Integrating Peridynamics to Material Point Method for Modelling Solids and Fracture Dynamics in High Velocity Impact

by

Yao Lyu

National Centre for Computer Animation

Faculty of Media & Communication

Bournemouth University

A thesis submitted in partial fulfilment of the requirements of Bournemouth University for the degree of *Doctor of Philosophy*

Oct. 2021

Copyright Statement

Acknowledgements

I wish to express my sincere thanks to my supervisors, Professor Jian Chang and Professor Jian J. Zhang, for their continuous support of my PhD study and research. Their insight and expertise led me through the nebulous path to becoming a researcher. I would particularly like to single out my supervisor Professor Jian Chang. He participated in every crunch, guiding me to embrace the wildest vision for research while breaking down problems to the finest details. Without his guidance and encouragement this thesis would never have materialised.

I would like to thank Ari Sarafopoulos and Dr. Shihui Guo who offered their valuable experience and advice for my research. I would like to thank Dr. Ouwen Li, Dr. Jinglu Zhang, Dr. Xiaokun Wang, Dr. Yinyu Nie and Dr. Min Jiang who offered unconditional help on various aspects in my research. I also would like to thank Sonia Ashby and Cansu Kurt Green who handled all the administrative tasks and cared my student life in Bournemouth University.

In addition, I would like to thank my awesome friends Mengqing Huang, Junheng Fang, Nan Xiang, Xiaoxiao Liu, Ruibin Wang, Yu Zhou, Dr. Tao Jiang, Dr. Yanran Li, Zhangcan Ding, Dr. Alexander Tereshin, Pei Tang and Kaixi Li for the invaluable friendship.

My great tribute is due to my parents for their loving consideration and support through my long student life. You are always there for me through all the ups and downs.

Finally, I gratefully acknowledge Chinese Scholarship Council for their financial support. I would extend my regards and blessings to all of those who supported me in any respects during the completion of my PhD.

Abstract

The desire for graphical methods to intuitively handle elastoplastic materials has grown hand in hand with the advances made in computer Graphics. Simulating physical materials with dynamic movements to photo-realistic resolution is still one of the most crucial and challenging topics, especially involving fractures. Material Point Method (MPM) presents a strong approach for animating elastoplastic materials due to its natural support for arbitrarily large topological deformations and intrinsic collision handling. However, the partial derivative based MPM brings underlying instability issue of handling discontinuous particle distributions and requires computationally expensive treatments to separate broken pieces. The objective of this thesis is to propose a novel MPM solver for robustly and intuitively animating scenarios containing fractures. We are inspired by Peridynamics (PD) which is oriented toward deformations with discontinuities. This study exploits the PD within the MPM scheme to mitigate the difficulties inherent in handling fractures.

First, we propose an integral-based MPM by adopting a PD integral energy density function to the MPM weak form and following the standard MPM discretization scheme. Novel elastic, plastic, viscoelastic and fracture models encoding PD bond concepts are designed as constitutive models. The integral-based MPM outweighs the differential-based MPM in both accuracy and stability.

To efficiently model myriad fragments with a MPM solver (especially in high speed impact scenarios), our second contribution

is to formulate a rigorous coupling governing equation which integrates the state-based PD within the MPM scheme (Superposition-based MPM) that features an automatic fractures modelling scheme. In SPB-MPM, PD evolves as a result of failure evolution in critical regions while the MPM derives entire problem domain. Giving a low-overhead PD computation to the MPM, this method allows for simulating a breadth of fracture effects, including ductile and brittle fractures.

The prominent features at high strain rate in high velocity impact are unattainable through general constitutive models. Our third contribution is to introduce a shock wave effects model and a metallic plastic model which are designed to capture the intricate and characteristic impact behaviours. We simulate a number of representative impact scenarios, including organic fruits, metallic materials and multi-material deformable objects, demonstrating the efficacy of our models.

Contents

List of Figures

List of Tables

Nomenclatures

Symbols	Description
ϵ	strain tensor
σ	Cauchy stress tensor
ρ	density
b	the body force
a	accelaretion
ϕ	total potential energy
ψ	potential energy density function
t^n	time at step n
Δ	time between adjacent two steps
θ_c	plastic critical compression parameter
θ_s	plastic critical stretching parameter
m_p	mass of particle p
v_p	velocity of particle p
x_p	position of particle p
V_p	volume of particle p

Material Point Method

C_p	affine matrix of particle p
M_p	quadratic particle-grid kernel
F_p	deformation gradient on particle p
F_{Ep}	elastic part of deformation gradient on particle p
F_{Pp}	plastic part of deformation gradient on particle p
ω_{ip}	weight function between particle p and node i
m_i	mass of node i
v_i	velocity of node i
x_i	position of node i
f_i	force on grid node i
Δx	size of one grid cell

Peridynamics

H_p	neighbour horizon of particle p
δ	the finite distance of PD horizon
X	original bond between x_p and x_q
Y	deformed bond between x_p and x_q
U	bond displacement
$T[x_q, t] \langle x_q - x_p \rangle$	force densities exerted on the bond Y
s_{pq}	connection status of bond between x_p and x_q
$\omega(X)$	weighting function of bond X
F_p	deformation gradient of particle p in the state-based PD
K_p	shape tensor of particle p in the state-based PD

Chapter 1

Introduction

1.1 Motivation

When a piece of cloth on the drying line flutters in the breeze, it presents unique wrinkles and bending features. A bullet shot at a fresh watermelon causes chunky melon rind to be peeled into the air and its juicy pink flesh to explode while mist sprays under the high impact forces. Falling containers are devastated after colliding with the ground and are left with irreversibly wrinkled bodies. After a wall is smashed by a metal bullet, the resulting myriad debris from the impact region propagates with the smashed bullet. The desire for graphical methods to intuitively animate deformable objects with physical properties has grown hand in hand with the advances made in computer graphics. However, simulation of photo-realistic resolution, especially of elastoplastic materials, is one of the most crucial and challenging topics due to the presence of severe shape changes and fracture fragments.

Impact scenario, containing intricate topological structures and myriad fragments with branching cracks, is a notoriously complex phenomenon to model. In particular, in high velocity impacts, object fragments encompass a wide spectrum of spatial scales, ranging from the immense number of particle-level debris formed by point clouds, to the broken chunks with irregular shapes. To model these phenomena requires one graphic method to be capable of stably managing severe deformed typologies, intuitively handling crack propagation and accurately detecting collision among numerous

fragments. Researchers take on a variety of approaches to model large deformation with fractures, these differing on the choice of discretization of the governing equations.

Eulerian methods are among those successful physically-based approaches in handling elastic solids undergoing large deformation (Levin et al. 2011). Through discretising space instead of object itself, Eulerian methods are free from mesh-distortion and naturally have a collision detection strategy but needs additional efforts to track the interface. Conversely, Lagrangian methods, through describing simulations with degrees-of-freedom (DOFs) that move with deformed objects, is able to explicitly track a object's deformed configuration in the spatial domain. Meshless Lagrangian methods, using particle representation, such as, Smoothed Particle Hydrodynamics (SPH) (Becker et al. 2009) and Position Based Dynamics (PBD) (Müller et al. 2007), show success in handling incompressible fluid and various deformable solid materials without the need of special mesh treatments while Lagrangian methods require collision detection mechanism to accurately handle interactions. Above approaches generally excel at some phenomena but would stumble(if not fail) at others.

Material Point Method (MPM), proposed as the generalisation of the Particle-In-Cell and Fluid Implicit Particle Method (FLIP) (Brackbill and Ruppel 1986) to solid mechanics, has been shown to be a strong hybrid Eulerian-Lagrangian method for simulating various materials (Sulsky et al. 1995). Stomakhin et al. (2013) first introduced the MPM to computer graphics for simulating snow. The MPM uses particles to track Lagrangian quantities and a background grid to accurately evaluate derivatives for computing forces. With the particle-grid transfer procedure, it has a natural ability to support arbitrarily large topological changes and a built-in collision detection mechanism. This method has been successfully applied in a diverse set of real life scenes including flowing granular media (Stomakhin et al. 2013, Klár et al. 2016), organic isotropic and anisotropic materials (Wolper et al. 2020). The MPM has also demonstrated its success in traditional mechanical applications. This is showcased by various simulations of scenarios involving impacts (Wallstedt and Guilkey 2007), blast induced fragmentation (Hu and

Chen 2006), high explosives with multiple hardening strategies (Ma et al. 2009a) and multi-scale problems (Ma et al. 2009b). The MPM shows great potential in modelling versatile materials. However, due to that the MPM involves the partial derivatives of particle displacement in stress formulation, one well-known issue of the traditional MPM is underlying numerical instability brought by animating discontinuous particle distribution. This formulation precludes the MPM for studies involving fractures. Wolper et al. (2019) initialised the Lagrangian particles with a phase field description which is capable to model crack propagation within a continuum body. This method shows promise for fractures producible by the MPM with some augmentations.

Standing in contrast to above methods involving partial differential equations in which spatial derivatives are not well defined at discontinuities, PD discretizes the momentum conservation equation using an integral formulation of particle displacement (Silling 2000, Silling et al. 2007, Silling 2010). So that the PD governing equations remains valid at discontinuities, and material damage (accumulated by micro-scale broken bonds) can be represented as part of constitutive models. PD is favourable to handle material discontinuities, including fracture initiation and propagation with arbitrary paths with deformable objects in computational physics community (Silling 2018). Computer graphics methods has also exploited to simulate fractures from Peridynamics perspective (He et al. 2017, Xu et al. 2018). PD is not without its drawbacks. It inherits the shortcomings of meshless Lagrangian methods, including incapable of handling intrinsic collision detection and requiring special treatments for boundary issues. Moreover, under extreme topology deformation, particles are distributed extremely different from initial configuration so that the bond connection becomes invalid, in which cases the fractures cannot be modelled properly.

This study is initially motivated by current needs towards photo-realistic fracture simulation in game and film industry. With the observation of impact scenarios which present chunky broken pieces with stable shapes and expanding debris clouds, the physics solver is required to be capable of eliminating existing numerical instability issues of modelling discontinuities and

capturing compelling fractures in a physically plausible manner. Our target is to tackle the difficulties of the state-of-art approaches in modelling fractures, especially in impact scenarios. More specifically, we aim to integrate PD as a powerful technique of handling arbitrary fractures to MPM and design a robust MPM solver. Our empirical model is based on phenomenological observations and examined by the theory devised by engineering applications. Our works have been kept efficient, allowing us to capture sufficient geometric detail with tractable computation time.

1.2 Main Challenge

Although related technologies of modelling elastoplastic deformation have been well studied in general computer graphic field, each with varying degrees of success, it is infeasible to directly use them to model fractures of impact scenarios by the fact that unpredictable fractures complicate the simulation. We summary main challenges in the state-of-art algorithms as follows:

Figure 1.1: Car crash effects in game Wreckfest.

1. Fig. 1.1 is an example of rich destruction effects in a car racing game, containing a wide number of fragments and debris cloud. While so-

phisticated graphical methods have been exploited in recent years, it appears that the demand of an enhanced method with more details, larger scenes and more needs for artistic control definitely brings huge computational burden.

2. Physics-based animation has matured, and today there is a wealth of graphical methods solving simulation problems, such as fluid and solid deformation. Most researches focus on exploring general life scenes, in which the movement of deformable object is within normal velocity range. Conversely, objects moving at high speed can generate intricate behaviours in a short time period because the consequent impact force usually has a greater effect than a smaller force applied over a proportionally longer time period. Moreover, the rapid impact generates shock waves. Existing methods which are designed for low-speed deformation cannot model such characteristic features.

3. The MPM stands as a powerful method for robustly simulating severe topological changes. This ability makes the MPM become a promising method to animate objects with extreme large deformation. However, the existing MPM adopts derivative-based stress formulation which leads to numerical instabilities representing the discontinuities. The well-known issue precludes the simulation where fractures appear in configuration, including branching cracks and propagating crack fronts. Existing MPM does need special treatments (i.e. multiple grid duplication (Wretborn et al. 2017)) for separating fragments, which is tedious and challenging to implement in impact scenarios. Most importantly, the MPM does not have a proper model for handling topological changes at large strain rate if a high-speed impactor is applied. All of above issues needs to be overcame for animating complex fracture scenarios.

1.3 Aims and Objectives

This thesis aims at investigating a novel method capable of intuitively animating fractures in impact scenarios. The MPM has showcased rich experience for simulating versatile materials while struggles to derive the discontinuities. Our focus is to alleviate the instability issues of the MPM in discontinuous particle distribution. Peridynamics, first proposed by Silling et al. (2007) as an nonlocal reformulation of classical solid mechanics using spatial integral governing equations instead of partial differential equations, demonstrates great potential in generating various fracture behaviours (He et al. 2017). In PD, unstructured particles are influenced directly by the material points located in their vicinity, leading to an accurate and easy implementation in simulating manifold crack dynamics. However, contrary to the MPM, PD brings computational cost when simulates an unbroken deformed object because each PD particle needs run iterations over its neighbourhood particles at each timestep. Our main inspiration is to integrate the strong ability of the PD in handling micro-scale fracture as an extension to the current MPM solver. Based on our aims, we identify our research questions and objectives as follows:

- **How to alleviate the numerical failures created by the discontinuities in the MPM?** The MPM uses the derivation of particle displacement to obtain stress tensor in the weak form. For discontinuous particle distribution, numerical errors may occur during the partial derivation computation throughout the simulation, leading to underlying instability issues.

- **How to efficiently modelling numerous debris created by impact forces?** Integrating the PD to the MPM obviously is a feasible solution to address instability issues of the traditional MPM while brings overhead computation to the entire MPM framework. In specific, looping the PD neighbourhood of each PD particle adds computational cost. To simulate impact scenarios containing myriad fragments and

numerous debris, superposing PD bond structures over the entire problem domain is inefficient. The challenge is to how to efficiently couple the PD and the MPM in a physically plausible manner.

- **How to describe the prominent features in high velocity impact scenarios?** The impact of a high speed impactor present ripples across the object, producing debris cloud behind the impact area and sometimes leading to varied material properties. Most hyperelastic constitutive models in computer graphics are designed for deformable objects in general scenarios which means deformation is set at low velocity strain rate and are not suitable to describe compelling effects at high strain rate. It is particularly difficult to propose a sophisticated model to capture above characteristic behaviours with visual realism and high efficacy.

Motivated by aforementioned challenges, this thesis aims to achieve three major objectives:

- Our first objective is to explore novel strategies of integrating the PD to the MPM solver. This coupling operation will enable us to better utilise the MPM to achieve stable deformed topology and intuitive fractures animation. This objective is achieved in Chapter 5.

- Our second objective is to improve the integration and seek for an efficient coupling scheme for intricate impact scenarios. Our main task is to obviate the difficulties of animating numerous fracturing fragments and unpredictable crack propagation in the MPM. This objective is achieved in Chapter 6.

- Our third objective is to propose constitutive models for modelling high speed impact scenarios. The prominent extreme volumetric explosion at large strain rate and shockwave transition within the objects need be considered. This objective is achieved in Chapter 7.

1.4 Contribution

The main contribution of this thesis is to incorporate the PD's strong ability of treating discontinuities in the MPM scheme. We achieve this idea in two different ways in terms of varied requirements in different applications. The detailed contributions in each chapter are summarised as follows:

- Our first contribution is to incorporate the integral-based force computation of PD to the MPM weak formulation, leading to an integral-based MPM, which outweighs the differential-based MPM in both accuracy and stability. We develop a simple yield function based on deviatoric flow theory to animate plasticity and an linearized PD theory to model viscoelastic materials within the augmented MPM solver. We provide an attractive method for producing a variety of elastoplastic materials and fracture with visual realism and high stability.

- For an efficient solution in impact scenes, our second contribution is to formulate a rigorous coupling governing equation which superposes the state-based PD over critical areas in MPM problem domain, resulting in a coupled solver: Superposition-based MPM (SPB-MPM). It is designed to efficiently capture shattered debris and severe topological changes in impact simulations. The SPB-MPM evolves PD region as a result of failure evolution. The PD is used as a natural choice of handling material fractures while MPM derives the entire configuration. Giving a low-overhead PD computation to current MPM scheme, we are allowed to simulate a breadth of fractures with the notable ability of arbitrarily large topological changes.

- Our third contribution is to propose a shock wave effects model and a metallic plastic model to describe the prominent features of high velocity impact, which are unattainable through common constitutive models. To demonstrate the volumetric response in shock-compressed solids, we adopt a simplified Mie-Grüenisen EOS as a hydropressure term in the constitutive model. To model metal behaviours under high strain rate, we introduce the Johnson-Cook strain and strain-rate

hardening scheme. Our models have been tested by simulating organic fruits, chocolate and other elastoplastic solids behaviours under impact condition. The results present intricate and characteristic features of high velocity impact scenarios.

1.5 List of Publications

- Lyu, Y., Zhang, J., Chang, J., Guo, S. and Zhang, J.J., 2019, June. Integrating peridynamics with material point method for elastoplastic material modeling. In Computer Graphics International Conference (pp. 228-239). Springer, Cham.
- Lyu, Y., Zhang, J., Sarafopoulos, A., Chang, J., Guo, S. and Zhang, J.J., 2020. Integral-Based Material Point Method and Peridynamics Model for Animating Elastoplastic Material. In Transactions on Computational Science XXXVII (pp. 91-108). Springer, Berlin, Heidelberg.
- Zhang, J., Lyu, Y., Wang, Y., Nie, Y., Yang, X., Zhang, J. and Chang, J., 2018, December. Development of laparoscopic cholecystectomy simulator based on unity game engine. In Proceedings of the 15th ACM SIGGRAPH European Conference on Visual Media Production (pp. 1-9).
- Zhang, J., Nie, Y., Lyu, Y., Li, H., Chang, J., Yang, X. and Zhang, J.J., 2020, October. Symmetric dilated convolution for surgical gesture recognition. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 409-418). Springer, Cham.

1.6 Outline of Thesis

This thesis is organised as follows:

- Chapter 2 reviews the state-of-the-art approaches for both Eulerian and Lagrangian dynamics methods. Following the overview of existing approaches, we briefly introduce Mass Spring Method, FEM and PIC/FLIP to help with understanding our methods.

- Chapter 3 discusses the deformation theory, two common constitutive models (Fixed Corotated model and Neo-Hookean model), elucidates the basic data flow of the MPM transfer procedures, and summarises two time integration methods.

- Chapter 4 introduces general concepts in current PD research, containing basic definitions in the bond-based PD, the state-based PD and failure laws.

- Chapter 5 proposes an integral-based MPM, which adds bonds for each material point and adopts an integral energy density function in the weak form.

- Chapter 6 introduces a coupling MPM scheme: SPB-MPM, which places the PD as local regions over particular parts of the MPM domain as an extension to handle fracture surfaces and crack routes.

- Chapter 7 presents a shock wave model and a metallic plastic model for simulation volumetric effects and metal hardening effects in high velocity scenarios.

- Chapter 8 concludes the paper and discuss the possible solutions to improving the current study.

Chapter 2

Literature Review

The MPM solver uses a Eulerian grid as the scratchpad and Lagrangian particles to carry physical quantities to simulate continuum configuration. This chapter starts with reviews of the state-of-art approaches in Eulerian, Lagrangian, hybrid Eulerian-Lagrangian view for simulating elastoplastic materials and brief introduction of several classic methods that will prove helpful in understanding the underlying mathematical framework of our works. With the growing needs of fracture effects in industry, our main concern is to extend current continuum mechanic theories with the ability of arbitrarily handling discontinuous particle distribution. Therefore, based on the basic insight into current dynamic methods, we also present an overview of material fracture approaches.

2.1 Eulerian View

In continuum mechanics, the deformation is usually represented with material space Ω_0 and deformed space Ω_t under deformation map $x = \phi(X, t)$ where x and X are world and material coordinates respectively. The deformation map (also known as flow map) $\phi(\cdot, t) : \Omega_0 \to \Omega_t$ for $\Omega_0, \Omega_t \subset R^d$, where d = 2 or 3 is the dimension of the simulated problem (or domain).

The introduction of Ω_0 and Ω_t brings with it two sets of coordinate systems, one for each frame of reference. In practice, these two are often taken to be identical, but when choosing one or another, the physical interpretation

changes. By representing a quantity as a function of the initial configuration $q(X,t)$, we describe how the state of the particles change during the simulation. This is known as the **Lagrangian view**, or material description. Instead, by using the current configuration $q(x,t)$ the change refers to fixed points in space which is a 3D Euclidean space where the laws of physics apply, and it is referred to as the **Eulerian view** or spatial description. The difference of above two description is shown in Fig. 2.1.

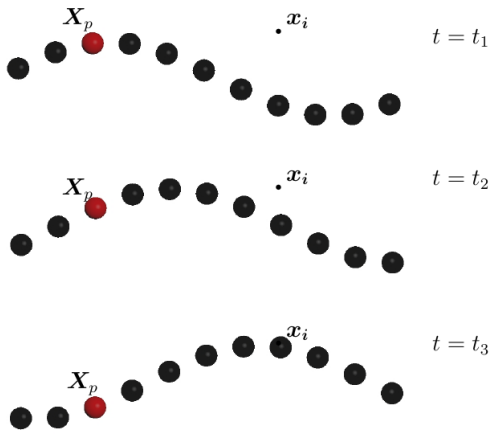

Figure 2.1: Difference between the Lagrangian and the Eulerian view. The X_p coordinate refers to the red particle, and moves in space (i.e. its world space coordinates x, y, z will change over time). The x coordinate is a world space coordinate, and is fixed in time.

Eulerian perspective well suites to some problems due to that it discretizes space instead of the object itself. For example, in graphical and biomechanical applications, the primary requirement is usually not the behaviour of the material (i.e. "will the material fail?") but how objects interact with each other in the physical world. The interaction usually is to deal with contact and other constraints, and the large deformations following these constraints. Through handling the dynamic fields, such as velocity field in the discretized space, Eulerian approaches can well accommodate these requirements.

One of its popular application is fluid simulation. Foster and Metaxas (1996) developed the first grid-based fully 3D water simulator in graphics

for realistically animating liquid phenomena. This work utilises the Navier-Stokes equations which couple momentum and mass conservation to completely describe fluid motion. Later, Carlson et al. (2002) augmented the Navier-Stokes equations with incompressible viscous properties and free surfaces and treated solid and nearly-solid materials as very high viscosity fluids to describe melting and solidification phenomena in a Eulerian manner. Goktekin et al. (2004) is built upon prior Eulerian methods for animating incompressible fluids with free surfaces by including additional elastic terms for elastic forces. These terms can be readily computed on rectilinear grids using a staggered discretization scheme, and the use of a Eulerian formulation easily accommodates modelling large flows. The main motivation for using Eulerian approaches in fluids simulation is that the regular grid simplifies the computation of spatial derivatives.

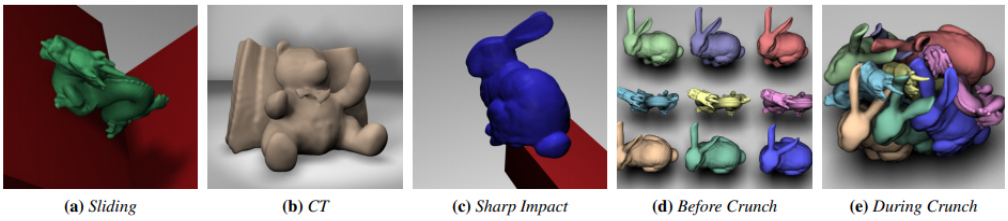

Figure 2.2: Eulerian solid simulation with contact (Levin et al. 2011)

For the elastoplastic solids, the constitutive equations are history dependent so material points must be followed. This is difficult to implement in a Eulerian scheme. The important difference between elastoplastic solids modelling and fluids simulation is that modelling fluids only requires physical quantities stored on the grid while solids simulations need reconstruct the material coordinates as the reference configuration. Even though, some researchers has addressed the problems and applied Eulerian methods to solids deformation for some particular problems through adapting techniques from computational fluid dynamics. The advantages of using Eulerian modelling solids are: 1) the nodes of the Eulerian mesh can be precisely collocated with the constraints. Levin et al. (2011) derived inequality constrained quadratic program, velocity level contact constraints and contact constraints on the

grid. This greatly reduces one algorithm's complexity as the same data structure is utilised for all stages of the simulation and the need to choose the resolution of the contact grid is removed. In such cases, constraints handling is arguably much more important than accurate representation of material behaviour. 2) Eulerian techniques are robust to large scale deformations because the mesh resolution is tied to the output in physical space rather than the rest shape of the material. Eulerian approaches have also been explored in mechanics field to study impact problems during which one or both participating objects are severely deformed. Benson (1995) adopted a Eulerian formulation as a feasible solution for contact and impact problems involving penetration and fracture. Tran and Udaykumar (2004) handled severe material deformation in a Eulerian setting on a fixed Cartesian mesh, in which well-developed high-accuracy shock capturing schemes are easily applied to compute nonlinear wave-propagation phenomena.

The reason for choosing one view or another depends on the problem of interest. For example, most fluids simulations use the Eulerian view partly due to the fact that it allows for arbitrarily large deformations. This is because the underlying mesh does not deform during the simulation. The Eulerian discretization is not without its drawbacks. In general, compared to a Lagrangian approach, Eulerian methods are free from mesh-distortion but additional efforts have to be made to track the interface. The mesh may be only partially covered by material. It is necessary to keep track of this coverage, and deal with it in the numerical methods. In terms of computing cost, the Eulerian method is faster than the Lagrangian method (Zhang and Chen 2007). Conversely, Lagrangian method gives detailed information of individual particles that can be crucial in many applications.

2.2 Lagrangian View

Terzopoulos and Fleischer (1988) pioneered the physically-based simulation of deformable solids in computer graphics, in which the simulation consists of DOFs that move with the deformed object, explicitly tracking an object's updated configuration in the spatial domain. This is a typical Lagrangian

approach. Methods of this type are usually distinguished by their selected topology definition: mesh-based methods or meshless methods. Finite element method (FEM) is among the first mesh-based methods to model solid deformation (O'brien and Hodgins 1999, O'brien et al. 2002). It still has imposed itself as currently one of the most powerful tool that enables highly efficient modelling and simulation of structures characterised by complex geometry and exposed to arbitrary boundary and initial conditions. DOFs in the FEM are stored on a volumetric mesh (on the vertices). Later, to reduce the required number of DOFs, Boundary element methods (BEM) gained some interests of the research (Hahn and Wojtan 2015). The BEM stores DOFs on a surface mesh and the boundary integral form of the governing equations is used instead of volumetric partial differential equations. These mesh-based methods well suites some simple applications, such as flowing garment, but stumbles at complex ones. Because they suffer from mesh distortion and element entanglement which usually require computationally intensive treatments.

Conversely, meshless methods showed success for large topology change of solids early on using particle representation (Pauly et al. 2005), such as Mass Spring Method with spring forces governed by Hooke's law (Liu et al. 2013), Peridynamics (PD) (Silling et al. 2007) which is similar to the Mass Spring System while with particular properties, Smoothed Particle Hydrodynamics (SPH) (Monaghan 1992) which samples the continuum fluid configuration as particles and Discrete Element Method (DEM) (Cundall and Strack 1979) which is closely related to molecular dynamics with a defined length of simulation. These point-based representation offers great simplicity in representing topological changes and discontinuities. Despite significant advances and the advent of more sophisticated techniques, these simple approach remains popular in animation (Levine et al. 2014), real-time muscle deformation (Nedel and Thalmann 1998) and robotic surgery (Kawamura et al. 2008). Some of above methods have been successfully embedded in computer graphics and computational mechanics open source code, such as Peridigm (Littlewood 2015) and Taichi (Hu 2020). In the following content, we will give a brief introduction of the Mass Spring Method and the FEM

to help with understanding the PD (further details in Chapter 4) and the mathematical framework of the MPM (in Chapter 3).

2.2.1 Mass Spring System

Mass Spring Method provides a simple practical method for modelling a wide variety of objects, including garment, hair and rigid bodies. In this method, continuum material is discretized as a collection of N points with positions x_i , for $i = 1, 2, ..., N$, interconnected by a set of linear springs with stiffness c_k and natural lengths l_k, for $k = 1, 2, ..., M$, where M is the number of entire springs. One simple expression of total potential energy $\phi(x_i)$ in Mass Spring System at any time is:

$$\phi(x_i) = \sum_k \frac{1}{2} c_k (|x_i - x_j| - l_k)^2 \tag{2.1}$$

where $|x_i - x_j|$ is the length of deformed spring k that interconnects point i situated at x_i and point j situated at x_j. Given the potential energy $\phi(x_i)$, internal forces can be derived as $f_i = -\frac{\partial \phi(x_i)}{\partial x_i}$. The Mass Spring System simplifies the energy calculation based on bonds while can not easily adapted with constitutive models in continuum solids theory.

2.2.2 Finite Element Method

The MPM uses the same weak formulation as in the FEM while these two methods choose different way to describe the the material domain Ω_0. Specifically, FEM subdivides simulation objects into smaller, simpler parts (finite elements) to solve the partial-differential equation in Eq. 2.2. The MPM describes Ω_0 using a set of material points. In this section we briefly introduce a classical FEM (O'brien and Hodgins 1999, Hughes 2012, Chitalu et al. 2020) and focus on how the FEM discretizes Eq. 2.2.

$$\nabla \cdot \sigma + b(x, t) = \rho a \tag{2.2}$$

where ρ is mass density, b is the body force, σ is the Cauchy stress and a is acceleration. ∇ denotes the divergence operator. The Cauchy stress σ is a

function of displacement fields $u(x), x \subset \Omega_t$. In the FEM, the weak form of above governing equation involves multiplying the differential equation by a test function, integrating by parts, and applying boundary conditions. Both the unknown variable and the test function are approximated by functions in a finite-dimensional function space as linear combinations of some basis shape functions. The function $u(x)$ is then obtained from the vector using relevant interpolation functions known as shape functions. The choice of shape functions depends on the discretization of Ω_0 into elements, and results in a system of algebraic equations through the weak form. We refer readers to Hughes (2012) for detailed weak formulation integration.

Assuming each element is a linear tetrahedron e specified by 4 nodes (vertices) and the displacements at its nodes are u_e, the interpolated displacement at $x \in e$ is

$$u(x) = \sum_{i=1}^{4} N_i(x) u_e^i \tag{2.3}$$

where N_i is the shape function of node i which has displacement u_e^i. Strain and stress are constant in e:

$$\epsilon_e(x) = B_e u_e; \quad \sigma_e(x) = D_e \epsilon_e(x) \tag{2.4}$$

where B_e is the discretized gradient matrix which contains the partial derivatives of the shape functions and D_e is the elasticity matrix which encodes material properties. With the stress tensor, updating FEM elements comes naturally.

For deformed objects with severe mesh distortion, the FEM needs periodical remeshing steps and remapping of state variables, making it infeasible to model arbitrarily large material deformations. Some researchers introduced extra treatments to alleviate above challenges, such as remeshing algorithms (Molinari 2002) and element enrichment (Koschier et al. 2017), as shown in Fig. 2.3.

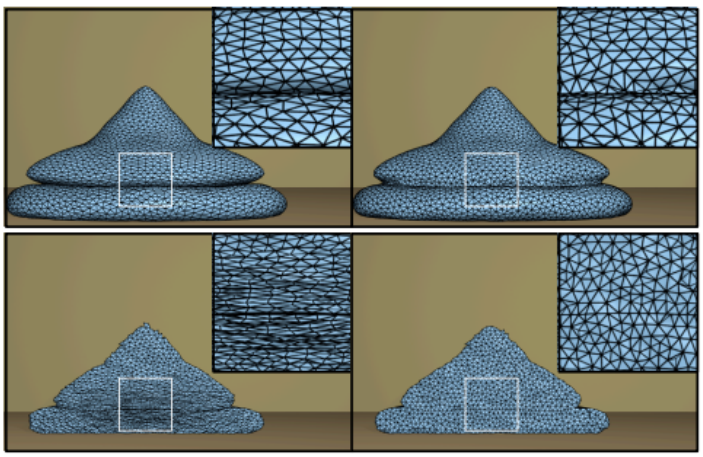

Figure 2.3: FEM mesh-based structures and post-remeshing step (Bargteil et al. 2007).

