The Optimal Transportation Meshfree Method for General Fluid Flows and Strongly Coupled Fluid-Structure Interaction Problems

Thesis by

Feras Habbal

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

California Institute of Technology

Pasadena, California

2009

(Defended May 5, 2009)

"The way to get started is to quit talking and begin doing."

-Walt Disney

Acknowledgments

There are many people who, over the years, have provided me invaluable support and friendship. At Caltech, I would like to thank my advisor Michael Ortiz as well as Professors Ravichandran and Phillips for their support and guidance. Furthermore, the staff at Caltech has been wonderful in their support and assistance. Also, I would like to genuinely thank my mentor at JPL, Claus Hoff, for his support and guidance. During my time at Caltech, I have had the distinct pleasure of working with a number of individuals who ended up being friends beyond school-related tasks, including the members of the group: Lenny, Julian, Bo, Ben, Luigi, Marcial, Gabriela, Celia, and Daniel, among other. I would like to thank my Caltech DJ friends Bodhi, Juanse, Patrick, Signe, and Joe for the Lazy Sundays sessions. Also, I would like to thank my officemates and friends, Mike and Francesco.

Outside of Caltech, I would like to thank the Mammoth crew including Jasmine, Charles, Steve, Chris, Marcie, and Alice for their friendship and lasting memories of powder days on the steeps. Also, I would like to thank Alan for his true friendship and for making me relearn everything I thought I knew. Furthermore, I would like to thank Dione, Chuck, Graeme, Joe, and the extended DPI group for their friendship and enlightening intellectual discourse.

I would like to thank my parents, Drs. Habbal, and especially my sister, Lou, who has provided invaluable inspiration as an unyielding role model. I would also like to thank my old friends (Kayvon, Ev, Seema, Caroline, and Nina) for their support through tough times. Also, I would like to thank my violin teacher, Susan, who taught me how to break problems into small, conquerable pieces. Furthermore, I would like to thank the Rundells for their years of love, support, and yummy goodies.

Finally, I would be amiss not to thank my dear Rose and her wonderful family for their love, support, and kindness.

Abstract

This thesis develops a novel meshfree numerical method for simulating general fluid flows. Drawing from concepts in optimal mass transport theory and in combination with the notion of material point sampling and meshfree interpolation, the optimal transport meshfree (OTM) method provides a rigorous mathematical framework for numerically simulating three-dimensional general fluid flows with general, and possibly moving boundaries (as in fluid-structure interaction simulations). Specifically, the proposed OTM method generalizes the Benamou-Brenier differential formulation of optimal mass transportation problems that leads to a multifield variational characterization of general fluid flows including viscosity, equations of state, and general geometries and boundary conditions. With the use of material point sampling in conjunction with local max-entropy shape functions, the OTM method leads to a meshfree formulation bearing a number of salient features. Compared with other meshfree methods that face significant challenges to enforce essential boundary conditions as well as couple to other methods, such as the finite element method, the OTM method provides a new paradigm in meshfree methods. The OTM method is numerically validated by simulating the classical Riemann benchmark example for Euler flow. Furthermore, in order to highlight the ability of the OTM to simulate Navier-Stokes flows within general, moving three-dimensional domains, and naturally couple with finite elements, an illustrative strongly coupled FSI example is simulated. This illustrative FSI example, consisting of a gas-inflated sphere impacting the ground, is simulated as a toy model of the final phase of NASA's landing scheme devised for Mars missions, where a network of airbags are deployed to dissipate the energy of impact.

List of Figures

Contents

Chapter 1

Introduction

In moving toward the goal of predictive science capabilities, numerically simulating multiphysics problems has highlighted the limitations of current numerical methods. Of such multiphysics problems, fluid-structure interaction (FSI) problems represent a broad class of physical phenomena with fascinating applications in the natural sciences, medical fields, and engineering disciplines. FSI simulations highlight the challenges associated with coupling the distinct computational frameworks (Eulerian, Lagrangian) traditionally employed for numerical simulations of fluids and structures, respectively. To address these challenges, this thesis develops a new optimal transportation meshfree (OTM) method for numerical simulations of general fluid flows in a completely Lagrangian setting. The OTM method extends the concepts from optimal mass transportation theory to characterize general Newtonian fluid flows. Here, the OTM method inherits a rigorous mathematical framework that leads to geometrically exact mass density updates. Combined with the use of meshfree interpolation for the velocity field and the concept of material point sampling for the density field, the proposed OTM method provides a first-of-a-kind meshfree scheme for numerical simulations of fluid flows with general equations of state, boundary conditions, and initial conditions. Furthermore, the OTM framework facilitates the use of local maximum-entropy (LME) meshfree shape functions [4] for dynamic simulations, which has not been previously demonstrated. Combined with the use of LME shape functions, the proposed OTM method for fluids bears the distinct advantage over other meshfree schemes by trivially accommodating essential boundary and coupling with other Lagrangian methods (i.e. finite elements). In this sense, the OTM method provides an ideal numerical framework for coupling a meshfree fluid description with a traditional finite element model of the deformable structure for strongly coupled three-dimensional FSI simulations.

FSI simulations aim to predict the (weakly or strongly) coupled behavior of deformable structures interacting with dynamic fluid flows. Within a biomedical and biological setting, the coupled behavior of fluids and deformable structures is apparent across the continuum length scale in nature. Within the human body, the dynamics of cells, blood flowing in arteries, as well as the operation of organs represent examples of the remarkable functional interplay between deformable structures and the embedded fluidic environment. Furthermore, there are numerous engineering applications involving deformable structures interacting with fluids (both exterior and interior flows) including basic flight and nautical mechanisms, parachutes, airbags, inflatable structures for space applications, as well as hydraulic machinery. Within the area of space-based structures, one FSI application of interest includes the airbag-assisted landing scheme developed by NASA for missions landing on Mars. In this scheme, prior to impact a network of airbags were deployed to dissipate the energy of the landing process and protect the lander. In all of these examples, numerical simulations play an integral role for scientific understanding and engineering developments. This is especially true for applications where the cost, risk, or environment make direct experiments prohibitive.

While traditional numerical methods have directly impacted engineering and scientific developments, the state of the art in computational investigation of coupled multi-physics problems, enabled by perpetual increases in performance and accessibility of modern computers, has exposed the limitations of conventional numerical methods: the finite difference (FDM), finite volume (FVM), finite element (FEM) methods, as well as strategies for coupling these methods. These traditional numerical methods share a common reliance on a numerical mesh that, in turn, either limits the ability to handle very large deformation processes (as in the FEM) or complicates modeling general and possibly moving boundaries or free surfaces (as in the FVM). Within the field of computational mechanics, the search for new numerical methods that mitigate the restrictions imposed by traditional grid-based methods has recently spawned a renewed interest in particle and meshfree (or meshless) methods. In addition to facilitating numerical simulation that traditional schemes are incapable of modeling, the inherent challenges and the overwhelming time consumed by generating and maintaining a high quality numerical mesh highlights the appeal of gridless (i.e. particle or meshfree) schemes.

1.1 Traditional Numerical Strategies

Although both represent continuum materials, the numerical strategies for simulating solid and fluid dynamics have traditionally been formulated in distinct reference frames: the Lagrangian (fixed) and Eulerian (spatial) reference frame, respectively. Here, figure 1.1, depicts the distinct boundary representation for an Eulerian and Lagrangian mesh associated with the FVM and FEM, respectively. In adopting a Lagrangian reference frame, the evolution and dynamics of specific points on the body are evaluated and tracked. However, in the Lagrangian setting, the reliance on a deforming mesh inherently restricts the allowable motion. In contrast, the Eulerian reference frame provides the evolution and dynamics at a given point in space. In adopting the Eulerian approach, the numerical mesh does not move or distort which, contrary to Lagrangian schemes, frees the restriction on the allowable deformation within an element. This feature, however, limits the ability of an Eulerian scheme to inherently capture boundaries and moving surfaces.

The FEM, formulated in the Lagrangian (fixed) reference frame, has dominated the field of computational solid mechanics as the preferred strategy for numerical simulations of structures. While this method has had significant success applied to numerical simulations of structures, simulations of large deformation processes (i.e. metal forming, fluid

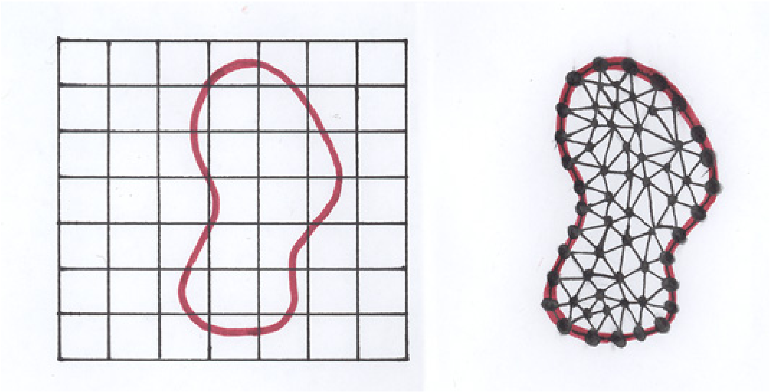

Figure 1.1: Domain and boundary representation in the finite volume method (right) and finite element method (left).

flows) as well as discrete phenomena (i.e. fracture, crack propagation) have exposed the main limitations of the FEM. These limitations are central to the reliance of the FEM on a deforming mesh. This reliance inherently restricts the allowable motion since, without remeshing, large deformations in the continuum lead to poor aspect ratios or inversion (i.e. mesh entanglement) of the deformed elements, leading to deteriorated accuracy or program termination. Furthermore, using the FEM for numerical simulations of fluid flows is largely limited due to the necessity of costly, and possibly not converging, mesh generation and update algorithms. A complete treatment of the FEM is detailed in [5].

The FVM method, posed in the Eulerian (spatial) reference frame, has had significant success for simulations of large deformation processes (i.e. fluid flows). The equations of motion for Newtonian fluid flows, formulated in the Eulerian reference frame, lead to the well known Navier-Stokes equations. When considering fluid flows using an easily integrated fixed domain, or control volume, the Navier-Stokes equations can be solved to a high order of accuracy using advanced FVMs. A detailed description of traditional computational fluid dynamics (CFD) methods employing FVMs can be found in the following references [6, 7]. Here, the main limitation of the FVM method is accurately capturing, as well as tracking, free boundaries or surfaces with a regular Eulerian mesh.

1.1.1 Numerical Strategies for FSI Simulations

Numerical solutions for strongly coupled FSI problems entail tremendous numerical challenges since the features of nonlinear solid mechanics, fluid mechanics, and their coupling must all be considered. Here, the coupling procedure is complicated by the fact that the computational models characterizing the fluid and solid responses tend to be in distinct reference frames. Given that neither the FVM nor the FEM can innately simulate both fluid and solid dynamics, several strategies for coupling the distinct Eulerian and Lagrangian framework have been proposed.

One of the most popular coupling strategies is the arbitrary-Lagrangian-Eulerian (ALE) method [8, 9, 10, 11]. The ALE method couples a fixed Eulerian mesh with a moving Lagrangian mesh through a buffered intermediate region where the conforming mesh moves according to a prescribed velocity. Here, the success of the ALE method relies on specifying the anticipated evolution of the mesh in the transition region. However, this demands some *a priori* knowledge of the solution which, in turn, restricts the types of analyses that can be simulated with the ALE method. Also, inherent in the ALE method is the requirement to update the mesh at each time step. This update procedure is not only computationally expensive but is further complicated by the lack of robust numerical algorithms providing a guaranteed converged solution for the mesh update for a general geometry. In connection with the concept of mesh adaption (r-adaption), Thoutireddy and Ortiz (2004) [12] proposed a variational arbitrary Lagrangian-Eulerian (VALE) method for determining the optimal mesh. Here, the distinguishing characteristic of the VALE method is that the variational principle simultaneously supplies the solution, the optimal mesh and the equilibrium shapes of the system for shape optimization problems.

In contrast with ALE methods, interface-capturing techniques do not require the meshes to conform at the interface. Sharp interface methods, such as the ghost fluid method [13] and the immersed boundary method [14] have been applied to formulate a mixed Lagrangian-Eulerian representation for FSI simulations [15, 16, 13, 17]. In the ghost fluid method, the discontinuities in the flow field (originating from the fluid-structure interface) are directly embedded in the solution by appropriately defining the ghost-fluid cell. Here, some of the finite volume cells are used as ghost fluid cells for enforcing the immersed moving wall boundary conditions. However, defining and updating the ghost fluid cells requires *a priori* specification of the anticipated structural deformation that restricts the allowable deformation and the types of applications that can be simulated.

The immersed boundary method (IB), proposed by Peskin [14], couples the fluid and structure response by distributing nodal forces and interpolating nodal velocities on the Eulerian and Lagrangian domains. Here, the IB method models the fluid using an Eulerian mesh and embeds the Lagrangian body as an elastic network of fibers. In the IB method, the fluid variables are evaluated on the Eulerian mesh using finite differences and subsequently interpolated onto the Lagrangian mesh by a smoothed approximation of the Dirac delta function. Here, the primary advantage of the IB method over the ALE method is that the fluid-structure interface is tracked automatically. This circumvents costly mesh generation and updating algorithms. However, one major obstacle of the IB formulation is the lack of an accurate solid body representation. Specifically, the IB formulation assumes that the structure is composed of one-dimensional beamlike structures that carry mass but occupy no volume within the fluid domain. This may be appropriate for fiberlike structures, but in general limits the application of the IB method from modeling more general structures such as membranes. Furthermore, the use of finite difference approximations places restrictions on the arrangement of computational nodes and their spacing, which limits the ability to resolve complex shapes and fluid boundaries.

While there are several traditional strategies for coupling the distinct numerical frameworks for FSI simulations, all of these methods represent a compromise. In contrast, a fully Lagrangian description for both the fluid and structural model, if robust, provides a convenient framework for simulating strongly coupled FSI applications. Here, the main obstacles to a fully Lagrangian description for FSI applications is the necessity to remesh the fluid domain. This complexity motivates the use of Lagrangian particle and meshfree methods as an appealing direction of research for numerical simulations of fluid flows. Particle and meshfree methods aim to possess the salient features of both the FEM and the FVM by accurately capturing complex or deforming boundaries, as in the FEM, without concern for mesh entanglement due to large deformations, as allowed for in the FVM. Bearing a Lagrangian framework, without the implicit restrictions of a Lagrangian mesh, furnishes an ideal framework for modeling the fluid and coupling the response to a structural description for FSI simulations. This goal provides the motivation for the OTM method developed in this thesis.

1.2 Numerical Solutions to Partial Differential Equations

Mathematically, the dynamic evolution of fluids, solids, and the coupled behavior amounts to solving a boundary value problem (BVP) consisting of a PDE (strong form), in a domain, $\boldsymbol{\Omega}$, with boundary, $\mathbf{d}\boldsymbol{\Omega}$, subject to essential boundary conditions (i.e. Dirichlet, displacement) on part or all of the domain, $\mathbf{d}\boldsymbol{\Omega}_\mathbf{1}$, and natural boundary conditions (i.e. Neumann, traction) on part or all of the domain, $\mathbf{d}\boldsymbol{\Omega}_\mathbf{2}$. A convenient strategy for numerically solving this BVP is to recast the strong form as a minimum variational principle (i.e. weak form). Furthermore, the BVP is represented as a minimum variational principle to provide a basis for comparing typical mesh-based methods with particle and meshfree methods for numerical simulations of the BVP.

1.2.1 Minimum Variational Problem

In general a BVP can be recast as a minimum variational problem by seeking a functional, $F[u]$, such that the original BVP is recovered by the Euler-Lagrange equations (i.e.

equivalent to the infimum of the desired functional). That is,

$$\begin{gathered}\inf F(\mathbf{u})\\ \mathbf{u}\in X\\ \mathbf{u}_{|d\Omega_1}=\bar{\mathbf{u}},\end{gathered}$$

where $X\ :\ \{\mathbf{u}\in H^1(\Omega)\ :\ \mathbf{u}|_{d\Omega_1}\}$ is the space of solutions, $\bar{u}$ are the prescribed essential boundary conditions, and $F[\mathbf{u}]$ is a functional of integrable type of the general form:

$$F[\mathbf{u}]=\int_{\Omega}f(\mathbf{x},\mathbf{u},\nabla\mathbf{u})d\mathbf{x}+\int_{d\Omega_2}g(\mathbf{x},\mathbf{u})d\mathbf{s}. \tag{1.1}$$

Then, consider variations of the functional in the path of integration that yield a minimum of the form,

$$\mathbf{v}=\mathbf{u}+\epsilon\eta,\quad \epsilon\in\mathbb{R}.$$

Here, the *test* (or weight) function, $\eta\in X$, is an *admissible* variation in the sense that requiring $\mathbf{v}=\bar{\mathbf{u}}$ on $d\Omega_1$ leads to the condition:

$$\epsilon\eta\big|_{d\Omega_1}=0.$$

In other words the test function is *admissible* if it satisfies the essential boundary conditions on $d\Omega_1$. In essence, the problem has been reduced to a one-dimensional minimization problem along a line. From the theory of calculus of variations [18], we note that $\mathbf{u}$ is a minimizer of F (F is Gâteau differentiable) if,

$$\frac{d}{d\epsilon}F(\mathbf{u}+\epsilon\eta)\big|_{\epsilon=0}=0\quad\forall\ \eta \text{ admissible}.$$

1.2.1.1 Principle of Virtual Work

Applying this definition to recast the functional, equation 1.1, into weak form yields:

$$\begin{aligned}
0 &= \frac{d}{d\epsilon}\left\{\int_\Omega f(\mathbf{x},\mathbf{u}+\epsilon\eta,\nabla(\mathbf{u}+\epsilon\eta))dx + \int_{d\Omega_2} g(\mathbf{x},\mathbf{u}+\epsilon\eta)ds\right\}\Big|_{\epsilon=0} \forall\,\eta \text{ admissible}\\
&= \left\{\int_\Omega \frac{df}{du_i}(\mathbf{x},\mathbf{u}+\epsilon\eta,\nabla(\mathbf{u}+\epsilon\eta))\eta_i + \frac{df}{du_{i,j}}(\mathbf{x},\mathbf{u}+\epsilon\eta,\nabla(\mathbf{u}+\epsilon\eta))\eta_{i,j}dx\right.\\
&\quad \left. + \int_{d\Omega_2}\frac{dg}{du_i}(\mathbf{x},\mathbf{u}+\epsilon\eta))\eta_i ds\right\}\Bigg|_{\epsilon=0}\\
&= \int_\Omega\left[\frac{df}{du_i}(\mathbf{x},\mathbf{u},\nabla\mathbf{u})\eta_i + \frac{df}{du_{i,j}}(\mathbf{x},\mathbf{u},\nabla\mathbf{u})\eta_{i,j}\right]dx + \int_{d\Omega_2}\frac{dg}{du_i}(\mathbf{x},\mathbf{u})\eta_i ds,
\end{aligned} \tag{1.2}$$

which is the *principle of virtual work.* Furthermore, integrating equation 1.2 by parts leads to the convenient local form:

$$\begin{aligned}
0 = &\int_\Omega\left[\frac{df}{du_i}(\mathbf{x},\mathbf{u},\nabla\mathbf{u}) - \frac{d}{dx_j}\frac{df}{du_{i,j}}(\mathbf{x},\mathbf{u},\nabla\mathbf{u})\right]\eta_i dx\\
&+ \int_{d\Omega_2}\left[\frac{dg}{du_i}(\mathbf{x},\mathbf{u})) + \frac{df}{du_{i,j}}\eta_j\right]\eta_i ds \quad \forall\,\eta \text{ admissible.}
\end{aligned} \tag{1.3}$$

Then, the Euler-Lagrange equations of motion are given by,

$$\begin{aligned}
\frac{df}{du_i}(\mathbf{x},\mathbf{u},\nabla\mathbf{u}) - \frac{d}{dx_j}\frac{df}{du_{i,j}}(\mathbf{x},\mathbf{u},\nabla\mathbf{u}) &= 0 \text{ in } \Omega\\
\frac{dg}{du_i}(\mathbf{x},\mathbf{u}) + \frac{df}{du_{i,j}}(\mathbf{x},\mathbf{u},\nabla\mathbf{u})\eta_i &= 0 \text{ on } d\Omega_2\\
\mathbf{u} &= \bar{\mathbf{u}} \text{ on } d\Omega_1.
\end{aligned} \tag{1.4}$$

As a consequence of this form, if $\mathbf{u}$ is a stationary point of equation 1.1 then it is a solution to the Euler-Lagrange equations.

1.2.1.2 Hamilton's Principle

As a brief illustration of this method consider Hamilton's principle. For a conservative mechanical system the action functional is given by,

$$S[\mathbf{q}(t)] = \int_{t_a}^{t_b} L(\mathbf{q}, \dot{\mathbf{q}})dt, \tag{1.5}$$

where $L(\mathbf{q}, \dot{\mathbf{q}})$ denotes the Lagrangian of the system, and $\mathbf{q} = (\mathbf{q}^1, \mathbf{q}^2, \ldots, \mathbf{q}^n)$ is a point in configuration space. Here, the Lagrangian is defined in terms of both the kinetic energy, K, and potential energy, E, of the system by,

$$L = K - E.$$

For a conservative mechanical system, the classical Lagrangian is given by,

$$L(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2}\dot{\mathbf{q}}^\top M \dot{\mathbf{q}} - V(\mathbf{q}),$$

where M is the (positive definite) symmetric mass matrix and V is the potential energy of the system. Then, Hamilton's principle states that the evolution (or trajectories) of the system, $q(t)$, between the states specified at times t_a and t_b are the curves that render the action functional stationary. That is,

$$\begin{aligned} \delta S &= 0 \\ &= \int_{t_a}^{t_b} \left[\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{q}}} - \frac{\partial L}{\partial \mathbf{q}}\right] \cdot \delta\mathbf{q} dt + \frac{\partial L}{\partial \dot{\mathbf{q}}} \cdot \delta\mathbf{q}\Big|_{t_a}^{t_b}, \end{aligned}$$

for all *admissible* variations, $\delta\mathbf{q}$ (i.e. variations that satisfy homogeneous conditions: $\delta\mathbf{q}_{t_a} = \delta\mathbf{q}_{t_b} = 0$). By directly applying the Euler-Lagrange equations,

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{q}}} - \frac{\partial L}{\partial \mathbf{q}} = 0, \tag{1.6}$$

the equations of motion for our Lagrangian are given by,

$$M\ddot{\mathbf{q}} = -\nabla V(\mathbf{q}).$$

As expected, the equations of motion resulting from directly applying the Euler-Lagrange equations to the Lagrangian of a conservative mechanical system leads to Newton's second law of motion (i.e. $\mathbf{F} = m\mathbf{a}$).

1.2.1.3 Lagrange-D'Alembert Principle

For mechanical systems with forcing and dissipation, the equations of motion can be obtained from the Lagrange-D'Alembert variational principle. In this setting we seek trajectories, $\mathbf{q}(t)$, such that

$$\delta \int_{t_a}^{t_b} L\left(\mathbf{q}(t), \dot{\mathbf{q}}(t)\right) dt + \int_{t_a}^{t_b} F\left(\mathbf{q}(t), \dot{\mathbf{q}}(t)\right) \cdot \delta\mathbf{q} dt = 0,$$

for all $\delta\mathbf{q}(t)$ admissible (i.e. $\delta\mathbf{q}(t_a) = \delta\mathbf{q}(t_b) = 0$). Here $F\left(\mathbf{q}(t), \dot{\mathbf{q}}(t)\right)$ is an arbitrary forcing function. Thus, the forcing or dissipation constraint is appended to the variation of the action directly. For fluids with viscosity the Lagrange-D'Alembert principle provides the basis for deriving the discrete equations of motion.

1.2.2 Numerical Interpolation and Approximation

Up to this point we have not addressed the character of the test functions, η (other than they are *admissible* by virtue of satisfying the essential boundary conditions) or how we plan on approximating the solution for u. To this end, consider discretizing the domain $\mathbf{\Omega}$ by N points (nodes) and identifying a basis for a finite-dimensional subspace X_h of our space of solutions X. Then, we can approximate the solution, $\mathbf{u}^h$, and the test function, η^h by considering linear combinations of basis functions, $\phi_I(\mathbf{x}), N_I(\mathbf{x}) \in X_h$ (not necessarily the same), of the form:

$$\begin{aligned}\mathbf{u}(\mathbf{x}) &\approx u^h(x) = \sum_{I=1}^{N} \mathbf{u}_I \phi_I(\mathbf{x}),\\ \eta(\mathbf{x}) &\approx \eta^h(\mathbf{x}) = \sum_{I=1}^{N} \eta_I N_I(\mathbf{x}),\end{aligned} \tag{1.7}$$

where $\mathbf{u}_I, \eta_I$ are the nodal degrees of freedom. Here, $\{N_I(\mathbf{x})\}$ are the test (or weight) functions and $\{\phi_I(\mathbf{x})\}$ are the basis (or trial) functions, commonly denoted as *shape functions*.

Paramount to choosing appropriate basis (shape) functions is the concept of nth order consistency for the approximation. In order to ensure that the interpolation is exact, we require that the given basis is capable of reproducing polynomials up to a required order n. For convergence in the weak form, equation 1.3, the basis functions must be capable of exactly reproducing polynomials of order greater than or equal to the order of the highest derivative appearing in the functional (consistency condition). In addition to guaranteeing consistency, convergence of a given method requires a notion of stability, which is related to the numerical integration technique employed in evaluating the weak form, equation 1.3. This thesis considers numerical solutions to second-order PDEs, which, when posed in the weak form, only posses first derivatives (i.e. by virtue of integration by parts). Thus, we require the shape functions to be first order consistent and reproduce linear fields exactly. This amounts to requiring

$$1 = \sum_{I=0}^{N} \phi_I(\mathbf{x}), \tag{1.8}$$

$$\mathbf{x} = \sum_{I=0}^{N} \mathbf{x}_I \phi_I(\mathbf{x}). \tag{1.9}$$

These requirements offer a wide range of flexibility for choosing the interpolation schemes for the test and trial functions. In particular, several classes of numerical methods, including the FEM, FVM, as well as meshfree and particle methods, are derived by specifying a choice for the functions N_I and ϕ_I.

