Part III
Point-based multiphysics modeling

Daniel Holz[1,2], **Stefan Rhys Jeske**[3], Fabian Löschner[3], Jan Bender[3], Yin Yang[4], Sheldon Andrews[2]

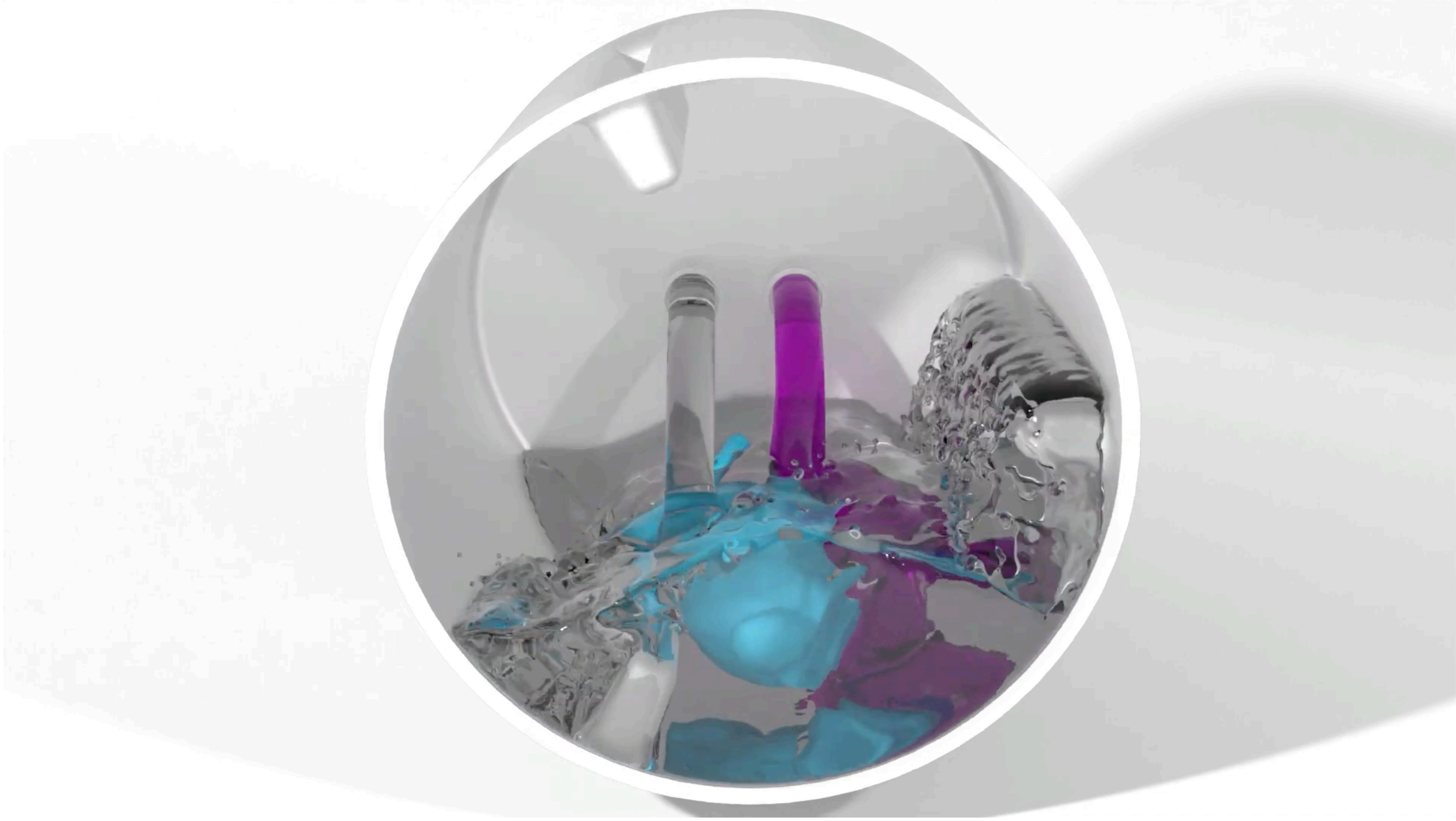
Unity

ÉTS Engineering for Industry ÉCOLE DE TECHNOLOGIE SUPÉRIEURE Université du Québec

RWTH AACHEN UNIVERSITY

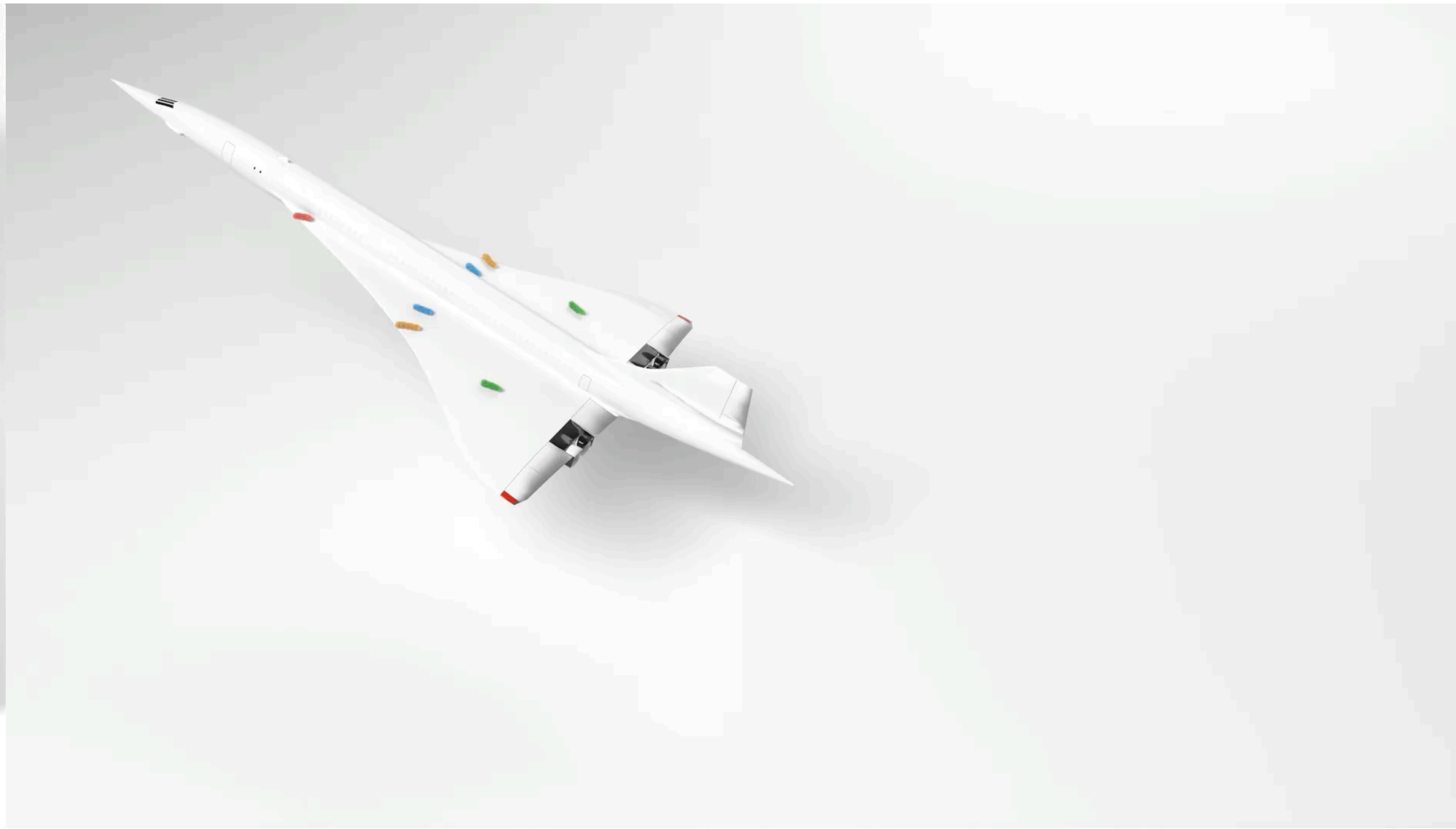
THE UNIVERSITY OF UTAH

Lagrangian Point-based Methods



A comparison of linear consistent correction methods for first-order SPH derivatives [Westhofen et al. 2023]

Eulerian and Hybrid Methods



Fast and versatile fluid-solid coupling for turbulent flow simulation [Lyu et al. 2021]

Overview

- Lagrangian Point-based Methods
  - Overview
  - Fluid
  - Solid and Rigid
  - Multiphysics Materials
- Eulerian and Hybrid Methods
  - Overview
  - Multiphysics Materials
- Summary

	Lagrangian Point-Based Methods (Sec. 2)	Eulerian & Hybrid Methods (Sec. 3)	Energy-Based Modeling (Sec. 4)	Constraint-Based Modeling (Sec. 5)
Deformables (elastic & plastic)	[MKN*04] [PKA*05] [SSP07] [BIT09] [MKB*10] [YJL*16] [YCL*17] [PGBT18] [CLC*20] [KBF*21] [KUKH23]	[SZS95] [CGFO06] [LLJ*11] [SSJ*14] [JSS*15] [YSB*15] [TLK16] [FGG*17] [GTJS17] [JGT17] [ZB17] [GHF*18] [HFG*18] [FLGJ19] [HGG*19] [SXH*21] [LLJ22] [TB22] [QLY*23] [LLH*24] [TLZ*24]	[BAV*10] [BUAG12] [SB12b] [SHST12] [BML*14] [GSS*15] [LBK17] [BOFN18] [SGK18] [LFS*20] [MEM*20] [LMY*22] [LCK22] [LLJ22] [KE22] [LFFJ*23]	[Jak01] [MHTG05] [MHHR06] [SLM06] [MMCK14] [BKCW14] [Cho14] [MCKM15] [CMM16] [DCB16] [MMC16] [BGAO17] [FM17] [ARM*19] [MEM*19] [WWB*19] [MMC*20] [MM21] [TTKA23] [CHC*24a] [Cet24] [MAK24] [SZDJ24] [YLL*24]
Granular Materials	[LD09] [AO11] [IWT13] [YJL*16] [YCL*17] [GHB*20]	[ZB05] [SSC*13] [DBD16] [KGP*16] [TGK*17] [GPH*18]		[Hol14] [MMCK14] [SWLB14] [FM17] [HG18] [NS18] [KKHS20] [YLL*24]
Rigid Bodies & Multibody Systems	[SSP07] [YCL*17] [GPB*19] [PT23]	[TB20] [TB22] [LLH*24] [TLZ*24]	[CDGB19] [MEM*20] [FLS*21] [CLL*22] [LKL*22]	[Bar94] [MC95] [ST96] [Bar96] [AP97] [Ste00] [Jak01] [Erl05] [MHTG05] [Lac07b, Lac07a] [GZO10] [MMCK14] [DCB16] [FM17] [MEM*19] [PAK*19] [WWB*19] [MMC*20] [MAK24]
Co-dimensional Structures	[MKB*10] [ZQC*14] [ZLQF15]	[JGT17] [GHF*18] [HGG*19] [LLH*24]	[GHDS03] [ST07] [BWR*08] [CSvRV18] [Kim20] [LKJ21] [CXY*23] [HB23] [SWP*23] [WB23] [LFFJB24]	[Jak01] [MHHR06] [GHF*07] [SL08] [SLNB10] [MKC12] [BKCW14] [MMCK14] [USS15] [MMC16] [KS16] [DKWB18] [ARM*19]
Fluids & Fluid Phenomena	[PW02] [MCG03] [SSP07] [BT07] [BIT09] [SP09] [Pri12] [SB12a] [AAT13] [ICS*14] [HWZ*15] [TDF*15] [BK17] [PT17] [YCL*17] [YML*17] [PGBT18] [WKBB18] [BKKW19] [CBG*19] [GPB*19] [WJL*20] [ZRS*20] [KBF*21] [LWB*21] [WDK*21] [LHWW22] [XRW*22] [JWL*23] [PT23] [XLYJ23] [ZLX*24] [YWX*24]	[Har62] [HW*65] [BR86] [FM96] [Sta99] [Pes02] [TUKF02] [CMT04] [ZB05] [CGFO06] [KFCO06] [CFL*07] [MCP*09] [SABS14] [SSJ*14] [ATW15] [JSS*15] [RGJ*15] [FGG*17] [GPH*18] [HFG*18] [JGT17] [ZB17] [FLGJ19] [GAB20] [HGMRT20] [TB20] [CKMR*21] [SXH*21] [QLDGJ22] [TB22] [STBA24] [QLY*23] [LLH*24] [TLZ*24]	[TB20] [TB21] [TB22] [XLYJ23]	[BLS12] [MM13] [MMCK14] [TNF14] [BGAO17] [XRW*22] [YLL*24]
Multi-Phase, Phase Transitions & Porous Flow	[MKN*04] [SSP07] [LAD08] [SP08] [BIT09] [LD09] [PC13] [RLY*14] [YCR*15] [YJL*16] [PGBT18] [CLC*20] [GHB*20] [WFM21] [RXL21] [RHLC22] [XWW*23] [YR23] [ZLX*24]	[SSJ*14] [ATW15] [GPH*18] [GAB20] [CKMR*21] [SXH*21] [LMLD22] [TLZ*24]		[MMCK14]
Other Phenomena	[LL10] [Pri12]	[WFL*19] [WDG*19] [SNZ*21] [FCK22] [CCL*22]	[CSvRV18] [CNZ*22] [WFFJB24]	[GZO10] [Cho14] [BCK*22]

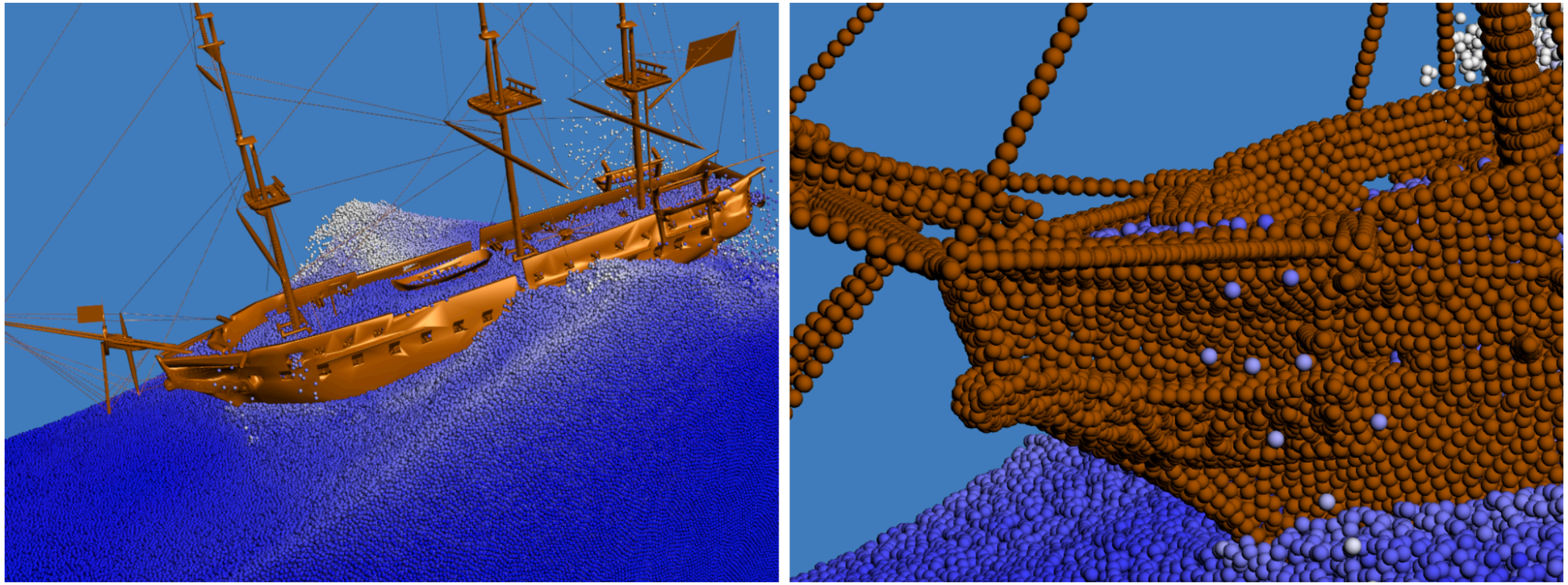
Overview

- Lagrangian Point-based Methods
  - Overview
  - Fluid
  - Solid and Rigid
  - Multiphysics Materials
- Eulerian and Hybrid Methods
  - Overview
  - Multiphysics Materials
- Summary

	Lagrangian Point-Based Methods (Sec. 2)	Eulerian & Hybrid Methods (Sec. 3)	Energy-Based Modeling (Sec. 4)	Constraint-Based Modeling (Sec. 5)
Deformables (elastic & plastic)	[MKN*04] [PKA*05] [SSP07] [BIT09] [MKB*10] [YJL*16] [YCL*17] [PGBT18] [CLC*20] [KBF*21] [KUKH23]	[SZS95] [CGFO06] [LLJ*11] [SSJ*14] [JSS*15] [YSB*15] [TLK16] [FGG*17] [GTJS17] [JGT17] [ZB17] [GHF*18] [HFG*18] [FLGJ19] [HGG*19] [SXH*21] [LLJ22] [TB22] [QLY*23] [LLH*24] [TLZ*24]	[BAV*10] [BUAG12] [SB12b] [SHST12] [BML*14] [GSS*15] [LBK17] [BOFN18] [SGK18] [LFS*20] [MEM*20] [LMY*22] [LCK22] [LLJ22] [KE22] [LFFJ*23]	[Jak01] [MHTG05] [MHHR06] [SLM06] [MMCK14] [BKCW14] [Cho14] [MCKM15] [CMM16] [DCB16] [MMC16] [BGAO17] [FM17] [ARM*19] [MEM*19] [WWB*19] [MMC*20] [MM21] [TTKA23] [CHC*24a] [Cet24] [MAK24] [SZDJ24] [YLL*24]
Granular Materials	[LD09] [AO11] [IWT13] [YJL*16] [YCL*17] [GHB*20]	[ZB05] [SSC*13] [DBD16] [KGP*16] [TGK*17] [GPH*18]		[Hol14] [MMCK14] [SWLB14] [FM17] [HG18] [NS18] [KKHS20] [YLL*24]
Rigid Bodies & Multibody Systems	[SSP07] [YCL*17] [GPB*19] [PT23]	[TB20] [TB22] [LLH*24] [TLZ*24]	[CDGB19] [MEM*20] [FLS*21] [CLL*22] [LKL*22]	[Bar94] [MC95] [ST96] [Bar96] [AP97] [Ste00] [Jak01] [Erl05] [MHTG05] [Lac07b, Lac07a] [GZO10] [MMCK14] [DCB16] [FM17] [MEM*19] [PAK*19] [WWB*19] [MMC*20] [MAK24]
Co-dimensional Structures	[MKB*10] [ZQC*14] [ZLQF15]	[JGT17] [GHF*18] [HGG*19] [LLH*24]	[GHDS03] [ST07] [BWR*08] [CSvRV18] [Kim20] [LKJ21] [CXY*23] [HB23] [SWP*23] [WB23] [LFFJB24]	[Jak01] [MHHR06] [GHF*07] [SL08] [SLNB10] [MKC12] [BKCW14] [MMCK14] [USS15] [MMC16] [KS16] [DKWB18] [ARM*19]
Fluids & Fluid Phenomena	[PW02] [MCG03] [SSP07] [BT07] [BIT09] [SP09] [Pri12] [SB12a] [AAT13] [ICS*14] [HWZ*15] [TDF*15] [BK17] [PT17] [YCL*17] [YML*17] [PGBT18] [WKBB18] [BKKW19] [CBG*19] [GPB*19] [WJL*20] [ZRS*20] [KBF*21] [LWB*21] [WDK*21] [LHWW22] [XRW*22] [JWL*23] [PT23] [XLYJ23] [ZLX*24] [YWX*24]	[Har62] [HW*65] [BR86] [FM96] [Sta99] [Pes02] [TUKF02] [CMT04] [ZB05] [CGFO06] [KFCO06] [CFL*07] [MCP*09] [SABS14] [SSJ*14] [ATW15] [JSS*15] [RGJ*15] [FGG*17] [GPH*18] [HFG*18] [JGT17] [ZB17] [FLGJ19] [GAB20] [HGMRT20] [TB20] [CKMR*21] [SXH*21] [QLDGJ22] [TB22] [STBA24] [QLY*23] [LLH*24] [TLZ*24]	[TB20] [TB21] [TB22] [XLYJ23]	[BLS12] [MM13] [MMCK14] [TNF14] [BGAO17] [XRW*22] [YLL*24]
Multi-Phase, Phase Transitions & Porous Flow	[MKN*04] [SSP07] [LAD08] [SP08] [BIT09] [LD09] [PC13] [RLY*14] [YCR*15] [YJL*16] [PGBT18] [CLC*20] [GHB*20] [WFM21] [RXL21] [RHLC22] [XWW*23] [YR23] [ZLX*24]	[SSJ*14] [ATW15] [GPH*18] [GAB20] [CKMR*21] [SXH*21] [LMLD22] [TLZ*24]		[MMCK14]
Other Phenomena	[LL10] [Pri12]	[WFL*19] [WDG*19] [SNZ*21] [FCK22] [CCL*22]	[CSvRV18] [CNZ*22] [WFFJB24]	[GZO10] [Cho14] [BCK*22]

Lagrangian Point-based Methods - Overview

- 👍 Simplest primitive
- 👍 Versatile description of many objects
- 👍 Suitable for topology changes
- 👍 “Locking” particle arrangements



Versatile rigid-fluid coupling for incompressible SPH [Akinci et al. 2012]

Lagrangian Point-based Methods - Method zoo

- Smoothed Particle Hydrodynamics (SPH)
- Moving-Least Squares (MLS)
- Reproducing Kernel Particle Method (RKPM)

✅ Multiphysical

✅ Graphics

- Discrete Element Method (DEM)*
- Moving Particle Semi-implicit Method (MPS)
- Peridynamics