2.3 Hybrid Eulerian-Lagrangian View

Existing approaches generally excel at some phenomena but would stumble (if not fail) at others. The primary strength of the Eulerian grid-based methods is the simplicity of the discretization and solution of intrinsic collision detection. Unfortunately, grid-based methods have difficulties with the advection part of the equations (Zhu and Bridson 2005). Lagrangian particle methods are considered suitable for modelling topological changes, however, the inherent loss of connectivity information would cause undesirable numerical fracture (Zhu et al. 2017) and surface reconstruction causes problems. Moreover, Lagrangian meshless methods suffer from extra treatments in handling interactions, especially for numerous granular materials. Grids and particles have complementary strengths and weaknesses. Many researchers in graphics have experimented with hybrid Eulerian grids and Lagrangian particle methods to obtain both robust computation and ease implementation. They use particles for basic geometry representation and for advection, and auxiliary grids to compute all the spatial interactions (i.e. boundary conditions, incompressibility, and friction forces). Harlow (1962) proposed the Particle-In-Cell (PIC) which updates pressure and viscosity on a Eulerian grid while advection is completed with Lagrangian particles for

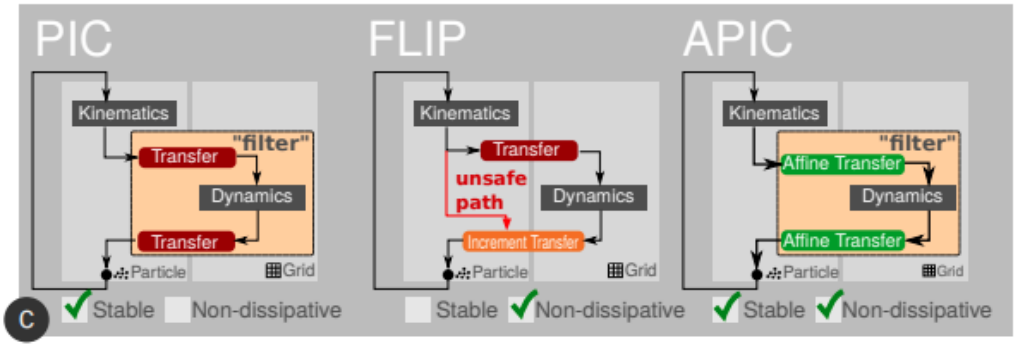

Figure 2.4: The basic dataflow and features of several simulation techniques: PIC, FLIP, APIC. The APIC outweights the PIC and the FLIP both in stability and non-dissipation (Jiang et al. 2015).

fluid simulation. This method has been introduced to computer graphics by Zhu and Bridson (2005) for simulating sand as an incompressible fluid technique. The PIC suffered from excessive numerical dissipation, which was cured later by the Fluid-Implicit-Particle (FLIP) method (Brackbill and Ruppel 1986). Jiang et al. (2015) have observed that the dissipation in the original PIC results from a loss of information when transferring between grid and particle. They augmented each particle with a locally affine, rather than locally constant, to prevent loss of information. The PIC is stable but removes some details during transfer. Then the FLIP was designed to cure this issue but is more noisy and at times, unstable. The APIC only stably removes the dissipation of PIC, but also allows for exact conservation of angular momentum across the transfers between particles and grid. We depict the basic dataflow and main differences between the PIC, FLIP and APIC in Fig. 2.4. Zhu and Yang (2010) animated sand surface flow layer using a standard Discrete Element Method (DEM) and represented the motion of immobile particles beneath the surface by 2D height field columns which is similar to a MAC grid. Sulsky et al. (1995) extended the compressible FLIP to a Eulerian-Lagrangian elastoplastic modelling method, leading to the Material Point Method (MPM) (Sulsky et al. 1995), which has been used to model solids at the level of individual grains amongst other things. The

hybrid methods avoid the computation of each physical granular object and are appropriate scalable.

In the following section, we introduce a general framework of the PIC and explain how basic data flows between particles and the grid.

2.3.1 Particle In Cell and FLIP

PIC is an early approach to simulate compressible flow that handled advection with particles. At each time step, the fluid variables at a grid point are initialised as a weighted average of nearby particle values, and then update on the grid with the non-advection part of the governing equations. The new particle values are interpolated from the updated grid values, and finally the particles move according to the grid velocity field.

The major problem with the PIC is the excessive numerical diffusion caused by repeatedly averaging and interpolating the fluid variables. Later, the FLIP was proposed to alleviate this problem. The crucial change was to make the particles the fundamental representation of the fluid, and use the auxiliary grid simply to advect particles according to the change computed on the grid. The main difference has been shown in Fig. 2.5.

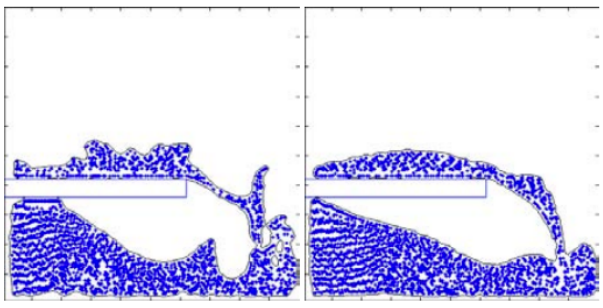

Figure 2.5: Fluid simulation by FLIP (left) and PIC (right). FLIP preserves small-scale velocities which are smoothed away by PIC (Zhu and Bridson 2005).

The PIC routine stores mass m_p, position x_p^n, and velocity v_p^n at time n. m_p does not change with time to ensure mass conservation during whole

simulation. We outline the transfer scheme of the PIC/FLIP and highlight the augmentation from FLIP in Algorithm 1.

Algorithm 1 PIC/FLIP

1: Initialise particle x_p^n and v_p^n
2: **for** each time step **do**
3: **for** grid node i=1,2,...,N **do**
4: Compute a weighted average of the nearby particle velocities on the grid
5: Apply forces to the velocities in a grid based update by solving the governing equations
6: For PIC: Interpolate the grid velocity back to the particles
7: For FLIP: Interpolate the grid velocity increment back to each particle.
8: **end for**
9: Restrict v_p^n under CFL condition and with boundaries constraints
10: Update particle position
11: **end for**

2.4 Fractures

For over thirty years, graphics research has grappled with the unique difficulties associated with simulating material fracture. With Terzopoulos and Fleischer's seminal work on the FEM to model cloth tearing and plastic material (Terzopoulos and Fleischer 1988), researchers started to explore more possibilities of the FEM to simulate notoriously difficult fracture problems. The FEM directly approximates the equations of continuum mechanics, and later has been adapted to achieve fracture through deforming and cutting element meshes individually under different fracture criteria. Use cases include O'Brien's augmented FEM for brittle phenomenon (O'brien and Hodgins 1999), ductile fracture by analysing stress tensors (O'brien et al. 2002), Müller's real-time linear elastic FEM model utilising a warped stiffness approach (Müller and Gross 2004) and Bao's fully rigid fracture model with an endowed tetrahedron mesh to provide stress maps (Bao et al. 2007).

These FEMs, however, face difficulties in handling geometric changes and ill-conditioned basis functions arising from the topological discontinuities which are usually caused by plasticity or crack propagation. Numerous approaches have been explored to address these problems. One solution is to remesh elements near topological discontinuities or the predicted crack tips with novel shape functions, leading to Extended-FEM (XFEM). Fig. 2.6 shows the procedures of how XFEM handles cracks on geometric level. Remeshing methods include the dynamic local meshing algorithm (Wicke et al. 2010), large viscoplasticity remesh strategy (Bargteil et al. 2007), discrete gradient descent flow method for refining and coarsening crack surfaces (Chen et al. 2014) and decoupling fracture surfaces from a deformation mesh to generate high quality crack performance (Chitalu et al. 2020). Another research direction is to combine the FEM with particle-based dynamics methods which offer great simplicity especially for scenes containing amorphous phenomenon, such as debris clouds caused by impacts and explosions. Zhang et al. (2006) converted failed FEM elements to mass-based particles as part of failure structures, and then used molecular dynamics to model discrete particles.

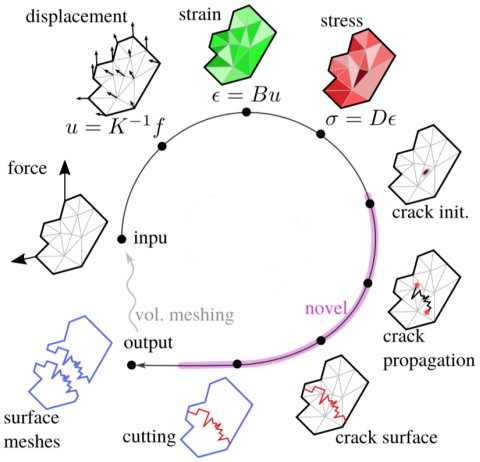

Figure 2.6: Illustrative summary of the different stages of XFEM processing crack propagation (Chitalu et al. 2020).

Compared to FEMs, meshless fracture approaches are powerful in representing most of crack patterns and debris due to the nature of non-mesh

connectivity. Meshless methods avoid the complex remeshing and cutting procedures. Smoothed particle hydrodynamics (SPH) has been used to in a Lagrangian framework for simulating fragmentation of cased explosives (Randles and Libersky 1996, Johnson et al. 1996). Another early work in meshless fracture simulation is Element-Free Galerkin (EFG) (Belytschko et al. 1995, Lu et al. 1995) which requires node-visibility algorithms. Researchers also explored the possibilities of the MPM to allow explicit cracks. A novel MPM with crack techniques (CRAMP) was then produced to track cracks with multiple velocity fields at special nodes near fracturing geometry (Nairn 2003). Recently, PD with an integral equation was introduced by Silling as a promising fracture method (Silling and Lehoucq 2010). In this method, cracks nucleate, grow, branch, merge, and arrest when and where it is energetically favourable for particles to do so. The state-based PD as an augmented version was developed to convert a constitutive model from the conventional theory of solid mechanics directly within the PD (Silling et al. 2007). Despite the promise, the added computational cost originating from looping PD bond structures often undermines its salient features, especially in the case of material non-linearity in three dimensions (Sun and Fish 2019). Also, the PD needs external collision detection mechanism to sidestep the challenges inherit from the particle based nature.

Most recently, some hybrid approaches appear to reduce computational burden through coupling of the intuitive PD with the stable FEM. In these methods, the solution domain is partitioned into PD, FEM, and some transition regions through a morphing strategy to complete communication between whole problem domain and local regions. Force-blending (Seleson et al. 2013), constitutive parameter morphing (Lubineau et al. 2012), and different sub-region coupling by means of interface elements (Liu and Hong 2012) are all examples of the typical morphing strategies. But these aforementioned methods are lack of both superior accuracy and computational efficiency due to extra treatment of blending area and boundary conditions. The superposition-based coupling PD and FEM was brought to the field through using of a rigorous governing equation to derive PD and FEM over one mesh (Sun and Fish 2019, Sun et al. 2019) without comprehensive

parameters- or force-blending treatment. Though the superposition coupling method is still new, it has shown some success within the engineering community at predicting crack initiation and propagation. We explore more possibilities of this superposition-based coupling strategy within computer graphics in this work. The aim is to achieve fractures with high visual fidelity and accurate computation.

Chapter 3

Material Point Method

MPM was designed as a generalisation of the PIC/FLIP solvers to computational solids (Sulsky et al. 1995), and first introduced to computer graphics by Stomakhin et al. (2013). It is a hybrid method, and as such combines a Eulerian grid with Lagrangian particles.

First, a continuous material is discretized into a set of particles. The particles store all information that will be carried on through the simulation such as position, velocity and other necessary properties related to the constitutive model. A grid is used in the background to perform calculations, more specifically to solve the equations of motion. Particles are rasterized to the grid by a weighting function. During this process, some attributes are transferred to the grid nodes. The momentum term updated on the grid is then transferred back to the particles, and finally the particles are advected. Afterwards, the grid is reinitialised to match the deformed configuration at a new simulation step.

MPM has been applied to a broad range of physical phenomena, such as snow (Stomakhin et al. 2013), sand (Klár et al. 2016), foam (Ram et al. 2015), knit (Jiang et al. 2017) and organic isotropic and anisotropic materials (Wolper et al. 2019 2020). Notably, the traditional MPM fails to model sharp separation of material points and cannot represent discontinuous velocities (Hu et al. 2018). Hu et al. (2018) proposed to replace the shape functions by a Galerkin-style Moving Least Squares (MLS) function in the stress divergence term, leading to a different force computation scheme that

does not require for evaluating the gradients of nodal shape functions while capable of providing almost identical visual results. Now the traditional MPM and the MLS-MPM both are popular methods and show great success in computer graphics with different applications.

This chapter will go into greater detail regarding the transfer procedures described above, elucidate the difference between above two MPMs and outline data flow of MPM.

3.1 Governing Equations

The MPM adopts the material space Ω_0 and the deformed space Ω_t with deformation map $x = \phi(X, t)$ in continuum mechanics theory. x and X are world and material coordinates respectively. It relies on the continuum approximation, avoiding the need to model every material point, and adopts the standard conservation equations for mass and momentum that will determine the motion of the material as in Eq. 3.1 and Eq. 3.2 (Stomakhin et al. 2013).

$$\frac{d\rho}{dt} + \rho\nabla \cdot v = 0 \tag{3.1}$$

$$\rho a = \nabla \cdot \sigma + \rho b \tag{3.2}$$

where ρ is density, t is time, v is velocity, a is acceleration, σ is the Cauchy stress, b is the body force. σ in implementation is determined by strain tensor and constitutive models. We will discuss the strain-stress relation in various constitutive models in terms of different materials in Chapter 3.3.

3.2 Method Outline

The basic idea behind the MPM is to use material points to track mass and momentum, and a regular Eulerian grid to help with evaluation of stress-based force and intrinsic collision detection. Similar to the PIC, particle p is initialised with position x_p, velocity v_p, mass m_p, deformation gradient F_p and

affine matrix C_p (only in the MLS-MPM). Grid node i carries grid velocity v_i, grid mass m_i and internal forces f_i. We use subscripts p, q for particle quantities and i, j, k for grid node quantities. The Lagrangian treatment of these quantities simplifies the discretization of the $\frac{d\rho}{dt}$ term in Eq. 3.1. The interpolation functions over the grid are used to discretize the $\nabla \cdot \sigma$ term using a weak form.

We present the standard MPM transfer pipeline (Stomakhin et al. 2013) and some augmentations in MLS-MPM (Hu et al. 2018) to elucidate the basic update procedures in one time step. The illustration of the interplay between the grid and particles is shown in Fig. 3.1. Superscript $n, n+1$ are for quantities at discrete time t^n and t^{n+1} where $\Delta t = t^{n+1} - t^n$.

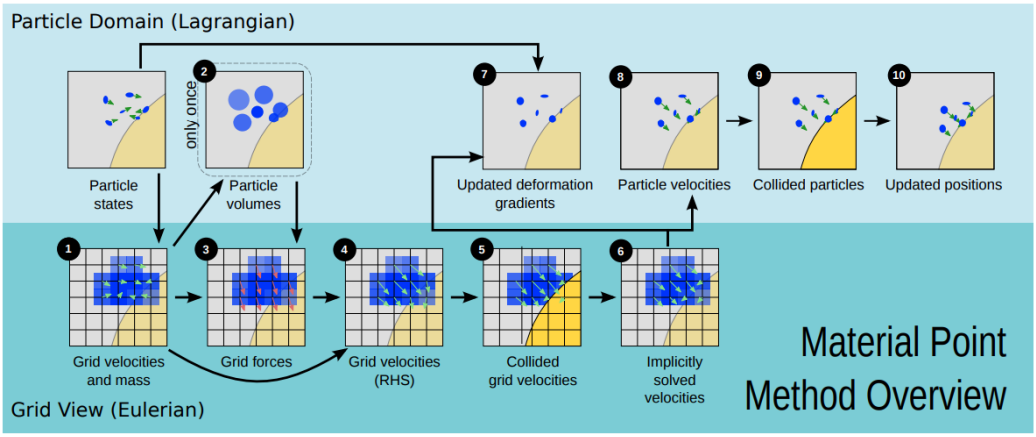

Figure 3.1: An overview of the MPM. The top and the bottom rows are steps that operate on particles while the middle depicts grid-based operations (Stomakhin et al. 2013).

1. **Particles to grid.** Mass is transferred to grid nodes using $m_i = \sum_p \omega_{ip} m_p$. Momentum is transferred to the grid using the traditional MPM: $mv_i^n = \sum_p \omega_{ip} m_p v_p^n$, or MLS-MPM (Hu et al. 2018): $mv_i^n = \sum_p \omega_{ip} m_p (v_p^n + C_p^n (x_i^n - x_p^n)$, where C_p is the affine matrix of particle x_p. w_{ip} is dyadic products of one-dimensional quadratic B-spline interpolation weight between particle p and node i, where $dx = \frac{1}{\Delta x}|x_p - x_i|$

(Jiang 2015).

$$N(dx) = \begin{cases} \frac{1}{2}(\frac{3}{2} - dx)^2 & 0 \leq dx \leq \frac{1}{2} \\ \frac{3}{4} - (dx-1)^2 & \frac{1}{2} \leq dx \leq \frac{1}{2} \\ \frac{1}{2}(dx - \frac{1}{2})^2 & 1 \leq dx \leq \frac{3}{2} \\ 0 & otherwise \end{cases} \tag{3.3}$$

2. **Grid update.** The internal force f_i^n of grid node is computed by the traditional MPM as: $f_i^n = -\sum_p V_p^n \sigma_p^n \nabla \omega_{ip}^n$ and the MLS-MPM : $f_i^n = -\sum_p \omega_{ip} V_p^0 M_p^{-1} \frac{\partial \psi}{\partial F}(F_p^n)(F_p^n)^T(x_i^n - x_p^n)$, where $M_p^{-1} = 4/\Delta x^2$ for a quadratic particle-grid kernel and $\psi(F_p^n)$ is potential energy density function. Δx is the length of one cubic grid cell. Update grid velocity v_i use explicit time integration or implicit time integration method (see in Chapter 3.4).

3. **Grid to particles.** Update particle velocities using $v_p^{n+1} = (1 - \alpha)v_{PICp}^{n+1} + \alpha v_{FLIPp}^{n+1}$. The PIC part is $v_{PICp}^{n+1} = \sum_i \omega_{ip} v_i^{n+1}$ and the FLIP part is $v_{FLIPp}^{n+1} = v_p^n + \sum_i \omega_{ip}(v_i^{n+1} - v_i^n)$. $\alpha = 0.95$ is adopted for snow simulation in Stomakhin et al. (2013). The position is updated: $x_p^{n+1} = x_p^n + \Delta t v_p^{n+1}$.

4. **Particles strain update.** The traditional MPM updates the particle deformation gradient F_p^{n+1} as: $F_p^{n+1} = (I + \Delta t \nabla v_p^{n+1})F_p^n$. The MLS-MPM first exports the affine matrix as: $C_p^{n+1} = \sum_i \omega_{ip} M_p^{-1} v_i^{n+1}(x_i^n - x_p^n)^T$, and then updates F_p^{n+1}: $F_p^{n+1} = (I + \Delta t C_p^{n+1})F_p^n$.

3.3 Constitutive Models

Constitutive models encompass ways of describing material responses to different mechanical and/or thermal loading conditions. Specifically, they use strain measurement and yields stress to describe the material response. Plasticity is also accounted in constitutive models. Stomakhin et al. (2013) introduced a multiplicative plasticity decomposition theory, in which the particle deformation gradient F_p can be decomposed into elastic part F_{Ep} and plastic part F_{Pp} by $F_p = F_{Ep}F_{Pp}$. This theory treats plasticity as a post-process to

restrict the deformation within a critical threshold, which has been proved plausible for modelling snow and silly rubber (Fang et al. 2019). It has been adopted by many MPM studies to handle general deformable plasticity. In this section, we first explain two elastic constitutive models, and then introduce the multiplicative plasticity decomposition theory in Stomakhin et al. (2013).

3.3.1 Fixed Corotated Model

An ideally elastic material means that the constitutive relationship is totally dependent on the current state of deformation, in which constitutive models can be derived from an energy density function ψ.

Given a potential energy density function ψ and a discretization of a continuum, one can usually write out the total potential energy function $\phi = \Sigma_p V_p^0 \psi(F_p)$ of the material and use it to derive forces. Continuum mechanics prefer to define energy density functions from deformation fields. In the MPM, ψ is determined from the particle deformation gradient F_p. The initial statement would result in:

$$\sigma_p = \frac{1}{J_p} \frac{\partial \psi(F_p)}{\partial F_p} {F_p}^T \tag{3.4}$$

where $J_p = det(F_p)$. The Cauchy stress σ_p captures the strain-stress relationship, and supplements the conservation equations Eq. 3.1 and Eq. 3.2 with the additional information needed to obtain the acceleration. There are several alternative measures of stress tensor defined to represent different material configurations. As shown in Table. 3.1, they are equivalent to each other with proper transformation. In our work, we adopt the Cauchy stress which is the most commonly used measure of stress. In the Table. 3.1, we adopt F as the deformation gradient from the continuum solid theory. With all definitions in hand, the last exploration of constitutive models is the energy density function $\psi(F_p)$.

Stomakhin et al. (2012) has developed a Fixed Corotated model which is a robust energy density function and keeps valid under inverted configurations.

Table 3.1: Representation and measurements of different stress tensors.

Stress tensor	Symbol	Area normal	Force	Relations
Cauchy	σ	spatial	spatial	
First Piola-Kirchoff	P	material	spatial	$P = J\sigma F^{-T}$
Second Piola-Kirchoff	S	material	material	$S = JF^{-1}\sigma F^{-T}$

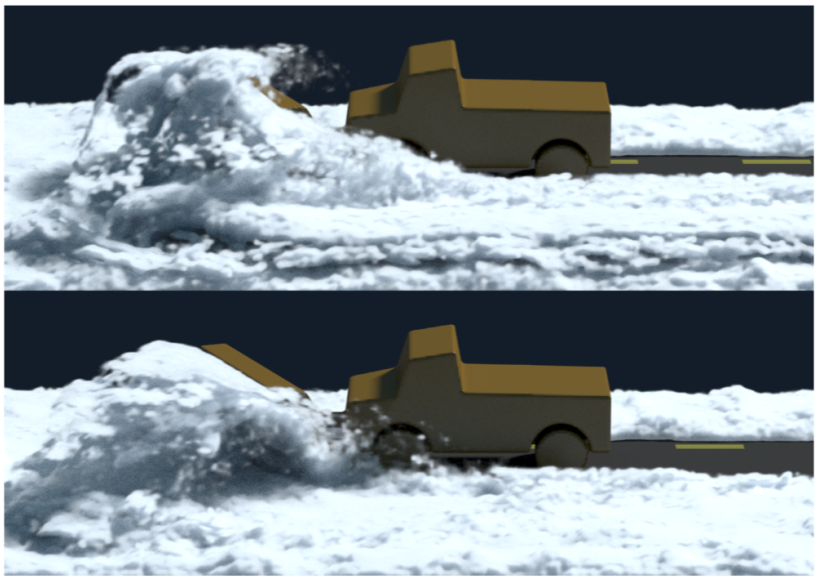

Figure 3.2: Animating snowplow by the Fixed Corotated model (Stomakhin et al. 2013).

This model has been successfully simulating practical snow dynamics, as shown in Fig. 3.2. The Fixed Corotated energy density is written as:

$$\psi_{Corotated} = \mu|F - R|_F^2 + \frac{\lambda}{2}(J-1)^2 \tag{3.5}$$

where R comes from the polar decomposition of $F = RS$, and its purpose is to remove the contribution of rigid body rotations from the potential energy. $J = det(F)$. *Lamé* parameters μ and λ are material properties that can be varied to get different material characteristics. They are commonly expressed in Young's modulus E and Poisson's ratio ν (Salençon 2012), which relate

to μ and λ by

$$\mu = \frac{E}{2(1+\nu)}, \lambda = \frac{\nu E}{(1+\nu)(1-2\nu)} \tag{3.6}$$

In Eq. 3.5, the Frobenius norm $|\cdot|_F^2$ is used. For a matrix A with entries a_{ij}, it is defined as below:

$$|A|_F^2 = \sqrt{\sum_i \sum_j |a_{ij}|^2} \tag{3.7}$$

The first item $\mu|F-R|_F^2$ on the right side of Eq. 3.5 represents the deviatoric portion of the energy density function relating to shape distortion that does not cause change in volume. The second term $\frac{\lambda}{2}(J-1)^2$ depicts the volume dilatation of the material.

3.3.2 Neo-Hookean Model

The Neo-Hookean model is proposed as a suitable model for describing large deformation of materials that has no intrinsic directional dependence (Smith et al. 2018), such as biological tissues (i.e. muscle and belly of a virtual human as shown in Fig. 3.3). The Neo-Hookean model has several variants. We implement a common Neo-Hookean energy density formulation in Bonet and Wood (1997):

$$\psi_{Neo} = \frac{\mu}{2}(I_C - 3) - \mu \log J + \frac{\lambda}{2}(\log J)^2 \tag{3.8}$$

where $I_C = tr(F_p^T F_p)$ is the first right Cauchy-Green invariant. Based on the hypothesis (Xu et al. 2015), we decompose the right side of above equation into: $\psi_{Neo,length} = \frac{\mu}{2}(I_C - 3)$ and $\psi_{Neo,volume} = -\mu \log J + \frac{\lambda}{2}(\log J)^2$. This decomposition can converge to the analyses of the deviatoric-volumetric energy decomposition in the last section.

When $\psi_{Neo,length}$ is unconstrained, this energy achieves its minimum when a deformation unit (element or particle) has collapsed to zero volume, i.e., when $I_C = 0 (\psi_{Neo,length} = -3)$. With the hard constraint $J_p = 1$ is imposed, the energy term is instead minimised at the volume preserving configuration that is the closest to the stretch space origin. $\psi_{Neo,length}$ is well behaved under

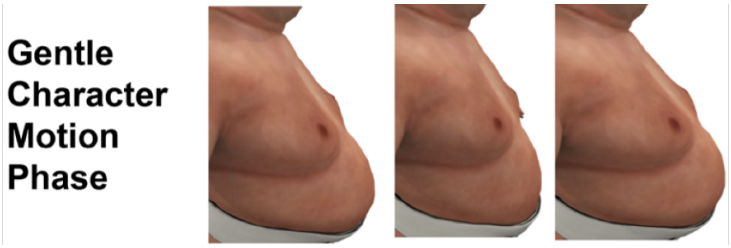

Figure 3.3: Virtual human deformation simulated by a Neo-Hookean model (Xu et al. 2015).

inversion. The energy relative to a zero-volume configuration is always well defined irrespective of a unit's current state.

$\psi_{Neo,volume}$ is volume-preserving penalty term. However, it presents numerical problems by growing without bound under compression, i.e., as $J \to 0$ and becoming undefined at $J = 0$. This can never happen in the real world. However, numerical simulations generally do not prevent such nonphysical deformations (Jiang 2015). This problematic term has been modified by the Fixed Corotated elastic model as $\frac{\lambda}{2}(J-1)^2$.

3.3.3 Plasticity

Some researchers used a specially designed finite-strain multiplicative plasticity law employing the **Drucker-Prager plasticity model** (Drucker and Prager 1952) to model snow (Stomakhin et al. 2013), in which only elastic deformation gradient F_{Ep} contributes to the stress. Before the plasticity decomposition, one factorisation of F_p is required: singular value decomposition (SVD), which is written by:

$$F_p = U_p \Sigma_p V_p^T, \Sigma_p = \begin{bmatrix} \sigma_0 & 0 & 0 \\ 0 & \sigma_1 & 0 \\ 0 & 0 & \sigma_2 \end{bmatrix} \tag{3.9}$$

where U_p and V_p^T are complex unitary matrix encoding rotation information. The diagonal entries Σ_p are known as the singular values of F_p. The plasticity is processed at the **Particles strain update** step. The updation of plasticity can be divided into three steps:

1) We assume that initially all changes get attributed to the elastic part of the deformation gradient as: $\hat{F}_{Ep}^{n+1} = (I + \Delta t \nabla v_p^{n+1}) F_{Ep}^n$ and $\hat{F_{Pp}^{n+1}} = F_{Pp}^n$.

2) Compute the SVD of elastic deformation gradient: $\hat{F}_{Ep}^{n+1} = U_p \hat{\Sigma}_p V_p^T$ and clamp the singular values to the permitted range $\Sigma_p = clamp(\hat{\Sigma}_p, [1 - \theta_c, 1 + \theta_s])$. θ_c and θ_s are critical compression and critical stretching parameters respectively.

3) Reconstruct updated elastic and plastic components of the deformation gradient: $F_{Ep}^{n+1} = U_p \Sigma_p V_p^T$ and $F_{Pp}^{n+1} = V_p \Sigma_p^{-1} U_p^T F_p^{n+1}$.

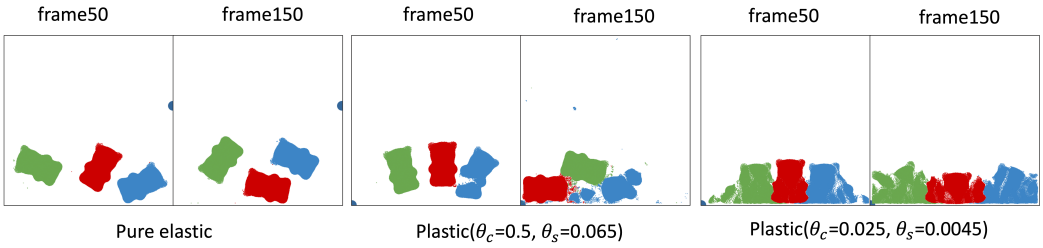

Figure 3.4: Animating smashed bears with elastic materials and elastoplastic materials. The middle and right pictures demonstrate varying settings of critical parameters.

Note that θ_c and θ_s represent the abilities of resisting compression and stretching respectively. When all plasticity steps vanish, the simulation derives ideally elastic deformation. Fig. 3.4 shows varying θ_c, θ_s settings and demonstrates their impact on the simulation. The Drucker-Prager plasticity model has been commonly used in versatile applications. We also have adopted this plastic model in our later studies.

3.4 Time Integration

At the step **Grid update**, we think of the elastoplastic response as defined from the material positions of the Eulerian grid nodes $\hat{x_i} = x_i + \Delta t v_i$, where x_i is the position of grid node i, then $\hat{x_i}$ would be the deformed location of that grid node given the current velocity v_i of the node. However, in the MPM, the grid nodes do not actually being transformed by their velocity because only the updated nodal velocity needs be transferred back to particles. There are

two ways to update the grid velocity after obtaining internal force f_i: explicit time integration and implicit time integration.

3.4.1 Explicit Method

Explicit time integration updates the nodal velocity v_i^{n+1} from all states at time n:

$$v_i^{n+1} = v_i^n + \Delta t f_i^n \tag{3.10}$$

The explicit integration solves the velocity update straightforwardly, however, it requires impractically small timestep to keep the numerical error in result bounded. For example, in order to simulate thin shells, the timestep should be sufficient small in order to detect subtle interactions. In particular, the explicit operates with a time step restricted by the usual Courant-Friedrichs-Lewy (CFL) condition that no particles are allowed to travel more than one portion of grid cell in each time step (De Moura and Kubrusly 2013) to avoid incorrect results.