FVM. Choosing $N_I(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_I)$ reduces equation 1.3 to a collocation scheme. In turn, the strong form is satisfied at individual points similar to a finite difference method. If we divide our domain into a set of nonoverlapping subdomains, $\{\mathbf{\Omega_k}\}$ and introduce a heavy-side function as the test function basis over the sub-domain, i.e.,

$$N_{\Omega_k}(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} \in \mathbf{\Omega_k}, \\ 0 & \text{else}, \end{cases}$$

then the weak form, equation 1.3 is localized to integrals over the subdomain, $\mathbf{\Omega_k}$. This is the starting point of the FVM, which is formulated by choosing the subdomains as easily integrated domains (i.e. squares for two dimensions, cubes for three dimensions) and converting the integrals over the body into surface integrals (via the divergence theorem). These integrals are subsequently evaluated by considering the fluxes in and out of the subdomain. A complete treatment of the FVM method for numerical simulations of fluid flows can be found in [6, 7].

FEM. Choosing the basis function for the test function to be the same as our shape function, i.e., $N_I = \phi_I$, leads to the Galerkin form (FEM). Here, the finite element shape functions are constructed by enforcing the conditions that the shape functions must be normalized functions and *natural* under restriction to face of the element cell. This implies that the FEM shape functions inherently contain the Kronecker-delta property, $\phi_I(\mathbf{x}_J) = \delta_{IJ}$, which is not generally true for meshfree shape functions. Choosing the basis for the test function as distinct, but associated to the trial functions by a transformation yields the Petrov-Galerkin form. A complete analysis of the FEM can be found in [5].

1.2.3 Numerical Integration

In the FEM, the weak form integrals in equation 1.3 must be numerically integrated. Here, typically a Gaussian quadrature rule is employed over the isoparametric elements. The general form for numerical integration rules is given by

$$\int_{\Omega} f(\mathbf{x}) d\mathbf{x} \approx \sum_{q=1}^{NQ} f(\theta_{\mathbf{q}}) W(\theta_{\mathbf{q}}).$$

Here, $W(\theta_{\mathbf{q}})$, is the numerical weight associated with the quadrature point θ_q, and NQ are the number of quadrature points in the integration rule. In contrast with traditional finite elements, numerical integration of the weak form, equation 1.3, for meshfree methods is not exact! This is not unexpected, since the lack of a mesh specifying the connections between related nodes complicates the concept of a geometrically well-defined area (for two dimensions) or volume (for three dimensions) for the integration weights.

1.3 Thesis Overview

This thesis develops a new meshfree method for simulating fluid flows with unprecedented mathematical formalism. Drawing on the theorems founded in optimal mass transportation theory, the OTM method extends the concepts and provides a rigorous formulation for numerical simulations of general fluid flows including Euler and Navier-Stokes flows. Here, the primary advantage of the proposed OTM method is tied directly to the inherently Lagrangian description. Combined with the use of LME shape functions, the OTM method couples with finite elements in a trivial manner. In turn, this provides a unique and ideal framework for numerical simulations of fluid-structure interaction (FSI) problems. Furthermore, the proposed OTM method naturally enables the use of LME shape functions for dynamic simulations, which has not been demonstrated in previous work.

Chapter 2 presents a review of commonly used particle and meshfree methods. As this area of numerical analysis has been recently rekindled, there is an almost equal plethora of ideas and approaches as there are applications. Indeed, the variety of methods is a testament to the fact that there is simply no single, overarching algorithm that is ideal for all applications. Depending on the physical phenomenon, length scales, boundary conditions, and accuracy requirements one or several methods may be appealing. All of the reviewed methods, as shown for the FVM and FEM, fall under a common framework of approximations in the sense of approximations in the weak form, equation 1.3. Chapter 3 provides an overview of the key concepts arising in optimal mass transportation theory that are extended for general fluid flows in chapter 4. Chapter 4 extends the concepts of optimal mass transportation theory to general fluid flows. Following the development of the scheme, the discrete forms and updates are provided. Chapter 5 provides numerical examples of the OTM method. Specifically, the OTM method is validated using a standard Riemann numerical benchmark problem for fluid flows to verify the ability of the OTM methods to capture shock waves. In order to showcase the OTM method's ability to seamlessly couple with the FEM as well as solve compressible gas dynamics in three dimensions with moving boundaries, we consider an illustrative three dimensional FSI simulation consisting of a gas-inflated thin-walled sphere undergoing large deformation while impacting the ground at high speed. Here, the balloon is modeled using membrane finite elements and the fluid is modeled as a compressible Newtonian gas using the proposed OTM method. This example is inspired by the continued success of the airbags used by NASA to dissipate the impact of landing on Mars in past missions.

Chapter 2

Particle and Meshfree Methods

This chapter presents a review of commonly used particle and meshfree methods along with the associated challenges of imposing essential boundary conditions, numerical integration, and coupling the schemes to other methods (i.e. with finite elements). Here, the focus of the review is on continuum-scale methods, so atomistic methods, such as Molecular Dynamics [19], are not discussed. In this chapter, particle and meshfree methods are classified independently. While both types of methods discretize the physical domain by a collection of nodes or particles, particle methods maintain all of the physical and mathematical properties on the particle itself. Physical properties (velocity, density, temperature, pressure, etc.) as well as relevant mathematical variables (gradients, volumetric deformations, etc.) are assigned to each particle and represent a neighborhood average of the field value around the particle. Here, each particle moves according to its own mass and applied forces (internal and external), where the external forces are evaluated by direct interactions with the neighboring particles. This is somewhat distinct from meshfree methods, which use the discretization for generating shape functions that potentially provide the integration scheme for the weak form, equation 1.3. Here, the solution is evaluated at the integration points and subsequently interpolated to the nodes to provide the evolution of the unknown field.

The interest in particle and meshfree methods has been recently rekindled in response to the increased computational costs and limitations of traditional grid-based methods. Attempts to simulate large-deformation processes involving moving boundary definition with traditional numerical methods have highlighted the limitations and motivated the need for new numerical methods. Furthermore, generating and maintaining a high-quality numerical mesh for dynamic simulations presents a tremendous challenge since there is no mathematical guarantee that an algorithm will converge for an arbitrary three-dimensional geometry. In turn, updating the deformed mesh requires significant attention and user intervention. Indeed, the majority of simulation efforts for most applications is consumed by mesh-related tasks. Of the presented meshfree shape function, LME shape functions poses a number of salient features that motivate their use in the proposed OTM method.

2.1 Why Gridless?

While the FEM, particle and meshfree methods rely on shape functions to approximate fields in the weak formulation of the BVP, particle and meshfree methods aim to generate shape functions based on nodal locations for a given discretization of the geometry (nodes/particles) without recourse to a grid (mesh) and the associated connectivity information. The lack of robust algorithms for generating an acceptable numerical mesh of an arbitrary three dimensional geometry forces engineers and scientists to spend the majority of their simulation and analysis efforts in simply generating the numerical mesh. In practice, the mesh generation process involves an iterative and often infuriating procedure of using automeshers or numerical algorithms combined with manual adjustments. For this reason alone, particle and meshfree methods present an appealing direction of research for static problems. When considering dynamic simulations of continuum materials undergoing large deformations, the unresolved issues of meshing are constantly revisited by the necessity to re-mesh the domain in order to avoid deteriorated accuracy of the solution due to poor aspect ratios of the mesh as well as termination of the simulation due to mesh entanglement and inversion (i.e. negative Jacobian). In this sense, meshfree methods bear the marked advantage over mesh-based approaches in that, ideally, no inherent limitations are placed on the allowable deformation of the continuum in consideration. Furthermore, another advantage over grid-based numerical methods lies in the fact that the time required to determine connectivity in meshfree methods (for generating the shape functions or integration scheme) can always be bounded in time, which is not generally the case for three-dimensional meshes. This distinction is exceedingly important for simulating large deformation processes since grid-based methods may require constant re-meshing to maintain well-behaved elements.

While there is tremendous potential for applying particle and meshfree methods to facilitate previously unthinkable simulations, there are several significant obstacles in the path of gridless methods gaining acceptance and widespread use in the engineering and scientific community. The inherent challenge of establishing integration weights for general domains (without connectivity information) as well as the commonly faced challenge of imposing essential boundary conditions all within a consistent mathematical framework represent significant obstacles that require new mathematical developments. The OTM method, detailed in chapter 4, provides a unique mathematical framework for meshfree methods. Specifically, the combination of material point sampling and conforming interpolation provides a unique meshfree algorithm endowed with geometrically exact mass density updates. Combined with the use of LME shape functions, the OTM method inherits the ability to enforce essential boundary conditions and couple with finite elements in a trivial manner.

2.2 Particle Methods

Historically, particle methods represent the earliest efforts of devising gridless numerical methods. Particle methods have had success in numerically modeling large deformation processes and discrete phenomena. Of these methods, smoothed particle hydrodynamics has been the most widely studied and used.

2.2.1 Smoothed Particle Hydrodynamics (SPH)

Smoothed particle hydrodynamics (SPH), developed in 1977 independently by Lucy [20] and Monaghan [21], is one of the oldest and most well-studied particle methods. The SPH method has been used in a wide range of applications including astrophysics problems such as star formation [22], as well as continuum problems involving sloshing fluid flows [23] and fracture of brittle solids [24]. Indeed several commercial software packages have coupled SPH routines into their numerical capabilities. As an illustration of the popularity of this method, consider that in the movie, *Lord of the Rings: Return of the King*, SPH was used to simulate the lava flow that the character, Gollum, fell into at the end of the movie.

While there have been many variations since its inception, SPH represents a general class of strong form collocation methods. SPH makes use of an integral, or *kernel*, approximation based on the following identity of the Dirac delta function, $\delta(\mathbf{x})$:

$$\mathbf{u}(\mathbf{x}) = \int u(\mathbf{y})\delta(\mathbf{x}-\mathbf{y})d\mathbf{y}.$$

The SPH method relies on approximating the delta function with a smoothing kernel, $w(\mathbf{x}, h)$. Specifically, the SPH approximation takes the form:

$$\mathbf{u}^h(\mathbf{x}) = \int_\Omega \mathbf{u}(\mathbf{y})w(\mathbf{x}-\mathbf{y}, h)d\mathbf{y}. \tag{2.1}$$

Here, h is the smoothing length that prescribes the radius of support for the smoothing kernel. According to Monaghan [25], a candidate smoothing kernel should have the following properties:

$\int_\Omega w(\mathbf{x}-\bar{\mathbf{x}},h)d\bar{\mathbf{x}} = 1$ (Normalization property)

$w(\mathbf{x}-\bar{\mathbf{x}},h) = 0$ for $|\mathbf{x}-\bar{\mathbf{x}}| > \kappa h$ (Compact support)

$w(\mathbf{x}-\bar{\mathbf{x}},h) > 0$ for all $\bar{x}$ in the support domain of x (Positivity property)

$\lim_{h\to 0} w(\mathbf{x}-\bar{\mathbf{x}},h) \to \delta(\mathbf{x}-\bar{\mathbf{x}})$ (Delta function property)

$w(\mathbf{x}-\bar{\mathbf{x}},h)$ is a monotonically decreasing function (Decay property)

$w(\mathbf{x}-\bar{\mathbf{x}},h)$ is a symmetric function (Symmetric property)

Here, the normalization condition amounts to satisfying the zeroth-order consistency condition (i.e. exactly approximating constant functions). The second condition localizes the global SPH approximation based on a scaling factor, κ, whose effect is to prescribe the size of the local neighbor (or support domain) of nodes that contribute to the evaluation of field variables. Positivity is required to avoid unphysical results, such as negative density or energy, and prevent numerical instabilities (i.e. spurious oscillations). The decay property is mathematically unnecessary but represents the intuitive notion adopted from basic physics: the magnitude of particle interactions is inversely related to increased separation. Furthermore, requiring the smoothing kernel to be a monotonically decreasing function ensures that the neighbor distance of particles directly correlates with their level of influence. Choosing the smoothing kernel to be an even, symmetric function is enforced out of convenience since this requirement eliminates the first error term in the Taylor series expansion of equation (2.1).

It should be noted that these conditions represent general, intuitive guidelines for generating smoothing kernels for the SPH method. Many of the kernel functions that have been proposed, in practice, violate these conditions. In particular, due to the fact that the SPH shape functions lack the Kronecker-delta property, the normalization or zeroth-order consistency condition (if uncorrected) is violated at the boundary. Furthermore, many of the proposed smoothing functions violate the positivity requirement over their support domain.

This typically leads to instabilities and spurious oscillations. Below are three commonly used smoothing kernels taken from [21], [25], and [26], respectively.

Gaussian Kernel

$$w(|\mathbf{x}-\bar{\mathbf{x}}|,h)=\frac{\sigma}{h^d}e^{-\left(\frac{|\mathbf{x}-\bar{\mathbf{x}}|}{h}\right)^2}$$

where $\sigma=\{\frac{1}{\sqrt{\pi}},\frac{1}{\pi},\frac{1}{\pi\sqrt{\pi}}\}$ for dimension, $d=1,2,3$.

Cubic Splines

$$w(|\mathbf{x}-\bar{\mathbf{x}}|,h)=\frac{\sigma}{h^d}\begin{cases}1-\frac{3}{2}\left(\frac{r}{h}\right)^2+\frac{3}{4}\left(\frac{r}{h}\right)^3 & 0\le\frac{r}{h}<1,\\ \frac{1}{4}\left(2-\frac{r}{h}\right)^3 & 1\le\frac{r}{h}<2,\\ 0 & \frac{r}{h}\ge 2,\end{cases}$$

where for normalization, $\sigma=\{\frac{2}{3},\frac{10}{7\pi},\frac{1}{\pi}\}$ for dimension, $d=1,2,3$.

Piecewise Quartic Splines

$$w(|\mathbf{x}-\bar{\mathbf{x}}|,h)=\frac{\sigma}{h^d}\begin{cases}\left(2.5-\frac{r}{h}\right)^4-5\left(1.5-\frac{r}{h}\right)^4+10\left(0.5-\frac{r}{h}\right)^4 & 0\le\frac{r}{h}<0.5,\\ \left(2.5-\frac{r}{h}\right)^4-5\left(1.5-\frac{r}{h}\right)^4 & 0.5\le\frac{r}{h}<1.5,\\ \left(2.5-\frac{r}{h}\right)^4 & 1.5\le\frac{r}{h}<2.5.\\ 0 & \frac{r}{h}\ge 2.5\end{cases}$$

In the SPH method, the integral (2.1) is discretized by a set of N interpolation points, or particles (i.e. using a trapezoidal rule for the numerical quadrature) resulting in the following approximation:

$$\begin{aligned} u^h(\mathbf{x}) &\approx \sum_{I=1}^{N} w(\mathbf{x}-\bar{\mathbf{x}})u_I\Delta x_I \qquad \text{(one dimension)}, \\ &\approx \sum_{I=1}^{N} w(\mathbf{x}-\bar{\mathbf{x}})u_I\Delta V_I \qquad \text{(three dimensions)}, \end{aligned}$$

where ΔV_I represents the volume associated with node I. As compared with equation 1.7, the SPH shape functions can be written as

$$\phi_I^{SPH}(\mathbf{x}) = w(\mathbf{x}-\mathbf{x}_I, h)\Delta V_I.$$

The SPH method has been one of the most widely used and studied particle methods. SPH, in its original form suffered due to a lack of consistency that led to degraded accuracy. In subsequent work [25, 27], symmetrization methods were proposed to allow the shape functions to reproduce linear fields and thus, conserve linear and angular momentum. Furthermore, the original SPH scheme was unstable for positive (tensile) pressures leading to unphysical results and spurious oscillation modes [28]. In [29], it was shown that the SPH method could be partially stabilized by adding stress points for integration. However, in [30] and [31] it was recognized that this tensile instability originated from using an Eulerian kernel with a Lagrangian description of motion. Here, it was also noted that the tensile instability for all particle methods can only be eliminated by using a Lagrangian kernel.

2.2.2 Reproducing Kernel Particle Methods (RKPM)

In 1995, Liu proposed the reproducing kernel particle method (RKPM) to improve the accuracy of the SPH method [32]. While the RKPM method derived from wavelet theory, the resulting approximation scheme is similar, if not identical, to moving least square (MLS) approximants [33]. In the RKPM, the kernel approximation is of the form:

$$\begin{aligned}\mathbf{u^h}(\mathbf{x}) &= \int_{\Omega_y} K(\mathbf{x},\mathbf{y})u(\mathbf{y})d\mathbf{y} \\ &= \int_{\Omega_y} C(\mathbf{x},\mathbf{x}-\mathbf{y})w(\mathbf{x}-\mathbf{y})\mathbf{u}(\mathbf{y})d\mathbf{y}, \qquad (2.2)\end{aligned}$$

where $C(\mathbf{x},\mathbf{x}-\mathbf{y})$ is the correction that restores consistency to the SPH shape functions, and $w(\mathbf{x}-\mathbf{y})$ is the weighting function. Note that if $K = w$, the RKPM approximation is exactly equivalent to the SPH approximation. Here the integral in equation 2.2 is discretized as

$$\mathbf{u}^h(\mathbf{x}) = \sum_{I=1}^{N} C(\mathbf{x},\mathbf{x}_I)w(\mathbf{x}-\mathbf{x}_I)u_I \triangle V_I,$$

where $\triangle V_I$ is the volume related to particle I. Considering the continuous counterpart to the MLS derivation in the preceding section, the RKPM approximants can be derived by considering the weighted L_2 norm:

$$J(\mathbf{x}) = \int_{\Omega} w(\mathbf{x}-\bar{\mathbf{x}})[\mathbf{u}^h(\mathbf{x},\bar{\mathbf{x}}) - \mathbf{u}(\bar{\mathbf{x}})]^2 d\bar{\mathbf{x}}.$$

Using the following approximation,

$$\mathbf{u}^h(\mathbf{x},\bar{\mathbf{x}}) = \sum_{I=1}^{m} p_I(\bar{\mathbf{x}})a_I(\mathbf{x}),$$

to evaluate the extremum of $J(\mathbf{x})$ leads to

$$\begin{aligned}\frac{\partial J}{\partial \mathbf{a}} &= 2\int_{\Omega} w(\mathbf{x}-\bar{\mathbf{x}})\left[\sum_I p_I(\bar{\mathbf{x}})a_I(\mathbf{x}) - u(\bar{\mathbf{x}})\right]\left[\sum_J p_J(\bar{\mathbf{x}})da_J(\mathbf{x})\right]d\bar{\mathbf{x}} \qquad (2.3) \\ &= 0\,\forall\, da_J. \qquad (2.4)\end{aligned}$$

Then, introducing the continuous counterpart of the discrete moment matrix,

$$\bar{A}_{IJ}(x) = \int_\Omega w(\mathbf{x}-\bar{\mathbf{x}})p_I(\bar{\mathbf{x}})p_J(\bar{\mathbf{x}})d\bar{\mathbf{x}},$$

and substituting this form into equation 2.3 gives

$$\sum_J \bar{A}_{IJ}(\mathbf{x})a_J(\mathbf{x}) = \int_\Omega w(\mathbf{x}-\bar{\mathbf{x}})p_I(\bar{\mathbf{x}})u(\bar{\mathbf{x}})d\bar{\mathbf{x}}.$$

Finally, solving for a_j and substituting back into the approximation provides the RKPM approximation:

$$\begin{aligned}\mathbf{u}^h(\mathbf{x}) &= \mathbf{p}^\top(x)\left[\int_\Omega w(\mathbf{x}-\bar{\mathbf{x}})\mathbf{p}(\bar{\mathbf{x}})\mathbf{p}^\top(\bar{\mathbf{x}})d\bar{\mathbf{x}}\right]^{-1}\int_\Omega w(\mathbf{x}-\bar{\mathbf{x}})\mathbf{p}(\bar{\mathbf{x}})\mathbf{u}(\bar{\mathbf{x}})d\bar{\mathbf{x}} \\ &= \int C(\mathbf{x},\bar{\mathbf{x}})w(\mathbf{x}-\bar{\mathbf{x}})\mathbf{u}(\bar{\mathbf{x}})d\Omega \\ &= \sum_{I=1}^N C(\mathbf{x},\mathbf{x}_I)w(\mathbf{x}-\mathbf{x}_I)\mathbf{u}_I\triangle V_I.\end{aligned}$$

Compared with the SPH approximant: $\mathbf{u}^h(\mathbf{x}) = \int_\Omega \mathbf{u}(\mathbf{y})w(\mathbf{x}-\mathbf{y},h)d\mathbf{y}$, the term $C(\mathbf{x},\bar{\mathbf{x}}) = p_I(\mathbf{x})\bar{A}_{IJ}^{-1}(\mathbf{x})p_J(\bar{\mathbf{x}})$ represents a correction term that restores consistency to the SPH kernel approximation. Here, if the integration weight $\triangle V_I = 1$, one exactly recovers the MLS approximation. Applications of RKPM for numerical simulations of fluid flows can be found in [34, 35].

2.2.3 Particle-In-Cell/Material Point Methods (PIC/MPM)

The particle-in-cell method (PIC) [36, 37], proposed in 1957, represents one of the earliest attempts to model fluid dynamics from a Lagrangian point of view. This method uses a Lagrangian description for the motion of the particles combined with an Eulerian mesh for interpolating information from the mass particles to the Eulerian mesh. Here, the goal for the approach is to draw on the advantages of both the Lagrangian and Eulerian frameworks. More recently, in 1995 Sulsky and co-workers [38, 39] extended the concepts of PIC to develop the material point method (MPM). In the MPM, a numerical mesh, comprised of Lagrangian material points, is used to discretize the domain. Here, the material points carry all of the physical properties such as mass, velocity, stress, and strain. Furthermore, the interaction of the material points is determined on a background Eulerian finite element mesh. Standard finite element shape functions are used to map the field information from the material points to the nodes of the bounding cell. Then, the material point field variables are updated from the solution, found on the Eulerian mesh, and mapped onto the corresponding Lagrangian points. The mass density for the MPM method is given by

$$\rho(\mathbf{x}, t) = \sum_{p=1}^{N} m_p \delta(\mathbf{x} - \mathbf{x}_p),$$

where $\delta(\mathbf{x})$ is the Dirac delta function. Substituting this form into the weak formulation, i.e., equation 1.3, converts the integral into a sum of quantities that are evaluated at the material points.

While the gradient and divergence terms are calculated on a background mesh, the MPM is viewed as a meshfree method since the interpolation is based on a moving domain of influence that continually updates with the deformation of the body. Although the MPM method does not introduce a new class of shape functions, the method cleverly utilizes the advantages of a mixed Lagrangian-Eulerian description to avoid the inherent challenges in each. Applications of the MPM method for numerical simulations of fluid flows and FSI applications can be found in [40, 41].

2.2.4 Vortex Methods (VM)

Vortex methods, developed independently by Rosenhead in 1930 [42], Chorin in 1973 [43], and Leonard in 1980 [44], have been successfully used in direct simulations of viscous incompressible flows and, more recently, have been extended for compressible flow [45]. Vortex methods formulate the Navier-Stokes equations as a vorticity-velocity formulation rather than as a velocity-pressure formulation. Furthermore, VMs discretize the domain into a finite number of moving particles, or *vortex blobs*. Here, bearing the Lagrangian framework, simply tracking the motion of the particles explicitly directly provides the vorticity of the flow. In this sense, VMs are inherently and fundamentally linked to the underlying physics which the formulation aims to solve. As in the SPH method, VMs use a smoothing kernel approximation, $\zeta_\epsilon(\mathbf{x})$, of the form:

$$\begin{aligned} f_\epsilon(\mathbf{x}) &= \int f(\mathbf{y})\zeta_\epsilon(\mathbf{x}-\mathbf{y})d\mathbf{y}, \\ f_\epsilon^h(\mathbf{x}) &= \sum_{p=1}^{N} w_p \zeta_\epsilon(\mathbf{x}-\mathbf{x}_p), \end{aligned}$$

where ϵ denotes the characteristic length of the kernel. Posed as a vorticity-velocity formulation, the Navier-Stokes equations describe the evolution of the vorticity field, $\omega = \nabla \times \mathbf{u}$. For incompressible flows, the equations governing the evolution of vorticity and velocity field in Eulerian form are given, respectively, as

$$\frac{\partial \omega}{\partial t} + (\mathbf{u}\cdot\nabla)\omega = (\omega\cdot\nabla)\mathbf{u} + \nu\triangle\omega,$$

$$\triangle\mathbf{u} = -\nabla\times\omega.$$

Posed in the Lagrangian reference frame, the vorticity-velocity formulation is given, respectively, as

$$\begin{aligned}\frac{d\omega_p}{dt} &= \left[\nabla \mathbf{u}(\mathbf{x}_p, t)\right] \omega_p + \nu \triangle \omega(\mathbf{x}_p),\\ \frac{d\mathbf{x}_p}{dt} &= \mathbf{u}(\mathbf{x}_p, t),\end{aligned}$$

where $\mathbf{x}_p$ and ω_p denote the locations and vorticity carried by the fluid element, or *vortex blob*. A classical way to relate the velocity and vorticity is through an integral representation. In particular, velocity calculations explicitly satisfying the far-field boundary conditions are based on the Biot-Savart law,

$$\mathbf{u}(\mathbf{x}) = \int \mathbf{K}(\mathbf{x} - \mathbf{y}) \times \omega(\mathbf{y}) d\mathbf{y} + \mathbf{U}_0.$$

Here, $\mathbf{U}_0$ is the solution to the Laplace's equation, $\triangle \mathbf{u} = 0$, and $K(\mathbf{z})$ denotes the Biot-Savart kernel for the Poisson equation (i.e. the rotational counterpart to the Green's function for the Poisson equation). Due to the function's inherent singularity, $\mathbf{K}$ is typically replaced with a mollified or smoothed form: $\mathbf{K}_\epsilon = \int \mathbf{K}(\mathbf{y}) \zeta_\epsilon(\mathbf{x} - \mathbf{y}) d\mathbf{y}$.

Furthermore, the *vortex blob* approximation for the vorticity-carrying particles with finite core size, ϵ, is given by

$$\omega_\epsilon^h(\mathbf{x}) = \sum_{p=1}^{N} v_p \omega_p \zeta_\epsilon(\mathbf{x} - \mathbf{x}_p),$$

where v_p represents the volume of the *vortex blob*. Then, the dynamics of the *vortex blobs* are determined by solving the following system of ODEs for the particle locations and vorticity:

$$\frac{d\mathbf{x}_p}{dt} = \sum_{q=1}^{N} v_q \mathbf{K}_\epsilon(\mathbf{x}_p - \mathbf{x}_q) \times \omega_q + \mathbf{U}_0(\mathbf{x}_p, t),$$
$$\frac{d\omega_p}{dt} = \left[\sum_{q=1}^{N} v_q \nabla \mathbf{K}_\epsilon(\mathbf{x}_p - \mathbf{x}_q) \times \omega_q\right] \omega_p + \frac{\nu}{\epsilon^2} \sum_{q=1}^{N} v_q \left[\omega_q - \omega_p\right] \eta_\epsilon(\mid \mathbf{x}_p - \mathbf{x}_q \mid) + \mathbf{F}(\mathbf{x}_p).$$

Here, $\mathbf{F}(\mathbf{x}_p)$ accounts for the generation of vorticity at solid boundaries and η_ϵ is a radially symmetric version of the smoothing kernel (i.e. $\eta_\epsilon = -\frac{1}{\epsilon^2}\frac{\zeta'(|\mathbf{x}|)}{|\mathbf{x}|}$). Applications of this method for numerical simulations of fluid flows can be found in [46], [45], [47], [48].

