Smoothed Particle Hydrodynamics
Techniques for the Physics Based Simulation of Fluids and Solids

Incompressibility

Dan Koschier  Jan Bender  Barbara Solenthaler  Matthias Teschner
SPH Fluid Solver

- Neighbor search
- Incompressibility
- Boundary handling
Outline

– Introduction
– Concepts
  – State equation
  – Iterative state equation
  – Pressure Poisson equation
– Current developments
Motivation

- Incompressibility is essential for a realistic fluid behavior
  - Less than 0.1% volume / density deviation in typical scenarios
- Inappropriate compression leads, e.g., to volume oscillations or volume loss
- Enforcing incompressibility significantly influences the performance
  - Simple approaches require small time steps
  - Expensive approaches work with large time steps
Approaches

– Minimization of density / volume errors
  – Measure difference of actual and desired density
  – Compute pressure and pressure accelerations that reduce density / volume deviations

– Minimization of velocity divergence
  – Measure the divergence of the velocity field
  – Compute pressure and pressure accelerations that reduce the divergence of the velocity field
Typical Implementation

- Split pressure and non-pressure acceleration
  \[ \frac{Dv(t)}{Dt} = -\frac{1}{\rho(t)} \nabla p(t) + a^{\text{nonp}}(t) \]
- Predict velocity after non-pressure acceleration
  \[ v^* = v(t) + \Delta t a^{\text{nonp}}(t) \]
- Compute pressure such that pressure acceleration either minimizes the divergence of \( v^* \) or the density error after advecting the samples with \( v^* \)
- Update velocity \( v(t + \Delta t) = v^* - \Delta t \frac{1}{\rho(t)} \nabla p(t) \)
  - Minimized density error / divergence at advected samples
Density Invariance vs. Velocity Divergence

- Continuity equation: \( \frac{D\rho_i}{Dt} = -\rho_i \nabla \cdot \mathbf{v}_i \)
  - Time rate of change of the density is related to the divergence of the velocity

\[
\begin{align*}
\frac{D\rho_i}{Dt} &= -\rho_i \nabla \cdot \mathbf{v}_i = 0 & \quad \nabla \cdot \mathbf{v}_i &= 0 \\
\frac{D\rho_i}{Dt} &= -\rho_i \nabla \cdot \mathbf{v}_i < 0 & \quad \nabla \cdot \mathbf{v}_i &> 0 \\
\frac{D\rho_i}{Dt} &= -\rho_i \nabla \cdot \mathbf{v}_i > 0 & \quad \nabla \cdot \mathbf{v}_i &< 0
\end{align*}
\]
Density Invariance vs. Velocity Divergence

- Density invariance
  - Measure and minimize density deviations
- Velocity divergence
  - Measure and minimize the divergence of the velocity field
  - Zero velocity divergence corresponds to zero density change over time $-\rho_i \nabla \cdot \mathbf{v}_i = \frac{D\rho_i}{Dt} = 0$, i.e. the initial density does not change over time
  - Notion of density is not required
Challenges

- Minimizing density deviations can result in *volume oscillations*
  - Density error is going up and down
  - Erroneous fluid dynamics
  - Only very small density deviations are tolerable, e.g. 0.1%

https://www.youtube.com/watch?v=hAPO0xBp5WU
Challenges

- Minimizing the velocity divergence can result in volume loss
  - Divergence errors result in density drift
  - No notion of actual density

[Zhu, Lee, Quigley, Fedkiw, ACM SIGGRAPH 2015]
SPH Graphics Research - Incompressibility

- State equation
  - [Becker 2007]

- Iterative state equation
  - PCISPH [Solenthaler 2009],
    LPSPH [He 2012],
    PBF [Macklin 2013]

- Pressure Poisson equation
  - IISPH [Ihmsen 2013],
    DFSPH [Bender 2015],
    [Cornelis 2018]
Incompressibility – Applications

– Fluids
– Elastic solids
– Rigid bodies
– Monolithic solvers with unified representations
Valley

up to 38M fluid particles interacting with more than 650 rigid bricks, highly viscous mud and an elastic tree

[Gissler et al., presented at ACM SIGGRAPH 2019]
Outline

- Introduction
- Concepts
  - State equation
  - Iterative state equation
  - Pressure Poisson equation
- Current developments
State Equation SPH (SESPH)