3.4.2 Implicit Method

The implicit formulation follows the Eq. 3.10 with using f_i^{n+1} instead. Evolving grid velocities v_i^{n+1} implicitly in time remains stable for large time steps (Sulsky and Kaul 2004) while requires an extra computation and is hard to implement compared to the explicit methods. Stomakhin et al. (2013), Hu et al. (2018) adopted a matrix-free Krylov implicit method which avoids reconstructing a sparse matrix in each time step and minimises the computational costs. The momentum equation in Eq. 3.11 is an implicit equation for the velocity, as presented. Different β decides the method of handling time integration: $\beta = 0$ for explicit integration, $\beta = 0.5$ for trapezoidal integration, and $\beta = 1$ for backward Euler implicit integration. We refer to Stomakhin et al. (2013) for further details.

$$v_i^{n+1} = v_i^n + \frac{\Delta t}{m_i}((1-\beta)f_i^n + \beta f_i^{n+1}) \approx v_i^n + \frac{\Delta t}{m_i}(\beta \Delta t \Sigma_j \frac{\partial f_i^n}{\partial \hat{x_j}} v_i^{n+1}) \tag{3.11}$$

Before evolving Eq. 3.11 with implicit description, we firstly introduce the Hessian of the potential with respect to $\hat{x}$. The action of this Hessian on an arbitrary increment δu can be expressed as:

$$\delta f_i = -\sum_j \frac{\partial^2 \phi}{\partial \hat{x}_i \partial \hat{x}_j}(\hat{x}) \delta u_j = -\sum_p V_p^0 A_p \left(F_{Ep}^n\right)^T \nabla w_{ip}^n \tag{3.12}$$

where A_p contains the second derivatives of potential energy density function $\psi(F_{Ep})$ with respect to the (elastic) deformation gradient F_{Ep}.

$$A_p = \frac{\partial^2 \psi}{\partial F_{Ep} \partial F_{Ep}} \left(F_{Ep}, F_{P_p}^n\right) : \left(\sum_j \delta u_j \left(\nabla w_{jp}^n\right)^T F_{Ep}^n\right) \tag{3.13}$$

In practice, it corresponds to a grid-to-particle gather step (for computing A_p) and a particle-to-grid scatter step (for accumulating δf_i).

Now we have all descriptions needed to evolve Eq. 3.11. With the notation $\frac{\partial^2 \phi^n}{\partial \hat{x}_i \partial \hat{x}_j} = -\frac{\partial f_i^n}{\partial \hat{x}_j}$, Eq. 3.11 leads to a (mass) symmetric system and becomes:

$$\begin{aligned} v_i^\star &= \sum_j \left(I\delta_{ij} + \beta \Delta t^2 m_i^{-1} \frac{\partial^2 \phi^n}{\partial \hat{x}_i \partial \hat{x}_j}\right) v_j^{n+1} \\ &= v_i^{n+1} + \frac{\beta \Delta t}{m_i} \sum_j \frac{\partial^2 \phi^n}{\partial \hat{x}_i \partial \hat{x}_j} \left(\Delta t v_j^{n+1}\right) \\ &= v_i^{n+1} - \frac{\beta \Delta t}{m_i} \delta f_i \end{aligned} \tag{3.14}$$

where $v_i^\star = v_i^n + \Delta t m_i^{-1} f_i^n$ is an explicit description of grid node velocity. With Eq. 3.12 and differentiation of constitutive models, we are able to solve the updated nodal velocities v_i^{n+1} through Eq. 3.14. Detailed differentiation could be found in the accompanying technical report (Stomakhin et al. 2013).

3.5 Simulation Result

We first implemented the traditional MPM with various plasticity settings, as shown in Fig. 3.4. The pure elastic bears can restore original shapes after being smashed on the ground while the plastic bears end up with fractures. The larger critical compression and stretching parameters leads to stiffer

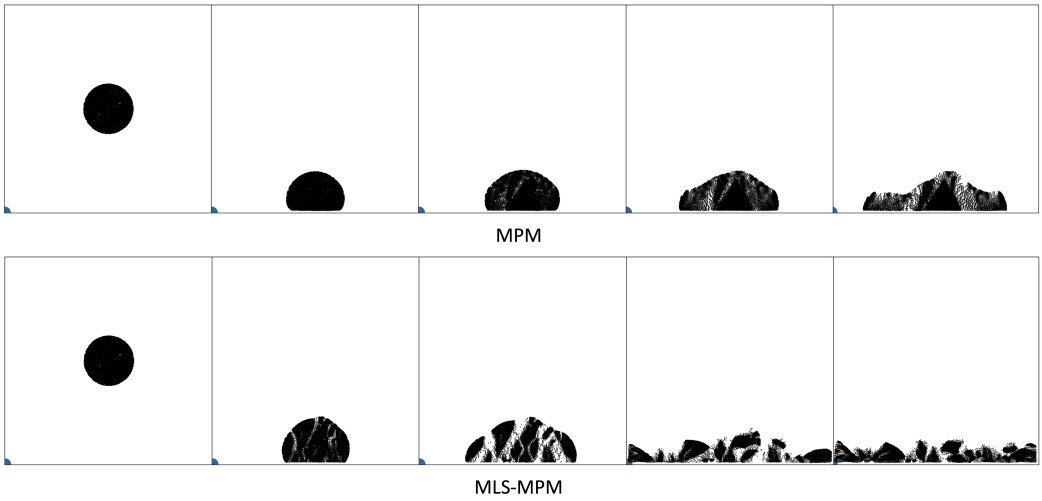

Figure 3.5: Comparison demonstration of the MPM and the MLS-MPM.

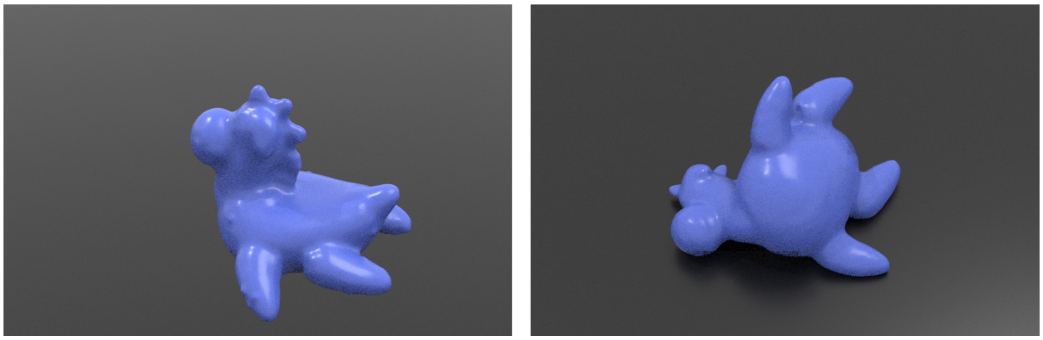

Figure 3.6: Falling elastic toy.

broken effects. Then we compare the traditional MPM with the MLS-MPM using a 2D demo in Fig. 3.5 with same settings ($\mu = 6 \times 10^3$, $\lambda = 4 \times 10^3$). Both methods are capable of handling fractures with plasticity model but the traditional MPM introduces a lot of visually unappealing damaged debris around crack surfaces. Finally, we demonstrate the efficacy of the MLS-MPM with two 3D animation in Fig. 3.6 and Fig. 3.7. Note that $\mu = 3 \times 10^3$, $\lambda = 1 \times 10^3$ for the elastic toy and $\mu = 6 \times 10^3$, $\lambda = 4 \times 10^3$, $\theta_c = 0.025$, $\theta_s = 0.0075$ for the castle.

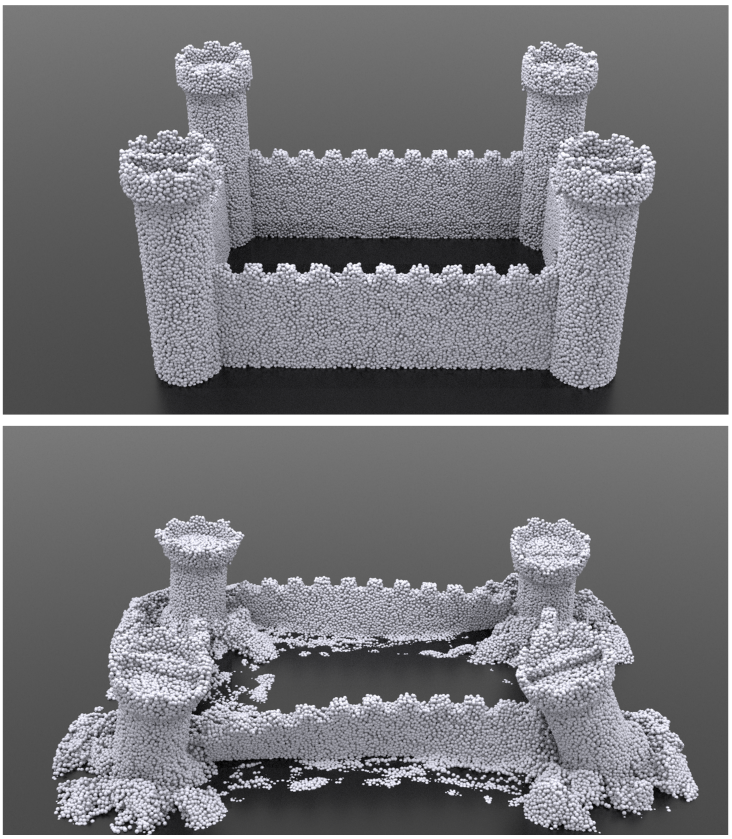

Figure 3.7: Collapsed castle made of granular materials.

3.6 Summary

This chapter goes through detailed transfer procedures of MPM with highlighting differences between the traditional MPM and the MLS-MPM. We introduce two hyperelastic constitutive models and the multiplicative plasticity model with accompanying experiments. We also give a brief description of explicit integration and implicit time integration for updating grid nodes' velocity. We present several 2D and 3D examples to demonstrate the features of the MPM in handling versatile materials.

Chapter 4

Peridynamics

Peridynamics (PD) is an nonlocal extension of continuum mechanics (Silling et al. 2007, Silling and Lehoucq 2010, Silling et al. 2017). PD discretizes a continuum object into unstructured particles. One PD particle is influenced directly by other particles located in its vicinity using virtual bonds. Deriving the bonds leads to an accurate and easy implementation in simulating manifold crack dynamics. In contrast to the classical continuum theory, the balance of linear momentum is formulated as an integral equation that remains valid in the presence of material discontinuities such as cracks.

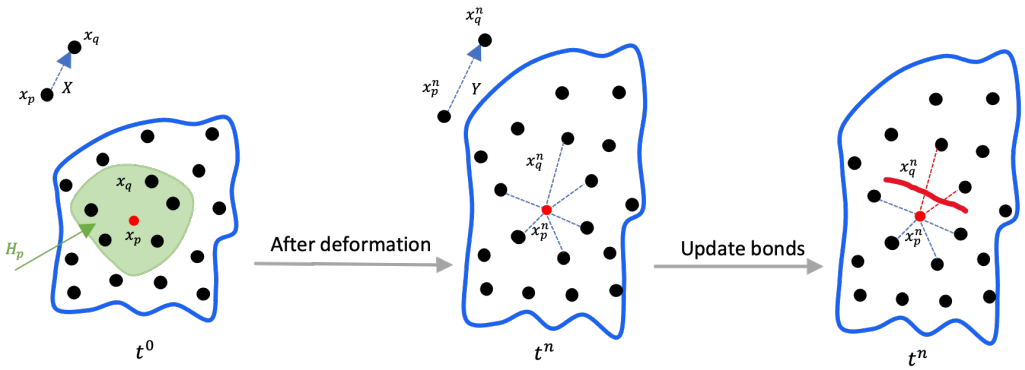

Figure 4.1: An overview of PD. The left depicts the initial configuration and the right side is the deformed configuration at time t^n.

Fig. 4.1 is an illustration of the PD discretization with particles. In PD, a material point x_p interacts directly with all neighbouring material points x_q within a finite distance δ, termed the horizon H_p. Each particle has a

volume term V_p. PD defines that a virtual bond X connects the point x_p to one of its neighbour x_q. To avoid ambiguity, we use Y to represent bond X in deformed configuration at time t, as shown in the middle of Fig. 4.1. Each bond has a status s_{pq} indicating active (connected) or broken (disconnection). This status is encoded in a weighting function. More details will be given in following sections.

PD simulates objects by collecting all interaction forces with neighbourhood particles and creating damages by breaking bonds over a critical range. It seems analogous to other meshless methods based on classical theory, and the difference lies in the choice of particles' vicinity. SPH (Liu and Liu 2010) is defined as a local continuum model in which the state of a particle is influenced by only particles in its immediate vicinity. Conversely, PD only interacts with the neighbouring particles defined at the beginning stage in simulation, thus referred as an nonlocal theory. As the radius of the PD horizon H_p becomes infinitely large, PD theory becomes the continuous version of the molecular dynamics model. As the radius becomes smaller, it becomes the continuum mechanics model.

The discretized PD balance of linear momentum at time t for the point x_p in the body B is given by:

$$\rho a = \sum_{H_p} \left\{ T[x_q, t] \left\langle x_q - x_p \right\rangle - T\left[x_p, t\right] \left\langle x_p - x_q \right\rangle \right\} V_q + b \tag{4.1}$$

where ρ is mass density, a denotes acceleration, b is the body force. $T[x_q, t] \left\langle x_q - x_p \right\rangle$ and $T\left[x_p, t\right] \left\langle x_p - x_q \right\rangle$ present a pair of force densities exerted on the particles x_p and x_q, respectively. The angle brackets representation $\langle \cdot \rangle$ is defined by Silling et al. (2017) as a function inside the horizon H_p, which they called as a state. Notably, the force density may not act along the direction of the bond $x_q - x_p$ or $x_p - x_q$. The formulation of the force density function generally fall into two main categories (Mitchell 2011): bond-based material models and state-based material models. This is decided by if the PD force density function can encode constitutive laws of materials. We will further explain how these two models work in following content.

4.1 Bond-based Peridynamics

Bond-based material models are constitutive laws in which the pairwise force acting between material points x_p and x_q is purely a function of their relative displacements (Silling 2000). The pairwise force always acts along the direction of the bond connecting x_p and x_q. The bond-based PD loops each neighbouring particle and summarises all paired forces. It has the advantages of being computationally inexpensive and robust, making it a good choice for simulations involving pervasive material failure. Levine et al. (2014) proved that the bond-based PD is simple to implement, trivially parallelised, and well-suited to animating brittle fracture, as shown in Fig. 4.2.

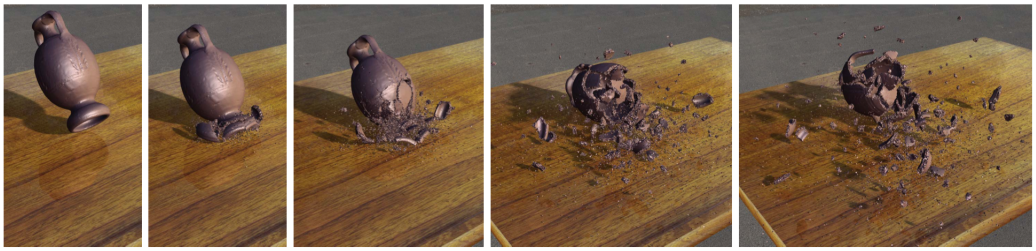

Figure 4.2: Vase undergoing fractures is animated by the bond-based PD (Levine et al. 2014)

The bond-based PD can be treated as spring-mass systems because that the interaction forces between material points are assumed to be equal in magnitude and opposite to each other. The particular differences are: 1) one mass spring nodes interact with adjacent nodes based on a mesh topology while one PD particle interacts with all particles within a given particular distance δ. 2) PD defines a simple strain metric for the bond between x_p and x_q which the spring mass system does not (Levine et al. 2014):

$$\varepsilon_{pq} = \frac{\|Y - X\| - X}{X} = \frac{\|Y - X\|}{X} - 1 \tag{4.2}$$

The bond force is:

$$t_{pq} = -K\varepsilon_{pq}\frac{Y}{\|Y\|}, K = \frac{18K}{\pi l^4} \tag{4.3}$$

where K is the bulk modulus. Here t_{pq} is the bond force acted on the deformed bond Y. When the force state $T[x_p, t]\langle x_q - x_p\rangle$ in Eq. 4.1 is replaced by the bond force t_{pq}, Eq. 4.1 becomes a standard bond-based PD momentum equation.

Notably, the straightforward calculation of bond forces significantly simplifies the computation while introduces several limitations including constraints on material constants, incapable of capturing incompressibility for plastic deformations, etc. Moreover, it can not produce many interesting elastic material behaviours, especially the Poisson effect. In order to overcome these limitations, Silling et al. (2007) developed an advanced version of the bond-based PD named as state-based PD. In the state-based PD, the interaction forces do not need to be equal in magnitude and opposite to each other.

4.2 State-based Peridynamics

A state can be viewed as one status of the local region around one PD particle, which is decided by all connected neighbourhood particles instead of one single bond. For example, the force state $T[x_p, t]\langle x_q - x_p\rangle$ is a function of the deformation of all the material points in H_p and H_q, and possibly other variables as well (Silling et al. 2007). The state-based PD is a more general case of the bond-based PD, permitting a general representation of the continuum mechanics using the stress-strain relation directly, for example, isotropic and anisotropic material models in Fig. 4.3. It adopts several bond concepts: the initial bond X, the deformed bond Y and the relative displacement vector is $U = Y - X$.

Designed for the purpose of adapting classical material constitutive models for use with PD, the state-based PD starts with the evaluation of approximation deformation gradient F_p of particle p:

$$F_p = [\Sigma_{H_p}\omega(X)Y \otimes XV_q] \cdot K_p^{-1} \tag{4.4}$$

where K_p is a symmetric shape tensor:

$$K_p = \Sigma_{H_p}\omega(X)X \otimes XV_q \tag{4.5}$$

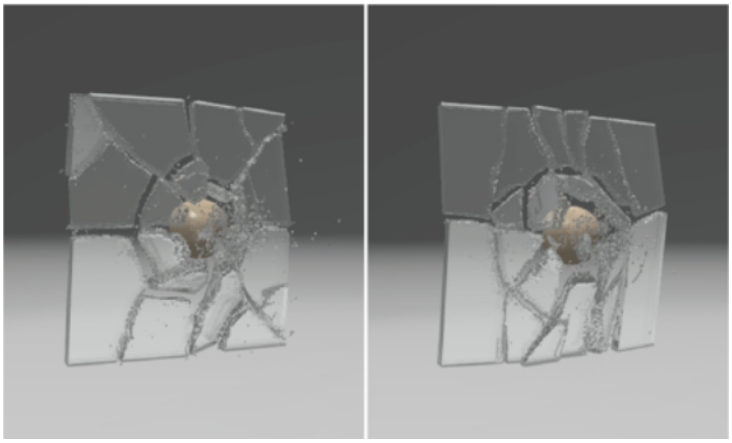

Figure 4.3: Isotropic and anisotropic brittle fractures in glasses animated by a state-base PD solver (Chen et al. 2018)

where $\omega(X)$ is chosen as an influence function for the bond X that is monotonically decreasing with respect to the distance between two particles and affected by bond breakage condition (more details in Chapter 4.3).

Similar to the continuum theories, the state-based PD formulates a strain tensor ε_p:

$$\varepsilon_p = \frac{1}{2}[\Sigma_{H_p}\omega(X)[(U \otimes X + X \otimes U)V_q] \cdot K_p^{-1} \tag{4.6}$$

With the strain tensor ϵ_p and the deformation gradient F_p, we are able to apply most constitutive models and obtain stress tensor, for example, the Fixed Corotated model in Eq. 3.5. Here we adopt the Cauchy stress tensor σ_p for simplicity. Then the force state can be expressed in terms of the σ_p as an intermediate step in integration rather than a formulation of partial derivatives.

$$T[x_p, t^n] < x_q - x_p >= \omega(X)det(F_p) \cdot (F_p)^{-1} \cdot \sigma_p \cdot K_p^{-1} \cdot X \tag{4.7}$$

where $det(F_p)$ is the determination of F_p.

The state-based PD model is able to convert the states to tensorial quantities (e.g. convert deformation gradient state to the force vector). With states, any material that can be modelled within the standard theory of continuum mechanics are able to be implemented through the state-based PD formulation, and retain the advantages of the PD for fractures. Note

that using the state-based PD may results in greater computational expense compared to the bond-based PD.

4.3 Modelling Fracture by Failure Law

PD includes a natural mechanism for modelling fractures through breaking bonds over a given bond failure law. For damaged material, each individual unbroken bond is tested by the bond failure law to update the connection status and won't be accounted at the next simulation step if it is broken. At the end of simulation, micro-scale broken bonds accumulate and form visually visible cracks.

A critical stretch bond damage model (using a failure criterion) is commonly employed as the failure law to eliminate bonds (Silling and Askari 2005, Levine et al. 2014, Sun et al. 2019). The connection status s_{pq} of bond between x_p and x_q can be simply assigned by this failure criterion. The breakage criterion is reached when bond elongation exceeds a prescribed critical value l_{max}, as described in Eq. 4.8.

$$s_{pq} = \begin{cases} 0 & \frac{|Y|-|X|}{|X|} > l_{max} \\ 1 & \frac{|Y|-|X|}{|X|} \leq l_{max} \end{cases} \tag{4.8}$$

Researchers also employed complex criterions with the state-based PD models for rich, artistically controllable fractures. Chen et al. (2018) combined the bond length failure criterion with a damage parameter on particle-level to offer artists brittle fractures. Foster et al. (2011) proposed a failure criterion relating one bond's critical energy density to energy release rate in terms of different materials. Our novel bond breakage criterion designed for different purpose of fracture schemes will be presented in Chapter 5 and Chapter 6, respectively.

4.4 Method Outline

The state-based PD enriches available options of simulation techniques. Our work is built upon the state-based PD for its ability of handling versatile

elastoplastic materials and complex phenomena. The state-based PD solver, executed by each particle x_p , in parallel, carries out four tasks (in Fig. 4.4). The detailed operations are presented in Algorithm 2.

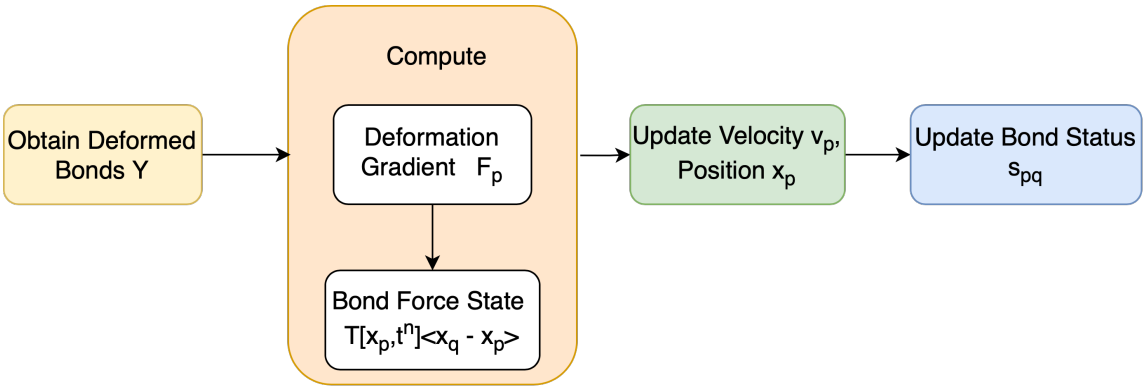

Figure 4.4: An illustration of how the state-based PD derives unstructured particles and produces fractures.

Algorithm 2 PD method

Initialise neighbourhood H_p using k-d tree
Initialise initial bond X and bond connection status s_{pq}
for each PD particle p **do**
    for each active bond (with particle q) **do**
        Obtain Y, U
        Update bond status by criterion in Eq. 4.8
        Compute K_p by Eq. 4.5, F_p by Eq. 4.4 and ϵ_p by Eq. 4.6
    end for
end for
for each PD particle p **do**
    for each active bond (with particle q) **do**
        Compute the force state $T[x_p, t^{n+1}] < x_q - x_p >$ by Eq. 4.7
    end for
    Compute force density by Eq. 4.1
    Update velocity v_p and position x_p
end for

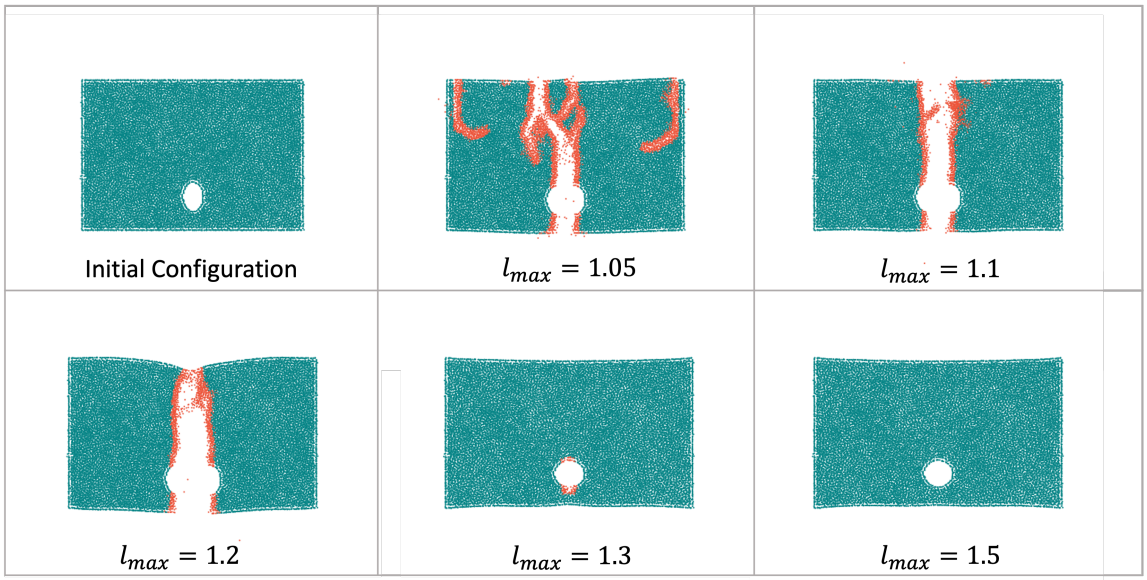

Figure 4.5: Tear the cloth by the state-based PD solver. We demonstrate crack effects with varying critical fracture threshold l_{max} both at frame 176.

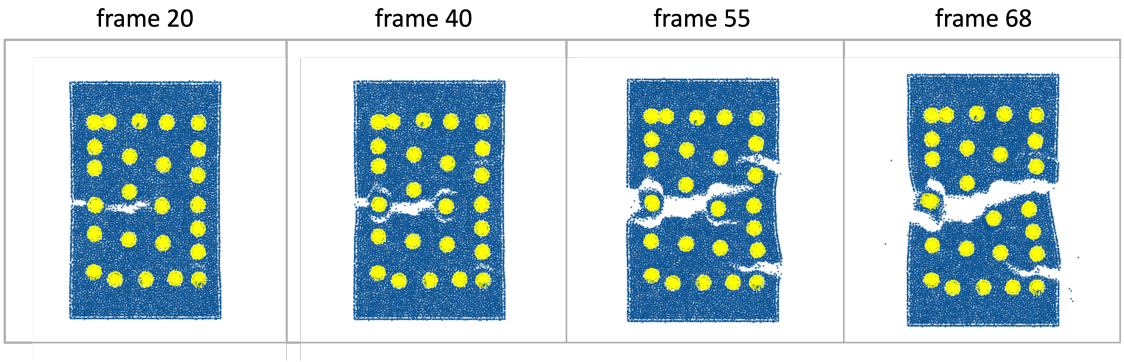

Figure 4.6: A multi-material objects is animated by the state-based PD solver. Different stages of fractures are presented.

4.5 Simulation Results

We demonstrate the advantages of using the state-based PD by applying it to several fracturing examples. First, we tear a piece of cloth in Fig. 4.5. Varying critical fracture threshold l_{max} defined in Chapter 4.3 has been tested. The cloth is easily to be teared with a small threshold. Fig. 4.6 shows that crack propagation with multiple crack fronts in a multi material object is attainable

by the state-based PD solver (yellow cores are stiffer than blue material). We also test 3D models, such as Fig. 4.7. The jello toy is stretched by a fork and fracturing at its joint. The state-based PD is capable of intuitively capturing fractures and cracks without any special treatments as used in the MPM (e.g. using the plasticity model to degrade stress tensor).

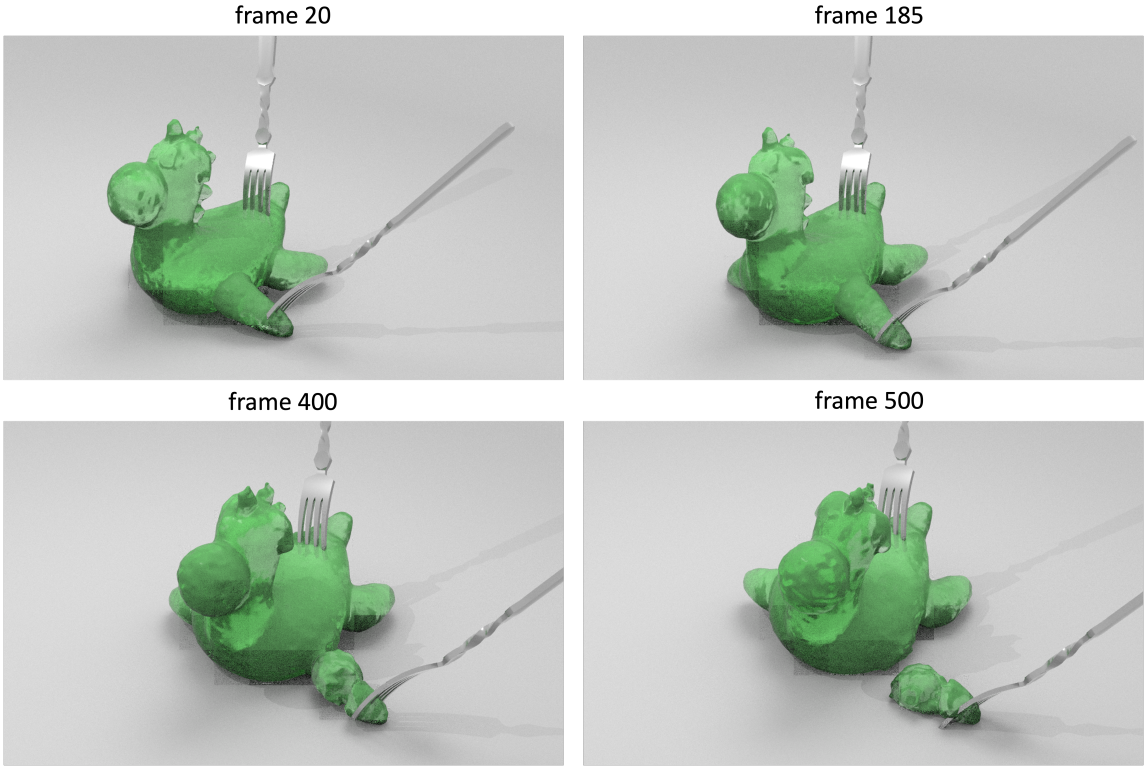

Figure 4.7: Stretch limbs of a jelly toy by the state-based PD.

4.6 Summary

This Chapter goes through the bond-based PD and the state-based PD methods. We also introduce a simple bond failure law that can be applied to most PD models. We demonstrate this powerful techniques with several experiments, showing the fracture initiation and propagation during the simulation.

