\delta\lambda, \hat{\tau}^-\lambda)_{\Omega^+} = (\delta\lambda, -\hat{\tau}^- s^-)_{\Omega^+} \tag{323}$$

$\forall(\delta\bar{u}^+, \delta\bar{u}^-, \delta\lambda) \in \bar{\mathcal{V}}^+ \times \bar{\mathcal{V}}^- \times \mathcal{W}$. The function space $\bar{\mathcal{U}}^+$, $\bar{\mathcal{U}}^-$, $\bar{\mathcal{V}}^+$, and $\bar{\mathcal{V}}^-$ are given as

$$\begin{aligned}
\bar{\mathcal{U}}^+ &\equiv \{u | u \in [H^1(\Omega^+)]^d\} \\
\bar{\mathcal{V}}^+ &\equiv \{w | w \in [H^1(\Omega^+)]^d\} \\
\bar{\mathcal{U}}^- &\equiv \{u | u \in [H^2(\Omega^-)]^d, u = g \text{ on } \partial\Omega_D^-\} \\
\bar{\mathcal{V}}^- &\equiv \{w | w \in [H^2(\Omega^-)]^d, w = 0 \text{ on } \partial\Omega_D^-\}
\end{aligned} \tag{324}$$

The matrix equations are given as follows

$$\sum_J \begin{bmatrix} \bar{K}_{IJ}^- - \hat{K}_{IJ}^{-\tau} & 0 & -\hat{G}_{IJ}^{-\tau\lambda} - \bar{G}_{IJ}^{-\lambda} \\ 0 & \bar{K}_{IJ}^+ & \bar{G}_{IJ}^{+\lambda} \\ \hat{G}_{IJ}^{-\tau\lambda^T} - \bar{G}_{IJ}^{-\lambda^T} & \bar{G}_{IJ}^{+\lambda^T} & G_{IJ}^{\lambda\tau\lambda} \end{bmatrix} \begin{Bmatrix} \bar{u}_J^- \\ \bar{u}_J^- \\ \lambda_J \end{Bmatrix} = \begin{bmatrix} \bar{F}_I^- + \hat{F}_I^\tau \\ \bar{F}_I^+ \\ -F_I^{\tau\lambda} \end{bmatrix}, \qquad \forall I \tag{325}$$

where $\bar{K}_{IJ}^-$, $\bar{K}_{IJ}^+$, $\bar{G}_{IJ}^{-\lambda}$, $\bar{G}_{IJ}^{+\lambda}$, $\bar{F}_I^-$, and $\bar{F}_I^+$ are given in Eqns. (308) to (313) and $\hat{K}_{IJ}^{-\tau}$, $\hat{G}_{IJ}^{-\tau\lambda}$, $G_{IJ}^{\lambda\tau\lambda}$, $\hat{F}_I^\tau$ and $F_I^{\tau\lambda}$ are given as follows

$$\hat{K}_{IJ}^{-\tau} = \int_{\Omega^-} \sum_{i=1}^{d} \left(\boldsymbol{\eta}_i k^- \bar{\boldsymbol{B}}_{I,i}^-\right)^T \hat{\tau}^- \sum_{j=1}^{d} \left(\boldsymbol{\eta}_j k^- \bar{\boldsymbol{B}}_{J,j}^-\right) d\Omega \tag{326}$$

$$\hat{G}_{IJ}^{-\tau\lambda} = \int_{\Omega^-} \sum_{i=1}^{d} \left(\boldsymbol{\eta}_i k^- \bar{\boldsymbol{B}}_{I,i}^-\right)^T \hat{\tau}^- N_J^\lambda d\Omega \tag{327}$$

$$G_{IJ}^{\lambda\tau\lambda} = \int_{\Omega^+} N_I^{\lambda^T} \hat{\tau}^- N_J^\lambda d\Omega \tag{328}$$

$$\hat{F}_I^\tau = \int_{\Omega^+} \sum_{i=1}^{d} \left(\boldsymbol{\eta}_i k^- \bar{\boldsymbol{B}}_{I,i}^-\right)^T \hat{\tau}^- s^- d\Omega \tag{329}$$

$$F_I^{\tau\lambda} = \int_{\Omega^+} \hat{\tau}^- N_J^{\lambda^T} s^- d\Omega \tag{330}$$

The algebraic operator $\hat{\tau}^-$ can also be derived from one-dimensional analysis (214) to (219) as:

$$\hat{\tau}^- = \frac{1}{12} \frac{h^2}{k^-} \tag{331}$$

References

[1] X. Wang, Y. Zheng, H. Liu, Z. Shen, Y. Hu, W. Li, Y. Gao and C. Guo, "Numerical study of the mechanism of explosive/impact welding using smoothed particle hydrodynamics method," *Materials & Design*, vol. 35, pp. 210-219, 2012.

[2] T. Rabczuk and J. Eibl, "Modelling dynamic failure of concrete with meshfree methods," *International Journal of Impact Engineering*, vol. 32, no. 11, pp. 1878-1897, 2006.

[3] J.-S. Chen, C. Pan, C.-T. Wu and W. K. Liu, "Reproducing kernel particle methods for large deformation analysis of non-linear structures," *Computer Methods in Applied Mechanics and Engineering*, vol. 139, no. 1-4, pp. 195-227, 1996.

[4] W. K. Liu, S. Hao, T. Belytschko, S. Li and C. T. Chang, "Multiple scale meshfree methods for damage fracture and localization," *Computational materials science*, vol. 16, no. 1-4, pp. 197-205, 1999.