– Compute pressure from the density deviation locally with one equation for each sample / particle
– Compute pressure acceleration
State Equations

– Pressure is proportional to density error
  – E.g. \( p_i = k \left( \frac{\rho_i}{\rho_0} - 1 \right) \) or \( p_i = k (\rho_i - \rho_0) \)
    – Referred to as compressible SPH
  – \( p_i = k \left( \left( \frac{\rho_i}{\rho_0} \right)^7 - 1 \right) \)
    – Referred to as weakly compressible SPH

Pressure values in SPH implementations should always be non-negative.
for all particle $i$ do
  find neighbors $j$

for all particle $i$ do
  $\rho_i = \sum_j m_j W_{ij}$
  $p_i = k\left(\frac{p_i}{\rho_0} - 1\right)$

Compute pressure with a state equation

for all particle $i$ do
  $a_{i,\text{nonp}} = \nu \nabla^2 v_i + g$
  $a_{i,p} = -\frac{1}{\rho_i} \nabla p_i$
  $a_i(t) = a_{i,\text{nonp}} + a_{i,p}$

for all particle $i$ do
  $v_i(t + \Delta t) = v_i(t) + \Delta t a_i(t)$
  $x_i(t + \Delta t) = x_i(t) + \Delta t v_i(t + \Delta t)$


**SESPH - Discussion**

- Compression results in pressure
- Pressure gradients result in accelerations from high to low density
- Simple computation, small time steps
- Larger stiffness $\rightarrow$ less compressibility $\rightarrow$ smaller time step
- Stiffness constant $k$ does not govern the pressure, but the compressibility of the fluid
Stiffness Constant – 1D Illustration

- Gravity cancels pressure acceleration
  \[ \mathbf{g} = -\mathbf{a}^p_i = \frac{1}{\rho_i} \nabla p_i = -\sum_j m_j \left( \frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} \right) \nabla W_{ij} \]
  \[ = \sum_j m_j \left( \frac{k(\rho_i - \rho_0)}{\rho_i^2} + \frac{k(\rho_j - \rho_0)}{\rho_j^2} \right) \nabla W_{ij} \]

- Differences between \( p_i \) and \( p_j \) are independent from \( k \)

- Smaller \( k \) results in larger density error \( \rho_i - \rho_0 \) to get the required pressure

\[
p_0 = \rho_0 g(h_1 - h) \\
p_1 = \rho_1 gh_1 \\
p_2 = \rho_2 g(h_1 + h)
\]
SESHPH with Splitting

- Split pressure and non-pressure accelerations
  - Non-pressure acceleration \( a_{i}^{\text{nonp}} \)
  - Predicted velocity \( v_{i}^{*} = v_{i}(t) + \Delta t a_{i}^{\text{nonp}} \)
  - Predicted position \( x_{i}^{*} = x_{i}(t) + \Delta t v_{i}^{*} \)
  - Predicted density \( \rho_{i}^{*}(x_{i}^{*}) \)
  - Pressure \( p \) from predicted density \( \rho_{i}^{*} \)
  - Pressure acceleration \( a_{i}^{p} \)
  - Final velocity and position
    \( v_{i}(t + \Delta t) = v_{i}^{*} + \Delta t a_{i}^{p} = v_{i}^{*} - \Delta t \frac{1}{\rho_{i}^{*}} \nabla p_{i} \)
    \( x_{i}(t + \Delta t) = x_{i}(t) + \Delta t v_{i}(t + \Delta t) \)
SESPH with Splitting

for all particle $i$ do
  find neighbors $j$

for all particle $i$ do
  $a_i^{\text{nonp}} = \nu \nabla^2 v_i + g$
  $v_i^* = v_i(t) + \Delta t a_i^{\text{nonp}}$

for all particle $i$ do
  $\rho_i^* = \sum_j m_j W_{ij} + \Delta t \sum_j m_j (v_i^* - v_j^*) \nabla W_{ij}$
  $p_i = k \left( \frac{\rho_i^*}{\rho_0} - 1 \right)$

for all particle $i$ do
  $a_i^p = -\frac{1}{\rho_i} \nabla p_i$

for all particle $i$ do
  $v_i(t + \Delta t) = v_i^* + \Delta t a_i^p$
  $x_i(t + \Delta t) = x_i(t) + \Delta t v_i(t + \Delta t)$

Density at predicted positions
Pressure at predicted positions
Differential Density Update