Chapter 5

Integral-based MPM For Animating Elastoplastic Material

5.1 Background

Physically-based modelling of elastoplastic material has been an active research topic for many years in computer graphics, particularly for its appealing application in visual effects industry. Scenes involving elastoplastic deformation are very common and varied, for example, clothes moving with wind, rubber toys bouncing on the floor, flowing honey, or the broken plastic board. In order to model such realistic behaviours under different circumstances, the robust simulation method needs to be capable of handling complex topological changes and various contact responses, such as collision and cohesion. To find the simulation method that can naturally model elastoplastic material along with complex topological changes is the current focus of the field.

In the MPM, a continuum body is discretized into a set of particles, also referred to as material points that are free to move on top of the background Eulerian Cartesian grid (Moutsanidis et al. 2019). With the hybrid settings, the MPM can naturally process deformed topologies and self-collisions. It also has been proved to be especially suitable for animating materials that undergo large deformations (Jiang et al. 2016, Hu et al. 2018). Despite its

physical realism and geometrical convenience, the traditional MPM solver has several disadvantages: 1) Due to the governing equation based on spatial derivatives of displacements, the results are sensitive to the underlying particle distribution (He et al. 2017). Moreover, it has difficulty in solving singularity along discontinuities. 2) To observe detailed interaction near boundaries, MPM has to maintain a fine resolution grid which brings high computational costs during information transfer back and forth between grid and particles in the whole simulation domain. While researchers have extensively studied refining regions of particular interest by using an adaptive grid (Gao et al. 2017), the ability to simulate detailed discontinuities dynamics, such as crack propagation, is still limited. 3) The MPM uses one background grid thus requires particular treatments to separate fragments. One approach is assigning each fragment with one grid and one velocity field so that the fragment can be separated (Homel and Herbold 2017). When numerous cracks are produced, the computation of transfer procedures on multiple grids can be very expensive.

PD theory defines that a point in a continuum interacts directly with other points separated from it within a finite distance. The advantage of the peridynamics is that its way of treating the discontinuous parts, which may appear in the continuum body as a result of large deformation, is exactly the same as continuous part. The linear PD treats the deformation more generally by processing the small deformation based on the referenced configuration, which provides some possibilities to simulate some particular materials with both solid and fluid properties, such as viscoelastic fluids. However, the PD, with particle-based nature, needs additional efforts to detect and handle physical collisions. Furthermore, PD originates from the solid mechanics, focusing on mechanical experiments. Currently there are only a few mature models and experiments in continuum mechanics being adopted by researchers for animating elastoplastic material with PD (Xu et al. 2018).

This chapter is inspired by the following observations: 1) The MPM and the PD both appear as particle-base methods. 2) For severe deformation, the MPM is robust and accurate to handle topological changes while cannot separate crack surfaces by itself. 3) The PD is able to intuitively animate crack

growth, branch without necessary remeshing strategies. We speculate that adding each MPM material point with the PD bonds (in Fig. 5.1) may provide some possibilities to obtain a stable MPM solver which well suits versatile materials and intuitive fractures modelling. With the augmentation of the

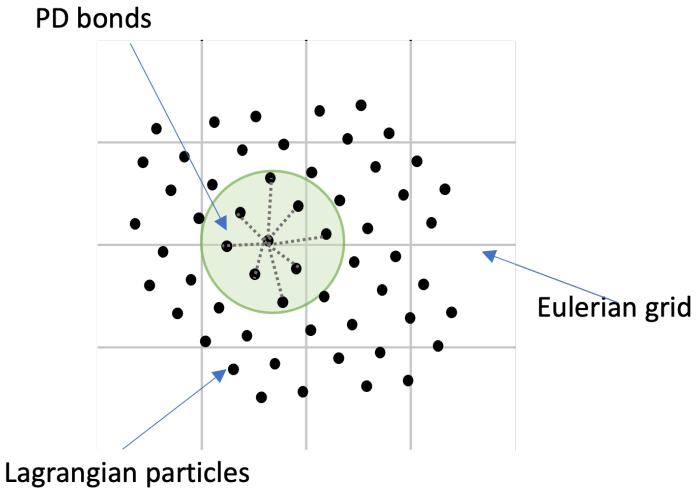

Figure 5.1: An illustration of the augmented PD bonds in our integral-based MPM.

PD bonds, the partial derivative based governing equation of the MPM can be replaced by an integral equation. Specifically, the MPM computes internal forces by integrating bonds deformations instead of the derivation computation of particle displacement, resulting in a novel integral-based MPM. This work has been summarised into publications (Lyu et al. 2019 2020).

Constitutive models formulated by traditional particle deformation gradient are not suitable for integral-based MPM. Motivated by the state-based PD, this integral-based MPM scheme describes the elastic, plastic and viscoelastic models with bond concepts as the novel constitutive models. We will go through the basic data flow and constitutive models in following content.

5.2 Method Outline

This chapter adopts the governing equation in the Eq. 3.2. Weak form is then obtained by multiplying the balance of momentum and integrating the governing equation over initial volume. Each material point is assigned with the PD bonds which connect the centre particle p with all adjacent particles within a given distance by virtual bonds. For one particle p, we propose an integral force density function $F^s(x_p)$ which encodes constitutive models. Then we reformulate the weak form as:

$$\int_\Omega \rho a \delta u_p d\Omega + \int_\Omega \rho F^s(x_p) \delta u_p d\Omega = \int_\Omega \rho b_p \delta u_p d\Omega + \int_{\Gamma_\tau} \rho \delta u_p \tau_p d\Gamma \qquad (5.1)$$

where Ω denotes the integrating region in the current configuration, ρ is density, a is acceleration, δu_p is the virtual displacement (infinitesimal feasible changes where constraints remain satisfied). b is the body force, for example, gravity. τ_p is the surface traction on part of the boundary Γ_τ. We will discuss the details of the integral force density function $F^s(x_p)$ in Chapter 5.3.

A thorough description of the MPM and the PD is given in Chapter 3 and Chapter 4 respectively, but we will sketch the main important notations of these two methods here: the material domain at time t^n is discretized with particles at x_p^n. Each particle has volume V_p, mass m_p, velocity v_p^n, and deformation matrix F_p, *Lamé* parameters μ_p and λ_p, plastic yield parameters ψ_p. In each time step, a new grid is generated. Grid node i is used to store nodal parameters, such as position x_i, mass m_i, velocity v_i, force f_i. Each material point is added with PD bonds structure: neighbourhood particle x_q, family H_p, initial bond X, deformed bond Y, bond force state $T[x_p, t^n] < x_q - x_p >$ and the weighing function $w(X)$.

Multiplying the traditional MPM shape function to terms in Eq. 5.1 obtains the integral-based MPM transfer scheme. The procedures are same as the traditional MPM while with different way of collecting the deformation of particles. In the traditional MPM, the **Grid update** step computes internal forces $f_i^n = -\sum_p V_p^n \sigma_p \nabla \omega_{ip}^n$ on the grid. In the integral-based MPM,

the new formulation of the internal forces is:

$$f_i^n = \sum_p V_p^n m_p w_{ip}^n F^s(x_p) \tag{5.2}$$

We present the traditional MPM transfer procedures (see Stomakhin et al. (2013) for details) to both elucidate the basic data flow as well as highlight the augmentations that make up our integral-based MPM in Fig. 5.2.

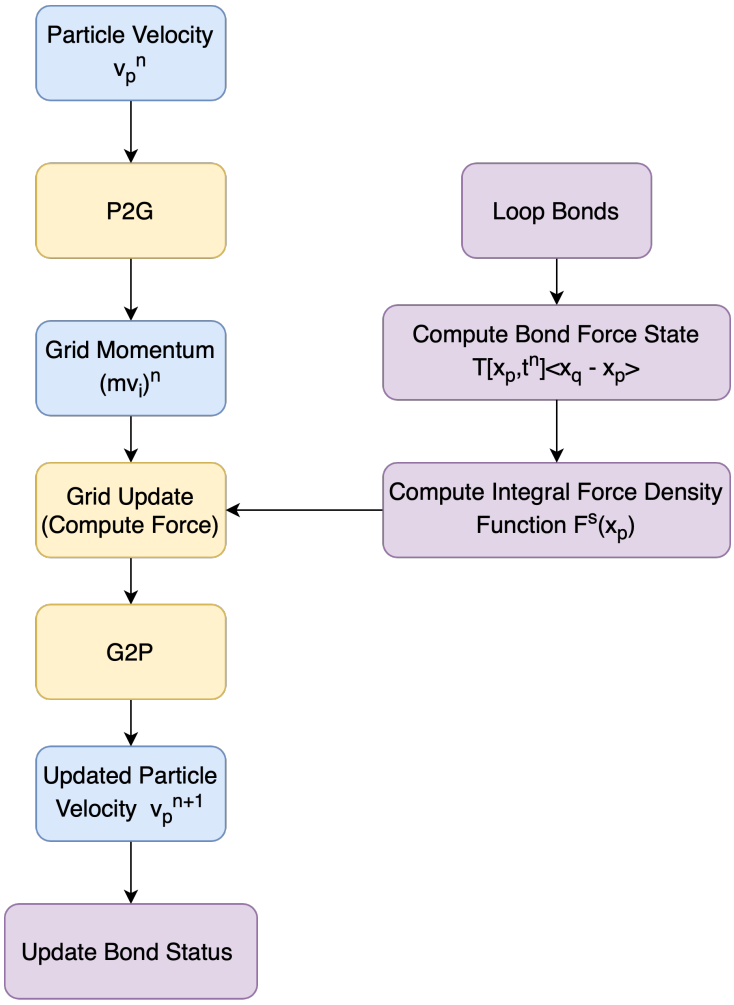

Figure 5.2: An overview of the integral-based MPM. Our augmentations are highlighted with colour purple.

The transfer procedures are depicted in Fig. 5.2: first, we transfer mass and momentum from particles to grid. Second, we use the elastic, viscoelas-

tic models to compute grid node internal force by Eq. 5.2 and update grid velocity. We transfer grid velocity back to particles and advect particle positions. Then update the particle deformation gradients and process them with return mapping rules for plasticity (see Chapter 5.3.2). The last step is to test all unbroken bonds with the failure criterion and remove broken bonds.

5.3 Constitutive Models

In contrast to the partial differential equations involving spatial derivatives of particle displacements $\frac{\delta u_p}{\delta x_p}$ in the traditional MPM, we describe the internal force density function in Eq. 5.3 using an integral formulation of pairwise bond forces. u_p is the displacement of particle p.

$$F^s(x_p) = \int_{H_p} [T[x_p, t^n] < x_q - x_p > -T[x_q, t^n] < x_p - x_q >]/\rho dH_p \quad (5.3)$$

5.3.1 Elastic Models

We start with the Fixed Corotated energy density function (Stomakhin et al. 2012) (presented in Chapter 3.3.1) in Eq. 5.4. We sketch the energy density function here:

$$\psi_{Corotated} = \mu \|F_p - R_p\|_{F_p}^2 + \frac{\lambda}{2}(J_p - 1)^2 \quad (5.4)$$

where R_p is the rotation matrix and $J_p = det(F_p)$. The energy density function can be decomposed into two parts : deviatoric part $\mu \|F_p - R_p\|_{F_p}^2$ and isotropic part $\frac{\lambda}{2}(J_p - 1)^2$. F_p is based on the spatial derivatives of displacement which brings underlying issues in computing singularity, such as discontinuities. Silling et al. (2007) redefined the F_p using bond concepts in the Eq. 4.4. Thus an average deformed bond is calculated as $\overline{Y} = F_p X$.

Following the continuum mechanics theory, the energy of each deformed bond contains deviatoric and isotropic components:

$$\psi = \sum_{H_p} w(X)(\mu E^{dev} + \frac{\lambda}{2} E^{iso}) \quad (5.5)$$

where $E^{dev} = (\frac{|\overline{Y}|}{|X|} - 1)^2$ describes an average deformed energy of particle p. It removes the influence from different bond length and enable it to simulate material with the same stiffness. $E^{iso} = (\frac{|Y|}{|X|} - 1)^2$ represents single bond energy which is similar to a mass spring system, referring to the volumetric response.

$(\mu E^{dev} + \frac{\lambda}{2} E^{iso})$ is the total energy of bond X. Its bond force density function $T[x_p, t^n] < x_q - x_p >$ can be obtained by derivating the entire bond energy with respect to deformed bond Y:

$$T[x_p, t^n] < x_q - x_p >= \frac{2\mu w(X)}{|X|^2}(\overline{Y} - |X| dir \overline{Y}) + \frac{\lambda w(X)}{|X|^2}(Y - |X| dir Y) \quad (5.6)$$

where $\frac{2\mu w(X)}{|X|^2}(|\overline{Y}| - |X|) dir(\overline{Y})$ involves the average deformed bond $\overline{Y}$, similar to sheer effects. $\frac{\lambda w(X)}{|X|^2}(|Y| - |X|) dir(Y)$ has the same direction as the deformed bond Y. $dir(Y)$ represents the normalised direction of vector Y. Substituting Eq. 5.6 and Eq. 5.3 into Eq. 5.2, the internal force of grid node i is:

$$f_i^n = \sum_p V_p^n m_p w_{ip}^n [\sum_{H_p} w(X)(T[x_p, t^n] < x_q - x_p > - T[x_q, t^n] < x_p - x_q >) V_q] \quad (5.7)$$

Notably, in Eq. 5.4, a rotation matrix R_p is required which is computed by a singular value decomposition (SVD) of the deformation gradient F_p. With our novel deformation gradient and energy density function, we naturally avoids the necessary SVD. Furthermore, we do not use it for three reasons: 1) The current bond state Y is all we need for processing grid internal forces, resulting in a faster local step. 2) The standard SVD implementations can have a dramatic impact on performance (Chao et al. 2010). Although it is not essential for performance to avoid the SVD, it is preferable not to implement the SVD. 3) We simply define the plasticity on the bond concept which is plausible and less complicated compare to the existing MPMs involving the SVD.

5.3.2 Plastic Models

Many methods, for example, Stomakhin et al. (2013), Zhu et al. (2015) took out part of the elastic deformation gradient tensor that exceeds the yield function and pushed it into the plastic deformation gradient calculation. We define the plastic deformation e_p on the bond, that means part of a bond's extension or compression could be plasticity. This part of extension does not contribute to the forces.

Our plasticity model is based on the theory that the plastic model is purely from deviatoric plastic flow (Chen et al. 2018). We assume $|\overline{Y}|-|X| \approx |Y|-|X|$ when the deformation is smooth enough under small neighbour horizon. Then the Eq. 5.6 becomes:

$$T[x_p,t^n] < x_q - x_p >= \frac{2\mu w(X)}{|X|^2}(|Y|-|X|)dir(\overline{Y}) + \frac{\lambda w(X)}{|X|^2}(|Y|-|X|)dir(Y) \tag{5.8}$$

where an unified displacement term can be extracted as: $e = (|Y|-|X|)/|X|$. Following the deviatoric plastic rule that plasticity exists in deviatoric deformation component, the unified displacement can be decomposed into isotropic and deviatoric parts: $e = e_{iso} + e_{dev}$. Plastic deformation e^p is then extracted from e_{dev}. When the plasticity model is applied, Eq. 5.8 is developed into:

$$\begin{aligned} T[x_p,t^n] < x_q - x_p > = & \frac{2\mu w(X)}{|X|}(e_{iso} + e_{dev} - e_p)dir(\overline{Y}) \\ & + \frac{\lambda w(X)}{|X|}(e_{iso} + e_{dev} - e_p)dir(Y) \end{aligned} \tag{5.9}$$

We define the yield function as $f(E_{dev})$:

$$f(E_{dev}) = \frac{(E_{dev})^2}{2} - \psi_p, E_{dev} = \frac{w(X)}{|X|}(2\mu+\lambda)(e_{dev} - e_p) \tag{5.10}$$

where ψ_p is a controllable plastic material parameter. We use $f(E_{dev})$ to judge if current configuration enters the plastic regime. If $f(E_{dev}) < 0$, the deformation of bond is still within the elastic domain. If $f(E_{dev}) > 0$, part of deformation occurred as plasticity. We project the deformation back to the

yield surface and add plastic increment Δe_p to e_p permanently in Eq. 5.11.

$$\Delta e^p = \frac{|X|}{\lambda}[E_{dev} - \sqrt{2\psi_p} sign(E_{dev})] \tag{5.11}$$

This model is still valid for elasticity when e^p varnishes in all above equations or ψ_p is set to extreme high. ψ_p offers the integral-based MPM with rich artistic control capabilities. With appropriate plastic parameter, the plastic model can be a user-controllable constitutive model for simulating elasticity and elastoplasticity simultaneously in our integral-based MPM solver.

5.3.3 Linear Viscoelastic Models

Viscoelastic materials behave with elastic resistance to deformation similarly to elastic objects, while presents complex non-Newtonian fluid characteristics. During the simulation, they undergo extreme large deformations and can not easily restore initial shapes, for example, flowing honey and whipped cream, that are different from elastic solid materials. With these observations, researchers model the viscoelasticity either augmenting elastic models or applying the materials with extreme plasticity. If the viscoelasticity is augmented from an elastic model, one research direction is to continuously augment the reference configuration by current deformed configuration. In another word, the current deformed configuration is equilibrated.

Silling (2010) proposed an linearized peridynamics theory for adapting a viscoelastic model with the state-based PD. This method superposes small deformation on an existing large deformation in the configuration at last step. The internal force density function takes the small deformation (small bond extension) to update the response of viscoelastic material. The current deformed bonds are used as the reference configuration. With this prerequisite, we modify the force density function in Eq. 5.8 by removing information from initial configuration. Here we utilise the PD bond structures and investigate large topological changes experienced by processing the small incremental deformation based on the current large deformation field.

Let u_p be the small displacement field superposed on the current deformation field. Linearizing the function $T[x_p, t^n] < x_q - x_p >$ near current

deformation field leads to:

$$T[x_p, t^n] < x_q - x_p >= T[x_p, t^{n-1}] < x_q - x_p > +M[x_p] \cdot U[x_p] \qquad (5.12)$$

where $T[x_p, t^{n-1}] < x_q - x_p >$ is the deformed bond force density at previous step, $U[x_p] = \sum_{H_p} w(X)(Y - X)$ is the bond displacement state. $M[x_p] = \nabla T[x_p, t^n] < x_q - x_p >$, is the modulus state. It is the *Fréchet* derivative of the force density function with respect to the bond $Y^{'}$ which is between x_p and one of its family particles x_o (both x_o and x_q are in H_p). $Y^{'}$ may not be Y. $M[x_p]$ replaces the strain in previous sections and represents the deformation of material particles in the linearized constitutive models. Detailed explanation can be found in Silling's work (Silling 2010 2011).

With the linearized equation of the bond between particle x_p and x_q, the internal force density function F^s is rewritten as :

$$\begin{aligned} F^s(x_p) = \int_\Omega \int_\Omega [T[x_p, t^{n-1}] < x_q - x_p > +M[x_p] \cdot U[x_p]) - \\ (T[x_q, t^{n-1}] < x_p - x_q > +M[x_q] \cdot U[x_p])] / \rho dH_q \end{aligned} \qquad (5.13)$$

Based on the assumption of the equilibrated deformation in Silling (2010), the reference configuration is equilibrated which indicates $\int_{H_p} (T[x_p, t^{n-1}] < x_q - x_p > -T[x_q, t^{n-1}] < x_p - x_q >)] / \rho dH_{x_p} + b = 0$.

The energy density function therefore is :

$$\begin{aligned} F^s(x_p) = & \int_{H_p} \int_{H_p} [M[x_p] < x_q - x_p, x_o - x_p > (u(x_o) - u(x_p))] / \rho dH_p dH_p \\ & - \int_{H_p} \int_{H_{x_q}} [M[x_q] < x_p - x_q, x_m - x_q > (u(x_m) - u(x_p))] / \rho dH_q dH_p \end{aligned} \qquad (5.14)$$

where x_q and x_o are family particles of the point x_p , x_m is the family particle of the point x_q. This function not only involves the family particles of point x_p, but also the family particles of point x_q. Thus the zone being influenced is fundamentally double than the horizon defined before. Fig. 5.3 shows the indirect interaction between x_p and x_m.

The linearization of the governing equation requires the current displacement and the modulus state, which are irrelevant to the initial configuration.

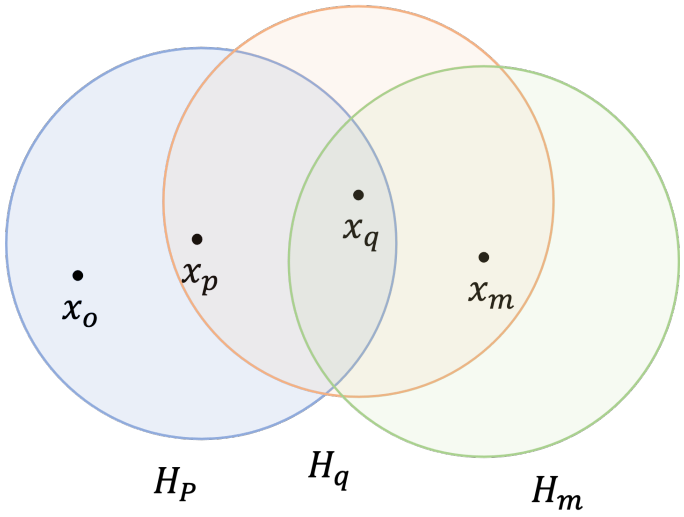

Figure 5.3: Material point x_p interacts with x_m even though they are outside each other's horizon because they are both within the horizon of an intermediate point x_q

It produces structureless material effects and is appropriate for colloidal materials, such as foam, cream and sponge. Furthermore, based on the PD, it is feasible to model discontinuous distributions, such as bubbles in the foam and dripping cream.

We adopt the viscoelastic model defined in Silling (2010), Silling et al. (2007) to achieve the viscous effects. This is motivated by the integral fluid constitutive model with one energy component. Then energy density function for viscoelastic materials is defined as:

$$\psi = \frac{\lambda\vartheta^2}{2} + \frac{\mu}{2}\sum_{H_p} w(|X|)(e - \frac{\vartheta}{3})^2 V_p \tag{5.15}$$

$$\vartheta = \frac{3}{S}\sum_{H_p} w(|X|)|Y|eV_p \tag{5.16}$$

$$S = \sum_{H_p} w(|X|)|Y|^2 V_p \tag{5.17}$$

Where e is the bond extension, as defined in the last section. ϑ represents the dilatation part of the bond extension. Thus the deviatoric part means the bond extension subtracts an isotropic expansion in family horizon. It contains

not only shear, but also any deformation from the family particles other than isotropic expansion. The viscoelastic force density function $T[x_p, t^{n-1}] < x_q - x_p >$ is written as:

$$T[x_p, t^{n-1}] < x_q - x_p >= (\frac{3\lambda}{S} - \frac{\mu}{3})w(|X|)|Y|\vartheta S + \mu w(|X|)edir(Y) \quad (5.18)$$

For two neighbour particles in H_p, we define $Y_1 = x_q - x_p$ and $Y_2 = x_o - x_p$ which are incorporated in the modulus state $M[x_p]$. Thus $M[x_p]$ is written as:

$$M[x_p] = (\frac{9\lambda}{S^2} - \frac{\mu}{S})w(|Y_1|)w(|Y_2|)Y_1 \otimes Y_2 + \gamma(Y_1)\Delta(Y_2 - Y_1) \quad (5.19)$$

$$\gamma(Y) = \lambda w(|Y|)(dir(Y) \otimes dir(Y)) \quad (5.20)$$

where Δ denotes the Dirac delta function in the simulation field. Substituting Eq. 5.19 and Eq. 5.18 into Eq. 5.2, the transfer procedure of using MPM to simulate viscoelastic material is completed.

5.4 Fractures and Failure Law

Modelling cracks is a well-known issue of the MPM (Liang et al. 2017). To model discontinuous particle distribution, special treatments for creating cracks and partitioning fracture fragments are needed. With the PD bonds, the discontinuities is straightforward to simulate in this MPM framework. If we simply remove over-deformed bonds like the strategy in Chapter 4.3, it leads to numerous small fragments in the deformed area rather than several crack lines after collision happens. We propose generating a crack cut by the fracture plane based on the analyses of single point and global deformation status in the gird cell.

First, considering that plasticity exists as permanent deformation, we first remove the plastic deformation from the failure criterion:

$$l = \frac{e - e^p}{|X|(1 + p_{inactive})} \quad (5.21)$$

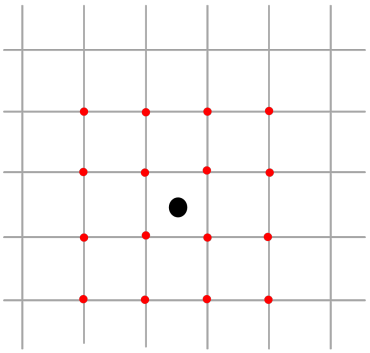

Figure 5.4: The information of one particle is transferred to neighbour 9 grid cells (information saved on 16 grid nodes) in 2D dimension in the MPM.

where $p_{inactive}$ is the percentage of broken bonds in total bonds in one grid cell.

We check each connected bond by comparing the l with the critical failure threshold $l_{threshold}$. Then cluster all particles with broken bonds into several groups based on their positions and normals. For each area, we find a central point and the largest deformed bond. The next step is to use the central point as the position and the direction of the largest deformed bond as the normal to construct a fracture plane for each area. Any bond intersected by the fracture plane will be removed. Accumulating all broken bonds forms in crack lines. If one grid has too many broken bonds, the active bonds in this grid cell are less likely to be removed. With the augmentation of $p_{inactive}$, this method offers an efficient strategy for artists to design fracture effects with desired fragments numbers and size.

As the preceding description, crack details is limited to grid resolution in the traditional MPM. In experiments, we transfer particle velocity to three grid cells (illustrated in Fig. 5.4) in any direction to get stable, smooth results. When two sides of crack line are within this range, they will share additional fragment information through transfer. Therefore the use of one grid leads to the effects that fragments hard to be separated. We alleviate this problem by applying two-time integration methods: material points on the crack surface are updated by its own bond forces; other material points which don't have any broken bonds (in the fragmented inner parts) are updated by grids as

normal.

5.5 Simulation Results

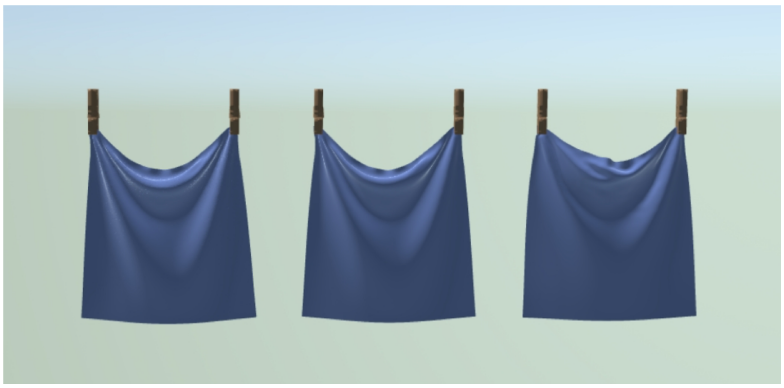

Figure 5.5: Different bending effects are achievable through animating cloth with varying material stiffness: $\mu = 2 \times 10^4$, $\mu = 1.5 \times 10^5$, $\mu = 1 \times 10^6$.

Several examples demonstrate the efficacy of our integral-based MPM, showing that this model is able to capture key features of elastic, plastic and viscoelastic materials. In particular, the fracture model can be applied to any above material. The implementation of our method is completed in Houdini software by Houdini Development Kit (HDK) language, including the material point discretization, physical-based solver, voronoi fracture generation, surface reconstruction and rendering. In the implementation, we compile above procedures using the HDK and generate several Houdini plugins. All plugins can be combined to model any materials with or without fractures. Table. 5.1 lists the modelling types, parameter settings and the performance data for all examples presented in this chapter.

Implementation. In this work, we have used three types of modelling geometries.

1. Mesh-based geometry: With given input surface, we construct mesh as the surface and distribute several layers of particles underneath the surface.

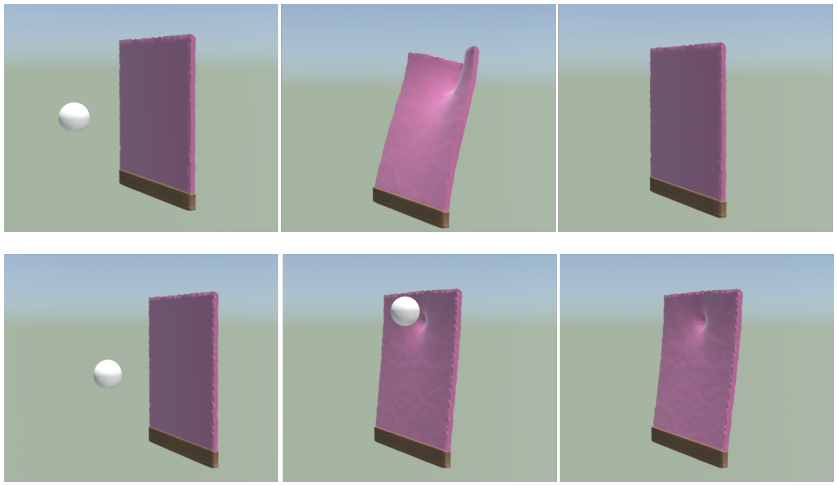

Figure 5.6: Throwing a ball to the elastic and plastic board shows visibly different results. In the first row, the elastic board restores its initial shape. The second row shows that plastic board keeps deformed topology.

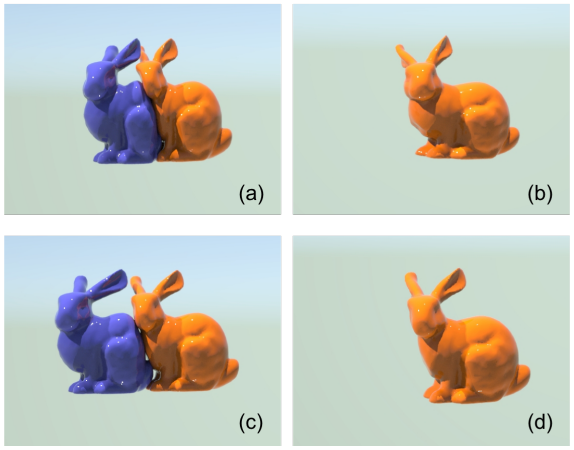

Figure 5.7: The collision between two identical rabbits with different materials. This example demonstrates the different deformation of elastic rabbits (in the first row) and plastic rabbits (in the second row). Collision happens in (a) and (c). The elastic rabbits are able to recover as in (b). The plastic rabbits deforms permanently in (d).

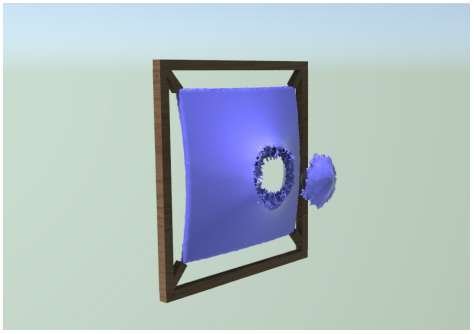

Figure 5.8: Simulation of a plastic wall when collided by a sphere.

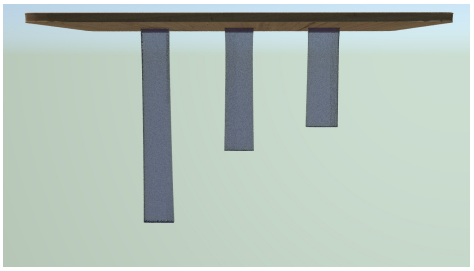

Figure 5.9: Stretch bars with different material parameters. From left to right, bending stiffness μ are: 0, 50, 500.