[5] S. Li, W. Hao and W. K. Liu, "Mesh-free simulations of shear banding in large deformation," *International Journal of solids and structures*, vol. 37, no. 48-50, pp. 7185-7206, 2000.

[6] W. K. Liu, S. Jun and Y. F. Zhang, "Reproducing kernel particle methods," *International Journal for Numerical Methods in Fluids*, vol. 20, no. 8-9, pp. 1081-1106, 1995.

[7] W. K. Liu, S. Jun, D. T. Sihling, Y. Chen and W. Hao, "Multiresolution reproducing kernel particle method for computational fluid dynamics," *International Journal for Numerical Methods in Fluids*, vol. 24, no. 12, pp. 1391-1415, 1997.

[8] J.-S. Chen, M. Hillman and S.-W. Chi, "Meshfree methods: progress made after 20 years," *Journal of Engineering Mechanics*, vol. 143, no. 4, p. 04017001, 2017.

[9] T. J. Hughes, "Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods," *Computer Methods in Applied Mechanics and Engineering*, vol. 127, no. 1-4, pp. 387-401, 1995.

[10] T. J. Hughes, L. P. Franca and G. M. Hulbert, "A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 73, no. 2, pp. 173-189, 1989.

[11] J. von Neumann and R. D. Richtmyer, "A method for the numerical calculation of hydrodynamic shocks," *Journal of Applied Physics*, vol. 21, no. 3, pp. 232-237, 1950.

[12] M. L. Wilkins, "Use of artificial viscosity in multidimensional fluid dynamic calculations," *Journal of Computational Physics*, vol. 36, no. 3, pp. 281-303, 1980.

[13] S. K. Godunov, "A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics," *Matematicheskii Sbornik,* vol. 89, no. 3, pp. 271-306, 1959.

[14] V. V. Rusanov, "The calculation of the interaction of non-stationary shock waves with barriers," *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki,* vol. 1, no. 2, pp. 267-279, 1961.

[15] P. L. Roe, "Approximate Riemann solvers, parameter vectors, and difference schemes," *Journal of Computational Physics,* vol. 43, no. 2, pp. 357-372, 1981.

[16] A. Harten, P. D. Lax and B. van Leer, "On upstream differencing and Godunov-type schemes for hyperbolic conservation laws," *SIAM review,* vol. 25, no. 1, pp. 35-61, 1983.

[17] J. K. Dukowicz, "A general, non-iterative Riemann solver for Godunov's method," *Journal of Computational Physics,* vol. 61, no. 1, pp. 119-137, 1985.

[18] B. van Leer, "Towards the ultimate conservative difference scheme. IV. A new approach to numerical convection," *Journal of Computational Physics,* vol. 23, no. 3, pp. 276-299, 1977.

[19] B. van Leer, "Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method," *Journal of Computational Physics,* vol. 21, no. 1, pp. 101-136, 1979.

[20] B. van Leer, "Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second-order scheme," *Journal of Computational Physics,* vol. 14, no. 4, pp. 361-370, 1974.

[21] G. van Albada, B. van Leer and W. Roberts, "A comparative study of computational methods in cosmic gas dynamics," in *Upwind and High-Resolution Schemes,* Berlin, Heidelberg, Springer, 1997, pp. 95-103.

[22] A. Harten and S. Osher, "Uniformly high-order accurate nonoscillatory schemes. I," *SIAM Journal on Numerical Analysis,* vol. 24, no. 2, pp. 279-309, 1987.

[23] A. Harten, B. Engquist, S. Osher and S. R. Chakravarthy, "Uniformly high order accurate essentially non-oscillatory schemes, III," *Journal of Computational Physics,* vol. 71, no. 2, pp. 231-303, 1987.

[24] C.-W. Shu and S. Osher, "Efficient implementation of essentially non-oscillatory shock-capturing schemes," *Journal of Computational Physics,* vol. 77, no. 2, pp. 439-471, 1988.

[25] I. Christie, D. F. Griffiths, A. R. Mitchell and O. C. Zienkiewicz, "Finite element methods for second order differential equations with significant first derivatives," *International Journal for Numerical Methods in Engineering,* vol. 10, no. 6, pp. 1389-1396, 1976.

[26] T. Hughes and A. Brooks, "A multidimensional upwinding scheme with no crosswind diffusion," *Finite Element Methods for Convection Dominated Flows,* vol. 34, 1979.

[27] A. N. Brooks and T. J. Hughes, "Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations," *Computer Methods in Applied Mechanics and Engineering,* vol. 32, no. 1-3, pp. 199-259, 1982.

[28] T. J. Hughes, "A theoretical framework for Petrov-Galerkin methods with discontinuous weighting functions: Application to the streamline-upwind procedure," *Finite Elements in Fluids,* vol. IV, pp. 47-65, 1982.

[29] T. J. Hughes, M. Mallet and M. Akira, "A new finite element formulation for computational fluid dynamics: II. Beyond SUPG," *Computer Methods in Applied Mechanics and Engineering,* vol. 54, no. 3, pp. 341-355, 1986.

[30] T. J. Hughes and T. Tezduyar, "Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations," *Computer Methods in Applied Mechanics and Engineering,* vol. 45, no. 1-3, pp. 217-284, 1984.