2.3 Meshfree Methods

While the area of numerical methods for mesh-independent methods has had significant attention and success for some applications, the apparent lack of mathematical proofs for fundamental concepts including numerical integration, convergence, stability, and completeness directly point to the infancy of this area of numerical research. In general, the goal of meshfree methods is to generate shape functions that can provide a convenient space of solution for the weak form of the BVP. Here, the domain is discretized by points or nodes without specifying or maintaining the associated connectivity of the discretization. Then, given a distribution of nodes $\{\mathbf{x}_i\}$, the algorithm given by each meshfree method provides the shape functions that are used to discretize the weak form, equation 1.3. The rest of this section provides an overview of the commonly used and historically relevant meshfree methods.

2.3.1 Moving Least Square Approximants (MLS)

Moving least square approximants are the basis for many meshfree methods. MLS approximants were originally introduced for smoothing and interpolating data by Lancaster and

Salkauskas (1981) [49]. MLS functions are *approximants* rather than interpolants, since $\phi_I^{MLS}(\mathbf{x}_J) \neq \delta_{IJ}$. MLS approximants consist of three components: a weight function associated with each node, a basis function (usually a polynomial), and a set of coefficients determined by the nodal positions. MLS shape functions possess the property that their continuity is directly related to the continuity of the weight function rather than the basis [50]. Thus, if the continuity of the basis is greater than the continuity of the weight function, the resulting approximation will inherit the continuity of the weight function. Therefore, even if one chooses a low order basis (i.e. linear), the level of desired continuity of the approximation can be imposed by choosing the weight function appropriately. Here, the weight function associated with a given node is chosen to have compact support outside of which the function's value is zero. Thus, the support of the weight function defines the node's domain of influence as the sub-domain region over which the node contributes to the approximation.

Assuming that the function $u(\mathbf{x})$ is sufficiently smooth on the domain of definition, the MLS local approximation around a fixed point in the domain, $\bar{\mathbf{x}}$, is given as

$$\begin{aligned} \mathbf{u}^h(\mathbf{x}, \bar{\mathbf{x}}) &= \sum_{I=1}^{m} p_I(\mathbf{x}) a_I(\bar{\mathbf{x}}) \\ &= \mathbf{P}^\top(\mathbf{x}) \mathbf{a}(\bar{\mathbf{x}}), \end{aligned}$$

where $p_I(\mathbf{x})$ is a complete polynomial basis of order m (i.e. $\mathbf{P}^\top(\mathbf{x}) = [1\ \mathbf{x}\ \mathbf{x}^2\ \ldots\ \mathbf{x}^m]$), and $a_i(\mathbf{x})$ is the set of unknown coefficients (i.e. $\mathbf{a}^\top(\mathbf{x}) = [a_o(\mathbf{x})\ a_1(\mathbf{x})\ a_2(\mathbf{x})\ \ldots\ a_m(\mathbf{x})]$) to be determined based on the nodal data. In particular, the coefficients, $a_i(\mathbf{x})$, are obtained by minimizing the discrete weighted L_2 norm, J, for the local approximation given by

$$
\begin{aligned}
J &= \sum_{I=1}^{N} w(\mathbf{x}-\mathbf{x}_I)\left[\mathbf{u}^h(\mathbf{x},\mathbf{x}_I)-\mathbf{u}(\mathbf{x}_I)\right]^2 \\
&= \sum_{I=1}^{N} w(\mathbf{x}-\mathbf{x}_I)\left[\sum_{i=1}^{m} p_I(\mathbf{x}_I)a_I(\mathbf{x})-\mathbf{u}(\mathbf{x}_I)\right]^2 \\
&= \sum_{I=1}^{N} w(\mathbf{x}-\mathbf{x}_I)\left[\mathbf{P}^\top(\mathbf{x}_I)\mathbf{a}(\mathbf{x})-u(\mathbf{x}_I)\right]^2 .
\end{aligned}
$$

Here, N is the number of nodes in the neighborhood of $\mathbf{x}$ for which the weight function is nonzero (i.e $w(\mathbf{x}-\mathbf{x}_I) \neq 0$). Subsequently, the unknown coefficients, $\mathbf{a}(\mathbf{x})$, are evaluated by considering the extremum of J:

$$
\begin{aligned}
\frac{\partial J}{\partial \mathbf{a}} &= \mathbf{A}(\mathbf{x})\mathbf{a}(\mathbf{x})-\mathbf{B}(\mathbf{x})\mathbf{u} \\
&= 0,
\end{aligned}
\tag{2.5}
$$

where $\mathbf{A}(x)$ is the moment matrix:

$$
\mathbf{A}(x) = \sum_{I=1}^{n} w(\mathbf{x}-\mathbf{x}_I)\mathbf{P}(\mathbf{x}_I)\mathbf{P}^\top(\mathbf{x}_I),
$$

and $\mathbf{B}(x)$ is defined by

$$
\begin{aligned}
\mathbf{B}(\mathbf{x}) &= \left[w(\mathbf{x}-\mathbf{x}_1)\mathbf{P}(\mathbf{x}_1)\ ,\ w(\mathbf{x}-\mathbf{x}_2)\mathbf{P}(\mathbf{x}_2)\ \ldots\ w(\mathbf{x}-\mathbf{x}_n)\mathbf{P}(\mathbf{x}_n)\right] \\
&= \mathbf{P}^\top(\mathrm{x})\mathbf{W}(\mathrm{x}).
\end{aligned}
$$

The solution to equation 2.5 leads to the linear problem,

$$
\mathbf{A}(\mathbf{x})\mathbf{a}(\mathbf{x}) = \mathbf{B}(\mathbf{x})\mathbf{u}, \tag{2.6}
$$

giving the unknown coefficients $\mathbf{a}(x)$ in terms of the nodal coordinates $\mathbf{u}^\top = [\ \mathbf{u}_1 \quad \ldots \quad \mathbf{u}_N\ ]$ as,

$$\mathbf{a}(\mathbf{x}) = \mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})\mathbf{u}.$$

Then, in the spirit of approximations of the form in equation 1.7, the MLS shape functions can be written as

$$\begin{aligned}\phi_a^{MLS}(\mathbf{x}) &= \sum_{i=1}^{m} p_i(\mathbf{x}) \left(\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x})\right)_{ia} \\ &= \mathbf{P}^\top(\mathbf{x})\mathbf{A}^{-1}(\mathbf{x})\mathbf{B}_a,\end{aligned}$$

where m is the order of the polynomial space, $p(\mathbf{x})$. For the specific case, $m = 0$, the MLS shape functions reduces to *Shepard functions*,

$$\phi_I^0(\mathbf{x}) = \frac{w(\mathbf{x} - \mathbf{x}_I)}{\sum_J w(\mathbf{x} - \mathbf{x}_J)}.$$

2.3.2 Diffuse Element Methods (DEM)

In 1992, Nayroles and co-workers [51] were the first to use moving least square approximants (MLS) in a Galerkin setting to formulate the diffuse element method (DEM). DEM was posed as a generalization of the FEM in the sense that replacing the FEM interpolation, valid on an element, with a weighted least-squares fit based on a variable number of nodes in a small neighborhood of a point $\bar{x}$, one achieves a *diffuse* interpolation. This led to the development of shape functions that are identical to MLS approximants.

Furthermore, directly evaluating the first derivative of the MLS approximants gives,

$$\begin{aligned}\phi_{I,j}(\mathbf{x}) &= \left(\mathbf{P}^\top\mathbf{A}^{-1}\mathbf{B}_I\right)_{,j} \\ &= \mathbf{P}^\top_{,j}\mathbf{A}^{-1}\mathbf{B}_I + \mathbf{P}^\top\mathbf{A}^{-1}_{,j}\mathbf{B}_I + \mathbf{P}^\top\mathbf{A}^{-1}\mathbf{B}_{I,j}.\end{aligned} \tag{2.7}$$

However, in evaluating the derivatives of field variables using MLS approximants, the DEM formulation incorrectly assumed that the second two terms of the derivative expression above were negligible. Thus, the following form of the derivative was employed:

$$\phi_{I,j}(\mathbf{x}) = \mathbf{P}_{,j}^{\top}\mathbf{A}^{-1}\mathbf{B}_{I}.$$

This simplification forces the DEM method to lose linear consistency. Furthermore, the original DEM method did not enforce essential boundary conditions correctly and incorrectly assumed that a low-order quadrature rule was sufficient.

2.3.3 Element-Free Galerkin Methods (EFG)

Building upon the DEM formulation, in 1994 Belytschko and co-workers [52] developed the element-free Galerkin method (EFG). The EFG method uses MLS approximants for the trial and test functions in the Galerkin weak form. The EFG method increases the accuracy of DEM by using the full form of the derivative, i.e., equation 2.7, of the MLS shape function to restore linear consistency.

In using MLS shape functions, a weighting function must be specified. Here, the main parameter of concern is the shape and size of the support domain for a given node. In general, circular or spherical supports are employed to maintain isotropic weights. Furthermore, the choice of weight function and subsequent radius of support has a significant effect on the solution as it dictates which nodes interact with each other. This support domain must be large enough to have a sufficient number of nodes to solve the linear system for the MLS approximants, equation 2.6. Typically, splines are used for the weighting function. One common weighting function used in EFG [53] is a cubic spline of the form:

$$w(r) = \begin{cases} \frac{2}{3} - 4r^2 + 4r^3 & r \leq \frac{1}{2}, \\ \frac{4}{3} - 4r + 4r^2 - \frac{4}{3}r^3 & \frac{1}{2} < r \leq 1, \\ 0 & r > 1, \end{cases}$$

where $r = \frac{d_I}{d_{IM}}$ is a normalized radius giving the ratio of the radius of support for node I, d_I, to the scaled distance that provide a sufficient number of neighboring nodes for evaluating equation 2.6.

Since nodal integration schemes were found to be unstable [33], integration of the weak form, equation 1.3, in EFG is performed by recourse to a background mesh or cells combined with Gaussian quadrature. In [54], it was noted that alignment of the cells with the support domains of the nodes as well as using high-order Gaussian quadrature rule was necessary for maintaining accuracy. This can be difficult to ensure, especially for dynamic simulations. In the search for lower-order quadrature rules for EFG, as well as in an effort to avoid the difficulties of matching the support and integration domains, recent work has focused on employing nodal integration with the concept of stress point integration for stabilization [55, 56].

Using MLS shape functions does not provide the Kronecker-delta property (i.e. $\phi_I(\mathbf{x}_J)$ $\neq \delta_{IJ}$). Thus, implementing essential boundary conditions requires specific attention. Several methods have been considered including modifying the weak formulation by using Lagrange multipliers [57] Nitsche's method [58], and by coupling to finite elements [53, 59].

2.3.4 Partition of Unity Finite Element Methods (PUM)

The set of shape functions $\{\phi_I\}$ consistent up to order n can also be viewed as a partition of unity (PU) of order n. For example, a partition of unity of order $n = 0$ fulfills the zero order consistency condition, $\sum_I \phi_I(\mathbf{x})$=1. As such, any PU can be used to reproduce constant functions (polynomials of order $n = 0$) exactly. The PU concept can be constructed based on a mesh, leading to the traditional finite element shape functions, or by using a gridless approach such as MLS approximants.

In 1997, Babuska and Melenk [60] proposed the concept of the partition of unity finite element method (PUFEM). In this approach, a domain is covered by a set of overlapping patches $\{\Omega_I\}$ that form a cover of the domain Ω. Each of these overlapping patches possesses an associated basis $\{\phi_I\}$ that satisfies the zero-order consistency condition. The essence of the PUFEM is to take a partition of unity and multiply it with another independent basis to *enrich*, or provide higher consistency to, the approximation scheme. This allows one to directly build in any $a\ priori$ knowledge of the solution into the approximation scheme. Specifically, in some circumstances, essential boundary conditions can be enforced by choosing the local approximation spaces to satisfy them inherently. PUMs employ an extrinsic basis $\{p_j(\mathbf{x})\}$ in addition to the intrinsic basis $\{\phi_i(\mathbf{x})\}$, which is chosen to satisfy the partition of unity. Here, the approximation in equation 1.7 takes the form:

$$\mathbf{u}^h(\mathbf{x}) = \sum_{i=1}^{N} \phi_i(\mathbf{x}) \sum_{j=1}^{l} p_j(\mathbf{x}))v_{ij},$$

where, $v_{ij} = (a_{i1,}\, a_{i2,}\, \ldots,\, a_{il})$. Here, the extrinsic basis, $\mathbf{p}(\mathbf{x})$, may consist of monomials, Taylor polynomials, Lagrange polynomials, or any other convenient functions. In the case where Lagrange polynomials are used for the extrinsic basis, the approximation scheme possesses the Kronecker delta property, which, as noted previously, facilitates imposing essential boundary conditions in a trivial manner. If the solution is known *a priori*, then this information can be built into the extrinsic basis to enhance the function and improve the accuracy. For example, in [60], the extrinsic basis, $\mathbf{p}^\top = [1, x, \ldots, x^{l-2}, \sinh(nx), \cosh(nx)]$ was used to demonstrate the ability to improve the accuracy of the solution to the Helmholtz equation in one dimension, as compared with a polynomial basis. The original PUM used Shepard functions $\phi_i^0(\mathbf{x})$ for the partition of unity and employed an extrinsic basis to enrich the consistency of the approximation.

2.3.5 *Hp*-Cloud Methods

In 1996, Duarte and Oden [61] proposed the *hp*-cloud method which draws on the PUM by building a partition of unity using MLS shape functions of order k. Here the approximation in equation 1.7 takes the form,

$$\mathbf{u}^h(\mathbf{x}) = \sum_{I=1}^{N} \phi_I(\mathbf{x}) \left(\mathbf{u}_I + \sum_{J=1}^{k} p_J(\mathbf{x})) v_{IJ} \right),$$

where $p_j(\mathbf{x})$ can be any monomial basis of order greater than k. The major advantage of this framework is to enable the extrinsic basis to vary from node to node in an effort to achieve hp-adaptivity. Here, h-refinement is achieved by adding smaller sub-domain coverings, or *clouds*, while keeping the same degree of consistency for the *cloud* shape functions. Also, p-enrichment follows either by increasing the order of the polynomial used or by adding *a priori* knowledge of the solution into the construct of the extrinsic basis (*cloud* shape functions) while maintaining a constant number of *clouds*.

2.3.6 Meshless Local Petrov-Galerkin Methods (MLPG)

In 1998, Atluri and Zhu [62] developed the meshless local Petrov-Galerkin method (MLPG), which considers the local weak form built over local sub-domains. In the MLPG method, the nodal trial and test functions are chosen to be different, in a Petrov-Galerkin sense. Here, the local weak forms are generated on overlapping sub-domains, $\tilde{\Omega}_i$, rather than with global weak forms. In relation to the other meshfree methods, the MLPG can be viewed as a concept rather than a method itself. While the MLPG does not introduce or generate new meshfree shape functions, any convenient meshfree approximants and test function can be employed in the scheme. Since the formulation is over local domains, the MLPG method only requires a local mesh for performing the numerical integration over the sub-domain.

Thus, MLPG shares a close resemblance to FVM or other sub-domain collocation methods [62, 63]. For this type of formulation, the proponents claim that it is more natural to perform integration over these overlapping regular-shaped domains rather than on a background mesh. This assertion led to the claim that the MLPG is a "truly meshfree" method [64]. Here the integration weights, for numerical integration on the sub-domains, are given by computing the areas (two dimensions) or volumes (three dimensions) for the overlapping, regular-shaped (i.e. circular or spherical) regions. A complete analysis of the MLPG method can be found in the monograph [65]. Also, applications of the MLPG method for numerical simulations of incompressible fluid flows are presented in [63].

2.3.7 Natural Element Methods (NEM)

The natural element method (NEM), developed in 1998 by Sukumar and co-workers [66], is a Galerkin-based meshfree method that uses polygonal interpolation (i.e natural neighbor interpolation) as shape functions. Sibson introduced a natural neighbor interpolation scheme for data fitting and smoothing that uses a weighted average to compute the interpolating function. Using these natural neighbor interpolants, the weak form integration for NEM, i.e., equation 1.3, is carried out by recourse to a background Delaunay triangulation. Recently, the connection between this type of interpolation as a special variety of maximum-entropy interpolation has been made in [67].

Other methods including the meshless finite element method (MFEM) [68] and its variation (i.e. particle finite element method (PFEM) [69]) use similar polygonal interpolation with schemes to speed up the computation of the shape functions as well as algorithms to resolve "sliver" elements (elements with poor aspect ratio, i.e., small area/volume ratios).

Sibson Interpolation

Sibson interpolation uses the second-order Voronoi diagram of the domain for defining the shape functions. As depicted in figure 2.1, given a distribution of nodes (in a plane for

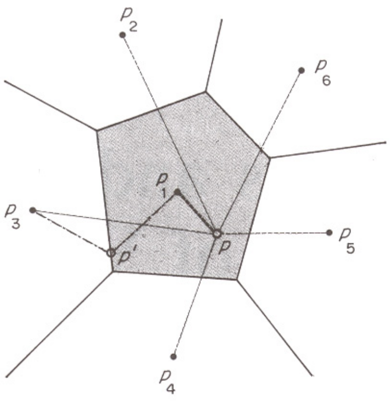

Figure 2.1: Voronoi cell [1]

simplicity), the Voronoi cell for a given node represents the domain that is closer to the node in Euclidean distance than any other node.

In other words, given a bounded domain Ω in d dimensions that is discretized by a set of m nodes, $N = \{n_1, n_2, \ldots, n_m\}$, the first-order Voronoi diagram $\mathcal{V}(N)$ is the set of the Voronoi cells, $V(n_i)$, associated with the point n_i, defined by,

$$V(n_i) = \{\mathbf{x} \in R^d \ : \ d(\mathbf{x}, \mathbf{x}_i) < d(\mathbf{x}, \mathbf{x}_j) \ \forall j \neq i\}$$

,where $d(\mathbf{x}, \mathbf{x}_i)$ is the appropriate metric, in this case the Euclidean norm. If, instead of considering those points that are closest to a given node, considering those locations that are closest to a pair of nodes defines the second-order Voronoi cell, V_{ij} corresponding to the two nodes. This concept is depicted in figure 2.2. Mathematically, the second-order Voronoi cell is given by

$$V_{ij} = \{\mathbf{x} \in R^2 \ : \ d(\mathbf{x}, \mathbf{x}_i) < d(\mathbf{x}, \mathbf{x}_j) < d(\mathbf{x}, \mathbf{x}_k) \ \forall k \neq i, j\}.$$

Constructing the second order Voronoi cell at a node, n_p, can be used to define the *natural neighbors* for interpolation. Specifically, the natural neighbor shape function for a node, n_p, with respect to a natural neighbor, n_I, is defined as the ratio of the area of the

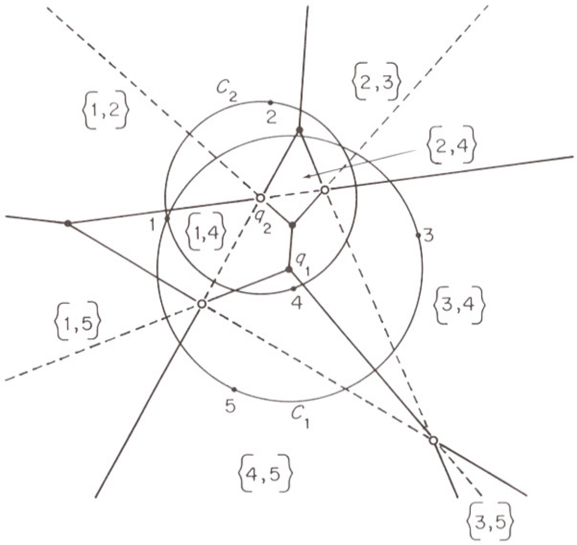

Figure 2.2: Ordinary Voronoi diagram (dashed lines) and second order Voronoi diagram (solid lines) generated from the same set of points[1].

second-order Voronoi cell, A_I, to the total area of the first-order Voronoi cell of n_p, $A(x_p)$. Thus, the natural neighbor Sibson interpolation takes the form

$$\phi_I(\mathbf{x}) = \frac{A_I(\mathbf{x})}{A(\mathbf{x})}\,, \quad A(\mathbf{x}) = \sum_{J=1}^{n} A(\mathbf{x}_J),$$

where n is the number of neighbors in the polyhedral cell. With this definition, the numerical approximation (i.e. equation 1.7) assumes the form:

$$\mathbf{u}^h(\mathbf{x}) = \sum_{I=1}^{n} \phi_I(\mathbf{x})\mathbf{u}_I,$$

which can be shown to satisfy linear consistency as well as define a partition of unity [66]. Combined with these shape functions, NEM uses a Delaunay triangulation as the background cells for performing the numerical integration of the weak form, i.e., equation 1.3.

2.3.8 Local Maximum-Entropy Approximation Schemes (LME)

In 2006, Arroyo and Ortiz [4] introduced local maximum-entropy approximation schemes (LME), which arise from the competing objectives of achieving an unbiased statistical inference and least-width (or local) shape functions. Here, the objective of statistical inference, based on the nodal data, is classically governed by Jaynes' principle of maximum entropy in information theory. Furthermore, local shape functions are found to be affine shape functions supported on a Delaunay triangulation of the node set. LME shape functions are derived to be *Pareto* optimal for these competing objectives and can be viewed as a smoothing or thermalization of the Delaunay triangulation of the node distribution. For a distribution of nodes where the Delaunay triangulation is not unique, LME schemes converge to a unique generalized (in the limit) Delaunay triangulation that maximizes the entropy of the approximation. Since the degree of thermalization is based on a single parameter, β, LME shape functions can be used to blend these nonlocal shape functions with localized finite element shape functions. In this way LME can be used to seamlessly transition from meshfree schemes to finite elements. In addition, LME shape functions belong to the set of convex approximations and inherit the weak Kronecker-delta property at the boundary. This is particularly important for imposing essential boundary conditions and represents a significant advantage over MLS and other proposed meshfree shape functions. Here, a brief overview of the development of local maximum-entropy shape functions is presented. A complete treatment of the concepts is formulated in [4].

Convex Approximation Schemes

Convex approximation schemes are characterized by the positivity of the shape functions and by being exact on affine functions. Consider a set of nodes, $X = \{\mathbf{x}_a, a = 1, \ldots, N\} \subset \mathbb{R}^d$. Then, the convex hull is defined by

$$\text{conv}\, X = \left\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} = \mathbf{X}\lambda(\mathbf{x}), \quad \lambda(\mathbf{x}) \in \mathbb{R}^N_+, \quad \mathbf{1} \cdot \lambda(\mathbf{x}) = 1\right\},$$

where $\mathbb{R}^N_+$ is the non-negative orthant, $\mathbf{1}$ represents the vector of size $\mathbb{R}^N$ with unit entries, and $\mathbf{X}$ is the $d \times N$ matrix whose columns are the coordinates of the position vectors of the nodes in the set, X. Since X is finite, conv X represents a compact convex polyhedron or polytope. Let $u :$ conv $X \to \mathbb{R}$ be a function whose values at the nodes $\{u_a, a = 1, \ldots, N\}$ are known. In terms of the shape functions, p_a : conv $X \to \mathbb{R}$, the approximation, $u^h(\mathbf{x})$, for $u(\mathbf{x})$ is given by

$$u^h(\mathbf{x}) = \sum_{a=1}^{N} u_a p_a(\mathbf{x}). \tag{2.8}$$

Enforcing the zero and first order consistency conditions, as well as positivity, amounts to the following conditions:

$$\sum_{a=1}^{N} p_a(\mathbf{x}) = 1 \quad \forall \mathbf{x} \in \text{conv } X$$

$$\sum_{a=1}^{N} \mathbf{x}_a p_a(\mathbf{x}) = \mathbf{x} \quad \forall \mathbf{x} \in \text{conv } X$$

$$p_a(\mathbf{x}) \geq 0 \quad \forall \mathbf{x} \in \text{conv } X,\ a = 1, \ldots, N$$

Here, the positivity requirement and partition of unity properties give the interpretation of the shape functions as coefficients of convex combinations. Since these conditions do not uniquely define the shape functions, additional constraints must be imposed to define a specific approximation scheme. In general, convex approximation schemes do not satisfy a strong Kronecker-delta property (i.e. $u^h(\mathbf{x}_a) = u_a$), but rather satisfy a weak Kronecker-delta property at the boundary. That is, if F is a face of conv X and the node $\mathbf{x}_a \notin F$, then $p_a = 0$ on F. This feature makes imposing boundary conditions trivial and analogous to finite element methods.

Global Maximum Entropy Schemes

As mentioned above, the requirements for a convex approximation scheme do not lead to unique approximants. In [4], a constraint involving the entropy of the approximation scheme is imposed. As an example of this constraint, consider a random variable that can take values given in a set of events $\{A_1, A_2, \ldots, A_n\}$ with associated probabilities $\{p_1, p_{2,} \ldots, p_n\}$. Then, the set of events and the associated probabilities jointly define a finite scheme, A:

$$A = \begin{pmatrix} A_1 & A_2 & \cdots & A_n \\ p_1 & p_2 & \cdots & p_n \end{pmatrix}$$

For concreteness, consider two finite schemes:

$$\begin{pmatrix} A_1 & A_2 \\ 0.5 & 0.5 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} A_1 & A_2 \\ 0.99 & 0.01 \end{pmatrix}$$

Here, the first scheme carries more uncertainty than the second one, since the outcome is almost certainly A_1 for the second scheme. The uncertainty associated with a finite scheme can also be interpreted as the amount of information gained by the realization of a random variable, which, in turn, eliminates the uncertainty completely. In 1948, Shannon [70], introduced the following measure of uncertainty or information entropy:

$$\begin{aligned} H(A) &= H(p_1, \ldots, p_n) \\ &= -\sum_{a=1}^{n} p_a \log p_a, \end{aligned}$$

with the extension $0 \log 0 = 0$ by continuity. Here, the function $H(A)$ is non-negative, symmetric, continuous, strictly concave and possesses the usual properties consistent with a measure of uncertainty.

