

A Survey on SPH Methods in Computer Graphics

MATERIALS

Dan Koschier Jan Bender Barbara Solenthaler Matthias Teschner



RWTHAACHEN
UNIVERSITY

ETH zürich

UNI
FREIBURG

Outline

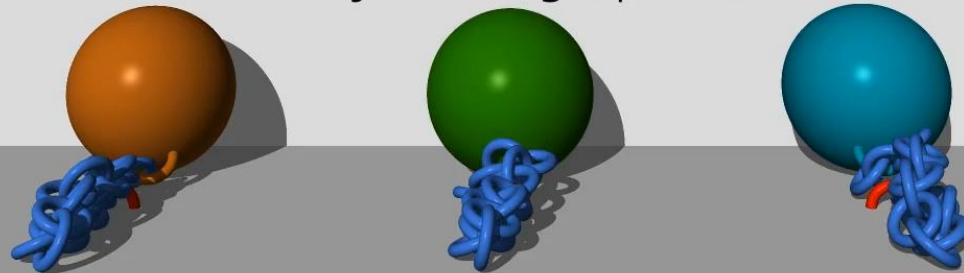
- Motivation
- Viscous Force
- Explicit Viscosity
- Implicit Viscosity
- Results

Motivation

Moored buoys

chain contacts are fully simulated
with our proposed rigid body solver
using only rigid-rigid contacts

up to 5M fluid particles
2M static rigid particles
149k dynamic rigid particles



Viscous Fluids



Outline

- Motivation
- Viscous Force
- Explicit Viscosity
- Implicit Viscosity
- Results

Classical Newtonian Fluid Model

Linear momentum equation:

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

- Density ρ
- Velocity \mathbf{v}
- Stress tensor \mathbf{T}
- Force density \mathbf{f}

Newtonian constitutive model:

$$\mathbf{T} = -p\mathbb{1} + 2\mu\mathbf{E}$$

$$\mathbf{E} = \frac{1}{2} (\nabla\mathbf{v} + (\nabla\mathbf{v})^T)$$

- Strain rate tensor \mathbf{E}
- Pressure p
- Identity $\mathbb{1}$
- Dynamic viscosity μ

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

Viscous Force

- Viscous force for incompressible fluids:

$$\mathbf{f}_{\text{visco}} = \mu \nabla^2 \mathbf{v}$$

- Recent SPH solvers either compute the divergence of the strain rate \mathbf{E} or directly determine the Laplacian of \mathbf{v} .
- Note that strain rate based approaches must enforce a divergence-free velocity field to avoid undesired bulk viscosity.

Outline

- Motivation
- Viscous Force
- Explicit Viscosity
- Implicit Viscosity
- Results

Explicit Viscosity

- Standard SPH discretization of this Laplacian:

$$\nabla^2 \mathbf{v}_i = \sum_j \frac{m_j}{\rho_j} \mathbf{v}_j \nabla^2 W_{ij}$$

- Disadvantages:
 - The Laplacian of the kernel changes its sign inside the support radius
 - Sensitive to particle disorder [Mon05, Pri12]

Explicit Viscosity

- Alternative: determine one derivative using SPH and the second one using finite differences.

$$\nabla^2 v_i \approx 2(d + 2) \sum_j \frac{m_j}{\rho_j} \frac{\mathbf{v}_{ij} \cdot \mathbf{x}_{ij}}{\|\mathbf{x}_{ij}\|^2 + 0.01h^2} \nabla W_{ij}$$

$$\mathbf{x}_{ij} = \mathbf{x}_i - \mathbf{x}_j, \quad \mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j, \quad d = \text{dimension}$$

- Advantages:
 - Galilean invariant
 - vanishes for rigid body rotation
 - conserves linear and angular momentum

XSPH

- Core idea of XSPH: reduce the particle disorder by smoothing the velocity field:

$$\hat{\mathbf{v}}_i = \mathbf{v}_i + \alpha \sum_j \frac{m_j}{\rho_j} (\mathbf{v}_j - \mathbf{v}_i) W_{ij}$$

- XSPH can also be used as artificial viscosity model.
- Advantage: The second derivative is not needed.
- Disadvantage: α is not physically meaningful.