[31] T. E. Tezduyar, M. Senga and D. Vicker, "Computation of inviscid supersonic flows around cylinders and spheres with the SUPG formulation and YZβ shock-capturing," *Computational Mechanics,* vol. 38, no. 4-5, pp. 469-481, 2006.

[32] T. E. Tezduyar and M. Senga, "SUPG finite element computation of inviscid supersonic flows with YZβ shock-capturing," *Computers & Fluids,* vol. 36, no. 1, pp. 147-159, 2007.

[33] J. Monaghan and R. A. Gingold, "Shock simulation by the particle method SPH," *Journal of Computational Physics,* vol. 52, no. 5, pp. 374-389, 1983.

[34] J. Monaghan, "SPH and Riemann solvers," *Journal of Computational Physics,* vol. 136, no. 2, pp. 298-307, 1997.

[35] S.-I. Inutsuka, "Reformulation of smoothed particle hydrodynamics with Riemann solver," *arXiv preprint astro-ph/0206401,* 2002.

[36] D. Hietel, K. Steiner and J. Struckmeier, "A finite-volume particle method for compressible flows," *Mathematical Models and Methods in Applied Sciences,* vol. 10, no. 09, pp. 1363-1382, 2000.

[37] R. Lohner, C. Sacco, E. Onate and S. Idelsohn, "A finite point method for compressible flow," *International Journal for Numerical Methods in Engineering,* vol. 53, no. 8, pp. 1765-1779, 2002.

[38] M. J. Roth, J.-S. Chen, T. R. Slawson and K. T. Danielson, "Stable and flux-conserved meshfree formulation to model shocks," *Computational Mechanics,* vol. 57, no. 5, pp. 773-792, 2016.

[39] M. J. Roth, J.-S. Chen, K. T. Danielson and T. R. Slawson, "Hydrodynamic meshfree method for high-rate solid dynamics using a Rankine-Hugoniot enhancement in a Riemann-SCNI framework," *International Journal for Numerical Methods in Engineering,* vol. 108, no. 12, pp. 1525-1549, 2016.

[40] J.-S. Chen, C.-T. Wu, S. Yoon and Y. You, "A stabilized conforming nodal integration for Galerkin mesh-free methods," *International Journal for Numerical Methods in Engineering,* vol. 50, no. 2, pp. 435-466, 2001.

[41] G. Zhou and M. Hillman, "A non-ordinary state-based Godunov-peridynamics formulation for strong shocks in solids," *Computational Particle Mechanics. In Press.*

[42] Y. Bazilevs, D. Kamensky, G. Moutsanidis and S. Shende, "Residual-based shock capturing in solids," *Computer Methods in Applied Mechanics and Engineering,* vol. 358, p. 112638, 2020.

[43] Y. Bazilevs, K. Kamran, G. Moutsanidis and D. a. O. E. Benson, "A new formulation for air-blast fluid--structure interaction using an immersed approach. Part I: basic methodology and FEM-based simulations," *Computational Mechanics,* vol. 60, no. 1, pp. 83-100, 2017.

[44] Y. Bazilevs, G. Moutsanidis, J. a. K. K. Bueno, D. Kamensky, M. C. Hillman, H. Gomez and J.-S. Chen, "A new formulation for air-blast fluid--structure interaction using an immersed approach: Part II—coupling of IGA and meshfree discretizations," *Computational Mechanics,* vol. 60, no. 1, pp. 101-116, 2017.

[45] D. Avesani, M. Dumbser and A. Bellin, "A new class of Moving-Least-Squares WENO-SPH schemes," *Journal of Computational Physics,* vol. 270, pp. 278-299, 2014.

[46] C.-T. Sun, P.-C. Guan, J. Jiang and O.-L. A. Kwok, "The weighted reconstruction of reproducing kernel particle method for one-dimensional shock wave problems," *Ocean Engineering,* vol. 149, pp. 325-340, 2018.

[47] C.-T. Sun, O.-L. A. Kwok, P.-C. Guan and W.-K. Shih, "Using Reproducing Kernel Particle Method for Shallow Water Problems," *Journal of Marine Science and Technology,* vol. 26, no. 3, pp. 431-440, 2018.

[48] J. Vila, "On particle weighted methods and smooth particle hydrodynamics," *Hyperbolic Problems: Theory, Numerics, Applications,* vol. 9, no. 2, pp. 161-209, 1999.

[49] E. Ortega, E. Onate and S. Idelsohn, "A finite point method for adaptive three-dimensional compressible flow calculations," *International Journal for Numerical Methods in Fluids,* vol. 60, no. 9, pp. 937-971, 2009.

[50] G.-S. Jiang and C.-W. Shu, "Efficient implementation of weighted ENO schemes," *Journal of Computational Physics,* vol. 126, no. 1, pp. 202-228, 1996.

[51] M. Pandolfi and D. D'Ambrosio, "Numerical instabilities in upwind methods: analysis and cures for the “carbuncle” phenomenon," *Journal of Computational Physics,* vol. 166, no. 2, pp. 271-301, 2001.

[52] J. J. Quirk, "A contribution to the great Riemann solver debate," in *Upwind and High-Resolution Schemes*, Dordrecht, Springer, 1997, pp. 550-569.

[53] J. L. Steger and R. Warming, "Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods," *Journal of Computational Physics,* vol. 40, no. 3, pp. 263-293, 1981.

[54] M.-S. Liou and C. J. Steffen Jr, "A new flux splitting scheme," *Journal of Computational Physics,* vol. 107, no. 1, pp. 23-39, 1993.

