Extending OpenSees for the Particle Finite Element Method

Michael H. Scott   Minjie Zhu

Oregon State University OSU

OpenSees Days 2012
UC Berkeley
August 16, 2012

Outline

1. Motivation
2. Particle Finite Element Method
   - Introduction to PFEM
   - Basic Equations
   - Fractional Step Method (FSM)
3. Implementation in OpenSees
   - Problems of the implementation of FSM
   - Options and solution to the implementation of FSM
   - Example
   - Mesh and Remesh
4. Sensitivity Analysis of PFEM
   - Introduction to the sensitivity
   - Computing sensitivity
   - Example

Fluid-Structure Interaction

- Recent natural disasters
- Simulating structural response to wave loading

OSU's Hinsdale Wave Research Lab (www.flickr.com/photos/scottfrey)

- Computational challenges
- The "OpenSees" homophone

Introduction to PFEM

Introduction to PFEM

- Lagrangian formulation
  - The position and physical properties of the particles are described in terms of the material or referential coordinates and time which is normally used in solid mechanics

Introduction to PFEM

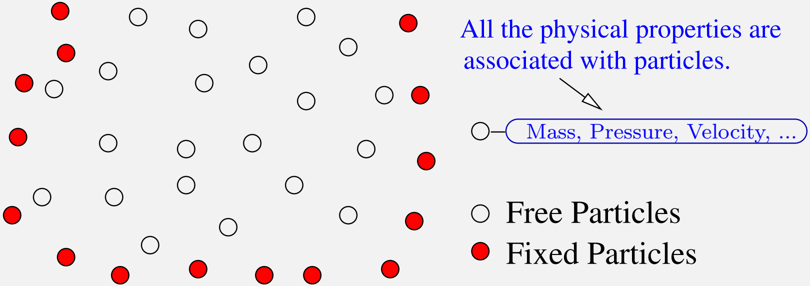
- Lagrangian formulation
  - The position and physical properties of the particles are described in terms of the material or referential coordinates and time which is normally used in solid mechanics
- Effective for solving free-surface flow problems
  - Changes in the position and physical properties as the material particles move in free-surface are easy to capture
  - Delaunay tessellation and Alpha shape method ⇒ mesh
  - Fractional Step Method ⇒ fast iterative solver
  - Finite calculus method ⇒ stabilize the incompressible condition

Introduction to PFEM

- Lagrangian formulation
  - The position and physical properties of the particles are described in terms of the material or referential coordinates and time which is normally used in solid mechanics
- Effective for solving free-surface flow problems
  - Changes in the position and physical properties as the material particles move in free-surface are easy to capture
  - Delaunay tessellation and Alpha shape method $\Rightarrow$ mesh
  - Fractional Step Method $\Rightarrow$ fast iterative solver
  - Finite calculus method $\Rightarrow$ stabilize the incompressible condition
- Natural for solving solid-fluid interaction problems

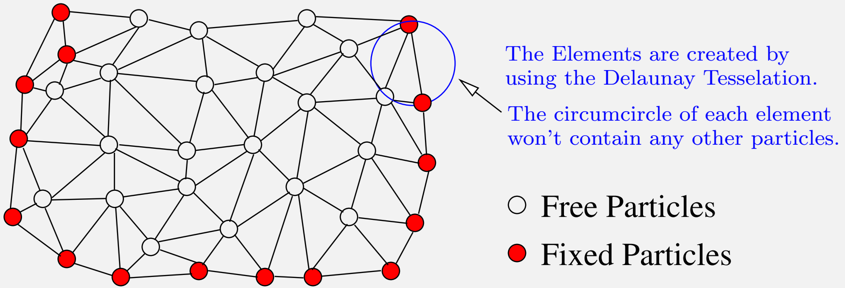
Introduction to PFEM

- Particle Based Method
- Finite Element Method
- Particles Moving and Elements Updating



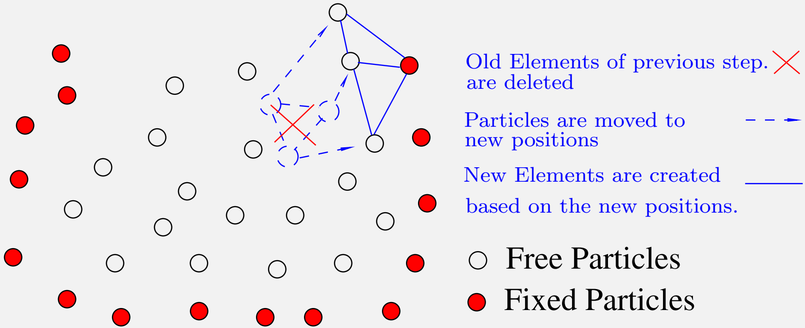
Introduction to PFEM

- Particle Based Method
- Finite Element Method
- Particles Moving and Elements Updating



Introduction to PFEM

- Particle Based Method
- Finite Element Method
- Particles Moving and Elements Updating



Basic Equations

Basic Equations

- Conservation Laws

Mass:

$$\frac{D\rho}{Dt} + \rho\frac{\partial u_i}{\partial x_i} = 0$$

Momentum:

$$\rho\frac{Du_i}{Dt} = -\frac{\partial}{\partial x_i}P + \frac{\partial}{\partial x_j}\sigma_{ij} + \rho f_i$$

Where,

ρ – Density, u_i – Velocities

x_i – Coordinates, P – Pressure

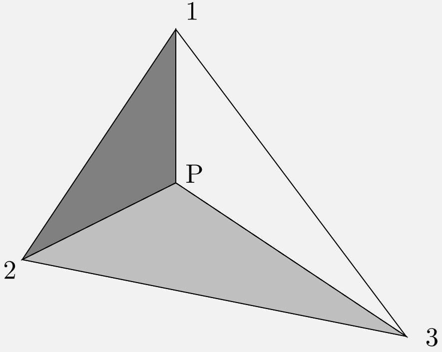
σ_{ij} – Cauchy Stress, f_i – Body forces

Basic Equations

- Linear interpolation (Unstable for incompressible continuum)

$$N_i = L_i \qquad i = 1, 2, 3 \text{ For 2D}$$

Where, N_i is the shape function, L_i is the area coordinate for node i



For point P, the area coordinates

$$L_1 = \frac{areaP23}{area123}$$

$$L_2 = \frac{areaP13}{area123}$$

$$L_3 = \frac{areaP12}{area123}$$

Basic Equations

- Discretized equations (stabilized by Finite Calculus Method)

$$\mathbf{M}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}$$

where,

$$\mathbf{M} = \begin{bmatrix} \bar{\mathbf{M}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

$$\mathbf{K} = \begin{bmatrix} \bar{\mathbf{K}} & -\mathbf{G} & \mathbf{0} \\ \mathbf{G}^T & \mathbf{L} & \mathbf{Q} \\ \mathbf{0} & \mathbf{Q}^T & \hat{\mathbf{M}} \end{bmatrix}$$

$$\mathbf{F} = \begin{bmatrix} \bar{\mathbf{F}} & \mathbf{0} & \mathbf{0} \end{bmatrix}^T$$

$$\mathbf{U} = \begin{bmatrix} \mathbf{v} & \mathbf{p} \end{bmatrix}^T$$

$$\dot{\mathbf{U}} = \begin{bmatrix} \dot{\mathbf{v}} & \dot{\mathbf{p}} \end{bmatrix}^T$$

$\bar{\mathbf{M}}$ – Mass matrix

$\bar{\mathbf{K}}$ – Stiffness matrix

$\mathbf{G}$ – Gradient operator

$\mathbf{L}$ – Laplacian operator

$\mathbf{Q}, \hat{\mathbf{M}}$ – Stabilization matrices

$\bar{\mathbf{F}}$ – Force vector

Basic Equations

- Discretized equations (stabilized by Finite Calculus Method)

$$\mathbf{M}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}$$

where,

$$\mathbf{M} = \begin{bmatrix} \bar{\mathbf{M}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

$$\mathbf{K} = \begin{bmatrix} \bar{\mathbf{K}} & -\mathbf{G} & \mathbf{0} \\ \mathbf{G}^T & \mathbf{L} & \mathbf{Q} \\ \mathbf{0} & \mathbf{Q}^T & \hat{\mathbf{M}} \end{bmatrix}$$

$$\mathbf{F} = \begin{bmatrix} \bar{\mathbf{F}} & \mathbf{0} & \mathbf{0} \end{bmatrix}^T$$

$$\mathbf{U} = \begin{bmatrix} \mathbf{v} & \mathbf{p} \end{bmatrix}^T$$

$$\dot{\mathbf{U}} = \begin{bmatrix} \dot{\mathbf{v}} & \dot{\mathbf{p}} \end{bmatrix}^T$$

$\bar{\mathbf{M}}$ – Mass matrix

$\bar{\mathbf{K}}$ – Stiffness matrix

$\mathbf{G}$ – Gradient operator

$\mathbf{L}$ – Laplacian operator

$\mathbf{Q}, \hat{\mathbf{M}}$ – Stabilization matrices

$\bar{\mathbf{F}}$ – Force vector

Blue matrices are related to the stability issues of incompressible fluid.

Fractional Step Method (FSM)

Although stabilized, the discretized equations

Fractional Step Method (FSM)

Although stabilized, the discretized equations

$$\bar{\mathbf{M}}\dot{\mathbf{v}} + \bar{\mathbf{K}}\mathbf{v} - \mathbf{G}\mathbf{p} = \mathbf{f}$$
$$\mathbf{G}^T\mathbf{v} + \mathbf{L}\mathbf{p} + \mathbf{Q}\boldsymbol{\pi} = \mathbf{0}$$
$$\mathbf{Q}^T\mathbf{p} + \hat{\mathbf{M}}\boldsymbol{\pi} = \mathbf{0}$$

Fractional Step Method (FSM)

Although stabilized, the discretized equations

$$\bar{\mathbf{M}}\dot{\mathbf{v}} + \bar{\mathbf{K}}\mathbf{v} - \mathbf{G}\mathbf{p} = \mathbf{f}$$
$$\mathbf{G}^T\mathbf{v} + \mathbf{L}\mathbf{p} + \mathbf{Q}\boldsymbol{\pi} = \mathbf{0}$$
$$\mathbf{Q}^T\mathbf{p} + \hat{\mathbf{M}}\boldsymbol{\pi} = \mathbf{0}$$

are ill-conditioned because the velocity and pressure fields are coupled.

Fractional Step Method (FSM)

Although stabilized, the discretized equations

$$\bar{\mathbf{M}}\dot{\mathbf{v}} + \bar{\mathbf{K}}\mathbf{v} - \mathbf{G}\mathbf{p} = \mathbf{f}$$
$$\mathbf{G}^T\mathbf{v} + \mathbf{L}\mathbf{p} + \mathbf{Q}\boldsymbol{\pi} = \mathbf{0}$$
$$\mathbf{Q}^T\mathbf{p} + \hat{\mathbf{M}}\boldsymbol{\pi} = \mathbf{0}$$

are ill-conditioned because the velocity and pressure fields are coupled.

Large errors using direct solvers or slow convergence for iterative solvers can be expected.

Fractional Step Method (FSM)

Although stabilized, the discretized equations

$$\bar{\mathbf{M}}\dot{\mathbf{v}} + \bar{\mathbf{K}}\mathbf{v} - \mathbf{G}\mathbf{p} = \mathbf{f}$$
$$\mathbf{G}^T\mathbf{v} + \mathbf{L}\mathbf{p} + \mathbf{Q}\boldsymbol{\pi} = \mathbf{0}$$
$$\mathbf{Q}^T\mathbf{p} + \hat{\mathbf{M}}\boldsymbol{\pi} = \mathbf{0}$$

are ill-conditioned because the velocity and pressure fields are coupled.

Large errors using direct solvers or slow convergence for iterative solvers can be expected.

To overcome this problem, the fractional step method (FSM) is used to solve the discretized equations.