Every convex approximation scheme can be characterized by its information entropy. As an example of biased schemes, consider an approximation scheme in one dimension with two nodes and associated shape functions $p_1(\hat{\mathbf{x}}), p_2(\hat{\mathbf{x}})$ at a point $\hat{\mathbf{x}}$. If we design our shape functions in such a way that $p_1(\hat{\mathbf{x}}) = 0.99$ and $p_2(\hat{\mathbf{x}}) = 0.01$ and use equation 2.8 for approximating a function $u^h(\hat{\mathbf{x}})$, then, by design, it is tacitly assumed that the solution $u(\hat{\mathbf{x}})$

is heavily influenced by the data at point $\mathbf{x}_1$ rather than by the data at point $\mathbf{x}_2$. However, if we had chosen $p_1(\hat{\mathbf{x}}) = 0.5$ and $p_2(\hat{\mathbf{x}}) = 0.5$, then as little bias as possible would have been placed on the relative influence of the nodes at point $\hat{\mathbf{x}}$. Indeed, constructing an approximation scheme such that the entropy, $H(A)$, is maximized intrinsically leads to a least-biased approximation scheme. As shown in [4], imposing this constraint leads to the unique approximation scheme given by the solution of the following maximum entropy (ME) program:

$$\begin{aligned} (ME) \qquad \text{maximize} \quad & H(\mathbf{p}) = -\sum_{a=1}^{N} p_a \log p_a \\ \text{subject to} \quad & p_a \geq 0,\ a = 1, \ldots, N \\ & \sum_{a=1}^{N} p_a = 1 \\ & \sum_{a=1}^{N} \mathbf{x}_a p_a = \mathbf{x} \end{aligned}$$

From a purely information-theoretical viewpoint, shape functions p_a resulting from the ME program are optimal, in a *least-biased* sense, convex approximation schemes. Since this program maximizes the information entropy throughout the domain, the resulting shape functions, p_a, are called global maximum-entropy shape functions.

Local Maximum Entropy Schemes

While global maximum-entropy shape functions are optimal in a *least-biased* sense, when some information about the solution is known it is convenient to incorporate this information into the approximation scheme. In particular, while the ME shape functions are completely nonlocal, from the point of view of basic physics it is tacitly understood that the state at a given point is more strongly influenced by those neighbors in close proximity rather than those further away. In continuum mechanics this concept amounts to the principle of Saint-Venant. In an attempt to embed this concept of locality into the approximation scheme, in [4], the following measure, defining a width for a shape function, p_a, was introduced:

$$w[p_a] = \int_{\Omega} p_a(\mathbf{x})|\mathbf{x} - \mathbf{x}_a|^2 d\mathbf{x}, \tag{2.9}$$

where $\Omega = \text{conv}\, X$. Here, equation 2.9 amounts to the second moment of the shape function, p_a, about $\mathbf{x}_a$. Then the most local approximation scheme is that which minimizes the total width given by

$$\begin{aligned} W[\mathbf{p}] &= \sum_{a=1}^{N} w[p_a] \\ &= \int_{\Omega} \sum_{a=1}^{N} p_a(\mathbf{x})|\mathbf{x} - \mathbf{x}_a|^2 d\mathbf{x}, \end{aligned} \tag{2.10}$$

subject to the zero and first order consistency constraints in equation 1.9. Since the functional in equation 2.10 does not involve the shape function derivatives, its minimization can be performed pointwise. This results in the following linear program attributed to Rajan [71]:

$$\begin{aligned} (RAJ) \qquad \text{For fixed } \mathbf{x} \text{ minimize} \quad & U(\mathbf{x}, \mathbf{p}) \equiv \sum_{a=1}^{N} p_a |\mathbf{x} - \mathbf{x}_a|^2 \\ \text{subject to} \quad & p_a \geq 0,\ a = 1, \ldots, N \\ & \sum_{a=1}^{N} p_a = 1 \\ & \sum_{a=1}^{N} \mathbf{x}_a p_a = \mathbf{x} \end{aligned}$$

In [71] it was demonstrated that the program (RAJ) has a solution if and only if $\mathbf{x} \in \text{conv}\, X$. Here, the function $U(\mathbf{x}, \cdot)$ is not strictly convex (it is linear) and the solution is not, in general, unique. Furthermore, in [71], it was noted that if the nodes are in general positions (i.e. none of the $(dim + 1)$ nodes in X are cospherical), then the program (RAJ) has a unique solution. This solution corresponds to piecewise affine shape functions supported on the unique Delaunay triangulation of the node set X.

In general it is not possible to achieve both goals of maximizing the entropy while minimizing the width for a convex approximation scheme. However, one can construct the LME approximation schemes as the *Pareto* optima between these competing objectives of maximizing both the entropy and locality for the approximation scheme. Here, *Pareto* optima is taken in the sense that there is no better optima than the selected solution. This leads to the program

$$
\begin{aligned}
(LME)_\beta \qquad \text{For fixed } \mathbf{x} \text{ minimize} \quad & f_\beta(\mathbf{x},\mathbf{p}) \equiv \beta U(\mathbf{x},\mathbf{p}) - H(\mathbf{p}), \\
\text{subject to} \quad & p_a \geq 0, \ a = 1, \ldots, N, \\
& \sum_{a=1}^{N} p_a = 1, \\
& \sum_{a=1}^{N} \mathbf{x}_a p_a = \mathbf{x},
\end{aligned}
$$

where $\beta \in [0, \infty]$ is the *Pareto* optimal parameter. Here, larger values of β lead to more local shape functions. For the extreme case, $\beta = \infty$, one recovers the linear program (RAJ). Also, for $\beta = 0$ the (ME) program is recovered. In [4], by considering the dual of the problem the following proposition provides a practical method for evaluating the (LME) shape functions.

Proposition 2.1: *Let aff X be the affine hull of X. Suppose aff$X = \mathbb{R}^d$, $\mathbf{x} \in$ int(conv X), and $\beta \in [0, \infty)$. Then the unique solution of the local max-ent problem $(LME)_\beta$ is*

$$
p_{\beta a}(x) = \frac{1}{Z(\mathbf{x}, \lambda^*(\mathbf{x}))} \exp\left[-\beta|\mathbf{x} - \mathbf{x}_a|^2 + \lambda^*(\mathbf{x}) \cdot (\mathbf{x} - \mathbf{x}_a)\right] \ , \ a = 1, \ldots, N,
$$

where

$$\lambda^*(\mathbf{x}) = \arg\min_{\lambda\in\mathbb{R}^d} \log Z(\mathbf{x},\lambda),$$

and $Z(\mathbf{x},\lambda)$ *is the partition function given as*

$$Z(\mathbf{x},\lambda) \equiv \sum_{i=1}^{N} \exp\left[-\beta|\mathbf{x}-\mathbf{x}_i|^2 + \lambda(\mathbf{x})\cdot(\mathbf{x}-\mathbf{x}_i)\right].$$

In addition, in [4], it was demonstrated that the minimizer $\lambda^*(\mathbf{x})$ is unique and that the solution to this program, $(LME)_\beta$, for the shape functions is efficient and robust. Furthermore, consider Fig. 2.3, which shows the effect of the parameter $\mu = \beta h^2$ on the locality of the shape functions. Specifically, note that as the locality of the approximation increases with increasing the user defined parameter, β. Thus, using the parameter, β provides a seamless manner for coupling non-local shape functions with localized finite element shape functions.

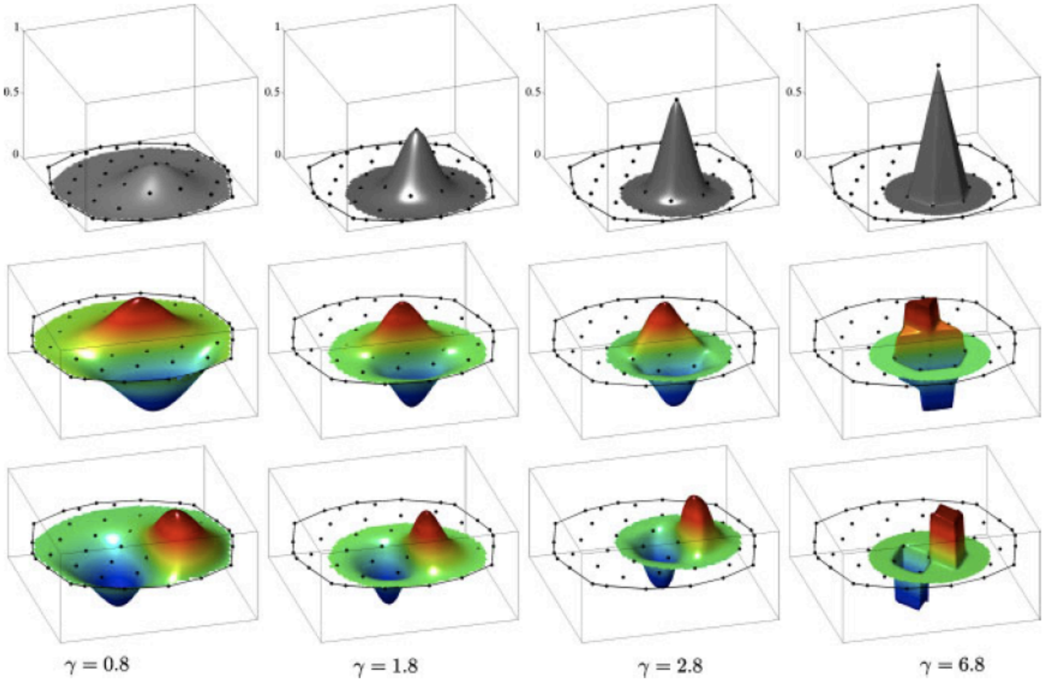

Figure 2.3: Local maximum-entropy shape functions and spatial derivatives (arbitrary scale) for a two-dimensional arrangement of nodes using several values of $\mu = \beta h^2$.

2.4 Summary

While the area of particle and meshfree method has been an active area of research for several decades, the state of mathematical understanding in terms of completeness and convergence is still in an infancy stage. While many methods have been proposed, the lack of maturity and acceptance of meshfree methods is due to several open questions regarding numerical integration of the Galerkin weak form, imposing essential boundary conditions, as well as the lack of mathematical proofs for convergence and stability. Practical issues involving numerical integration, handling boundary conditions, as well as other method-specific requirements are highly active areas of numerical research. In chapter 4, a novel meshfree method founded in the theory of optimal transportation is presented. In this OTM method, LME shape functions are chosen due to their noted salient features.

Numerical integration in meshfree methods has been an active topic of research. In contrast to finite elements, integration in meshfree methods is not exact. Methods that are based on the weak form typically use one of three different types of integration schemes: nodal integration, stress-point integration, or integration based on a background mesh (i.e. Gauss quadrature). From a computational point of view, nodal integration is the easiest and cheapest way to formulate the discrete equations. However, as noted for the SPH and EFG method, nodal integration leads to a tensile instability emanating from rank deficiency (as in reduced finite elements) [30]. While adding stress points for numerical integration has been found to alleviate tensile instabilities for Lagrangian kernels, introducing stress points requires some notion of a mesh [57] and hence constrains the allowable deformation. In general, the most accurate way to obtain the governing equations is by recourse to a grid-based numerical quadrature rule (i.e. Gauss quadrature). Typically, meshfree methods use a background mesh for numerical integration typically with more quadrature points as compared with Gaussian integration for finite elements [50].

Since, meshfree shape functions do not, in general, possess the Kronecker-delta property (i.e. $\phi_I(x_J) = \delta_{IJ}$), imposing essential boundary conditions poses a significant challenge. Many methods for enforcing essential boundary conditions have been suggested, including applying Lagrange multipliers [52] in the variational weak form, penalty methods [72], coupling to finite elements [59, 58, 73], as well as employing Nitsche's method (i.e. restore consistency to penalty formulation) [74, 75, 76]. In stark contrast, since LME shape functions possess a weak Kronecker-delta property, applying essential boundary conditions is analogous to the FEM and trivial to implement.

Coupling particle or meshfree methods with finite elements, if robust, provides an ideal framework for solving strongly coupled fluid-structure interaction problems. In addition to FSI simulations there are a number of applications (i.e. fracture or crack propagation) where coupling meshfree methods with finite elements is advantageous. While computing FEM shape functions is faster than meshfree shape functions, meshfree methods can provide additional versatility to traditional numerical methods (i.e. FEM) in areas of anticipated crack propagation, large deformations, or other discrete phenomenon where a mesh is inconvenient. Indeed, there has been significant attention attributed to coupling meshfree methods, specifically the SPH and EFG methods, with finite elements [77, 78, 35]. Here, the approaches are analogous to the methods employed in enforcing essential boundary conditions (i.e. penalty method, Lagrange multipliers, Nitsche's method). Often, transition regions are specified by the user *a priori*, where special rules are employed for blending the shape functions via ramp functions to restore consistency [53]. Again, this is in sharp contrast with LME, which by choosing the parameter β large enough (i.e. consider Fig. 2.3), or by using exactly $d + 1$ nodes (where d is the dimension) for constructing the shape function, LME shape functions either seamlessly blends with or reduces to linear finite elements.

Chapter 3

Optimal Mass Transportation

This chapter reviews the key concepts of optimal mass transportation theory to provide a background for the extension to fluid flows, detailed in the next chapter. In addition to detailing the basic equations, the discrete forms are derived to provide a basis for presenting the intricacies of the discretization imparted for fluid flows. In the next chapter, the Benamou-Brenier variational formalism is extended to general fluid flows leading to the development of a new meshfree method: the OTM method.

The analysis of mass transportation was first considered from a civil engineering point of view by Monge in 1781 [79]. Here, the problem was formulated in [2] as a means of obtaining the best manner (relative to a specified cost) of displacing parcels of mass from one point to another. For example, consider figure 3.1, which depicts the concept of transporting mass (i.e. moving a pile of dirt to fill a hole).

Following the work of Kantorovich in 1942 [80] a significant amount of attention has

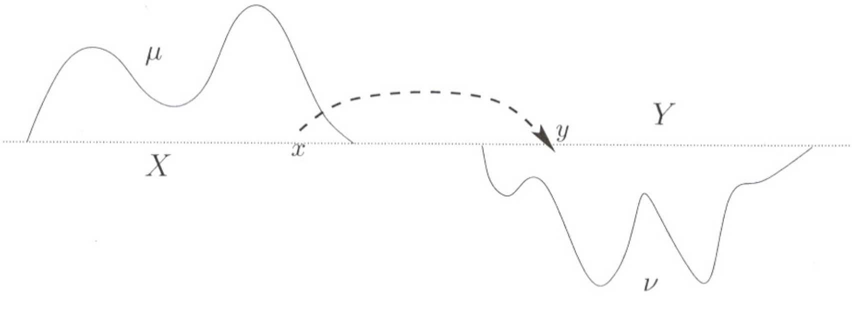

Figure 3.1: Mass transportation problem [2].

been garnered toward this optimal mass transport problem including [81, 82, 83, 84, 85, 86, 87].

3.1 Mass Transportation Problem

We begin by considering the motion of non-interacting particles to present the connection between optimal transportation and Lagrangian dynamics of continuous distributions of mass. In particular, we draw from the theory of optimal transportation to formulate the kinetic energy of the system and, by extension, the action directly in terms of its mass density. This defines a minimum principle whose minimizers are the time histories of the mass density. From this minimum variational principle, a temporal and spatial discretization for the non-interacting flow of particles is introduced leading to the discrete set of equations of motion.

3.1.1 Optimal Mass Transportation

Optimal transportation theory derives its importance from the fact that it supplies a powerful and useful mathematical foundation for a number of areas of mechanics and physics. A thorough and rigorous account of the theory may be found in the monographs of Evans [88] and Villani [2]. Here, for simplicity, mass densities are formally used in lieu of more rigorous measure-theoretical notation, which is nevertheless evident from the expressions.

Consider the transportation of mass as the flow of an inviscid fluid comprised of non-interacting particles in $\mathbb{R}^n$ at zero temperature. Then, the motion of the fluid over a time interval $[a, b]$ is governed by the coupled equations:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \tag{3.1a}$$

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) = 0. \tag{3.1b}$$

Here, ρ is the mass density of the fluid and $\mathbf{v}$ is the velocity field. equation 3.1a is the equation of conservation of mass and equation 3.1b is the equation of conservation of linear momentum. Furthermore, assume that the fluid has finite total mass,

$$M = \int \rho d\mathbf{x}, \tag{3.2}$$

and that mass does not *leak* to infinity,

$$\lim_{R\to\infty} \int_{\partial B_R} \rho\mathbf{v} \cdot \mathbf{n} dx = 0, \tag{3.3}$$

where B_R is the ball of radius R. Then, in considering the time rate of change of mass,

$$\dot{M} = \lim_{R\to\infty} \int_{B_R} \frac{\partial \rho}{\partial t} d\mathbf{x} = -\lim_{R\to\infty} \int_{B_R} \nabla \cdot (\rho\mathbf{v}) d\mathbf{x} = -\lim_{R\to\infty} \int_{\partial B_R} \rho\mathbf{v} \cdot \mathbf{n} d\mathbf{x} = 0. \tag{3.4}$$

It follows that the total mass of the fluid, M, remains constant during the flow. Furthermore, assume that the initial and final mass densities are specified by

$$\rho(x, a) = \rho_a(x), \tag{3.5a}$$

$$\rho(\mathbf{x}, b) = \rho_b(\mathbf{x}). \tag{3.5b}$$

Then, this outlined problem is analogous to the *transportation problem* of finding the flow that transports the initial mass density, ρ_a, to the final one, ρ_b.

3.1.2 Benamou-Brenier Variational Formulation

The transportation problem just enunciated can be recast as an *optimal transportation* problem. Benamou and Brenier [89] noted that problem in equations 3.1, and 3.5 admits the variational characterization:

$$\inf_{(\rho,\mathbf{v})\in X} A(\rho,\mathbf{v}) \tag{3.6a}$$

$$\text{subject to: } \frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{v}) = 0, \tag{3.6b}$$

where the action over the time interval (a, b) is given by,

$$A(\rho,\mathbf{v}) = \int_a^b K(\rho,\mathbf{v})\,dt \tag{3.7}$$

, in terms of the kinetic energy,

$$K(\rho,\mathbf{v}) = \int \frac{\rho}{2}|\mathbf{v}|^2\,d\mathbf{x}. \tag{3.8}$$

Here, the natural space of solutions is

$$X = \left\{(\rho,\mathbf{v}) \in C\left([a,b]; L^1(\mathbb{R}^n;[0,\infty))\right) \times C\left([a,b]; L^2(\mathbb{R}^n;\mathbb{R}^n)\right)\right\}.$$

Recall that $L^1(\mathbb{R}^n)$ is the Lebesgue space of integrable functions over $\mathbb{R}^n$. Thus $\{d\mu = \rho\,dx,\ \rho \in L^1(\mathbb{R}^n;[0,\infty))\}$ is the set of measures that are absolutely continuous with respect to the Lebesgue measure. Physically, the restriction of ρ to $L^1(\mathbb{R}^n;[0,\infty))\big)$ ensures that every subset of $\mathbb{R}^n$ with a well-defined volume can be assigned a well-defined, non-negative mass. In addition, $L^2(\mathbb{R}^n;\mathbb{R}^n)$ is the Lebesgue space of square-integrable vector-valued functions over $\mathbb{R}^n$. Physically, the restriction of v to $L^2(\mathbb{R}^n;\mathbb{R}^n)$, in conjunction with the restriction of ρ to $L^1(\mathbb{R}^n;[0,\infty))$, ensures that the velocity fields have finite kinetic energy. Finally,$C\big([a,b];L^1(\mathbb{R}^n;[0,\infty))\big)$ and $C\big([a,b];L^2(\mathbb{R}^n;\mathbb{R}^n)\big)$ are the space of time-continuous functions taking values in $L^1(\mathbb{R}^n;[0,\infty))$ and $L^2(\mathbb{R}^n;\mathbb{R}^n)$, respectively.

The minimization problem, equation 3.6, can be formally verified to be equivalent to equations 3.1 and 3.5). To this end, the constraint, in equation 3.1b, in enforced by means of a Lagrange multiplier, ϕ. This leads to the extended action given by,

$$A(\rho, \mathbf{v}, \phi) = \int_a^b \left\{ \int\!\!\int \left[\frac{\rho}{2} |\mathbf{v}|^2 + \big(\rho_{,t} + \nabla \cdot (\rho \mathbf{v})\big)\phi \right] d\mathbf{x} \right\} dt. \tag{3.9}$$

Rendering the action, A, stationary with respect to the three fields $(\rho, \mathbf{v}, \phi)$ yields

$$\begin{gathered} 0 = \delta A(\rho, \mathbf{v}, \phi) = \int_a^b \left\{ \int\!\!\int \left[\frac{\delta\rho}{2} |v|^2 + \big(\delta\rho_{,t} + \nabla \cdot (\delta\rho \mathbf{v})\big)\phi \right] \mathbf{dx} \right\} dt + \\ \int_a^b \left\{ \int\!\!\int \big[\big(\rho \mathbf{v} \cdot \delta \mathbf{v} + \nabla \cdot (\rho \delta \mathbf{v})\phi\big)\big] d\mathbf{x} \right\} dt + \int_a^b \left\{ \int\!\!\int \big[\big(\rho_{,t} + \nabla \cdot (\rho \mathbf{v})\big)\delta\phi\big] \, d\mathbf{x} \right\} dt \end{gathered} \tag{3.10}$$

Integrating by parts, with the aid of equation 3.3, and rendering all admissible variations stationary gives,

$$\frac{1}{2}|v|^2 - \phi_{,t} - \nabla\phi \cdot v = 0, \tag{3.11a}$$

$$\rho \mathbf{v} - \rho \nabla \phi = 0, \tag{3.11b}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0. \tag{3.11c}$$

The third of these equations is the continuity equation 3.1b. In addition, from equation 3.11b it follows that

$$\mathbf{v} = \nabla \phi \ \text{ in } \operatorname{supp} \rho. \tag{3.12}$$

Here, $\operatorname{supp} f$ denotes the support of a measurable function f. Thus, the stationary points of the action equation 3.9 correspond to potential flow. Using this relation, equation 3.11a simplifies to

$$\frac{\partial \phi}{\partial t} + \frac{1}{2}|\mathbf{v}|^2 = 0. \tag{3.13}$$

Then, equation 3.1b can be readily verified by taking gradients of equation 3.13, multiplying by ρ and using equations 3.11c and 3.12.

The Benamou and Brenier variational characterization, i.e., equation 3.6, of the transport problem in equations 3.1 and 3.5 admits a compelling reformulation within the context of the Monge-Kantorovich optimal transportation framework. Specifically, Benamou and Brenier [89, 2], showed that the minimizers of the action in equation 3.7 are given in terms of McCann's *displacement* $interpolation$ [2]:

$$\varphi(\mathbf{x}, t) = \frac{b-t}{b-a}\mathbf{x} + \frac{t-a}{b-a}T(\mathbf{x}), \tag{3.14}$$

through the relations,

$$v(\mathbf{x}, t) = \frac{\partial \varphi}{\partial t}(\varphi^{-1}(\mathbf{x}, t), t)\,, \tag{3.15a}$$

$$\rho(\mathbf{x}, t) = \rho_a\big(\varphi^{-1}(\mathbf{x}, t)\big) / \det\big(\nabla\varphi(\varphi^{-1}(\mathbf{x}, t), t)\big)\,, \tag{3.15b}$$

where $T = \varphi(\cdot, b)$ is the optimal transference mass of ρ_a into ρ_b in the sense of the cost function

$$I(T) = \int |T(\mathbf{x}) - \mathbf{x}|^2 \rho_a(\mathbf{x})\, d\mathbf{x}, \tag{3.16}$$

i.e.,

$$\begin{aligned} \mathcal{T}_2(\rho_a, \rho_b) &\equiv \inf\left\{I(T) :\ T \text{ measurable},\ \rho_a(\mathbf{x}) = \rho_b\left(T(\mathbf{x})\right)\det\left(\nabla T(\mathbf{x})\right)\right\} \\ &= \inf\left\{(b-a)A(\rho, \mathbf{v}) :\ (\rho, \mathbf{v}) \in V(\rho_a, \rho_b)\right\}, \end{aligned} \tag{3.17}$$

where $V(\rho_a, \rho_b)$ is the set of pairs $(\rho, \mathbf{v}) \in X$ such that $\cup_{t\in[a,b]}\mathrm{supp}\,\rho(\cdot, t)$ is bounded. Furthermore, equation 3.1a is satisfied weakly in a distributional sense, i.e., $\rho(\cdot, a) = \rho_a$ and $\rho(\cdot, b) = \rho_b$.

Considering equation 3.17, the minimum cost of transportation is directly related to the Wasserstein distance:

$$d_W(\rho_a, \rho_b) = \left\{ \inf_{\sigma\in\Gamma(\rho_a,\rho_b)} \int\int |\mathbf{x} - y|^{\mathbf{2}} \sigma(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} \right\}^{1/2}, \tag{3.18}$$

where the infimum is taken over the space of Radon measures of mass, M, with finite second moments and marginals:

$$\int \sigma(\mathbf{x}, \mathbf{y}) d\mathbf{y} = \rho_a(\mathbf{x}), \tag{3.19a}$$

$$\int \sigma(\mathbf{x}, \mathbf{y}) d\mathbf{x} = \rho_b(\mathbf{y}). \tag{3.19b}$$

Then, from equation 3.18

$$\mathcal{T}_2(\rho_a, \rho_b) = d_W^2(\rho_a, \rho_b). \tag{3.20}$$

Thus, the minimum cost of optimal transportation is given directly by the Wasserstein distance between the initial and final mass densities.

A theorem by Ambrosio in [2] demonstrates how to take variations of the minimum cost of transportation $\mathcal{T}_2(\rho_a, \rho_b)$ with respect to ρ_b. Specifically, suppose that $\rho_a, \rho_b \in L^1(\mathbb{R}^n; [0, \infty))$ and have finite second moments. Let $\rho(\mathbf{x}, t)$, $t \in (b - \epsilon, b + \epsilon)$, be a path in $L^1(\mathbb{R}^n; [0, \infty))$ of mass densities with finite second moments such that $\rho(\cdot, 0) = \rho_b$. Then, consider the continuity equation for some velocity field ξ that is C^1 and globally bounded, given by

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \xi) = 0. \tag{3.21}$$

Then,

$$\frac{d}{dt}\mathcal{T}_2(\rho_a, \rho)_{|t=0} = 2\int \langle T(x) - x, \xi(T(x))\rangle d\rho_a(x), \tag{3.22}$$

where T is the optimal transference mapping from ρ_a to ρ_b.