[55] G.-C. Zha and E. Bilgen, "Numerical solutions of Euler equations by using a new flux vector splitting scheme," *International Journal for Numerical Methods in Fluids,* vol. 17, no. 2, pp. 115-144, 1993.

[56] E. F. Toro and M. Vazquez-Cendon, "Flux splitting schemes for the Euler equations," *Computers & Fluids,* vol. 70, pp. 1-12, 2012.

[57] C.-T. Wu and M. Koishi, "A meshfree procedure for the microscopic analysis of particle-reinforced rubber compounds," *Interaction and Multiscale Mechanics,* vol. 2, no. 2, pp. 129-151, 2009.

[58] R. Batra, M. Porfiri and D. Spinello, "Treatment of material discontinuity in two meshless local Petrov-Galerkin (MLPG) formulations of axisymmetric transient heat conduction," *International Journal for Numerical Methods in Engineering,* vol. 61, no. 14, pp. 2461-2479, 2004.

[59] A. Duster, J. Parvizian, Z. Yang and E. Rank, "The finite cell method for three-dimensional problems of solid mechanics," *Computer Methods in Applied Mechanics and Engineering,* vol. 197, no. 45-48, pp. 3768-3782, 2008.

[60] I. Babuska, "The finite element method for elliptic equations with discontinuous coefficients," *Computing,* vol. 5, no. 3, pp. 207-213, 1970.

[61] C. Bernardi, Y. Maday and A. T. Patera, "Domain decomposition by the mortar element method," in *Asymptotic and Numerical Methods for Partial Differential Equations with Critical Parameters*, Vols. 269-286, Springer, 1993.

[62] I. Babuska, "The finite element method with Lagrangian multipliers," vol. 20, no. 3, pp. 179-192, 1973.

[63] J. E. Roberts and J.-M. Thomas, "Mixed and hybrid methods," 1991.

[64] E. a. M. N. a. W. B. Bechet, "A stable Lagrange multiplier space for stiff interface conditions within the extended finite element method," *International Journal for Numerical Methods in Engineering,* vol. 78, no. 8, pp. 931-954, 2009.

[65] R. Stenberg, "On some techniques for approximating boundary conditions in the finite element method," *Journal of Computational and applied Mathematics,* vol. 63, no. 1-3, pp. 139-148, 1995.

[66] P. Hansbo, "Nitsche's method for interface problems in computa-tional mechanics," *GAMM-Mitteilungen,* vol. 28, no. 2, pp. 183-206, 2005.

[67] S. Fernandez-Mendez and A. Huerta, "Imposing essential boundary conditions in mesh-free methods," *Computer Methods in Applied Mechanics and Engineering,* vol. 193, no. 12-14, pp. 1257-1275, 2004.

[68] H. M. Mourad, J. Dolbow and I. Harari, "A bubble-stabilized finite element method for Dirichlet constraints on embedded interfaces," *International Journal for Numerical Methods in Engineering*, vol. 69, no. 4, pp. 772-793, 2007.

[69] J. E. Dolbow and L. P. Franca, "Residual-free bubbles for embedded Dirichlet problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 45-48, pp. 3751-3759, 2008.

[70] T. J. Hughes, G. R. Feijoo, L. Mazzei and J.-B. Quincy, "The variational multiscale method—a paradigm for computational mechanics," *Computer Methods in Applied Mechanics and Engineering*, vol. 166, no. 1-2, pp. 3-24, 1998.

[71] F. Brezzi, M.-O. Bristeau, L. P. Franca, M. Mallet and G. Roge}, "A relationship between stabilized finite element methods and the Galerkin method with bubble functions," *Computer Methods in Applied Mechanics and Engineering*, vol. 96, no. 1, pp. 117-129, 1992.

[72] J. Janela, A. Lefebvre and B. Maury, "A penalty method for the simulation of fluid-rigid body interaction," *ESAIM: Proceedings*, vol. 14, pp. 115-123, 2005.

[73] P. a. F. R. a. D. E. Blanco, "A variational framework for fluid-solid interaction problems based on immersed domains: theoretical bases," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 25-28, pp. 2353-2371, 2008.

[74] R. Glowinski, T.-W. Pan, T. I. Hesla and D. D. Joseph, "A distributed Lagrange multiplier/fictitious domain method for particulate flows," *International Journal of Multiphase Flow*, vol. 25, no. 5, pp. 755-794, 1999.

[75] L. Zhang, A. Gerstenberger, X. Wang and W. K. Liu, "Immersed finite element method," *Computer Methods in Applied Mechanics and Engineering*, vol. 193, no. 21-22, pp. 2051-2067, 2004.

[76] W. K. Liu, S. Tang and S. Tang, "Mathematical foundations of the immersed finite element method," *Computational Mechanics*, vol. 39, no. 3, pp. 211-222, 2007.

[77] L. Cordes and B. Moran, "Treatment of material discontinuity in the element-free Galerkin method," *Computer Methods in Applied Mechanics and Engineering*, vol. 139, no. 1-4, pp. 75-89, 1996.

[78] Y. Krongauz and T. Belytschko, "EFG approximation with discontinuous derivatives," *International Journal for Numerical Methods in Engineering*, vol. 41, no. 7, pp. 1215-1233, 1998.

[79] T. Rabczuk, R. Gracie, J.-H. Song and T. Belytschko, "Immersed particle method for fluid--structure interaction," *International Journal for Numerical Methods in Engineering*, vol. 81, no. 1, pp. 48-71, 2010.