🤔 Multiphysical

🤔 Graphics

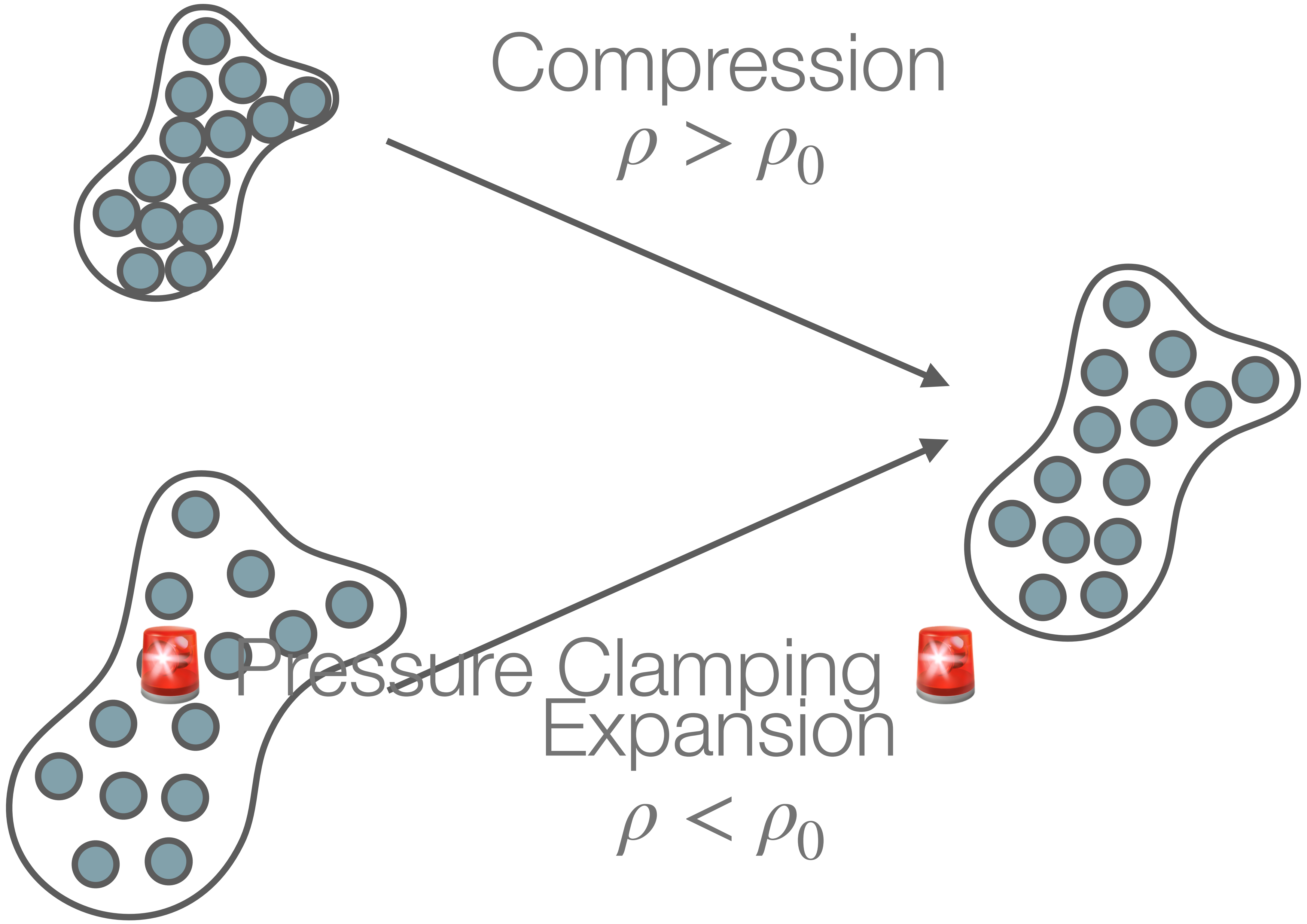
Lagrangian Point-based Methods - Equations of Motion

$$\rho \frac{D\mathbf{v}}{Dt} = \boxed{-\nabla p} + \mu \nabla^2 \mathbf{v} + \mathbf{f} \qquad \frac{D\rho}{Dt} = -\rho(\nabla \cdot \mathbf{v}) = 0$$

Pressure-Poisson Equation (PPE)

$$\boxed{\Delta t \nabla^2 p = \frac{D\rho}{Dt}}$$

$$\boxed{\Delta t \nabla^2 p = \rho(\nabla \cdot \mathbf{v})}$$

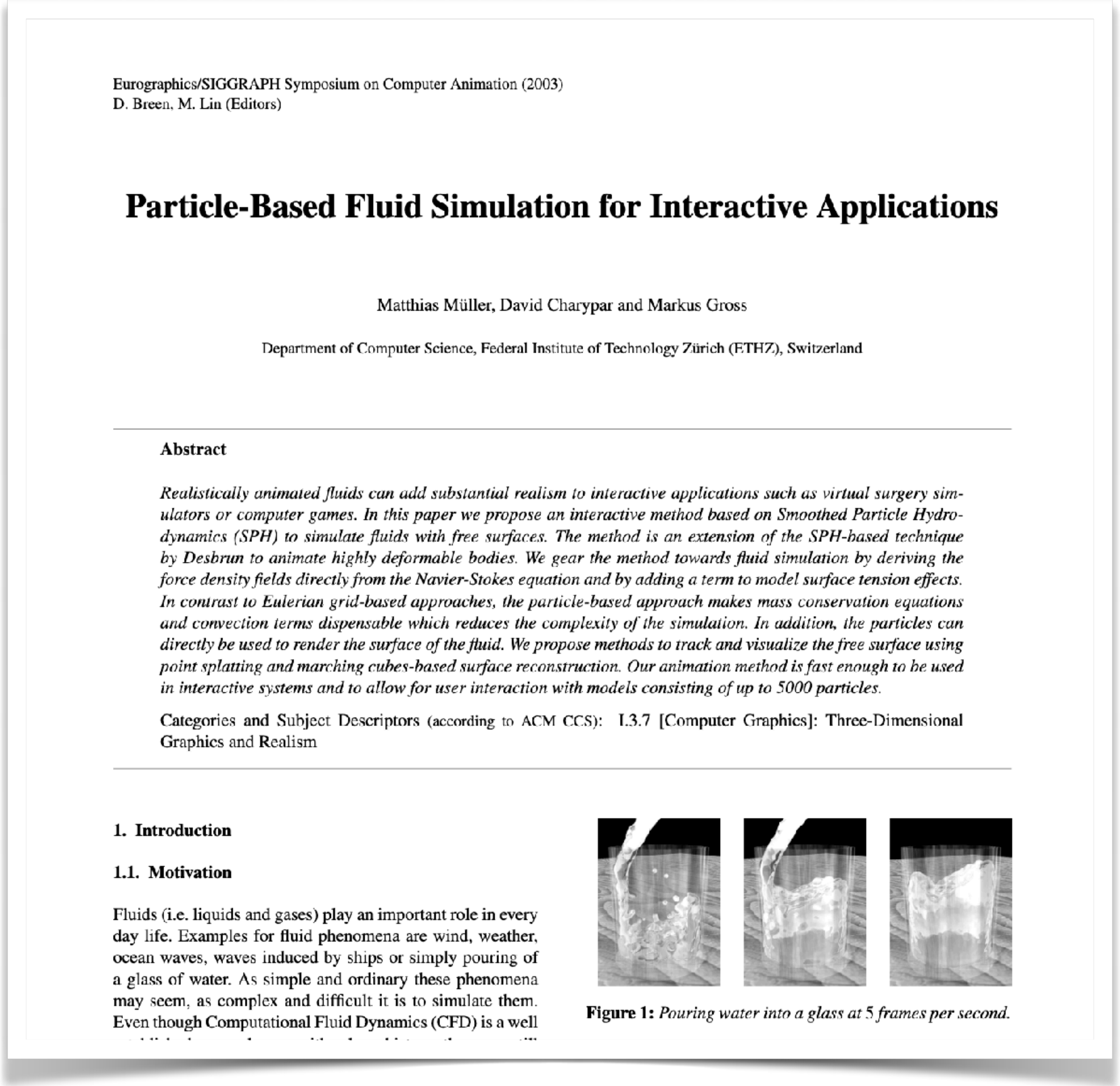


Lagrangian Point-based Methods - Fluid

$$\rho \frac{D\mathbf{v}}{Dt} = -\nabla p + \mu \nabla^2 \mathbf{v} + \mathbf{f}$$

Pressure $-\nabla p$

- Explicit [Müller et al. 2003]



Eurographics/SIGGRAPH Symposium on Computer Animation (2003)
D. Breen, M. Lin (Editors)

Particle-Based Fluid Simulation for Interactive Applications

Matthias Müller, David Charypar and Markus Gross

Department of Computer Science, Federal Institute of Technology Zürich (ETHZ), Switzerland

Abstract

Realistically animated fluids can add substantial realism to interactive applications such as virtual surgery simulators or computer games. In this paper we propose an interactive method based on Smoothed Particle Hydrodynamics (SPH) to simulate fluids with free surfaces. The method is an extension of the SPH-based technique by Desbrun to animate highly deformable bodies. We gear the method towards fluid simulation by deriving the force density fields directly from the Navier-Stokes equation and by adding a term to model surface tension effects. In contrast to Eulerian grid-based approaches, the particle-based approach makes mass conservation equations and convection terms dispensable which reduces the complexity of the simulation. In addition, the particles can directly be used to render the surface of the fluid. We propose methods to track and visualize the free surface using point splatting and marching cubes-based surface reconstruction. Our animation method is fast enough to be used in interactive systems and to allow for user interaction with models consisting of up to 5000 particles.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism

1. Introduction

1.1. Motivation

Fluids (i.e. liquids and gases) play an important role in every day life. Examples for fluid phenomena are wind, weather, ocean waves, waves induced by ships or simply pouring of a glass of water. As simple and ordinary these phenomena may seem, as complex and difficult it is to simulate them. Even though Computational Fluid Dynamics (CFD) is a well

Figure 1: *Pouring water into a glass at 5 frames per second.*

Lagrangian Point-based Methods - Fluid

$$\rho \frac{D\mathbf{v}}{Dt} = -\nabla p + \mu \nabla^2 \mathbf{v} + \mathbf{f}$$

Pressure $-\nabla p$

- Explicit [Müller et al. 2003]
- Constraint-based [Bodin et al. 2012] [Macklin and Müller 2013]


IEEE TRANSACTIONS OF VISUALIZATION AND COMPUTER GRAPHICS 1

Constraint Fluids

Kenneth Bodin, Claude Lacoursière, Martin Servin

Position Based Fluids

Miles Macklin * Matthias Müller †

NVIDIA

Abstract

In fluid simulation, enforcing incompressibility is crucial for realism; it is also computationally expensive. Recent work has improved efficiency, but still requires time-steps that are impractical for real-time applications. In this work we present an iterative density solver integrated into the Position Based Dynamics framework (PBD). By formulating and solving a set of positional constraints that enforce constant density, our method allows similar incompressibility and convergence to modern smoothed particle hydrodynamic (SPH) solvers, but inherits the stability of the geometric, position based dynamics method, allowing large time steps suitable for real-time applications. We incorporate an artificial pressure term that improves particle distribution, creates surface tension, and lowers the neighborhood requirements of traditional SPH. Finally, we address the issue of energy loss by applying vorticity confinement as a velocity post process.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation;

Keywords: fluid simulation, SPH, PCISPH, constraint fluids, position based dynamics

Links: DL PDF

(a) Real-time rendered fluid surface using ellipsoid splatting

(b) Underlying simulation particles

Figure 1: Bunny taking a bath. 128k particles, 2 sub-steps, 3 density iterations per frame, average simulation time per frame 10ms.

1 Introduction

Fluids, in particular liquids such as water, are responsible for many visually rich phenomena, and simulating them has been an area of long-standing interest and challenge in computer graphics. There are a variety of techniques available, but here we focus on particle methods, which are popular for their simplicity and flexibility.

Smoothed Particle Hydrodynamics (SPH) [Monaghan 1992][1994] is a well known particle based method for fluid simulation. It

jarola 2009], but small time steps remain a requirement, limiting real-time applications.

For interactive environments, robustness is a key issue: the simulation must handle degenerate situations gracefully. SPH algorithms often become unstable if particles do not have enough neighbors for accurate density estimates. The typical solution is to try to avoid these situations by taking sufficiently small time steps, or by using sufficiently many particles, at the cost of increased computation.

Lagrangian Point-based Methods - Fluid

$$\rho \frac{D\mathbf{v}}{Dt} = \boxed{-\nabla p} + \mu \nabla^2 \mathbf{v} + \mathbf{f}$$

Pressure $-\nabla p$

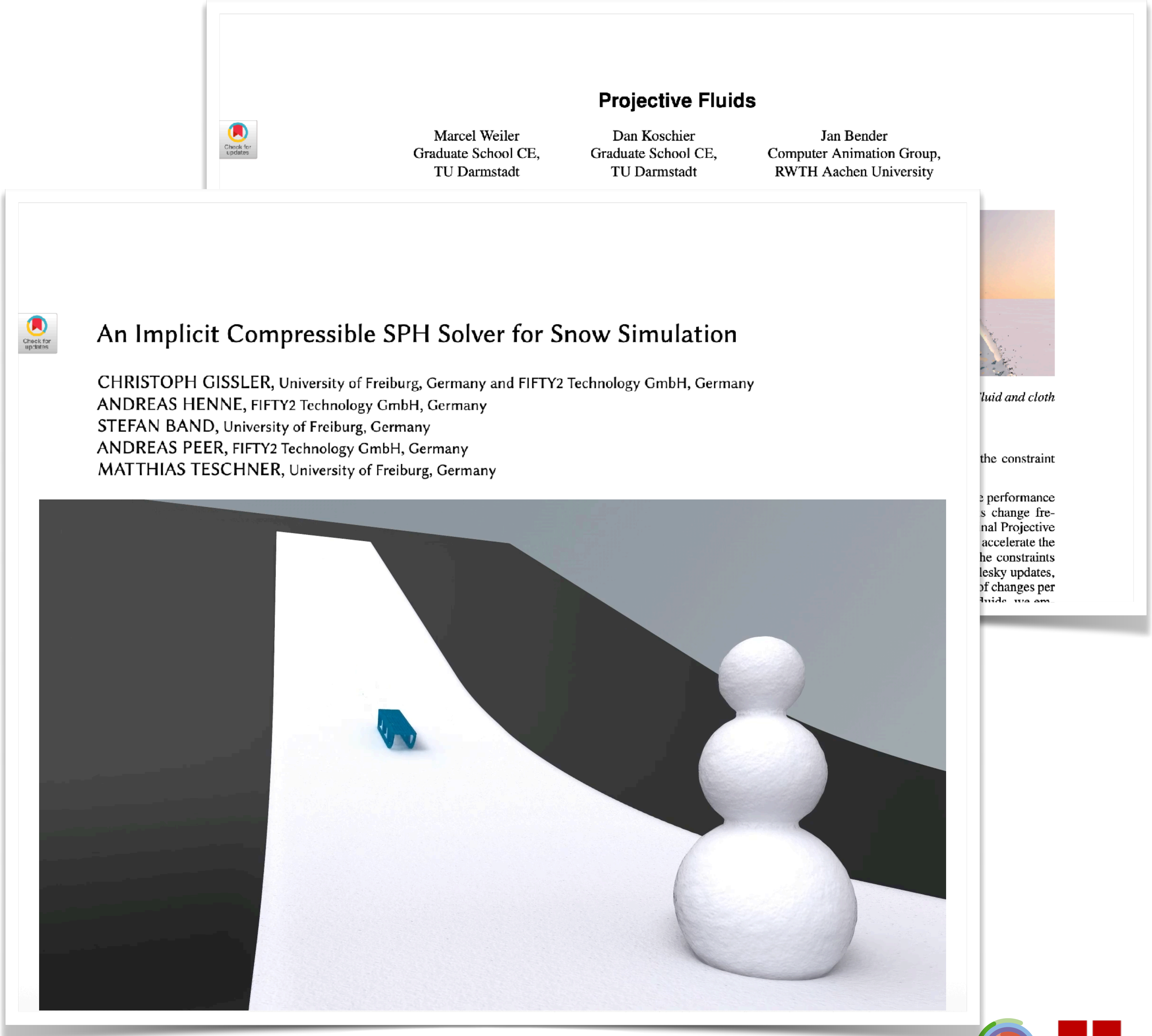
- Explicit [Müller et al. 2003]
- Constraint-based [Bodin et al. 2012] [Macklin and Müller 2013]
- Implicit [Solenthaler and Pajarola 2009] [Ihmsen et al. 2014] [Bender et al. 2017]

Lagrangian Point-based Methods - Fluid

$$\rho \frac{D\mathbf{v}}{Dt} = -\nabla p + \mu \nabla^2 \mathbf{v} + \mathbf{f}$$

Pressure $-\nabla p$

- Explicit [Müller et al. 2003]
- Constraint-based [Bodin et al. 2012] [Macklin and Müller 2013]
- Implicit [Solenthaler and Pajarola 2009] [Ihmsen et al. 2014] [Bender et al. 2017]
- Compressible [Gissler et al. 2020] [Weiler et al. 2016]


Projective Fluids

Marcel Weiler, Graduate School CE, TU Darmstadt
Dan Koschier, Graduate School CE, TU Darmstadt
Jan Bender, Computer Animation Group, RWTH Aachen University

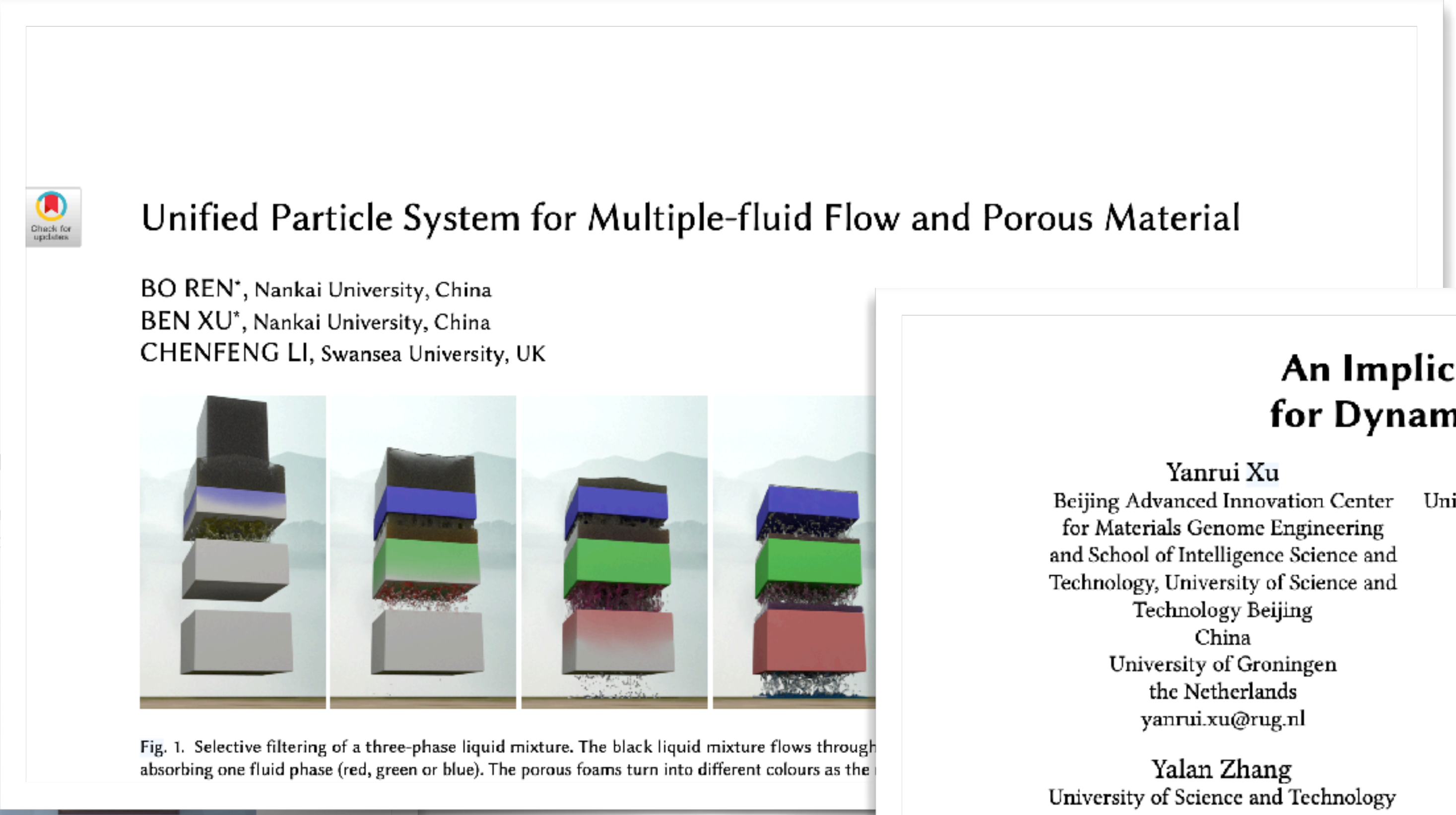
An Implicit Compressible SPH Solver for Snow Simulation

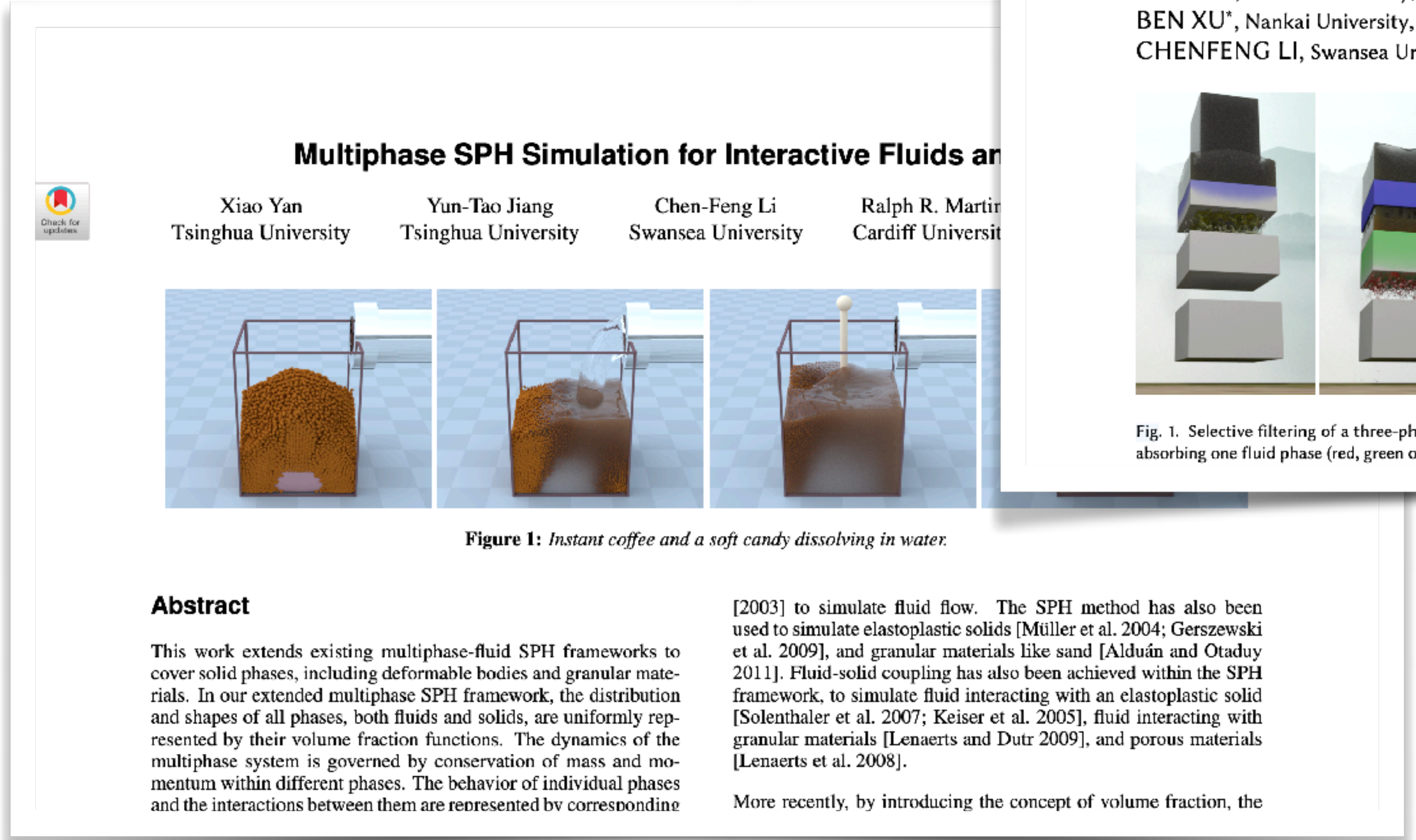
CHRISTOPH GISSLER, University of Freiburg, Germany and FIFTY2 Technology GmbH, Germany
ANDREAS HENNE, FIFTY2 Technology GmbH, Germany
STEFAN BAND, University of Freiburg, Germany
ANDREAS PEER, FIFTY2 Technology GmbH, Germany
MATTHIAS TESCHNER, University of Freiburg, Germany