– Density at advected positions is often approximated without advecting the samples

– Continuity equation and time discretization

$$\frac{D\rho_i}{Dt} = -\rho_i \nabla \cdot \mathbf{v}_i \quad \frac{\rho_i^* - \rho_i(t)}{\Delta t} = -\rho_i \nabla \cdot \mathbf{v}_i^*$$

– SPH discretization

$$\frac{\rho_i^* - \sum_j m_i W_{ij}}{\Delta t} = -\rho_i \left( -\frac{1}{\rho_i} \sum_j m_j (\mathbf{v}_i^* - \mathbf{v}_j^*) \nabla W_{ij} \right)$$

– Predicted density due to the divergence of $\mathbf{v}_i^*$

$$\rho_i^* = \sum_j m_i W_{ij} + \Delta t \sum_j m_j (\mathbf{v}_i^* - \mathbf{v}_j^*) \nabla W_{ij} \quad \text{Approximate density at predicted positions: } \mathbf{x}_i^* = \mathbf{x}_i(t) + \Delta t \mathbf{v}_i^*$$
**SESPH with Splitting - Discussion**

- Consider competing accelerations
- Take effects of non-pressure accelerations into account when computing the pressure acceleration
- Incompressibility has highest priority
Outline

– Introduction
– Concepts
  – State equation
  – Iterative state equation
  – Pressure Poisson equation
– Current developments
Iterative SESPH with Splitting

- Pressure accelerations are iteratively refined
  - Non-pressure acceleration 
    \[ a_i^{\text{nonp}} \]
  - Predicted velocity
    \[ \mathbf{v}_i^*(t) = \mathbf{v}_i(t) + \Delta t a_i^{\text{nonp}} \]
  - Iterate until convergence
  - Density from predicted position
    \[ \rho_i^*(\mathbf{x}_i, \mathbf{v}_i^*) \]
  - Pressure from predicted density
    \[ p_i \]
  - Pressure acceleration
    \[ a_i^p \]
  - Refine predicted velocity
    \[ \mathbf{v}_i^* = \mathbf{v}_i^* + \Delta t a_i^p \]
  - Final velocity and position
    \[ \mathbf{v}_i(t + \Delta t) = \mathbf{v}_i^* \]
    \[ \mathbf{x}_i(t + \Delta t) = \mathbf{x}_i(t) + \Delta t \mathbf{v}_i(t + \Delta t) \]
Iterative SESPH with Splitting - Motivation

- Iterative update is parameterized by a desired density error
- Provides a fluid state with a guaranteed density error
- Stiffness parameter and form of the state equation govern the convergence rate
Iterative SESPH with Splitting

for all particle $i$ do
  find neighbors $j$

for all particle $i$ do
  $a_{i}^{\text{nonp}} = \nu \nabla^2 v_i + g$  ;  $v_i^* = v_i(t) + \Delta t a_{i}^{\text{nonp}}$

repeat
  for all particle $i$ do
    $\rho_i^* = \sum_j m_j W_{ij} + \Delta t \sum_j m_j (v_i^* - v_j^*) \nabla W_{ij}$
    $p_i = k(\frac{\rho_i^*}{\rho_0} - 1)$

  for all particle $i$ do
    $v_i^* = v_i^* - \Delta t \frac{1}{\rho_i^*} \nabla p_i$

until $\rho_i^* - \rho_0 < \eta$  
  user-defined density error

for all particle $i$ do
  $v_i(t + \Delta t) = v_i^*$  ;  $x_i(t + \Delta t) = x_i(t) + \Delta t v_i(t + \Delta t)$
Iterative SESPH - Variants

– Different quantities are accumulated
  – Velocity changes (local Poisson SPH LPSPH)
  – Pressure (predictive-corrective SPH PCISPH) [Solenthaler 2009]
    – Advantageous, if pressure is required for other computations
  – Distances (position-based fluids PBF)
    \[ \Delta x_i = -\frac{1}{\rho_0} \sum_j \left( \frac{p_i}{\beta_i} + \frac{p_j}{\beta_j} \right) \nabla W_{ij} \]