[80] S. Masuda and H. Noguchi, "Analysis of structure with material interface by meshfree method," *Computer Modeling in Engineering and Science*, vol. 11, no. 3, p. 131, 2006.

[81] Y.-C. Yoon and J.-H. Song, "Extended particle difference method for weak and strong discontinuity problems: part I. Derivation of the extended particle derivative approximation for the representation of weak and strong discontinuities," *Computational Mechanics,* vol. 53, no. 6, pp. 1087-1103, 2014.

[82] Y.-C. Yoon and J.-H. Song, "Extended particle difference method for weak and strong discontinuity problems: part II. Formulations and applications for various interfacial singularity problems," *Computational Mechanics,* vol. 53, no. 6, pp. 1105-1128, 2014.

[83] D. Wang, J.-S. Chen and L. Sun, "Homogenization of magnetostrictive particle-filled elastomers using an interface-enriched reproducing kernel particle method," *Finite Elements in Analysis and Design,* vol. 39, no. 8, pp. 765-782, 2003.

[84] D. Wang, Y. Sun and L. Li, "A discontinuous Galerkin meshfree modeling of material interface," *Computer Modeling in Engineering and Sciences,* vol. 45, no. 1, p. 57, 2009.

[85] C. Wu, Y. Guo and E. Askari, "Numerical modeling of composite solids using an immersed meshfree Galerkin method," *Composites Part B: Engineering,* vol. 45, no. 1, pp. 1397-1413, 2013.

[86] J. J. Koester and J.-S. Chen, "Conforming window functions for meshfree methods," *Computer Methods in Applied Mechanics and Engineering,* vol. 347, pp. 588-621, 2019.

[87] H. Lu, D. Wan Kim and W. Kam Liu, "Treatment of discontinuity in the reproducing kernel element method," *International Journal for Numerical Methods in Engineering,* vol. 63, no. 2, pp. 241-255, 2005.

[88] C. Wu, D. Wang and Y. Guo, "An immersed particle modeling technique for the three-dimensional large strain simulation of particulate-reinforced metal-matrix composites," *Applied Mathematical Modelling,* vol. 40, no. 4, pp. 2500-2513, 2016.

[89] J.-S. Chen and H.-P. Wang, "New boundary condition treatments in meshfree computation of contact problems," *Computer Methods in Applied Mechanics and Engineering,* vol. 187, no. 3-4, pp. 441-468, 2000.

[90] T. J. Hughes, W. K. Liu and T. K. Zimmermann, "Lagrangian-Eulerian finite element formulation for incompressible viscous flows," *Computer Methods in Applied Mechanics and Engineering,* vol. 29, no. 3, pp. 329-349, 1981.

[91] A. A. Johnson and T. E. Tezduyar, "Advanced mesh generation and update methods for 3D flow simulations," *Computational Mechanics,* vol. 23, no. 2, pp. 130-143, 1999.

[92] R. Glowinski, T.-W. Pan and J. Periaux, "A fictitious domain method for Dirichlet problem and applications," *Computer Methods in Applied Mechanics and Engineering,* vol. 111, no. 3-4, pp. 283-303, 1994.

[93] F. P. Baaijens, "A fictitious domain/mortar element method for fluid-structure interaction," *International Journal for Numerical Methods in Fluids,* vol. 35, no. 7, pp. 743-761, 2001.

[94] H. Wang and T. Belytschko, "Fluid-structure interaction by the discontinuous-Galerkin method for large deformations," *International Journal for Numerical Methods in Engineering,* vol. 77, no. 1, pp. 30-49, 2009.

[95] H. Wang, J. Chessa, W. K. Liu and T. Belytschko, "The immersed/fictitious element method for fluid-structure interaction: volumetric consistency, compressibility and thin members," *International Journal for Numerical Methods in Engineering,* vol. 74, no. 1, pp. 32-55, 2008.

[96] C. S. Peskin, "Flow patterns around heart valves: a numerical method," *Journal of Computational Physics,* vol. 10, no. 2, pp. 252-271, 1972.

[97] X. Wang and W. K. Liu, "Extended immersed boundary method using FEM and RKPM," *Computer Methods in Applied Mechanics and Engineering,* vol. 193, no. 12-14, pp. 305-1321, 2004.

[98] W. K. Liu, Y. Liu, D. Farrell, L. Zhang, X. S. a. F. Y. a. P. N. Wang, Y. Zhang, C. Bajaj, J. Lee, X. Chen and H. Hsua, "Immersed finite element method and its applications to biological systems," *Computer Methods in Applied Mechanics and Engineering,* vol. 195, no. 13-16, pp. 1722-1749, 2006.

[99] L. Zhang and M. Gay, "Immersed finite element method for fluid-structure interactions," *Journal of Fluids and Structures,* vol. 23, no. 6, pp. 839-857, 2007.

[100] D. Sulsky, Z. Chen and H. L. Schreyer, "A particle method for history-dependent materials," *Computer Methods in Applied Mechanics and Engineering,* vol. 118, no. 1-2, pp. 179-196, 1994.

[101] M. Uhlmann, "An immersed boundary method with direct forcing for the simulation of particulate flows," *Journal of Computational Physics,* vol. 209, no. 2, pp. 448-476, 2005.

[102] T. Belytschko and J. I. Lin, "A three-dimensional impact-penetration algorithm with erosion," *International Journal of Impact Engineering,* vol. 5, no. 1-4, pp. 111-127, 1987.