Fractional Step Method (FSM)

- Predictor of velocity $\mathbf{v}^*$

$$\mathbf{v}^* = \mathbf{v}^n - \Delta t \bar{\mathbf{M}}^{-1} \left[\bar{\mathbf{K}} \mathbf{v}^j - \mathbf{G} \mathbf{p}^j - \mathbf{f} \right]$$

Fractional Step Method (FSM)

- Predictor of velocity $\mathbf{v}^*$

$$\mathbf{v}^* = \mathbf{v}^n - \Delta t \bar{\mathbf{M}}^{-1} \left[\bar{\mathbf{K}} \mathbf{v}^j - \mathbf{G} \mathbf{p}^j - \mathbf{f} \right]$$

- Pressure increment $\delta \mathbf{p}$

$$\delta \mathbf{p} = -(\mathbf{L} + \Delta t \mathbf{S})^{-1} \left[\mathbf{G}^T \mathbf{v}^* + \mathbf{Q} \boldsymbol{\pi}^j + \mathbf{L} \mathbf{p}^j \right]$$

where,

$$\mathbf{S} = \mathbf{G}^T \bar{\mathbf{M}}^{-1} \mathbf{G}$$

Fractional Step Method (FSM)

- Predictor of velocity $\mathbf{v}^*$

$$\mathbf{v}^* = \mathbf{v}^n - \Delta t \bar{\mathbf{M}}^{-1} \left[\bar{\mathbf{K}} \mathbf{v}^j - \mathbf{G}\mathbf{p}^j - \mathbf{f} \right]$$

- Pressure increment $\delta \mathbf{p}$

$$\delta \mathbf{p} = -(\mathbf{L} + \Delta t \mathbf{S})^{-1} \left[\mathbf{G}^T \mathbf{v}^* + \mathbf{Q}\pi^j + \mathbf{L}\mathbf{p}^j \right]$$

where,

$$\mathbf{S} = \mathbf{G}^T \bar{\mathbf{M}}^{-1} \mathbf{G}$$

- Corrector of velocity $\mathbf{v}^{j+1}$

$$\mathbf{v}^{j+1} = \mathbf{v}^* + \Delta t \bar{\mathbf{M}}^{-1} \mathbf{G} \delta \mathbf{p}$$

Fractional Step Method (FSM)

- Predictor of velocity $\mathbf{v}^*$

$$\mathbf{v}^* = \mathbf{v}^n - \Delta t \bar{\mathbf{M}}^{-1} \left[\bar{\mathbf{K}} \mathbf{v}^j - \mathbf{G} \mathbf{p}^j - \mathbf{f} \right]$$

- Pressure increment $\delta \mathbf{p}$

$$\delta \mathbf{p} = -(\mathbf{L} + \Delta t \mathbf{S})^{-1} \left[\mathbf{G}^T \mathbf{v}^* + \mathbf{Q} \boldsymbol{\pi}^j + \mathbf{L} \mathbf{p}^j \right]$$

where,

$$\mathbf{S} = \mathbf{G}^T \bar{\mathbf{M}}^{-1} \mathbf{G}$$

- Corrector of velocity $\mathbf{v}^{j+1}$

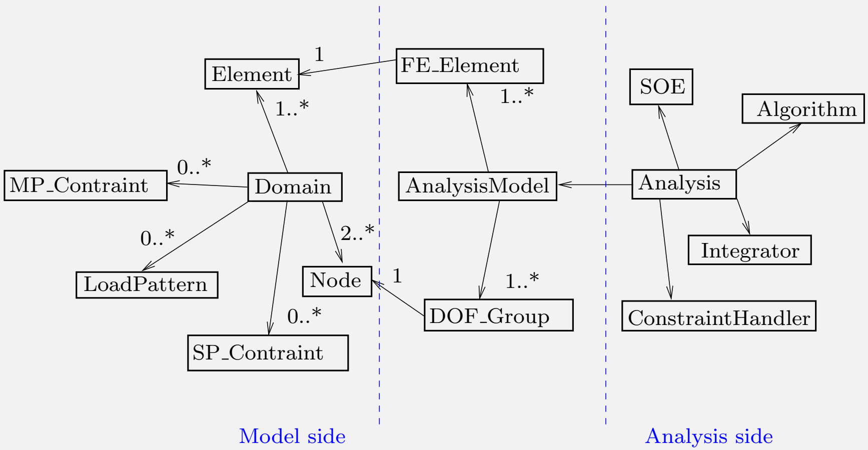
$$\mathbf{v}^{j+1} = \mathbf{v}^* + \Delta t \bar{\mathbf{M}}^{-1} \mathbf{G} \delta \mathbf{p}$$

- Pressure gradient $\boldsymbol{\pi}^{j+1}$

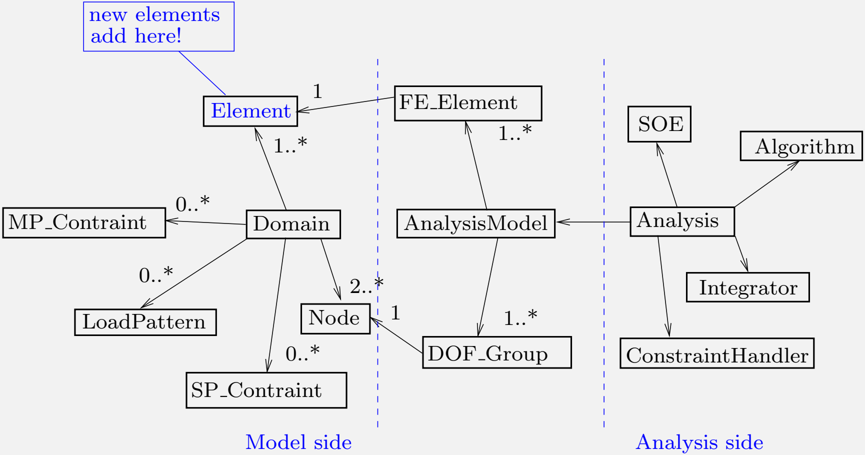
$$\boldsymbol{\pi}^{j+1} = -\hat{\mathbf{M}}^{-1} \mathbf{Q}^T \mathbf{p}^{j+1}$$

Outline

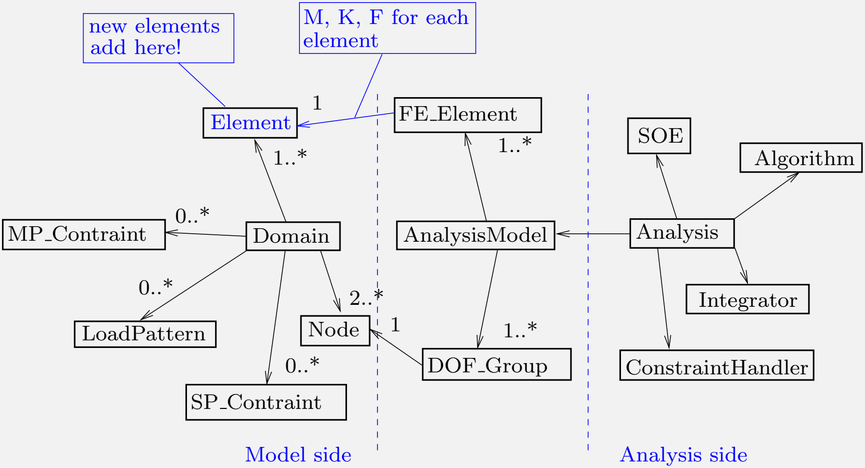
OpenSees class diagram



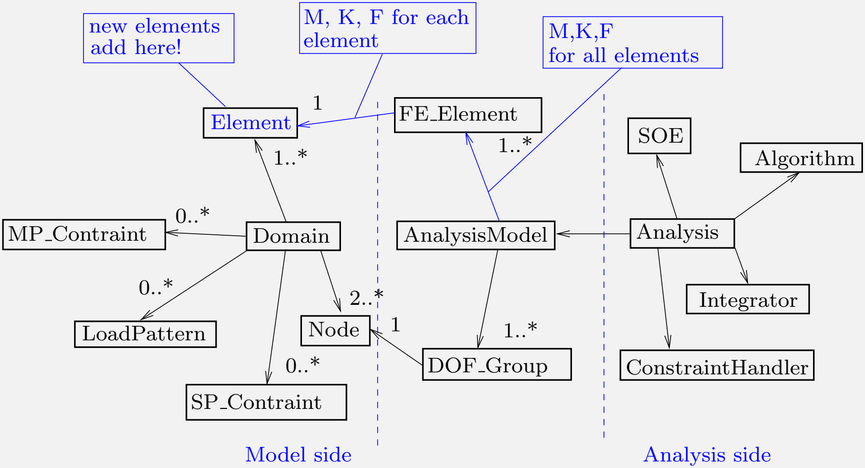
Flow of assembling local matrices to global



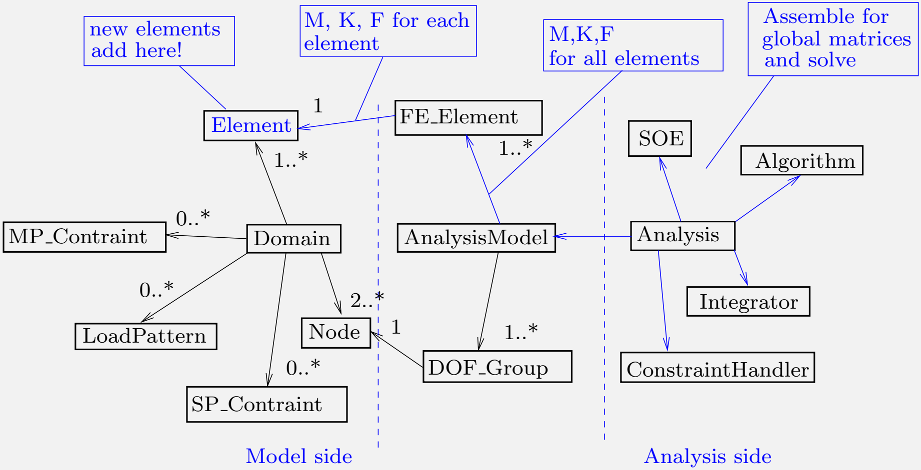
Flow of assembling local matrices to global



Flow of assembling local matrices to global



Flow of assembling local matrices to global



Problems of the implementation of FSM

Problems of the implementation of FSM

- FSM requires partitioned matrices since velocity and pressure are separated,

$$\mathbf{M} = \begin{bmatrix} \bar{\mathbf{M}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} \bar{\mathbf{K}} & -\mathbf{G} & \mathbf{0} \\ \mathbf{G}^T & \mathbf{L} & \mathbf{Q} \\ \mathbf{0} & \mathbf{Q}^T & \hat{\mathbf{M}} \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \bar{\mathbf{F}} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

Problems of the implementation of FSM

- FSM requires partitioned matrices since velocity and pressure are separated,

$$\mathbf{M} = \begin{bmatrix} \bar{\mathbf{M}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} \bar{\mathbf{K}} & -\mathbf{G} & \mathbf{0} \\ \mathbf{G}^T & \mathbf{L} & \mathbf{Q} \\ \mathbf{0} & \mathbf{Q}^T & \hat{\mathbf{M}} \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \bar{\mathbf{F}} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

- Matrices multiplications and inversions are applied on the partitioned matrices. Therefore partitioning of matrices has to be performed at the global level. Three inversions, $\bar{\mathbf{M}}^{-1}$, $\mathbf{S}^{-1}$ and $\hat{\mathbf{M}}^{-1}$ occur in a single iteration while normally there is only one in OpenSees.

Problems of the implementation of FSM

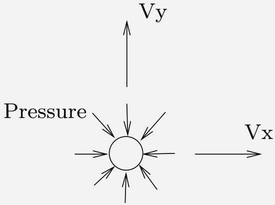
- FSM requires partitioned matrices since velocity and pressure are separated,

$$\mathbf{M} = \begin{bmatrix} \bar{\mathbf{M}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} \bar{\mathbf{K}} & -\mathbf{G} & \mathbf{0} \\ \mathbf{G}^T & \mathbf{L} & \mathbf{Q} \\ \mathbf{0} & \mathbf{Q}^T & \hat{\mathbf{M}} \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \bar{\mathbf{F}} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

- Matrices multiplications and inversions are applied on the partitioned matrices. Therefore partitioning of matrices has to be performed at the global level. Three inversions, $\bar{\mathbf{M}}^{-1}$, $\mathbf{S}^{-1}$ and $\hat{\mathbf{M}}^{-1}$ occur in a single iteration while normally there is only one in OpenSees.
- From the flow chart shown before, global matrices are only available in the analysis side. So we can only partition matrices in the analysis side.

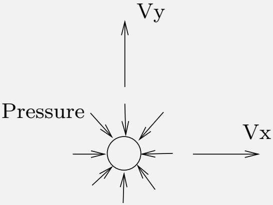
Problems of the implementation of FSM

- FSM is strongly related to the model. It assumes each node has 3 DOFs, vx, vy and pressure. Any implementation of FSM is coupled to the model information.



Problems of the implementation of FSM

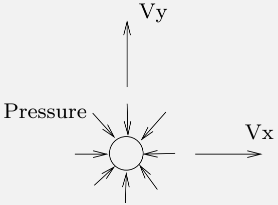
- FSM is strongly related to the model. It assumes each node has 3 DOFs, vx, vy and pressure. Any implementation of FSM is coupled to the model information.



- The philosophy of OpenSees requires model and analysis uncoupled.

Problems of the implementation of FSM

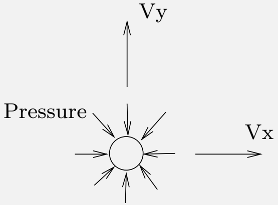
- FSM is strongly related to the model. It assumes each node has 3 DOFs, vx, vy and pressure. Any implementation of FSM is coupled to the model information.



- The philosophy of OpenSees requires model and analysis uncoupled.
- This requires a model side implementation.

Problems of the implementation of FSM

- FSM is strongly related to the model. It assumes each node has 3 DOFs, vx, vy and pressure. Any implementation of FSM is coupled to the model information.