– Different EOS and stiffness constants are used
  – \( \tilde{p}_i = k(\rho_i - \rho_0) \) with \( k = \frac{\rho_i^* r_i^2}{2\rho_0 \Delta t^2} \) in local Poisson SPH
  – \( p_i = k(\rho_i - \rho_0) \) with \( k = \frac{\rho_0^2}{2m_i^2 \Delta t^2 (\sum_j \nabla W_{ij}^0 \cdot \nabla W_{ij}^0 + \sum_j (\nabla W_{ij}^0 \cdot \nabla W_{ij}^0))} \) in PCISPH
  – \( \tilde{p}_i = k(\frac{\rho_i}{\rho_0} - 1) \) with \( k = 1 \) in PBF
Predictive-Corrective Incompressible SPH - PCISPH

- **Goal**: Computation of pressure accelerations $a^p_i$ that result in rest density $\rho_0$ at all particles

- **Formulation**: Density at the next time step should equal the rest density

\[
\rho(t+\Delta t) = \rho_0 = \sum_i m_i W_{ij} + \Delta t \sum_i m_j (v_i^* - v_j^*) \nabla W_{ij} + \Delta t \sum_i m_j (\Delta t a^p_i - \Delta t a^p_j) \nabla W_{ij}
\]

**Discretized continuity equation**
PCISPH - Assumptions

– **Simplifications** to get one equation with one unknown:
  – Equal pressure at all neighboring samples

\[
\mathbf{a}_i^P = - \sum_j m_j \left( \frac{p_i}{\rho_i} + \frac{p_j}{\rho_j} \right) \nabla W_{ij} \approx -m_i \frac{2p_i}{\rho_0^2} \sum_j \nabla W_{ij}
\]

\[
\rho_0 - \rho_i^* = \Delta t^2 \sum_j m_j \left( -m_i \frac{2p_i}{\rho_0^2} \sum_j \nabla W_{ij} + m_j \frac{2p_j}{\rho_0^2} \sum_k \nabla W_{jk} \right) \nabla W_{ij} \quad \text{Unknown pressures } \rho_i \text{ and } \rho_j
\]

– For sample \( j \), only consider the contribution from \( i \)

\[
\rho_0 - \rho_i^* = \Delta t^2 \sum_j m_j \left( -m_i \frac{2p_i}{\rho_0^2} \sum_j \nabla W_{ij} + m_i \frac{2p_i}{\rho_0^2} \nabla W_{ji} \right) \nabla W_{ij} \quad \text{Unknown pressure } \rho_i
\]

\[
\rho_0 - \rho_i^* = \Delta t^2 m_i^2 \frac{2p_i}{\rho_0^2} \sum_j \left( \nabla W_{ij} - \nabla W_{ji} \right) \nabla W_{ij} = -\Delta t^2 m_i^2 \frac{2p_i}{\rho_0^2} \left( \sum_j \nabla W_{ij} \cdot \sum_j \nabla W_{ij} + \sum_j (\nabla W_{ij} \cdot \nabla W_{ij}) \right)
\]
PCISPH - Solution

– Solve for unknown pressure:

\[
\rho_0 - \rho_i^* = -\Delta t^2 m_i^2 \frac{2p_i}{\rho_0^2} \left( \sum_j \nabla W_{ij} \cdot \sum_j \nabla W_{ij} + \sum_j (\nabla W_{ij} \cdot \nabla W_{ij}) \right)
\]

\[
p_i = \frac{\rho_0^2}{2\Delta t^2 m_i^2 (\sum_j \nabla W_{ij} \cdot \sum_j \nabla W_{ij} + \sum_j (\nabla W_{ij} \cdot \nabla W_{ij}))} (\rho_i^* - \rho_0) \quad (p_i = k(\rho_i^* - \rho_0))
\]

Intuition: This pressure causes pressure accelerations that cause velocity changes that correspond to a divergence that results in rest density at the sample.

\[
\rho(t + \Delta t) = \rho_0 = \rho_i^* + \Delta t \sum_j m_j (\Delta t \alpha_i^p - \Delta t \alpha_j^p) \nabla W_{ij}
\]
PCISPH - Discussion

- Pressure is computed with a state equation \( p_i = k_i (\rho_i^* - \rho_0) \)
- \( k_i \) is not user-defined
- Instead, an optimized value \( k_i \) is derived and used
- Pressure is iteratively refined
PCISPH - Performance

– Typically three to five iterations for density errors between 0.1% and 1%
– Speed-up factor over non-iterative SESPH up to 50
  – More computations per time step compared to SESPH
  – Significantly larger time step than in SESPH
  – Speed-up dependent on scenario
– Non-linear relation between time step and iterations
  – Largest possible time step does not necessarily lead to an optimal overall performance
Comparison