[103] P. Randles and L. D. Libersky, "Smoothed particle hydrodynamics: some recent improvements and applications," *Computer Methods in Applied Mechanics and Engineering,* vol. 139, no. 1-4, pp. 375-408, 1996.

[104] T. Rabczuk and J. Eibl, "Simulation of high velocity concrete fragmentation using SPH/MLSPH," *International Journal for Numerical Methods in Engineering,* vol. 56, no. 10, pp. 1421-1444, 2003.

[105] J.-S. Chen, S.-W. Chi, C.-H. Lee, S.-P. Lin, C. Marodon, M. J. Roth and T. Slawson, "A multiscale meshfree approach for modeling fragment penetration into ultra high-strength concrete," ERDC/GSL TR-11-35, U.S. Army Engineer Research and Development Center, Vicksburg, MS, 2011.

[106] S.-W. Chi, C.-H. Lee, J.-S. Chen and P.-C. Guan, "A level set enhanced natural kernel contact algorithm for impact and penetration modeling," *International Journal for Numerical Methods in Engineering,* vol. 102, no. 3-4, pp. 839-866, 2015.

[107] J. A. Sherburn, M. J. Roth, J. Chen and M. Hillman, "Meshfree modeling of concrete slab perforation using a reproducing kernel particle impact and penetration formulation," *International Journal of Impact Engineering,* vol. 86, pp. 96-110, 2015.

[108] R. W. Macek and S. A. Silling, "Peridynamics via finite element analysis," *Finite Elements in Analysis and Design,* vol. 43, no. 15, pp. 1169-1178, 2007.

[109] X. Lai, B. Ren, H. Fan, S. Li, C. Wu, R. A. Regueiro and L. Liu, "Peridynamics simulations of geomaterial fragmentation by impulse loads," *International Journal for Numerical and Analytical Methods in Geomechanics,* vol. 39, no. 12, pp. 1304-1330, 2015.

[110] B. Ren, H. Fan, G. L. Bergel, R. A. Regueiro, X. Lai and S. Li, "A peridynamics-SPH coupling approach to simulate soil fragmentation induced by shock waves," *Computational Mechanics,* vol. 55, no. 2, pp. 287-302, 2015.

[111] S. Attaway, M. Heinstein and J. Swegle, "Coupling of smooth particle hydrodynamics with the finite element method," *Nuclear Engineering and Design,* vol. 150, no. 2-3, pp. 199-205, 1994.

[112] G. R. Johnson, "Linking of Lagrangian particle methods to standard finite element methods for high velocity impact computations," *Nuclear Engineering and Design,* vol. 150, no. 2-3, pp. 265-274, 1994.

[113] P.-C. Guan, S.-W. Chi, J.-S. Chen, T. Slawson and M. J. Roth, "Semi-Lagrangian reproducing kernel particle method for fragment-impact problems," *International Journal of Impact Engineering,* vol. 38, no. 12, pp. 1033-1047, 2011.

[114] W. K. Liu and Y. Chen, "Wavelet and multiple scale reproducing kernel methods," *International Journal for Numerical Methods in Fluids,* vol. 21, no. 10, pp. 901-931, 1995.

[115] W. K. Liu, Y. Chen, R. A. Uras and C. T. Chang, "Generalized multiple scale reproducing kernel particle methods," *Computer Methods in Applied Mechanics and Engineering,* vol. 139, no. 1-4, pp. 91-157, 1996.

[116] M. O. Ruter and J.-S. Chen, "An enhanced-strain error estimator for Galerkin meshfree methods based on stabilized conforming nodal integration," *Computers & Mathematics with Applications,* vol. 74, no. 9, pp. 2144-2171, 2017.

[117] Y. You, J.-S. Chen and H. Lu, "Filters, reproducing kernel, and adaptive meshfree method," *Computational Mechanics,* vol. 31, no. 3-4, pp. 316-326, 2003.

[118] J.-S. Chen, M. Hillman and M. Ruter, "An arbitrary order variationally consistent integration for Galerkin meshfree methods," *International Journal for Numerical Methods in Engineering,* vol. 95, no. 5, pp. 387-418, 2013.

[119] M. Hillman and J.-S. Chen, "An accelerated, convergent, and stable nodal integration in Galerkin meshfree methods for linear and nonlinear mechanics," *International Journal for Numerical Methods in Engineering,* vol. 107, no. 7, pp. 603-630, 2016.

[120] J.-S. Chen and D. Wang, "A constrained reproducing kernel particle formulation for shear deformable shell in Cartesian coordinates," *International Journal for Numerical Methods in Engineering,* vol. 68, no. 2, pp. 151-172, 2006.

[121] N. H. Kim, K. K. Choi, J.-S. Chen and M. E. Botkin, "Meshfree analysis and design sensitivity analysis for shell structures," *International Journal for Numerical Methods in Engineering,* vol. 53, no. 9, pp. 2087-2116, 2002.

[122] J.-S. Chen, C. Pan, C. Roque and H.-P. Wang, "A Lagrangian reproducing kernel particle method for metal forming analysis," *Computational Mechanics,* vol. 22, no. 3, pp. 289-307, 1998.

[123] J.-S. Chen, H.-P. Wang, S. Yoon and Y. You, "Some recent improvements in meshfree methods for incompressible finite elasticity boundary value problems with contact," *Computational Mechanics,* vol. 25, no. 2-3, pp. 137-156, 2000.