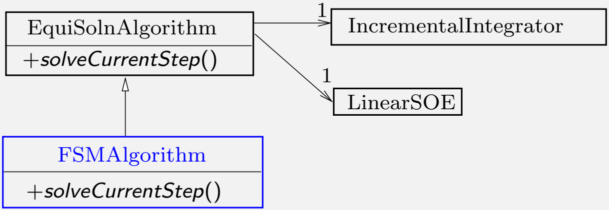
- The philosophy of OpenSees requires model and analysis uncoupled.
- This requires a model side implementation.
- Model side? Analysis side?

Option 1 : add a new algorithm

The first option is to implement FSM in a new algorithm `FSMAlgorithm`,

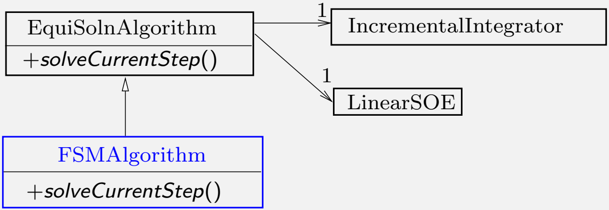
Option 1 : add a new algorithm

The first option is to implement FSM in a new algorithm `FSMAlgorithm`,



Option 1 : add a new algorithm

The first option is to implement FSM in a new algorithm `FSMAlgorithm`,



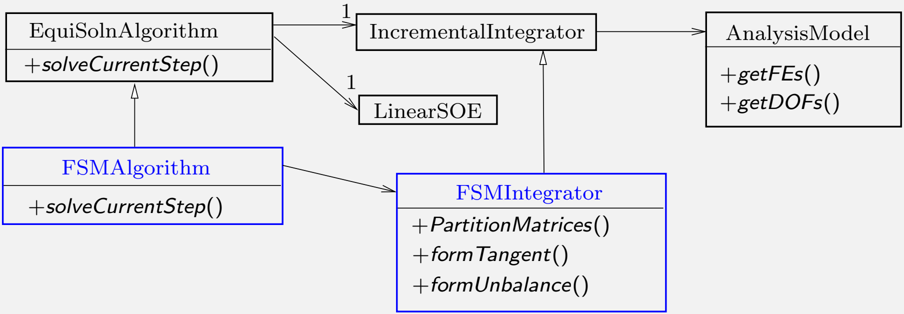
We want use `FSMAlgorithm` to partition matrices in the global level. But `FSMAlgorithm` does not have the DOFs information. To get the information, another new class `PFEMIntegrator` has to be added also.

Option 2 : add a new algorithm and integrator

The second option is to create both an algorithm and integrator,

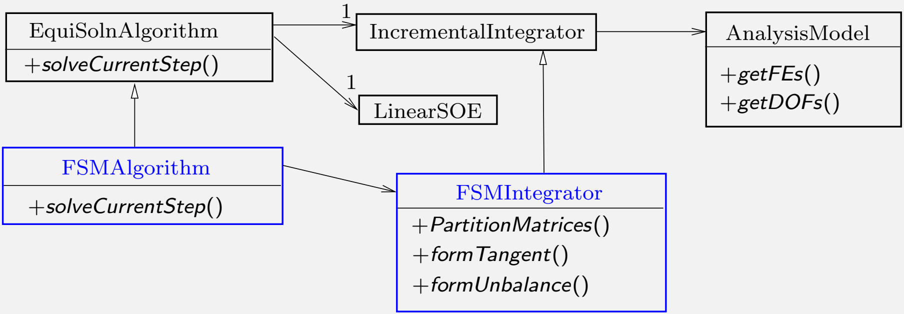
Option 2 : add a new algorithm and integrator

The second option is to create both an algorithm and integrator,



Option 2 : add a new algorithm and integrator

The second option is to create both an algorithm and integrator,



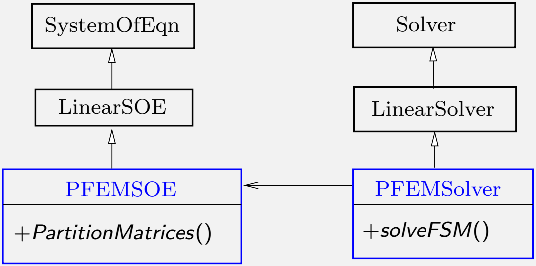
However, this implementation makes the algorithm and integrator too specialized for FSM and needs new methods in the interface of the FSM algorithm and integrator. The FSM algorithm and integrator are coupled to the model information.

Option 3: Add a SOE and solver

The third option is to add a `LinearSOE` and `Solver`,

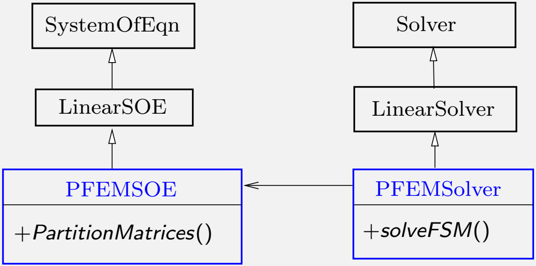
Option 3: Add a SOE and solver

The third option is to add a `LinearSOE` and `Solver`,



Option 3: Add a SOE and solver

The third option is to add a `LinearSOE` and `Solver`,



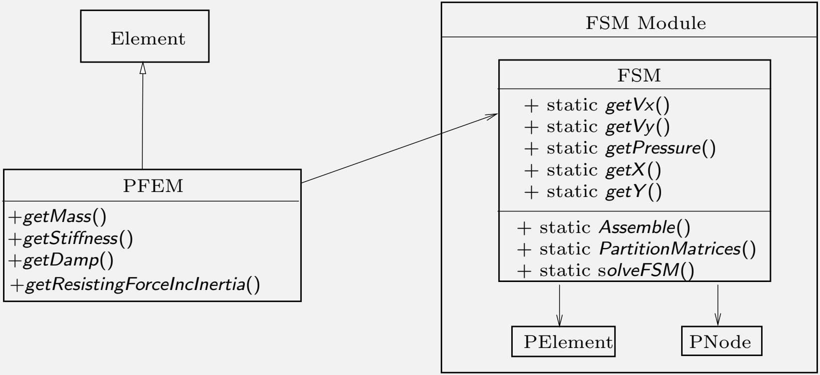
This option has the same problem with option 1 that neither `LinearSOE` nor `Solver` knows the DOFs information. They cannot partition without knowing IDs and only work with each other.

A model side solution: `Adapter`

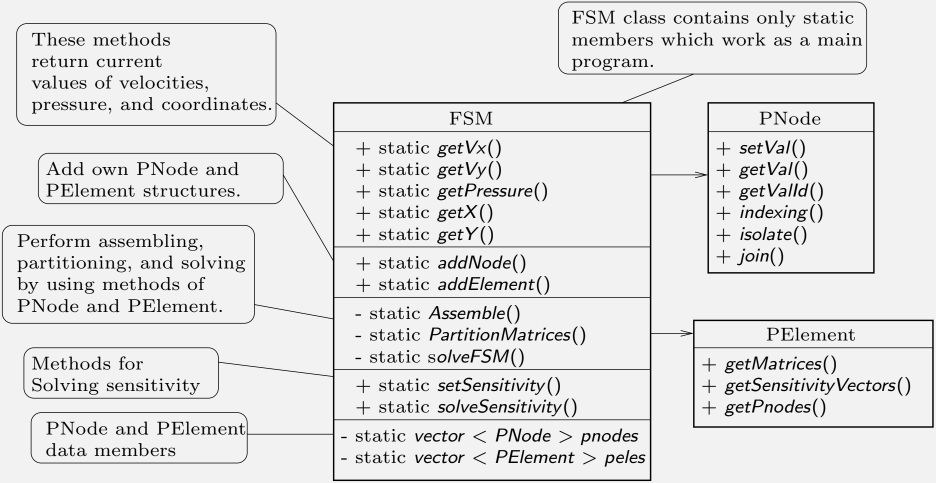
The solution is a model side implementation, not an analysis side implementation.

A model side solution: Adapter

The solution is a model side implementation, not an analysis side implementation.



A model side solution: `Adapter`



A model side solution: `Adapter`

- The independent FSM module is indeed a small FE program that can complete the job all by itself and returns results to `PFEM` elements.

A model side solution: Adapter

- The independent FSM module is indeed a small FE program that can complete the job all by itself and returns results to PFEM elements.
- The FSM module contains FSM, PNode and PElement classes.

A model side solution: Adapter

- The independent FSM module is indeed a small FE program that can complete the job all by itself and returns results to PFEM elements.
- The FSM module contains FSM, PNode and PElement classes.
- Consistent with published PFEM implementations for FSI.

A model side solution: `Adapter`

- The independent FSM module is indeed a small FE program that can complete the job all by itself and returns results to `PFEM` elements.
- The FSM module contains `FSM`, `PNode` and `PElement` classes.
- Consistent with published PFEM implementations for FSI.
- `PFEM` works as a wrapper to ask `FSM` for information needed.

A model side solution: `Adapter`

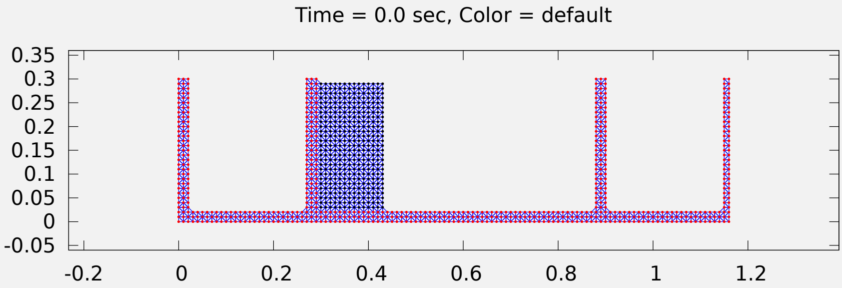
- The independent FSM module is indeed a small FE program that can complete the job all by itself and returns results to `PFEM` elements.
- The FSM module contains `FSM`, `PNode` and `PElement` classes.
- Consistent with published PFEM implementations for FSI.
- `PFEM` works as a wrapper to ask `FSM` for information needed.
- Pressures are hidden from OpenSees. `PFEM` elements can be treated uniformly as solid elements in OpenSees.

A model side solution: `Adapter`

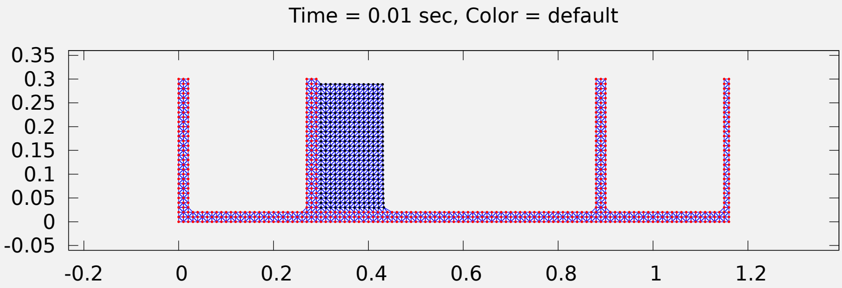
- The independent FSM module is indeed a small FE program that can complete the job all by itself and returns results to `PFEM` elements.
- The FSM module contains `FSM`, `PNode` and `PElement` classes.
- Consistent with published PFEM implementations for FSI.
- `PFEM` works as a wrapper to ask `FSM` for information needed.
- Pressures are hidden from OpenSees. `PFEM` elements can be treated uniformly as solid elements in OpenSees.
- `PNode` and `PElement` are standalone structures used by `FSM` for storing its nodes and elements

Water column collapse

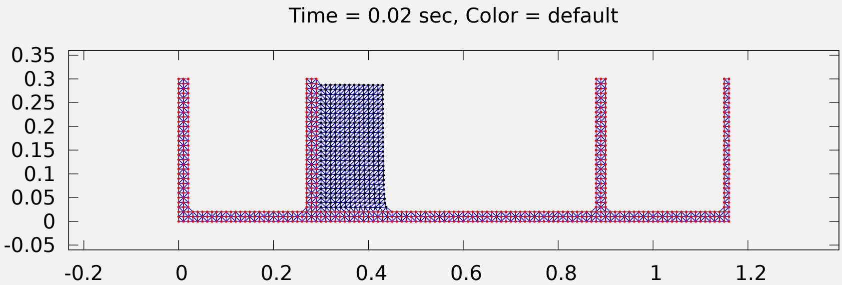
- A standard test problem of PFEM, involving large domain changing and strong nonlinearity.
- A water column is kept in a container by an unseen vertical board. Start by removing the board.