3.2 Numerical Discretization

This section provides the numerical discretization for the optimal mass transportation scheme presented above. As expected for non-interacting masses, the scheme leads to the evolution of the position and velocity as a purely *ballistic* update. Building on the method for discretizing the optimal transportation of mass, chapter 4 extends these developments to numerical simulations of general fluid flows. In this sense, the developments in this section are used for motivating the intricacies involved in extending the discretization scheme to general fluid flows.

3.2.1 Time Discretization

First, we turn to the question of time discretization of the action in equation 3.7. To this end, let $t_0 = a < t_1 < \cdots < t_N = b$ be a discretization of the time interval $[a, b]$. Also, recall that $(1/2)d_W^2(\rho_a, \rho_b)/(b - a)$ gives the exact minimum of the action $A(\rho, v)$ over the entire time interval $[a, b]$. Building upon this identity, we can define the semidiscrete action directly in terms of densities as:

$$A_d(\rho_1, \ldots, \rho_{N-1}) = \sum_{k=0}^{N-1} \frac{1}{2}\frac{\mathcal{T}_2(\rho_k, \rho_{k+1})}{t_{k+1} - t_k}. \tag{3.23}$$

If no further approximation is introduced, then the infimum of A_d over $\{\rho_1, \ldots, \rho_{N-1}\} \in$

$[L^1(\mathbb{R}^n;[0,\infty))]^{N-1}$ is equivalent to $(1/2)\mathcal{T}_2(\rho_a,\rho_b)/(b-a)$ as expected. Furthermore, this scheme is exact! Here, the discrete motion consists of incremental transference maps $\varphi_{k\to k+1}$ transporting ρ_k into ρ_{k+1} over the time internal $[t_k, t_{k+1}]$ in an optimal, with respect to the cost function equation 3.16, sense.

Then, the discrete equations of motion follow by rendering the discrete action stationary. Taking variations of equation 3.23 with respect to ρ_k, with the aid of equation 3.22, leads to

$$\langle \delta A_d, \xi_k \rangle = \int \left\{ \rho_k \left(\frac{\varphi_{k\to k+1}(\mathbf{x}) - \mathbf{x}}{t_{k+1} - t_k} + \frac{\varphi_{k\to k-1}(\mathbf{x}) - \mathbf{x}}{t_k - t_{k-1}} \right) \cdot \xi_k \right\} dx, \tag{3.24}$$

where,

$$\varphi_{k\to k-1} = \varphi_{k-1\to k}^{-1}. \tag{3.25}$$

In equation 3.24, $\varphi_{k\to k+1}$ denotes the optimal transference mapping from ρ_k to ρ_{k+1}. Then, the mass-density update is defined as

$$\rho_{k+1} \circ \varphi_{k\to k+1} = \rho_k / \det\left(\nabla \varphi_{k\to k+1}\right). \tag{3.26}$$

Here, it is specifically noted that these mass-density updates sharply differ from conventional Eulerian algorithms, which rely on some direct time-discretization of the continuity equation. Specifically, the mass-density updates in equation 3.26 are geometrically exact!

Integrating by parts equation 3.24, using equation3.3, and enforcing stationarity with respect to to the variations in ξ_k gives:

$$\rho_k \left(\frac{\varphi_{k\to k+1} - \mathrm{id}}{t_{k+1} - t_k} + \frac{\varphi_{k\to k-1} - \mathrm{id}}{t_k - t_{k-1}} \right) = 0, \tag{3.27}$$

where id is the identity mapping. These equations are jointly satisfied by setting

$$\varphi_{k\to k+1}(\mathbf{x}) = \varphi(\varphi^{-1}(\mathbf{x}, t_k), t_{k+1}), \tag{3.28}$$

where φ is given McCann's displacement interpolation in equation 3.14. This shows, as expected, that the discretization in equation 3.24 is indeed exact.

3.2.2 Spatial Discretization

Next, we turn to the question of spatial discretization of the semidiscrete action given in equation 3.23. A natural and computationally convenient spatial discretization may be imparted by considering mass densities of the type

$$\rho_{h,k}(\mathbf{x}) = \sum_{p=1}^{M} m_{p,k}\delta\big(\mathbf{x} - \mathbf{x}_{p,k}\big), \tag{3.29}$$

where $\mathbf{x}_{p,k}$ represents the position at time t_k of a *material point* with mass m_p and $\delta\big(\mathbf{x} - \mathbf{x}_{p,k}\big)$ is the Dirac-delta distribution centered at $\mathbf{x}_{p,k}$. Then, from [2], a fully discrete action is obtained by inserting equation 3.29 into equation 3.23. This, in turn, provides the definition of the discrete transportation problem.

By considering mass distributions of the form equation 3.29, the original space of solutions $L^1(\mathbb{R}^n; [0,\infty))$ has been expanded to a larger space, $\mathcal{M}(\mathbb{R}^n)$, of Radon measures. In computing the minimum cost of transportation $\mathcal{T}_2(\rho_{h,k}, \rho_{h,k+1})$ between two consecutive discrete mass densities, the incremental optimal transference mappings $\varphi_{h,k\to k+1}$ transporting $\rho_{h,k}$ into $\rho_{h,k+1}$ are interpreted simply as rearrangements of the point in the set $\{\mathbf{x}_{1,k}, \ldots, \mathbf{x}_{M,k}\}$ into the point set $\{\mathbf{x}_{1,k+1}, \ldots, \mathbf{x}_{M,k+1}\}$. In addition, the incremental mass conservation relation in equation 3.26 must be understood in a weak or distributional sense. Specifically, we require that, for all test functions η,

$$\int \rho_k(\mathbf{x})\eta(\mathbf{x})\, d\mathbf{x} = \int \rho_{k+1}(\mathbf{y})\eta\big(\varphi^{-1}_{k\to k+1}(\mathbf{y})\big) d\mathbf{y}. \tag{3.30}$$

For discrete mass distributions of the form in equation 3.29 the above requirement reduces to

$$\sum_{p=1}^{M} m_{p,k}\eta(\mathbf{x}_{p,k}) = \sum_{p=1}^{M} m_{p,k+1}\eta(\mathbf{x}_{p,k}), \tag{3.31}$$

which must be satisfied for all test functions η. This furnishes the relation:

$$m_{p,k} = m_{p,k+1} = m_p. \tag{3.32}$$

Thus, the material points must have constant mass. Then, the fully discrete action takes the form:

$$A_h(\rho_{h,1}, \ldots, \rho_{h,N-1}) = \sum_{k=0}^{N-1} \sum_{p=1}^{M} \frac{m_p}{2} \frac{|\mathbf{x}_{p,k+1} - \mathbf{x}_{p,k}|^2}{t_{k+1} - t_k}, \tag{3.33}$$

providing the semi-discrete action of a system of non-interacting mass particles. Finally, the corresponding discrete Euler-Lagrange equations are given by,

$$\frac{\mathbf{x}_{p,k+1} - \mathbf{x}_{p,k}}{t_{k+1} - t_k} - \frac{\mathbf{x}_{p,k} - \mathbf{x}_{p,k-1}}{t_k - t_{k-1}} = 0, \tag{3.34}$$

or, equivalently:

$$\mathbf{x}_{p,k+1} = \mathbf{x}_{p,k} + (t_{k+1} - t_k)\frac{\mathbf{x}_{p,k} - \mathbf{x}_{p,k-1}}{t_k - t_{k-1}}. \tag{3.35}$$

Equation 3.35 provides an update for the positions of the material points. In this update the velocity of material point, located at $\mathbf{x}_p$, at time t_k is given by,

$$\mathbf{v}_{p,k} \equiv \frac{\mathbf{x}_{p,k} - \mathbf{x}_{p,k-1}}{t_k - t_{k-1}}. \tag{3.36}$$

Using this definition, equation 3.35 reduces to the particularly simple form:

$$\mathbf{x}_{p,k+1} = \mathbf{x}_{p,k} + (t_{k+1} - t_k)\mathbf{v}_{p,k}. \tag{3.37}$$

Simply stated, the update in equation 3.37 defines the *ballistic* motion of a material point.

Specifically, the *ballistic* update linearly projects the initial positions $\mathbf{x}_{p,0}$ to the final positions $\mathbf{x}_{p,N}$ through the material point's velocity, $\mathbf{v}_p$.

The next chapter extends the concepts outlined in this chapter to general fluid flows. While the evolution of the equations of motion for the optimal transportation of non-interacting mass is consistent with *ballistic* motion, introducing equations of states for modeling general fluid flows effectively changes the update to derive from the competition between inertia and internal energy. Furthermore, deriving the discrete forms, combined with the use of LME shape functions leads to the formulation of the OTM method for numerically simulating general fluid flows.

Chapter 4

The Optimal Transportation Meshfree (OTM) Method for Fluid Flows

This chapter develops the proposed OTM method for numerically simulating generalized fluid flows. By generalizing the Benamou-Brenier [89] differential formulation for optimal mass transportation, we extend the optimal mass transportation framework for non-interacting fluids, developed in chapter 3, to model compressible Euler and Navier-Stokes fluid flows. For these systems, inertia competes with free energy and viscosity in determining the flow. Conveniently, the free energy of a fluid can be expressed directly in terms of its mass density and, in the case of non-interacting fluids, equation 3.23. Thus, the resulting semidiscrete action can be expressed directly in terms of the mass density.

The proposed OTM framework leads to a multi-field variational characterization of fluid flows including viscosity, equations of state, as well as general geometries and boundary conditions. Specifically, the governing variational principle is well suited to discretization by a combination of conforming interpolation of the velocity field and pointwise sampling of the local material state. Due to the salient attributes presented in chapter 2, LME shape functions are chosen for interpolating the velocity field.

4.1 Barotropic Fluid Flows

For simplicity, we shall assume that the flow is $barotropic$ in that there is a functional relation between P and ρ independent of temperature of the form

$$p = \rho^2 \frac{\partial f}{\partial \rho}(\rho), \tag{4.1}$$

for some potential function $f(\rho)$.

For example, an isothermal flow is barotropic with $f(\rho) = e(\rho, \theta)$. For barotropic flows we introduce the dual function,

$$f^*(\kappa) = \sup_{\rho \geq 0} \big\{\rho\big(\kappa - f(\rho)\big)\big\}, \tag{4.2}$$

with the property that

$$\rho = \frac{\partial f^*}{\partial \kappa}(\kappa). \tag{4.3}$$

4.1.1 Ideal Gas Equation of State

For an ideal gas, the incomplete equation of state is given by,

$$pV = nR\theta = Nk_B\theta, \tag{4.4}$$

where n is the number of moles, $R = 8.3145$ J/mol K is the universal gas constant, N is the number of molecules, $k_B = 1.38066 \times 10^{-23}$ J/K $= 8.617385 \times 10^{-5}$ eV/K, k $= R/N_A$ is Boltzmann's constant, and $N_A = 6.0221 \times 10^{23}$ /mol is Avogadro's number. Alternatively, the ideal gas relation is given by,

$$p = \frac{k_B\theta}{m}\rho, \tag{4.5}$$

where m is the mass of one molecule of gas. Integrating, the free energy density per unit mass of an ideal gas follows as:

$$e(\rho, \theta) = \frac{k_B \theta}{m} \log \frac{\rho}{\rho_0}. \tag{4.6}$$

4.1.2 Isothermal Fluid Flows

Thus, for isothermal flows the barotropic potential is given by

$$f(\rho) = \frac{k_B \theta}{m} \log \frac{\rho}{\rho_0}, \tag{4.7}$$

where the temperature, θ, remains constant. Then, the Euler-Lagrange equation corresponding to equation4.2 is given by

$$\frac{\partial}{\partial \rho}\{\rho\big(\kappa - f(\rho)\big)\} = \kappa - \frac{k_B \theta}{m}\big(\log \frac{\rho}{\rho_0} + 1\big) = 0, \tag{4.8}$$

or,

$$\rho = \frac{\rho_0}{e} \exp\left(\frac{m\kappa}{k_B \theta}\right) \geq 0. \tag{4.9}$$

Finally, the dual is given by

$$f^*(\kappa) = \frac{k_B \theta \rho_0}{em} \exp\left(\frac{m\kappa}{k_B \theta}\right). \tag{4.10}$$

4.1.3 Adiabatic Fluid Flows

For reversible, adiabatic flows (i.e. isentropic) of an ideal gas, the following polytropic relationship characterizes the behavior of the fluid:

$$p = C\rho^\gamma, \quad \rho \geq 0 \tag{4.11}$$

Here, C is a constant related to the initial state and $\gamma = c_p/c_v = 1 + R/c_v$ is the ratio of specific heats, c_p andc_v (constant pressure and constant volume, respectively). Examples of poytropic processes include the compression and expansion of an ideal gas. In this case,

$$f(\rho) = \frac{C}{\gamma - 1}\rho^{\gamma-1}, \quad \rho \geq 0, \tag{4.12}$$

The Euler-Lagrange equation corresponding to equation 4.2 is given by,

$$\frac{\partial}{\partial\rho}\{\rho\big(\kappa - f(\rho)\big)\} = \kappa - \frac{C\gamma}{\gamma - 1}\rho^{\gamma-1} = 0, \tag{4.13}$$

which leads to,

$$\rho = \begin{cases} \left[(\gamma - 1)\kappa/(C\gamma)\right]^{1/(\gamma-1)}, & \text{if } \kappa \geq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{4.14}$$

Then, the dual is given by

$$f^*(\kappa) = \begin{cases} C\left[(\gamma - 1)\kappa/(C\gamma)\right]^{\gamma/(\gamma-1)}, & \text{if } \kappa \geq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{4.15}$$

4.1.4 Incompressible Fluid Flow

For an incompressible fluid, the potential function is given by

$$f^*(\kappa) = \rho_0\kappa, \tag{4.16}$$

where ρ_0 is the constant mass density. This fact can be verified by noting,

$$\rho = \frac{\partial f^*}{\partial\kappa}(\kappa) = \rho_0. \tag{4.17}$$

4.2 OTM Method for Euler Fluid Flows

We begin by considering the flow of a compressible inviscid fluid in $\mathbb{R}^n$. The motion of the fluid over a time interval $[a, b]$ is governed by the coupled equations

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \tag{4.18a}$$

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v} + p\mathbf{I}) = \rho b, \tag{4.18b}$$

where p is the pressure, b are body forces. As in the previous optimal transportation problem in equation 3.1 we assume that mass does not leak to infinity, equation 3.3. Also, from equation 4.18a it follows that the total mass of the system remains constant throughout the flow. Again, we suppose that the initial and final mass densities are provided as in equation 3.5. Then, this problem translates to a *transportation problem* of finding the flow that transports the initial mass density, ρ_a, to the final one, ρ_b.

In addition, we assume that the body forces derive from a scalar potential, u, given by:

$$\mathbf{b} = \nabla u. \tag{4.19}$$

Under these conditions, the action of the fluid differs from the bare action, i.e., equation 3.7, in that the fluid has a free energy and moves under the action of a body-force potential. Adding the corresponding terms to equation 3.7 the action assumes the form:

$$A(\rho, \mathbf{v}) = \int_a^b \left\{ \int \left[\frac{\rho}{2} |\mathbf{v}|^2 - \rho(f(\rho) - u) \right] d\mathbf{x} \right\} dt. \tag{4.20}$$

In analogy to the case of optimal transportation of non-interacting mass, we expect the flow to render the action in equation 4.20 stationary (no longer necessarily a minimum!) under the constraint of the continuity equation, equation 4.18a. This defines the stationary-point problem given by,

$$\delta A(\rho, \mathbf{v}) = 0\,, \tag{4.21a}$$

$$\text{subject to: } \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{4.21b}$$

For the OTM method, this stationarity principle replaces the Benamou and Brenier minimum principle equation 3.6. The stationary-point problem, equation 4.21, is equivalent to equations 3.5 and 4.18, which can be verified by enforcing the continuity constraint in equation 4.18a using a Lagrange multiplier, ϕ. This process leads to the extended action:

$$A(\rho, \mathbf{v}, \phi) = \int_a^b \left\{ \int \left[\frac{\rho}{2} |\mathbf{v}|^2 - \rho(f(\rho) - u) + \big(\rho_{,t} + \nabla \cdot (\rho \mathbf{v})\big)\phi \right] d\mathbf{x} \right\} dt. \tag{4.22}$$

Then, rendering this functional stationary gives

$$\begin{gathered} 0 = \delta A(\rho, v, \phi) = \\ \int_a^b \left\{ \int \left[\frac{\delta \rho}{2} |v|^2 - \left(f(\rho) + \rho \frac{\partial f}{\partial \rho}(\rho) - u \right) \delta \rho + \big(\delta \rho_{,t} + \nabla \cdot (\delta \rho v)\big)\phi \right] dx \right\} dt + \\ \int_a^b \left\{ \int \left[\big(\rho v \cdot \delta v + \nabla \cdot (\rho \delta v)\phi \big) \right] dx \right\} dt + \int_a^b \left\{ \int \left[\big(\rho_{,t} + \nabla \cdot (\rho v)\big) \delta \phi \right] dx \right\} dt. \end{gathered} \tag{4.23}$$

Integrating by parts, with the aid of equation 3.3, and rendering all admissible variations stationary gives,

$$\frac{1}{2} |v|^2 - f - \rho \frac{\partial f}{\partial \rho} + u - \phi_{,t} - \nabla \phi \cdot \mathbf{v} = 0, \tag{4.24a}$$

$$\rho \mathbf{v} - \rho \nabla \phi = 0, \tag{4.24b}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0. \tag{4.24c}$$

The third of these equations is the continuity equation 4.18a. In addition, from equation 4.24b it follows that:

$$\mathbf{v} = \nabla \phi \ \text{ in supp}\, \rho. \tag{4.25}$$

Thus, the stationary points of the action, equation 4.22, correspond to potential flow. Taking gradients in equation 4.24a and multiplying by ρ gives,

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) + \rho \nabla \left(\rho \frac{\partial f}{\partial \rho} + f \right) = \rho \mathbf{b}. \tag{4.26}$$

In addition, equation 4.1 provides the identity:

$$\nabla p = \nabla \left(\rho^2 \frac{\partial f}{\partial \rho} \right) = (\nabla \rho) \rho \frac{\partial f}{\partial \rho} + \rho \nabla \left(\rho \frac{\partial f}{\partial \rho} \right) = \rho \nabla \left(\rho \frac{\partial f}{\partial \rho} + f \right). \tag{4.27}$$

Inserting this identity into equation 4.26 gives Euler's equation of motion, i.e., equation 4.18b, as required.

4.2.1 Time Discretization

Now, the aim is to formulate a time discretized, or *semidiscrete*, approximate action for compressible Euler flows that extends the semi-discrete action in equation 3.23 for non-interacting flows to general fluid flows. Specifically, the free energy of the fluid and the body-force distribution are the additional terms considered in this extension. Using a trapezoidal approximation rule for the corresponding action terms provides a semi-discrete action of the form:

$$A_d(\rho_1, \ldots, \rho_{N-1}) = \sum_{k=0}^{N-1} \left\{ \frac{1}{2} \frac{\mathcal{T}_2(\rho_k, \rho_{k+1})}{(t_{k+1} - t_k)^2} - \frac{1}{2} [U(\rho_k) + U(\rho_{k+1})] \right\} (t_{k+1} - t_k), \tag{4.28}$$

which is expressed directly in terms of densities. Here, U is the total internal energy of the fluid given by,

$$U(\rho) = \int \rho(f(\rho) - u)\, d\mathbf{x}. \tag{4.29}$$

Then, the discrete equations of motion follow by rendering the semi-discrete, equation 4.28, action stationary. Taking variations of equation 4.28 with respect to ρ_k with the aid of equation 3.22 gives

$$\begin{aligned}\langle \delta A_d, \xi_k \rangle = \int \Bigg\{ \rho_k \left(\frac{\varphi_{k\to k+1}(\mathbf{x}) - \mathbf{x}}{t_{k+1} - t_k} + \frac{\varphi_{k\to k-1}(\mathbf{x}) - \mathbf{x}}{t_k - t_{k-1}} \right) \cdot \xi_k \\ - \frac{t_{k+1} - t_{k-1}}{2} \rho_k [f(\rho_k) - u(\mathbf{x}, t_k)] \nabla \cdot (\rho_k \xi_k) \Bigg\}\, d\mathbf{x},\end{aligned} \tag{4.30}$$

where $\varphi_{k\to k+1}$ is a transference map transporting ρ_k into ρ_{k+1} optimally over the time interval $[t_k, t_{k+1}]$. Here, we adopt the notation in equation 3.25, such that the identity in equation 3.26 holds. Integrating by parts equation 4.30, using equation 3.3 and the identities equations 4.19 and 4.27 we obtain

$$\begin{aligned}\langle \delta A_d, \xi_k \rangle = \int \Bigg\{ \rho_k \left(\frac{\varphi_{k\to k+1}(\mathbf{x}) - \mathbf{x}}{t_{k+1} - t_k} + \frac{\varphi_{k\to k-1}(\mathbf{x}) - \mathbf{x}}{t_k - t_{k-1}} \right) \cdot \xi_k \\ + \frac{t_{k+1} - t_{k-1}}{2} [\nabla p(\rho_k) - \rho_k b_k] \cdot \xi_k \Bigg\}\, d\mathbf{x}.\end{aligned} \tag{4.31}$$

Here, we write $b_k(\cdot) = b(\cdot, t_k)$. Enforcing stationarity with respect to to all variations, ξ_k, gives

$$\frac{2\rho_k}{t_{k+1} - t_{k-1}} \left(\frac{\varphi_{k\to k+1} - \mathrm{id}}{t_{k+1} - t_k} + \frac{\varphi_{k\to k-1} - \mathrm{id}}{t_k - t_{k-1}} \right) + \nabla p_k = \rho_k b_k, \tag{4.32}$$

where, again, we write $p_k(\cdot) = p(\cdot, t_k)$. In comparing equation 4.32 with equation 3.27, it is noted that the discrete motion now results from the competition between the inertia and internal energy of the fluid. Specifically, the inertia of the fluid, represented by the first term in equation 4.32, aims to transport ρ_k into ρ_{k+1} over the time internal $[t_k, t_{k+1}]$ optimally with respect to the cost function in equation 3.16. Unlike the optimal transportation of non-interacting mass, for general fluid flows the internal energy of the fluid provides a competing objective for the transportation cost. Thus, compared with non-interacting mass transportation, the internal energy of the fluid introduces discrete percussions at discrete times that cause the trajectory to deviate from the inertia-driven, *ballistic* path, for non-interacting mass.

4.2.1.1 OTM Update

Equation 4.32 defines a semidiscrete central-difference scheme that can be solved forward explicitly. Specifically, the forward solution takes the form:

(i) Initialization: Set $k = 0$, $\rho_0 = \rho_a$.

(ii) Given ρ_k, solve (4.32) for the incremental deformation mapping $\varphi_{k\to k+1}$, with the result:

$$\varphi_{k\to k+1} = \mathrm{id} + (t_{k+1} - t_k)\left\{\frac{\mathrm{id} - \varphi_{k\to k-1}}{t_k - t_{k-1}} + \frac{t_{k+1} - t_{k-1}}{2}\frac{\rho_k b_k - \nabla p_k}{\rho_k}\right\} \tag{4.33}$$

(iii) Update the mass density according to identity (3.26).

(iv) Reset $k \leftarrow k + 1$. If $k = N$ exit. Otherwise go to (ii).

This forward solution has the usual structure of explicit time-integration schemes in that the incremental deformation mapping $\varphi_{k\to k+1}$ is computed directly from the initial conditions at the beginning of the time step, including the velocity estimate,

$$v_k \equiv \frac{\mathrm{id} - \varphi_{k\to k-1}}{t_k - t_{k-1}}, \tag{4.34}$$

and the out-of-balance forces $\rho_k b_k - \nabla p_k$. However, as noted earlier, this time-integration algorithm, i.e., equations 3.26 and 4.33 differs from conventional Eulerian algorithms, which rely on some direct time-discretization of the continuity equation. Specifically, the mass density update, given in equation 3.26, is geometrically exact!

4.2.2 Spatial Discretization

In order to obtain a fully discrete action for computations, we proceed to effect a spatial discretization of the semi-discrete action in equation 4.28. In particular, we wish to extend the material-point formalism introduced in section (3.2.2) for non-interacting fluids to the current setting of fluids with equations of state. Recall that, material points were introduced by adopting concentrated mass densities of the form in equation 3.29. In the previous chapter, the operation of inserting these mass densities into the semi-discrete action, equation 3.23, or directly into the semi-discrete Euler-Lagrange equation, equation 3.24, was mathematically well-defined since, for a non-interacting fluid, the mass density entered the variation of the action linearly. However, for a compressible fluid the mass density enters non-linearly in the variation of the semi-discrete action, equation 4.30. Thus, the insertion of equation 3.29 directly into equation 4.30 no longer makes sense mathematically.

In order to overcome this difficulty, we note that equation 3.29 combines two mathematically distinct operations that become blurred in the context of non-interacting fluids but that can be carefully separated for general fluid flows.

The first operation is the approximation of the usual Lebesgue measure of volume $\mathcal{L}$ by the discrete measures,

$$\mathcal{L}_{h,k} = \sum_{p=1}^{M} V_{p,k}\, \delta_{x_{p,k}}, \tag{4.35}$$

that are concentrated at the *material points* locations, $x_{p,k}$, with a corresponding discrete volume, $V_{p,k}$. Thus, for any smooth function f we have

$$\int f d\mathcal{L}_{h,k} = \sum_{p=1}^{M} f(x_{p,k})\, V_{p,k}. \tag{4.36}$$

In addition, the usual push-forward operation for measures, given by

$$\int \eta(\mathbf{y})d\mathbf{y} = \int \eta\big(\varphi_{k\to k+1}(\mathbf{x})\big) \det\big(\nabla\varphi_{k\to k+1}(\mathbf{x})\big)\, d\mathbf{x}, \tag{4.37}$$

now becomes

$$\sum_{p=1}^{M} \eta(\mathbf{x}_{p,k+1})\, V_{p,k+1} = \sum_{p=1}^{M} \eta\big(\varphi_{k\to k+1}(\mathbf{x}_{p,k})\big) \det\big(\nabla\varphi_{k\to k+1}(\mathbf{x}_{p,k})\big)\, V_{p,k}\,. \tag{4.38}$$

However, the material points are convected by the flow, which amounts to the condition:

$$\mathbf{x}_{p,k+1} = \varphi_{k\to k+1}(\mathbf{x}_{p,k}). \tag{4.39}$$

Inserting this identity into equation 4.38 and noting that η is arbitrary defines the material-point volume update as

$$V_{p,k+1} = \det\big(\nabla\varphi_{k\to k+1}(x_{p,k})\big)\, V_{p,k}. \tag{4.40}$$

The second operation subsumed in equation 3.29 is the identification of discrete mass distributions as measures that are absolutely continuous with respect to the discrete volume measure $\mathcal{L}_{h,k}$, with Radon-Nykodim density $\rho_{h,k}$, in the same manner as continuous mass distributions are defined by measures that are absolutely continuous with respect to the Lebesgue measure, $\mathcal{L}$, with density ρ_k. Owing to the discreteness of $\mathcal{L}_{h,k}$, the corresponding mass densities $\rho_{h,k}$ are defined simply by assigning mass density values $\rho_{p,k}$ to every material point:

$$\rho_{h,k}(\mathbf{x}) = \sum_{p=1}^{M} \rho_{p,k} V_{p,k} \delta\big(\mathbf{x} - \mathbf{x}_{p,k}\big). \tag{4.41}$$

Comparing equation 4.41) and equation 3.29 provides the identity:

$$m_p = \rho_{p,k} V_{p,k}, \tag{4.42}$$

or,

$$\rho_{p,k} = \frac{m_p}{V_{p,k}}. \tag{4.43}$$

Equation 4.47 provides the relationship between the mass density of the material point located at $\mathbf{x}_{p,k}$ with respect to its mass and volume.