[124] H.-P. Wang, C.-T. Wu and J.-S. Chen, "A reproducing kernel smooth contact formulation for metal forming simulations," *Computational Mechanics,* vol. 54, no. 1, pp. 151-169, 2014.

[125] J.-S. Chen, R. R. Basava, Y. Zhang, R. Csapo, V. Malis, U. Sinha, J. Hodgson and S. Sinha, "Pixel-based meshfree modelling of skeletal muscles," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization,* vol. 4, no. 2, pp. 73-85, 2016.

[126] H. Wei, J.-S. Chen and M. Hillman, "A stabilized nodally integrated meshfree formulation for fully coupled hydro-mechanical analysis of fluid-saturated porous media," *Computers & Fluids,* vol. 141, pp. 105-115, 2016.

[127] H. Wei, J.-S. Chen, F. Beckwith and J. Baek, "A naturally stabilized semi-Lagrangian meshfree formulation for multiphase porous media with application to landslide modeling," *Journal of Engineering Mechanics,* vol. under review, 2019.

[128] J.-S. Chen, C.-T. Wu and T. Belytschko, "Regularization of material instabilities by meshfree approximations with intrinsic length scales," *International Journal for Numerical Methods in Engineering,* vol. 47, no. 7, pp. 1303-1322, 2000.

[129] J.-S. Chen, X. Zhang and T. Belytschko, "An implicit gradient model by a reproducing kernel strain regularization in strain localization problems," *Computer Methods in Applied Mechanics and Engineering,* vol. 193, no. 27-29, pp. 2827-2844, 2004.

[130] H. Wei and J.-S. Chen, "A damage particle method for smeared modeling of brittle fracture," *International Journal for Multiscale Computational Engineering,* vol. 16, no. 4, pp. 303-324, 2018.

[131] T.-H. Huang, J.-S. Chen, H. Wei, M. J. Roth, J. A. Sherburn, J. E. Bishop, M. R. Tupek and E. H. Fang, "A MUSCL-SCNI approach for meshfree modeling of shock waves in fluids," *Computational Particle Mechanics, https://doi.org/10.1007/s40571-019-00248-x,* no. 7, p. 329–350, 2020.

[132] J. A. Sherburn, M. J. Roth, J. Chen and M. Hillman, "Meshfree modeling of concrete slab perforation using a reproducing kernel particle impact and penetration formulation," *International Journal of Impact Engineering,* vol. 86, pp. 96-110, 2015.

[133] S. Li and W. K. Liu, Meshfree particle methods, Berlin: Springer Science & Business Media, 2007.

[134] J. Dolbow and T. Belytschko, "Numerical integration of the Galerkin weak form in meshfree methods," *Computational Mechanics,* vol. 23, no. 3, pp. 219-230, 1999.

[135] Y. Lu, T. Belytschko and L. Gu, "A new implementation of the element free Galerkin method," *Computer Methods in Applied Mechanics and Engineering,* vol. 113, no. 3-4, pp. 397-414, 1994.

[136] J.-S. Chen, W. Hu, M. Puso, Y. Wu and X. Zhang, "Strain smoothing for stabilization and regularization of Galerkin meshfree methods," in *Meshfree Methods for Partial Differential Equations III*, Berlin, Springer, 2007, pp. 57-75.

[137] J. Chen and Y. Wu, "Stability in Lagrangian and semi-Lagrangian reproducing kernel discretizations using nodal integration in nonlinear solid mechanics," in *Computational Methods in Applied Sciences*, Dordrecht, 2007.

[138] J. Chen, W. Hu, M. Puso, Y. Wu and X. Zhang, "Strain smoothing for stabilization and regularization of Galerkin meshfree method," *Lecture Notes in Computational Science and Engineering,* vol. 57, pp. 57-76, 2006.

[139] J. Chen, M. Hillman and M. Rüter, "An arbitrary order variationally consistent integration for Galerkin meshfree methods," *International Journal for Numerical Methods in Engineering,* vol. 95, no. 5, pp. 387-418, 2013.

[140] E. F. Toro, Riemann solvers and numerical methods for fluid dynamics: a practical introduction, Berlin Heidelberg: Springer Science & Business Media, 2013.

[141] P. D. Lax, Hyperbolic systems of conservation laws and the mathematical theory of shock waves, New York: SIAM, 1973.

[142] J. D. Anderson and J. Wendt, Computational Fluid Dynamics, Berlin, Germany: Springer, 1995.

[143] L. Davison, Fundamentals of shock wave propagation in solids, Springer Science and Business Media, 2008.

[144] R. J. LeVeque, "Conservative Methods for Nonlinear Problems," in *Numerical Methods for Conservation Laws*, Berlin, Springer, 1990, pp. 122-135.

[145] J. Mandal and V. Panwar, "Robust HLL-type Riemann solver capable of resovling contact discontinuity," *Computers & Fluids,* vol. 63, pp. 148-164, 2012.

[146] S. Davis, "Simplified second-order Godunov-type methods," *SIAM Journal on Scientific and Statistical Computing,* vol. 9, no. 3, pp. 445-473, 1988.

[147] T. J. Barth and D. C. Jespersen, "The design and application of upwind schemes on unstructured meshes," NASA Ames Research Center, Moffett Field, 1989.

[148] S. Gottlieb and C.-W. Shu, "Total variation diminishing Runge-Kutta schemes," *Mathematics of Computation of the American Mathematical Society,* vol. 67, no. 221, pp. 73-85, 1998.