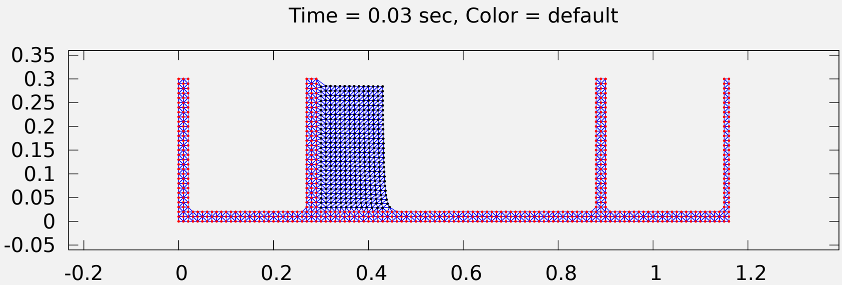
Water column collapse



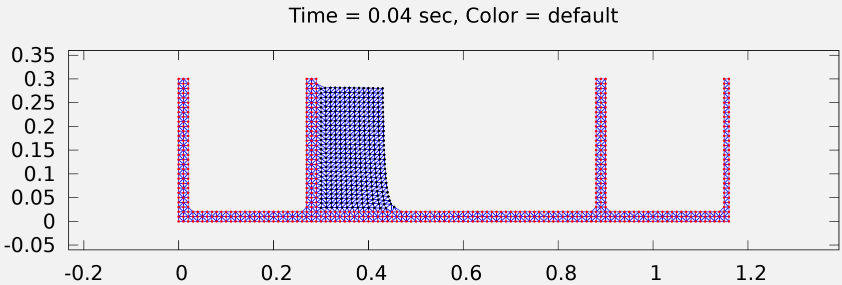
Water column collapse



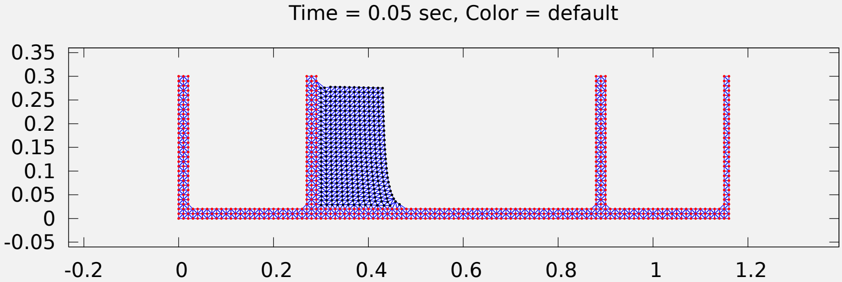
Water column collapse



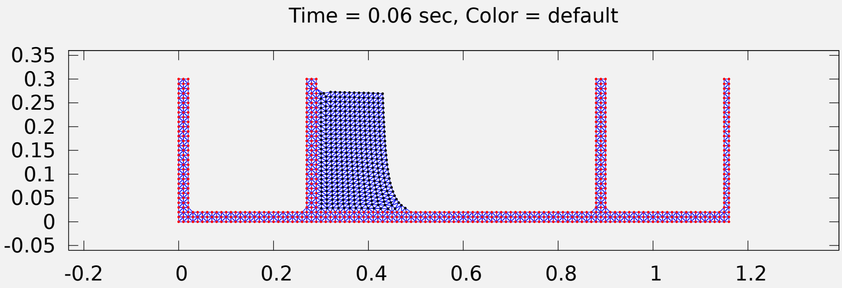
Water column collapse



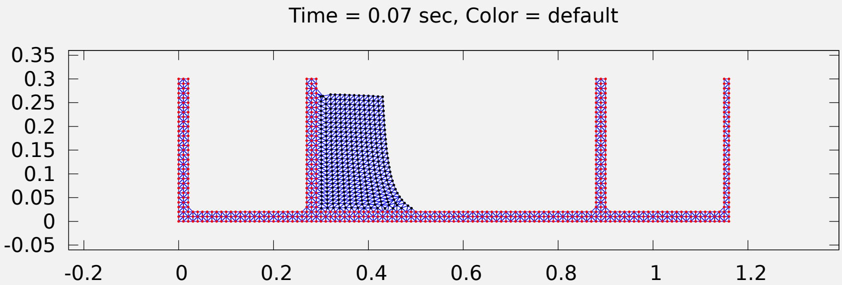
Water column collapse



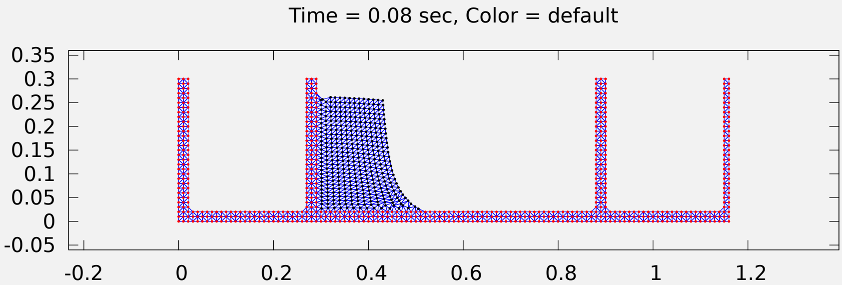
Water column collapse



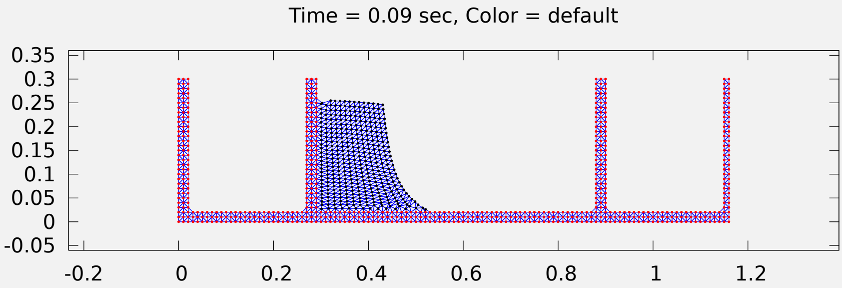
Water column collapse



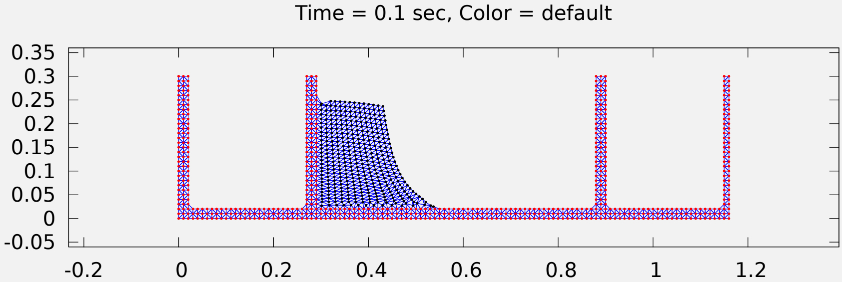
Water column collapse



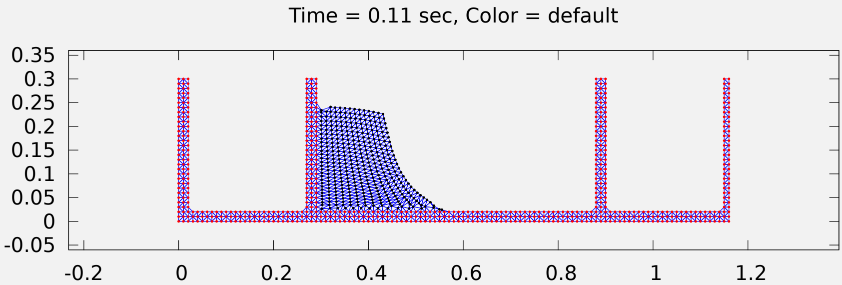
Water column collapse



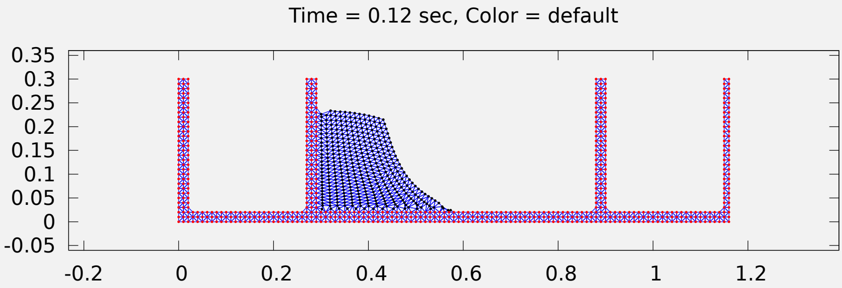
Water column collapse



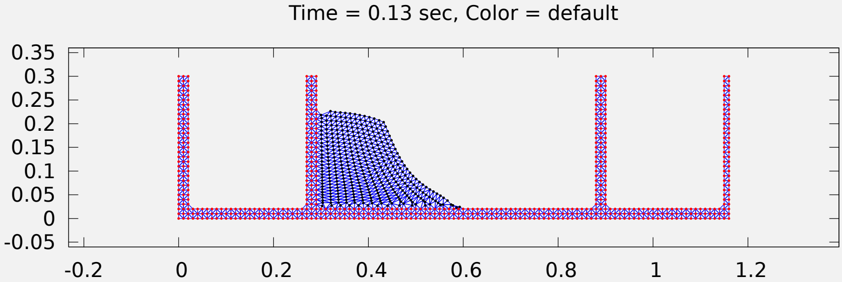
Water column collapse



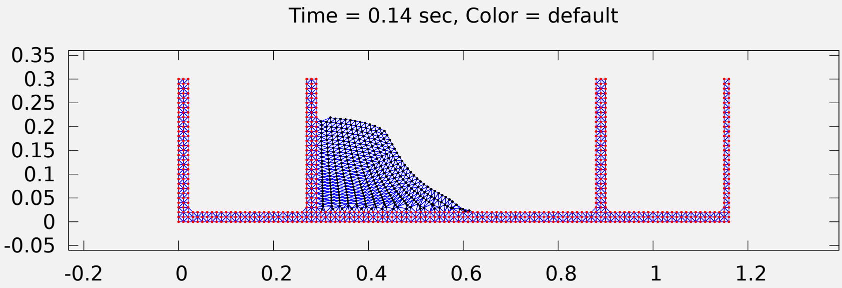
Water column collapse



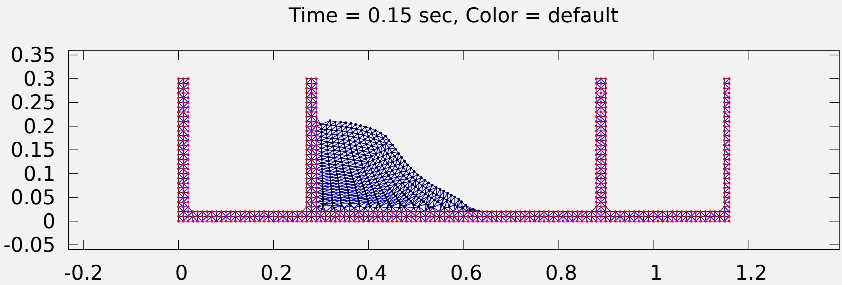