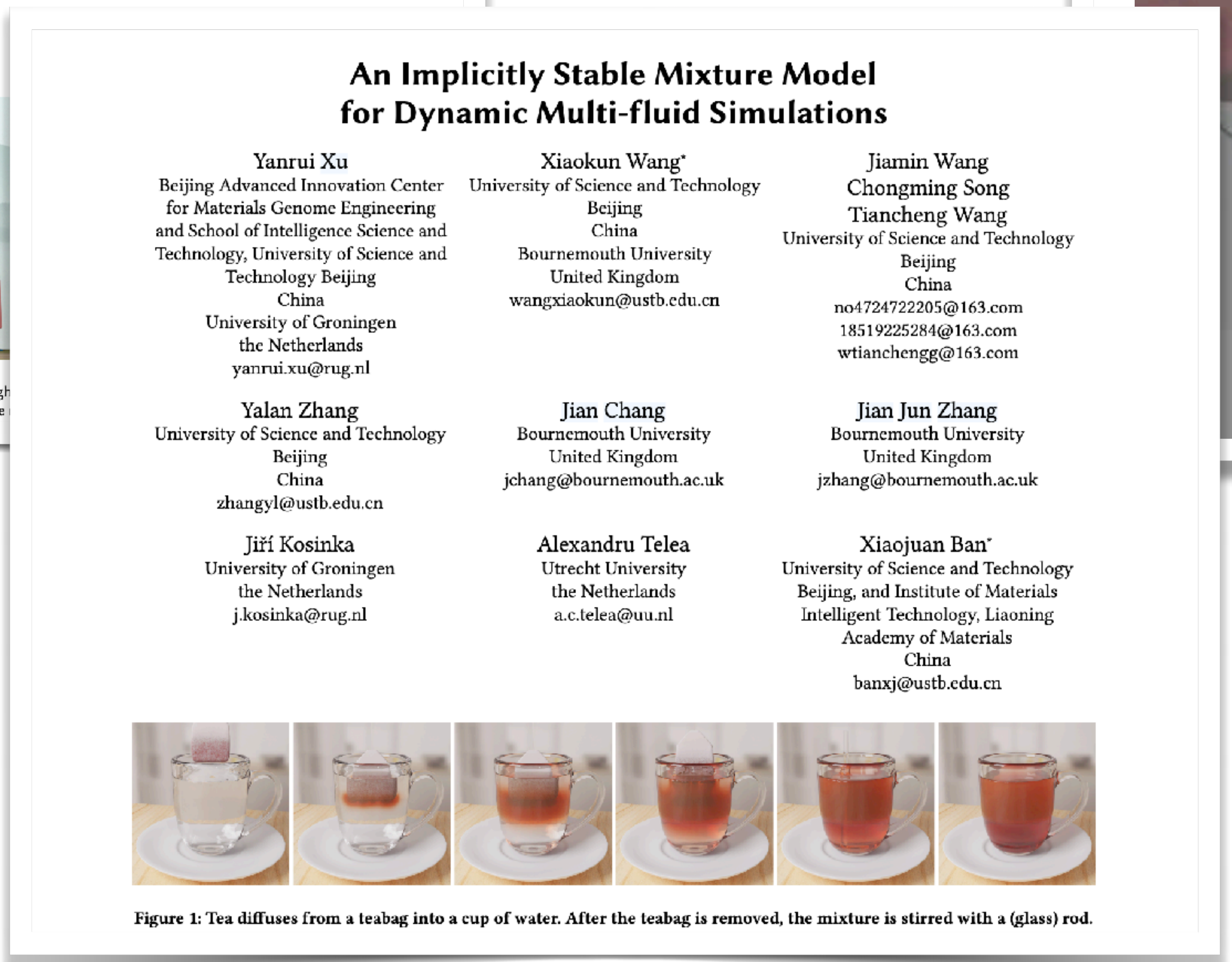
Lagrangian Point-based Methods - Fluid

$$\rho \frac{D\mathbf{v}}{Dt} = -\nabla p + \mu \nabla^2 \mathbf{v} + \mathbf{f}$$

- Multiphase
  - Multi-fluid [Solenthaler and Pajarola 2008]
  - Mixing and dissolution [Xu et al. 2023] [Ren et al. 2021] [Yan et al. 2016]



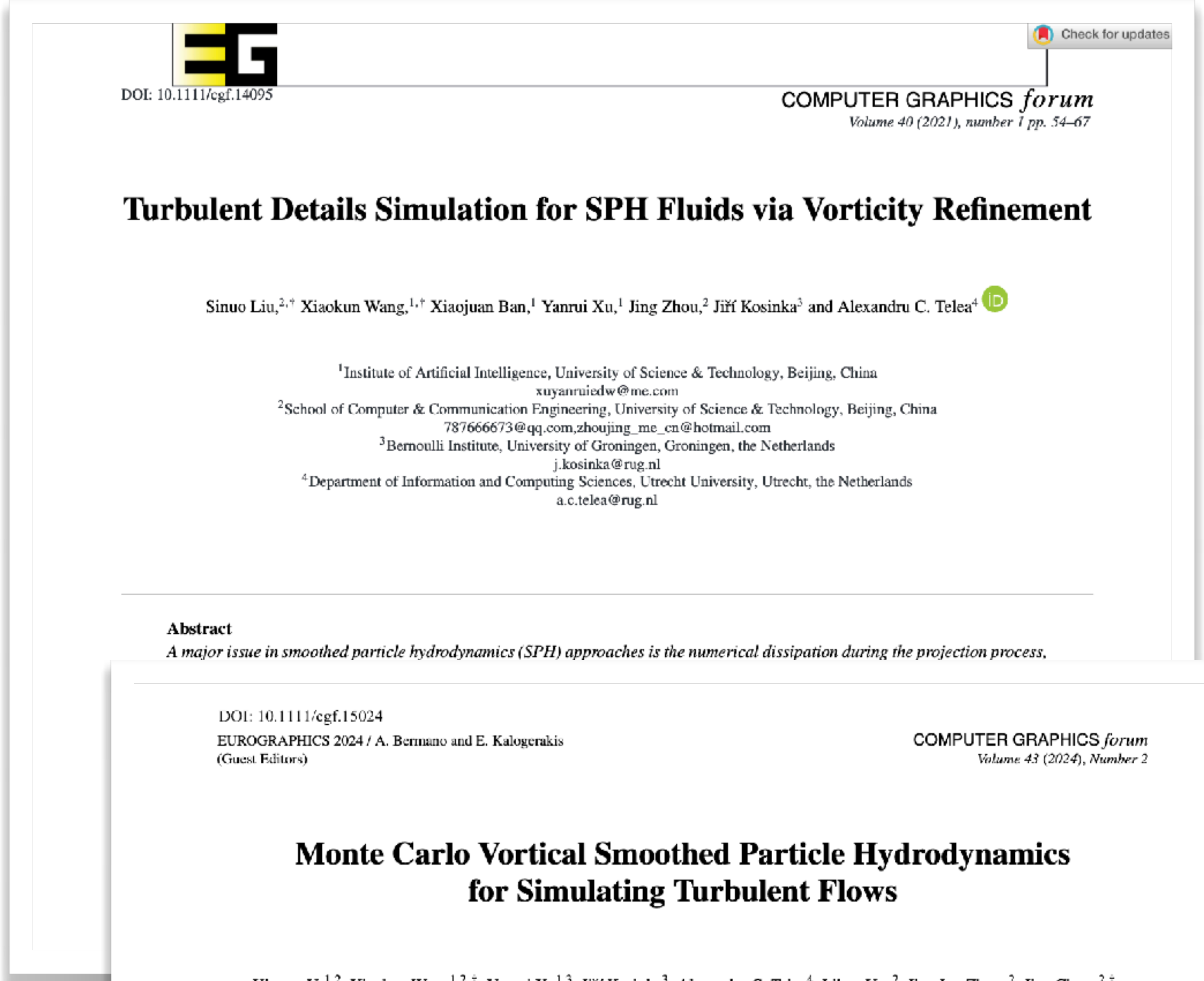




Lagrangian Point-based Methods - Fluid

$$\rho\frac{D\mathbf{v}}{Dt} = -\nabla p + \mu\nabla^2\mathbf{v} + \mathbf{f}$$

- Multiphase
  - Multi-fluid [Solenthaler and Pajarola 2008]
  - Mixing and dissolution [Xu et al. 2023] [Ren et al. 2021] [Yan et al. 2016]
- Turbulence
  - Micropolar [Bender et al. 2019]
  - Vorticity Refinement Monte Carlo [Ye et al. 2024] [Liu et al. 2021]

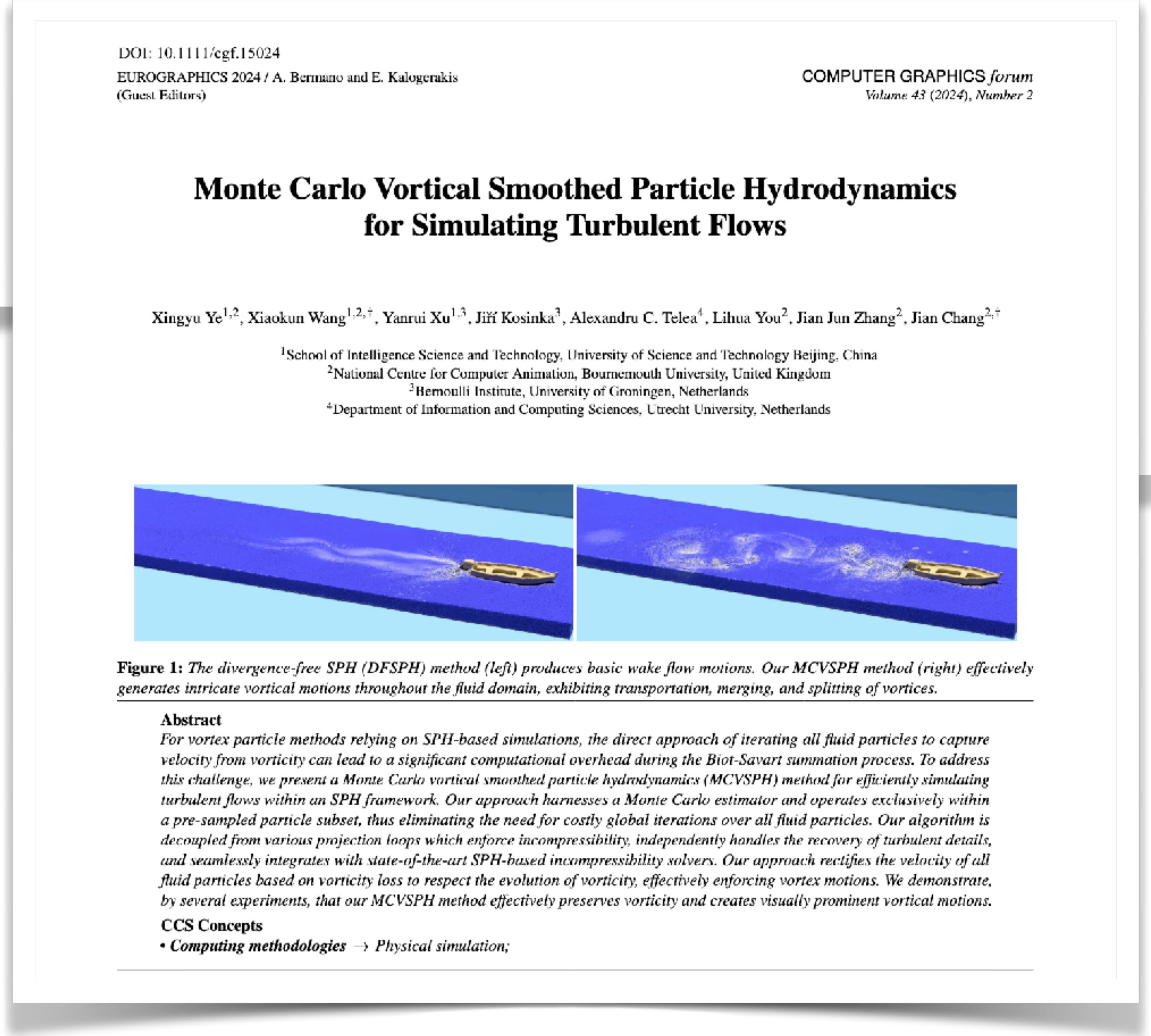

DOI: 10.1111/cgf.14095
COMPUTER GRAPHICS *forum*
Volume 40 (2021), number 1 pp. 54–67

Turbulent Details Simulation for SPH Fluids via Vorticity Refinement

Sinuo Liu,[2,†] Xiaokun Wang,[1,†] Xiaojuan Ban,[1] Yanrui Xu,[1] Jing Zhou,[2] Jiří Kosinka[3] and Alexandru C. Telea[4]

[1]Institute of Artificial Intelligence, University of Science & Technology, Beijing, China
xuyanruiedw@me.com
[2]School of Computer & Communication Engineering, University of Science & Technology, Beijing, China
787666673@qq.com,zhoujing_me_cn@hotmail.com
[3]Bernoulli Institute, University of Groningen, Groningen, the Netherlands
j.kosinka@rug.nl
[4]Department of Information and Computing Sciences, Utrecht University, Utrecht, the Netherlands
a.c.telea@rug.nl

Abstract
A major issue in smoothed particle hydrodynamics (SPH) approaches is the numerical dissipation during the projection process,


DOI: 10.1111/cgf.15024
EUROGRAPHICS 2024 / A. Bermano and E. Kalogerakis
(Guest Editors)
COMPUTER GRAPHICS *forum*
Volume 43 (2024), Number 2

Monte Carlo Vortical Smoothed Particle Hydrodynamics for Simulating Turbulent Flows

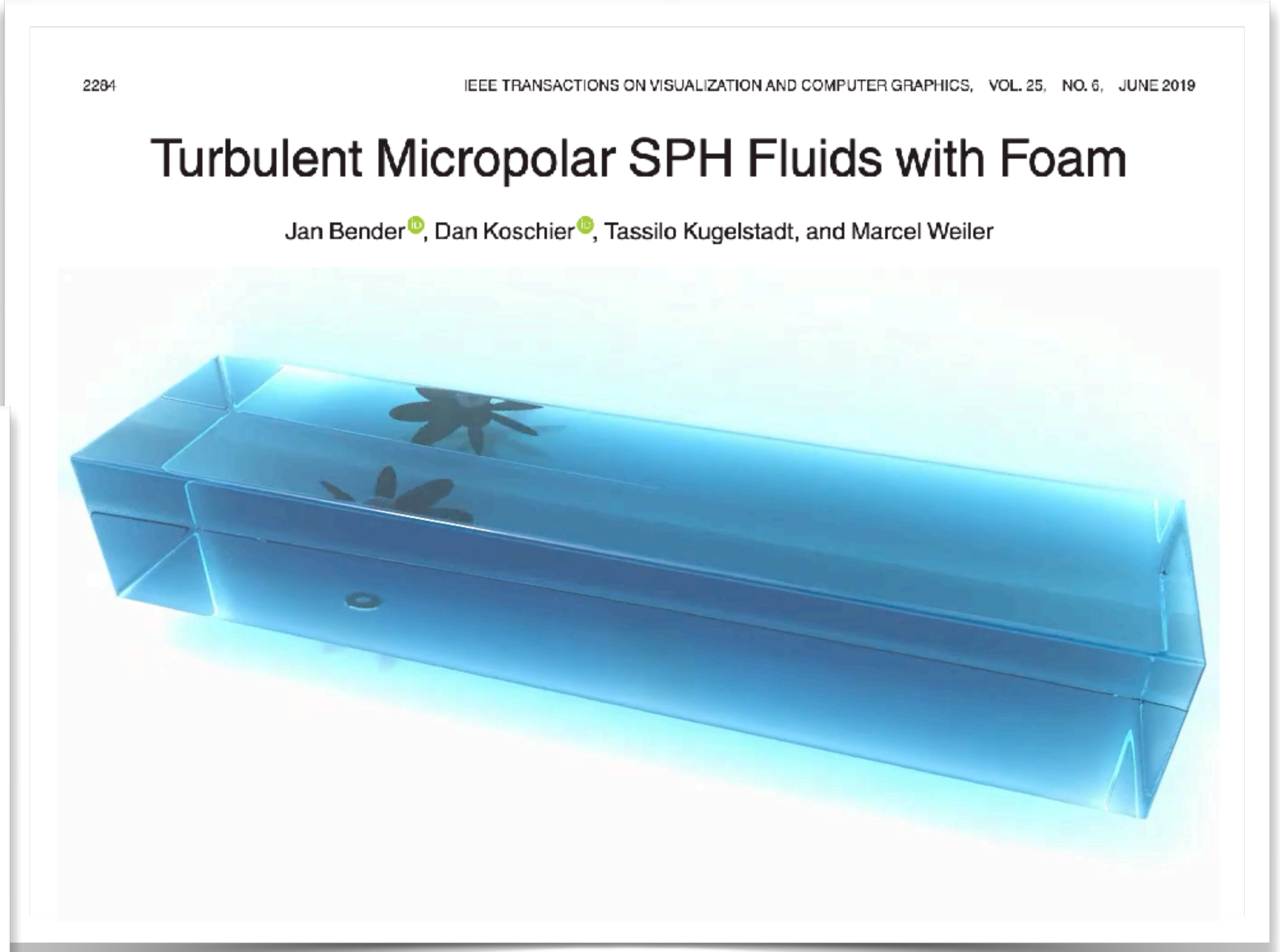
Xingyu Ye[1,2], Xiaokun Wang[1,2,†], Yanrui Xu[1,3], Jiří Kosinka[3], Alexandru C. Telea[4], Lihua You[2], Jian Jun Zhang[2], Jian Chang[2,†]

[1]School of Intelligence Science and Technology, University of Science and Technology Beijing, China
[2]National Centre for Computer Animation, Bournemouth University, United Kingdom
[3]Bernoulli Institute, University of Groningen, Netherlands
[4]Department of Information and Computing Sciences, Utrecht University, Netherlands

Figure 1: *The divergence-free SPH (DFSPH) method (left) produces basic wake flow motions. Our MCVSPH method (right) effectively generates intricate vortical motions throughout the fluid domain, exhibiting transportation, merging, and splitting of vortices.*

Abstract
For vortex particle methods relying on SPH-based simulations, the direct approach of iterating all fluid particles to capture velocity from vorticity can lead to a significant computational overhead during the Biot-Savart summation process. To address this challenge, we present a Monte Carlo vortical smoothed particle hydrodynamics (MCVSPH) method for efficiently simulating turbulent flows within an SPH framework. Our approach harnesses a Monte Carlo estimator and operates exclusively within a pre-sampled particle subset, thus eliminating the need for costly global iterations over all fluid particles. Our algorithm is decoupled from various projection loops which enforce incompressibility, independently handles the recovery of turbulent details, and seamlessly integrates with state-of-the-art SPH-based incompressibility solvers. Our approach rectifies the velocity of all fluid particles based on vorticity loss to respect the evolution of vorticity, effectively enforcing vortex motions. We demonstrate, by several experiments, that our MCVSPH method effectively preserves vorticity and creates visually prominent vortical motions.