- PCISPH [Solenthaler 2009]
  - Iterative pressure computation
  - Large time step
- WCSPH [Becker and Teschner 2007]
  - Efficient to compute
  - Small time step
- Computation time for the PCISPH scenario is 20 times shorter than WCSPH
Outline

– Introduction
– Concepts
  – State equation
  – Iterative state equation
  – Pressure Poisson equation
– Current developments


Introduction

– Pressure causes pressure accelerations that cause velocity change that cause displacements such that particles have rest density

– Projection schemes solve a linear system to compute the respective pressure field
  – PCISPH uses simplifications to compute pressure per particle from one equation. Solving a linear system is avoided.
Derivation

\[ \frac{Dv(t)}{Dt} = -\frac{1}{\rho} \nabla p(t) + a_{\text{nonp}}(t) \]

Velocity change per time step due to pressure acceleration and non-pressure acceleration

\[ v^* = v(t) + \Delta t a_{\text{nonp}}(t) \]

Predicted velocity after non-pressure acceleration

\[ v(t + \Delta t) = v^* - \Delta t \frac{1}{\rho} \nabla p(t) \]

Velocity after all accelerations

\[ v(t + \Delta t) - v^* = -\Delta t \frac{1}{\rho} \nabla p(t) \]

Velocity change due to pressure acceleration

\[ \nabla \cdot (v^* - v(t + \Delta t)) = \nabla \cdot \left( \Delta t \frac{1}{\rho} \nabla p(t) \right) \]

Divergence of the velocity change due to pressure acceleration
Derivation

\[ \nabla \cdot (\mathbf{v}^* - \mathbf{v}(t + \Delta t)) = \nabla \cdot \left( \Delta t \frac{1}{\rho} \nabla p(t) \right) \]

\[ \nabla \cdot \mathbf{v}^* - \nabla \cdot \mathbf{v}(t + \Delta t) = \nabla \cdot \left( \Delta t \frac{1}{\rho} \nabla p(t) \right) \]

Constraint: \[ \nabla \cdot \mathbf{v}(t + \Delta t) = 0 \]

Divergence of the final velocity field should be zero, i.e. no density change per time

\[ \nabla \cdot \mathbf{v}^* = -\nabla \cdot (\Delta t \mathbf{a}^P) \]

Divergence of the velocity change due to pressure acceleration should cancel the divergence of the predicted velocity

\[ \rho \nabla \cdot \mathbf{v}^* = \Delta t \nabla^2 p(t) \]

Pressure Poisson equation with unknown pressure
Density Invariance vs. Velocity Divergence

– Pressure Poisson equation PPE that minimizes the velocity divergence: \( \Delta t \nabla^2 p(t) = \rho \nabla \cdot v^* \)

– PPE that minimizes the density error: \( \Delta t \nabla^2 p(t) = \frac{\rho_0 - \rho^*}{\Delta t} \)

– Derivation:

\[
\frac{D\rho(t+\Delta t)}{Dt} + \rho(t + \Delta t) \nabla \cdot v(t + \Delta t) = 0
\]

Constraint: \( \rho(t + \Delta t) = \rho_0 \)

\[
\frac{\rho_0 - \rho(t)}{\Delta t} + \rho_0 \nabla \cdot \left( v^* - \Delta t \frac{1}{\rho_0} \nabla p(t) \right) = 0
\]

\[
\frac{\rho_0 - (\rho(t) - \Delta t \rho_0 \nabla \cdot v^*)}{\Delta t} - \Delta t \nabla^2 p(t) = 0
\]

\[
\rho^* = \rho(t) - \Delta t \rho_0 \nabla \cdot v^*
\]
Interpretation of PPE Forms

- **Velocity divergence:** 
  \[ -\Delta t \frac{1}{\rho} \nabla^2 p = -\nabla \cdot \mathbf{v}^* \]

  - Pressure $p$ causes a pressure acceleration $-\frac{1}{\rho} \nabla p$ that causes a velocity change $-\Delta t \frac{1}{\rho} \nabla p$ whose divergence $\nabla \cdot (-\Delta t \frac{1}{\rho} \nabla p)$ cancels the divergence $\nabla \cdot \mathbf{v}^*$ of the predicted velocity, i.e.
  \[ \nabla \cdot \mathbf{v}^* + \nabla \cdot (-\Delta t \frac{1}{\rho} \nabla p) = 0 \]