[149] H. Schardin, "High frequency cinematography in the shock tube," *The Journal of Photographic Science,* vol. 5, no. 2, pp. 17-19, 1957.

[150] S.-M. Chang and K.-S. Chang, "On the shock-vortex interaction in Schardin's problem," *Shock Waves,* vol. 10, no. 5, pp. 333-343, 2000.

[151] E. Lee, M. Finger and W. Collins, "JWL equation of state coefficients for high explosives," Lawrence Livermore National Laboratory (LLNL), Livermore, CA (United States), 1973.

[152] D. L. Chapman, "VI. On the rate of explosion in gases," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science,* vol. 47, no. 284, pp. 90-104, 1899.

[153] E. Jouguet, "Sur la propagation des reactions chimiques dans les gaz," *J. Maths. Pure Appl.*, vol. 7, p. 347, 1905.

[154] B. Dobratz and P. Crawford, "LLNL Explosives Handbook: Properties of Chemical Explosives and Explosive Simulants," Lawrence Livermore National Laboratory, Livermore , CA, 1985.

[155] W. Beltman and J. Shepherd, "Linear elastic response of tubes to internal detonation loading," *Journal of Sound and Vibration,* vol. 252, no. 4, pp. 617-655, 2002.

[156] C. L. Mader, Numerical modeling of explosives and propellants, CRC press, 2007.

[157] J. Nitsche, "Uber ein Variationsprinzip zur Losung von Dirichlet-Problemen bei Verwendung von Teilraumen, die keinen Randbedingungen unterworfen sind," *Abhandlungen aus dem mathematischen Seminar der Universitat Hamburg,* vol. 36, no. 1, pp. 9-15, 1971.

[158] T. J. Hughes, L. P. Franca and M. Balestra, "A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuska-Brezzi condition: A stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations," *Computer Methods in Applied Mechanics and Engineering,* vol. 59, no. 1, pp. 85-99, 1986.

[159] K. Nakshatrala, A. Masud and K. Hjelmstad, "On finite element formulations for nearly incompressible linear elasticity," *Computational Mechanics,* vol. 41, no. 4, pp. 547-561, 2008.

[160] M. Puso, J. Chen, E. Zywicz and W. Elmer, "Meshfree and finite element nodal integration methods," *International Journal for Numerical Methods in Engineering,* vol. 74, no. 3, pp. 416-446, 2008.

[161] M. Hillman and J.-S. Chen, "Nodally integrated implicit gradient reproducing kernel particle method for convection dominated problems," *Computer Methods in Applied Mechanics and Engineering,* vol. 299, pp. 381-400, 2016.

[162] Z. Li, T. Lin and X. Wu, "New Cartesian grid methods for interface problems using the finite element formulation," *Numerische Mathematik,* vol. 96, no. 1, pp. 61-98, 2003.

[163] M. L. Kachanov, B. Shafiro and I. Tsukrov, Handbook of elasticity solutions, Berlin, Germany: Springer Science & Business Media, 2013.

[164] C. G. Dengiz and K. Yildizli, "Effect of phase shape and size on the FEM analysis of indentation test of duplex stainless steel microstructures," *SN Applied Sciences,* vol. 2, no. 4, pp. 1-11, 2020.

[165] R. Codina, "Stabilized finite element approximation of transient incompressible flows using orthogonal subscales," *Computer Methods in Applied Mechanics and Engineering,* vol. 191, no. 39-40, pp. 4295-4321, 2002.

[166] X. Wang and L. T. Zhang, "Modified immersed finite element method for fully-coupled fluid-structure interactions," *Computer Methods in Applied Mechanics and Engineering,* vol. 267, pp. 150-169, 2013.

[167] J. T. Gretarsson, N. Kwatra and R. Fedkiw, "Numerically stable fluid-structure interactions between compressible flow and solid structures," *Journal of Computational Physics,* vol. 230, no. 8, pp. 3062-3084, 2011.

[168] M. Arienti, P. Hung, E. Morano and J. E. Shepherd, "A level set approach to Eulerian--Lagrangian coupling," *Journal of Computational Physics,* vol. 185, no. 1, pp. 213-251, 2003.

[169] V. Pasquariello, G. Hammerl, F. Orley, S. Hickel, C. Danowski, A. Popp, W. A. Wall and N. A. Adams, "A cut-cell finite volume-finite element coupling approach for fluid-structure interaction in compressible flow," *Journal of Computational Physics,* vol. 307, pp. 670-695, 2016.

[170] J. Giordano, G. Jourdan, Y. Burtschell, M. a. Z. D. Medale and L. Houas, "Shock wave impacts on deforming panel, an application of fluid-structure interaction," *Shock Waves,* vol. 14, no. 1-2, pp. 103-110, 2005.

[171] J. Wu and S. Chew, "Field performance and numerical modeling of multi-layer pavement system subject to blast load," *Construction and Building Materials,* vol. 52, pp. 177-188, 2014.

[172] G.-M. Ren, H. Wu, Q. Fang and X.-Z. Kong, "Parameters of Holmquist-Johnson-Cook model for high-strength concrete-like materials under projectile impact," *International Journal of Protective Structures,* vol. 8, no. 3, pp. 352-367, 2017.

[173] F. G. Friedlander, The wave equation on a curved space-time, Cambridge, UK: Cambridge university press, 1976.

[174] G. F. Kinney and K. J. Graham, Explosive shocks in air, Berlin, German: Springer Science & Business Media, 2013.