Water column collapse



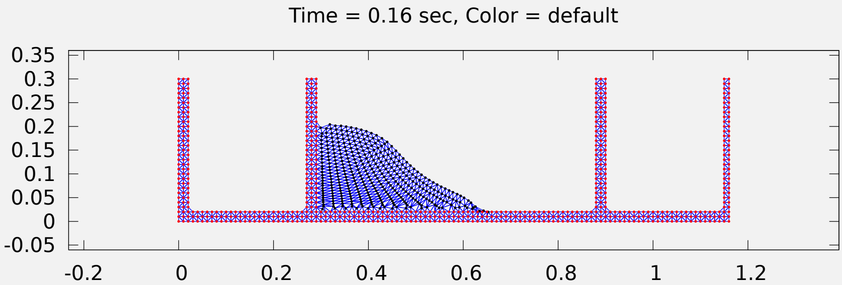
Water column collapse



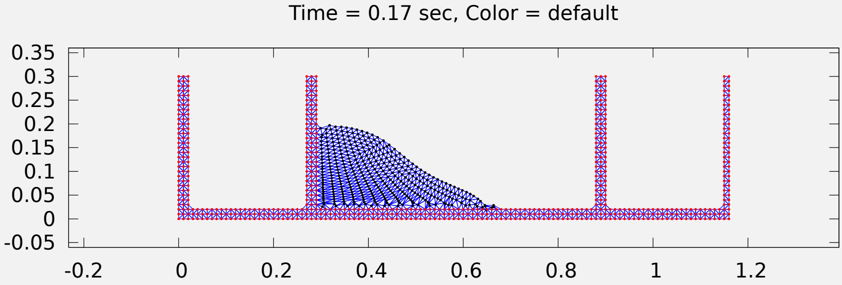
Water column collapse



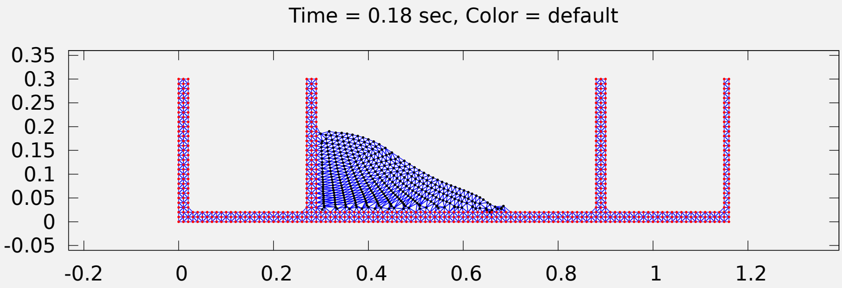
Water column collapse



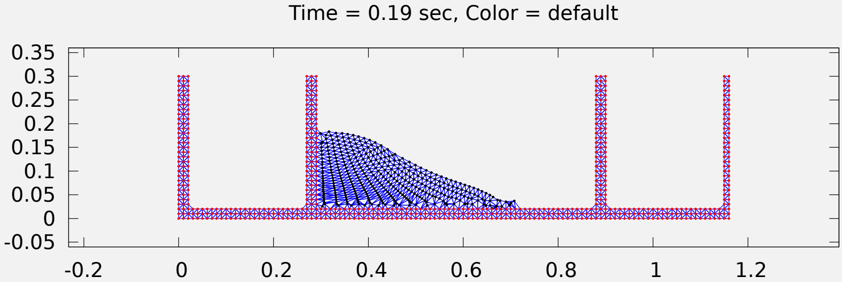
Water column collapse



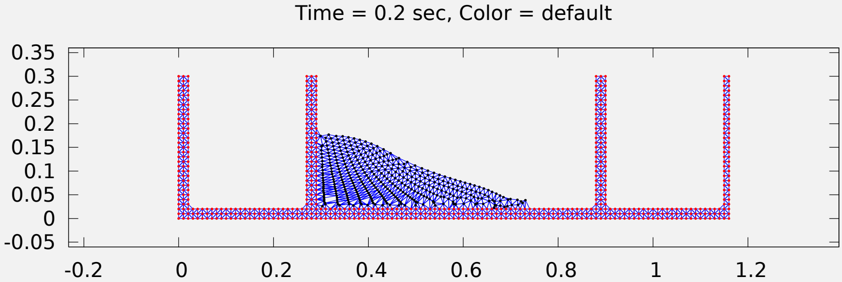
Water column collapse



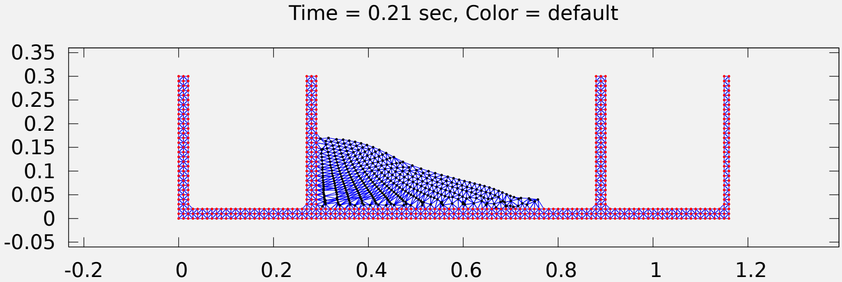
Water column collapse



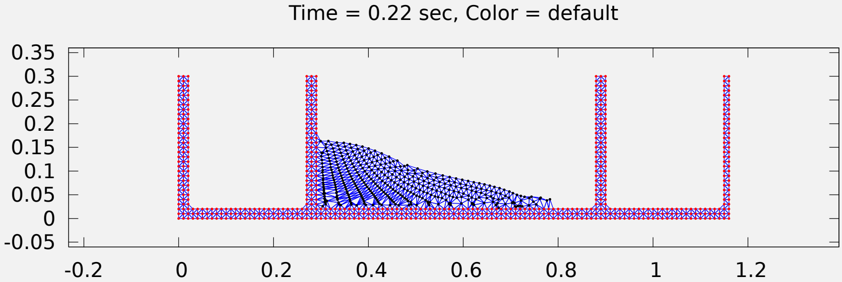
Water column collapse



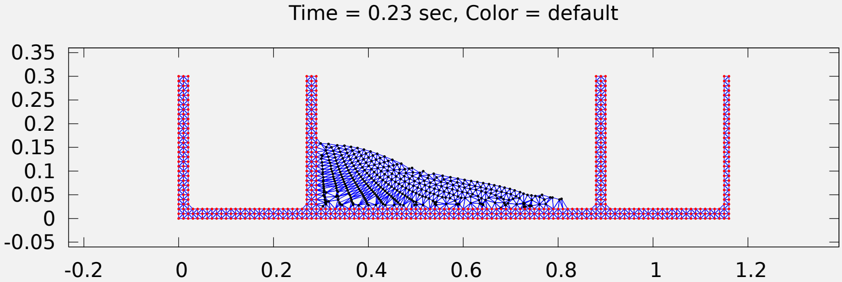
Water column collapse



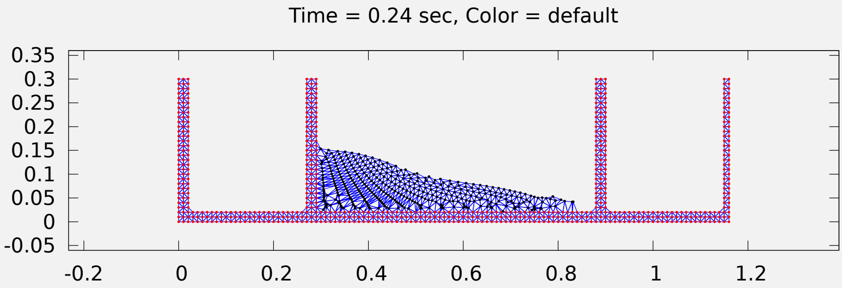
Water column collapse



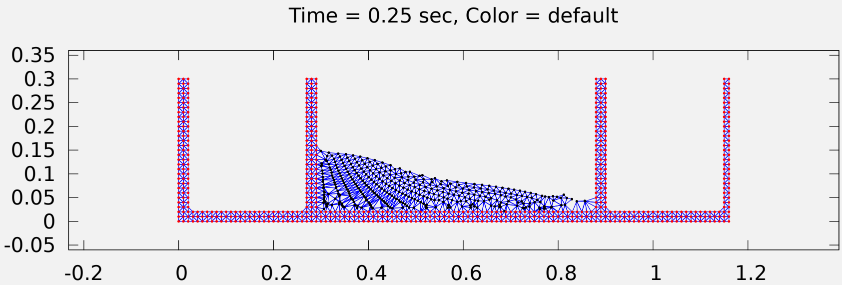
Water column collapse



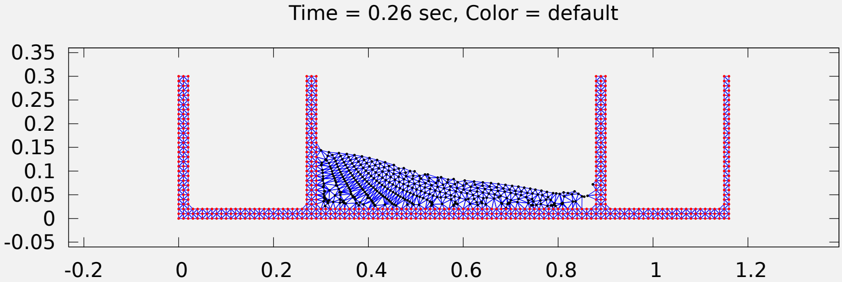
Water column collapse



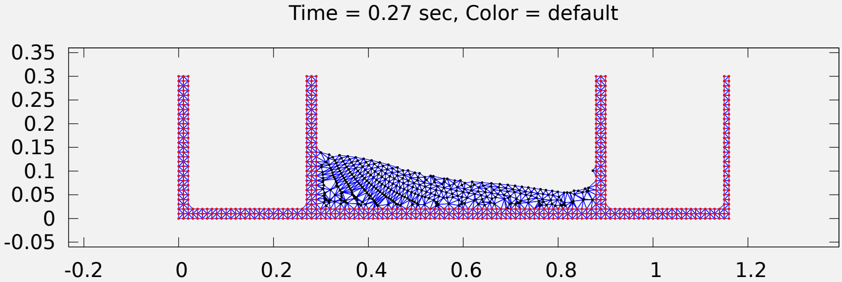