- **Density invariance:** 
  \[ -\Delta t \nabla^2 p = -\frac{\rho_0 - \rho^*}{\Delta t} \]

  - The divergence $\nabla \cdot (-\Delta t \frac{1}{\rho} \nabla p)$ multiplied with density $\rho$ is a density change per time that cancels the predicted density error per time $\frac{\rho_0 - \rho^*}{\Delta t}$, i.e.
  \[ \frac{\rho_0 - \rho^*}{\Delta t} + \rho \nabla \cdot (-\Delta t \frac{1}{\rho} \nabla^2 p) = 0 \]
**PPE Solver**

- Linear system with unknown pressure values $A p = s$
  - One equation per particle $(A p)_i = s_i \quad (\Delta t < \nabla^2 p_i > = \frac{\rho_0 - <\rho_i^* >}{\Delta t})$
- Iterative solvers
  - Conjugate Gradient
  - Relaxed Jacobi
- Fast computation per iteration
  - Few non-zero entries in each equation
  - Matrix-free implementations
  - Very few information per particle

$<A>$ is a discretized form of $A$
**PPE Solver**

- Very large time steps
- Convergence dependent on the formulation
  - SPH discretization of $\nabla^2 p$
  - Source term (velocity divergence or density invariance)
- Accuracy issues
  - Volume drift for velocity divergence
  - Oscillations for density invariance
**PPE Discretization**

- **Implicit incompressible SPH (IISPH) [Ihmsen et al. 2014]**
  - PPE with density invariance as source term: $\Delta t^2 \nabla^2 p = \rho_0 - \rho^*$
  - Computation of $\rho_i^*$:
    \[
    \rho_i^* = \rho_i + \Delta t \sum_j m_j v_{ij}^* \nabla W_{ij} \quad \text{with} \quad v_{ij}^* = v_i + \Delta t a_{i}^{\text{nonp}}
    \]
  - Computation of $\Delta t^2 \nabla^2 p_i$:
    \[
    \Delta t^2 \nabla^2 p_i = -\Delta t \rho_i \nabla \cdot (\Delta t a_i^p) = \Delta t^2 \sum_j m_j (a_i^p - a_j^p) \cdot \nabla W_{ij}
    \]
    with
    \[
    a_i^p = -\frac{1}{\rho_i} \nabla p_i = -\sum_j m_j \left( \frac{v_i}{\rho_i^2} + \frac{v_j}{\rho_j^2} \right) \nabla W_{ij}
    \]
PPE System - IISPH

- **PPE**
  \[ \Delta t^2 \nabla^2 p_i = \rho_0 - \rho_i^* \]
  
  density change due to pressure accelerations negative of the predicted density error

- Discretized PPE
  - **System:** \[ A p = s \]
  - **Per particle:** \[ \Delta t^2 \sum_j m_j (\alpha_i^p - \alpha_j^p) \nabla W_{ij} = \rho_0 - \rho_i^* \]
    \[ \alpha_i^p = - \sum_j m_j \left( \frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} \right) \nabla W_{ij} \]
  - **Interpretation:**
    \[ \Delta t \sum_j m_j (\Delta t \alpha_i^p - \Delta t \alpha_j^p) \nabla W_{ij} = \rho_0 - \rho_i^* \]
    \[ \Delta t \sum_j m_j (v_i^p - v_j^p) \nabla W_{ij} = \rho_0 - \rho_i^* \]
    \[ \Delta t \cdot \rho_i \cdot \nabla \cdot v_i^p = \rho_0 - \rho_i^* \]
  
  Pressure accelerations cause a velocity change \( \mathbf{v}^p \) whose divergence causes a density change.
PPE Solver - IISPH

- Relaxed Jacobi: $p_i^{l+1} = \max \left( p_i^l + \omega \frac{s_i - (Ap_i^l)}{a_{ii}}, 0 \right)$
  - For IISPH, typically $\omega = 0.5$
  - Diagonal element $a_{ii}$
    - Accumulate all coefficients of $p_i$ in $\Delta t^2 \sum_j m_j (\alpha_i^p - \alpha_j^p) \nabla W_{ij}$
    - $a_{ii} = \Delta t^2 \sum_j m_j \left( - \sum_j \frac{m_j}{\rho_j^2} \nabla W_{ij} \right) \cdot \nabla W_{ij} + \Delta t^2 \sum_j m_j \left( \frac{m_j}{\rho_i^2} \nabla W_{ji} \right) \cdot \nabla W_{ij}$