Outline

- Motivation
- Viscous Force
- Explicit Viscosity
- Implicit Viscosity
- Results

Takahashi et al. 2015

- Compute the viscous force as divergence of the strain rate \mathbf{E} :

$$\mathbf{f}_{\text{visco}} = \mu \nabla \cdot (\nabla \mathbf{v}_i + (\nabla \mathbf{v}_i)^T) = \mu \sum_j m_j \left(\frac{2\mathbf{E}_i}{\rho_i^2} + \frac{2\mathbf{E}_j}{\rho_j^2} \right) \nabla W_{ij}$$

- Implicit integration scheme

$$\mathbf{v}(t + \Delta t) = \mathbf{v}^* + \frac{\Delta t}{\rho} \mathbf{f}_{\text{visco}}(t + \Delta t)$$

- Solve linear system using the conjugate gradient method (CG).

Peer et al. 2015/2016

- Decompose velocity gradient: $\nabla \mathbf{v} = \mathbf{R} + \mathbf{V} + \mathbf{S}$
- Reduce shear rate by user-defined factor $0 \leq \xi \leq 1$:

$$\nabla \mathbf{v}^{\text{target}} = \mathbf{R} + \mathbf{V} + \xi \mathbf{S}$$

- Reconstruct velocity field by solving linear system with CG:

$$\mathbf{v}_i(t + \Delta t) = \frac{1}{\rho_i} \sum_j m_j \left(\mathbf{v}_j(t + \Delta t) + \frac{\nabla \mathbf{v}_i^{\text{target}} + \nabla \mathbf{v}_j^{\text{target}}}{2} \mathbf{x}_{ij} \right) W_{ij}$$

Bender & Koschier 2017

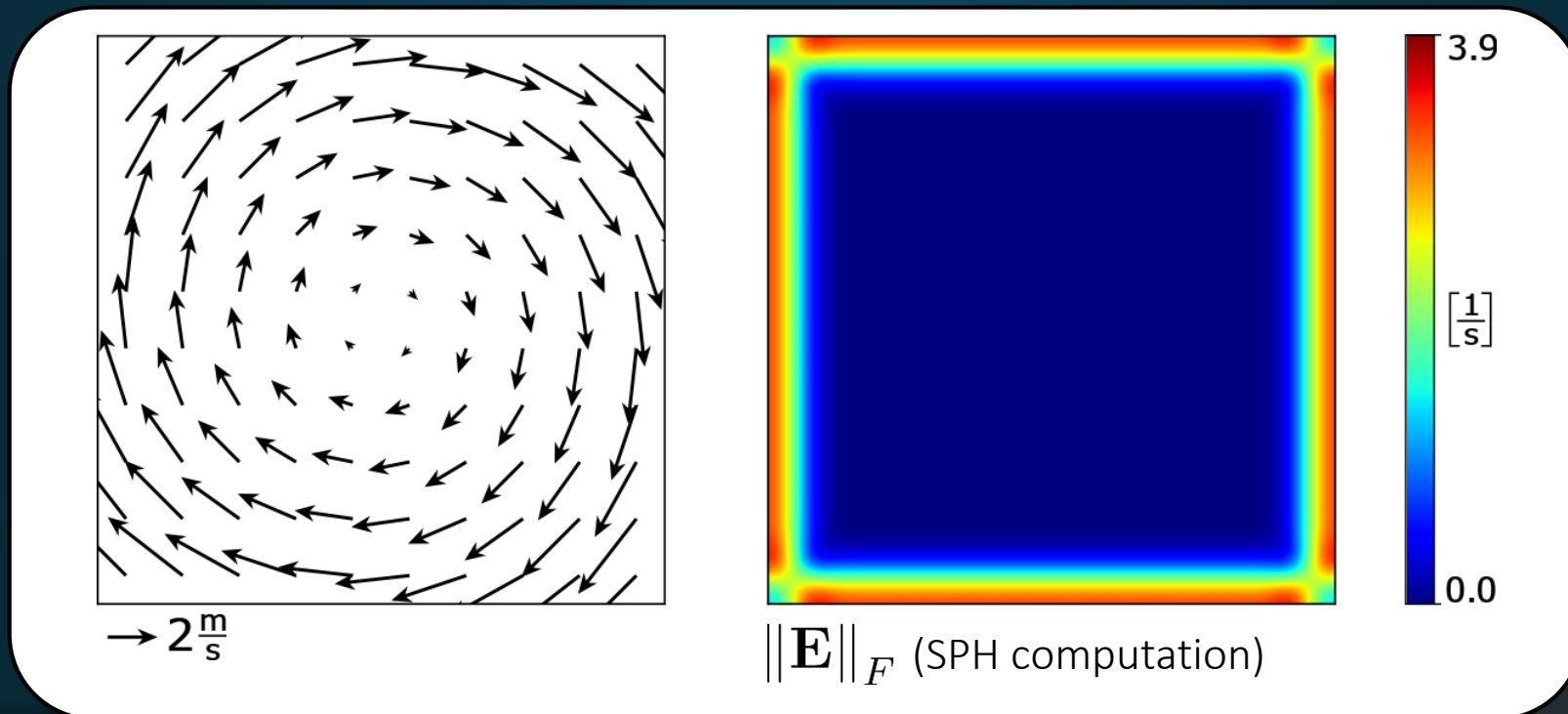
- Define velocity constraint for each particle with user-defined factor:

$$\mathbf{C}_i(\mathbf{v}) = \mathbf{E}_i - \gamma \mathbf{E}_i = \mathbf{0}, \quad 0 \leq \gamma \leq 1$$

- The constraint is a 6D function due to the symmetric strain tensor.
- Solve linear system for corresponding Lagrange multipliers.

Strain Rate Computation

- The introduced methods are based on the strain rate.
- However, computing the strain rate using SPH leads to errors at the free surface due to particle deficiency.



Weiler et al. 2018

- ◉ Implicit integration scheme

$$\mathbf{v}(t + \Delta t) = \mathbf{v}^* + \frac{\Delta t}{\rho} \mu \nabla^2 \mathbf{v}(t + \Delta t)$$

- ◉ Compute Laplacian as

$$\nabla^2 \mathbf{v}_i = 2(d + 2) \sum_j \frac{m_j}{\rho_j} \frac{\mathbf{v}_{ij} \cdot \mathbf{x}_{ij}}{\|\mathbf{x}_{ij}\|^2 + 0.01h^2} \nabla W_{ij}$$

- ◉ Solve linear system using a meshless conjugate gradient method.
- ◉ Laplacian approximation avoids problems at the free surface.

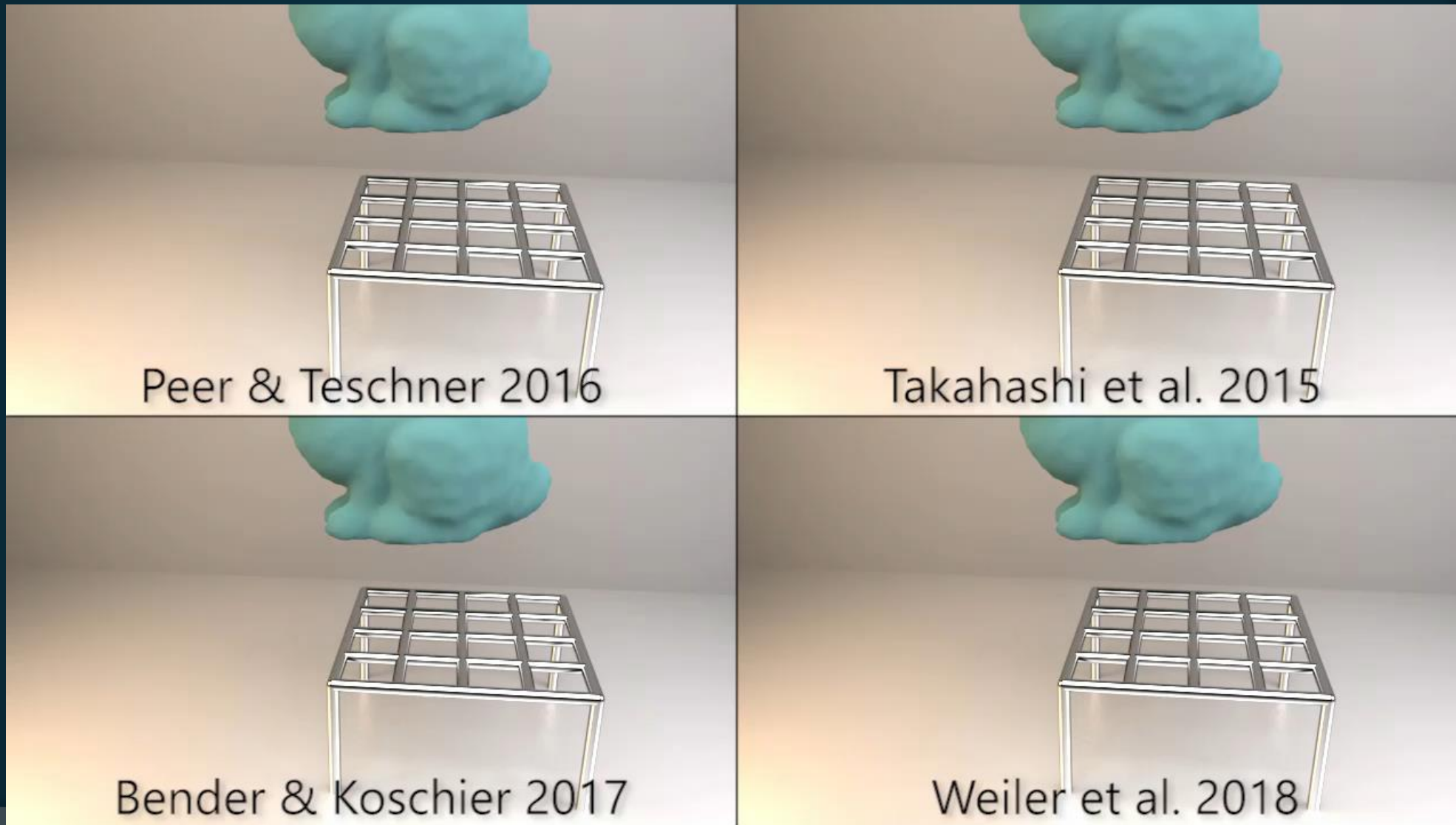
Discussion

- The strain rate based formulations lead to errors and artifacts at the free surface, which is avoided by directly computing the Laplacian of the velocity field.
- The viscosity parameters of Bender and Peer depend on the temporal and spatial resolution.
- Peer's reconstruction of the velocity field is fast but introduces a significant damping => not suitable for low viscous flow
- Takahashi et al. require the second-ring neighbors => low performance