The identities in equations 4.39, 4.40, and 4.47) provide the update algorithm for the position, volume, and mass density of the material points. However, it should be noted that the gradient of the transference map $\varphi_{k\to k+1}$ appears explicitly in equation 4.40. Therefore, in contrast to the case of non-interacting fluids, where it suffices to track the motion of the material points, now the transference maps $\varphi_{k\to k+1}$ must be approximated by conforming interpolations of the form:

$$\varphi_{h,k\to k+1}(\mathbf{x}) = \sum_{a=1}^{N} \mathbf{x}_{a,k+1} N_{a,k}(\mathbf{x}). \tag{4.44}$$

Here, $\{\mathbf{x}_{a,k+1},\ a = 1, \ldots, N\}$ are coordinates of nodes on the configuration at time t_{k+1} and $N_{a,k}(\mathbf{x})$ are conforming shape functions defined over the configuration at time t_k with the properties consistent with exact interpolation of linear functions, namely,

$$\sum_{a=1}^{N} N_{a,k}(\mathbf{x}) = 1, \tag{4.45a}$$

$$\sum_{a=1}^{N} \mathbf{x}_{a,k} N_{a,k}(\mathbf{x}) = \mathbf{x}. \tag{4.45b}$$

For discrete transference maps of the form equation 4.44, the material-point update equation 4.39 is given by

$$\mathbf{x}_{p,k+1} = \sum_{a=1}^{N} \mathbf{x}_{a,k+1} N_{a,k}(\mathbf{x}_{p,k}). \tag{4.46}$$

Here, the linear consistency conditions in equation 4.45 leaves considerable latitude in the choice of shape functions. In the OTM method LME shape functions [4], computed from convected nodal coordinates are employed for the salient features presented in chapter 2. Also, the shape functions are denoted as *backward compatible* if the following condition holds:

$$\mathbf{x}_{p,k-1} = \sum_{a=1}^{N} \mathbf{x}_{a,k-1} N_{a,k}(\mathbf{x}_{p,k}). \tag{4.47}$$

Furthermore, a convenient strategy for implementing the updates in equation 4.40 is to initialize the material point volumes and densities by means of an initial triangulation of the domain.

Inserting the preceding approximations into equation 4.28 gives the discrete action:

$$\begin{aligned} A_h(\rho_{h,1}, \ldots, \rho_{h,N-1}) = \sum_{k=0}^{N-1} \sum_{p=1}^{M} m_p \bigg\{ \frac{1}{2} \frac{|\mathbf{x}_{p,k+1} - \mathbf{x}_{p,k}|^2}{(t_{k+1} - t_k)^2} \\ - \frac{1}{2} [\big(f(\rho_{p,k}) - u(\mathbf{x}_{p,k})\big) + \big(f(\rho_{p,k+1}) - u(\mathbf{x}_{p,k+1})\big)] \bigg\} (t_{k+1} - t_k), \end{aligned} \tag{4.48}$$

where equation 4.47 is tacitly understood to be in force. Taking variations of A_h gives the discrete Euler-Lagrange equations:

$$\begin{aligned} \langle \delta A_h, \xi_h \rangle = \sum_{k=0}^{N-1} \sum_{p=1}^{M} m_p \bigg\{ \left(\frac{\mathbf{x}_{p,k+1} - \mathbf{x}_{p,k}}{t_{k+1} - t_k} + \frac{\mathbf{x}_{p,k} - \mathbf{x}_{p,k-1}}{t_k - t_{k-1}} \right) \cdot \xi_{h,k}(\mathbf{x}_{p,k}) \\ - \frac{t_{k+1} - t_{k-1}}{2} \left[\frac{p(\rho_{p,k})}{\rho_{p,k}} \nabla \cdot \xi_{h,k}(\mathbf{x}_{p,k}) + b_k(\mathbf{x}_{p,k}) \cdot \xi_{h,k}(\mathbf{x}_{p,k}) \right] \bigg\}. \end{aligned} \tag{4.49}$$

Here, $\xi_{h,k}(\mathbf{x})$ are discrete admissible virtual displacements given by,

$$\xi_{h,k}(\mathbf{x}) = \sum_{a=1}^{N} \xi_{a,k} N_{a,k}(\mathbf{x}). \tag{4.50}$$

Also, note that equation 4.49 can be obtained, alternatively, by inserting the spatial discretization into the semi-discrete Euler-Lagrange equations, i.e., equation 4.30, directly.

Enforcing stationarity with respect to to all variations, ξ_h, gives:

$$\frac{2}{t_{k+1}-t_{k-1}}\left(M_k\frac{\mathbf{x}_{k+1}-\mathbf{x}_k}{t_{k+1}-t_k}-\mathbf{p}_k\right)=\mathbf{f}_k, \tag{4.51}$$

where, $x_k \equiv \{x_{1,k},\ldots,x_{N,k}\}$ is the nodal coordinate array at time t_k, and the linear momentum of node a at time t_k is given by,

$$p_{k,a}=\sum_{p=1}^{M} m_p\frac{\mathbf{x}_{p,k}-\mathbf{x}_{p,k-1}}{t_k-t_{k-1}}N_{a,k}(\mathbf{x}_{p,k}). \tag{4.52}$$

Furthermore, the consistent mass matrix box for nodes a and b at time t_k, is given by,

$$M_{k,ab}=\sum_{p=1}^{M} m_pN_{a,k}(\mathbf{x}_{p,k})N_{b,k}(\mathbf{x}_{p,k})\,\mathbf{I}. \tag{4.53}$$

The out-of-balance nodal forces are given by,

$$f_{k,a}=\sum_{p=1}^{M}\left[V_{p,k}p(\rho_{p,k})\nabla N_{a,k}(\mathbf{x}_{p,k})+m_p\mathbf{b}_k(\mathbf{x}_{p,k})N_{a,k}(\mathbf{x}_{p,k})\right]. \tag{4.54}$$

In the special case of backward-compatible shape functions (i.e. equation 4.47), equation 4.51 provides the central difference scheme:

$$\frac{2}{t_{k+1}-t_{k-1}}\mathbf{M}_k\left(\frac{\mathbf{x}_{k+1}-\mathbf{x}_k}{t_{k+1}-t_k}-\frac{\mathbf{x}_k-\mathbf{x}_{k-1}}{t_k-t_{k-1}}\right)=\mathbf{f}_k. \tag{4.55}$$

4.2.2.1 OTM Update

The above equations define a finite-dimensional semi-discrete central-difference scheme that can be solved forward explicitly. Then, the forward solution is given by the following scheme:

(i) Initialization: Set $k=0$, initialize material point volumes, densities, shape functions.

(ii) Given $\mathbf{x}_{k-1}$, $\mathbf{x}_k$, ρ_k, solve (4.51) for the updated nodal coordinates, with the result:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + (t_{k+1} - t_k)\mathbf{M}_k^{-1}(\mathbf{p}_k + \frac{t_{k+1} - t_{k-1}}{2}\mathbf{f}_k) \tag{4.56}$$

(iii) Update the material point coordinates, volumes, mass densities according to the identities:

$$x_{p,k+1} = \varphi_{h,k\to k+1}(\mathbf{x}_{p,k}) \tag{4.57a}$$

$$V_{p,k+1} = V_{p,k+1} \det \nabla\varphi_{h,k\to k+1}(\mathbf{x}_{p,k}) \tag{4.57b}$$

$$\rho_{p,k+1} = \frac{m_p}{V_{p,k+1}} \tag{4.57c}$$

(iv) Calculate shape functions $N_{a,k+1}(\mathbf{x}_{p,k+1})$ and derivatives $\nabla N_{a,k+1}(\mathbf{x}_{p,k+1})$

(v) Reset $k \leftarrow k+1$. If $k = N$ exit. Otherwise go to (ii).

4.2.3 The OTM Method for Navier-Stokes Fluid Flows

This section extends the OTM to fluids with dissipation (i.e. viscosity). Following the developments in [90], we add viscosity to the formulation by reformulating the semi-discrete equations of motion as a minimum principle:

$$\Phi(\varphi_{k\to k+1}) \to \inf! \tag{4.58}$$

For conservative systems, (i.e. Euler fluid flow), [90] gives the incremental potential energy by,

$$\Phi(\varphi_{k\to k+1}) = \int \left\{ \frac{\rho_k}{2} \frac{|\varphi_{k\to k+1} - \varphi^{\text{pre}}_{k\to k+1}|^2}{(t_{k+1} - t_k)^2} + w(\nabla\varphi_{k\to k+1}) \right\} dx. \tag{4.59}$$

Here, φ^{pre} is a predictor incremental deformation mapping and we write

$$w(\nabla\varphi_{k\to k+1}) = \frac{1}{2}\left(\rho_k\left(f(\rho_k) - u_k\right) + \rho_{k+1}\left(f(\rho_{k+1}) - u_{k+1}\right)\right).$$

Minimization of this incremental potential energy, $\Phi(\varphi_{k\to k+1})$, gives the Euler-Lagrange equations:

$$\rho_k \frac{\varphi_{k\to k+1} - \varphi^{\text{pre}}_{k\to k+1}}{(t_{k+1} - t_k)^2} + \nabla p_k = \rho_k b_k. \tag{4.60}$$

Here, the predictor φ^{pre} can be chosen such that equation 4.60 matches equation 4.32 for the semi-discrete Euler-Lagrange equation, with the result:

$$\varphi^{pre} = \varphi_{k\to k+1} - 2\frac{(t_{k+1} - t_k)^2}{(t_{k+1} - t_{k-1})}\left(\frac{\varphi_{k\to k+1} - \text{id}}{t_{k+1} - t_k} + \frac{\varphi_{k\to k-1} - \text{id}}{t_k - t_{k-1}}\right).$$

In order to add viscous dissipation to the OTM formulation for Euler fluid flows, let $\phi(\varphi_{k\to k+1})$ be a material frame indifferent function such that

$$\langle D^2\phi(\text{id})d, d\rangle = \frac{\lambda}{2}\text{tr}(d)^2 + \mu\|d\|^2,$$

where λ and μ are viscosities. In the following calculations, ϕ is chosen to be of the Hadamard form (i.e. a neo-Hookean potential extended to the compressible range). Then, the incremental strain energy density that serves as a joint potential for the pressure and the viscous forces, extended from [90, 91], is given by

$$w(\nabla\varphi_{k\to k+1}) = \rho_k\left(f(\rho_k) - u_k\right) + \rho_{k+1}\left(f(\rho_{k+1}) - u_{k+1}\right) + \frac{\phi\left(\nabla\varphi_{k\to k+1}\right)}{t_{k+1} - t_k}.$$

In terms of this extended energy-dissipation density, the incremental viscous problem retains the same form as equations 4.58 and 4.59 presented for the conservative problem. Furthermore, the spatial discretization retains the same form as the conservative case.

Chapter 5

Numerical Examples

This chapter demonstrates the use of the OTM method for numerical simulations of Euler and Navier-Stokes fluid flows. In an effort to numerically validate the OTM method, the classical Riemann numerical benchmark test is simulated. Specifically, the Riemann numerical benchmark test verifies the ability of the numerical code to capture shock waves. Following this benchmark test, an illustrative FSI example is presented to highlight the ability of the OTM method. Here, the inspiration for this particular example is provided by the continued success of inflatable structures for space-based structures. Specifically, this FSI example is motivated by NASA's use of inflated networks of airbags, surrounding the lander, to dissipate the energy of impact for previous Mars missions.

5.1 Numerical Benchmark Test: Riemann Problem

The Riemann benchmark problem provides one of the most popular and classical numerical benchmark tests for evaluating the capability and accuracy of a given gas-dynamics code or numerical method. The Riemann problem tests the accuracy of a method related to solving the one-dimensional time dependent Euler equations combined with an ideal equation of state and discontinuous initial conditions. This example is especially convenient for assessing the accuracy of the numerical solution since the Riemann problem has an exact mathematical solution. This numerical benchmark problem received significant

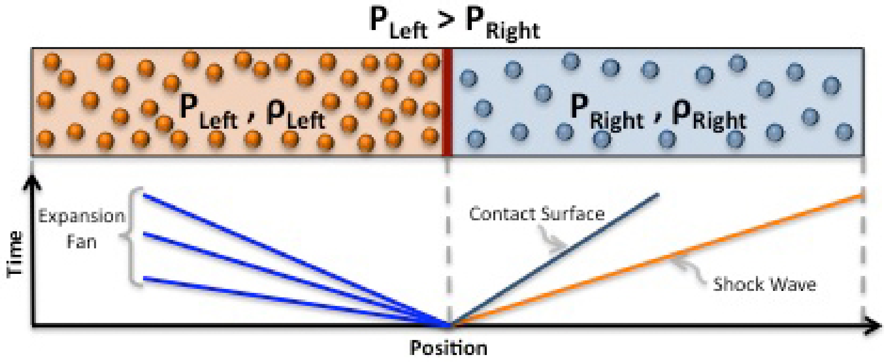

Figure 5.1: A general schematic for the problem and the resulting waves generated upon rupturing the diaphragm.

attention by Sod in 1978 [92]. Thus, in the context of the parameters used in this example the Riemann problem is synonymous with the Sod test problem.

Physically, the Riemann problem corresponds to a shock tube experiment consisting of a long tube containing two gases of different properties separated by a thin diaphragm. At the onset of the experiment, the diaphragm is ruptured and the evolution of the resulting waves can be tracked. If we use two gases of the same species but with different pressures and densities, both experiments as well as the exact mathematical solution predict three waves: a shock wave, contact surface, and a fan of expansion waves emanating from the rupture point. Figure 5.1 depicts the general experimental setup and the resulting waves in space-time. A complete analysis and presentation of the Riemann problem and traditional numerical solvers can in [93].

5.1.1 Model Formulation

Solving the one-dimensional time dependent Euler equations with an ideal gas law amounts to solving the system of conservation equations (mass, momentum, and energy) combined with a specific material model.

$$\rho_{,t} + \nabla\cdot(\rho v) = 0,$$

$$\rho(v,_t + v \cdot \nabla v) + \nabla p = \rho b,$$

$$E,_t + \nabla \cdot (v(E + p)) = 0,$$

$$E = \rho(e + \frac{1}{2}u^2),$$

where ρ is the density, v is the velocity, p is the thermodynamic pressure, E is the total energy per unit volume, and e is the internal energy per unity mass. If we adopt the equation of state for an ideal gas:

$$P = \rho \bar{R} T,$$

where $\bar{R} = \frac{R}{M}$, R is the universal gas constant and M is the molar mass of the gas, then the internal energy can be written as:

$$e = \frac{p}{(\gamma - 1)\rho}.$$

Furthermore, considering reversible adiabatic (isentropic) processes that, coupled with the ideal gas equation of state, lead to the following relationship for a polytropic process:

$$\frac{p_1}{\rho_1^\gamma} = \frac{p_2}{\rho_2^\gamma} = C_{polytropic},$$

$$\gamma = \frac{C_p}{C_v},$$

where $C_{polytropic}$ is constant for the process of going between states 1 and 2, and γ is the ratio of specific heats. Furthermore, recall from equation 4.14 that the energy is given by

$$e(\rho) = \frac{C_{polytropic}}{(\gamma - 1)} \rho^{(\gamma - 1)}.$$

5.1.2 Analytic Solution

The following system of equations in one dimension can be solved by the method of characteristics. This procedure reveals that there are three characteristics, corresponding to the eigenvalues

$$\begin{aligned}\lambda_1 &= u,\\ \lambda_2 &= u + a,\\ \lambda_3 &= u - a,\end{aligned}$$

where $a = \sqrt{\gamma \frac{p}{\rho}}$ is the wave speed. Consider the schematic in figure 5.1, and the four distinct regions:

- (L) : To the left of the rarefaction waves
- $(L_\star)$: In between the rarefaction and contact surface
- $(R_\star)$: In between the contact surface and the shock wave
- (R): To the right of the shock wave

Then, an iterative procedure follows by solving for the unknowns, $(p_\star, u_\star, \rho_{L_\star}, \rho_{R_\star})$ with the provided data $(p_L, p_R, \rho_L, \rho_R)$. Here, $p_\star$ is found as the root of the equation:

$$f = f_L + f_R + (u_r - u_l),$$

where, consistent with figure 5.1,

$$\begin{aligned}f_L &= \frac{2a}{\gamma - 1}\left[\left(\frac{p_\star}{p_L}\right)^{\frac{\gamma-1}{2\gamma}} - 1\right],\\ f_R &= (p_\star - p_R)\left[\frac{2}{\rho_R\left((\gamma+1)p_\star + (\gamma-1)p_R\right)}\right]^{\frac{1}{2}}.\end{aligned}$$

Then, the velocity between the expansion waves and the shock wave is evaluated as:

$$u_\star = \frac{1}{2}(u_L + u_R) + \frac{1}{2}\left(f_R(p_\star) - f_L(p_\star)\right).$$

Subsequently, the following relations hold across the expansion fan in the left region:

$$\begin{aligned}
\rho_{L_\star}(x,t) &= \rho_L \left[\frac{2}{\gamma+1} + \frac{\gamma-1}{(\gamma+1)a_L}\left(u_L - \frac{x}{t}\right)\right]^{\frac{2}{\gamma-1}} \\
u_\star(x,t) &= \frac{2}{\gamma+1}\left[a_L + \frac{\gamma-1}{2}u_L + \frac{x}{t}\right] \\
p_\star(x,t) &= p_L \left[\frac{2}{\gamma+1} + \frac{\gamma-1}{(\gamma+1)\,a_L}\left(u_L - \frac{x}{t}\right)\right]^{\frac{2\gamma}{\gamma-1}}.
\end{aligned}$$

Similarly, given the pressure and velocity in the star region, $(p_\star, u_\star)$, the relationship between the density ratio across the shock wave and the shock speed, S_R, (in the stationary frame) is given by

$$\begin{aligned}
\rho_{R_\star} &= \rho_R \left[\frac{\frac{p_\star}{p_R} + \frac{\gamma-1}{\gamma+1}}{\frac{\gamma-1}{\gamma+1}\frac{p_\star}{p_R} + 1}\right], \\
S_R &= u_R + a_R \left[\frac{\gamma+1}{2\gamma}\frac{p_\star}{p_R} + \frac{\gamma-1}{2\gamma}\right]^{\frac{1}{2}}.
\end{aligned}$$

Finally, the wave speeds are used to update the position of the three waves. Then, we can recursively apply this algorithm to solve for the evolution of the thermodynamic quantities in space and time.

5.1.3 Numerical Implementation

This section overviews the implementation of the OTM method for the Riemann numerical benchmark test. For this application, the OTM method is initialized by specifying the geometry, spatial discretization, material point elements, gas properties, boundary conditions, initial conditions, and the time step. While the OTM is a meshfree method, we refer to *material point elements*, or simply elements, as the discretized regions associated with a material point. Compared with traditional finite elements, the material points can be viewed as a one-point quadrature scheme. However, in sharp contrast with the FEM, the number of nodes associated with each quadrature point varies at each time-step based on the evolving neighborhood around the quadrature point.

The geometry for this one-dimensional problem is simply a line of fixed length that is discretized into N elements, with N material points placed at the midpoint of the element. Furthermore, the initial quadrature weight is taken as the initial length of the element. Initially, $d+1$ nodes (d dimension) are associated with each material point. This is chosen out of convenience for calculating the LME shape functions, since the value of each shape function is inherently prescribed (i.e. $\frac{1}{d+1}$). In turn, this provides the exact value of $\lambda^*(x_{qp})$ that is used in the LME shape function calculation. Furthermore, for the evaluation of the LME shape functions, the parameter β is chosen with respect to the average nodal spacing by specifying the normalized parameter $\mu = \beta h^2$. Here, as displayed in figure 2.3, μ provides a useful indication of the locality of a given LME shape function.

Also, as part of the initialization of the elements, the initial gas parameters (i.e. p_L, ρ_L, p_R, ρ_R, γ) are specified for each side of the shock tube. Furthermore, it is assumed that the gases on both sides of the membrane are initially stationary (i.e. $v = 0$). Specifying the pressure, or density, ratio across the diaphragm provides the wave speeds of the resulting shocks and indeed the complete, analytical solution to compare the numerical results with.

Furthermore, the stable time step for the explicit Nemwark time integration scheme is determined based on the Courant-Friedrichs-Lewy condition (CFL),

$$\frac{u\triangle t}{\triangle x} < C, \tag{5.1}$$

where u is the velocity, $\triangle x$ is the spacing, and C is a constant chosen based on the type of PDE considered. For this Riemann problem, the constant is chosen as $C = \frac{1}{20}$. Also, while we are interested in the solution before the waves experience reflections, fixed boundary conditions are applied to the endpoints of the geometry to completely specify the problem.

After initialization, the initial mesh is abandoned. Then, the scheme provided in section (4.2.2.1) provides the routine for updating the nodal locations, the material point quadrature weight, and the material point location. Furthermore, at each step in the dynamic simulation the LME shape functions are re-evaluated based on the new distribution of nodes. Due to numerical limits in precision, in order to avoid errors arising from calculating the exceedingly small contributions found at the tails of the Gaussian-like LME shape functions, it is practical to employ a search algorithm to find the nearest neighbors that effectively contribute to the evaluation of a given LME shape function. Specifically, for this one-dimension problem, it is sufficient (in terms of speed) to employ an exhaustive search for finding the neighboring nodes contributing to each LME shape function. Also, since evaluating $\lambda^*(x_{qp})$ for the LME shape functions involves an extremely challenging (from a numerical standpoint) optimization problem, it is efficient and often necessary (to avoid termination) to store and use the previous converged value of $\lambda^*(x_{qp})$ in the optimization routine. Here, recalling that the material point mass remains constant, the LME shape functions are also used to update the mass for all of the nodes.

Artificial Viscosity

Numerical simulations involving the propagation of discontinuities (i.e. shocks) for inviscid fluids require specific care for capturing the behavior on each side of the discontinuity. Here, the challenge of resolving an infinitesimal discontinuity for inviscid fluids leads to oscillations analogous to the Gibbs phenomenon in Fourier series approximations. In an effort to stabilize the simulation, an artificial viscosity scheme (adapted from [94]) is added to the model. Here, the principle goal of adding artificial viscosity is to stabilize the simulations involving steep shocks, whose thickness is too small to be resolved by any discretization. The addition of artificial viscosity was first proposed in 1950 by Neumann and Richtmyer

[95]. Here, the intention was to spread the shock front over several grid points without disrupting the structure of the shock (speed, shape, etc.) or introducing spurious oscillations. In the proposed model of artificial viscosity, the magnitude of the introduced viscosity depends on grid spacing. This feature ensures that the contributions vanish with decreasing grid size. Furthermore, the model proposed in [94] possesses the salient attributes of being formulated within a general finite-deformation frame and satisfying material-frame indifference exactly.

From [94], the effective viscosity coefficient, η_{eff}, includes a physical and artificial component:

$$\eta_{eff} = \eta + \Delta\eta.$$

Here, η is the physical viscosity and $\Delta\eta$ is the artificial viscosity coefficient. Following [94], the artificial viscosity coefficient at a given material point assumes the form,

$$\Delta\eta = \begin{cases} max\ \left(0,\ \frac{3}{4} l\rho(c_1 \Delta u - c_L a)\right) & \Delta u < 0, \\ 0 & \Delta u > 0. \end{cases}$$

Here, l is a measure of the size of the element, Δu is a measure of the velocity jump across the element, a is the characteristic wave speed, and (c_1, c_L) are user-defined coefficients. A convenient way of ensuring that this formulation is material frame indifferent is to specify Δu as a function of, or the time derivatives of, the Jacobian of the deformation combined with selecting l in the following way:

$$\begin{aligned} \Delta u &= l \frac{\partial \log J}{\partial t}, \\ l &= (J d! |K|)^{\frac{1}{d}}, \end{aligned}$$

where d is the dimension and K is the element volume.

5.1.4 Results

In accordance with the traditional Sod test problem, the following parameters were used for the initial conditions in the simulation:

$$\begin{aligned} p_L &= 1.0 \\ \rho_L &= 1.0 \\ p_R &= 0.1 \\ \rho_R &= 0.25 \end{aligned}$$

The results from this numerical simulation (using 400 nodes) are compared with the analytical solution and the relevant properties of interest (P and ρ) are plotted with the exact solution in figure 5.2 and figure 5.3, respectively.

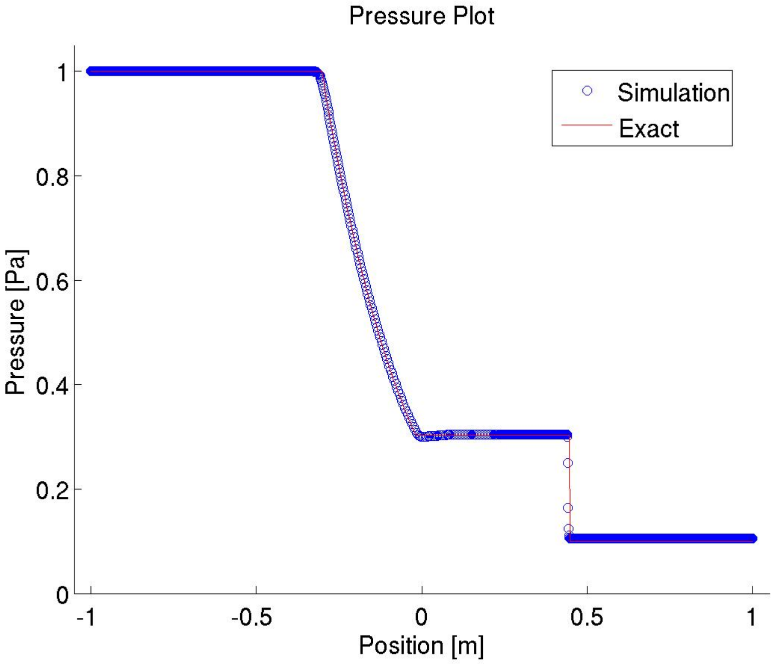

Figure 5.2: Pressure profile compared with the exact solution at time, t = 0.25 s.