- Note, that the first pressure update is $p_i^1 = 0 + \omega \frac{s_i}{a_{ii}} = \omega \frac{\rho_0 - \rho^*}{a_{ii}}$ State equation
- Using the incompressible PPE variant IISPH with one solver iteration corresponds to compressible state-equation SPH with $p_i = -\frac{\omega}{a_{ii}}(\rho^* - \rho_0)$
PPE Solver Implementation - IISPH

- Initialization:
  \[ \rho_i = \sum_j m_j W_{ij}, \quad a_{ii} = \ldots \]
  \[ \mathbf{v}_{i}^{*} = \mathbf{v}_{i} + \Delta t \mathbf{a}_{i}^{\text{nonp}} \]
  \[ s_{i} = \rho_0 - \rho_{i} - \Delta t \sum_j m_j \mathbf{v}_{i}^{*} \nabla W_{ij} \]
  \[ p_{i}^{0} = \max \left( \omega \frac{s_{i}}{a_{ii}}, 0 \right) \]

- Solver update in iteration \( l \):
  - First loop:
    \[ (\mathbf{a}_{i}^{p})^{l} = -\sum_j m_j \left( \frac{p_{i}^{l}}{\rho_{i}^{2}} + \frac{p_{j}^{l}}{\rho_{j}^{2}} \right) \nabla W_{ij} \]
  - Second loop:
    \[ (\mathbf{A} \mathbf{p})^{l} = \Delta t^2 \sum_j m_j \left( (\mathbf{a}_{i}^{p})^{l} - (\mathbf{a}_{j}^{p})^{l} \right) \nabla W_{ij} \]
    \[ p_{i}^{l+1} = \max \left( p_{i}^{l} + \omega \frac{s_{i} - (\mathbf{A} \mathbf{p})^{l}}{a_{ii}}, 0 \right) \quad \text{If } a_{ii} \text{ not equal to zero} \]
    \[ (\rho_{i}^{\text{error}})^{l} = (\mathbf{A} \mathbf{p})^{l} - s_{i} \quad \text{Continue until error is small} \]
Boundary Handling - IISPH

- PPE: \( \Delta t^2 \nabla^2 p_f = \rho_0 - \rho_f^* = \rho_0 - \rho_f + \Delta t \rho_0 \nabla \cdot \mathbf{v}_f^* \)

- Discretized PPE: \( A \mathbf{p} = \mathbf{s} \)

\[
\begin{align*}
(A \mathbf{p})_f &= \Delta t^2 \sum_{f_f} m_{f_f} \left( a^p_f - a^p_{f_f} \right) \nabla W_{f_f f} + \Delta t^2 \sum_{b_f} m_{b_f} a^p_f \nabla W_{f_f b_f} \\
&\quad + \sum_{f_f} \frac{m_{f_f}}{\rho_f^2} \left( p_f + \frac{p_f}{\rho_f^2} \right) \nabla W_{f_f f} - \gamma \sum_{b_f} m_{b_f} \frac{p_f}{\rho_f^2} \nabla W_{f_f b_f} \\
&\quad + \Delta t \sum_{f_f} m_{f_f} \left( \mathbf{v}_f^* - \mathbf{v}_{f_f}^* \right) \nabla W_{f_f f} - \Delta t \sum_{b_f} m_{b_f} \left( \mathbf{v}_f^* - \mathbf{v}_{b_f} (t + \Delta t) \right) \nabla W_{f_f b_f}
\end{align*}
\]

Index \( f \) indicates a fluid sample.  
Index \( b \) indicates a boundary sample.  
\( f_f \) indicates a fluid neighbor of \( f \).  
\( b_f \) indicates a boundary neighbor of \( f \).
Boundary Handling - IISPH