CCS Concepts
• ***Computing methodologies*** → *Physical simulation;*


2294 IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 25, NO. 6, JUNE 2019

Turbulent Micropolar SPH Fluids with Foam

Jan Bender, Dan Koschier, Tassilo Kugelstadt, and Marcel Weiler

Lagrangian Point-based Methods - Fluid

$$\rho \frac{D\mathbf{v}}{Dt} = -\nabla p + \mu \nabla^2 \mathbf{v} + \mathbf{f}$$

- Multiphase
  - Multi-fluid [Solenthaler and Pajarola 2008]
  - Mixing and dissolution [Xu et al. 2023] [Ren et al. 2021] [Yan et al. 2016]
- Turbulence
  - Micropolar [Bender et al. 2019]
  - Vorticity Refinement Monte Carlo [Ye et al. 2024] [Liu et al. 2021]
- Surface Tension
  - Curvature [Xing et al. 2022] [Zorilla et al. 2020] [Müller et al. 2003]
  - Cohesion [Jeske et al. 2023] [Yang et al. 2017]

Lagrangian Point-based Methods - Rigid

- Sample with Particles
  - Force-based [Becker et al. 2009]
  - Pressure-based [Akinci et al. 2013] [Akinci et al. 2012]
  - Rigid-rigid & rigid-fluid [Probst and Teschner 2023] [Gissler et al. 2019]

Lagrangian Point-based Methods - Rigid

- Sample with Particles
  - Force-based [Becker et al. 2009]
  - Pressure-based [Akinci et al. 201…] [Akinci et al. 2012]
  - Rigid-rigid & rigid-fluid [Probst and Teschner 2023] [Gissler et al. 2019]
- Implicit boundaries [Winchenbach et al. 2020] [Bender et al. 2020] [Fujisawa and Miura 2015]

An Efficient Boundary Handling with a Modified Density Calculation for SPH

Makoto Fujisawa[1] and Kenjiro T. Miura[2]

[1]University of Tsukuba, Japan
[2]Shizuoka University, Japan

Semi-Analytic Boundary Handling Below Particle Resolution for Smoothed Particle Hydrodynamics

RENE WINCHENBACH, University of Siegen
RUSTAM AKHUNOV, University of Siegen
ANDREAS KOLB, University of Siegen

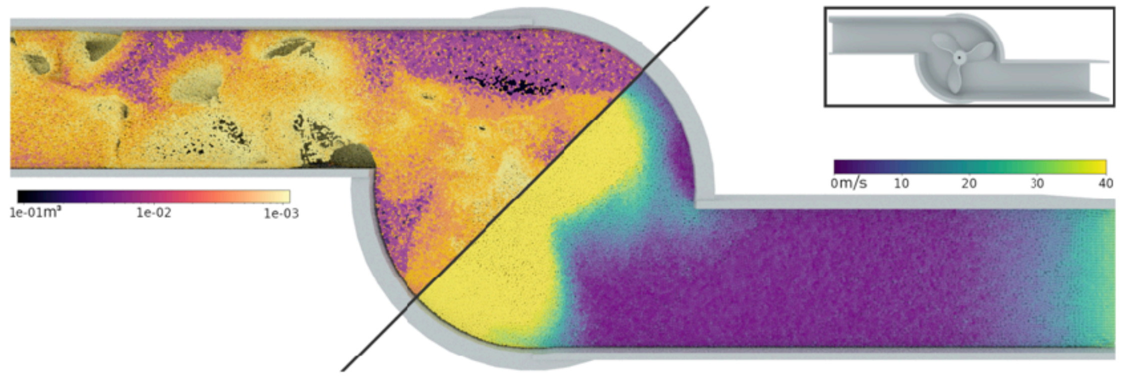
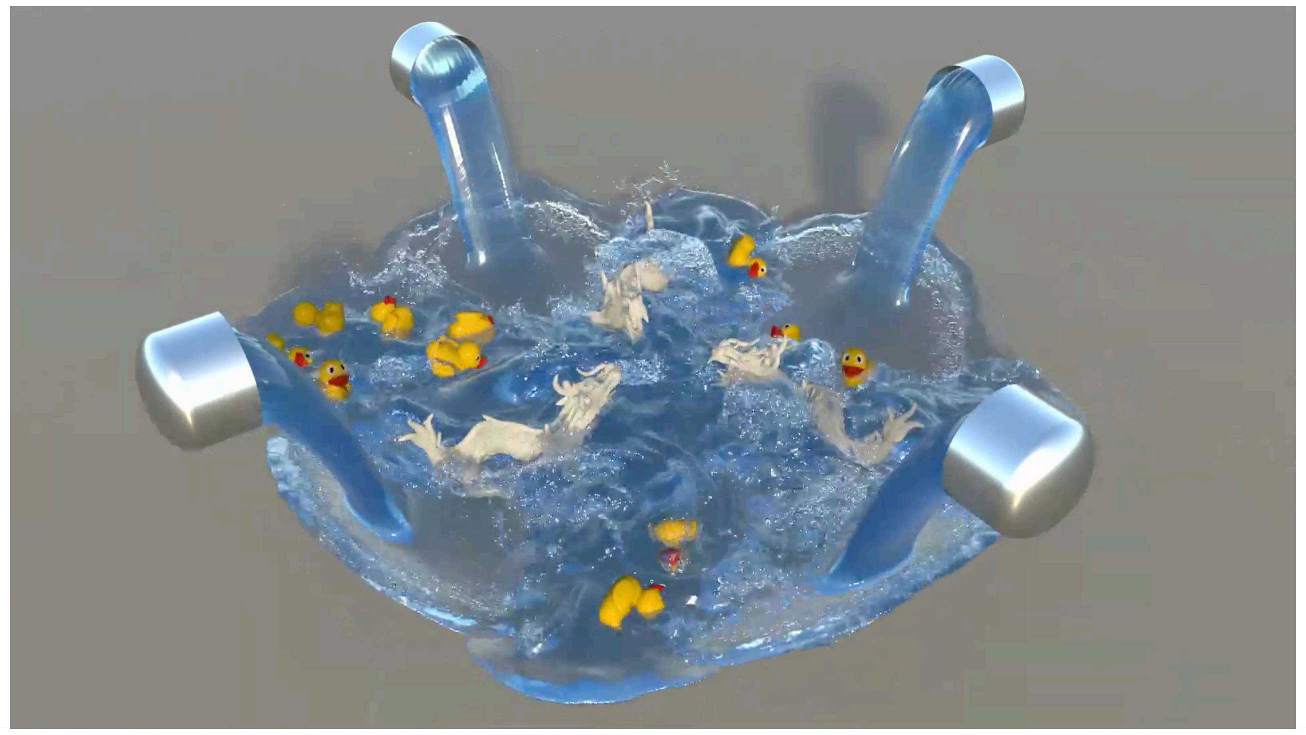


Fig. 1. Our novel semi-analytical boundary handling method enables fluid-rigid interactions even under difficult conditions and can be directly combined with any spatially adaptive simulation technique.

2982 IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 26, NO. 10, OCTO…

Implicit Frictional Boundary Handling for SPH

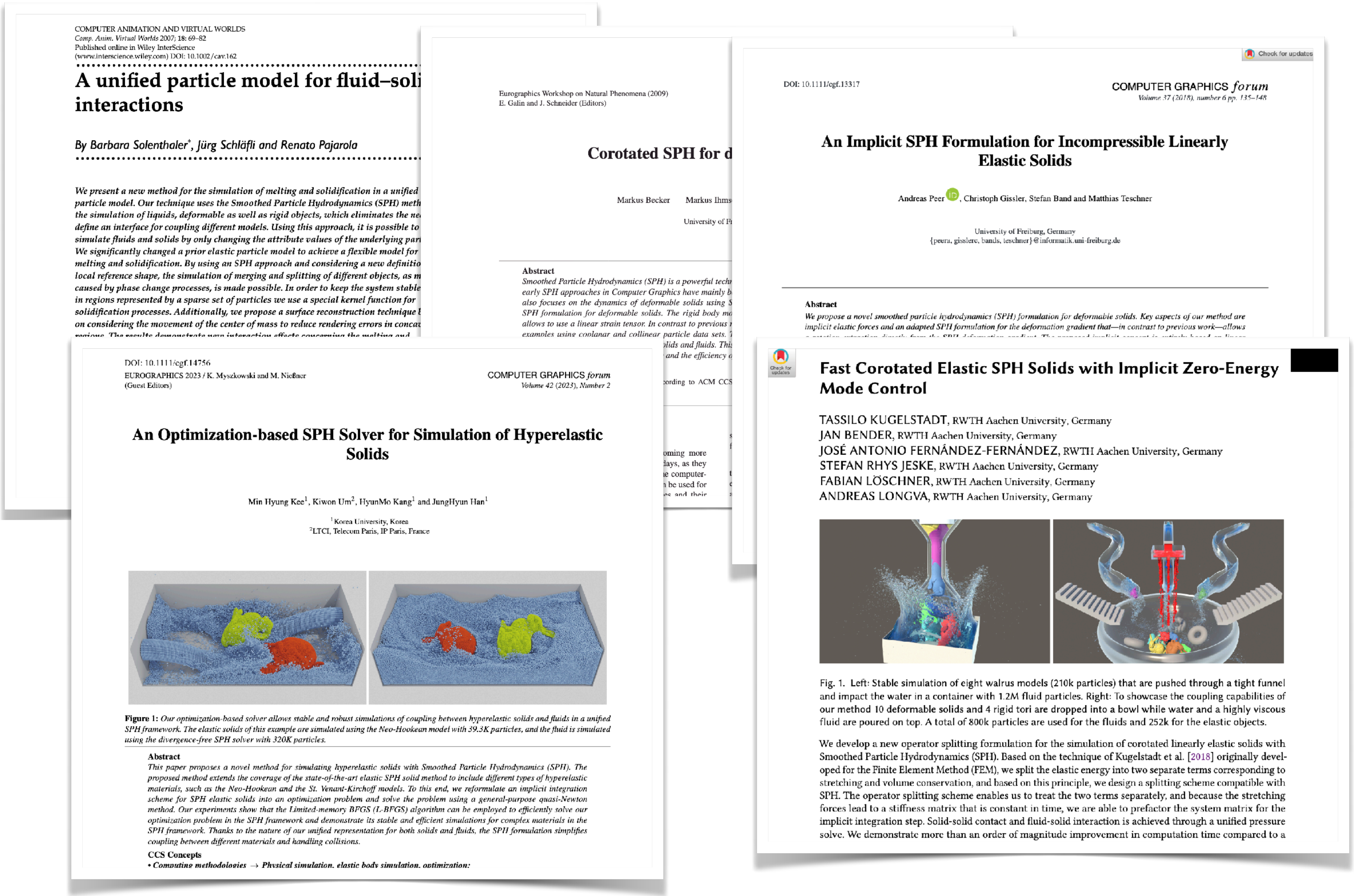
Jan Bender, Tassilo Kugelstadt, Marcel Weiler, and Dan Koschier



Lagrangian Point-based Methods - Solids

$$\rho \frac{D\mathbf{v}}{Dt} = \boxed{\nabla \cdot \sigma} + f$$

- Elasticity
  - Linear [Solenthaler et al. 2007]
  - Corotational [Becker et al. 2009]
  - Implicit coronated linear [Kugelstadt et al. 2021] [Peer et al. 2018]
  - Non-linear [Kee et al. 2023]



Lagrangian Point-based Methods - Solids

$$\rho \frac{D\mathbf{v}}{Dt} = \boxed{\nabla \cdot \sigma} + f$$

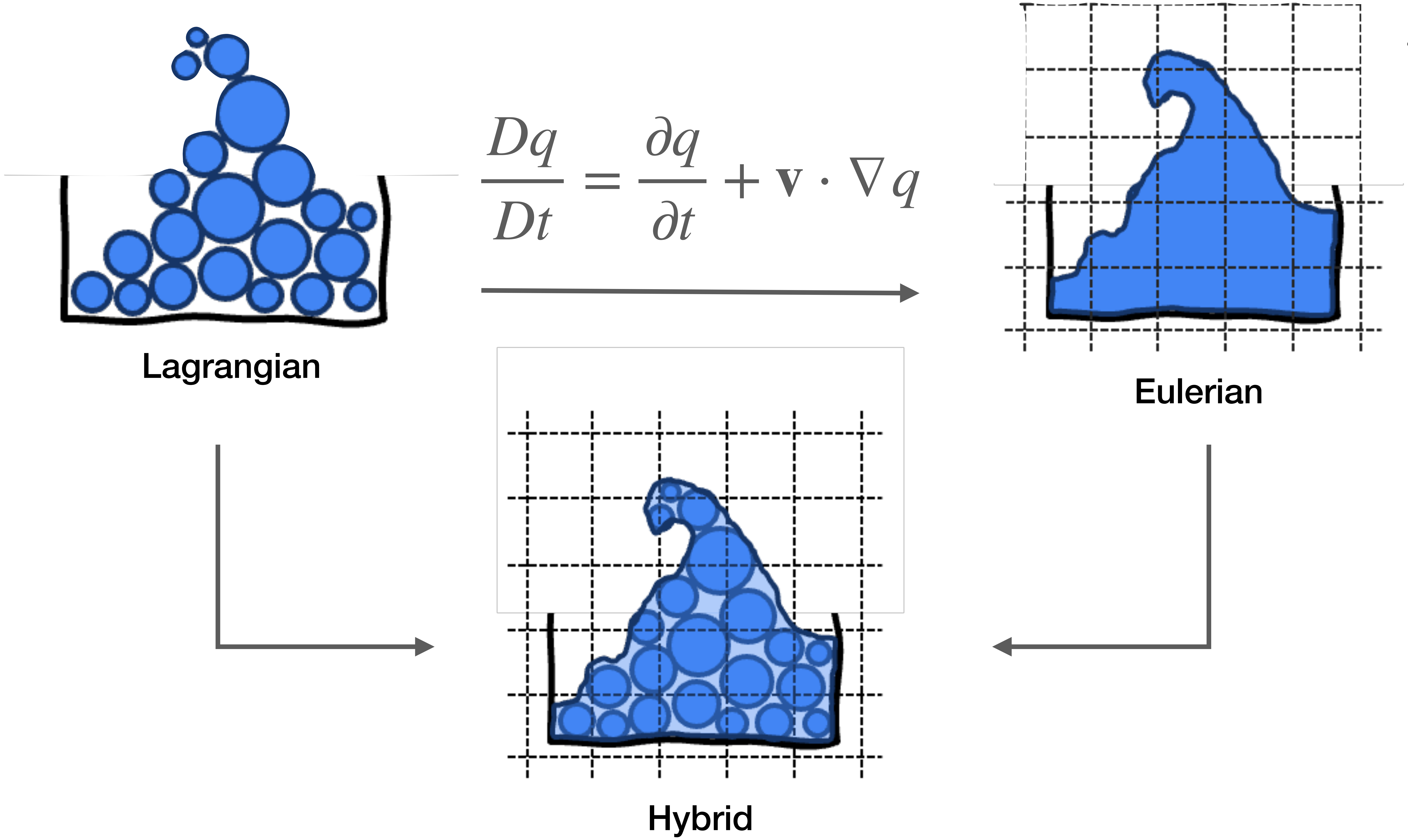
- Elasticity
  - Linear [Solenthaler et al. 2007]
  - Corotational [Becker et al. 2009]
  - Implicit coronated linear [Kugelstadt et al. 2021] [Peer et al. 2018]
  - Non-linear [Kee et al. 2023]
- Plasticity
  - Mohr Coulomb [Lenaerts and Dutré 2009]
  - Drucker Prager [Alduán and Otaduy 2011] [Ihmsen et al. 2013]
  - Snow [Gissler et al. 2020]

Overview

- Lagrangian Point-based Methods
  - Overview
  - Fluid
  - Solid and Rigid
  - Multiphysics Materials
- Eulerian and Hybrid Methods
  - Overview
  - Multiphysics Materials
- Summary

	Lagrangian Point-Based Methods (Sec. 2)	Eulerian & Hybrid Methods (Sec. 3)	Energy-Based Modeling (Sec. 4)	Constraint-Based Modeling (Sec. 5)
Deformables (elastic & plastic)	[MKN*04] [PKA*05] [SSP07] [BIT09] [MKB*10] [YJL*16] [YCL*17] [PGBT18] [CLC*20] [KBF*21] [KUKH23]	[SZS95] [CGFO06] [LLJ*11] [SSJ*14] [JSS*15] [YSB*15] [TLK16] [FGG*17] [GTJS17] [JGT17] [ZB17] [GHF*18] [HFG*18] [FLGJ19] [HGG*19] [SXH*21] [LLJ22] [TB22] [QLY*23] [LLH*24] [TLZ*24]	[BAV*10] [BUAG12] [SB12b] [SHST12] [BML*14] [GSS*15] [LBK17] [BOFN18] [SGK18] [LFS*20] [MEM*20] [LMY*22] [LCK22] [LLJ22] [KE22] [LFFJ*23]	[Jak01] [MHTG05] [MHHR06] [SLM06] [MMCK14] [BKCW14] [Cho14] [MCKM15] [CMM16] [DCB16] [MMC16] [BGAO17] [FM17] [ARM*19] [MEM*19] [WWB*19] [MMC*20] [MM21] [TTKA23] [CHC*24a] [Cet24] [MAK24] [SZDJ24] [YLL*24]
Granular Materials	[LD09] [AO11] [IWT13] [YJL*16] [YCL*17] [GHB*20]	[ZB05] [SSC*13] [DBD16] [KGP*16] [TGK*17] [GPH*18]		[Hol14] [MMCK14] [SWLB14] [FM17] [HG18] [NS18] [KKHS20] [YLL*24]
Rigid Bodies & Multibody Systems	[SSP07] [YCL*17] [GPB*19] [PT23]	[TB20] [TB22] [LLH*24] [TLZ*24]	[CDGB19] [MEM*20] [FLS*21] [CLL*22] [LKL*22]	[Bar94] [MC95] [ST96] [Bar96] [AP97] [Ste00] [Jak01] [Erl05] [MHTG05] [Lac07b, Lac07a] [GZO10] [MMCK14] [DCB16] [FM17] [MEM*19] [PAK*19] [WWB*19] [MMC*20] [MAK24]
Co-dimensional Structures	[MKB*10] [ZQC*14] [ZLQF15]	[JGT17] [GHF*18] [HGG*19] [LLH*24]	[GHDS03] [ST07] [BWR*08] [CSvRV18] [Kim20] [LKJ21] [CXY*23] [HB23] [SWP*23] [WB23] [LFFJB24]	[Jak01] [MHHR06] [GHF*07] [SL08] [SLNB10] [MKC12] [BKCW14] [MMCK14] [USS15] [MMC16] [KS16] [DKWB18] [ARM*19]
Fluids & Fluid Phenomena	[PW02] [MCG03] [SSP07] [BT07] [BIT09] [SP09] [Pri12] [SB12a] [AAT13] [ICS*14] [HWZ*15] [TDF*15] [BK17] [PT17] [YCL*17] [YML*17] [PGBT18] [WKBB18] [BKKW19] [CBG*19] [GPB*19] [WJL*20] [ZRS*20] [KBF*21] [LWB*21] [WDK*21] [LHWW22] [XRW*22] [JWL*23] [PT23] [XLYJ23] [ZLX*24] [YWX*24]	[Har62] [HW*65] [BR86] [FM96] [Sta99] [Pes02] [TUKF02] [CMT04] [ZB05] [CGFO06] [KFCO06] [CFL*07] [MCP*09] [SABS14] [SSJ*14] [ATW15] [JSS*15] [RGJ*15] [FGG*17] [GPH*18] [HFG*18] [JGT17] [ZB17] [FLGJ19] [GAB20] [HGMRT20] [TB20] [CKMR*21] [SXH*21] [QLDGJ22] [TB22] [STBA24] [QLY*23] [LLH*24] [TLZ*24]	[TB20] [TB21] [TB22] [XLYJ23]	[BLS12] [MM13] [MMCK14] [TNF14] [BGAO17] [XRW*22] [YLL*24]
Multi-Phase, Phase Transitions & Porous Flow	[MKN*04] [SSP07] [LAD08] [SP08] [BIT09] [LD09] [PC13] [RLY*14] [YCR*15] [YJL*16] [PGBT18] [CLC*20] [GHB*20] [WFM21] [RXL21] [RHLC22] [XWW*23] [YR23] [ZLX*24]	[SSJ*14] [ATW15] [GPH*18] [GAB20] [CKMR*21] [SXH*21] [LMLD22] [TLZ*24]		[MMCK14]
Other Phenomena	[LL10] [Pri12]	[WFL*19] [WDG*19] [SNZ*21] [FCK22] [CCL*22]	[CSvRV18] [CNZ*22] [WFFJB24]	[GZO10] [Cho14] [BCK*22]

Eulerian and Hybrid Methods - Overview

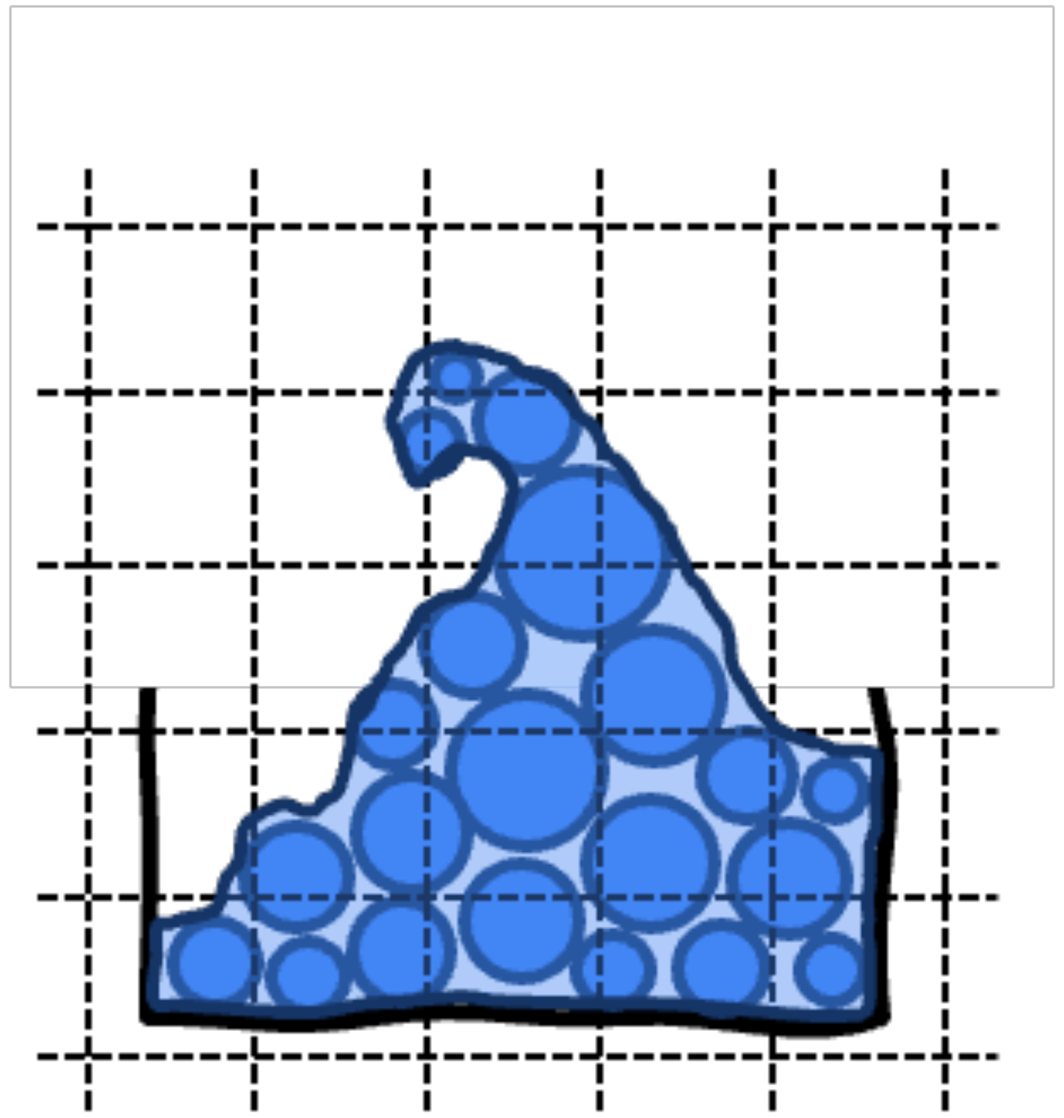


$$\frac{Dq}{Dt} = \frac{\partial q}{\partial t} + \mathbf{v} \cdot \nabla q$$

```
Algorithm [Bridson 2008]
=========
1. Explicit Grid Force
   Computation
2. Pressure Projection
3. Advection
```

- 👍 Spatial Derivatives
- 👍 Energy conservation
- 👍 Boundary conditions
- 👍 Parallelization