Table 5.1: Experiment Settings.

	Type	Δx	$\lambda(MPa)$	$\mu(MPa)$	ψ_p	$l_{threshold}$	$\Delta t(ms)$
Cloth	Mesh	0.005	1×10^6	$2\times10^6, 1\times10^5, 1.5\times10^5$	1×10^{25}	1×10^{25}	0.1
Elastic board	Mesh	0.02	3×10^6	1.5×10^5	1×10^{25}	1×10^{25}	0.02
Plastic board	Mesh	0.02	5×10^6	1.5×10^5	100	1×10^{25}	0.0001
Elastic bunny	Particle	0.01	1×10^5	1×10^5	1×10^{25}	1×10^{25}	0.1
Plastic bunny	Particle	0.01	1×10^6	3×10^5	30	1×10^{25}	0.01
Broken board	Voronoi	0.005	1×10^6	5×10^4	500	0.05	0.001
Stretching bar	Mesh	0.05	500	0,50,500	1×10^{25}	1×10^{25}	100
Honey	Particle	0.05	500	100		0.0304	10
Cream pie	Particle	0.04	5×10^3	500		0.035	10

2. Particle-based geometry: For specific examples, such as the viscous elastic material, we represent initial objects by volume and scatter unstructured particles within the volume, for simplicity.

3. Voronoi cell geometry: Houdini designs the voronoi cell particularly for fractures or crack propagation. Each voronoi cell is an irregular polygon. We treat its centre point as the material point. The bond represents the connection between adjacent voronoi cells.

Note that ghost particles are added outside the object surface to guarantee that each material point has a similar family density at the initial step.

Model validation. We demonstrate our method by several examples. Fig. 5.5 shows the a piece of garment anchored by clothes pegs, with varied bending parameters μ. It shows that realistic and fine wrinkles can be

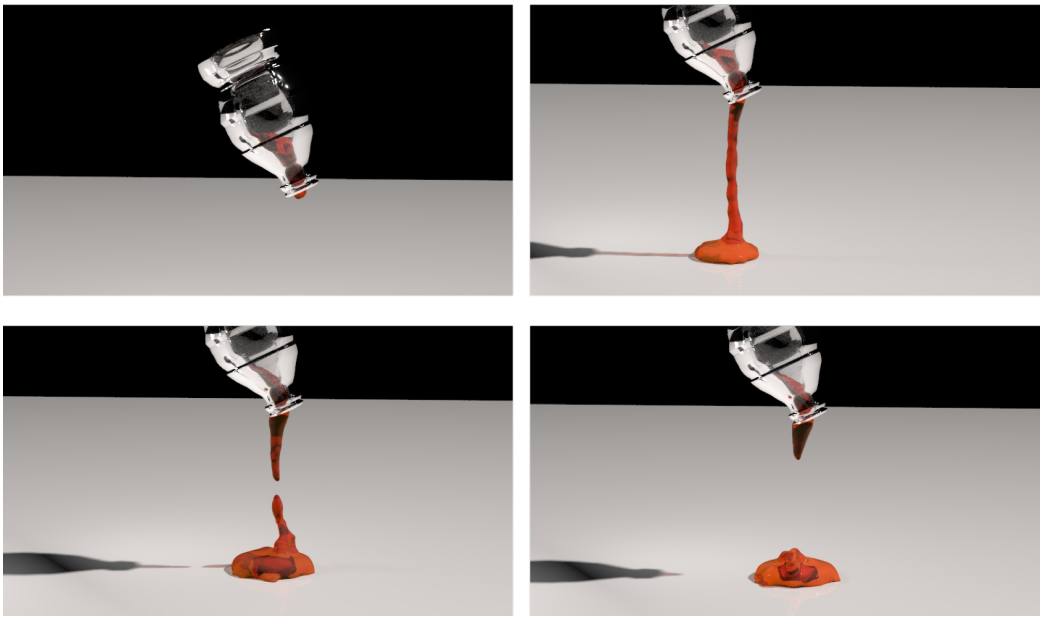

Figure 5.10: Linear peridynamics theory is used for modelling viscoelastic fluids, such as honey.

achieved by tuning μ. Fig. 5.6 shows key difference between elasticity and plasticity through shooting a ball to a elastic board and a plastic board. The elastic board restores its shape after the collision while the plastic board keeps the damage permanently. This method can handle objects with complex typologies, as shown in Fig. 5.7. Fig. 5.8 shows ductile fractures modelled by voronoi polygons. Fig. 5.9 demonstrates different material stiffness leads to various bar deformation. Our linearized peridynamics model within the MPM is able to represent the viscoelastic behaviours, which presents large deformation similar to fluid while keeps the ability of elastic resistance. Fig. 5.10 shows that honey flows from a glass bottle and drops on the floor. Fig. 5.11 shows that when whipped cream is thrown to the wall, it flows like dense fluid, whilst slightly retaining its original shape.

5.6 Summary

In this chapter, we add material points with PD bonds. With this augmentation, we propose to incorporate the integral-based force computation of

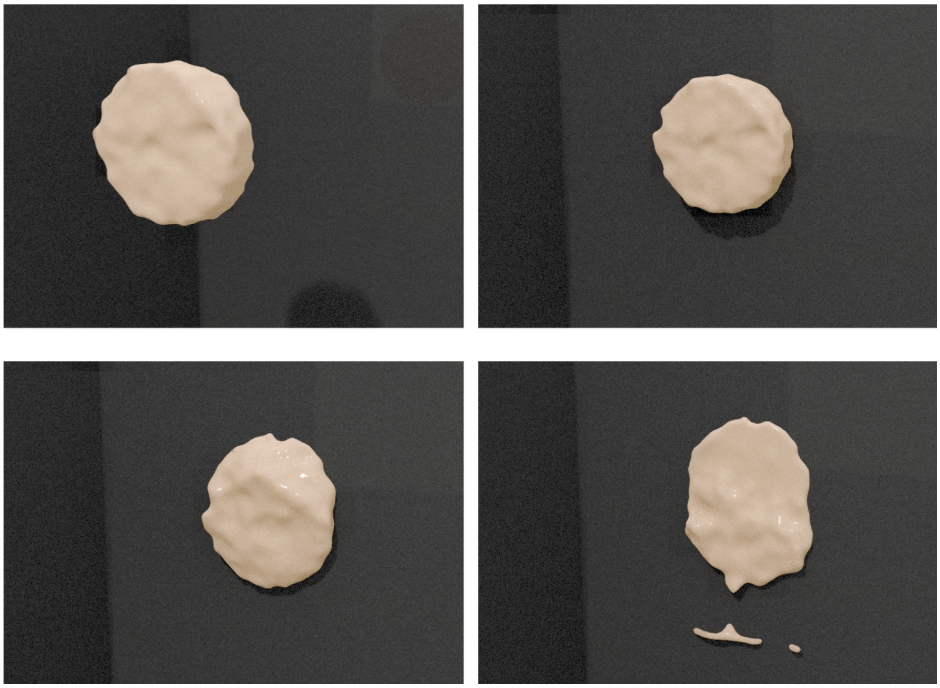

Figure 5.11: A pie of whipped cream is thrown to the wall to show the effects of viscoelasticity.

the PD to the internal forces in the traditional MPM, which outweighs the differential-based MPM in both accuracy and stability. Within this novel MPM transfer scheme, we introduce the elastic, plastic and viscoelastic constitutive models. Specifically, we adopt a simple yield function based on deviatoric flow theory to animate plasticity. A failure law is incorporated to produce fractures. This Chapter provides an attractive method for producing a variety of elastoplastic materials and fracture with visual realism and high stability.

Chapter 6

Superposition-based Material Point Method (SPB-MPM)

6.1 Introduction

Whether it be crashed cars, exploded watermelons, or collapsed building shot by bullets, high speed impact scenarios are ubiquitous in daily life and a modern visual effects. Modelling these phenomena is notoriously difficult due to the fact that objects moving at high speed can generate intricate interactions in a short time period between surrounding materials, leading to extreme devastation and a tremendous amount of debris acting like a small, local explosion ahead of the impact area. Accurately capturing these unpredictable fracturing fragments is necessarily important. But with the increasing need of sophisticated fracture scenarios in visual effects industry, the efficiency of a method handling numerous fragments has also been set as another priority in the research.

With numerous state-of-art methods to handle large topological changes, the focus of simulating fractures has centred on accommodating the discontinuities. Some works rely on geometric decomposition instead of physical definitions of cracks (Ne and Fiume 1999). Other physics-based methods explore basis functions to account for discontinuities in materials, included are the Extended Finite Element Method (XFEM) (Kaufmann et al. 2009) and the Affine Particle In Cell method (APIC) (Jiang et al. 2015). The traditional MPM fails to model sharp separation of material points and cannot

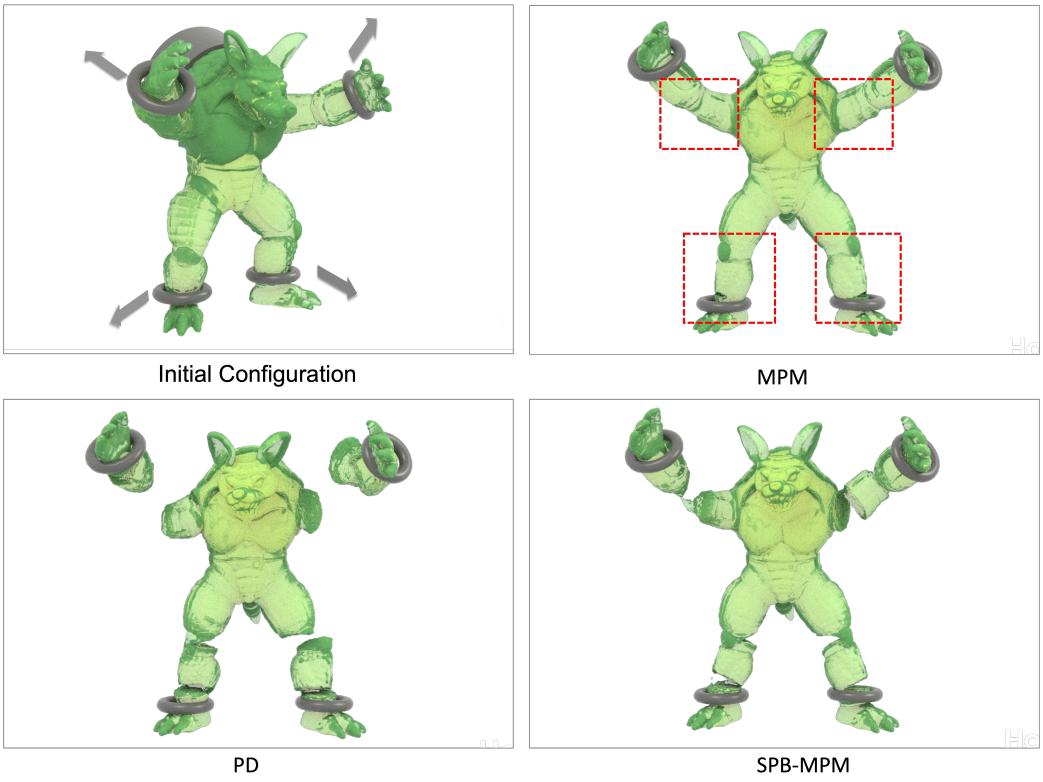

Figure 6.1: Stretched armadillo. We compare a stretched armadillo using the MLS-MPM, the state-based PD and our SPB-MPM (both applied to pure elastic materials). Compared to the PD, the SPB-MPM provides almost identical visual results and consumes less computation time.

represent discontinuous velocities (Hu et al. 2018). When coupled with a plasticity model, the MPM is able to separate a pile of snow (Stomakhin et al. 2013), however, it still struggles to separate elastic continuum materials and model complex fractures. Later, the Moving Least Squares Material Point Method (MLS-MPM) adopted a Moving Least Squares shape function in the MPM which leads to a APIC force formulation to overcome numerical failures when handling discontinuities (Hu et al. 2018). A continuum damage material point method (CD-MPM) was then proposed to treat cracks using a continuum damage field and evolve the damage variables to track crack surfaces within the MPM scheme (Wolper et al. 2019). CD-MPM shows great promise for simulating a breadth of fractures in organic materials while is

designed for general fracture cases, for example, stretched jello and smashed cookies. Due to that the continuum phase field is evolved within the MPM transfer scheme, it naturally inherits the limitation of using the grid, for example, the crack details heavily depends on the grid resolution. Moreover, if it is applied to the impact scenarios, a delicate balance is required between the overhead computation brought by the continuum phase field and myriad fractures on a sufficient large grid resolution. Instead placing a continuum field over the entire grid, we seek for an efficient solution to handle particular areas around fractures presented by the high velocity impact applications.

The PD theory (Silling et al. 2007) as an nonlocal reformulation of classical solid mechanics using spatial integral governing equations instead of partial differential equations, demonstrates great potential in generating various fracture behaviours (He et al. 2017). Contrary to the MPM, the PD brings computational cost when simulates an unbroken deformed object because each PD particle needs run iterations over its neighbourhood at every step. Some researchers dived into the literature and came up some coupling schemes through combining the PD with the FEM or other mesh-based methods. They usually use special treatments to handle the transition area, such as force-blending (Seleson et al. 2013), constitutive parameter morphing (Lubineau et al. 2012), and different sub-region coupling by means of interface elements (Liu and Hong 2012), but both of these operations are lack of accuracy and not physically plausible. The FEM-PD coupling scheme (Sun and Fish 2019, Sun et al. 2019) places PD particles over some particular FEM meshes to enrich crack details. These meshes with scattered PD particles are named as PD patches. This coupling method does not need the computational-costly treatment of the transition region between the FEM and the local strategy PD. Considering the MPM also adopts the same weak formulation as in the FEM (Nguyen 2014), we are inspired to couple the PD and the MPM, resulting in a Superposition-based MPM (SPB-MPM) coupling method for two reasons: 1) both the MPM and the PD are particle-based methods. It is feasible to transform some MPM material points to PD patches. 2) The MPM Eulerian grid is placed below all particles. It can be utilised to collect PD particle quantities and transfer them back to the

MPM. Thus parameter morphing in the transition area is unnecessary in our coupling method.

With the coupling scheme, our next concern is how to properly place PD particles over particular MPM regions where fractures may appear. Sun et al. (2019) initialised the PD patches around a pre-defined crack tip on the FEM meshes. It is not a physically plausible approach to place PD particles over a given area, especially not applicable in computer graphics due to the fact that crack tips are always unpredictable. In this study, we relate the generation of PD patches to the released energy theory. Specifically, we compute released energy on the grid nodes in a Eulerian manner and generate a PD patch when one grid cell reaches the critical released energy. That means that crack possibly appear in this grid cell at the next simulation step. We incorporate a Eigenerosion method (Pandolfi and Ortiz 2012) and develop it within the MPM scheme. The Eigenerosion method has been used to corrode broken elements in FEM through accumulating released energy in adjacent structures. Similarly, we find 'corrode' grid cells (here we name them as Eigenerosion Enhanced grid cells) and prepare these grid cells with PD patches. With the PD's strong ability of handling discontinuities, we are able to simulate arbitrary discontinuities (e.g. shattering debris clouds, ductile fractures and brittle fractures). Giving a low-overhead PD computation to the current MPM scheme, we are allowed to simulate a breadth of fractures with the notable ability of arbitrarily large topological changes. In this Chapter, we incorporate the state-based PD (Silling et al. 2007) to model the same classical constitutive materials as in the MPM to ensure we simulate the local and entire simulation domains with same material properties.

6.2 Methodology

As mentioned in the introduction, we seek a solution to embed the PD within a MPM scheme for intuitively modelling large amounts of fragments with manifold cracks while robustly and stably keeping damaged chunks during the impact process. Superposition-based FEM-PD coupling method (Sun et al. 2019) stands as an efficient and rigorous strategy for treating cracks by

placing PD patches over parts of FEM mesh. It treats the FEM mesh as background and project PD information on it, thus transition regions between two methods are avoidable. A PD patch evolves in the one FEM mesh element which is approached by crack tip. The Superposition-based FEM-PD is able to simulate FEM and PD in a computationally efficient manner. This coupling method inspires us to investigate and develop a tailored approach for efficiently modelling a large quantity of fragments within a continuum body. We adopt the rigorous coupling governing equations while use the MPM discretization scheme, resulting in the Superposition-based MPM (SPB-MPM). We illustrate the particle-mesh structure of the coupling FEM-PD and our SPB-MPM in Fig 6.2.

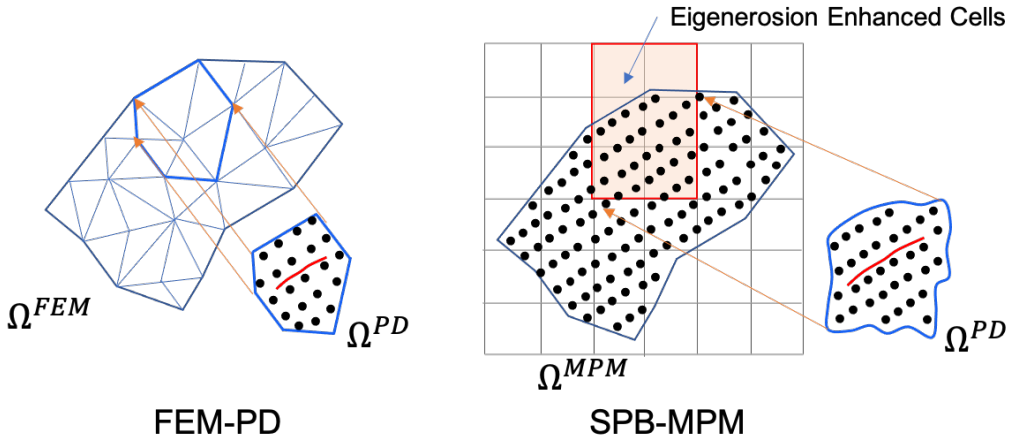

Figure 6.2: An illustrated comparison of PD patch generation in FEM coupling method and SPB-MPM coupling method (SPB-MPM).

We start to describe the SPB-MPM by defining two problem domains: a MPM point domain over the entire body Ω^{MPM^0} and a PD particles domain Ω^{PD^0} ($\Omega^{PD} \subset \Omega^{MPM}$) (as in Fig. 6.2). The superscript depicts the time t. The deformation map $x = \Psi(X, t)$ where x and X are world and material coordinates respectively. MPM is used to derive the entire configuration. After MPM updates the deformable object as Ω^{MPM^t}, PD is brought to evolve particles in Ω^{PD^t}.

In the MPM, the simulation objects are discretized using quadrature particles as material points to carry Lagrangian physical quantities (mass m_p, volume V_p, velocity v_p, position x_p, deformation gradient F_p, affine matrix C_p, internal energy e_p for shockwave effects models) and a Cartesian grid as a scratchpad to evaluate internal forces. The grid nodes carry grid velocity v_i, grid mass m_i, internal forces f_i and Grid release energy G_i. Eigenerosion enhanced cells are defined with $EE_i = 1$ for our coupling method. The PD particles inherit unstructured MPM particle properties (mass m_p, volume V_p, velocity v_p and position x_p). PD also defines bond concepts between particle x_p and one of its neighbouring particle x_q: initial bond $X = x_q^0 - x_p^0$, deformed bond $Y = x_q^n - x_p^n$. Due to the PD non-local features, PD particles are initialised with virtual bonds which are represented using: the Horizon H_p containing a list of neighbourhood particles within a given radius δ and the force state vector $T[x_p, t^n] < x_q - x_p >$. As in the state-based PD (in Chapter 4), $T[x_p] < x_q - x_p >$ represents the force state vector of particle x_p with the bond from x_p to x_q. After collecting all force vectors in the Horizon, PD particles are updated using explicit time integration. We use subscript i, j, k to denote grid node quantities and p, q to denote all particle quantities, superscripts MPM and PD denote the MPM and the PD domain respectively. And superscripts n and $n+1$ represent simulation timestep n and $n+1$. Most of the quantities have been explained in previous chapters. We sketch important quantities in Table. 6.1.

Table 6.1: SPB-MPM quantities.

Material particles	m_p, v_p, V_p, e_p
Eulerian grid	$m_i, x_i, v_i, f_i, G_i, EE_i$
MPM (SPB-MPM)	$C_p, F_p^{MPM}, \sigma_p^{MPM}$
PD (SPB-MPM)	$H_p, \delta, F_p^{PD}, \sigma_p^{PD}, T[x_p, t^n] < x_q - x_p >$

6.2.1 Governing Equation

Our work adopts the rigorous FEM-PD coupling governing equations (Sun et al. 2019) in Eq. 6.1.

$$\rho a = \nabla \cdot \sigma + b \tag{6.1}$$

where ρ is mass density, b is the body force, σ is the Cauchy stress and a is acceleration. The Cauchy stress σ is a function of displacement fields. The definition of displacement field u in Eq. 6.2 is approximated by superimposing displacements resulting from MPM configuration u^{MPM} with PD patches u^{PD}, which is $\sigma(u^{MPM} + u^{PD})$ in Ω^{MPM} and $\sigma(u^{MPM})$ in $\Omega^{MPM} \setminus \Omega^{PD}$ (MPM domain subtracts PD domain).

$$u = \begin{cases} u^{MPM} & x \subset \Omega^{MPM} \setminus \Omega^{PD} \\ u^{MPM} + u^{PD} & x \subset \Omega^{PD} \end{cases} \tag{6.2}$$

Through multiplying the differential governing equation by a test function (here w^{MPM} and w^{PD} are the test functions of MPM domain of Ω^{MPM} and the PD domain Ω^{PD}, respectively), the weak formulation of the coupling governing equation is obtained in Eq. 6.3. The test function is chosen from kinematically admissible displacements satisfying geometric constraints.

$$\begin{aligned} \int_{\Omega^{PD}} (w^{MPM} + w^{PD}) \cdot (\nabla \cdot \sigma(u^{MPM} + u^{PD}) + b(x,t) - \rho a) dx \\ + \int_{\Omega^{MPM} \setminus \Omega^{PD}} w^{MPM} \cdot (\nabla \cdot \sigma(u^{MPM}) + b(x,t) - \rho a) dx = 0 \end{aligned} \tag{6.3}$$

As an non-local extension of the classical solid mechanics, the PD uses different mechanism from the MPM. Considering the PD method does not directly affect the dynamics in $\Omega^{MPM} \setminus \Omega^{PD}$, we are able to restate above weak form in two equations with respect to different test functions: w^{MPM} and w^{PD}, and carry our discussion about the MPM dynamics and the PD dynamics separately. Appendix A gives full details of weak form separation.

For MPM, the terms involving w^{MPM} in Eq. (6.3) are:

$$-\int_{\Omega^{PD}} \sigma(u^{MPM}+u^{PD}) : \nabla w^{MPM} dx + \int_{\Omega^{MPM}} b(x,t)\cdot w^{MPM} dx \\ -\int_{\Omega^{MPM}\setminus\Omega^{PD}} \sigma_(u^{MPM}) : \nabla w^{MPM} dx = \int_{\Omega^{MPM}} \rho a \cdot w^{MPM} dx \quad (6.4)$$

If $u^{PD}=0$, the first and third terms on the left hand side are combined to become $-\int_{\Omega^{MPM}} \sigma(u^{MPM}) : \nabla w^{MPM} dx$, and Eq. (6.4) degenerates to a standard MPM weak form. The test function can typically be approximated by functions in a finite-dimensional function space as linear combinations of some basis shape functions. Here we adopt the MLS shape functions (Hu et al. 2018) and follow the standard MPM to discretize the above weak form. More details of MLS discretization process can be found in the MLS-MPM (Hu et al. 2018) and the APIC (Jiang et al. 2015).

The rest of Eq. (6.3) involving the PD test function w^{PD} is written in Eq. (A.1). Eq. (A.1) - Eq. (A.6) in Appendix A complete the proof of transforming the continuum weak form to integral PD equation. Here we highlight the conclusion: the internal energy term in the PD domain can be further rewritten from a continuum theory to an integral-based formulation:

$$\int_{\Omega^{PD}} \sigma(u^{MPM}+u^{PD}) : \nabla w^{PD} dx = \int_{\Omega^{PD}} T[u^{MPM}+u^{PD},t] dY^P dx \\ = -\int_{\Omega^{PD}}\int_{H_x} w^{PD}\cdot(T[x_p,t]<x_q-x_p> - T[x_q,t]<x_p-x_q>) dx \quad (6.5)$$

Where $T[x_p,t^n]<x_q-x_p>$ is the bond force between PD particles x_p and x_q at the timestep t^n. The state-based PD uses an integral of bond forces and avoids calculation of the spatial derivatives of particle displacement.

To summarise our work in this section, we evolve the coupling governing equation into the coupling weak form. We split the weak form to two equilibrium equations as in Eq. 6.4 and Eq. 6.5 according to the MPM test function and PD test function. They can be discretized into a standard MPM scheme and a state-based PD integral formulation.

6.2.2 State-based Peridynamics

The PD updates procedure has been explained in Chapter 4. We simply introduce the core stages of the state-based PD for better understanding the SPB-MPM. We loop each active family particle of x_p and accumulate all bond force pairs $T[x_p, t^n] < x_q - x_p >$ and $T[x_q, t^n] < x_p - x_q >$:

$$\rho a_p^{PD} = \Sigma_{H_p} \omega^{PD} (T[x_p, t^n] < x_q - x_p > -T[x_q, t^n] < x_p - x_q >) V_q^0 \quad (6.6)$$

where a_p^{PD} is acceleration of PD particle x_p, V_q^0 is the volume of particle x_q, initialised by the MPM. $\omega^{PD}(X)$ is chosen as an influence function for bond X that is monotonically decreasing with respect to the distance between particles and affected by bond breakage condition (in Eq. 6.13). The state-based PD model is able to convert the states to tensorial quantities and permit a general representation of the continuum mechanics using the stress-strain relation directly. Note that the forces acting on the PD particles are not necessarily constrained to the direction of the bonds. It is designed to match versatile constitutive models under the classical elasticity theory (He et al. 2017, Chen et al. 2018).

Aiming to converge to the classic constitutive models, the state-based PD introduces a symmetric shape tensor $K(x_p) = \Sigma_{H_p} \omega^{PD}(X) X \otimes X V_q^0$ and a PD deformation gradient tensor $F_p^{PD} = [\Sigma_{H_p} \omega^{PD}(X) Y \otimes X V_q^0] \cdot K(x_p)^{-1}$.

The strain tensor ϵ_p^{PD} is formulated as:

$$\epsilon_p^{PD} = \frac{1}{2} [\Sigma_{H_p} \omega^{PD}(X) [(U \otimes X + X \otimes U) V_q^0] \cdot K(x_p)^{-1} \quad (6.7)$$

The stress tensor σ_p^{PD} can be obtained from the strain tensor ϵ_p^{PD} with any given constitutive model. Note that we also propose a shock wave effects model which can be incorporated with the SPB-MPM for high speed impact deformation (in Chapter 7). The force vector is expressed in terms of the Cauchy stress σ_p^{PD} as an intermediate step in Eq. 6.8.

$$T[x_p, t^n] < x_q - x_p > = \omega^{PD}(X) det(F_p^{PD}) \cdot (F_p^{PD})^{-1} \cdot \sigma_p^{PD} \cdot K(x_p)^{-1} \cdot X \quad (6.8)$$

where $det(F_p^{PD})$ is the determination of deformation gradient F_p^{PD}. Substituting Eq. 6.8 into Eq. 6.6, we are able to update PD particles' velocity and position.

Notably, in this chapter, we adopt the classical Fixed Corotated model and the Neo-Hookean hyperelastic model as the constitutive models of the SPB-MPM. The SPB-MPM requires the local PD and the entire MPM domain derived by the same constitutive model.

6.2.3 Eigenerosion Enhanced Method

Eigenerosion has been proposed in Finite Element Analysis (FEA) to simulate fracture in mesh-representation applications (Pandolfi and Ortiz 2012). It restricts the elements in a binary sense by comparing an energetic formulation to energy release threshold: the elements can be either intact (in this case material behaviour is elastic) or eroded (in this case elements have no load bearing capacity). Stochino et al. (2017) separated the energetic formulation into compression and degraded tension portion so that the elements along cracked regions can be separated after losing bearing capacities. The resulting crack propagates within eroded elements with high energy. We adopt the Eigenerosion, which is a variational formulation of Griffth fracture theory (Pandolfi and Ortiz 2012). It approximates the Griffith theory using an integral form of adjacent structure. The released energy at crack surfaces is defined as:

$$\int_{\Gamma} G_p dX \approx G_p \frac{|C_\epsilon|}{2\epsilon} \tag{6.9}$$

where Γ is crack surfaces, G_p is the energy release rate representing the energy required to create a new fracture surface per a unit area of the material, C_ϵ is the ϵ's neighbourhood field, and $|C_\epsilon|$ denotes the volume of C_ϵ.

We assume grid cells in the MPM can be virtually eroded (in reality, the eroded grid cells are highlighted as PD patch) when its energy exceeds the given critical released energy. Motivated by above studies, we choose the Eulerian background grid to compute the released energy formulation.

The MPM grid nodes collect released energy through transferring the particle Cauchy stress σ_p^{MPM} to grid nodes. Discretizing Eq. (6.9) forms a

Eulerian description of particle released energy distribution in Eq. (6.10).

$$G_i = \frac{l_0}{m_i} \sum_p \omega_{ip}^{MPM} m_p \Phi(\sigma_p^+) \tag{6.10}$$

where $\sigma^+ = \sum_{i=0}^{d} \sigma_i n_i \otimes n_i$, σ_i and n_i are the i^{th} eigenvalue and eigenvector of the stress tensor σ_p^{MPM} respectively. $\Phi(\sigma^+) = \sigma^+ : \sigma^+$. ω_{ip}^{MPM} is the quadratic B-spline basis function representing interpolation weight between particle p and MPM grid node i. The Eulerian description of the released energy on the grid obviates the spurious energy carried on some material particles.

A grid cell with high energy, potentially containing crack lines and crack tips, are transferred into a PD patch (i.e. material particles within these grid cells are transited into PD particles). As cracks propagate and branch, they must be considered to be irreversible. In our study, the transition of a particle from the MPM to the PD is monotonically. The grid cells containing PD particles always have the highest released energy. Inspired by Wolper et al. (2019), we simply introduce a history dependent variable, $EE_p = max(EE_p, \Phi(\sigma^+))$, to replace $\Phi(\sigma^+)$ in Eq. (6.10).

6.3 Method Outline

As discussed in Chapter 6.2.1, we apply the MLS shape function to discretize Eq. (6.4), leading Eq. (6.4) converge to a standard MLS-MPM transfer scheme. SPB-MPM's core augmentation of the standard MPM is adding PD evolution within G2P step as a post process for efficiently animating cracks. Here we describe the MLS-MPM transfer procedures (Hu et al. 2018) and highlight our PD augmentation at the G2P step. n and $n+1$ symbolise quantities at timestep t^n and t^{n+1}, respectively. $\Delta t = t^{n+1} - t^n$.

6.3.1 Full Method

The data flow and relations of the main modules below are illustrated in Fig. 6.3.