- Diagonal element

\[
a_{ff} = \Delta t^2 \sum_{ff} m_{ff} \left( - \sum_{ff} \frac{m_{ff}}{\rho_{ff}^2} \nabla W_{ff} - 2\gamma \sum_{fb} \frac{m_{fb}}{\rho_0^2} \nabla W_{ff} \right) \nabla W_{ff} \\
+ \Delta t^2 \sum_{ff} m_{ff} \left( \frac{m_{f}}{\rho_f^2} \nabla W_{ff} \right) \nabla W_{ff} \\
+ \Delta t^2 \sum_{fb} m_{fb} \left( - \sum_{ff} \frac{m_{ff}}{\rho_{ff}^2} \nabla W_{ff} - 2\gamma \sum_{fb} \frac{m_{fb}}{\rho_0^2} \nabla W_{ff} \right) \nabla W_{ff}
\]
IISPH with Boundary - Implementation

- Initialization:  
  \[ \rho_f = \sum_{f} m_{f} W_{f f} + \sum_{b} m_{b} W_{f b} \quad a_{f f} = \ldots \]
  \[ v_f^* = v_f + \Delta t a_{f}^{\text{nonp}} \]
  \[ s_f = \rho_0 - \rho_f - \Delta t \sum_{f} m_{f} v_{f f}^* \nabla W_{f f} - \Delta t \sum_{b} m_{b} v_{f b}^* \nabla W_{f b} \]
  \[ p_{f}^{0} = \max \left( \omega \frac{s_f}{a_{f f}}, 0 \right) \]

- Solver update in iteration /:
  - First loop:  
    \[ (a_{f}^{p})^l = - \sum_{f} m_{f} \left( \frac{p_{f}^{l}}{\rho_{f}^{2}} + \frac{v_{f f}^{l}}{\rho_{f}^{2}} \right) \nabla W_{f f} - \gamma \sum_{b} m_{b} 2 \frac{p_{f}^{l}}{\rho_{f}^{2}} \nabla W_{f b} \]
  - Second loop:  
    \[ (A p_{f}^{l}) = \Delta t^2 \sum_{f} m_{f} \left( (a_{f}^{p})^{l} - (a_{f}^{p})^{l} \right) \nabla W_{f f} + \Delta t^2 \sum_{b} m_{b} (a_{f}^{p})^{l} \nabla W_{f b} \]
    \[ p_{f}^{l+1} = \max \left( p_{f}^{l} + \omega \frac{s_{f} - (A p_{f}^{l})}{a_{f f}}, 0 \right) \quad \text{If } a_{f f} \text{ not equal to zero} \]
    \[ (\rho_{f}^{\text{error}})^l = (A p_{f}^{l}) - s_{f} \quad \text{Continue until error is small} \]
IISPH vs. PCISPH

- Breaking dam
  - 100k samples with diameter 0.05m, 0.01% ave density error

<table>
<thead>
<tr>
<th>$\Delta t$ [s]</th>
<th>avg. iter.</th>
<th>total comp. time [s]</th>
<th>avg. iter.</th>
<th>total comp. time [s]</th>
<th>PCISPH / IISPH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>pressure</td>
<td>overall</td>
<td>pressure</td>
<td>overall</td>
</tr>
<tr>
<td>0.0005</td>
<td>4.3</td>
<td>540</td>
<td>1195</td>
<td>2.2</td>
<td>148</td>
</tr>
<tr>
<td>0.00067</td>
<td>7.2</td>
<td>647</td>
<td>1145</td>
<td>2.9</td>
<td>149</td>
</tr>
<tr>
<td>0.001</td>
<td>14.9</td>
<td>856</td>
<td>1187</td>
<td>4.9</td>
<td>164</td>
</tr>
<tr>
<td>0.0025</td>
<td>66.5</td>
<td>1495</td>
<td>1540</td>
<td>18.4</td>
<td>242</td>
</tr>
<tr>
<td>0.004</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>33.5</td>
<td>273</td>
</tr>
<tr>
<td>0.005</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>45.8</td>
<td>297</td>
</tr>
</tbody>
</table>

- Largest possible time step does not necessarily result in the best performance
Up to 500 million fluid samples
Outline

– Introduction
– Concepts
  – State equation
  – Iterative state equation
  – Pressure Poisson equation
– Current developments
Current Developments

– DFSPH [Bender 2015]
  – Combination of two PPEs (inspired by [Hu 2007])
  – Resolving compressibility and removing velocity divergence in two steps
  – Currently the most efficient solver
– [Cornelis 2018]
  – Various formulations for combining two PPEs
– [Fuerstenau 2017]
  – Discretization of the Laplacian