Eulerian and Hybrid Methods - Overview



Hybrid

Lagrangian 🔗 Eulerian = MPM

✨Mass conservation

✨Transient grid

✨Parallelization

✨GPU suitable

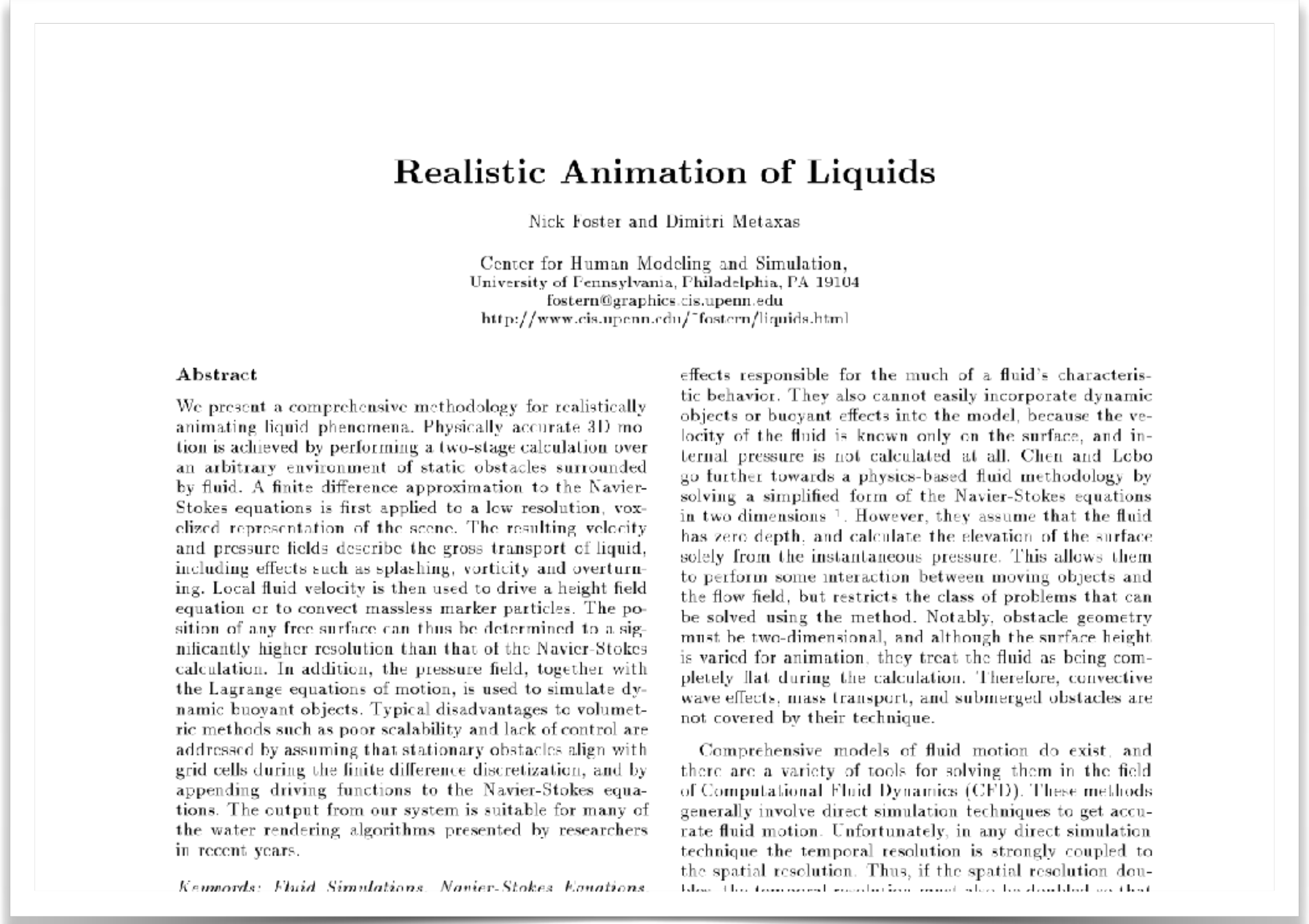
Algorithm [Jiang et al. 2016]
=========

1. Particle-to-Grid
2. Grid Velocity Computation
3. Explicit Grid Force Computation
4. Grid Velocity Update
5. Deformation Gradient Update
6. Grid-to-Particle Transfer
7. Particle Advection

Eulerian (steps 3–5)

Eulerian and Hybrid Methods - Eulerian Coupling

- Rasterize [Takahashi et al. 2002]
  [Foster and Metaxas 1996]



Realistic Animation of Liquids

Nick Foster and Dimitri Metaxas

Center for Human Modeling and Simulation,
University of Pennsylvania, Philadelphia, PA 19104
fostern@graphics.cis.upenn.edu
http://www.cis.upenn.edu/~fostern/liquids.html

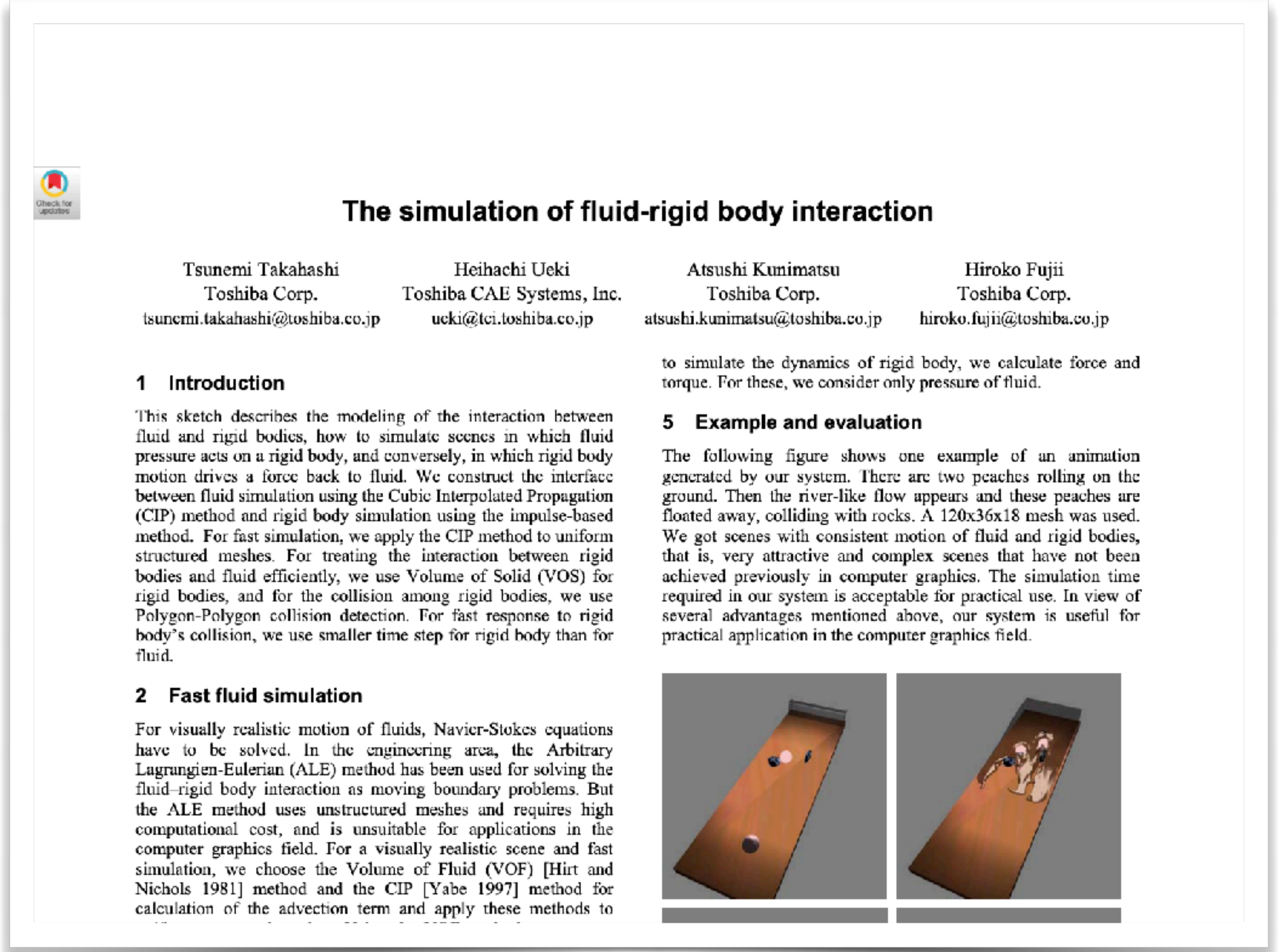
Abstract

We present a comprehensive methodology for realistically animating liquid phenomena. Physically accurate 3D motion is achieved by performing a two-stage calculation over an arbitrary environment of static obstacles surrounded by fluid. A finite difference approximation to the Navier-Stokes equations is first applied to a low resolution, voxelized representation of the scene. The resulting velocity and pressure fields describe the gross transport of liquid, including effects such as splashing, vorticity and overturning. Local fluid velocity is then used to drive a height field equation or to convect massless marker particles. The position of any free surface can thus be determined to a significantly higher resolution than that of the Navier-Stokes calculation. In addition, the pressure field, together with the Lagrange equations of motion, is used to simulate dynamic buoyant objects. Typical disadvantages to volumetric methods such as poor scalability and lack of control are addressed by assuming that stationary obstacles align with grid cells during the finite difference discretization, and by appending driving functions to the Navier-Stokes equations. The output from our system is suitable for many of the water rendering algorithms presented by researchers in recent years.

Keywords: Fluid Simulations, Navier-Stokes Equations,

effects responsible for the much of a fluid's characteristic behavior. They also cannot easily incorporate dynamic objects or buoyant effects into the model, because the velocity of the fluid is known only on the surface, and internal pressure is not calculated at all. Chen and Lobo go further towards a physics-based fluid methodology by solving a simplified form of the Navier-Stokes equations in two dimensions [1]. However, they assume that the fluid has zero depth, and calculate the elevation of the surface solely from the instantaneous pressure. This allows them to perform some interaction between moving objects and the flow field, but restricts the class of problems that can be solved using the method. Notably, obstacle geometry must be two-dimensional, and although the surface height is varied for animation, they treat the fluid as being completely flat during the calculation. Therefore, convective wave effects, mass transport, and submerged obstacles are not covered by their technique.

Comprehensive models of fluid motion do exist, and there are a variety of tools for solving them in the field of Computational Fluid Dynamics (CFD). These methods generally involve direct simulation techniques to get accurate fluid motion. Unfortunately, in any direct simulation technique the temporal resolution is strongly coupled to the spatial resolution. Thus, if the spatial resolution dou-



The simulation of fluid-rigid body interaction

Tsunemi Takahashi, Toshiba Corp., tsunemi.takahashi@toshiba.co.jp
Heihachi Ueki, Toshiba CAE Systems, Inc., ueki@tci.toshiba.co.jp
Atsushi Kunimatsu, Toshiba Corp., atsushi.kunimatsu@toshiba.co.jp
Hiroko Fujii, Toshiba Corp., hiroko.fujii@toshiba.co.jp

1 Introduction

This sketch describes the modeling of the interaction between fluid and rigid bodies, how to simulate scenes in which fluid pressure acts on a rigid body, and conversely, in which rigid body motion drives a force back to fluid. We construct the interface between fluid simulation using the Cubic Interpolated Propagation (CIP) method and rigid body simulation using the impulse-based method. For fast simulation, we apply the CIP method to uniform structured meshes. For treating the interaction between rigid bodies and fluid efficiently, we use Volume of Solid (VOS) for rigid bodies, and for the collision among rigid bodies, we use Polygon-Polygon collision detection. For fast response to rigid body's collision, we use smaller time step for rigid body than for fluid.

2 Fast fluid simulation

For visually realistic motion of fluids, Navier-Stokes equations have to be solved. In the engineering area, the Arbitrary Lagrangien-Eulerian (ALE) method has been used for solving the fluid–rigid body interaction as moving boundary problems. But the ALE method uses unstructured meshes and requires high computational cost, and is unsuitable for applications in the computer graphics field. For a visually realistic scene and fast simulation, we choose the Volume of Fluid (VOF) [Hirt and Nichols 1981] method and the CIP [Yabe 1997] method for calculation of the advection term and apply these methods to

to simulate the dynamics of rigid body, we calculate force and torque. For these, we consider only pressure of fluid.

5 Example and evaluation

The following figure shows one example of an animation generated by our system. There are two peaches rolling on the ground. Then the river-like flow appears and these peaches are floated away, colliding with rocks. A 120x36x18 mesh was used. We got scenes with consistent motion of fluid and rigid bodies, that is, very attractive and complex scenes that have not been achieved previously in computer graphics. The simulation time required in our system is acceptable for practical use. In view of several advantages mentioned above, our system is useful for practical application in the computer graphics field.

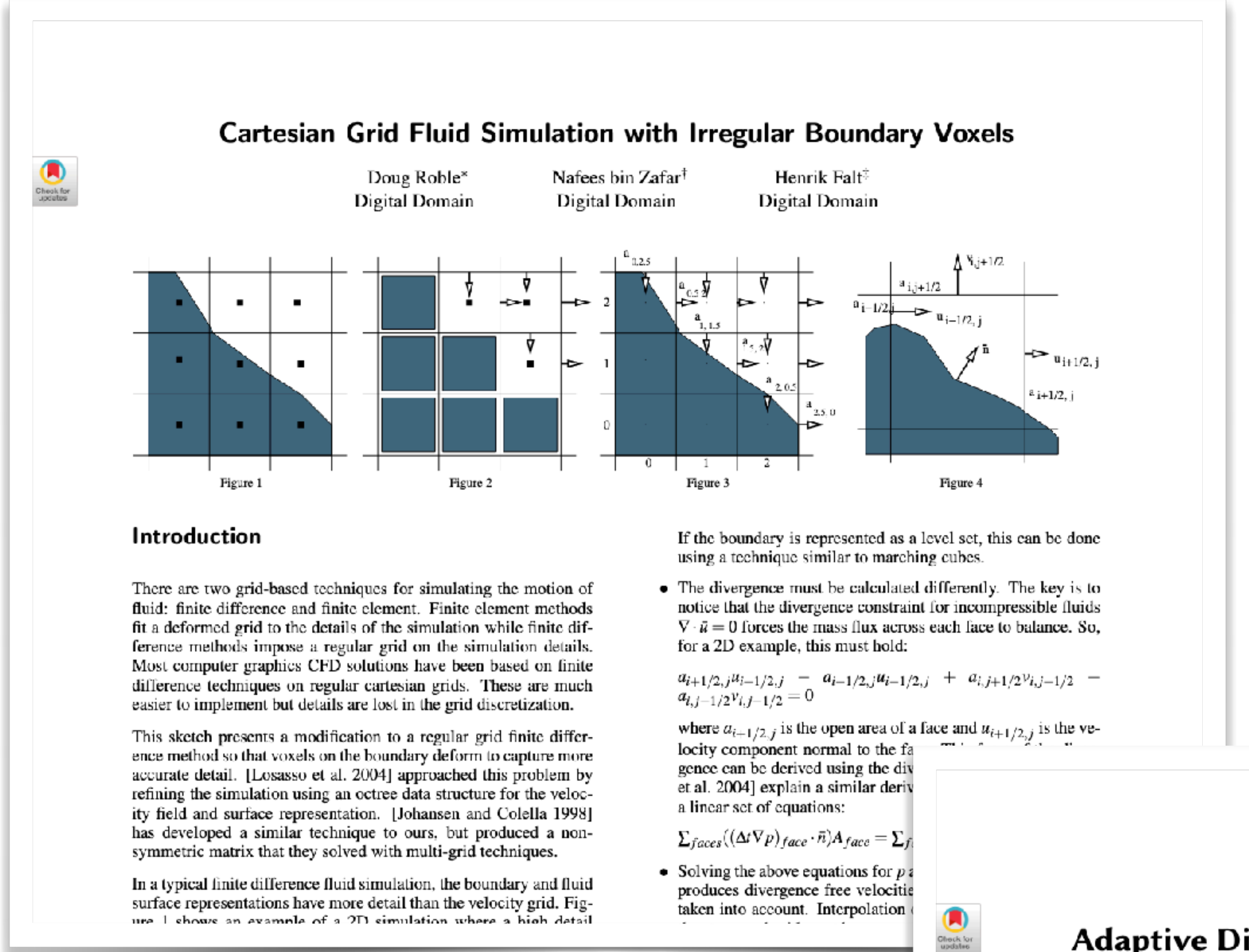
Eulerian and Hybrid Methods - Eulerian Coupling

- Rasterize [Takahashi et al. 2002]
  [Foster and Metaxas 1996]

- Immersed boundary method
  [Li et al. 2020] [Carlson et al. 2004] [Peskin 2002]

Eulerian and Hybrid Methods - Eulerian Coupling

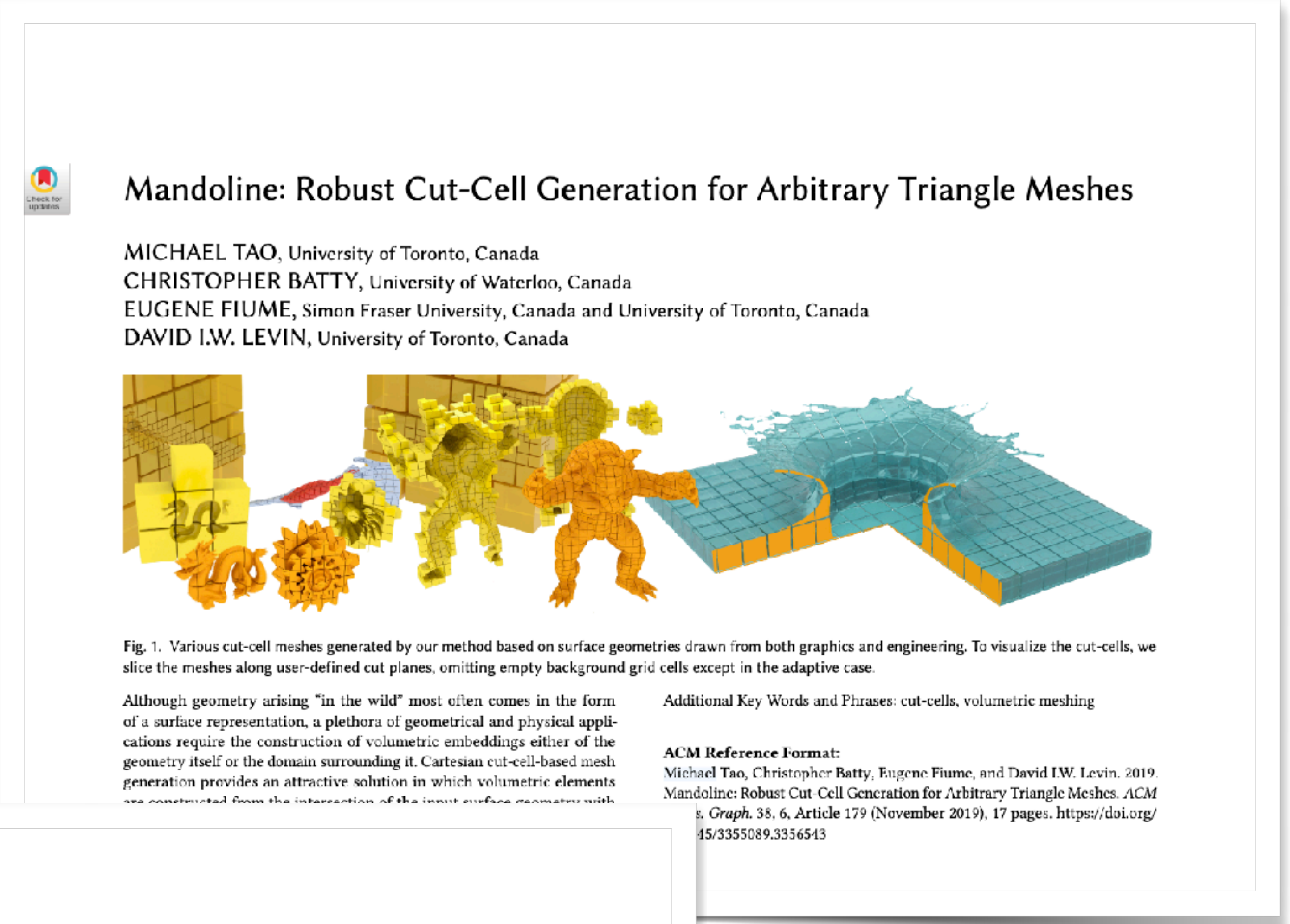
- Rasterize [Takahashi et al. 2002] [Foster and Metaxas 1996]
- Immersed boundary method [Li et al. 2020] [Carlson et al. 2004] [Peskin 2002]
- Cut-Cell [Shi et al. 2024] [Tao et al. 2019] [Roble et al. 2005]


Cartesian Grid Fluid Simulation with Irregular Boundary Voxels

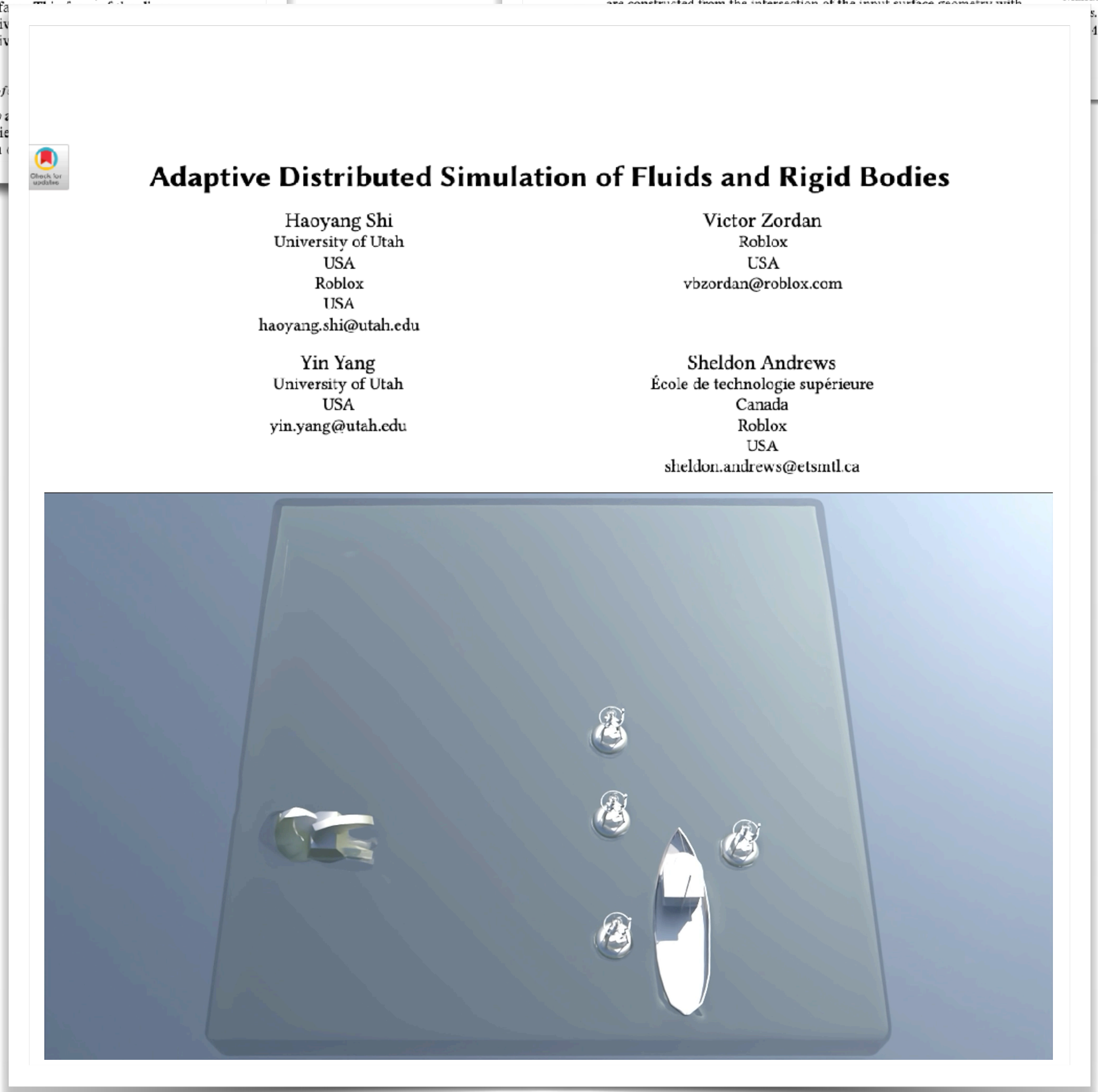
Doug Roble — Digital Domain
Nafees bin Zafar — Digital Domain
Henrik Falt — Digital Domain