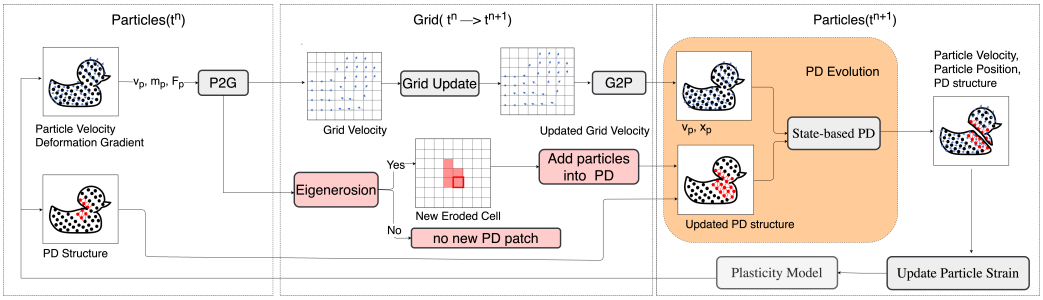

Figure 6.3: SPB-MPM data flow illustration. We elucidate the data flow in the MLS-MPM. Our augmentation is highlighted by red colour (Eigenerosion on the grid) and orange colour (PD method).

Particles to grid(P2G) Mass and momentum are transferred to grid nodes using the MLS-MPM (Hu et al. 2018, Jiang et al. 2015): $m_i = \sum_p \omega_{ip}^{MPM} m_p$, $mv_i^n = \sum_p \omega_{ip}^{MPM} m_p (v_p^n + C_p^n (x_i^n - x_p^n)$, where C_p is affine matrix of particle x_p. Compute nodal released energy by Eq. (6.10).

Grid Update The internal force f_i^n of grid node is computed using the MLS-MPM Hu et al. (2018): $f_i^n = -\sum_p \omega_{ip}^{MPM} V_p^0 M_p^{-1} \frac{\partial \psi}{\partial F}(F_p^{MPM^n})$ $(F_p^{MPM^n})^T (x_i^n - x_p^n)$, where $M_p^{-1} = 4/\Delta x^2$ for a quadratic particle-grid kernel. Identify the Eigenerosion Enhanced cell (EEc) by the fracture rule (Schmidt et al. 2009): if $G_i \geq G_p$, then $EE_i = 1$. Particles within EEc are added into Ω^{PD}.

Grid to particles(G2P) Update particle velocity: $\hat{v}_p^{n+1} = \sum_i \omega_{ip}^{MPM} v_i^{n+1}$. Then compute PD particle forces, evaluate bond connection status, and correct position using explicit time integration (details discussed in Chapter 6.3.2).

Particles strain update Grid velocity v_i^{n+1} needs to be transferred again due to PD evolution which is vital to accurately compute the particle velocity gradient: $C_p^{n+1} = \sum_i \omega_{ip}^{MPM} M_p^{-1} v_i^{n+1} (x_i^n - x_p^n)^T$. The material point deformation gradient is calculated as: $F_p^{MPM^{n+1}} = (I + \Delta t C_p^{n+1}) F_p^{MPM^n}$. Modify the deformation gradient if a plastic model is applied.

6.3.2 Peridynamics Evolution

At the step **Grid to particles(G2P)**, after the MPM obtains the particle velocity $\hat{v}_p^{n+1}$ at time t^{n+1}, PD is brought to correct particles' velocity and position by Eq. 6.6. Note that hat notation indicates an intermediate data, it will be corrected to v_p^{n+1} by the PD. Due to all PD forces are computed to correct $\hat{v}_p^{n+1}$ to v_p^{n+1}, we use $\hat{v}_p$ and v_p and omit the time superscript. Algorithm 3 describes the PD evolution in the SPB-MPM, including computing bond forces and updating bond connecting status.

Algorithm 3 PD Evolution

Initialise neighbourhood H_p using k-d tree
Run MPM step until we have updated particle velocity $\hat{v}_p^{n+1}$
for each PD particle p **do**
    for each active particle q **do**
        $X = x_q^0 - x_p^0$
        $Y = \hat{x}_q^{n+1} - \hat{x}_p^{n+1}$
        $U = \hat{u}_q^{n+1} - \hat{u}_p^{n+1}$
        $\omega^{PD}(X) = \frac{|X|}{l_0}$
        $K(x_p) += \omega^{PD}(X) X \otimes X V_q$
        $F_p^{PD} += \omega^{PD}(X) Y \otimes X \cdot K(x_p) V_q$
        $\epsilon_p^P += \frac{1}{2}\omega^{PD}(X)(U \otimes X + X \otimes U) \cdot K(x_p) V_q$
    end for
end for
for each PD particle p **do**
    for each active particle q **do**
        $\sigma_p^{PD} = p(V_q, e_q) I + 2\mu(F_p^{PD} - pI\ )$
        $T[x_p, t^{n+1}] < x_q - x_p >= \omega^{PD}(X) det(F_p^{PD}) \cdot (F_p^{PD})^{-1} \cdot \sigma_p^{PD} \cdot K(x_p) \cdot X$
        $T[x_p, t^{n+1}] += T[x_p, t^{n+1}] < x_q - x_p >$
    end for
end for
$v_p^{n+1} = \hat{v}_p^{n+1} + \frac{T[x_p, t^{n+1}]}{m_p} dt \quad x_p^{n+1} = \hat{x}_p^{n+1} + v_p^{n+1} dt$
for each PD particle p **do**
    for each active particle q **do**
        $\mu(X) = (T[x_p, t^{n+1}] < x_q - x_p > - T[x_q, t^{n+1}] < x_p - x_p >) \cdot U$
        if $\mu(X) > \mu_c$ **then** $\omega^{PD}(X) = 0$
end for
end for

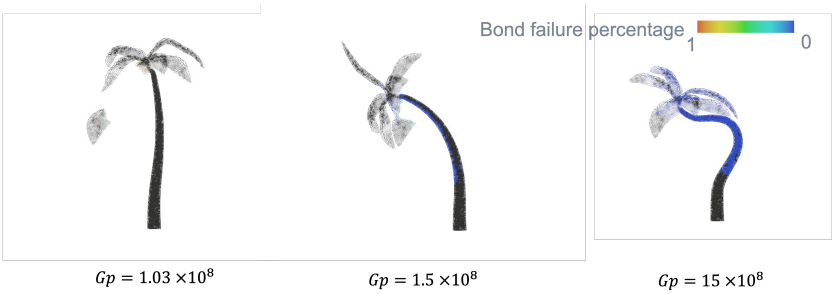

Figure 6.4: PD failure criterion. Stretching one leave away from a palm tree (with different energy release rate G_p), illustrating the controllable fracture effects with visualisation of PD particle broken bond percentage.

Various formulations of breakage criterion lead to rich, artistically controllable fractures. The breakage criterion of PD could be simply reached when bond elongation exceeds a prescribed critical value (Levine et al. 2014) (introduced in Chapter 4.3) or combined with a damage level parameter of PD particles to offer artists brittle fractures (Chen et al. 2018). Foster et al. (2011) proposed a bond failure criterion relating the critical energy density to the energy release rate. In this study, we adopt the energy-based failure criterion (Sun et al. 2019) and form a bond energy formulation by combining the bond force and particle's relative displacement vector $U = Y - X$ in Eq. (6.11). Furthermore, our SPB-MPM model can combine with all above failure laws (including the Eq. 4.8 in Chapter 4.3) with adjustable parameters for controllable fracture effects.

$$W(|X|) = (T[x_p, t^n] < x_q - x_p > -T[x_q, t^n] < x_p - x_q >) \cdot U \quad (6.11)$$

where $W(|X|)$ is the released energy of bond X.

The critical released energy of a bond is given by:

$$W_c = \begin{cases} \frac{3G_p}{2tl_0{}^3} & \text{in 2D} \\ \frac{4G_p}{\pi l_0{}^4} & \text{in 3D} \end{cases} \quad (6.12)$$

where t is the surface thickness. Varying material energy release rate leads to different cracking effects, as shown in Fig. 6.4. The weight function of

bond X is computed as a history-dependent influence function:

$$\omega(|X|)^{PD} = \begin{cases} 0 & W(|X|) > W_c \\ \frac{|X|}{l_0} & W(|X|) \leq W_c \end{cases} \tag{6.13}$$

Once a bond is broken, it will not be taken into account in the calculation of the PD deformation gradient at the next simulation step. Through accumulating all broken bonds, this method allows cracks to initiate, coalesce and propagate within domain Ω^{PD} without prescribed external crack routes.

6.4 Simulation

We present various fracture examples demonstrating the efficacy of the SPB-MPM, showing our model is able to simulate brittle and ductile fracture patterns with high visual fidelity. All examples were run on an Intel Core E5-1650 CPU with 12 threads at 3.20 GHz. The material parameters and grid settings are organised in Table 6.2. We implement all examples using the Taichi programming language (Hu et al. 2018 2019) and complete surface reconstruction and rendering as post-process by the SideFX Houdini.

6.4.1 Results

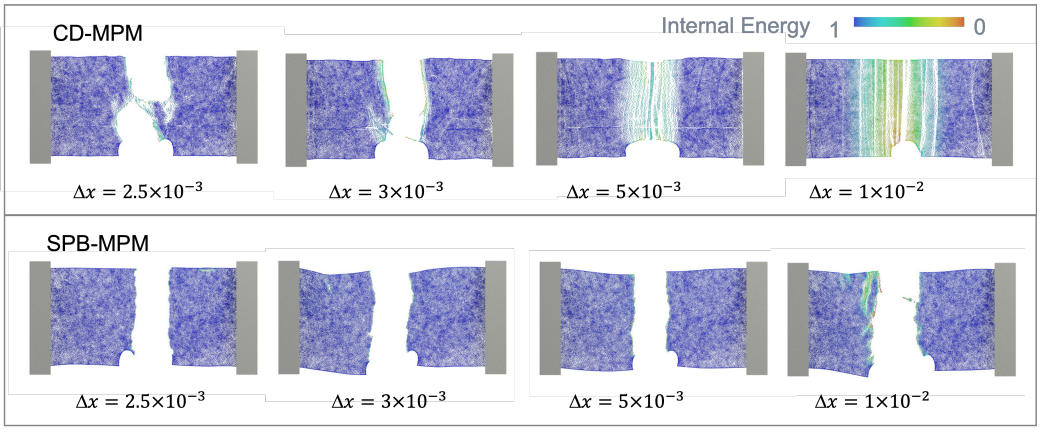

Figure 6.5: Stretched bars. We stretch bottom and top of a bar with different grid size; we compare against the CD-MPM to show that modelling fractures with the SPB-MPM is barely affected by grid settings.

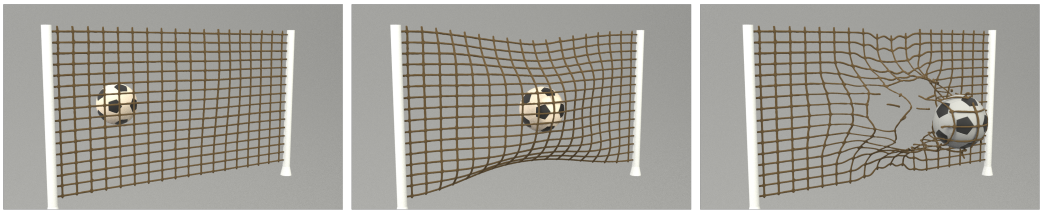

Figure 6.6: Throw a soccer ball to break the net. The ball is caught by the net by the MPM and breaks through the net by the SPB-MPM.

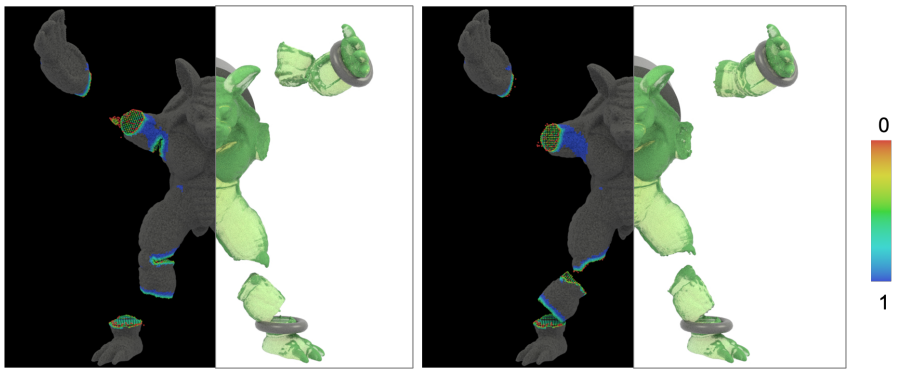

Figure 6.7: Armadillo PD particle visualisation. (Left) an elastic material simulated by the SPB-MPM. (Right) with the plastic model clamping the deformation gradient F_p, damage appears at the weakest region (knees and elbows). We visualise the percentage of active bonds for each PD particle. Red represents a PD particle having lost all connected bonds. Blue represents a PD particle fully connected with all family particles. Grey particles are pure MPM material points.

6.4.2 Discussion

We present various fracture examples attainable through using the SPB-MPM. We'd like to start with some 2D demos. In Fig. 6.4, we explore fracture effects of material critical energy release rate Gp. It is difficult to tear leaves from the palm tree with the largest energy release rate Gp. This demonstrates that the SPB-MPM treats fracture in a physically plausible manner. We carry out a comparison experiment between the CD-MPM and the SPB-MPM in Fig. 6.5. The grid size Δx varies from 0.0025 to 0.01.

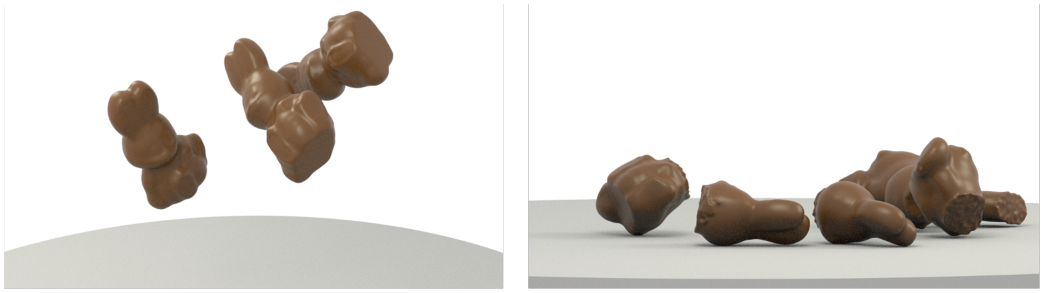

Figure 6.8: Falling chocolate bunnies. We throw chocolates to a porcelain plate to demonstrate the efficacy of our model in handling brittle fractures.

Table 6.2: Material parameters and grid settings.

Example	Δx	$\Delta t/timestep$	N	ρ	E	ν	G_p, Hp	plastic model
Stretched bars	7×10^{-3}	5×10^{-6}	14K	1	1×10^4	0.2	$(1.2, 1.3, 1.5, 1.7, 3.0) \times 10^8, 100$	$\theta_c = 6 \times 10^{-1}, \theta_c = 4 \times 10^{-1}$
Armadillo	3×10^{-3}	1×10^{-6}	356K	1	8×10^3	0.3	$7 \times 10^9, 100$	
Chocolates	4×10^{-3}	5×10^{-7}	442K	1	5×10^5	0.2	$9 \times 10^8, 100$	$\theta_c = 4 \times 10^{-2}, \theta_c = 5.5 \times 10^{-3}$

For all Δx, the SPB-MPM obtains similar crack effects while the CD-MPM only works well at fine grid resolution because the SPB-MPM defines the fracture on the bond level and the CD-MPM is heavily dependent to the background grid. This experiment demonstrates that our method is able to achieve detailed crack surfaces without introducing large computation by fine grid settings and further provides evidence for the SPB-MPM's superior ability.

The SPB-MPM is also capable of producing arresting fracture effects in 3D examples. In Fig. 6.1, we stretch the limbs of a jello armadillo and compare the SPB-MPM against the MLS-MPM and the state-based PD. The SPB-MPM obtains realistic fractures around joints (i.e. knees and elbows). Conversely, the MLS-MPM does not present any fractures even under large deformation. We present the same simulation in Fig. 6.7 with a visualisation of PD particles' active bond percentage. Fig.6.6 presents a net is broken through by a soccer ball by the SPB-MPM (right) and the comparison unbroken effects by the MPM (middle). Our method is also able to animate brittle fractures. In Fig. 6.8, we throw several chocolates against a porcelain plate. The chocolates are broken into pieces with brittle fracturing after bouncing on the plate several times. Furthermore, in our work, the energy

release rate used in the Eigenerosion should theoretically be the same as in PD bond's breakage criterion. However, we allow these two quantities to be set individually allowing for more diverse artistic control of the fracture patterns.

Introducing the PD brings extra computation over MPM scheme. Moreover, the PD evolution is heavily dependant on PD particles evolved in simulation. We break a single simulation frame of SPB-MPM into its subroutines in Fig. 6.9. This demo is performed in Fig. 6.6. Adding 14.4% computation time by the PD to the existing MPM computation is an efficient augmentation to the current MPM framework.

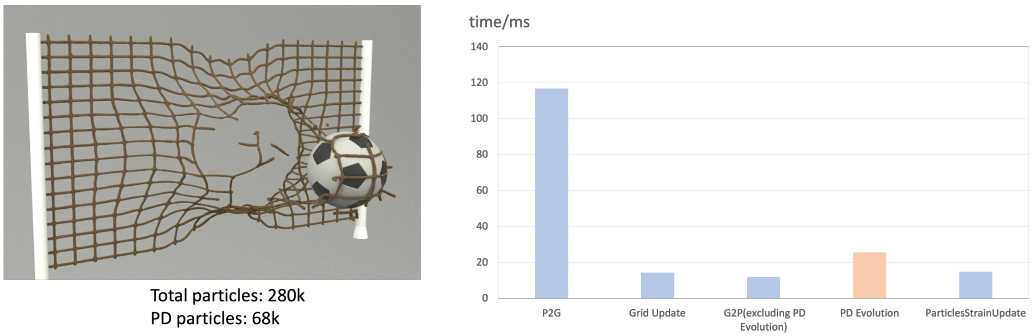

Figure 6.9: We decompose the timing consumed by each step: P2G, Grid Update, G2P(excluding PD Evolution), PD Evolution and Particles Strain Update performed on a demo shown in Fig. 6.6. The PD Evolution is always dependant on PD particle number in the simulation.

6.5 Conclusion

In this chapter, we present a novel MPM framework which superposes PD patches on the MPM as a local solution for handling fractures, leading to a hybrid MPM method: SPB-MPM. The SPB-MPM provides an intuitive solution of handling fractures, where bonds are broken directly relating to damage, offering physically plausible and stable topological deformations. Several various fracture examples has been tested to demonstrate the efficacy of the SPB-MPM, showing this method is able to simulate brittle and ductile fracture patterns with high visual fidelity. Giving a low overhead

computation over the current MPM scheme, the SPB-MPM is regarded as an efficient strategy to address the shortcomings of the existing MPM in simulating fractures.

Chapter 7

Modelling Shockwave effects In High Speed Impact Scenarios

7.1 Introduction

Shock wave is usually produced by impacts and explosions. It carries extreme energy and can propagate through any elastic medium such as air, water, or a solid substance. Zel'Dovich and Raizer (2002) and Zhang (2016) characterised the shockwave as an abrupt, nearly discontinuous, violent change in pressure, temperature, and density of the medium. Forces carrying shock wave energy usually have greater effects than smaller forces applied over a proportionally longer time period, thus the travel of shock waves leads to devastative deformation and a tremendous amount of debris acting like a small, local explosion ahead of the impact area.

In high speed scenarios, objects collided by a high speed impactor can generate intricate interactions in a short time period between surrounding materials. They present complex behaviours by the fact that objects are left with broken pieces followed by large amount of debris. In addition, the material properties can dramatically change during the sudden impact. In this case, the strain-stress relation applied under low velocity scenarios, such as Fixed Corotated elastic model for animation of snow dynamics (Stomakhin et al. 2013), Saint Venant–Kirchhoff model for granular materials (Klár et al. 2016) and stable Neo-Hookean model for hyperelastic materials of virtual

characters (Smith et al. 2018), are not able to describe above phenomena in high speed impact.

Shockwave modelling has been recognised as a common physical phenomenon in many applications of impact engineering, such as high explosive detonation, high speed impact and large material expansion. There exist many approaches in mechanics to model high velocity impacts in a way that encodes shock waves and relative hardening effects in the material response at high strain rate. Ma et al. (2009a) incorporated several Equations of State (EOS) with an adaptive MPM to capture the volumetric response, strain rate hardening effect and thermal softening effects in the application of high explosive scenarios. Meshless methods with Lagrangian points also showed success to model such behaviours with the growing versatility and popularity of PD (Silling et al. 2017). In this work, the PD has been adapted to a Eulerian form to simulate shock waves with a thermodynamic theory for modelling bird strikes. Zhang et al. (2006) replaced the pressure term (calculated by the Mie-Grüenisen Equation of State) by a hydrostatic pressure in the constitutive model, demonstrating the shock wave effects in computer graphics for modelling smashed fragile materials, for example, vases falling on the floor.

Considering that in impact scenarios the debris cloud always appears with broken pieces, we need our model to be capable of presenting key features of shock wave impact while keeping stable shapes for the remaining bodies as broken chunks (see in Fig. 7.2)). Motivated by the decomposition of stress tensor and strategy of incorporating an Equation of State into the MPM Ma et al. (2009a), we introduce a strength model from thermodynamics theory (Silling et al. 2017) and adapt it with the stable Neo-Hookean model (Smith et al. 2018). This model relates the strain-stress relation to the high pressure produced by the rapid impact while obviating the instability caused by extreme volume response. Additionally, to model general metal behaviours under high velocity impact, we adopt a J2 plasticity model presented (Zhang et al. 2013) with an associative flow rule to compute the plastic strain increment for the Johnson-Cook model (Ma et al. 2009a). This avoids the traditional, tedious iterations in stress projection and is capable

of solving the plastic changes in an efficient, linear manner. Strain effects and strain rate hardening effects are therefore considered in our model. Our method is able to simulate the characteristic impact behaviours of organic fruits, rigid bodies, and metal materials.

7.2 Constitutive Models

As described in Chapter 7.1, modelling shock wave effects in high speed impact scenarios is notoriously challenging. We expect one method to be capable of robustly treating broken chunks while also capturing abrupt features following rapid impacts. The stable Neo-Hookean model, designed as an non-linear hyperelastic energy function, demonstrates great success for modelling fleshy appearance of virtual characters (Smith et al. 2018). It exhibits superior volume preservation, and is robust to extreme kinematic rotations and inversions. As such, we choose the stable Neo-Hookean as our basis model and replace the pressure term with our shock wave pressure. In the following, we first give a brief description of the stable Neo-Hookean model for deformable objects, and then introduce our shock wave model.

7.2.1 Stable Neo-Hookean Model

We introduce a Neo-Hookean model in the preceding discussion in Chapter 3. The stable Neo-Hookean model has been proposed as a stable version of the classic Neo-Hookean model. It retains the rich, volumetric character appearance of the classic Neo-Hookean model but does not need any filter operations. This method starts from the comprehensive consideration about four perspectives (inversion stability, reflection stability, rest stability and meta-stability under degeneracy) and provides a stable hyperelastic energy function. In this section, we start by examining how the stable Neo-Hookean modified above four properties based on the Neo-Hookean model, and then derive its energy function and stress tensor.

The Neo-Hookean model is: $\Psi_{Neo} = \frac{\mu}{2}(I_C - 3) - \mu \log J_p + \frac{\lambda}{2}(\log J_p)^2$. Based on the Valanis-Landel hypothesis (Xu et al. 2015), many hyperelastic energies can be separated into length (1D), area (2D), and volume (3D)

components. As such, $\Psi_{Neo,length} = \frac{\mu}{2}(I_C - 3)$, $\Psi_{Neo,volume} = -\mu \log J_p + \frac{\lambda}{2}(\log J_p)^2$.

Inversion stability The current version of $\Psi_{Neo,length}$ is well behaved under inversion. The volume term from $\Psi_{Neo,volume}$ clearly presents numerical difficulties when $J_p < 0$. Martin et al. (2011) migrated this issue by using $\Psi_{Neo,volume} = \frac{\lambda}{2}(J_p - 1)^2$ which is bounded, well defined and invertible. The stable Neo-Hookean adopted this augmentation to sidestep the need for any inversion handling.

Rest stability Bonet and Wood (1997) mentioned that the hyperelastic energy should vanish at identity in the solid mechanics literature. This theory restricts $\Psi_{Neo,volume}$. In practice, the PK1 stress tensor resolves to zero. To recover rest stability, the stable Neo-Hookean modifies $\Psi_{Neo,volume} = \frac{\lambda}{2}(J_p - 1)^2$ to shift the root from 1 to α as $\Psi_{Neo,volume} = \frac{\lambda}{2}(J_p - \alpha)^2$, where $\alpha = 1 + \frac{\mu}{\lambda} - \frac{\mu\lambda}{4}$.

Meta-Stability under Degeneracy According to the Drucker stability analysis (Bower 2009), the energy function needs be examined under degeneracy. It requires the method to remain valid when an element has been crushed to a plane, line, or point. For the undesired numerical errors in degenerating to both lines and points, the stable Neo-Hookean adds a regularised origin barrier $log(I_C + 1)$ that inserts a peak of negative definiteness in line cases while through adding constant 1 to smooth away the logarithmic singularity.

Reflection stability The energy is well behaved regardless of the reflection convention used in the SVD.

With above observation, the final stable Neo-Hookean energy is:

$$\Psi_{Neo} = \frac{\mu}{2}(I_C - 3) + \frac{\lambda}{2}(J_p - \alpha)^2 - \frac{\mu}{2}log(I_C + 1) \tag{7.1}$$

The first Piola-Kirchhoff stress tensor is:

$$P = \mu \left(1 - \frac{1}{I_C + 1}\right) F_p + \lambda(J_p - \alpha)\frac{\partial J_p}{\partial F_p} \tag{7.2}$$

Notably, Smith et al. (2018) phrased the Fixed Corotated model as an linearization of the Neo-Hookean energy and positioned the stable Neo-Hookean

model on a spectrum of successive approximations to the Neo-Hookean energy. The generality analyses and the energy decomposition rules inspire us to investigate a shock wave model within the volume portion of the stable Neo-Hookean energy.

7.2.2 Shockwave Effects Model

The stress tensor arises from the energy density as: $\sigma = \frac{1}{J_p}\frac{\partial \psi}{\partial F_p}(F_p){F_p}^T = \frac{1}{J_p}(\frac{\partial \psi_D}{\partial F_p}(F_p) + \frac{\partial \psi_{volume}}{\partial F_p}(F_p)){F_p}^T$ in the MLS-MPM, where $\psi(F_p^n)$ is the chosen energy density function, ψ_D and ψ_{volume} are shearing and volumetric portion of $\psi(F_p)$, σ is the Cauchy stress tensor of particle p. We decompose the stable Neo-Hookean model as $\psi_D = \frac{\mu}{2}(I_C - 3) - \frac{\mu}{2}log(I_C + 1)$, $\psi_{volume} = \frac{\lambda}{2}(J_p - \alpha)^2$. This decomposition is based on the linearization analysis in Smith et al. (2018). The Cauchy stress tensor is thus obtained by transforming the first Piola-Kirchoff stress using $\sigma = J_p^{-1} P F_p^T$ and Eq. 7.2. Intuitively, the Cauchy stress tensor can be regarded as the combination of shearing changes (deviatoric stress portion) and volume changes (pressure portion): $\sigma = \sigma_D + \sigma_P$.

To capture volumetric changes in objects following high speed impacts, we replace the pressure term σ_P in the Cauchy stress tensor to a hydrostatic pressure term as $\sigma_P = -p(V_p^n, e_p^n)I$ from an EOS (further discussion in Chapter 7.2.3) where I is an identity matrix and the pressure $p(V_p^n, e_p^n)$ is a scalar quantity. V_p^n denotes current particle volume and e_p^n depicts the current particle internal energy density.

The shock wave stress tensor can be fitted into both standard MPM and our SPB-MPM. In SPB-MPM, a deformable object contains two domains: Ω^{MPM} and Ω^{PD}. Intuitively two constitutive models are necessary for two dynamic lines, however, we choose the above adaptive constitutive model for both Ω^{MPM} and Ω^{PD} (the strain tensor is either from the deformation gradient F_p^{MPM} in the MPM or F_p^{PD} in the state-based PD. then the stress σ_p^{MPM} for MPM and σ_p^{PD} can be obtained from our shock wave effects model). We adopt one constitutive model for two dynamic lines for two reasons: 1) the two configurations ($w^{PD} \in \Omega^{MPM}$) have the exact same material

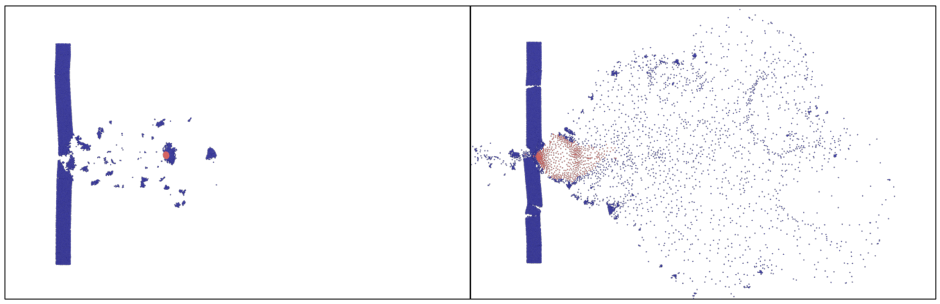

Figure 7.1: Sphere impactor (red) collides with 2D brittle wall (blue). The debris cloud attainable through our shock wave modelling presents similar effects in Watson and Steinhauser (2017).

properties. 2) the state-based PD is capable of using classical constitutive material models. We just compute $p(V_p^n, e_p^n)$ in the MPM once and reuse it for the state-based PD. The internal energy density e_p^n is formulated through integrating the energy changes over time from the initial configuration with e_p^0 for each material point p by:

$$\begin{aligned} C_{pavg} &= \frac{1}{2}(C_p + C_p^T) \\ \Delta e_p^n &= \Delta t C_{p_{avg}} : \sigma_p^n \end{aligned} \tag{7.3}$$

7.2.3 Mie-Grüenisen Equation of State

The Mie-Grüenisen EOS is used to describe material volumetric changes under high pressure condition in a shock-compressed solid at a given temperature (Lemons and Lund 1999). Silling et al. (2017) proposed a thermodynamics form of PD which also adopts the Mie-Grüenisen EOS for modelling fluid-like shock waves for metals in a Eulerian manner. This work offers us a good opportunity to describe the volume response by determining the pressure in impact condition.

In this paper, experiments are set under an isothermal condition which is free of temperature changes. We adopt a simplified variation of the Mie-Grüenisen model (Silling et al. 2017) (in Eq. (7.4)). We define a scalar

Figure 7.2: A multi-material cube with a stiff shell is shot with an elastic bullet. We show the different impact behaviours through comparing the Fixed Corotated elastic model with the shockwave effects model handled by our shockwave effects model (different bulk wave speed c_0) are shown.

quantity $\zeta_p = \frac{\rho_p}{\rho_p^0} - 1 = \frac{V_p^0}{V_p^n} - 1 = \frac{1}{J_p} - 1$ to depict relative particle density changes, where $J_p = det(F_p^{MPM})$. The linear dependency between shockwave velocity v_s and particle velocity v_p is defined as $v_s = c_0 + Sv_p$. S is a constant, usually set up as 1.1 and c_0 is the bulk wave speed. Pressure $p(V_p^n, e_p^n)$ is updated by:

$$p(V_p^n, e) = \begin{cases} \frac{\rho_0 c_0^2 \zeta_p [1+(1-\frac{1}{2}\gamma)\zeta_p]}{[1+(S-1)\zeta_p]^2} + \gamma(e_p^n - e_p^0) & \zeta_p \geq 0 \\ \rho_0 c_0^2 \zeta_p + \gamma(e_p^n - e_p^0) & \zeta_p < 0 \end{cases} \tag{7.4}$$

where $\gamma = 1.6$ is the Mie-Grüenisen parameter. Fig. 7.2 visualises the influence of bulk wave speed c_0 on impact behaviours in the second and third rows.