Water column collapse



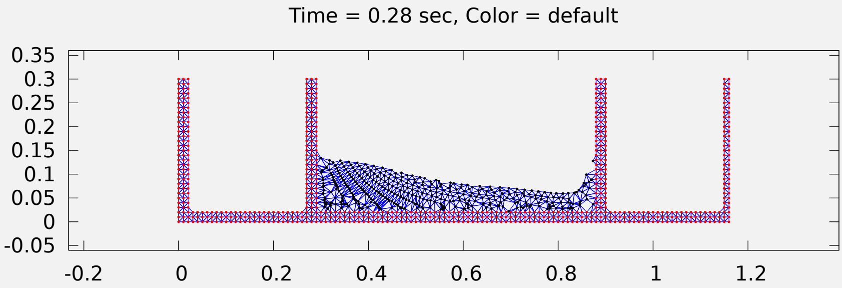
Water column collapse



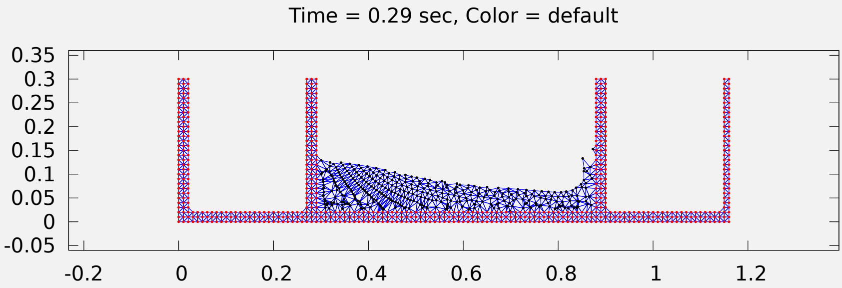
Water column collapse



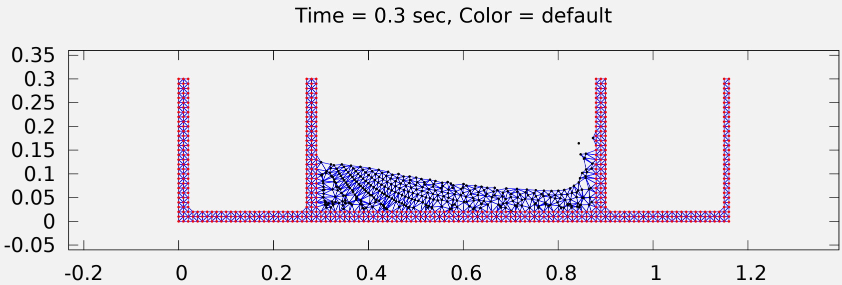
Water column collapse



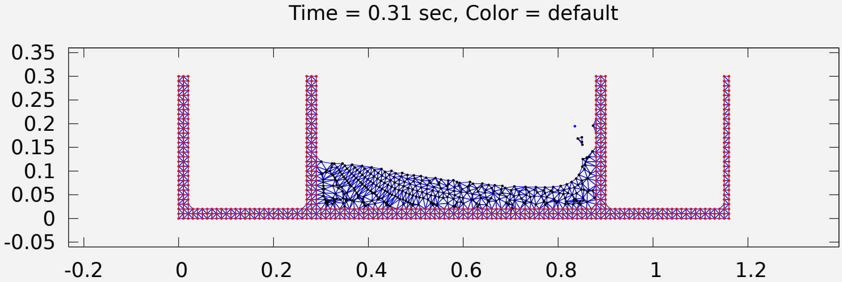
Water column collapse



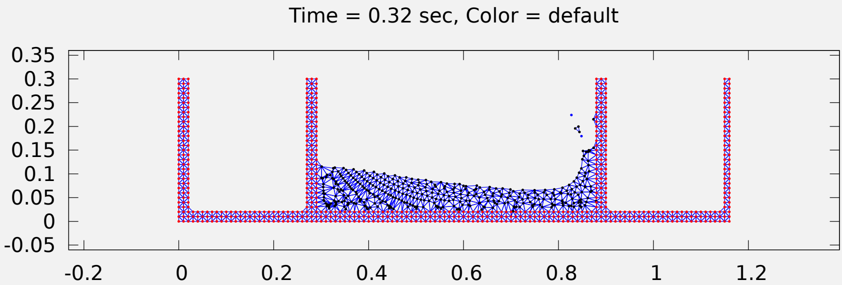
Water column collapse



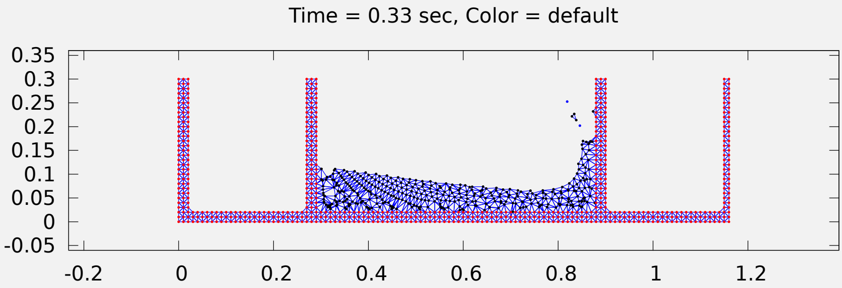
Water column collapse



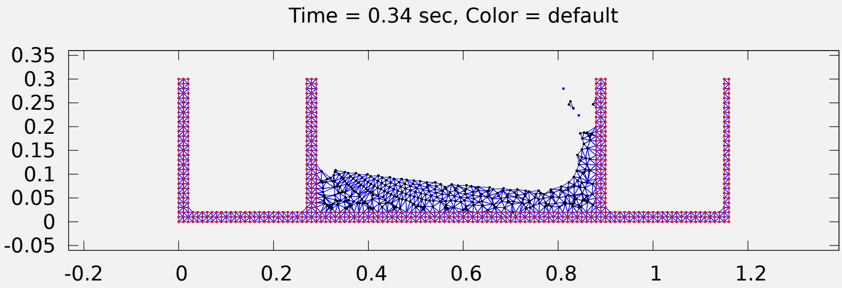
Water column collapse



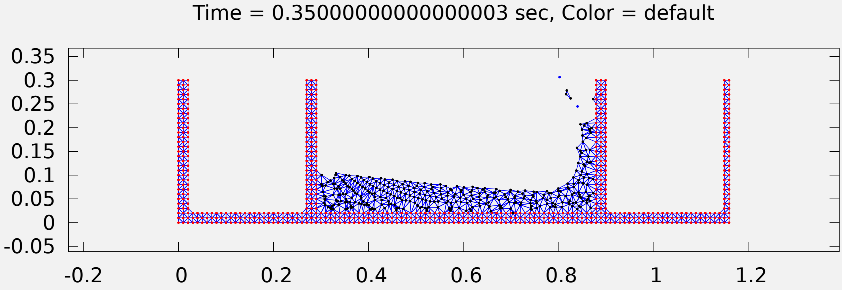
Water column collapse



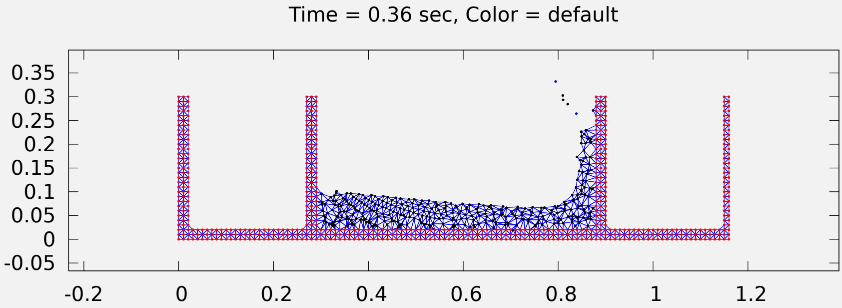
Water column collapse



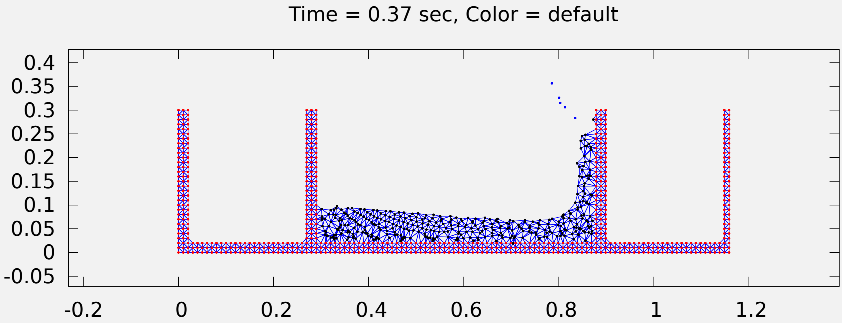
Water column collapse



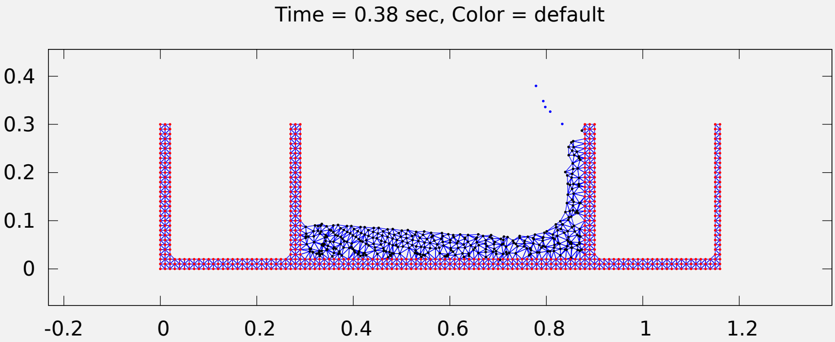
Water column collapse



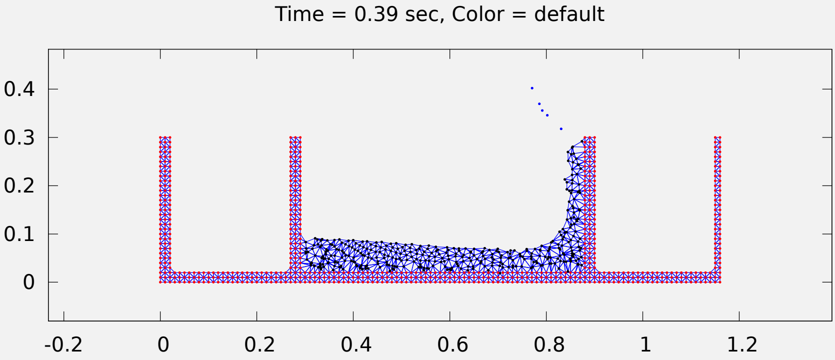
Water column collapse