Figure 1 Figure 2 Figure 3 Figure 4

Introduction


Mandoline: Robust Cut-Cell Generation for Arbitrary Triangle Meshes

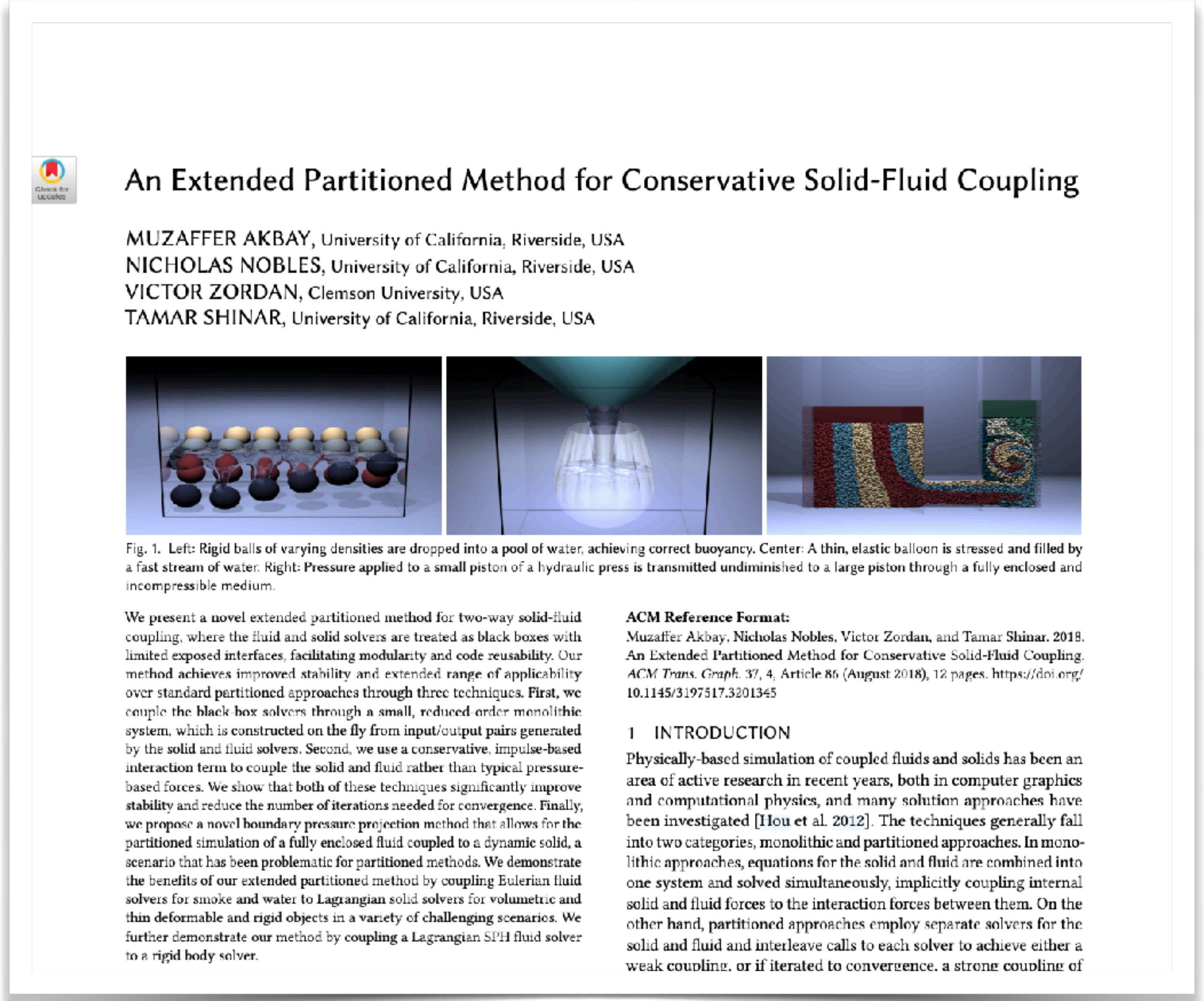
MICHAEL TAO, University of Toronto, Canada
CHRISTOPHER BATTY, University of Waterloo, Canada
EUGENE FIUME, Simon Fraser University, Canada and University of Toronto, Canada
DAVID I.W. LEVIN, University of Toronto, Canada


Adaptive Distributed Simulation of Fluids and Rigid Bodies

Haoyang Shi, University of Utah, USA, Roblox, USA, haoyang.shi@utah.edu
Victor Zordan, Roblox, USA, vbzordan@roblox.com
Yin Yang, University of Utah, USA, yin.yang@utah.edu
Sheldon Andrews, École de technologie supérieure, Canada, Roblox, USA, sheldon.andrews@etsmtl.ca

Eulerian and Hybrid Methods - Eulerian Coupling

- Rasterize [Takahashi et al. 2002]
  [Foster and Metaxas 1996]
- Immersed boundary method
  [Li et al. 2020] [Carlson et al. 2004] [Peskin 2002]
- Cut-Cell [Shi et al. 2024] [Tao et al. 2019]
  [Roble et al. 2005]
- Rigid-fluid coupling
  - Partitioned [Akbay et al. 2018]
  - Monolithic [Takahashi and Batty 2022]
    [Takahashi and Batty 2020]



An Extended Partitioned Method for Conservative Solid-Fluid Coupling

MUZAFFER AKBAY, University of California, Riverside, USA
NICHOLAS NOBLES, University of California, Riverside, USA
VICTOR ZORDAN, Clemson University, USA
TAMAR SHINAR, University of California, Riverside, USA

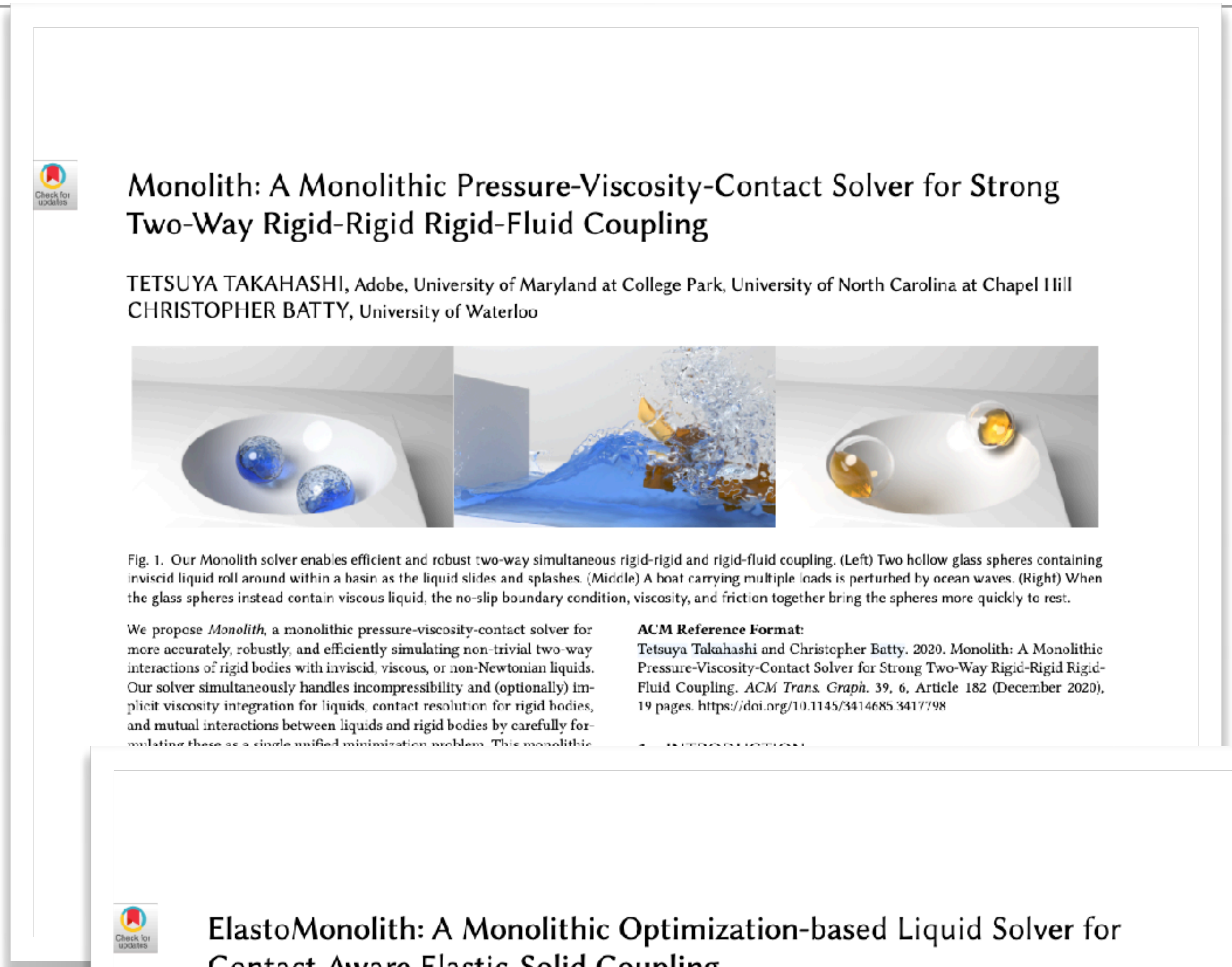
Fig. 1. Left: Rigid balls of varying densities are dropped into a pool of water, achieving correct buoyancy. Center: A thin, elastic balloon is stressed and filled by a fast stream of water. Right: Pressure applied to a small piston of a hydraulic press is transmitted undiminished to a large piston through a fully enclosed and incompressible medium.

We present a novel extended partitioned method for two-way solid-fluid coupling, where the fluid and solid solvers are treated as black boxes with limited exposed interfaces, facilitating modularity and code reusability. Our method achieves improved stability and extended range of applicability over standard partitioned approaches through three techniques. First, we couple the black-box solvers through a small, reduced-order monolithic system, which is constructed on the fly from input/output pairs generated by the solid and fluid solvers. Second, we use a conservative, impulse-based interaction term to couple the solid and fluid rather than typical pressure-based forces. We show that both of these techniques significantly improve stability and reduce the number of iterations needed for convergence. Finally, we propose a novel boundary pressure projection method that allows for the partitioned simulation of a fully enclosed fluid coupled to a dynamic solid, a scenario that has been problematic for partitioned methods. We demonstrate the benefits of our extended partitioned method by coupling Eulerian fluid solvers for smoke and water to Lagrangian solid solvers for volumetric and thin deformable and rigid objects in a variety of challenging scenarios. We further demonstrate our method by coupling a Lagrangian SPH fluid solver to a rigid body solver.

ACM Reference Format:
Muzaffer Akbay, Nicholas Nobles, Victor Zordan, and Tamar Shinar. 2018. An Extended Partitioned Method for Conservative Solid-Fluid Coupling. *ACM Trans. Graph.* 37, 4, Article 86 (August 2018), 12 pages. https://doi.org/10.1145/3197517.3201345

1 INTRODUCTION

Physically-based simulation of coupled fluids and solids has been an area of active research in recent years, both in computer graphics and computational physics, and many solution approaches have been investigated [Hou et al. 2012]. The techniques generally fall into two categories, monolithic and partitioned approaches. In monolithic approaches, equations for the solid and fluid are combined into one system and solved simultaneously, implicitly coupling internal solid and fluid forces to the interaction forces between them. On the other hand, partitioned approaches employ separate solvers for the solid and fluid and interleave calls to each solver to achieve either a weak coupling, or if iterated to convergence, a strong coupling of



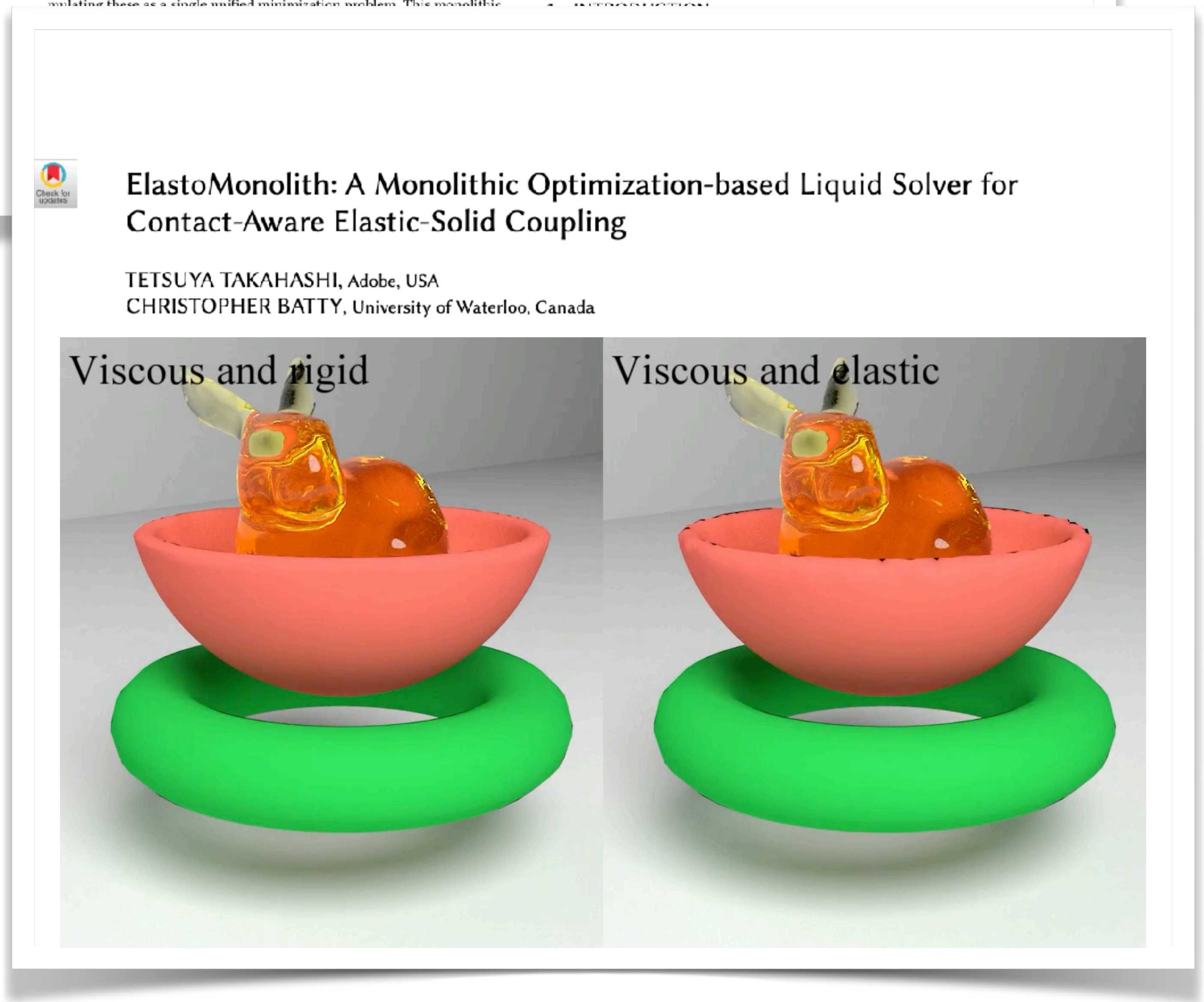
Monolith: A Monolithic Pressure-Viscosity-Contact Solver for Strong Two-Way Rigid-Rigid Rigid-Fluid Coupling

TETSUYA TAKAHASHI, Adobe, University of Maryland at College Park, University of North Carolina at Chapel Hill
CHRISTOPHER BATTY, University of Waterloo

Fig. 1. Our Monolith solver enables efficient and robust two-way simultaneous rigid-rigid and rigid-fluid coupling. (Left) Two hollow glass spheres containing inviscid liquid roll around within a basin as the liquid slides and splashes. (Middle) A boat carrying multiple loads is perturbed by ocean waves. (Right) When the glass spheres instead contain viscous liquid, the no-slip boundary condition, viscosity, and friction together bring the spheres more quickly to rest.

We propose *Monolith*, a monolithic pressure-viscosity-contact solver for more accurately, robustly, and efficiently simulating non-trivial two-way interactions of rigid bodies with inviscid, viscous, or non-Newtonian liquids. Our solver simultaneously handles incompressibility and (optionally) implicit viscosity integration for liquids, contact resolution for rigid bodies, and mutual interactions between liquids and rigid bodies by carefully for-

ACM Reference Format:
Tetsuya Takahashi and Christopher Batty. 2020. Monolith: A Monolithic Pressure-Viscosity-Contact Solver for Strong Two-Way Rigid-Rigid Rigid-Fluid Coupling. *ACM Trans. Graph.* 39, 6, Article 182 (December 2020), 19 pages. https://doi.org/10.1145/3414685.3417798

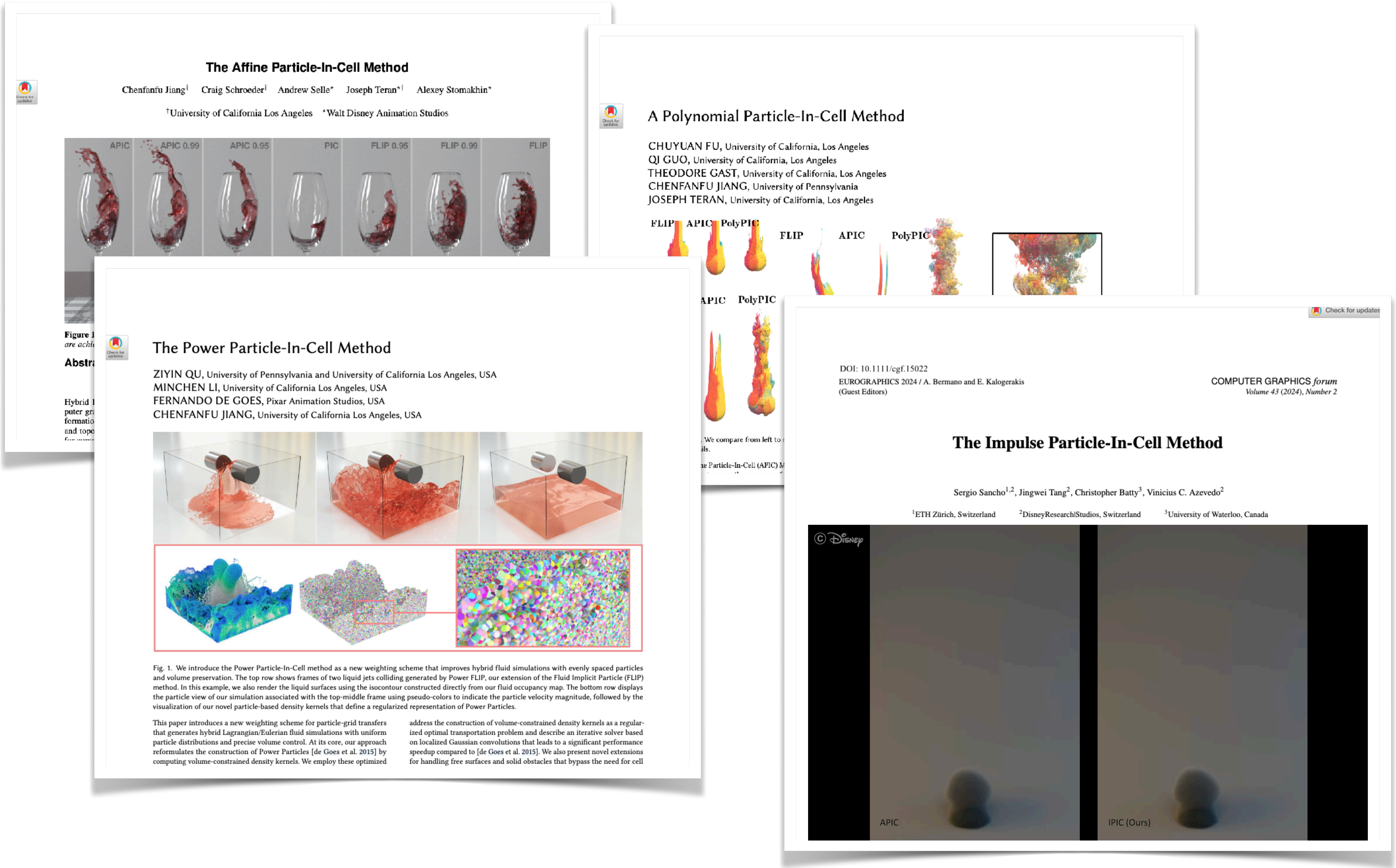


ElastoMonolith: A Monolithic Optimization-based Liquid Solver for Contact-Aware Elastic-Solid Coupling