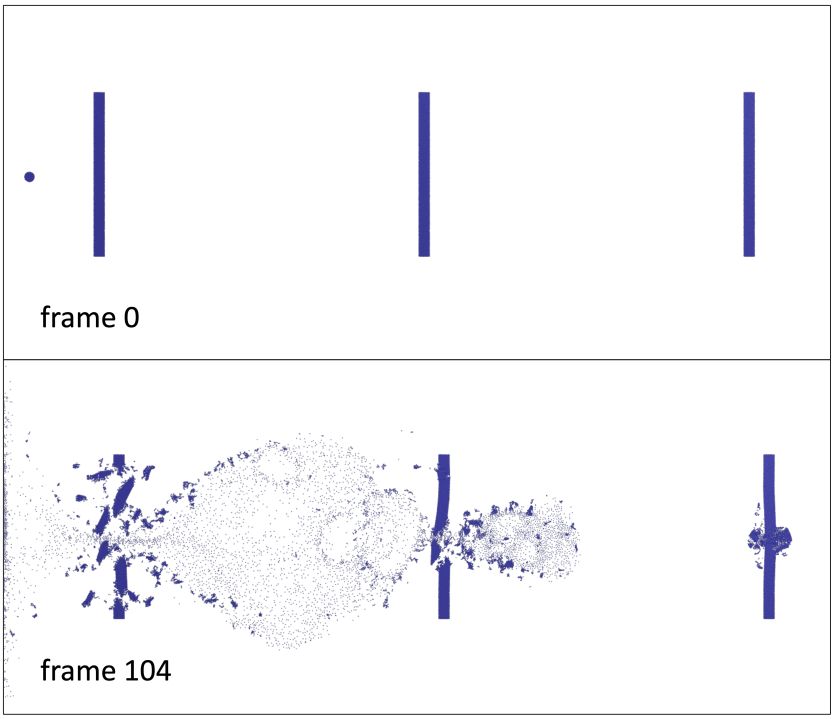

Figure 7.3: Shoot a bullet to multiple walls. Different impact velocities create different level of destruction.

7.3 Plasticity

Metals are traditionally produced not to easily exceed the yield strength in any operational conditions. However, in high speed impact scenarios, such as car crashes and metal boards being shot, modelling the behaviours of metals need take plastic deformation and damage into account. Most plastic models in the literature are only suitable in common life scenarios which appears at low strain rate during deformation, such as the snow plasticity model based on the decomposition of deformation gradient Stomakhin et al. (2013), Drucker-Prager model for sand simulation (Klár et al. 2016), and a temperature incorporated plasticity model for baking (Ding et al. 2019). All these models formulate the yield criterion based on the stress tensor. Conversely, the yield criterion of a metal at large strain rate is irrelevant to the pressure term in stress tensor (Zhang et al. 2013, Wilson 2002). To obtain metallic plastic deformation in a physically plausible manner, Zhang et al.

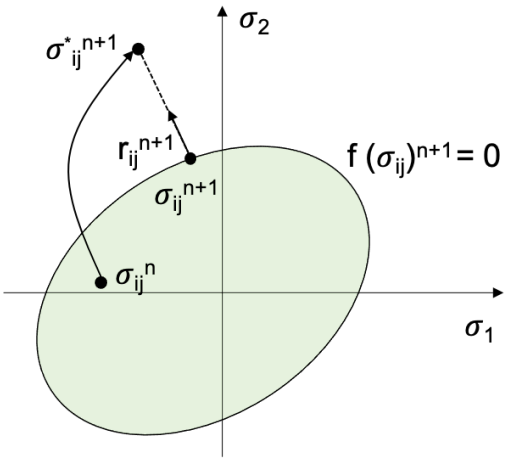

Figure 7.4: J2 plasticity radial return mapping process with von-Mises yield criterion. We visualise the predictor-corrector process by a radial return mapping method. After MPM updates particle strain at timestep t^{n+1}, the predicted deviatoric stress tensor $\sigma_{ij}^{n+1^*}$ violates the yield criterion. We project it back to the yield surface as σ_{ij}^{n+1} with an associative flow rule (the plastic strain increment vector r_{ij} is perpendicular to the yield surface).

(2013) concluded that a J2 flow rule with respect to the deviatoric portion of stress tensor is necessary. In this case, the plastic increment is independent to the volumetric portion of the stress tensor. With these considerations, we adopt the J2 flow rule as the return mapping algorithm in our metal plastic model.

In mechanics, researchers always design plasticity models based on true experimental data in order to give an accurate and precise estimation for plastic flow of the studied metal materials. Each type of metal material has unique properties under given conditions. Our aim is to obtain controllable visual effects of deformable objects in the computer graphics world rather than the accurate mathematical calculations. Therefore, our model ignores sophisticated examinations of material properties, and we instead concentrate on deriving an empirical model based on general phenomenological observations. Even so, our plasticity model is based on theory and models devised for engineering applications. We adopt a simplified Johnson-Cook metal plasticity model with J2 plasticity flow rule in this section to accom-

modate plastic deformation for metals in high speed impact scenarios. This model allows us to remove plastic deformation from the deviatoric portion of the stress tensor.

Notably, when apply this metallic model to SPB-MPM in which PD is incorporated for discontinuities in some local regions, we process the plasticity at the **PartilceStrainUpdate** step. We do not store plasticity for bonds for the SPB-MPM for simplicity and ease of implementation, though the Chapter 5 proves that the PD is able to process plasticity as part of bond extension.

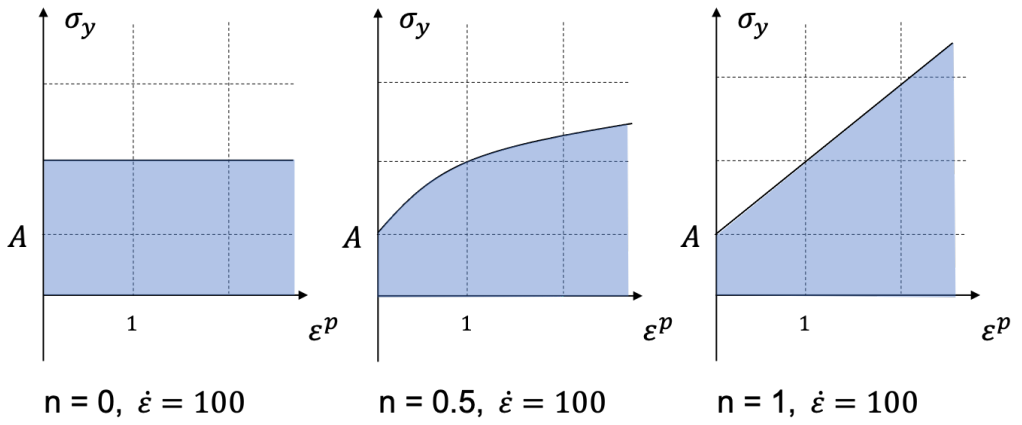

Figure 7.5: Johnson-Cook plasticity model with strain hardening exponent n. We visualise the yield surface with different choices of strain hardening exponent under the same strain rate. When n is set to 0, the yield criterion becomes a von-Mises yield criterion (left).

7.3.1 Flow rule

The fundamental features of a plasticity model are the yield function and flow rules. Contrasting to the common multiplicative plasticity rule which decomposes plastic deformation from the deformation gradient F_p (Stomakhin et al. 2013, Ding et al. 2019), we adopt an additive plasticity decomposition based on the strain rate by $\dot{\epsilon} = \dot{\epsilon^e} + \dot{\epsilon^p}$, where $\dot{\epsilon}$, $\dot{\epsilon^e}$ and $\dot{\epsilon^p}$ are strain rate, elastic portion of strain rate and plastic portion of strain rate, respectively. The flow

Figure 7.6: Falling containers. The breadth of metal collision behaviours attainable with the Johnson-Cook plastic model is illustrated. We demonstrate metallic deformation with various strain hardening exponents n corresponding to Fig. 7.5.

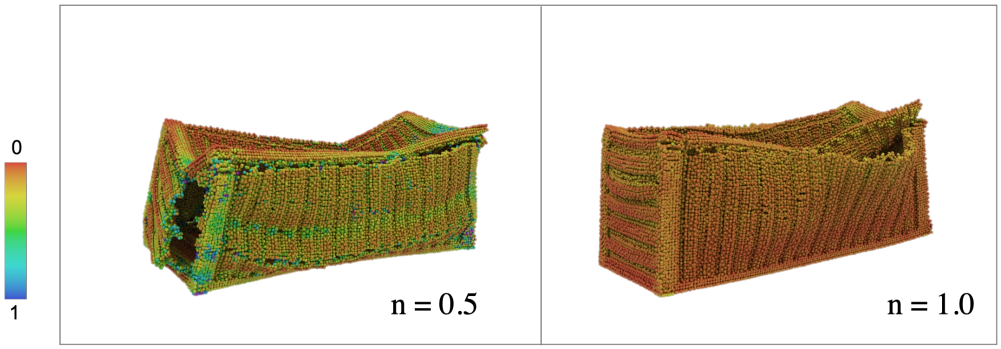

Figure 7.7: We visualise the accumulated equivalent plastic strain of falling containers for each particle in Fig. 7.6.

rule defines the relationship between the plastic strain increment and deviatoric stress increment. We present the yield condition $f(s, \epsilon^p) = s - \sigma_y(\epsilon^p) \leq 0$ as a function of equivalent stress $s = \sqrt{3J_2} = \sqrt{\frac{3}{2}\sigma_D^{MPM}\sigma_D^{MPM}}$ (proportional to magnitude of second invariant of the deviatoric stress) and equivalent plastic strain ϵ^p from the additive decomposition of strain tensor rate. Here σ_D^{MPM} is the deviatoric portion of the stress tensor predicted by our shock wave constitutive model (further details given in Chapter 7.3.2). $\sigma_y(\epsilon^p)$ is the yield stress function with respect to ϵ^p. When a prediction gives $f(s, \epsilon^p) \geq 0$, the predicted strain contains a plastic strain increment ($\Delta\epsilon^p \geq 0$) and a following projection is required to yield $f(s, \epsilon^p)$ to 0, otherwise the strain tensor still stays within elastic regime. The J2 flow model is able to calculate the plastic stress increment linearly by assuming that the corrected deviatoric

stress is perpendicular to the yield surface (illustrated in Fig. 7.4). The linearization of the yield condition is explained in Eq. (7.5) and the approximation of the plastic strain increment is described in Eq. (7.6). The whole process can be found in Appendix B.

$$f(s,\epsilon^p) + \frac{\partial f(s,\epsilon^p)}{\partial \sigma_{ij}}\Delta\sigma_{ij} + \frac{\partial f(s,\epsilon^p)}{\partial \epsilon^p}\Delta\epsilon^p = 0 \tag{7.5}$$

Here $\frac{\partial f(s,\epsilon^p)}{\partial \sigma_{ij}^{MPM}} = r_{ij}$ where r_{ij} is the direction of plastic flow, $\Delta\sigma_{ij}^{MPM}$ represents the difference between predicted stress tensor and corrected stress tensor since $\Delta\sigma_{ij}^{MPM} = \sigma_{ij}^{MPM*} - \sigma_{ij}^{*}$ and $\frac{\partial f(s,\epsilon^p)}{\partial \epsilon^p} = -\frac{\partial \sigma_y(\epsilon^p)}{\partial \epsilon^p} = -E^p$ where E^p is the plastic modulus. When E^p is set to 0, the yield function becomes a von-Mises yield criterion $f(s) = s - K^2$ with a constant K. In this case, the material presents a perfectly elastic-plastic response. E^p also can be a function of ϵ^p, $\dot{\epsilon}^p$ and temperature for hardening and softening effects. The plastic strain increments are updated at timestep t^{n+1} as:

$$\Delta\epsilon^p = \frac{s - \sigma_y(\epsilon^p)}{3\mu + E^p} \tag{7.6}$$

where μ is the same shearing modulus of *Lamé* parameters. It should be noted that all data flows are operated in the deviatoric portion of both elastic and plastic strains. Volume changes do not contain plastic deformation in J2 flow rules as $\Delta\epsilon_{dd}^p = 0$, where d is the problem dimension.

7.3.2 Johnson-Cook Model

Many materials exhibit a significant dependence of yield and flow phenomena on the rate. Studies of metallic behaviours indicate that the yield stress exhibits significant changes beyond a critical strain rate (Oxley 1963). Inspired by Zhang et al. (2016), we adopt the Johnson-Cook model as the yield stress. The Johnson-Cook model is a function of tensile flow stress, in accordance with strain hardening and strain-rate hardening scheme where the two components are able to work separately. This relates yield stress with plastic strain and plastic strain rate as in Eq. (7.7).

$$\sigma_y(\epsilon^p, \dot{\epsilon}^p) = (A + B(\epsilon^p)^n)(1 + Cln\frac{\dot{\epsilon}^p}{\dot{\epsilon}_0}) \tag{7.7}$$

where A is the yield stress at a reference temperature and strain rate, B and C are material constants representing the coefficients of strain hardening and strain rate hardening respectively. $\dot{\epsilon}_0$ is a reference equivalent plastic strain rate which we keep equal to 1.0 per second in all experiments to reduce the exploration complexity of parameter tuning. n is the strain hardening exponent. As claimed in Chapter 7.3.1, if we set $n = 0$, the yield function becomes a von-Mises type yield criterion with $E^p = 0, A = K^2$. Following Zhang et al. (2016), as the Johnson-Cook yield stress is an nonlinear function of ϵ^p, we compute yield stress derivatives with respect to ϵ^p_{eq} by Taylor series expansion and obtain:

$$E_p = \frac{\partial \sigma_y(\epsilon^p, \dot{\epsilon}^p)}{\partial \epsilon^p} = Bn(\epsilon^p)^{(n-1)}(1 + Cln\frac{\dot{\epsilon}^p}{\dot{\epsilon}_0}) \tag{7.8}$$

By integrating Eq. 7.8 to Eq. 7.6, we are able to update plastic strain tensor, and the rest of the strain tensor is elastic strain tensor.

7.4 Simulation Results

We now present our methods with a breadth of materials to demonstrate the efficacy of our constitutive models in comparison with other constitutive models. Most of our experiments are set with a high speed impactor. All examples were run on an Intel Core E5-1650 CPU with 12 threads at 3.20 GHz. The material parameters and grid settings are organised in Table 7.1. We implement all demos using the Taichi programming language (Hu et al. 2018 2019) and complete surface reconstruction and rendering as post-process by the SideFX Houdini.

7.4.1 Shock wave effects

We start the examinations of the shock wave effects model with some 2D demos. Fig. 7.1 depicts a sphere striking a wall in 2D, producing a debris cloud behind the wall and abrupt damage on the parts away from the impacted area while the stable Neo-Hookean hyperelastic model just presents general broken chunks. Our work shows the similar shattering patterns as in

Table 7.1: Material parameters and grid settings of Shock Wave Effects Model (Plasticity model: Drucker-Parger model (DP) and Johnson-Cook model (JC)).

Example	Δx	$\Delta t/timestep$	N	ρ	E	ν	G_p, Hp	c_0	plastic model
2D wall	2×10^{-4}	1.8×10^{-6}	39K	1	3×10^5	0.2		2092	$DP(\theta_c = 5 \times 10^{-2}$, $\theta_s = 1.5 \times 10^{-2})$
Gelatin cube	5×10^{-3}	6×10^{-6}	106K	1	5×10^2	0.2		2092	
Multiple walls	2×10^{-4}	1.8×10^{-6}	39K	1	3×10^5	0.2		2092	$DP(\theta_c = 5 \times 10^{-2}$, $\theta_s = 1.5 \times 10^{-2})$
Multi-material cube	7×10^{-3}	5×10^{-6}	47K	1	5×10^4	0.2		1092, 2092	$DP(\theta_c = 4 \times 10^{-2}$, $\theta_s = 1.5 \times 10^{-2})$
Watermelon	2.5×10^{-3}	2×10^{-7}	535K	1	$(4 \times 10^4$, $4 \times 10^3)$	0.3	$7 \times 10^{10}, 200$	2092	$DP(\theta_c = 4 \times 10^{-2}$, $\theta_s = 6 \times 10^{-3})$
Apple	2.5×10^{-3}	2×10^{-8}	188K	1	$(3 \times 10^6$, $1 \times 10^6)$	0.2	$3 \times 10^{15}, 200$	2092	$DP(\theta_c = 4 \times 10^{-2}$, $\theta_s = 6 \times 10^{-3})$
Container	2.8×10^{-3}	1×10^{-6}	76K	1	5×10^6	0.35		2092	$JC(A = 1557, B = 625$, $C = 1, n = 0, 0.5, 1)$
Cars	2.8×10^{-3}	1×10^{-6}	340K	1	5×10^5	0.2		2092	$JC(A = 1557, B = 625$, $C = 1, n = 0.5)$

a hyper velocity impact research (Watson and Steinhauser 2017). The shock wave effects model can also handle a wide range of impact speed. For example, in Fig. 7.3, we shoot a bullet to break through several walls. With the descending speed of the bullet, the impact creates different level of destruction. We further present key shockwave-impact features in Fig. 7.2 through comparing the Fixed Corotated elastic model against our shockwave effects model (with different bulk wave speed c_0) simulating a multi-material object. In the second and third rows, our model exhibits abrupt damage with shattering particles when a sphere collides with the multi-material cubes. Also notice the shock wave effects model with greater c_0 carries greater energy and produces more destructive damages. We visualise the wave transition for showing the shockwave directly. We conduct an experiment in which a metal bullet is shot at a piece of gelatin cube in Fig. 7.8. It achieves compatible visual effects as in a shooting experiments in Fig. 7.9. Compared to the stable Neo-Hookean model, our model can better capture the clear path caused by the bullet and the fluid-like bubble surface movement as a result of the shockwave.

It is feasible to integrate the shock wave effects model to the SPB-MPM

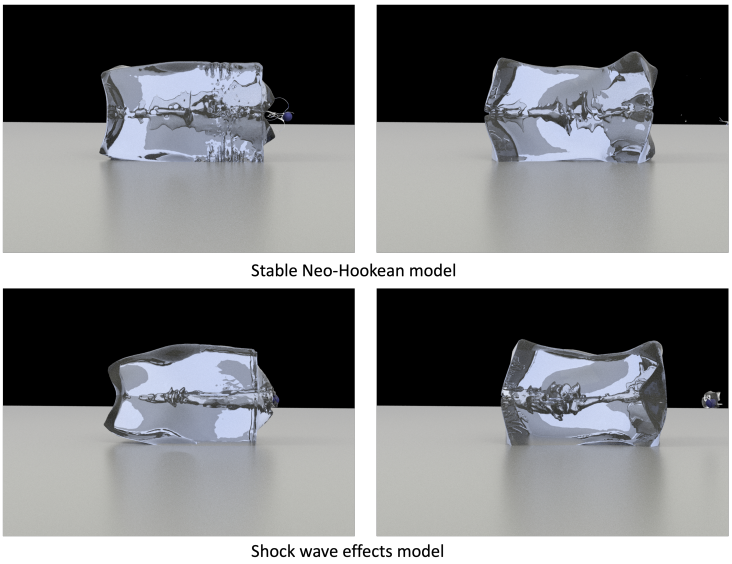

Figure 7.8: A metal bullet is shot at a piece of clear gelatin. The bullet passes through the cuboid. Compared to the stable Neo-Hookean hyperelastic model, our model is able to better capture the clear path with bubble following behind due to the huge impact force.

fracture scheme. The intuitive exploration of a breadth of organic and elastic material behaviours in impact scenes has been demonstrated in following experiments. We shoot a bullet to a multi-material watermelon (we set the rind as a stiff hyperelastic material) in Fig. 7.10 and an apple (the apple skin is treated with a greater Youngs modulus E) in Fig. 7.11. The watermelon is broken into pieces by impact forces, with juicy red flesh and seeds spraying in the air. Similar to the watermelon, the apple exhibits organic fractures and large amounts of juice droplets splashing from the damaged pulp with high visual fidelity. Note that for the organic fruit, multi-material cubes in Fig. 7.2 and walls, we adopt the finite-strain multiplicative plasticity law employing the Drucker-Prager plasticity model (Stomakhin et al. 2013).

7.4.2 Plasticity

For metallic deformation under high velocity impact, we simulate metals with our Johnson-Cook plastic scheme. Inspired by Ma et al. (2009a), the strain hardening exponent n is usually chosen $(0, 1)$ to depict metal's ability to resist

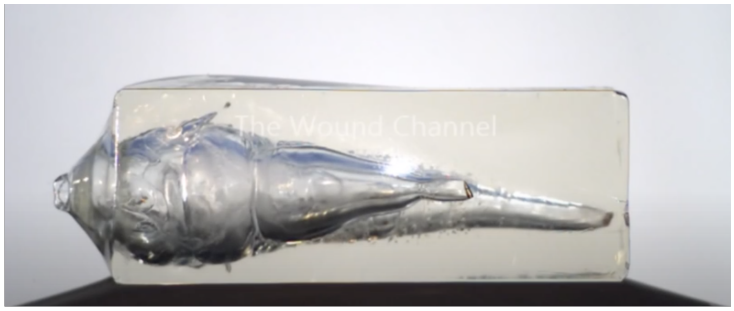

Figure 7.9: A reference picture in a shooting experiment that a bullet is fired into clear ballistic gelatin (Channel 2015).

plastic deformation. In Fig. 7.6, we test containers with various choices of n: $0, 0.5, 1$. For $n = 0.5$ (middle) and $n = 1$ (right), the container behaves like a stiffer material compared to the softer, more plastic result on the left side. This is due to the strain increase during collision, resulting in a greater yield stress in the yield criterion. For $n = 1$ the container exhibits the least plastic deformation. We visualise the equivalent plastic strain of all particles in Fig. 7.7. These plastic deformation distributions are consistent with our visual effects. This can be seen in $n = 0.5$ by matching the wrinkle on the side of the container in Fig. 7.6 to a blue wrinkle of high plastic deformation in Fig. 7.7. We also demonstrate a common car crash scene in Fig. 7.12 to further display the metallic deformation in impact scenario. In this case, the rubber wheels are only simulated by the shock wave effects model. We do not apply the SPB-MPM coupling framework to metal plasticity. Metal tearing is caused by plasticity.

7.5 Conclusion

In this chapter, we propose a shock wave effects model and a metallic plastic model to describe the prominent features of high velocity impact, which are unattainable through general constitutive models. To demonstrate the volumetric response in shock-compressed solids, we adopt the simplified Mie-Grüenisen EOS as a hydropressure term to replace the pressure term in the stable Neo-Hookean model. To model metallic behaviours at high strain

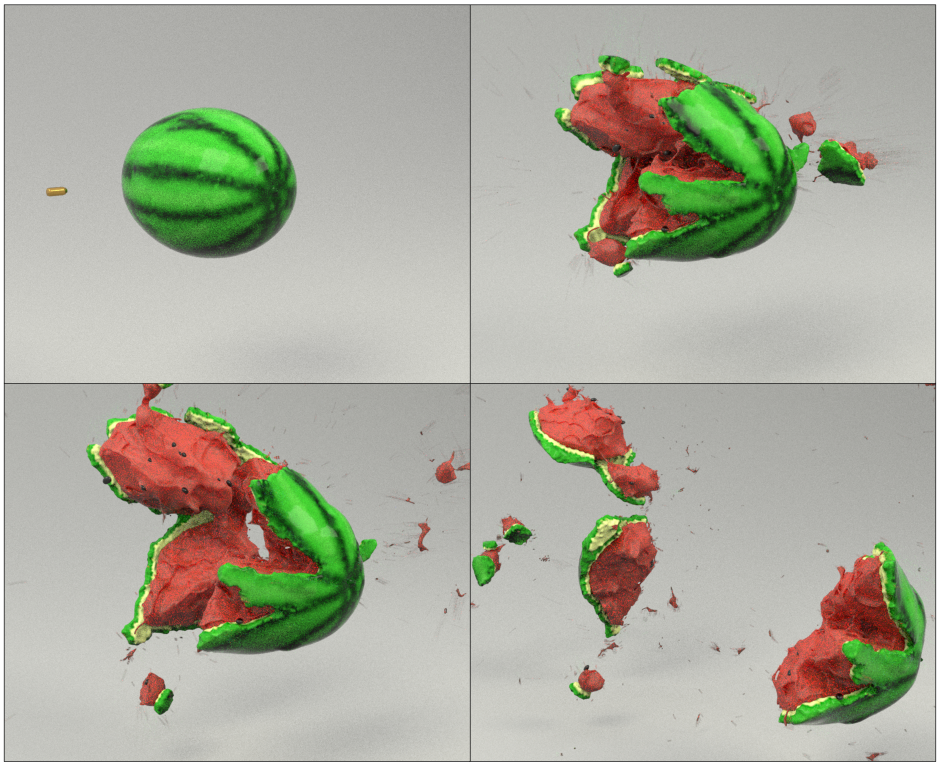

Figure 7.10: A metal bullet is shot at a fresh watermelon. The impact forces cause juicy pink flesh to explode and spray mist in the air, showcasing intricate fractures and realistic dynamics.

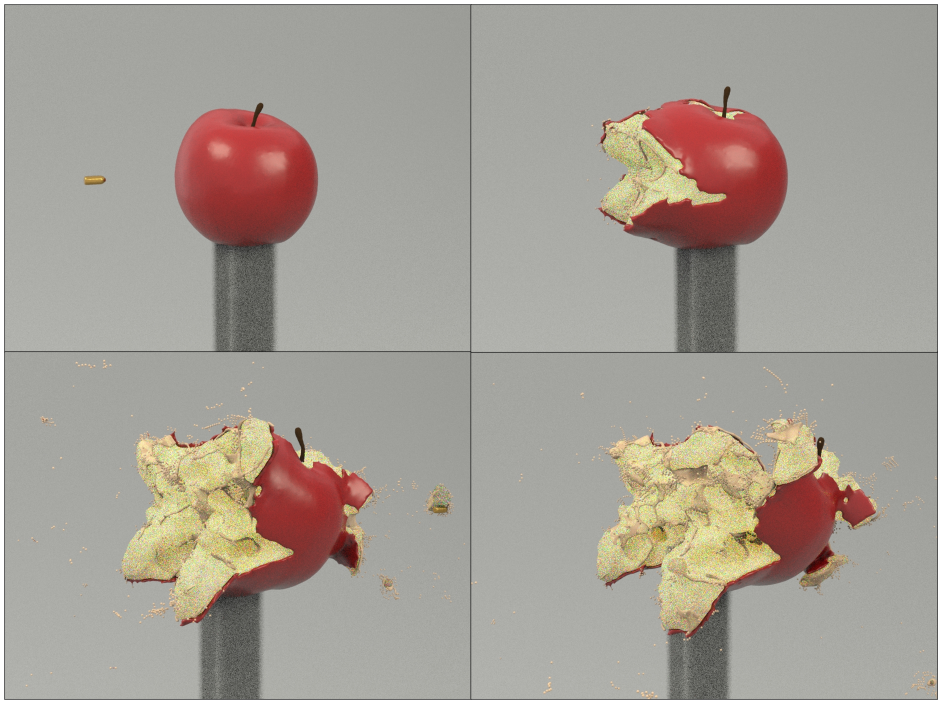

Figure 7.11: We shoot a bullet at an apple fixed on a tube, illustrating dynamic, organic fracture and debris spray using the shock wave effects model.

rate, we adopt the Johnson-Cook strain and strain-rate hardening scheme. We showcase a wide variety of demos to illustrate the compelling features of high velocity impact scenarios our model can reproduce, obtaining a breadth of fracture effects of organic fruits, soft bodies and multi-material deformable objects.

Figure 7.12: Car crash scene. The shock wave effects and the metal plastic model have been applied to metallic materials.

Chapter 8

Conclusion and Future Work

In this chapter, we conclude the thesis by summarising our works. We also discuss the possible future work directions to resolve current limitations.

8.1 Conclusion

The simulation of elastoplastic materials undergoing large deformation and topology change is a popular topic in Computer Graphics, Mechanical Engineering, and other fields. Modelling impact behaviours is one of their most important applications. In this situation, how to cleverly represent material geometry topological changes as well as stably animate fractures with arbitrary crack patterns remains a challenging problem. This thesis has reviewed state of the art technologies from the perspective of Eulerian/Lagrangian views, and have exploited two popular methods of the existing computer graphics methods: Material Point Method and Peridynamics. This thesis focuses on the development of novel hybrid simulation methods based on MPM and PD to robustly and intuitively animate elastoplastic materials with compelling fracture effects (e.g. brittle and ductile fractures). Each method in this thesis independently solves an existing problem or proposes a novel approach to certain simulation scenarios. Together, they contribute to the unified goal of simulating elastoplastic materials as well as fracture effects.

The traditional MPM exhibits instability numerical failures around discontinuous particle distribution. To heal this problem, we propose an integral-based MPM by adopting a PD integral energy density function to replace the partial-derivative stress term in the weak form from the continuum mechanics theory. We discretize the weak form using the standard MPM shape function with the exception that the term involved partial derivatives is augmented by our integral force density function. This augmentation sidesteps the special treatments of the standard MPM in handling the discontinuous particle distribution. We design the integral energy density function with elastic, plastic, viscoelastic and fracture models with PD formulation. Finally, a bond failure law is combined to remove particular PD bonds. The integral-based MPM outweighs the differential-based MPM in stability. The details of this work have been discussed in Chapter 5.

However, the integral-based MPM adds PD bonds for each material point which is costly in computation. For example, one material point in unbroken area also needs loop all family particles at each time step. Taking one step further from the first method, we look for a novel PD-MPM combination to focus on broken areas. Because handling myriad fragments produced by impact forces, especially in high velocity impact scenarios, requires an efficient strategy to handle numerous evolving cracks. Our second contribution is to formulate a rigorous coupling governing equation which integrates the state-based PD with the MPM (Superposition-based MPM). The PD evolves as a result of failure evolution in the critical areas of the MPM domain. As such, only few MPM particles in high released energy region are transformed into PD particles. The SPB-MPM is able to robustly derive entire problem domain while retains the PD abilities of intuitively animating fractures. Giving a low-overhead PD computation to the current MPM, this method allows for simulating a breadth of fracture effects, including tracking branching and evolving crack fronts. The details of this work have been discussed in Chapter 6.

Moreover, general constitutive models in continuum solid theory are not able to produce the prominent features in high velocity impact scenarios.

The characteristic behaviours appears under high strain rate and varying environment conditions. The rapid impact usually introduces shock wave transition within objects. Our third contribution is to introduce a shock wave effects model and a metallic plastic model. The shock wave effects model augments the stable Neo-Hookean model with a hydropressure term. The metallic plastic model adopts the addictive plasticity theory and hardening schemes. Above models are capable of demonstrating intricate and characteristic impact behaviours in simulating organic fruits, other elasto-plastic solids and metallic materials undergoing impact. The details of this work have been discussed in Chapter 7.

8.2 Future Work

The research in this thesis laid a foundation for the future work of MPM in modelling fractures and high velocity impacts, and opened up several new directions.