Outline

- Motivation
- Viscous Force
- Explicit Viscosity
- Implicit Viscosity
- Results

Quality Comparison



Coiling

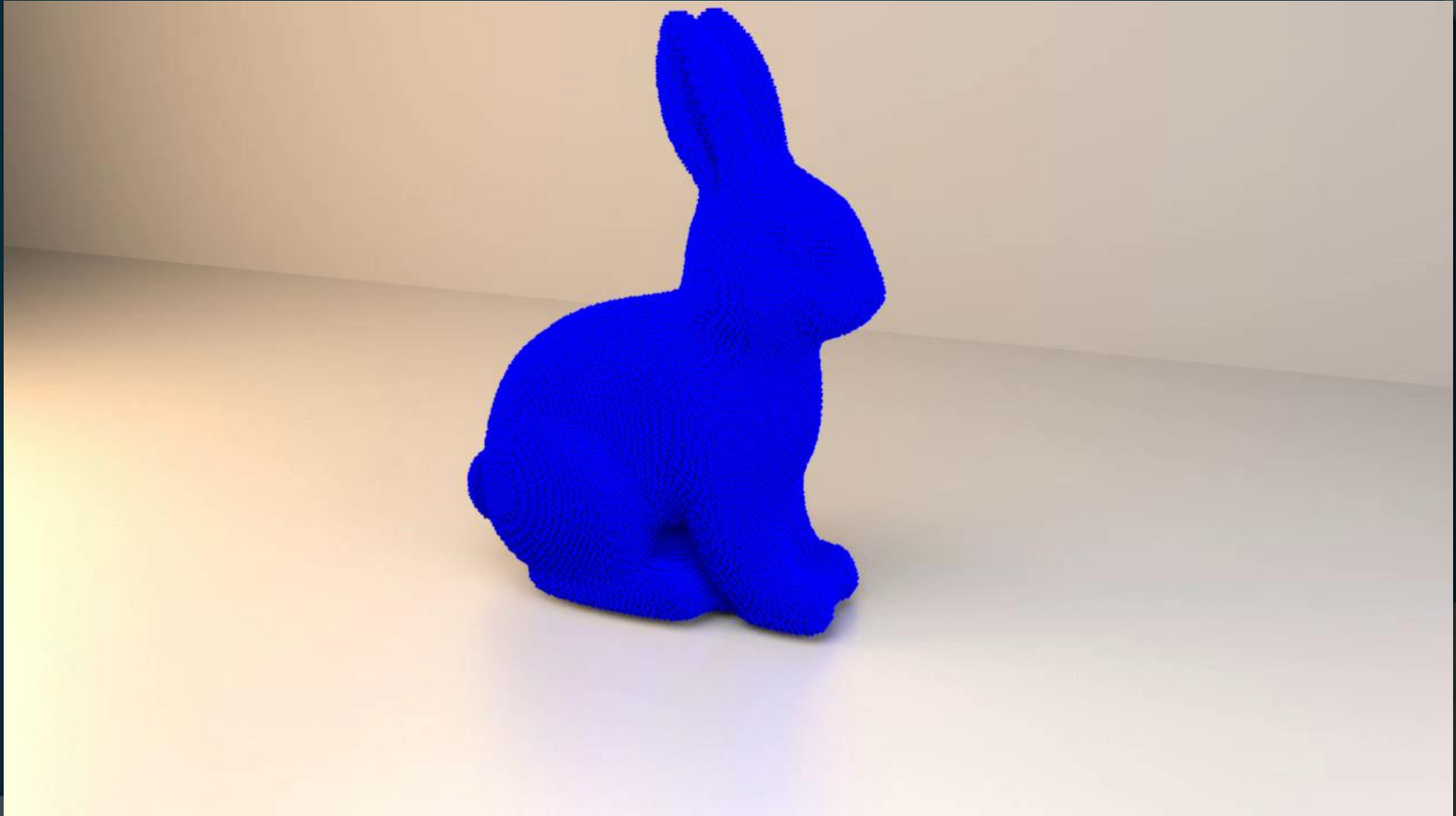
Peer &
Teschner 2016

Bender &
Koschier 2017

Takahashi
et al. 2015

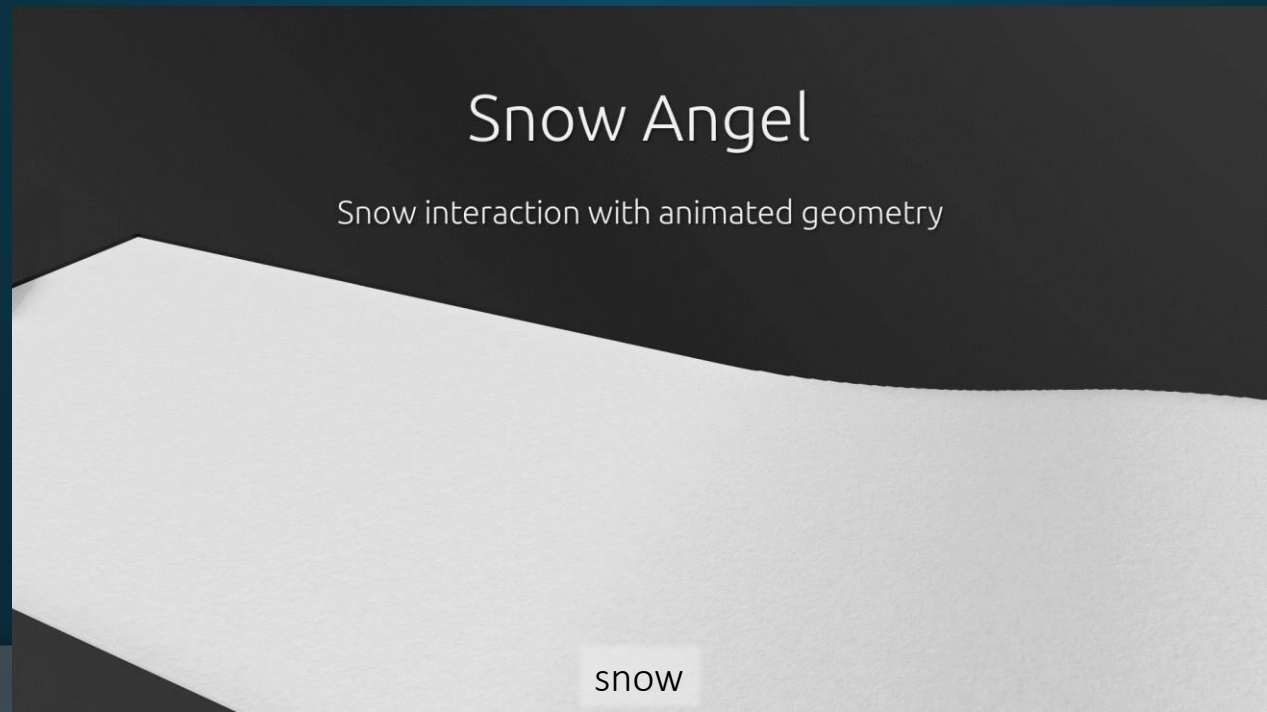
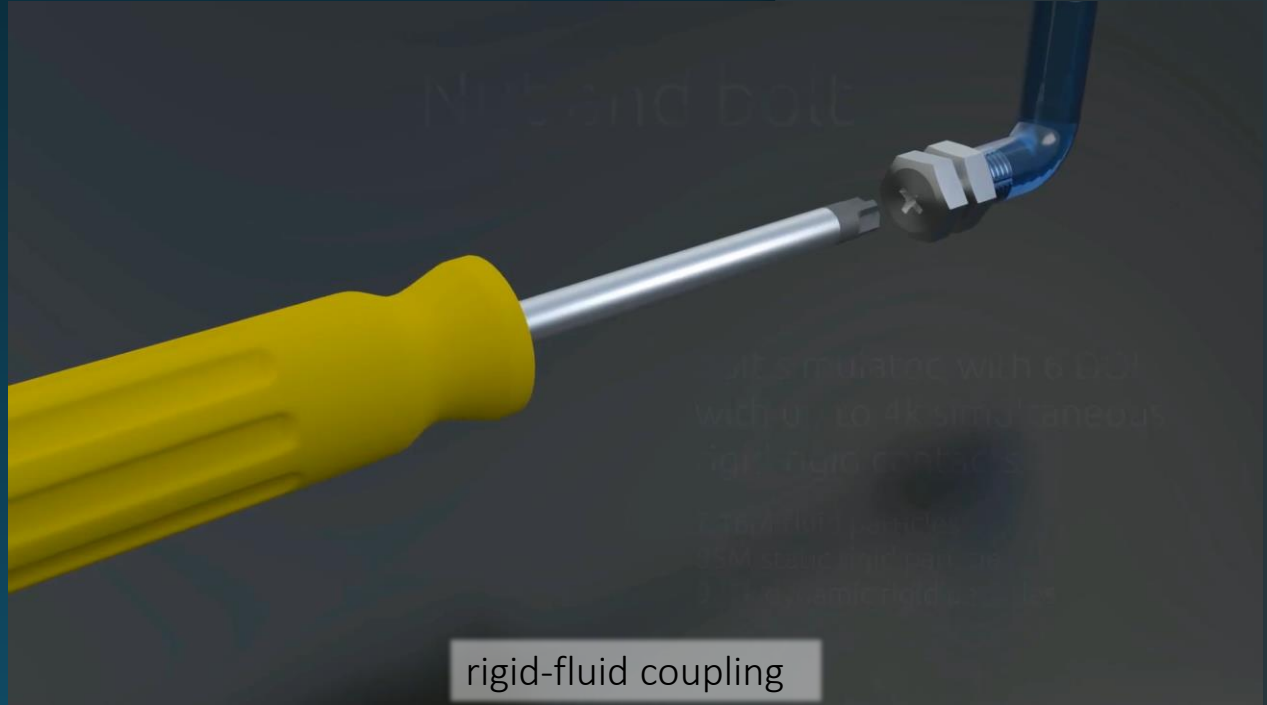
Weiler et al.
2018

Temperature-Dependent Viscosity



Summary

- Low viscous flow
 - Explicit methods are cheap and well-suited
 - Approximation of Laplacian yields better results while XSPH is slightly faster
- Highly viscous fluids
 - Implicit methods are recommended to guarantee stability
 - Strain rate based SPH formulations lead to artifacts at the free surface
 - A formulation based directly on the Laplacian avoids this problem and generate more realistic results.



Want to try it out?

- SPLisHSPlasH is our open-source SPH library (MIT License)
 - explicit and implicit viscosity methods
 - various pressure solvers
 - different boundary handling approaches
 - effects: surface tension, vorticity, air drag
 - multi-phase simulations
 - deformable solids
 - static and dynamic rigid bodies
 - emitters
 - Python scripting
 - exporters to Maya, Blender, Paraview
 - ...



<https://github.com/InteractiveComputerGraphics/SPLisHSPlasH>