TETSUYA TAKAHASHI, Adobe, USA
CHRISTOPHER BATTY, University of Waterloo, Canada

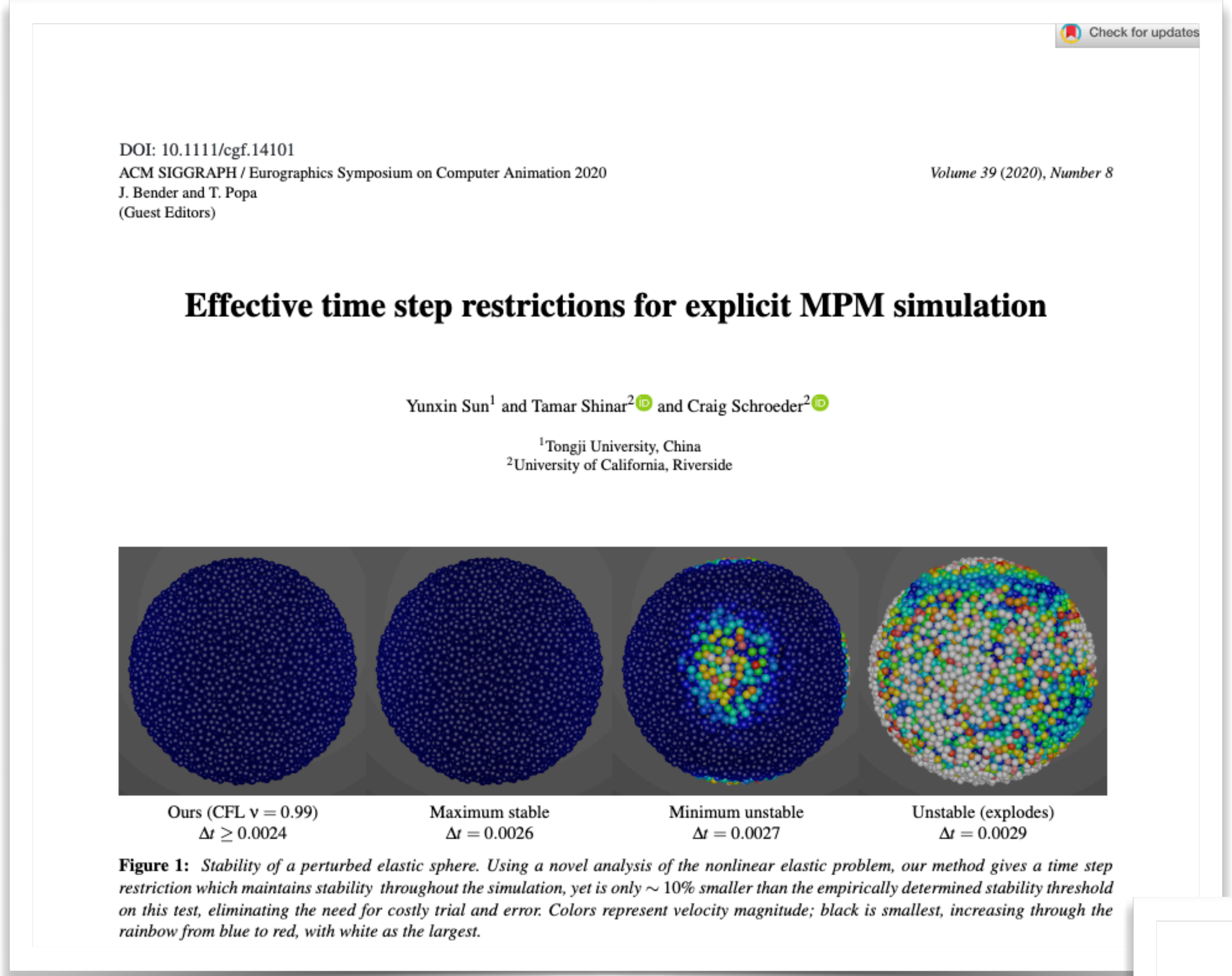
Eulerian and Hybrid Methods - Hybrid Stability and Multiphysics

- Point-to-grid transfers
  - Affine PIC [Jiang et al. 2015]
  - Poly PIC [Fu et al. 2017]
  - Power PIC [Qu et al. 2022]
  - Impulse PIC [Sancho et al. 2024]



Eulerian and Hybrid Methods - Hybrid Stability and Multiphysics

- Point-to-grid transfers
  - Affine PIC [Jiang et al. 2015]
  - Poly PIC [Fu et al. 2017]
  - Power PIC [Qu et al. 2022]
  - Impulse PIC [Sancho et al. 2024]
- Time-stepping
  - Explicit [Bai and Schroeder 2022] [Wang et al. 2020] [Sun et al. 2020]


Check for updates

DOI: 10.1111/cgf.14101
ACM SIGGRAPH / Eurographics Symposium on Computer Animation 2020 — *Volume 39 (2020), Number 8*
J. Bender and T. Popa
(Guest Editors)

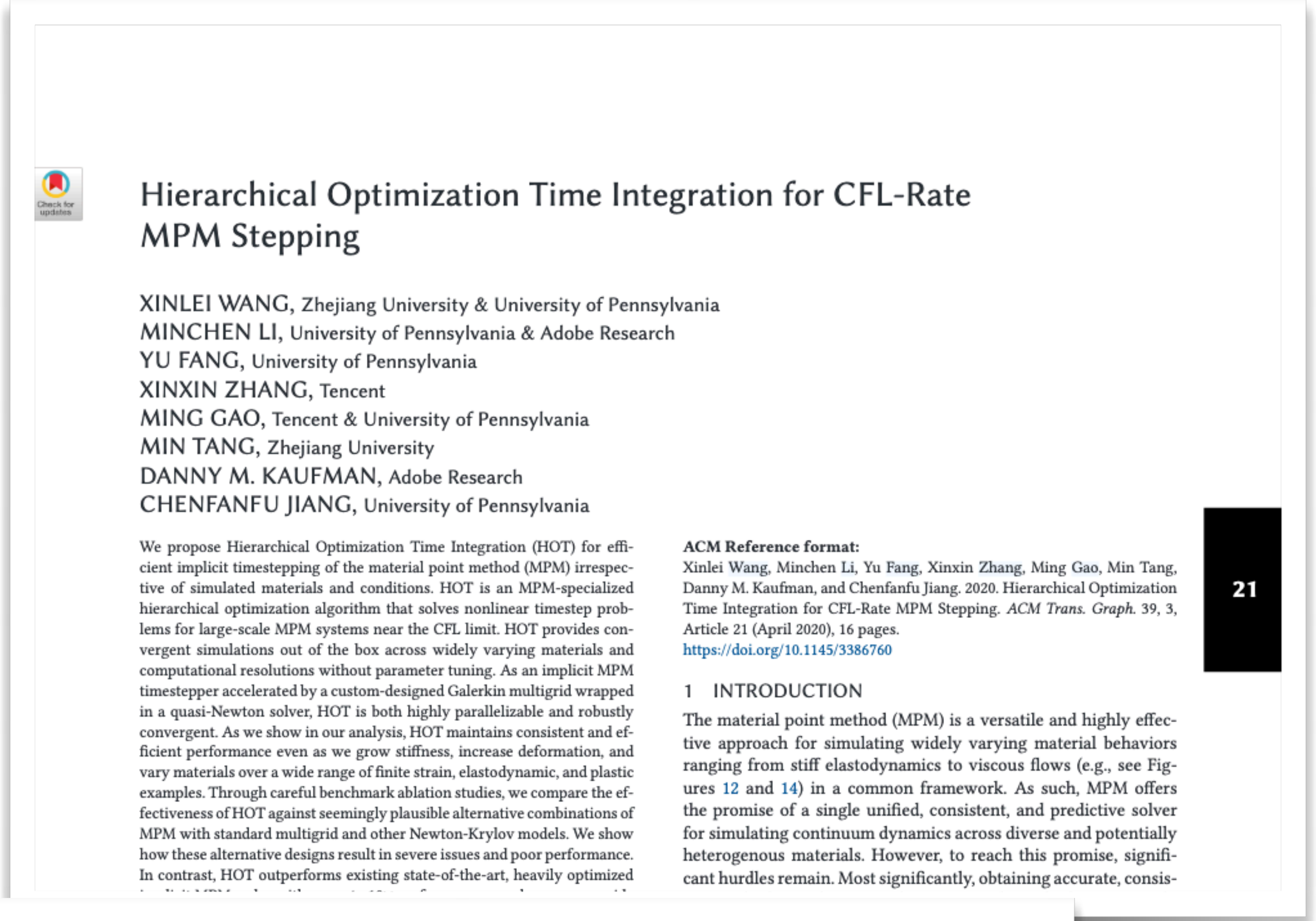
Effective time step restrictions for explicit MPM simulation

Yunxin Sun[1] and Tamar Shinar[2] and Craig Schroeder[2]

[1]Tongji University, China
[2]University of California, Riverside

Ours (CFL $\nu = 0.99$) $\Delta t \geq 0.0024$ | Maximum stable $\Delta t = 0.0026$ | Minimum unstable $\Delta t = 0.0027$ | Unstable (explodes) $\Delta t = 0.0029$

Figure 1: *Stability of a perturbed elastic sphere. Using a novel analysis of the nonlinear elastic problem, our method gives a time step restriction which maintains stability throughout the simulation, yet is only ~ 10% smaller than the empirically determined stability threshold on this test, eliminating the need for costly trial and error. Colors represent velocity magnitude; black is smallest, increasing through the rainbow from blue to red, with white as the largest.*


Check for updates

Hierarchical Optimization Time Integration for CFL-Rate MPM Stepping

XINLEI WANG, Zhejiang University & University of Pennsylvania
MINCHEN LI, University of Pennsylvania & Adobe Research
YU FANG, University of Pennsylvania
XINXIN ZHANG, Tencent
MING GAO, Tencent & University of Pennsylvania
MIN TANG, Zhejiang University
DANNY M. KAUFMAN, Adobe Research
CHENFANFU JIANG, University of Pennsylvania

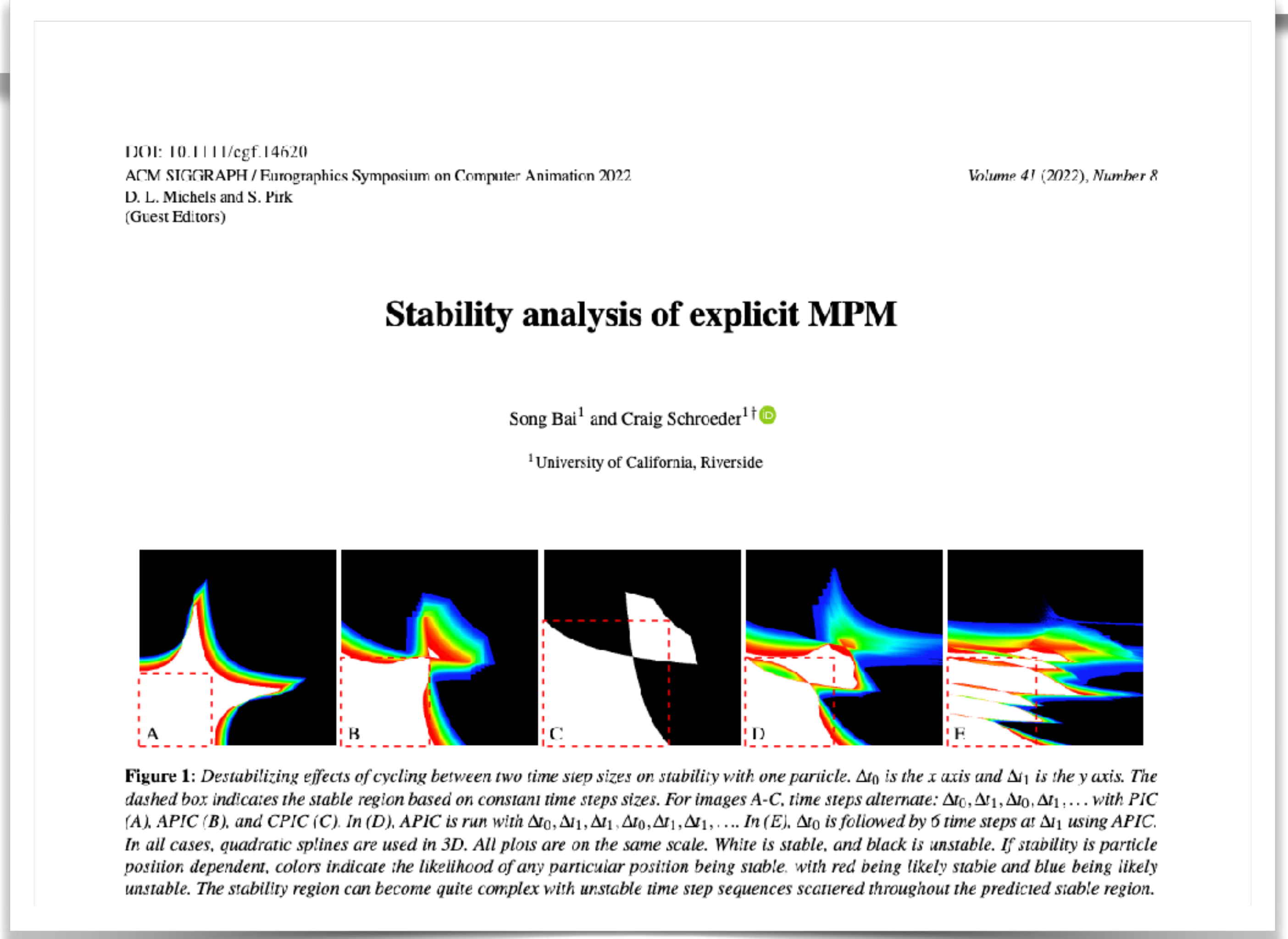
We propose Hierarchical Optimization Time Integration (HOT) for efficient implicit timestepping of the material point method (MPM) irrespective of simulated materials and conditions. HOT is an MPM-specialized hierarchical optimization algorithm that solves nonlinear timestep problems for large-scale MPM systems near the CFL limit. HOT provides convergent simulations out of the box across widely varying materials and computational resolutions without parameter tuning. As an implicit MPM timestepper accelerated by a custom-designed Galerkin multigrid wrapped in a quasi-Newton solver, HOT is both highly parallelizable and robustly convergent. As we show in our analysis, HOT maintains consistent and efficient performance even as we grow stiffness, increase deformation, and vary materials over a wide range of finite strain, elastodynamic, and plastic examples. Through careful benchmark ablation studies, we compare the effectiveness of HOT against seemingly plausible alternative combinations of MPM with standard multigrid and other Newton-Krylov models. We show how these alternative designs result in severe issues and poor performance. In contrast, HOT outperforms existing state-of-the-art, heavily optimized

ACM Reference format:
Xinlei Wang, Minchen Li, Yu Fang, Xinxin Zhang, Ming Gao, Min Tang, Danny M. Kaufman, and Chenfanfu Jiang. 2020. Hierarchical Optimization Time Integration for CFL-Rate MPM Stepping. *ACM Trans. Graph.* 39, 3, Article 21 (April 2020), 16 pages.
https://doi.org/10.1145/3386760

21

1 INTRODUCTION

The material point method (MPM) is a versatile and highly effective approach for simulating widely varying material behaviors ranging from stiff elastodynamics to viscous flows (e.g., see Figures 12 and 14) in a common framework. As such, MPM offers the promise of a single unified, consistent, and predictive solver for simulating continuum dynamics across diverse and potentially heterogenous materials. However, to reach this promise, significant hurdles remain. Most significantly, obtaining accurate, consis-


DOI: 10.1111/cgf.14620
ACM SIGGRAPH / Eurographics Symposium on Computer Animation 2022 — *Volume 41 (2022), Number 8*
D. L. Michels and S. Pirk
(Guest Editors)

Stability analysis of explicit MPM

Song Bai[1] and Craig Schroeder[1†]

[1]University of California, Riverside

A B C D E

Figure 1: *Destabilizing effects of cycling between two time step sizes on stability with one particle. Δt_0 is the x axis and Δt_1 is the y axis. The dashed box indicates the stable region based on constant time steps sizes. For images A-C, time steps alternate: $\Delta t_0, \Delta t_1, \Delta t_0, \Delta t_1, \ldots$ with PIC (A), APIC (B), and CPIC (C). In (D), APIC is run with $\Delta t_0, \Delta t_1, \Delta t_1, \Delta t_0, \Delta t_1, \Delta t_1, \ldots$. In (E), Δt_0 is followed by 6 time steps at Δt_1 using APIC. In all cases, quadratic splines are used in 3D. All plots are on the same scale. White is stable, and black is unstable. If stability is particle position dependent, colors indicate the likelihood of any particular position being stable, with red being likely stable and blue being likely unstable. The stability region can become quite complex with unstable time step sequences scattered throughout the predicted stable region.*

Eulerian and Hybrid Methods - Hybrid Stability and Multiphysics

- Point-to-grid transfers
  - Affine PIC [Jiang et al. 2015]
  - Poly PIC [Fu et al. 2017]
  - Power PIC [Qu et al. 2022]
  - Impulse PIC [Sancho et al. 2024]
- Time-stepping
  - Explicit [Bai and Schroeder 2022] [Wang et al. 2020] [Sun et al. 2020]
  - Implicit [Hyde et al. 2020] [Fang et al. 2019] [Hu et al. 2018] [Gast et al. 2015]

Eulerian and Hybrid Methods - Hybrid Materials

- Snow [Stomakhin et al. 2013]

A material point method for snow simulation

Alexey Stomakhin*† Craig Schroeder† Lawrence Chai* Joseph Teran*† Andrew Selle*

†University of California Los Angeles *Walt Disney Animation Studios

Abstract

Snow is a challenging natural phenomenon to visually simulate. While the graphics community has previously considered accumulation and rendering of snow, animation of snow dynamics has not been fully addressed. Additionally, existing techniques for solids and fluids have difficulty producing convincing snow results. Specifically, *wet* or *dense* snow that has both solid- and fluid-like properties is difficult to handle. Consequently, this paper presents a novel snow simulation method utilizing a user-controllable elasto-plastic constitutive model integrated with a hybrid Eulerian/Lagrangian Material Point Method. The method is continuum based and its hybrid nature allows us to use a regular Cartesian grid to automate treatment of self-collision and fracture. It also naturally allows us to derive a grid-based semi-implicit integration scheme that has conditioning independent of the number of Lagrangian particles. We demonstrate the power of our method with a variety of snow phenomena including complex character interactions.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation I.6.8 [Simulation and Modeling]: Types of Simulation—Animation

Keywords: material point, snow simulation, physically-based modeling

Links: DL PDF WEB

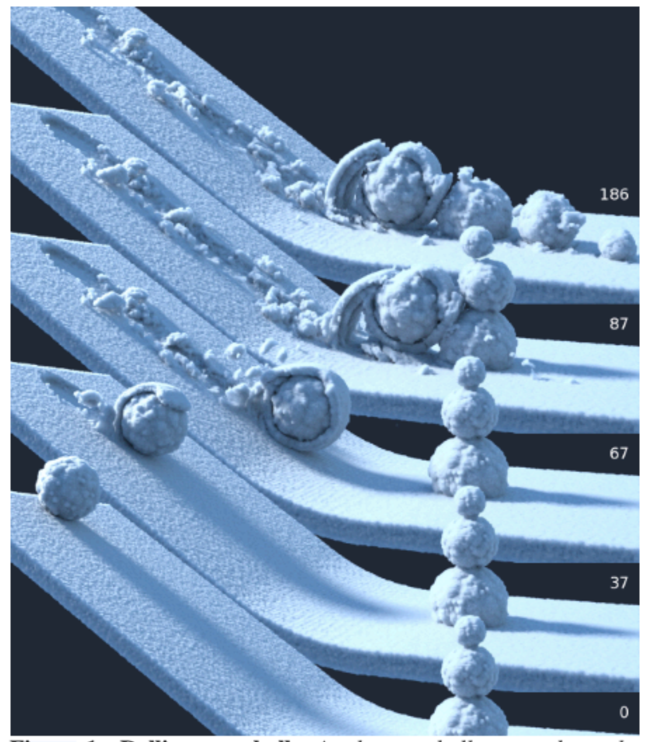


Figure 1: Rolling snowball. *As the snowball moves down the hill, compressed snow sticks, demonstrating that we can handle so-called packing snow effect.* ©*Disney.*

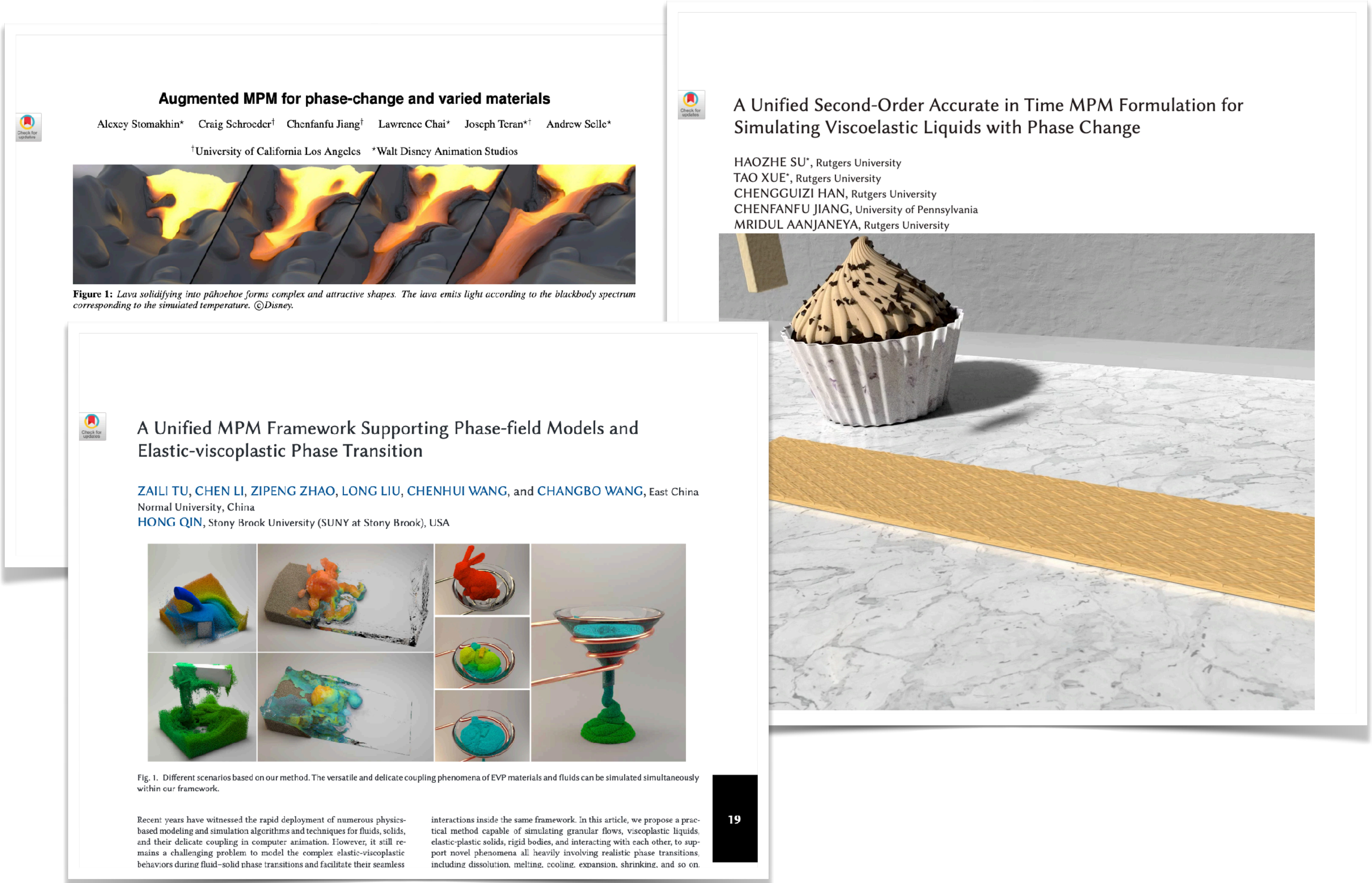
1 Introduction

Snow dynamics are amazingly beautiful yet varied. Whether it is *powder snow* fluttering in a skier's wake, foot steps shattering an *icy snow* crust or even *packing snow* rolled into balls to make a snow-

versa), it is not the most optimal strategy. When solids and fluids are needed simultaneously, researchers have developed two-way coupled systems to get good accuracy and performance for both phenomena. Unfortunately, snow has continuously varying phase

Eulerian and Hybrid Methods - Hybrid Materials

- Snow [Stomakhin et al. 2013]
- Phase-change [Tu et al. 2024] [Su et al. 2021] [Stomakhin et al. 2014]



Augmented MPM for phase-change and varied materials

Alexey Stomakhin* Craig Schroeder† Chenfanfu Jiang† Lawrence Chai* Joseph Teran*† Andrew Selle*

†University of California Los Angeles *Walt Disney Animation Studios

Figure 1: Lava solidifying into pāhoehoe forms complex and attractive shapes. The lava emits light according to the blackbody spectrum corresponding to the simulated temperature. ©Disney.