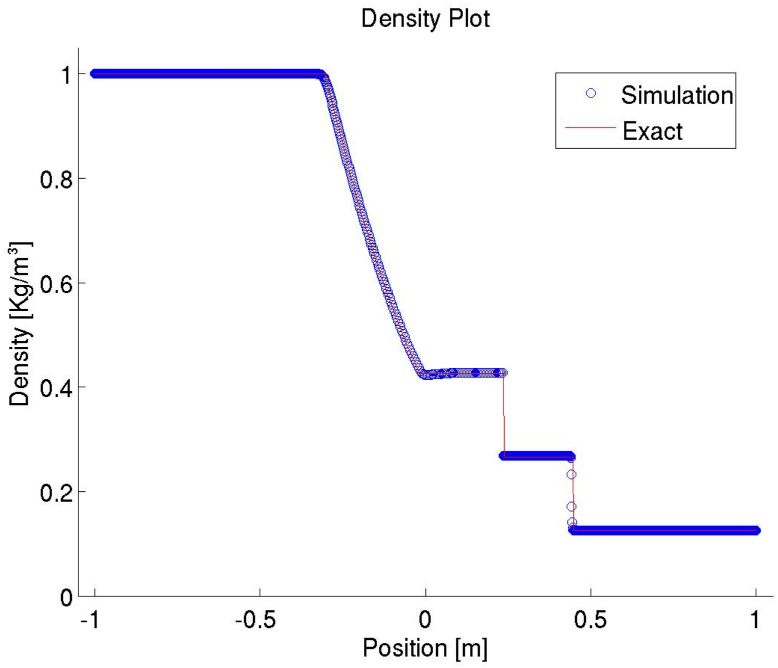

Figure 5.3: Density profile compared with the exact solution at time, t = 0.25 s.

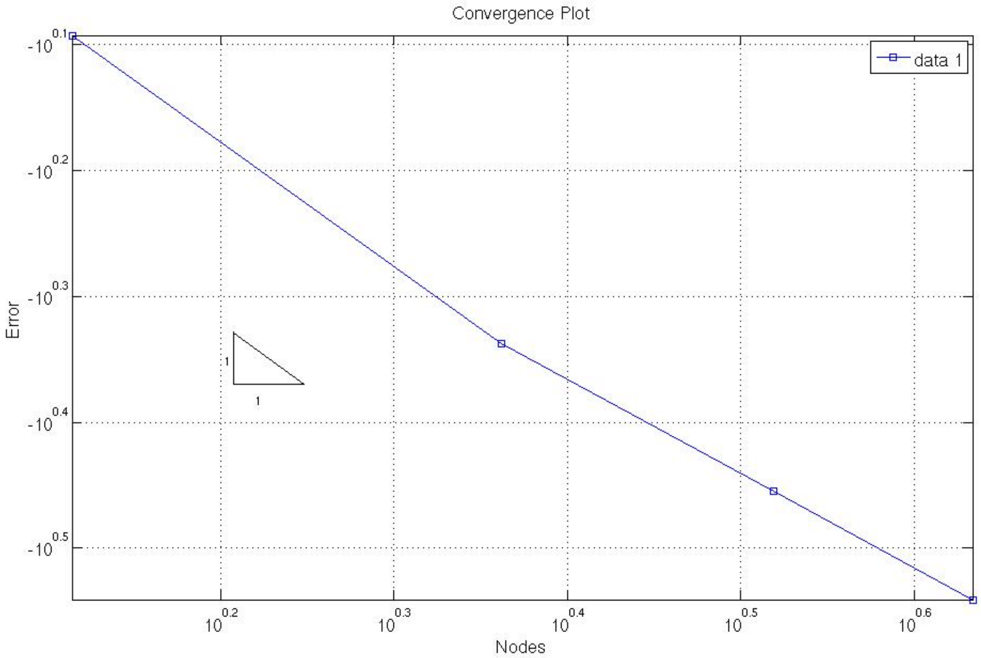

Figure 5.4: Convergence rate for Riemann benchmark problem.

The results for the pressure and density profiles (figures 5.2 and 5.3) are in excellent agreement with the exact solution indicating that the OTM method is exceedingly capable of capturing expansion and compression shock waves. Furthermore, the convergence plot, depicted in figure 5.4, indicates that the OTM method provides linear convergence for the Riemann benchmark problem as expected.

5.2 FSI Numerical Simulation

Deployable structures represent a broad category of structures with applications spanning many engineering fields. Space science is one area where instrumenting deployable structures is of particular interest. Here, the size and weight constraints, imposed by limitations in launch technology, have traditionally stifled the size of telescopes, support structures, and satellites. In an effort to push the limits of the sizes and capabilities of telescopes and space-bound instruments, inflatable gossamer structures (i.e. thin-shell structures) present a particularly attractive concept for space-based applications. Gossamer structures are lightweight by design, have a small packing volume relative to their deployed volume, and can assume complex shapes [96]. Once deployed, gossamer structures can provide a stiff framework for next-generation telescopes, serve as support structures for instruments, or provide habitats for future space exploration missions [97]. Due to the extravagant costs associated with current launch technology, estimated between \$5,000 and \$20,000 per kg [98], numerical simulations of space-based deployable structures are paramount to developing viable designs.

Another remarkable use of inflatable space-based structures is demonstrated during the final stage of NASA's entry, descent, and landing (EDL) scheme, depicted in figure 5.2, that was developed for past Mars missions. Prior to impacting the surface, a network of airbags attached to the lander was inflated in order to dissipate the final amount of energy imparted during the impact of landing.

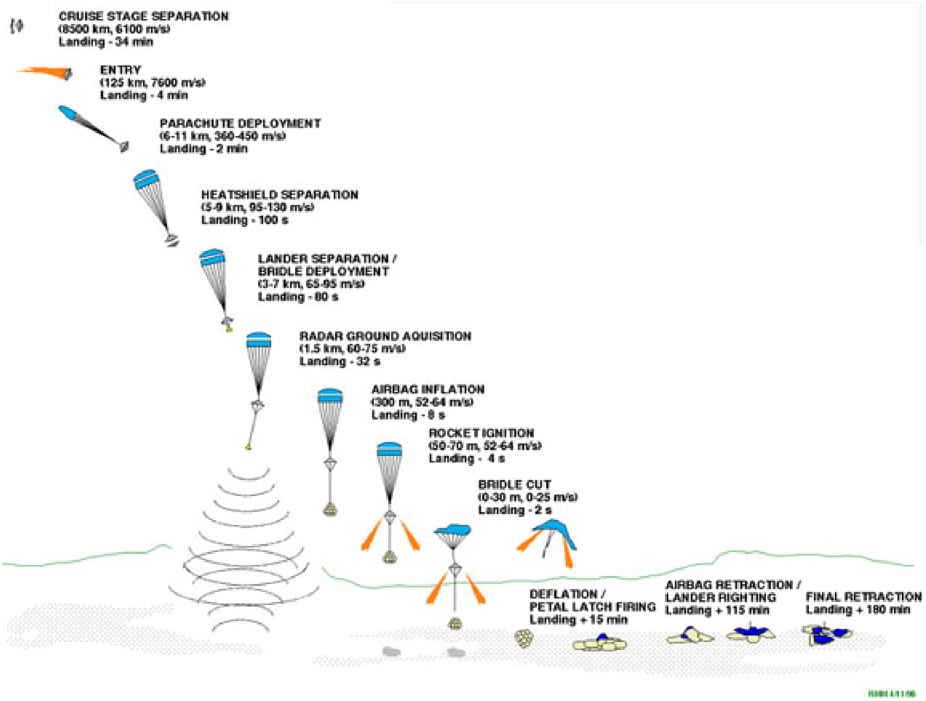

Figure 5.5: Entry, descent and landing (EDL) for Pathfinder landing on Mars [3].

While there is significant interest in modeling this FSI-based landing procedure, numerical studies must rely on commercial software, all of which face the limitations outlined with regard to traditional, mesh-based numerical methods. Specifically, the models embedded in these codes consider the fluid inside the inflatable structure as a single control volume. However, in doing so, the fluid field variables remains uniform. Thus, the ability to capture gradients of field values, such as varying pressures or densities, aside from capturing shocks, is simply lacking. Furthermore, while using other coupled methods (i.e. IB or ALE methods) may appear attractive, the exceedingly large deformations imparted to the structure makes this framework undesirable.

In an effort to highlight the salient attributes of the OTM method with respect to the current state-of-the-art numerical capabilities, an illustrative FSI example is simulated. Here, we consider a three-dimensional strongly coupled FSI application consisting of a gas-filled sphere impacts the ground at high speed. This application is inspired by the success of NASA's recent Mars missions using airbag technology to provide the last stage of entrydecent and landing shown below in figure 5.2.

5.2.1 Illustrative Application: Impact of a Gas-Inflated Sphere

Given the motivation of simulating a wide-range of FSI problems within the space-science field, we consider an illustrative example to demonstrate the capability and flexibility of the OTM method. Here, we consider solving the Navier-Stokes equations of motion for a compressible, viscous fluid in an arbitrary three-dimensional domain. Specifically, the sphere is characterized using membrane finite elements, the details of which can be found in [5], and material model for the membrane is chosen as a classical Neo-hookean material.

5.2.1.1 Model Formulation

Here we specify the material of the membrane to share similar properties with Kapton and Vectran (used in NASA's previous airbags). Furthermore, we model the gas as a compressible Newtonian fluid following an ideal equation of state and model the deformation process of the gas-filled ball as an adiabatic process. In the simulations presented below, the following parameters for modeling the compressible Newtonian gas are employed:

$$\begin{aligned} \rho &= 1.25 \text{ kg/m}^3 \\ P &= 1.5 \text{ psi} \\ \gamma &= 1.4 \\ \mu &= 1.8\text{e}^{-5} \text{ Pa s} \\ R &= 1.0 \text{ m.} \end{aligned}$$

Also, the material parameters for the membrane are given as

$$\rho = 0.0915 \text{ kg/m}^2,$$

$$t = 1 \text{ mm}.$$

Here, the stiffness of the membrane is chosen such that the initial configuration is in equilibrium for a given size ball, density, and pressure of the gas using the relation

$$E = \frac{PR}{2t}.$$

Furthermore, the timestep is chosen using the CFL condition, equation 5.1, based on the initial prescribed velocity, wave speeds in the membrane as well as the local wave speed of the gas,

$$\triangle t_{stable} = C \ \min \left\{ \sqrt{\frac{E_{mem}}{\rho_{mem}}}, \sqrt{\gamma \frac{P}{\rho}}, \sqrt{\frac{h}{V_{presc}}} \right\}. \tag{5.2}$$

In the following simulation the impact of the structure with the ground is modeled by considering the surface of the ground as a spring. Here, the stiffness is chosen to be as stiff as possible without creating artificial artifacts or diminishing the conservation of total energy. For this application, the spring stiffness was chosen as,

$$K_{FloorPotential} = 5.0 \text{ e}^4\text{N/m}^2.$$

Also, the prescribed initial velocity for this simulation was

$$V_{prec} = -100 \text{ m/s}.$$

5.2.1.2 Numerical Implementation

Then, the procedure for initializing the simulation follows as:

1. Construct a Delaunay triangulation for the sphere. Here, a mesh is created only as a means of initializing the locations of the material points (at the barycenters of the elements) and the initial quadrature weight (computed volume). Following this initialization step, the mesh is no longer required or used in the simulation procedure.

2. Set up the LME shape functions and derivatives. Here, initially we begin with $d + 1$ nodes as the support neighborhood for the material point. By adopting this initial configuration one can exactly and explicitly determine λ^*. Since we will need to establish neighbor lists during the simulation, as part of the initialization procedure we adopt a search algorithm. In this case we employ an orthogonal range query (ORQ), developed by Sean Mauch [99], as our search routine and update the data structure at each time step during the simulation.

3. Initialize the elements with the appropriate materials. i.e., Neo-hookean for the membrane elements and compressible Navier-Stokes for the material point elements.

4. Establish the time step for the simulation based on the CFL criteria, i.e., equation 5.2.

5. At this stage we employ an explicit dynamics routine to evolve the equations of motion. At each step during the dynamic simulation, we reset the material point quadrature weights and location (using the deformation mapping), evaluate the neighbors list for each material point, construct the LME shape functions, and update the mass prior to incrementing the explicit dynamics update.

5.2.2 Results

The procedure outlined above provides update implemented for this application. Here, figure 5.6 depicts the initial discretized configuration for the membrane (triangulation) and the fluid (material points) used in the simulation.

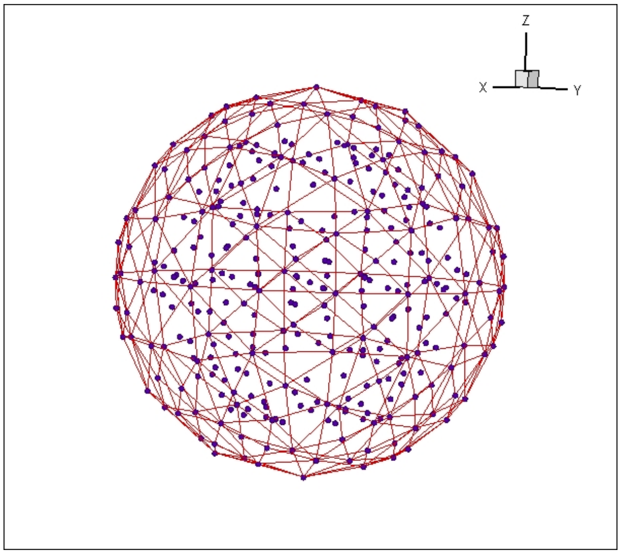

Figure 5.6: Initial configuration of the gas-filled ball.

The following results depict the largest extent of deformation of the sphere during impact. Figure 5.8 depicts the contour levels of the Von Mises stress criteria on the membrane surface. Figure 5.7 depict the contour levels of the Von Mises stress criteria for the fluid (i.e. section view).

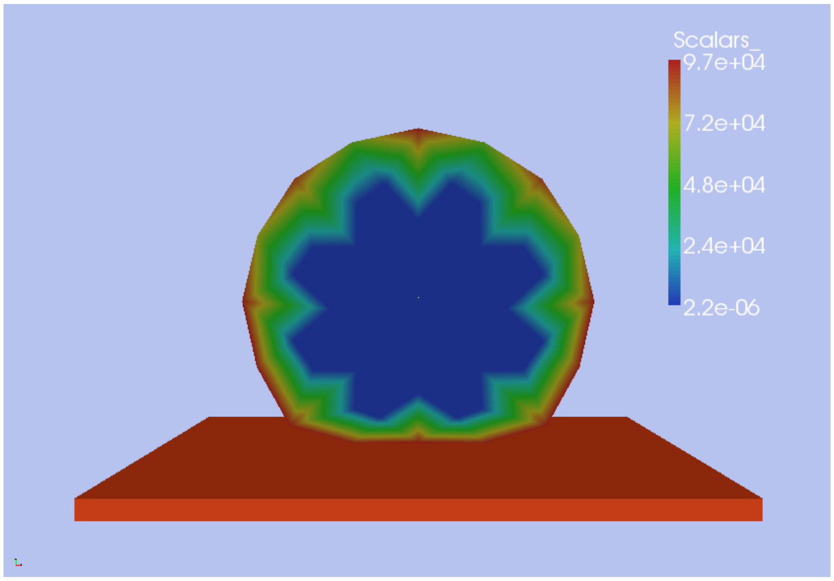

Figure 5.7: Von Mises stress criteria of gas at impact (midplane section).

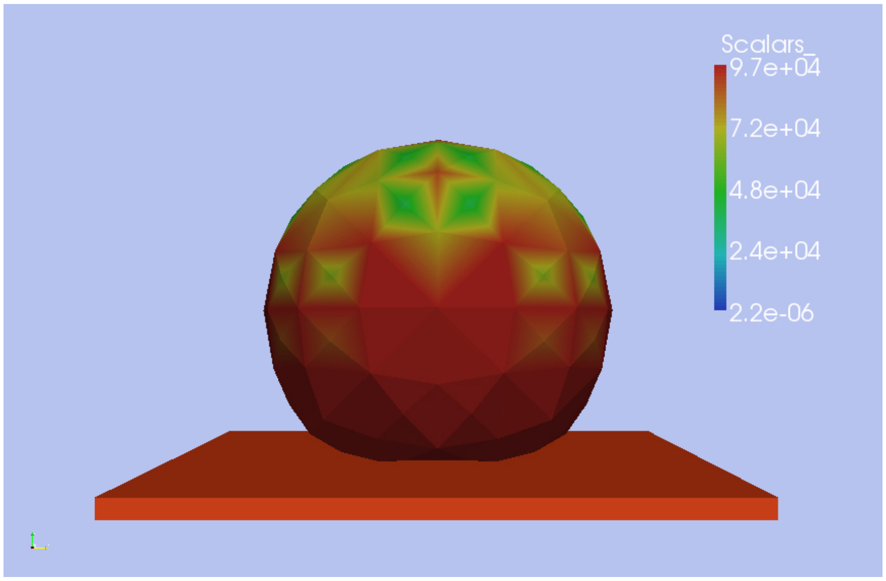

Figure 5.8: Von Mises stress criteria on surface of membrane at impact.

Figure 5.10 depict the contour levels of pressure on the membrane surface. Figure 5.9 depict the contour levels of pressure of the fluid (i.e. section view).

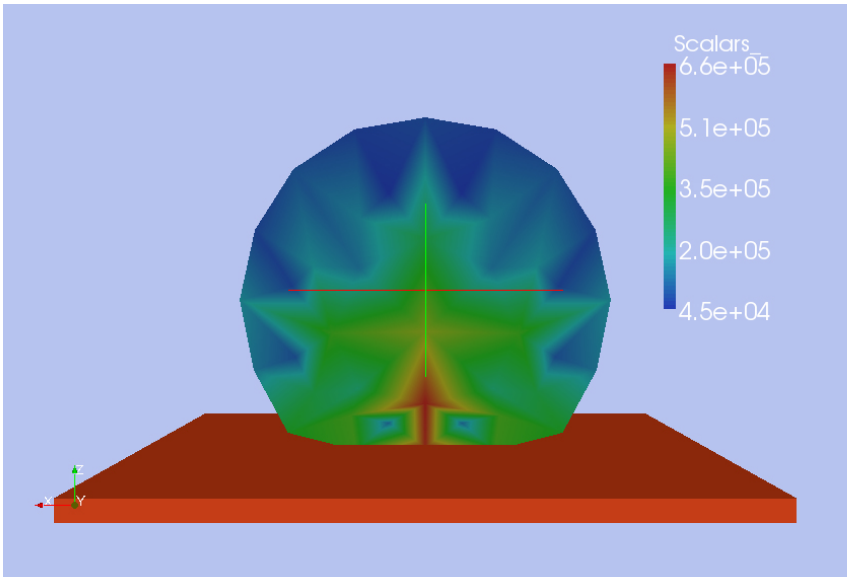

Figure 5.9: Pressure of the gas at impact (midplane section).

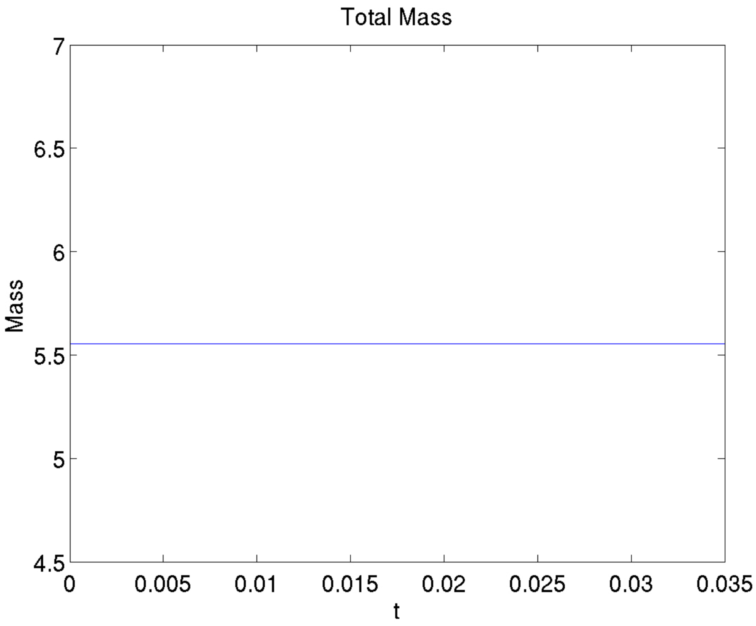

Figure 5.11: Total mass vs time.

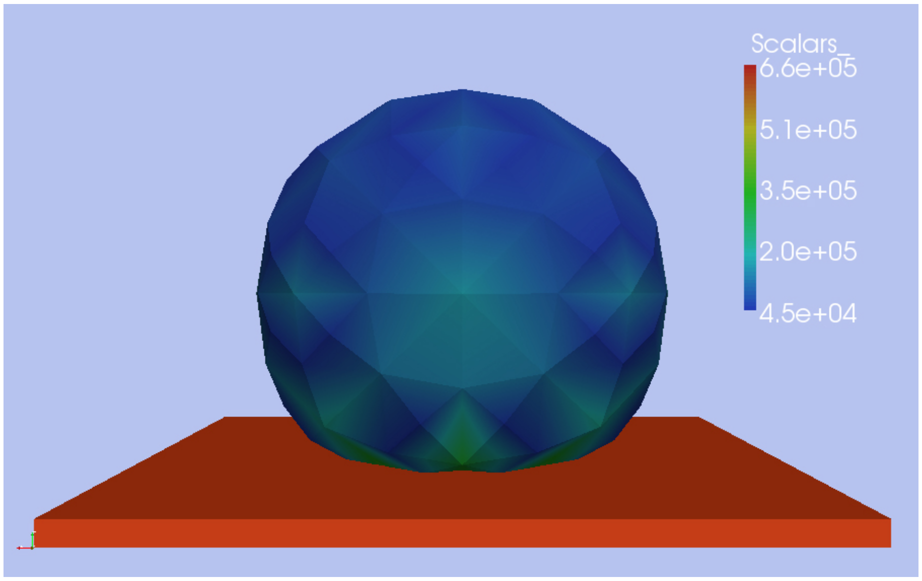

Figure 5.10: Pressure on surface of membrane at impact.

Furthermore, the conserved quantities (i.e. mass, energy, linear momentum, angular momentum) are plotted in figures 5.11, 5.16, 5.13, 5.14, 5.15, and 5.2.2, respectively.

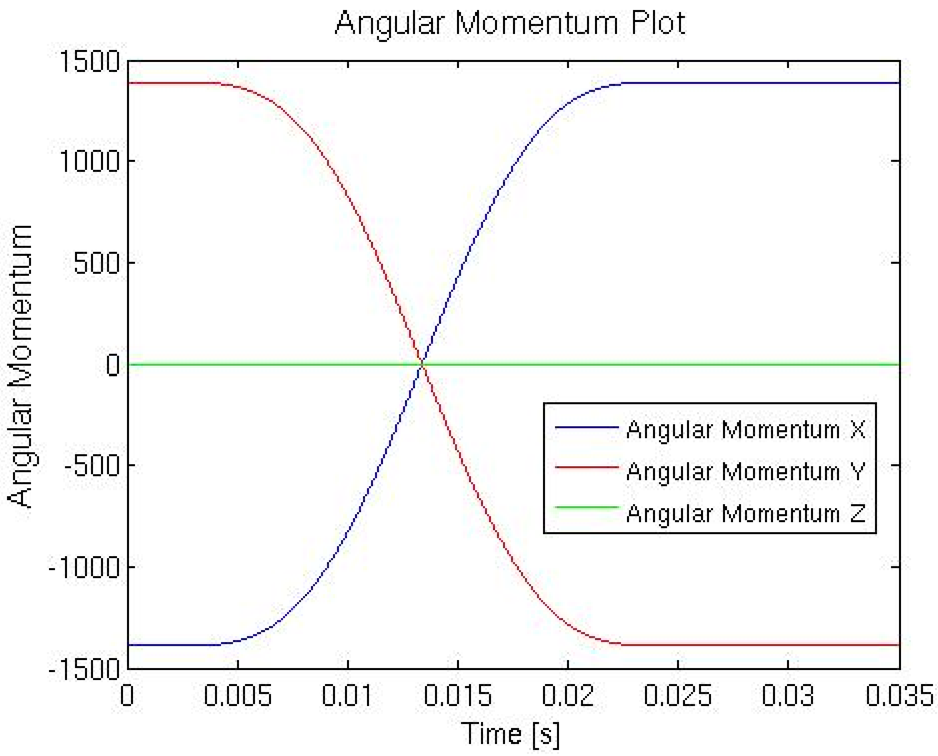

Figure 5.12: Angular momentum vs time.

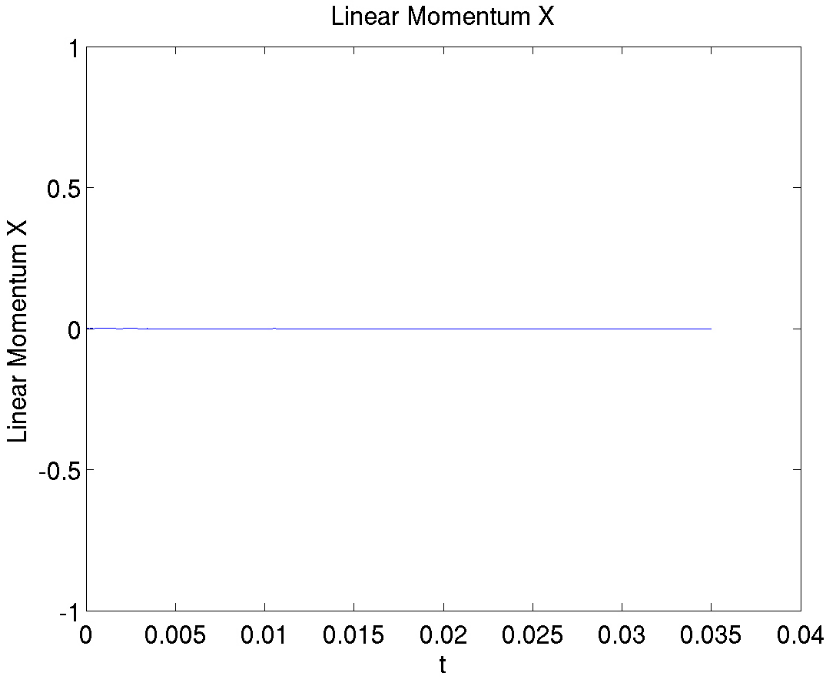

Figure 5.13: Linear momentum (x component) vs time.