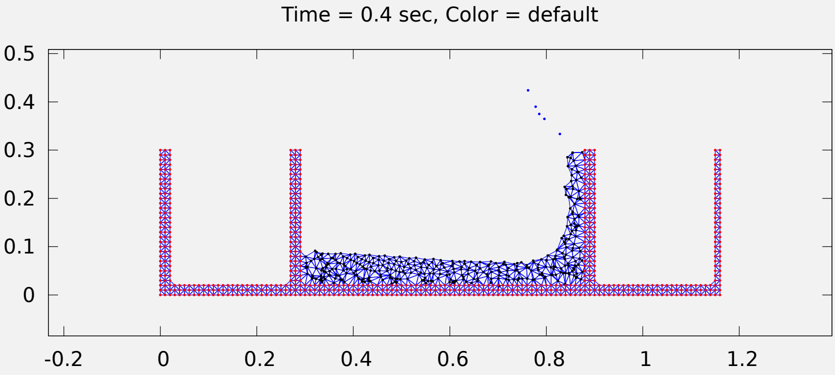
Water column collapse



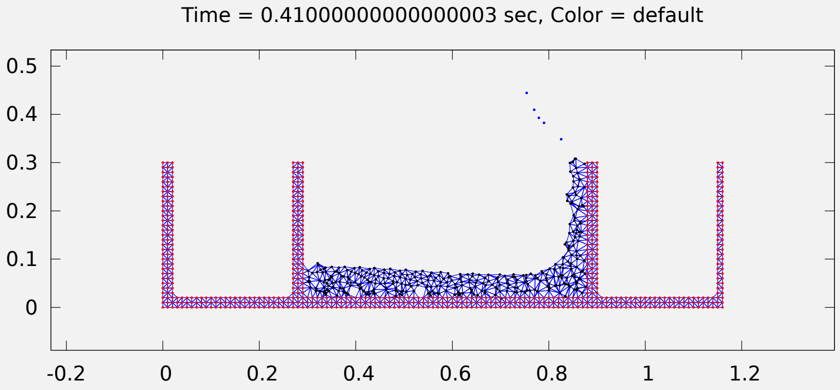
Water column collapse



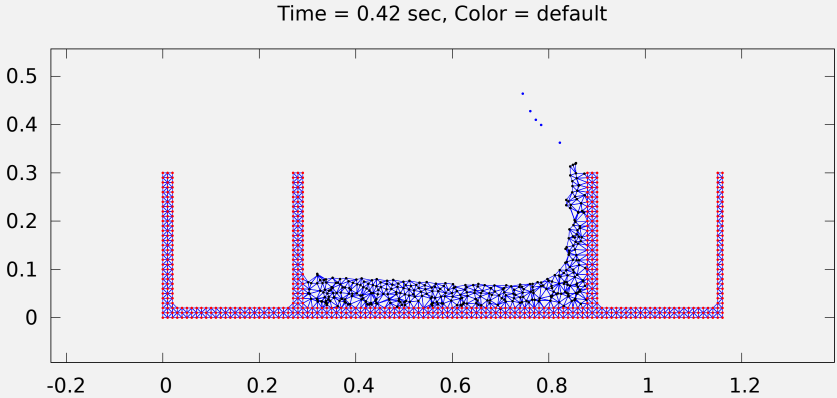
Water column collapse



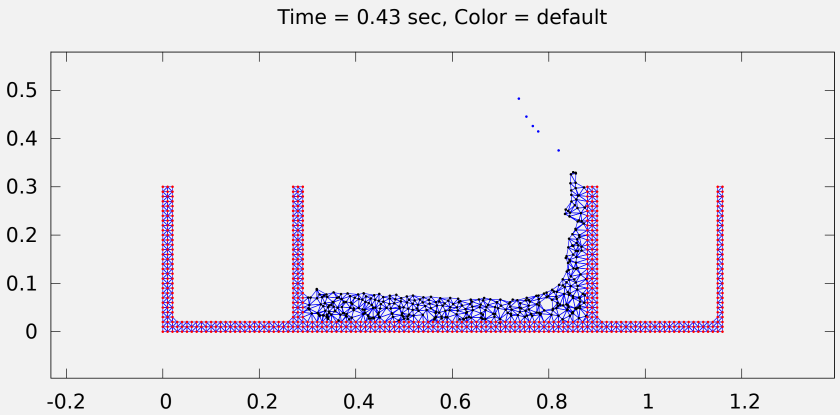
Water column collapse



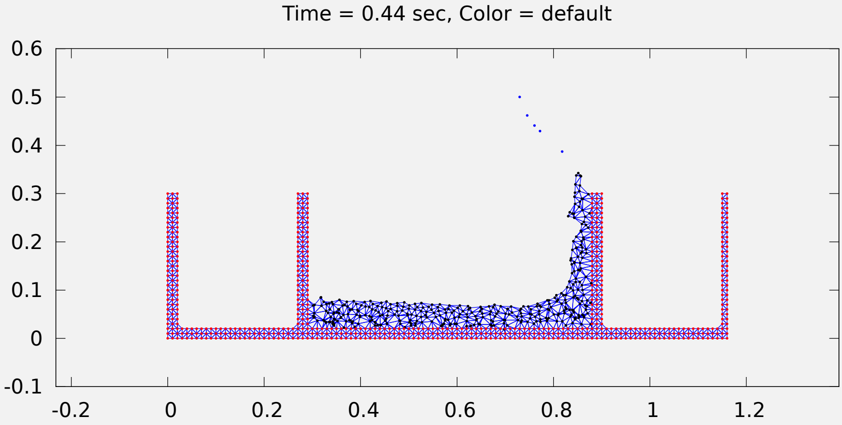
Water column collapse



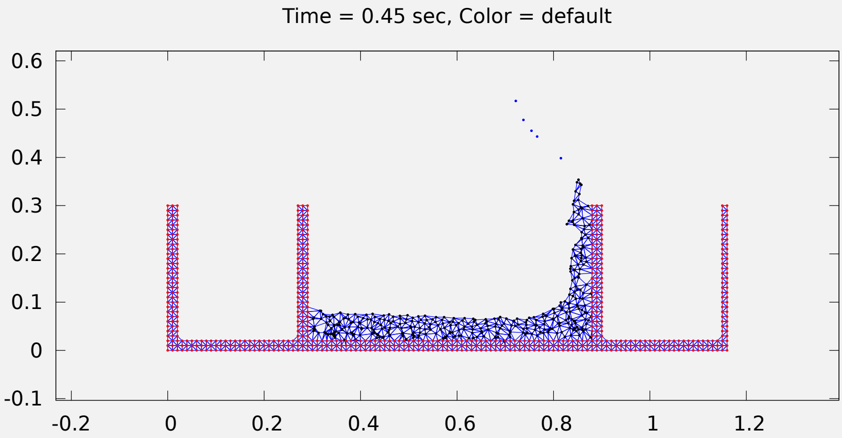
Water column collapse



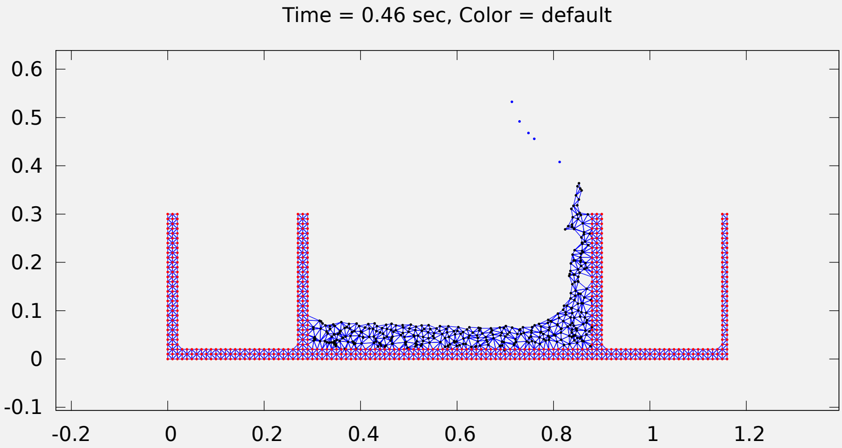
Water column collapse



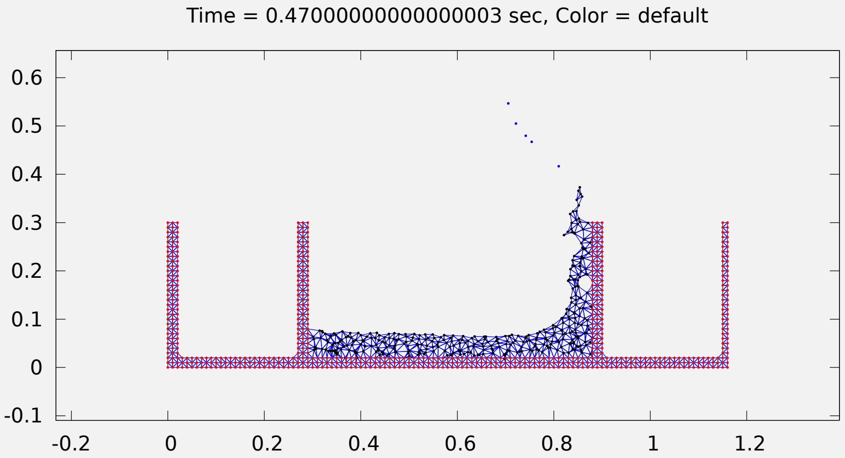
Water column collapse



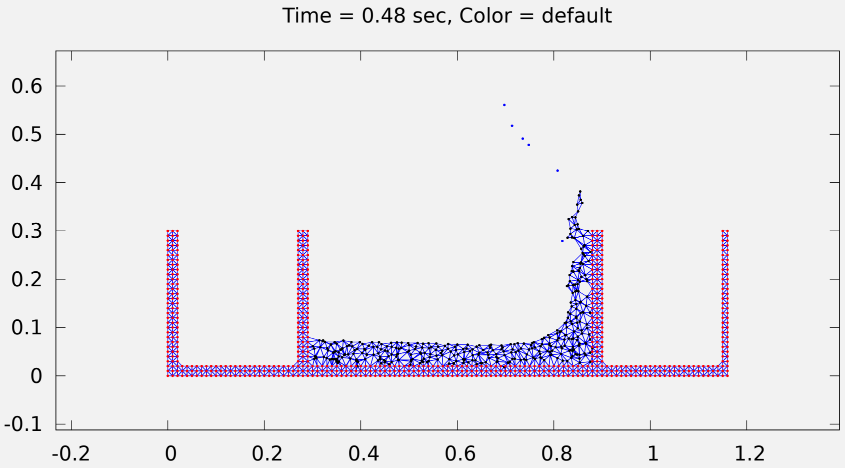
Water column collapse



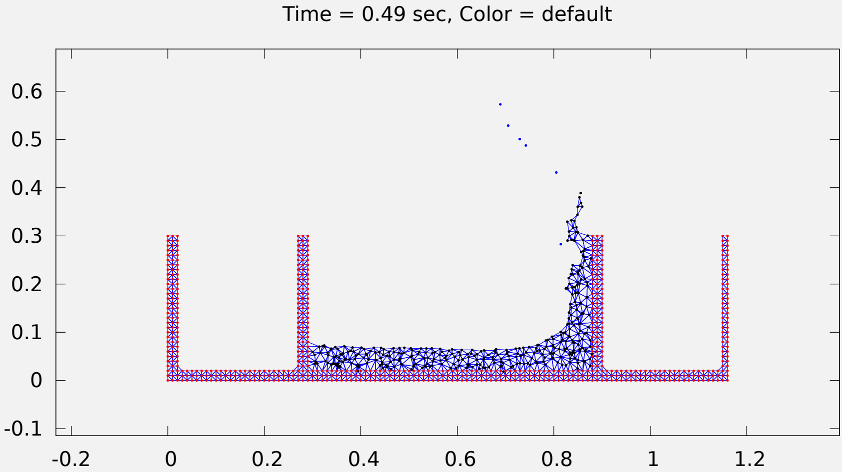
Water column collapse



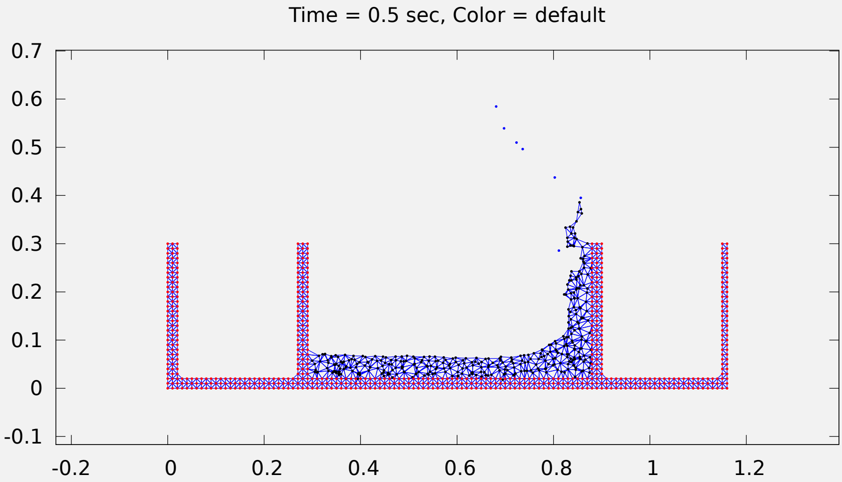
Water column collapse



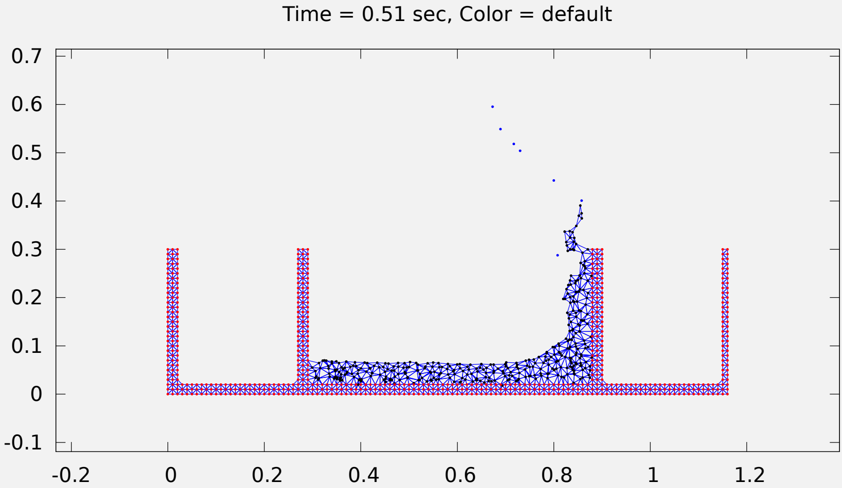
Water column collapse



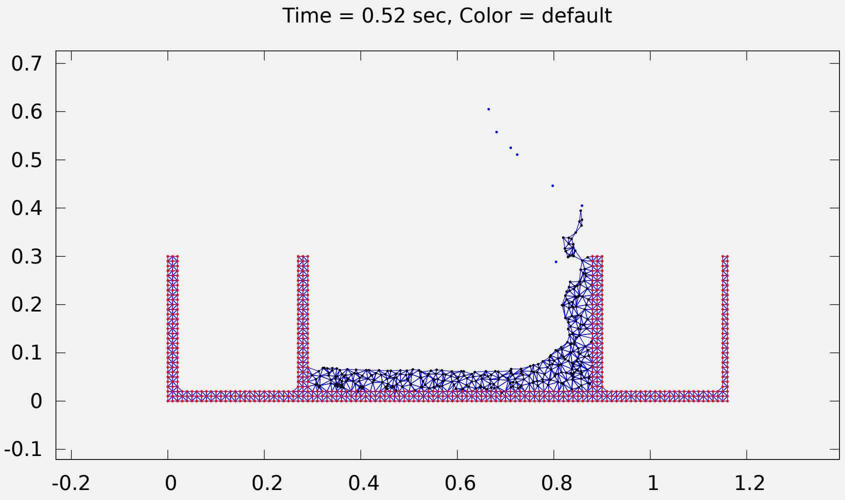