A Unified Second-Order Accurate in Time MPM Formulation for Simulating Viscoelastic Liquids with Phase Change

HAOZHE SU*, Rutgers University
TAO XUE*, Rutgers University
CHENGGUIZI HAN, Rutgers University
CHENFANFU JIANG, University of Pennsylvania
MRIDUL AANJANEYA, Rutgers University

A Unified MPM Framework Supporting Phase-field Models and Elastic-viscoplastic Phase Transition

ZAILI TU, CHEN LI, ZIPENG ZHAO, LONG LIU, CHENHUI WANG, and CHANGBO WANG, East China Normal University, China
HONG QIN, Stony Brook University (SUNY at Stony Brook), USA

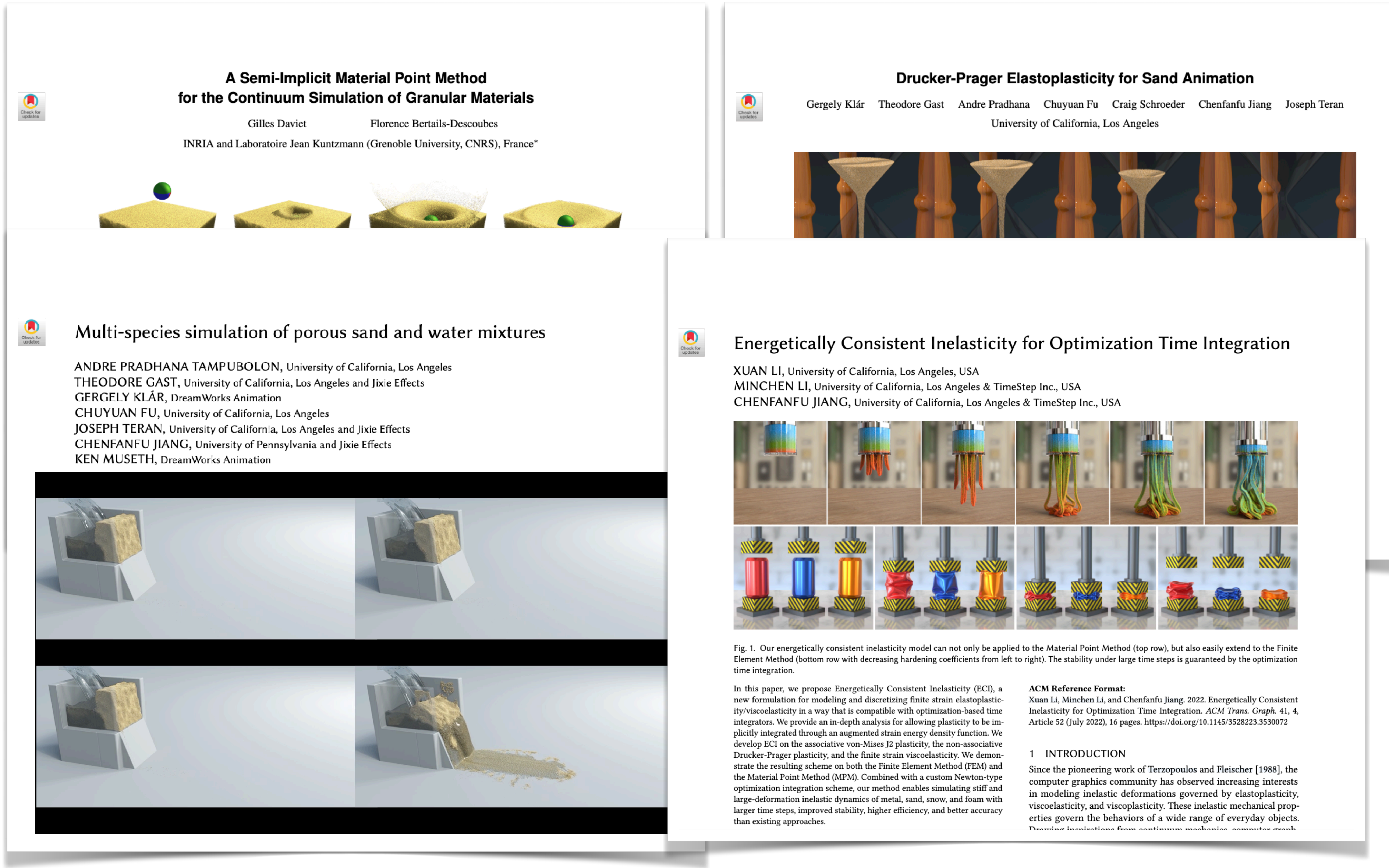
Fig. 1. Different scenarios based on our method. The versatile and delicate coupling phenomena of EVP materials and fluids can be simulated simultaneously within our framework.

Recent years have witnessed the rapid deployment of numerous physics-based modeling and simulation algorithms and techniques for fluids, solids, and their delicate coupling in computer animation. However, it still remains a challenging problem to model the complex elastic-viscoplastic behaviors during fluid–solid phase transitions and facilitate their seamless interactions inside the same framework. In this article, we propose a practical method capable of simulating granular flows, viscoplastic liquids, elastic-plastic solids, rigid bodies, and interacting with each other, to support novel phenomena all heavily involving realistic phase transitions, including dissolution, melting, cooling, expansion, shrinking, and so on.

19

Eulerian and Hybrid Methods - Hybrid Materials

- Snow [Stomakhin et al. 2013]
- Phase-change [Tu et al. 2024] [Su et al. 2021] [Stomakhin et al. 2014]
- Granular [Li et al. 2022] [Tampubolon et al. 2017] [Klár et al. 2016] [Daviet and Bertails-Descoubes 2016]



A Semi-Implicit Material Point Method for the Continuum Simulation of Granular Materials

Gilles Daviet   Florence Bertails-Descoubes

INRIA and Laboratoire Jean Kuntzmann (Grenoble University, CNRS), France*

Drucker-Prager Elastoplasticity for Sand Animation

Gergely Klár   Theodore Gast   Andre Pradhana   Chuyuan Fu   Craig Schroeder   Chenfanfu Jiang   Joseph Teran

University of California, Los Angeles

Multi-species simulation of porous sand and water mixtures

ANDRE PRADHANA TAMPUBOLON, University of California, Los Angeles
THEODORE GAST, University of California, Los Angeles and Jixie Effects
GERGELY KLÁR, DreamWorks Animation
CHUYUAN FU, University of California, Los Angeles
JOSEPH TERAN, University of California, Los Angeles and Jixie Effects
CHENFANFU JIANG, University of Pennsylvania and Jixie Effects
KEN MUSETH, DreamWorks Animation

Energetically Consistent Inelasticity for Optimization Time Integration

XUAN LI, University of California, Los Angeles, USA
MINCHEN LI, University of California, Los Angeles & TimeStep Inc., USA
CHENFANFU JIANG, University of California, Los Angeles & TimeStep Inc., USA

Fig. 1. Our energetically consistent inelasticity model can not only be applied to the Material Point Method (top row), but also easily extend to the Finite Element Method (bottom row with decreasing hardening coefficients from left to right). The stability under large time steps is guaranteed by the optimization time integration.

In this paper, we propose Energetically Consistent Inelasticity (ECI), a new formulation for modeling and discretizing finite strain elastoplasticity/viscoelasticity in a way that is compatible with optimization-based time integrators. We provide an in-depth analysis for allowing plasticity to be implicitly integrated through an augmented strain energy density function. We develop ECI on the associative von-Mises J2 plasticity, the non-associative Drucker-Prager plasticity, and the finite strain viscoelasticity. We demonstrate the resulting scheme on both the Finite Element Method (FEM) and the Material Point Method (MPM). Combined with a custom Newton-type optimization integration scheme, our method enables simulating stiff and large-deformation inelastic dynamics of metal, sand, snow, and foam with larger time steps, improved stability, higher efficiency, and better accuracy than existing approaches.

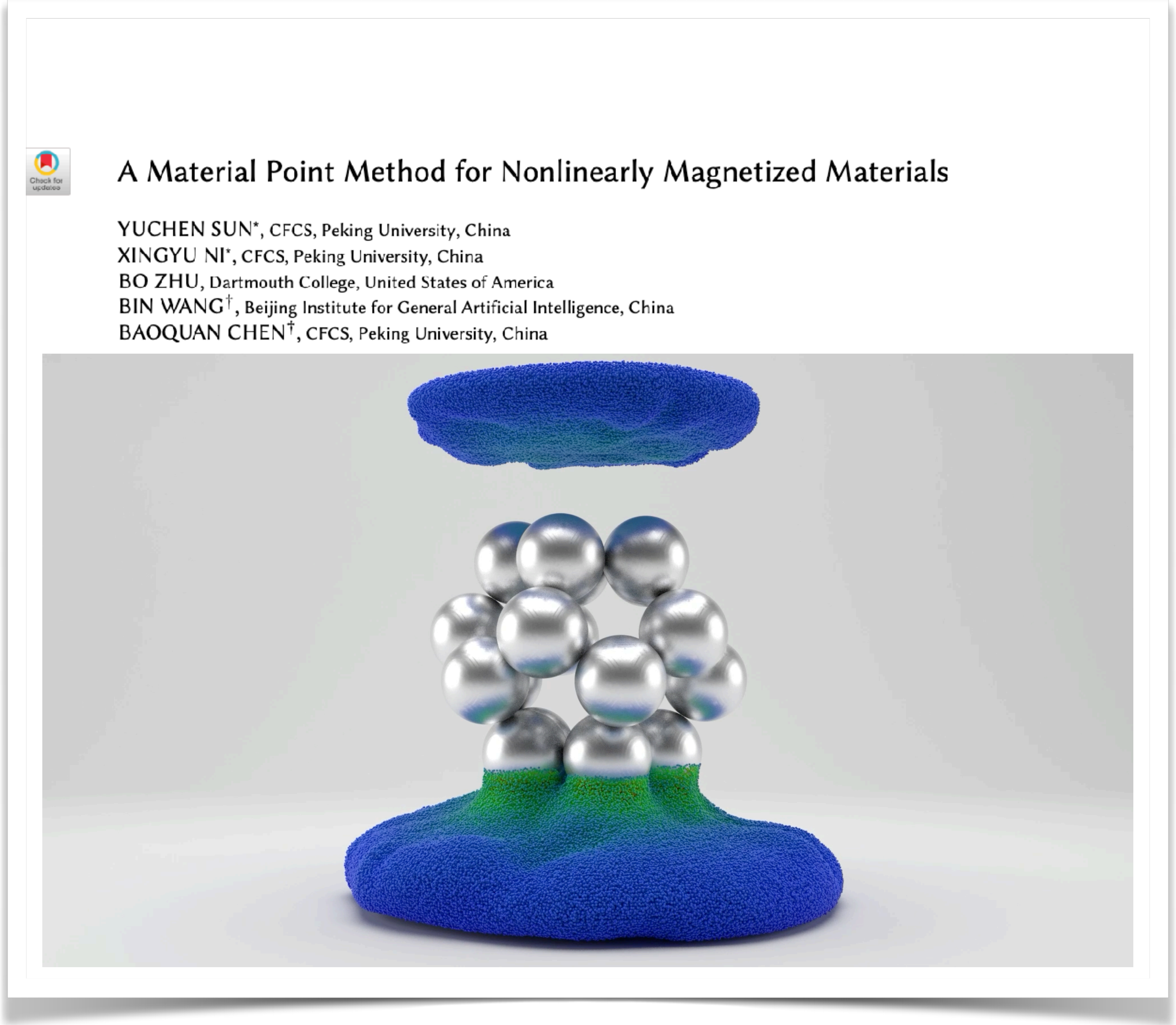
ACM Reference Format:
Xuan Li, Minchen Li, and Chenfanfu Jiang. 2022. Energetically Consistent Inelasticity for Optimization Time Integration. *ACM Trans. Graph.* 41, 4, Article 52 (July 2022), 16 pages. https://doi.org/10.1145/3528223.3530072

1 INTRODUCTION

Since the pioneering work of Terzopoulos and Fleischer [1988], the computer graphics community has observed increasing interests in modeling inelastic deformations governed by elastoplasticity, viscoelasticity, and viscoplasticity. These inelastic mechanical properties govern the behaviors of a wide range of everyday objects.

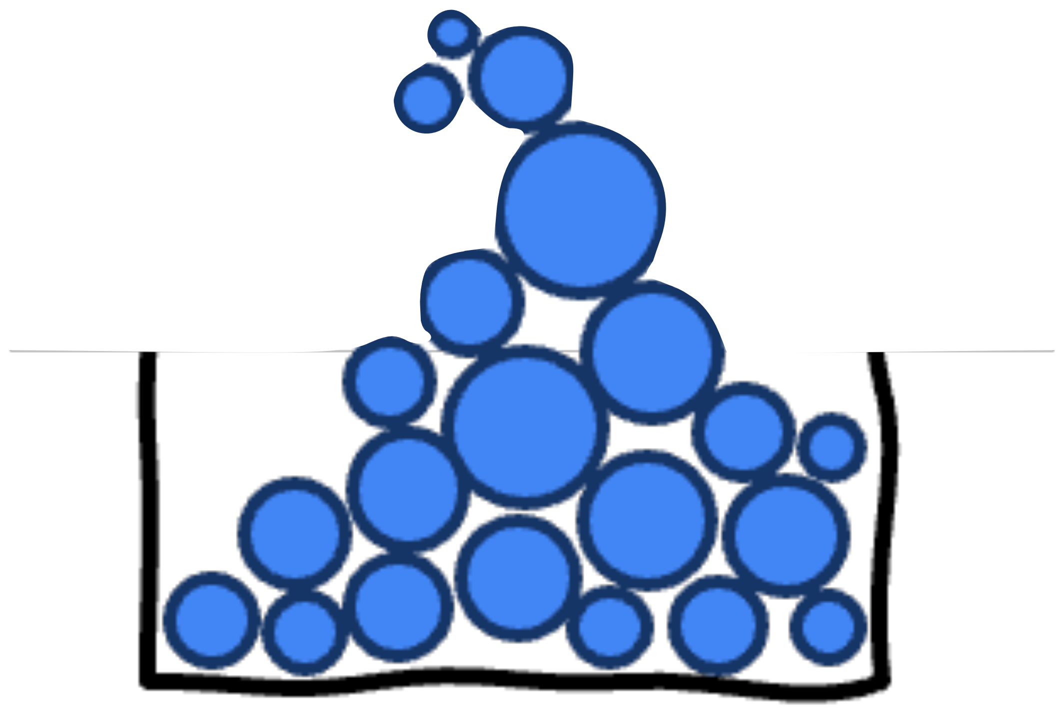
Eulerian and Hybrid Methods - Hybrid Multiphysics Materials

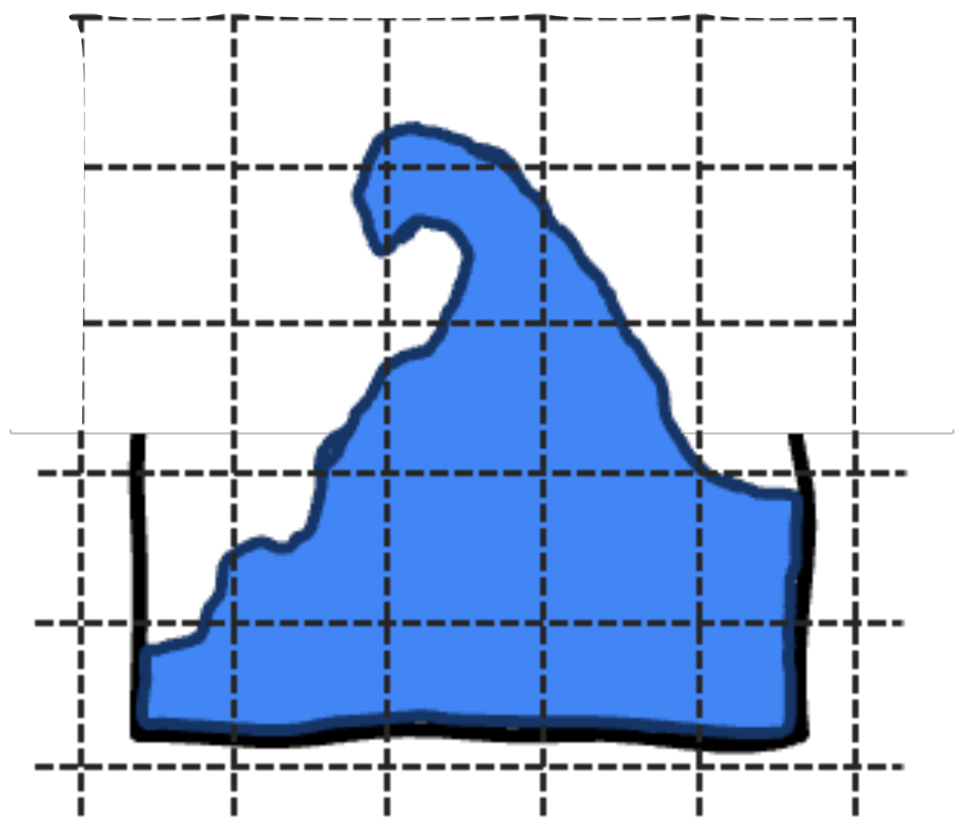
- Snow [Stomakhin et al. 2013]
- Phase-change [Tu et al. 2024]
  [Su et al. 2021] [Stomakhin et al. 2014]
- Granular
  [Li et al. 2022] [Tampubolon et al. 2017]
  [Klár et al. 2016]
  [Daviet and Bertails-Descoubes 2016]
- Magnetic [Sun et al. 2021]

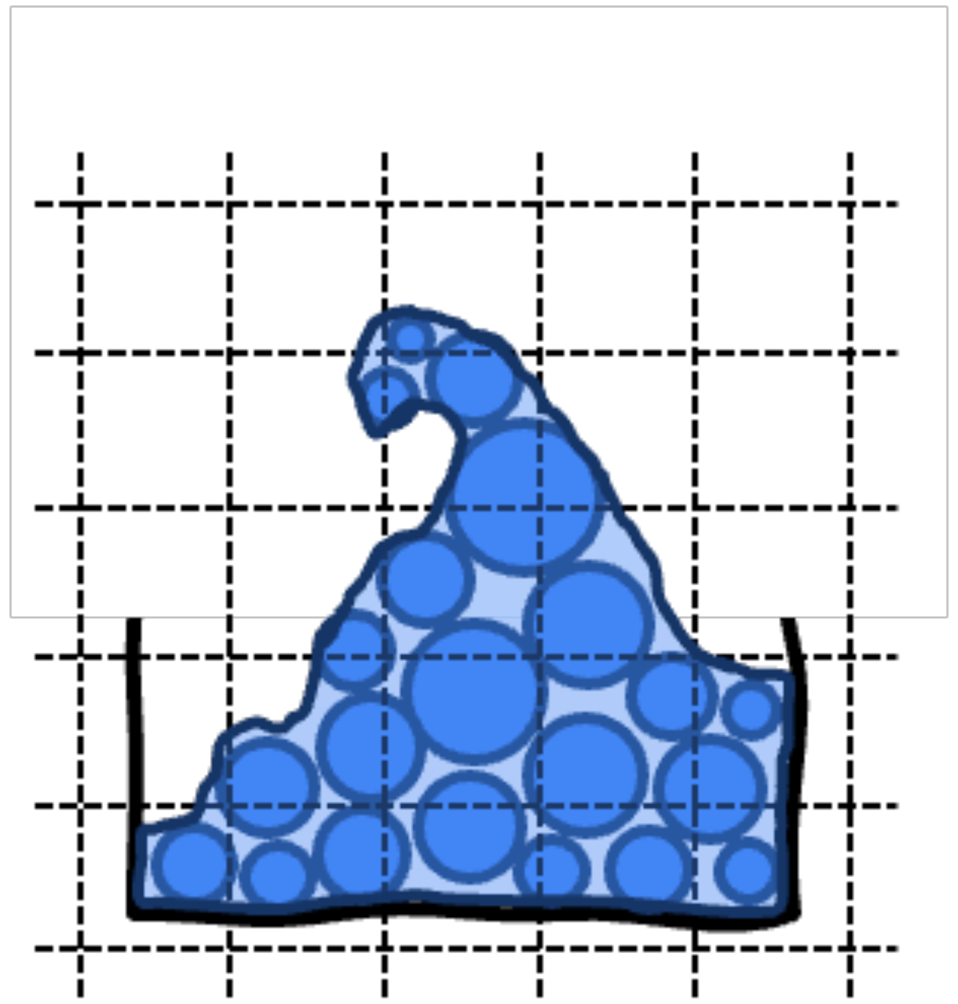

A Material Point Method for Nonlinearly Magnetized Materials

YUCHEN SUN*, CFCS, Peking University, China
XINGYU NI*, CFCS, Peking University, China
BO ZHU, Dartmouth College, United States of America
BIN WANG†, Beijing Institute for General Artificial Intelligence, China
BAOQUAN CHEN†, CFCS, Peking University, China

Part III - Summary







Lagrangian	Eulerian	Hybrid
🏆 Topology changes	🏆 Air and smoke	🏆 Topology changes
🏆 Contact	🏆 Turbulence	🏆 Complex material models
🏆 Fluid-rigid coupling	🏆 Bounded domains	🏆 Cutting and fracture