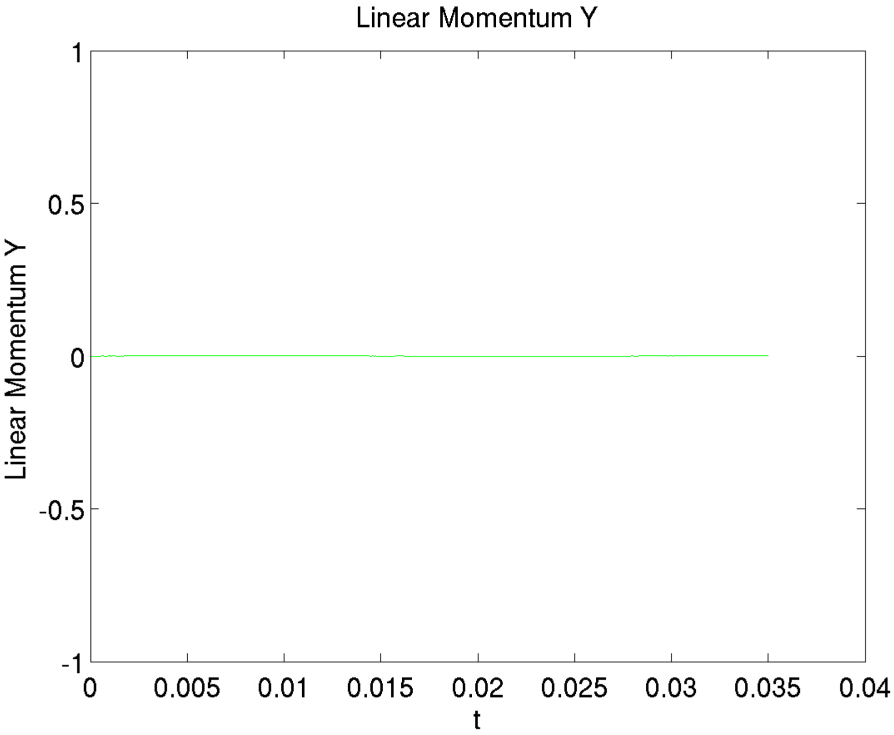

Figure 5.14: Linear momentum (y component) vs time.

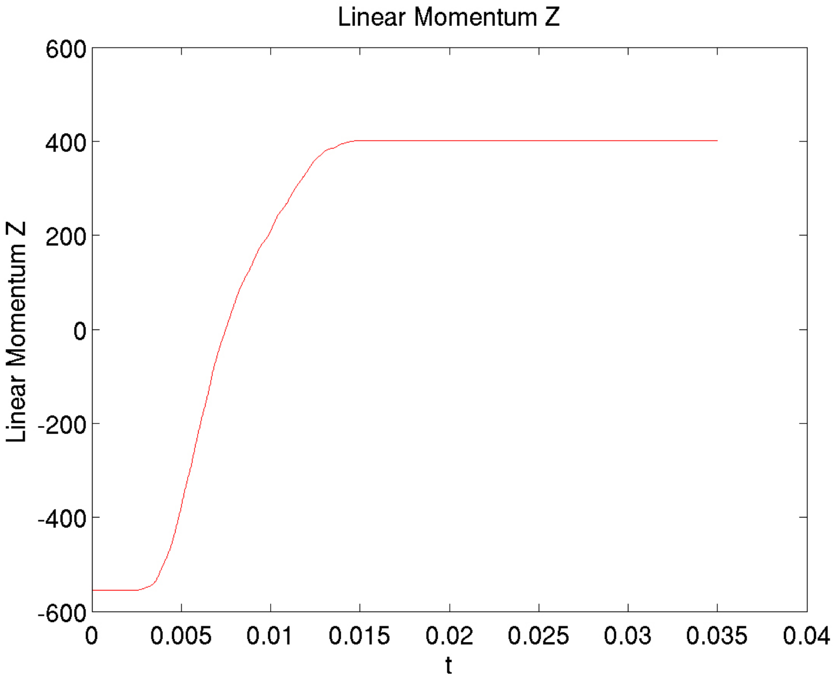

Figure 5.15: Linear momentum (z component) vs time.

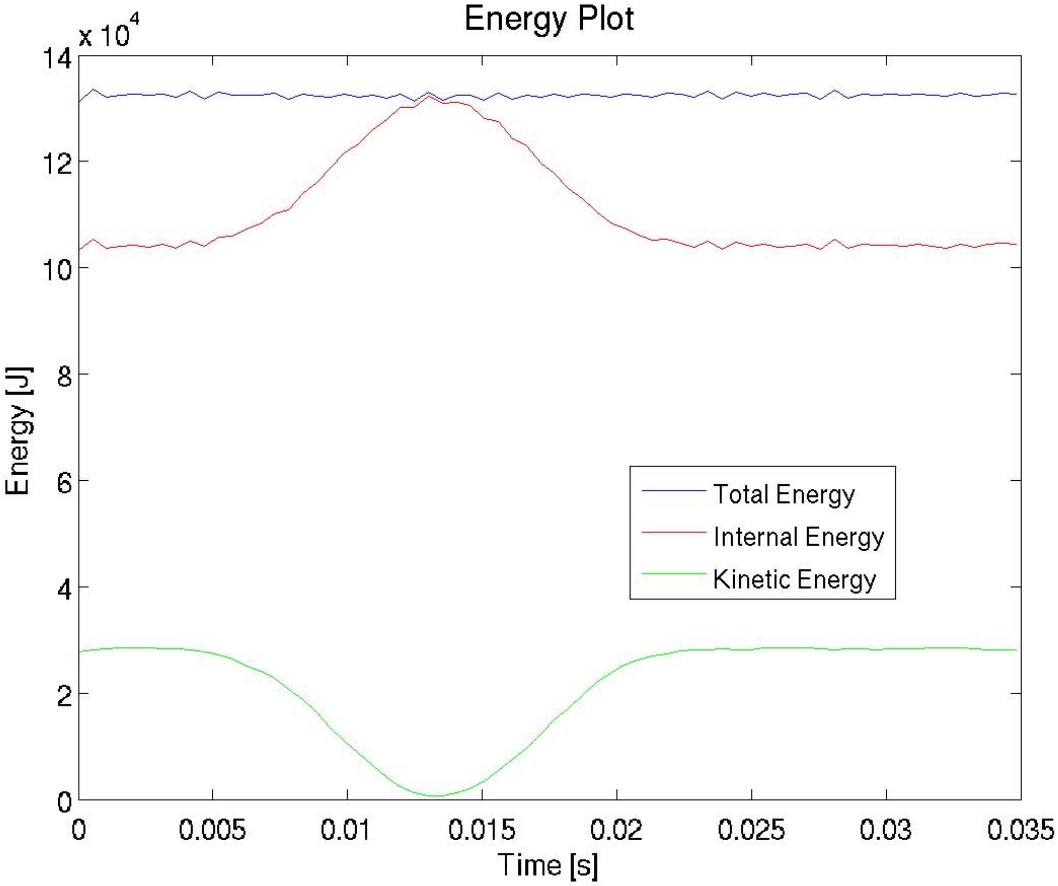

Figure 5.16: System energy vs time.

While there is still room for improvement (mostly by considering larger models with more neighbors), these results demonstrate first-of-a-kind capabilities of the OTM method in the sense of the meshfree methods for fluids as well as compared with traditional methods for coupled FSI calculations. Here, the size of the numerical models is limited by the calculation of the LME shape functions. Numerical methods for efficiently calculating the LME shape functions for dynamic simulations remain an open problem requiring additional developments. Higher resolution results will be presented in the forthcoming publication.

Chapter 6

Conclusions

This thesis developed the OTM method as a new meshfree method for numerically simulating fluid flows without regard to a numerical mesh. Here, the OTM method represents one approach within the broad field of particle and meshfree numerical methods. Having reviewed the current, popular meshfree methods, it is apparent that there is no one *best* approach. However, the OTM provides a unique paradigm for meshfree methods by providing a rigorous mathematical formulation including a number of salient attributes including: geometrically exact mass density updates, easily implemented essential boundary conditions, and trivial coupling with finite elements.

By employing LME shape functions the OTM method avoids the inherent obstacle of enforcing essential boundary conditions found in the other methods. Indeed, the flexibility and ease with which the OTM couples with the traditional FEM approach is remarkable when compared with other meshfree methods that require extreme care and consideration for individual cases of coupling.

Furthermore, the ability to solve Navier-Stokes equations in three dimensions with a general, moving domain alone makes the OTM method exceedingly desirable. In this thesis we verified the ability of the OTM method to capture shock waves by simulating the standard Riemann benchmark example. In the chapter 5, an illustrative FSI example was simulated to highlight the ability of the OTM method to simulate three-dimensional compressible Navier-Stokes flows coupled with a deforming membrane boundary modeled by finite elements.

Future work

While the initial aim of these simulations was to simulate the complete EDL airbag landing sequence, the computational expense in the current numerical framework (EUREKA), prohibits large calculations. To move forward with larger calculations (i.e. $\sim$10,000 - 100,000 material points) it is imperative to develop a parallel computational scheme. Specifically, while the calculation of the LME shape functions entails significant computational costs, the evaluation of the LME shape functions can be performed in parallel, which would significantly speed up the simulation. Furthermore, with regards to the LME shape functions, there are still many open questions regarding how to efficiently evaluate $\lambda^*(x_{mp})$ that may require new or novel combinations of numerical optimization techniques.

Finally, there is a plethora of viable applications. Up to this point we have considered internal flows, however, there is no requirement placed by the OTM method restricting simulations involving external flows. With regard to the illustrative FSI example in 5.2.1, better results will be presented in the forthcoming publicationOverall, the OTM method represents an exciting new meshfree approach facilitating numerical simulations of general three-dimensional fluid flows combined with complex, and possibly moving, boundaries that can be directly coupled with other Lagrangian numerical methods.

Bibliography

[1] B. Boots A. Okabe and K. Sugihara. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. Probability and Mathematical Statistics. John Wiley and Sons, 1992.

[2] C. Villani. *Topics in Optimal Transportation Theory*, volume 58 of *Graduate Studies in Mathematics*. American Mathematical Society: Providence, Rhode Island, 2003.

[3] R. D. Braun P. H. Kallemeyn D. A. Spencer, R. C. Blanchard and S. W. Thurman. Mars Pathfinder entry, descent, and landing reconstruction. *Journal of Spacecraft and Rockets*, 36(3):357–366, 1999.

[4] M. Arroyo and M. Ortiz. Local *Maximum-Entropy* approximation schemes: A seamless bridge betweel finite elements and meshfree methods. *International Journal for Numerical Methods in Engineering*, 65:2167–2202, 2006.

[5] K. J. Bathe. *Finite Element Procedures*. Prentice Hall, 1996.

[6] J. H. Ferziger and M. Peric. *Computational Methods for Fluid Dynamics*. Springer-Verlag, 2002.

[7] C. B. Laney. *Computational Gasdynamics*. Cambridge University Press, 1998.

[8] S. Guiliani J. Donea and J. P. Halleux. An arbitrary Lagrangian-Eulerian finite element method for transient fluid-structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 33:689–723, 1982.

[9] M. Lesoinne C. Farhat and P. LeTallec. Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity. *Computer Methods in Applied Mechanics and Engineering*, 157:95–114, 1998.

[10] S. Rugonyi and K. J. Bathe. On finite element analysis of fluid flows fully coupled with structural interactions. *CMES - Computer Modeling in Engineering & Sciences*, 2:195–212, 2001.

[11] T. Tezduyar K. Stein, R. Benney and J. Potvin. Fluid-structure interactions of a cross parachute: Numerical simulation. *Computer Methods in Applied Mechanics and Engineering*, 191:673–687, 2001.

[12] P. Thoutireddy and M. Ortiz. A variational r-adaption and shape-optimization method for finite-deformation elasticity. *International Journal of Numerical Methods in Engineering*, 61:1–21, 2004.

[13] B. Merriman R. P. Fedkiw, T. Aslam and S. Osher. A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *Journal of Computational Physics*, 151:457–492, 1999.

[14] C.S. Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 10:220–252, 1977.

[15] C.W. Hirt and B.D. Nichols. Volume of fluid (vof) method for the dynamics of free boundaries. *Journal Computational Physics*, 39:382–408, 1981.

[16] J. Glimm and D. Marchesin. A numerical method for two phase flow with an unstable interface. *Journal Computational Physics*, 39:171–200, 1981.

[17] R. P. Fedkiw. Coupling an Eulerian fluid calculation to a Lagrangian solid calculation with the ghost fluid method. *Journal of Computational Physics*, 175:200–224, 2002.

[18] G. Arfken. *Mathematical Methods for Physicists*. Academic Press, 2nd edition, 1970.

[19] B. J. Alder and T. E. Wainright. Phase transition for a hard sphere system. *Journal of Chemical Physics*, 27:1208–1209, 1957.

[20] L. B. Lucy. Numerical approach to testing the fission hypothesis. *Astronomical Journal*, 82:1013–1024, 1977.

[21] R.A. Gingold and J.J. Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Mon. Not. Royal Astronomy Society*, 181:375–389, 1977.

[22] I. A. Bonnell M. R. Bate and V. Bromm. The formation of a star cluster predicting the properties of stars and brown dwarfs. *Monthly Notices of the Royal Astronomical Society*, 399:577–599, 2003.

[23] A. Colagrossi and M. Landrini. Numerical simulation of interfacial flows by smoothed particle hydrodynamics. *Journal of Computational Physics*, 191:448–475, 2003.

[24] W. Benz and E. Asphaug. Simulations of brittle solids using smoothed particle hydrodynamics. *Computer Physics Communications*, 87:253–265, 1995.

[25] J. J. Monaghan. Why particle methods work. *Siam Journal Scientific Computation*, 3(4):423–433, 1982.

[26] J. P. Morris. *Analysis of Smoothed Particle Hydrodynamics with Applications*. PhD thesis, Monash University, Melbourne, Australia, 1996.

[27] P. W. Randles and L. D. Libersky. Recent improvements in sph modeling of hypervelocity impact. *International Journal of Impact Engineering.*, 20:525–532, 1997.

[28] M. W. Heinstein F. J. Mello J. W. Swegle, S. W. Attaway and D. L. Hicks. An analysis of smoothed particle hydrodynamics. Sandia Report SAND93-2513, Sandia National Laboratories, Albuquerque, NM 87185, 1994.

[29] C. T. Dyka and R. P. Ingel. An approach for tensile instability in smoothed particle hydrodynamics. *Computation and Structures*, 57:573–580, 1995.

[30] W. K. Liu T. Belytschko, Y. Guo and S. P. Xiao. A unified stability analysis of meshless particle methods. *International Journal of Numerical Methods in Engineering*, 48:1359–1400, 2000.

[31] T. Belytschko T. Rabczuk and S.P.Xiao. Stable particle methods based on Lagrangian kernels. *Computational Methods in Applied Mechanical Engineering*, 193:1035–1063, 2004.

[32] S. Jun W. K. Liu and Y.F. Zhang. Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids*, 20:1081–1106, 1995.

[33] D. Organ M. Fleming T. Belytschko, Y. Krongauz and P. Krysl. Meshless methods: An overview and recent developments. *Computational Methods in Applied Mechamical Engineering*, 139:3–47, 1996.

[34] D. Diachin F. Gunther, W. K. Liu and M. A. Christon. Multi-scale meshfree parallel computations for viscous, compressible flows. *Computational Methods in Applied Mechanical Engineering*, 190:279–303, 2000.

[35] G. J. Wagner L. T. Zhang and W. K. Liu. Modeling and simulation of fluid structure interaction by meshfree and fem. *Communications in Numerical Methods in Engineering*, 19:615–621, 2003.

[36] F. H. Harlow. Hydrodynamic problems involving large fluid distortion. *J. Assoc. Compos.*, 4:137, 1957.

[37] F. H. Harlow. The particle-in-cell method for numerical solutions of problems in fluid dynamics. *Methods in Computatinoal Physics*, 3:319–343, 1964.

[38] S. J. Zhou D. Sulsky and H. L. Schreyer. Application of a particle-in-cell method to solid mechanics. *Computation Physics Community.*, 87:136–252, 1995.

[39] Z. Chen and R. Brannon. An evaluation of the material point method. Sandia Report SAND2002-0482, Sandia National Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550, February 2002.

[40] D. Sulsky A. R. York II and H. L. Schreyer. The material point method for simulation of thin membranes. *International Journal for Numerical Methods in Engineering*, 44:1429–1456, 1999.

[41] D. Sulsky A. R. York II and H. L. Schreyer. Fluid-membrane interaction based on the material point method. *International Journal for Numerical Methods in Engineering*, 48:901–924, 2000.

[42] L. Rosenhead. The spread of vorticity in the wake behind a cylinder. *Proceedings of the Royal Society London*, A 127:590–612, 1930.

[43] A. J. Chorin and P. S. Bernard. Discretization of a vortex sheet, with an example of roll-up. *Journal of Computational Physics*, 13 (3):423–429, 1973.

[44] A. Leonard. Vortex methods for flow simulation. *Journal of Computational Physics*, 37:289–335, 1980.

[45] G. H. Cottet and P. D. Koumoutsakos. *Vortex Methods: Theory and Practice*. Cambridge University Press, 2000.

[46] T. Colonius J. D. Eldredge and A. Leonard. A vortex particle method for two-dimensional compressible flow. *Journal of Computational Physics*, 179:371–399, 2002.

[47] P. Koumoutsakos. Multiscale flow simulations using particles. *Annual Review of Fluid Mechanics.*, 37:457–487, 2005.

[48] A. Leonard L. A. Barba and C. B. Allen. Vortex method with meshless spatial adaption for accurate simulation of viscous, unsteady vortical flows. *International Journal for Numerical Methods in Fluids*, 47:841–848, 2005.

[49] P. Lancaster and K. Salkauskas. Surfaces generated by moving least squares methods. *Mathematics of Computation*, 37:141–158, 1981.

[50] J. Dolbow T. Belytschko, Y. Krongauz and C. Gerlach. On the completeness of meshfree particle methods. *International Journal for Numerical Methods in Engineering*, 43:785–819.

[51] G. Touzot B. Nayroles and P. Villon. Generalizing the finite element methods: Diffuse approximation and diffuse elements. *Computational Mechanics*, 10:307–318, 1992.

[52] Y. Y. Lu T. Belytschko and L. Gu. Element-free Galerkin methods. *International Journal of Numerical Methods in Engineering*, 37:229–256, 1994.

[53] D. Organ T. Belytschko and Y. Krongauz. A coupled finite element-element-free Galerkin method. *Computational Mechanics*, 17:186–195, 1995.

[54] J. Dolbow and T. Belytschko. Numerical integration of the Galerkin weak form in meshfree methods. *Computational Mechanics*, 23:219–230, 1999.

[55] T. P. Fries and T. Belytschko. Convergence and stabilization of stress-point integration in mesh-free and particle methods. *International Journal for Numerical Methods in Engineering*, 74:1067–1087, 2008.

[56] S. Yoon J. S. Chen, C. T. Wu and Y. You. A stabilized conforming nodal integration for Galerkin mesh-free methods. *International Journal for Numerical Methods in Engineering*, 50:435–466, 2001.

[57] T. Belytschko and S. P. Xiao. A bridging domain method for coupling continua with molecular dynamics. *Computer Methods in Applied Mechanics and Engineering*, 193:1645–1669, 2004.

[58] S. Fernandez-Mendez A. Huerta and W. K. Liu. Enrichment and coupling of the finite element and meshless methods. *International Journal for Numerical Methods in Engineering*, 48:1615–1636, 2000.

[59] Y. Krongauz and T. Belytschko. Enforcement of essential boundary conditions in meshless approximations using finite elements. *Computer Methods in Applied Mechanics and Engineering*, 131:133–145, 1996.

[60] I. Babuska and J. M. Melenk. The partition of unity finite element method. *International Journal of Numerical Methods in Engineering*, 40:727–758, 1997.

[61] C. A. Duarte and J. T. Oden. An hp adaptive method using clouds. *Computational Methods in Applied Mechanics and Engineering*, 139:237–262, 1996.

[62] S. N. Atluri and T. Zhu. A new meshless local Petrov-Galerkin (mlpg) approach in computational mechanics. *Computational Mechanics*, 22:117–127, 1998.

[63] H. Lin and S.N. Atluri. The meshless local Petrov-Galerkin (mlpg) method for solving incompressible navier-stokes equations. *CMES*, 2:117–142, 2001.

[64] S. N. Atluri and T. Zhu. New concepts in meshless methods. *International Journal for Numerical Methods in Engineering*, 47:537–556, 2000.

[65] S. N. Atluri. *The Meshless Method for Domain and Bie Discretizations*. Tech Science Press, 2004.

[66] B. Moran N. Sukumar and T. Belytschko. The natural element method in solid mechanics. *International Journal for Numerical Methods in Engineering*, 43:839–887, 1998.

[67] N. Sukumar. Recent advances in the construction of polygonal finite element interpolants. *Archives of Computational Methods in Engineering*, 13(1):129–163, 2006.

[68] E. Oñate S. R. Idelsohn and F. Del Pin. A Lagrangian meshless finite element method applied to fluid-structure interaction problems. *Computers and Structures*, 81:655–671, 2003.

[69] W. K. Liu S. Hao and T. Belytschko. Moving particle finite element method with global smoothness. *International Journal for Numerical Methods in Engineering*, 59(7):1007–1020, 2004.

[70] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.

[71] V. T. Rajan. Optimality of the Delaunay triangulation in R^d. *Discrete and Computational Geometry*, 12(2):189–202, 1994.

[72] T. Kawashima H. Noguchi and T. Miyamura. Element free analyses of shell and spatial structures. *International Journal for Numerical Methods in Engineering*, 47:1215–1240, 2000.

[73] S. Fernandez-Mendez A. Huerta and W. K. Liu. A comparison of two formulations to blend finite elements and mesh-free methods. *Computer Methods in Applied Mechanics and Engineering*, 193:1105–1117, 2004.

[74] R. Becker. Mesh adaptation for dirichlet flow control via nitsche's method. *Communications in Numerical Methods in Engineering*, 18(9):669–680, 2002.

[75] J. Freund and R. Stenberg. On weakly imposed boundary conditions for second order problems. In *Proceeding of the International Conference on Finite Elements in Fluids - New Trends and Applications*, Venezia, 1995.

[76] P. Hansbo and M. G. Larson. Discontinuous Galerkin methods for incompressible and nearly incompressible elasticity by Nitsche's method. *Computer Methods in Applied Mechanics and Engineering*, 191(17-18):1895–1908, 2002.

[77] M. W. Heinstein S. W. Attaway and J. W. Swegle. Coupling of smoothed particle hydrodynamics with the finite element method. *Nuclear Engineering and Design*, 150:199–205, 1994.

[78] G. R. Liu and M. B. Liu. *Smoothed Particle Hydrodynamics: a meshfree particle method*. World Scientific, 2003.

[79] G. Monge. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie Royale des Sciences de Paris*, pages 666–704, 1781.

[80] L. V. Kantorovich. On the translocation of masses. *C. R. (Doklady) Acad. Sci. URSS (N. S.)*, 37:199–201, 1942.

[81] Y. Brenier. Extended Monge-Kantorovich theory. Technical report, CIME, 2001.

[82] J. D. Benamou and Y. Brenier. A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem. *Numerische Mathematik*, 84(3):375–393, 2000.

[83] Y. Brenier J.-D. Benamou and K. Guittet. The Monge-Kantorovich mass transfer and its computational fluid mechanics formulation. *International Journal for Numerical Methods in Fluids*, 40:21–30, 2002.

[84] Y. Brenier J.-D. Benamou and K. Guittet. Numerical analysis of a multi-phasic mass transport problem. *Contemporary Mathematics*, 353:1–18, 2004.

[85] W. Gangbo and R. J. McCann. Shape recognition via Wasserstein distance. *Quarterly of Applied Mathematics*, 58(4):705–737, 2000.

[86] K. Guittet. On the time-continuous mass transport problem and its approximation by augmented Lagrangian techniques. *SIAM Journal on Numerical Analysis*, 41:382–399, 2003.

[87] F. Otto. The geometry of dissipative evolution equation: the porous medium equation. *Communications in Partial Differential Equations*, 26(1-2):101–174, 2001.

[88] L. C. Evans. *Partial Differential Equations and Monge-Kantorovich Mass Transfer*. Current Developments in Mathematics. International Press: Cambridge, MA, 1997.

[89] J.-D. Benamou and Y. Brenier. A numerical method for the optimal time-continuous mass transport and related problems. In *Monge-Ampére Equation: Applications to Geometry and Optimization*, pages 1–11. American Mathematical Society: Providence, Rhode Island, 1999.

[90] R. Radovitzky and M. Ortiz. Lagrangian finite element analysis of Newtonian fluid flows. *International Journal of Numerical Methods in Engineering*, 43:607–619, 1998.

[91] R. Radovitzky and M. Ortiz. Error estimation and adaptive meshing in strongly nonlinear dynamic problems. *Computer Methods in Applied Mechanics and Engineering*, 172:203–240, 1999.

[92] G. A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics*, 27:1–31, 1978.

[93] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. Springer, 2nd edition, 1999.

[94] R. Radovitzky A. Lew and M. Ortiz. An artificial-viscosity method for the Lagrangian analysis of shocks in solids with strength on unstructured, arbitrary-order tetrahedral meshes. *Journal of Computer-Aided Materials Design*, 8:213–231, 2001.

[95] J. Von Neumann and R. D. Richtmyer. A method for the numerical calculation of hydrodynamic shocks. *Journal of Applied Physics*, 21:232–237, 1950.

[96] C. H. M. Jenkins, editor. *Gossamer Spacecraft: Membrane and Inflatable Structures Technology for Space Applications*. Vol. 191 Progress in Astronautics and Aeronautics. AIAA, 2001.

[97] W. J. Larson and J. R. Wertz, editors. *Space Mission Analysis and Design*. Space Technology Series. Microcosm Press, 1999.

[98] T. Chase J. Whetzal, C. H. Jenkins and J. Sears. Recent advances in ultra-lightweight mirror technology. Technical report, September 2004.

[99] S. Mauch. *Efficient Algorithms for Solving Static Hamilton-Jacobi Equations*. PhD thesis, California Institute of Technology, 2006.