Surface reconstruction. This paper obtains intricate fracture details and arresting impact phenomenon with high visual fidelity. However, the final effects are limited to the surface reconstruction strategy. Most of the demonstrations in this thesis are particle-based. To export compelling effects needs to reconstruct surface based on particles. This study does not thoroughly explore surface reconstruction algorithm. Instead, we adopt the VDB volume reconstruction in SideFX Houdini which reconstructs volume based on voxels and automatically builds a smooth surface. The VDB loses some details on object surface and edge sharpness of fragments. Wang et al. (2019) proposed a mesh-cutting strategy based on the MPM simulation data and treated it a post process after the simulation. This method is noteworthy due to that it allows for mesh-based object input and uses mesh cutting, sewing approach to handle fractures on surface. It avoids the smoothness operation of the VDB reconstruction. However, the generation of new meshes is cruel in practical implementation. The mesh cutting and sewing is important in visualising brittle fractures, which is necessary in the future work.

Anisotropic fracture modelling. We provide computer graphics with a J2 return mapping method that enables metallic materials to be deformed with hardening effects. This linear mapping method, with an associative flow rule, successfully avoids the tedious iterations in projecting stress tensor back to the Johnson-Cook yield surface. However, the associative flow rule assumes that the corrected stress tensor is perpendicular to the yield surface which means this linear model is not suitable to anisotropic plastic materials. The anisotropic plasticity has been solved by an inspiring work (Wolper et al. 2019) with an non-associative flow rule. It is designed with the continuum damage field. Our Eigenerosion scheme can be extended to an anisotropic damage model when encoding fibre direction within the energy term in of Eq. (6.10) as $\Phi(\sigma^+) = A\sigma^+ : \sigma^+ A$, where A is a matrix encoding two normalised direction vectors (leading to a similar formulation in the work (Wolper et al. 2020)). The studies in this research focuses on isotropic materials with the superior PD method. Above reformulation of the $\Phi(\sigma^+)$ term in Eigenerosion method provides us an opportunity to model anisotropic fractures with the outstanding features of the SPB-MPM, such as brittle glasses.

Phenomena modelling. Simulations of high velocity impact was initially explored by researchers in mechanical field (Silling et al. 2017, Watson and Steinhauser 2017). One of their research aims is to borrow the scientific analyses for solving some necessarily important safety concerns in impact scenarios, such as impact of a bird on an aircraft structure and its ingestion by a jet engine in aviation. These simulations are based on comprehensive data, including specific material properties under given room temperature from real life experiments. Furthermore, the researchers were able to examine the computational results with accurate data extracted in real life experiments. Our method currently focuses on exhibiting phenomena based on general observations, such as car crash. Thus we simplify the models and EOS from the comprehensive mechanical field studies and introduce them to computer graphics. As such, we do not thoroughly conduct numerical analysis for our methods. However, we assuage this limitation by comparing the method with some footage shots (as in Fig. 7.8 and Fig. 7.9). We concluded that

our methods are able to obtain similar fracture effects. Future work direction is to explore our methods on more serious experiments (e.g. bird strike and explosion behaviours). Strictly integrating material physical properties to current methods enables us to expand the research in mechanical safety topics and helps with examination our methods with experimental data.

Appendix A

PD weak form

The updated Lagrangian time discretization of the weak form in Eq. (6.3) leads to an integration form of different domain contributions (Jiang et al. 2016):

$$\begin{aligned}\int_{\Omega^{PD}} \rho a \cdot (w^{MPM} + w^{PD}) dx + \int_{\Omega^{MPM} \setminus \Omega^{PD}} \rho a \cdot w^{MPM} dx = \\ - \int_{\Omega^{PD}} \sigma(u^{MPM} + u^{PD}, t) : \nabla w^{MPM} dx + \int_{\Omega^{PD}} b(x,t) w^{PD} dx \\ - \int_{\Omega^{PD}} \sigma(u^{MPM} + u^{PD}, t) : \nabla w^{PD} dx \\ - \int_{\Omega^{MPM} \setminus PD} \sigma(u^{MPM}) : \nabla w^{MPM} dx + \int_{\Omega^{MPM}} b(x,t) w^{MPM} dx \end{aligned} \quad \text{(A.1)}$$

where variables are defined as in Equation (6.1)-(6.3) and $\int_{\Omega^{MPM}} b(x,t) w^{MPM} dx = \int_{\Omega^{MPM} \setminus PD} b(x,t) w^{MPM} dx + \int_{\Omega^{PD}} b(x,t) w^{MPM} dx$.

After subtracting Equation (6.4), which contains the MPM related items from above formulation, we obtain the weak form of the governing equation related to the PD domain as:

$$\int_{\Omega^{PD}} \rho(x,t) a \cdot (w^{PD}) dx = - \int_{\Omega^{PD}} \sigma(u^C + u^P, t^n) : \nabla w^P dx + \int_{\Omega^{PD}} b(x,t) \cdot w^{PD} dx \quad \text{(A.2)}$$

PD, when used to capture local fracture dynamics, is mainly driven by local deformation state instead of its own body force (and acceleration) so we drop body force (and acceleration) items here for simplicity. Following Sun et al. (2019), we can express the remaining integration $\int_{\Omega^{PD}} \sigma(u^C + u^P, t^n) : \nabla w^P dx$

into the form of internal energy stored in PD bonds. Starting with a given deformed bond state Y between a pair of PD particles x_p and x_q, we define the PD deformation gradient F_p^{PD} and PD force vector T_i^{PD}:

$$F_{ik} = \int_{H_p} \omega^{PD}(X) Y_i X_l K_{kl} dV_q \tag{A.3}$$

$$T_i^{PD} = \omega(X) det(F) F_{ij}^{-1} \sigma_{ij} K_{kl} X_l \tag{A.4}$$

where $\omega(X)$ is a weight function for bond X. Following Sun et al. (2019), we assume the internal energy increase on all the bonds in the family as they deform from state p to q with a small change $dY = w^q - w^p$:

$$\begin{aligned}\int_{\Omega^{PD}} T^{PD} \cdot dY dx &= \int_{\Omega^{PD}} \int_{H_p} \omega(X) det(F) F_{ij}^{-1} \sigma_{ij} K_{kl} X_l dY_i dV_q dx \\ &= \int_{\Omega^{PD}} (det(F) F_{ij}^{-1} \sigma_{ij}) (\int_{H_p} \omega(X) dY_i X_l K_{kl} dV_q) dx \\ &= \int_{\Omega^{PD}} det(F) F_{ij}^{-1} \sigma_{ij} dF_{ik} dx = -\int_{\Omega^{PD}} \sigma(u) : \nabla w^{PD} dx\end{aligned} \tag{A.5}$$

This is a backwards proof. Then we seek to write the left side of above equation in the form of classic PD integration. Then internal energy function can also be expressed in Equation (A.6).

$$\begin{aligned}&\int_{\Omega^{PD}} T^{PD} \cdot dY dx = \int_{\Omega^{PD}} \int_{\Omega^{PD}} (w_q^{PD} - w_p^{PD}) T[x_p, t^n] < x_q - x_p > dx_p dx_q \\ &= \int_{\Omega^{PD}} \int_{\Omega^{PD}} w_q^{PD} T[x_p, t^n] < x_q - x_p > dx_p dx_q - \int_{\Omega^{PD}} \int_{\Omega^{PD}} w_p^{PD} T[x_p, t^n] < x_q - x_p > dx_p dx_q \\ &= \int_{\Omega^{PD}} \int_{\Omega^{PD}} w_p^{PD} T[x_q, t^n] < x_p - x_q > dx_p dx_q - \int_{\Omega^{PD}} \int_{\Omega^{PD}} w_p^{PD} T[x_p, t^n] < x_q - x_p > dx_p dx_q \\ &= -\int_{\Omega^{PD}} \int_{H_p} w_p^{PD} (T[x_p, t^n] < x_q - x_p > - T[x_q, t^n] < x_p - x_q >) dV_q dx_p\end{aligned} \tag{A.6}$$

where force vector T^{PD} of point p is summation of all bond forces defined on bond vector $< x_q - x_p >$ with x_q is a connected neighbor point of x_p. We can write $T^{PD} = \int_{H_p} T[x_p, t^n] < x_q - x_p > dV_q$. Without lose generality, $T^{PD} = \int_{\Omega^{PD}} T[x_p, t^n] < x_q - x_p > dx_q$. Here H_p defines the neighbourhood. In the above proof, we swap the dummy index q with p to get the final result.

Appendix B

J2 Plasticity Flow Rule

With a fourth-order stiffness tensor of material properties C_{ijkl} (in our work, $C_{ijkl} = 2\mu$), μ is the shear modulus computed from Young's Modulus $E$and possion ratio ν. We describe the plastic strain increment as $\Delta\epsilon_{ij}^{p^{n+1}}$. Below is the return mapping algorithm.

$$\begin{aligned} \sigma_{ij}^{n+1} &= \sigma_{ij}^{n+1^*} + \Delta\sigma_{ij} \\ \Delta\sigma_{ij} &= -C_{ijkl}\Delta\epsilon_{ij}^{p^{n+1}} \end{aligned} \tag{B.1}$$

where $\sigma_{ij}^{n+1^*}$ is the predicted deviatoric stress tensor, σ_{ij}^{n+1} is the corrected stress tensor on the yield surface in Fig. 7.4. With associative flow rule, the plastic strain increment vector is normal to the yield surface so thus the relation of plastic strain increment and equivalent plastic increment is:

$$\Delta\epsilon_{ij}^{p^{n+1}} = \sqrt{\frac{3}{2}}\Delta\epsilon^{p^{n+1}} n_{ij} \tag{B.2}$$

The direction of plastic flow in associative flow rule can be obtained with :

$$\begin{aligned} r_{ij} = \frac{\partial f(s,\epsilon^p)}{\partial\sigma_{ij}} = \frac{3s_{ij}}{2s} &= \sqrt{\frac{3}{2}} n_{ij} \\ n_{ij} &= \sqrt{\frac{3}{2}}\frac{s_{ij}}{s} \end{aligned} \tag{B.3}$$

Substitutve the above equations to Eq. (7.5), we obtain:

$$\begin{aligned} f(s,\epsilon^p) + \frac{\partial f(s,\epsilon^p)}{\partial \sigma_{ij}}\Delta\sigma_{ij} + \frac{\partial f(s,\epsilon^p)}{\partial \epsilon^p}\Delta\epsilon^{p^{n+1}} \\ = f(s,\epsilon^p) + \sqrt{\frac{3}{2}}n_{ij}(-2\mu\sqrt{\frac{3}{2}}\Delta\epsilon^{p^{n+1}}n_{ij}) - E^p\Delta\epsilon^{p^{n+1}} \\ = f(s,\epsilon^p) + \sqrt{\frac{3}{2}}n_{ij}(-2\mu\sqrt{\frac{3}{2}}\Delta\epsilon^{p^{n+1}}n_{ij}) - E^p\Delta\epsilon^{p^{n+1}} \\ = f(s,\epsilon^p) - 3\mu\Delta\epsilon^{p^{n+1}} - E^p\Delta\epsilon^{p^{n+1}} = 0 \end{aligned} \quad \text{(B.4)}$$

Then we update the equivalent plastic strain and yield stress as following:

$$\Delta\epsilon^{p^{n+1}} = \frac{f(s,\epsilon^p)}{3\mu + E^p} \quad \text{(B.5)}$$

$$\epsilon^{p^{n+1}} = \epsilon^{p^n} + \Delta\epsilon^{p^{n+1}} \quad \text{(B.6)}$$

$$\sigma_y^{n+1} = \sigma_y^n + E^p\Delta\epsilon^{p^{n+1}} \quad \text{(B.7)}$$

Finally, we project the deviatoric stress tensor back to yield surface by:

$$\sigma_{ij}^{n+1} = \frac{\sigma_y^{n+1}}{s^{n+1^*}}\sigma_{ij}^{n+1^*} \quad \text{(B.8)}$$

where s^{n+1^*} is equivalent stress in terms of predicted deviatoric stress tensor $\sigma_{ij}^{n+1^*}$.

References

Bao, Z., Hong, J.-M., Teran, J. and Fedkiw, R., 2007. Fracturing rigid materials. *IEEE Transactions on Visualization and Computer Graphics*, 13 (2), 370–378.

Bargteil, A. W., Wojtan, C., Hodgins, J. K. and Turk, G., 2007. A finite element method for animating large viscoplastic flow. *ACM transactions on graphics (TOG)*, 26 (3), 16–es.

Becker, M., Ihmsen, M. and Teschner, M., 2009. Corotated sph for deformable solids. *NPH*, 27–34.

Belytschko, T., Organ, D. and Krongauz, Y., 1995. A coupled finite element-element-free galerkin method. *Computational Mechanics*, 17 (3), 186–195.

Benson, D., 1995. A multi-material eulerian formulation for the efficient solution of impact and penetration problems. *Computational Mechanics*, 15 (6), 558–571.

Bonet, J. and Wood, R. D., 1997. *Nonlinear continuum mechanics for finite element analysis*. Cambridge university press.

Bower, A. F., 2009. *Applied mechanics of solids*. CRC press.

Brackbill, J. U. and Ruppel, H. M., 1986. Flip: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions. *Journal of Computational physics*, 65 (2), 314–343.

Carlson, M., Mucha, P. J., Van Horn III, R. B. and Turk, G., 2002. Melting and flowing. *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 167–174.

Channel, T. M. G., 2015. Incredible super slow motion bullet impact! - m855a1. URL `https://www.youtube.com/watch?v=fX4ODh1g4eM`.

Chao, I., Pinkall, U., Sanan, P. and Schröder, P., 2010. A simple geometric model for elastic deformations. *ACM transactions on graphics (TOG)*, 29 (4), 38.

Chen, W., Zhu, F., Zhao, J., Li, S. and Wang, G., 2018. Peridynamics-based fracture animation for elastoplastic solids. *Computer Graphics Forum*, Wiley Online Library, volume 37, 112–124.

Chen, Z., Yao, M., Feng, R. and Wang, H., 2014. Physics-inspired adaptive fracture refinement. *ACM Transactions on Graphics (TOG)*, 33 (4), 1–7.

Chitalu, F. M., Miao, Q., Subr, K. and Komura, T., 2020. Displacement-correlated xfem for simulating brittle fracture. *Computer Graphics Forum*, Wiley Online Library, volume 39, 569–583.

Cundall, P. A. and Strack, O. D., 1979. A discrete numerical model for granular assemblies. *geotechnique*, 29 (1), 47–65.

De Moura, C. A. and Kubrusly, C. S., 2013. The courant–friedrichs–lewy (cfl) condition. *AMC*, 10 (12).

Ding, M., Han, X., Wang, S., Gast, T. F. and Teran, J. M., 2019. A thermomechanical material point method for baking and cooking. *ACM Transactions on Graphics (TOG)*, 38 (6), 1–14.

Drucker, D. C. and Prager, W., 1952. Soil mechanics and plastic analysis or limit design. *Quarterly of applied mathematics*, 10 (2), 157–165.

Fang, Y., Li, M., Gao, M. and Jiang, C., 2019. Silly rubber: an implicit material point method for simulating non-equilibrated viscoelastic and elastoplastic solids. *ACM Transactions on Graphics (TOG)*, 38 (4), 1–13.

Foster, J. T., Silling, S. A. and Chen, W., 2011. An energy based failure criterion for use with peridynamic states. *International Journal for Multiscale Computational Engineering*, 9 (6).

Foster, N. and Metaxas, D., 1996. Realistic animation of liquids. *Graphical models and image processing*, 58 (5), 471–483.

Gao, M., Tampubolon, A. P., Jiang, C. and Sifakis, E., 2017. An adaptive generalized interpolation material point method for simulating elastoplastic materials. *ACM Transactions on Graphics (TOG)*, 36 (6), 223.

Goktekin, T. G., Bargteil, A. W. and O'Brien, J. F., 2004. A method for animating viscoelastic fluids. *ACM SIGGRAPH 2004 Papers*, 463–468.

Hahn, D. and Wojtan, C., 2015. High-resolution brittle fracture simulation with boundary elements. *ACM Transactions on Graphics (TOG)*, 34 (4), 1–12.

Harlow, F. H., 1962. The particle-in-cell method for numerical solution of problems in fluid dynamics. Technical report, Los Alamos Scientific Lab., N. Mex.

He, X., Wang, H. and Wu, E., 2017. Projective peridynamics for modeling versatile elastoplastic materials. *IEEE transactions on visualization and computer graphics*, 24 (9), 2589–2599.

Homel, M. A. and Herbold, E. B., 2017. Field-gradient partitioning for fracture and frictional contact in the material point method. *International Journal for Numerical Methods in Engineering*, 109 (7), 1013–1044.

Hu, W. and Chen, Z., 2006. Model-based simulation of the synergistic effects of blast and fragmentation on a concrete wall using the mpm. *International journal of impact engineering*, 32 (12), 2066–2096.

Hu, Y., 2020. The taichi programming language. *ACM SIGGRAPH 2020 Courses*, 1–50.

Hu, Y., Anderson, L., Li, T.-M., Sun, Q., Carr, N., Ragan-Kelley, J. and Durand, F., 2019. Difftaichi: Differentiable programming for physical simulation. *arXiv preprint arXiv:1910.00935*.

Hu, Y., Fang, Y., Ge, Z., Qu, Z., Zhu, Y., Pradhana, A. and Jiang, C., 2018. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Transactions on Graphics (TOG)*, 37 (4), 1–14.

Hughes, T. J., 2012. *The finite element method: linear static and dynamic finite element analysis*. Courier Corporation.

Jiang, C., 2015. *The material point method for the physics-based simulation of solids and fluids*. University of California, Los Angeles.

Jiang, C., Gast, T. and Teran, J., 2017. Anisotropic elastoplasticity for cloth, knit and hair frictional contact. *ACM Transactions on Graphics (TOG)*, 36 (4), 1–14.

Jiang, C., Schroeder, C., Selle, A., Teran, J. and Stomakhin, A., 2015. The affine particle-in-cell method. *ACM Transactions on Graphics (TOG)*, 34 (4), 1–10.

Jiang, C., Schroeder, C., Teran, J., Stomakhin, A. and Selle, A., 2016. The material point method for simulating continuum materials. *ACM SIGGRAPH 2016 Courses*, 1–52.

Johnson, G. R., Stryk, R. A. and Beissel, S. R., 1996. Sph for high velocity impact computations. *Computer methods in applied mechanics and engineering*, 139 (1-4), 347–373.

Kaufmann, P., Martin, S., Botsch, M., Grinspun, E. and Gross, M., 2009. Enrichment textures for detailed cutting of shells. *ACM SIGGRAPH 2009 papers*, 1–10.

Kawamura, K., Kobayashi, Y. and Fujie, M. G., 2008. Basic study on real-time simulation using mass spring system for robotic surgery. *International Workshop on Medical Imaging and Virtual Reality*, Springer, 311–319.

Klár, G., Gast, T., Pradhana, A., Fu, C., Schroeder, C., Jiang, C. and Teran, J., 2016. Drucker-prager elastoplasticity for sand animation. *ACM Transactions on Graphics (TOG)*, 35 (4), 1–12.

Koschier, D., Bender, J. and Thuerey, N., 2017. Robust extended finite elements for complex cutting of deformables. *ACM Transactions on Graphics (TOG)*, 36 (4), 1–13.

Lemons, D. S. and Lund, C. M., 1999. Thermodynamics of high temperature, mie–gruneisen solids. *American Journal of Physics*, 67 (12), 1105–1108.

Levin, D. I., Litven, J., Jones, G. L., Sueda, S. and Pai, D. K., 2011. Eulerian solid simulation with contact. *ACM Transactions on Graphics (TOG)*, 30 (4), 1–10.

Levine, J. A., Bargteil, A. W., Corsi, C., Tessendorf, J. and Geist, R., 2014. A peridynamic perspective on spring-mass fracture. *Symposium on Computer Animation*, Citeseer, 47–55.

Liang, Y., Benedek, T., Zhang, X. and Liu, Y., 2017. Material point method with enriched shape function for crack problems. *Computer Methods in Applied Mechanics and Engineering*, 322, 541–562.

Littlewood, D. J., 2015. Roadmap for peridynamic software implementation. *SAND Report, Sandia National Laboratories, Albuquerque, NM and Livermore, CA*.

Liu, M. and Liu, G., 2010. Smoothed particle hydrodynamics (sph): an overview and recent developments. *Archives of computational methods in engineering*, 17 (1), 25–76.

Liu, T., Bargteil, A. W., O'Brien, J. F. and Kavan, L., 2013. Fast simulation of mass-spring systems. *ACM Transactions on Graphics (TOG)*, 32 (6), 1–7.

Liu, W. and Hong, J.-W., 2012. A coupling approach of discretized peridynamics with finite element method. *Computer Methods in Applied Mechanics and Engineering*, 245, 163–175.

Lu, Y., Belytschko, T. and Tabbara, M., 1995. Element-free galerkin method for wave propagation and dynamic fracture. *Computer Methods in Applied Mechanics and Engineering*, 126 (1-2), 131–153.

Lubineau, G., Azdoud, Y., Han, F., Rey, C. and Askari, A., 2012. A morphing strategy to couple non-local to local continuum mechanics. *Journal of the Mechanics and Physics of Solids*, 60 (6), 1088–1102.

Lyu, Y., Zhang, J., Chang, J., Guo, S. and Zhang, J. J., 2019. Integrating peridynamics with material point method for elastoplastic material modeling. *Computer Graphics International Conference*, Springer, 228–239.

Lyu, Y., Zhang, J., Sarafopoulos, A., Chang, J., Guo, S. and Zhang, J. J., 2020. Integral-based material point method and peridynamics model for animating elastoplastic material. *Transactions on Computational Science XXXVII*, Springer, 91–108.

Ma, S., Zhang, X., Lian, Y. and Zhou, X., 2009a. Simulation of high explosive explosion using adaptive material point method. *Computer Modeling in Engineering and Sciences (CMES)*, 39 (2), 101.

Ma, S., Zhang, X. and Qiu, X., 2009b. Comparison study of mpm and sph in modeling hypervelocity impact problems. *International journal of impact engineering*, 36 (2), 272–282.

Martin, S., Thomaszewski, B., Grinspun, E. and Gross, M., 2011. Example-based elastic materials. *ACM SIGGRAPH 2011 papers*, 1–8.

Mitchell, J. A., 2011. A nonlocal ordinary state-based plasticity model for peridynamics. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States).

Molinari, J., 2002. Finite element simulation of shaped charges. *Finite elements in analysis and design*, 38 (10), 921–936.

Monaghan, J. J., 1992. Smoothed particle hydrodynamics. *Annual review of astronomy and astrophysics*, 30 (1), 543–574.

Moutsanidis, G., Kamensky, D., Zhang, D. Z., Bazilevs, Y. and Long, C. C., 2019. Modeling strong discontinuities in the material point method using a single velocity field. *Computer Methods in Applied Mechanics and Engineering*, 345, 584–601.

Müller, M. and Gross, M. H., 2004. Interactive virtual materials. *Graphics interface*, volume 2004, 239–246.

Müller, M., Heidelberger, B., Hennix, M. and Ratcliff, J., 2007. Position based dynamics. *Journal of Visual Communication and Image Representation*, 18 (2), 109–118.

Nairn, J. A., 2003. Material point method calculations with explicit cracks. *Computer Modeling in Engineering and Sciences*, 4 (6), 649–664.

Ne, M. and Fiume, E., 1999. A visual model for blast waves and fracture. *Proceedings of the 1999 conference on Graphics interface*, Citeseer, volume 99, 193–202.

Nedel, L. P. and Thalmann, D., 1998. Real time muscle deformations using mass-spring systems. *Proceedings. Computer Graphics International (Cat. No. 98EX149)*, IEEE, 156–165.

Nguyen, V. P., 2014. Material point method: basics and applications. *Cardiff University, Cardiff, UK*.

O'brien, J. F., Bargteil, A. W. and Hodgins, J. K., 2002. Graphical modeling and animation of ductile fracture. *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, 291–294.

O'brien, J. F. and Hodgins, J. K., 1999. Graphical modeling and animation of brittle fracture. *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 137–146.

Oxley, P. L. B., 1963. Rate of strain effect in metal cutting.

Pandolfi, A. and Ortiz, M., 2012. An eigenerosion approach to brittle fracture. *International Journal for Numerical Methods in Engineering*, 92 (8), 694–714.

Pauly, M., Keiser, R., Adams, B., Dutré, P., Gross, M. and Guibas, L. J., 2005. Meshless animation of fracturing solids. *ACM Transactions on Graphics (TOG)*, 24 (3), 957–964.

Ram, D., Gast, T., Jiang, C., Schroeder, C., Stomakhin, A., Teran, J. and Kavehpour, P., 2015. A material point method for viscoelastic fluids, foams and sponges. *Proceedings of the 14th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 157–163.

Randles, P. and Libersky, L. D., 1996. Smoothed particle hydrodynamics: some recent improvements and applications. *Computer methods in applied mechanics and engineering*, 139 (1-4), 375–408.

Salençon, J., 2012. *Handbook of continuum mechanics: General concepts thermoelasticity*. Springer Science & Business Media.

Schmidt, B., Fraternali, F. and Ortiz, M., 2009. Eigenfracture: an eigendeformation approach to variational fracture. *Multiscale Modeling & Simulation*, 7 (3), 1237–1266.

Seleson, P., Beneddine, S. and Prudhomme, S., 2013. A force-based coupling scheme for peridynamics and classical elasticity. *Computational Materials Science*, 66, 34–49.

Silling, S. A., 2000. Reformulation of elasticity theory for discontinuities and long-range forces. *Journal of the Mechanics and Physics of Solids*, 48 (1), 175–209.

Silling, S. A., 2010. Linearized theory of peridynamic states. *Journal of Elasticity*, 99 (1), 85–111.

Silling, S. A., 2011. A coarsening method for linear peridynamics. *International Journal for Multiscale Computational Engineering*, 9 (6).

Silling, S. A., 2018. Modeling penetration and perforation with peridynamics. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States).

Silling, S. A. and Askari, E., 2005. A meshfree method based on the peridynamic model of solid mechanics. *Computers & structures*, 83 (17-18), 1526–1535.

Silling, S. A., Epton, M., Weckner, O., Xu, J. and Askari, E., 2007. Peridynamic states and constitutive modeling. *Journal of Elasticity*, 88 (2), 151–184.

Silling, S. A. and Lehoucq, R. B., 2010. Peridynamic theory of solid mechanics. *Advances in applied mechanics*, 44, 73–168.

Silling, S. A., Parks, M. L., Kamm, J. R., Weckner, O. and Rassaian, M., 2017. Modeling shockwaves and impact phenomena with eulerian peridynamics. *International Journal of Impact Engineering*, 107, 47–57.

Smith, B., Goes, F. D. and Kim, T., 2018. Stable neo-hookean flesh simulation. *ACM Transactions on Graphics (TOG)*, 37 (2), 1–15.

Stochino, F., Qinami, A. and Kaliske, M., 2017. Eigenerosion for static and dynamic brittle fracture. *Engineering Fracture Mechanics*, 182, 537–551.

Stomakhin, A., Howes, R., Schroeder, C. and Teran, J. M., 2012. Energetically consistent invertible elasticity. *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, Eurographics Association, 25–32.

Stomakhin, A., Schroeder, C., Chai, L., Teran, J. and Selle, A., 2013. A material point method for snow simulation. *ACM Transactions on Graphics (TOG)*, 32 (4), 1–10.

Sulsky, D. and Kaul, A., 2004. Implicit dynamics in the material-point method. *Computer Methods in Applied Mechanics and Engineering*, 193 (12-14), 1137–1170.

Sulsky, D., Zhou, S.-J. and Schreyer, H. L., 1995. Application of a particle-in-cell method to solid mechanics. *Computer physics communications*, 87 (1-2), 236–252.

Sun, W. and Fish, J., 2019. Superposition-based coupling of peridynamics and finite element method. *Computational Mechanics*, 64 (1), 231–248.

Sun, W., Fish, J. and Zhang, G., 2019. Superposition of non-ordinary state-based peridynamics and finite element method for material failure simulations. *Meccanica*, 1–19.

Terzopoulos, D. and Fleischer, K., 1988. Modeling inelastic deformation: viscolelasticity, plasticity, fracture. *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, 269–278.

Tran, L. and Udaykumar, H., 2004. A particle-level set-based sharp interface cartesian grid method for impact, penetration, and void collapse. *Journal of Computational Physics*, 193 (2), 469–510.

Wallstedt, P. and Guilkey, J., 2007. Improved velocity projection for the material point method. *Computer Modeling in Engineering and Sciences*, 19 (3), 223.

Wang, S., Ding, M., Gast, T. F., Zhu, L., Gagniere, S., Jiang, C. and Teran, J. M., 2019. Simulation and visualization of ductile fracture with the material point method. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 2 (2), 1–20.

Watson, E. and Steinhauser, M. O., 2017. Discrete particle method for simulating hypervelocity impact phenomena. *Materials*, 10 (4), 379.

Wicke, M., Ritchie, D., Klingner, B. M., Burke, S., Shewchuk, J. R. and O'Brien, J. F., 2010. Dynamic local remeshing for elastoplastic simulation. *ACM Transactions on graphics (TOG)*, 29 (4), 1–11.

Wilson, C. D., 2002. A critical reexamination of classical metal plasticity. *J. Appl. Mech.*, 69 (1), 63–68.

Wolper, J., Chen, Y., Li, M., Fang, Y., Qu, Z., Lu, J., Cheng, M. and Jiang, C., 2020. Anisompm: Animating anisotropic damage mechanics: Supplemental document. *ACM Trans. Graph*, 39 (4).

Wolper, J., Fang, Y., Li, M., Lu, J., Gao, M. and Jiang, C., 2019. Cd-mpm: Continuum damage material point methods for dynamic fracture animation. *ACM Trans. Graph.*, 38 (4).

Wretborn, J., Armiento, R. and Museth, K., 2017. Animation of crack propagation by means of an extended multi-body solver for the material point method. *Computers & Graphics*, 69, 131–139.

Xu, H., Sin, F., Zhu, Y. and Barbič, J., 2015. Nonlinear material design using principal stretches. *ACM Transactions on Graphics (TOG)*, 34 (4), 1–11.

Xu, L., He, X., Chen, W., Li, S. and Wang, G., 2018. Reformulating hyperelastic materials with peridynamic modeling. *Computer Graphics Forum*, Wiley Online Library, volume 37, 121–130.

Zel'Dovich, Y. B. and Raizer, Y. P., 2002. *Physics of shock waves and high-temperature hydrodynamic phenomena*. Courier Corporation.

Zhang, N., Zhou, X., Sha, D., Yuan, X., Tamma, K. K. and Chen, B., 2006. Integrating mesh and meshfree methods for physics-based fracture and debris cloud simulation. *SPBG*, 145–154.

Zhang, X., Chen, Z. and Liu, Y., 2016. *The material point method: a continuum-based particle method for extreme loading cases*. Academic Press.

Zhang, X., Lian, Y., Liu, Y. and Zhou, X., 2013. Material point method. *Beijing: Tsinghua University*.

Zhang, Z. and Chen, Q., 2007. Comparison of the eulerian and lagrangian methods for predicting particle transport in enclosed spaces. *Atmospheric environment*, 41 (25), 5236–5248.

Zhang, Z.-X., 2016. *Rock fracture and blasting: theory and applications*. Butterworth-Heinemann.

Zhu, B., Lee, M., Quigley, E. and Fedkiw, R., 2015. Codimensional non-newtonian fluids. *ACM Transactions on Graphics (TOG)*, 34 (4), 115.

Zhu, B. and Yang, X., 2010. Animating sand as a surface flow. *Eurographics (short papers)*, 9–12.

Zhu, F., Zhao, J., Li, S., Tang, Y. and Wang, G., 2017. Dynamically enriched mpm for invertible elasticity. *Computer Graphics Forum*, Wiley Online Library, volume 36, 381–392.

Zhu, Y. and Bridson, R., 2005. Animating sand as a fluid. *ACM Transactions on Graphics (TOG)*, 24 (3), 965–972.