Water column collapse



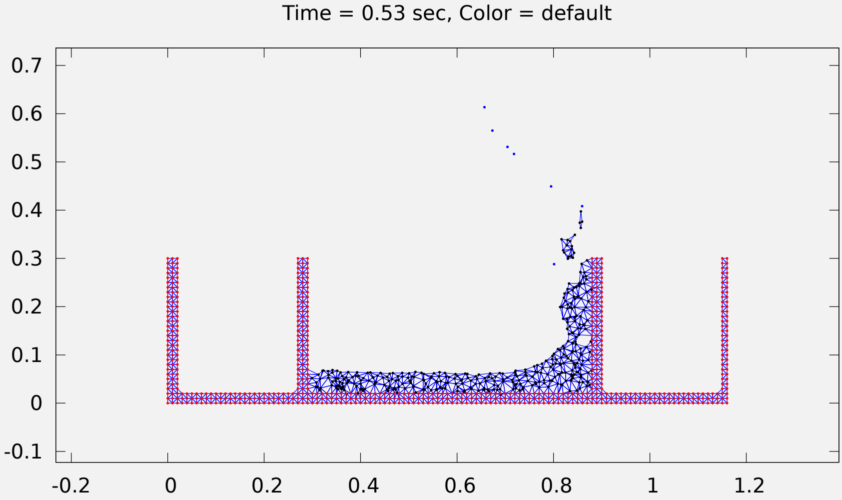
Water column collapse



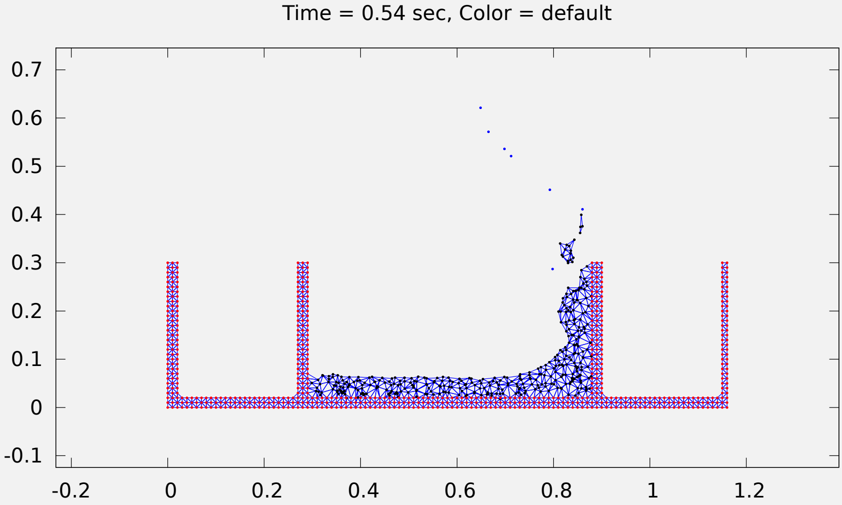
Water column collapse



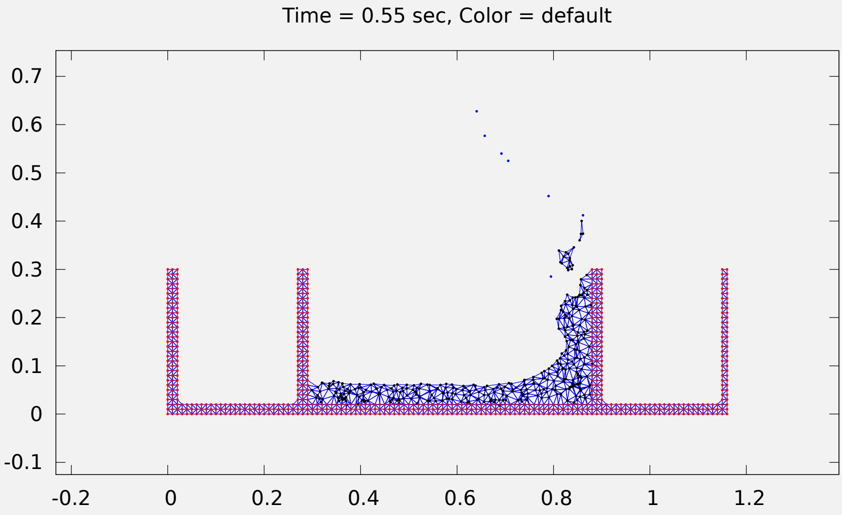
Water column collapse



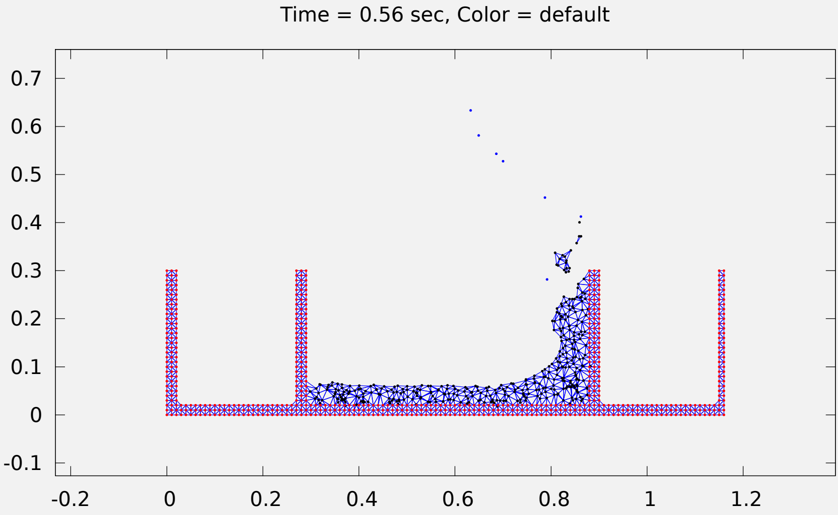
Water column collapse



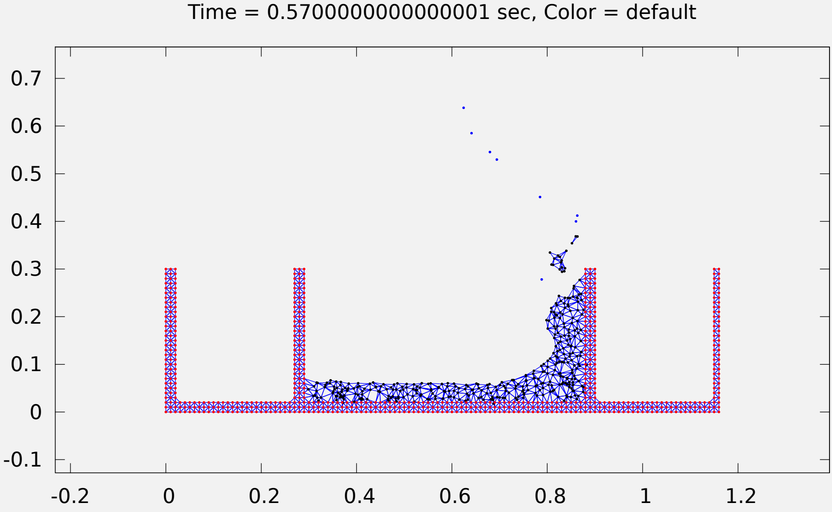
Water column collapse



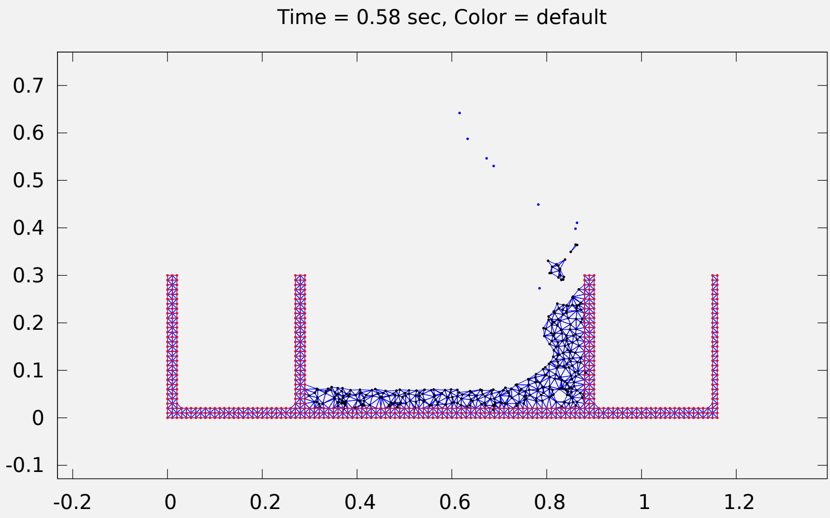
Water column collapse



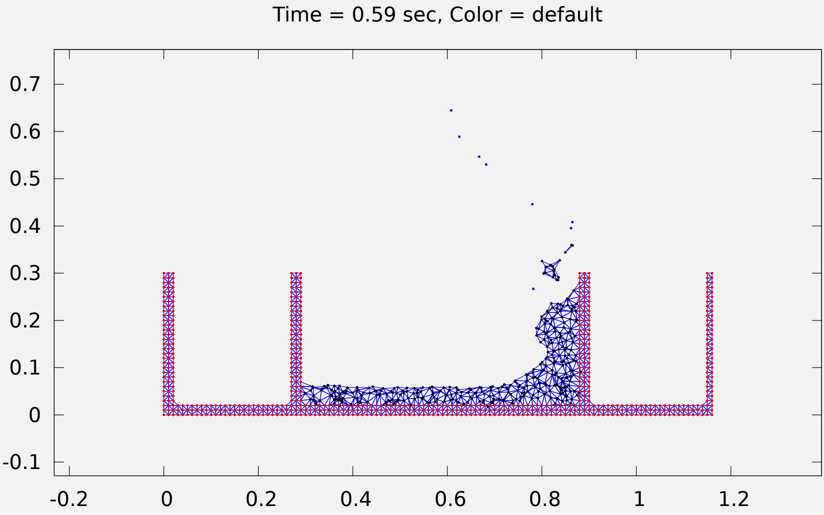
Water column collapse



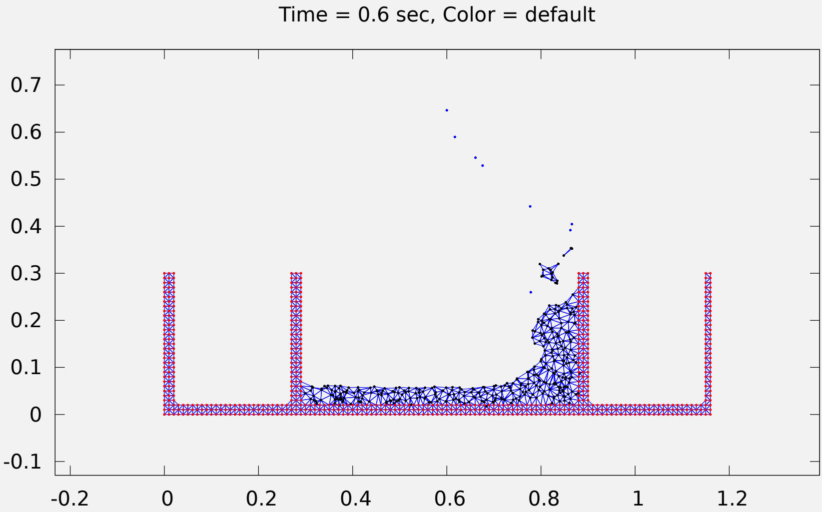
Water column collapse



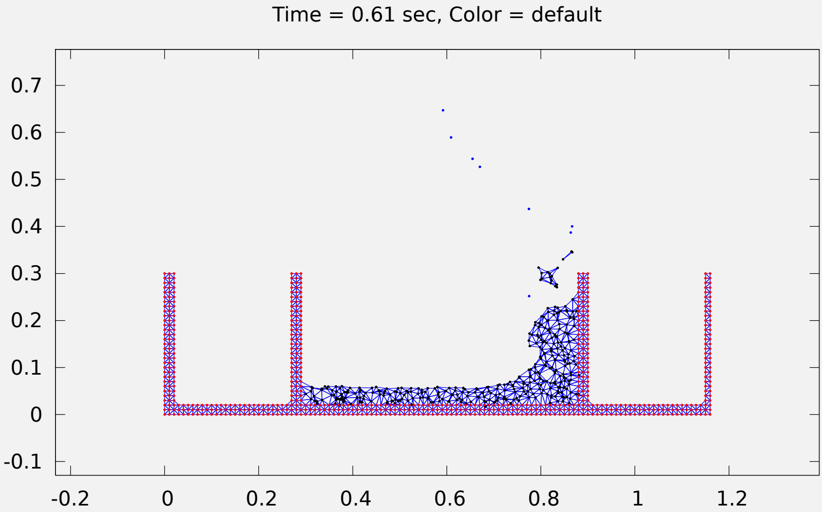
Water column collapse



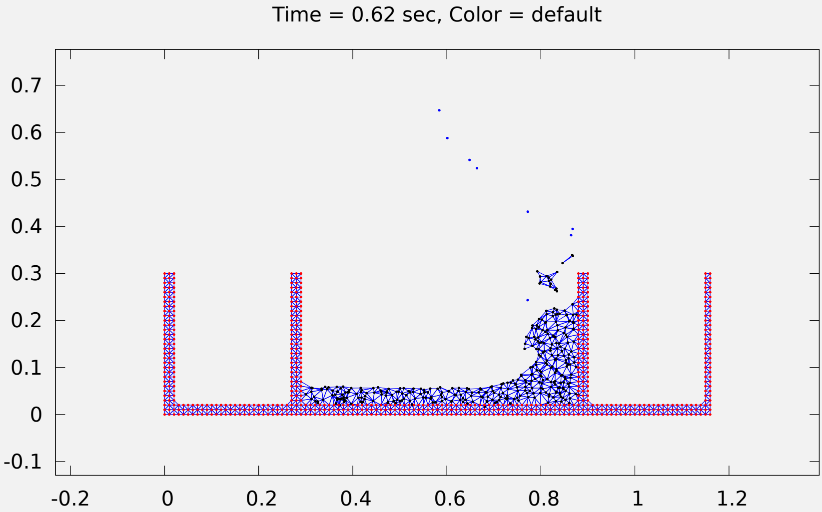
Water column collapse



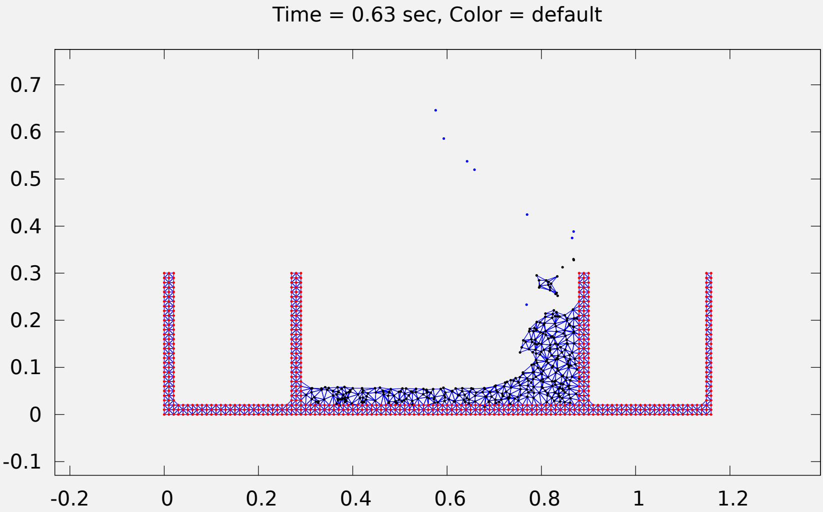
Water column collapse



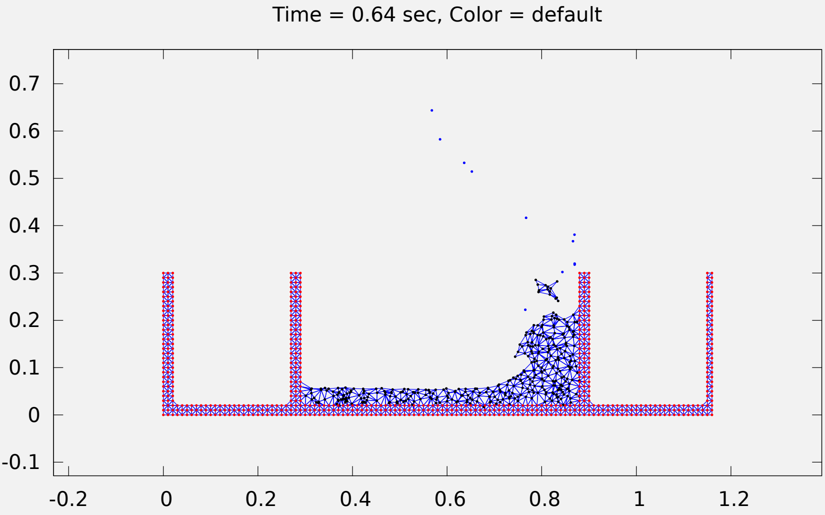
Water column collapse



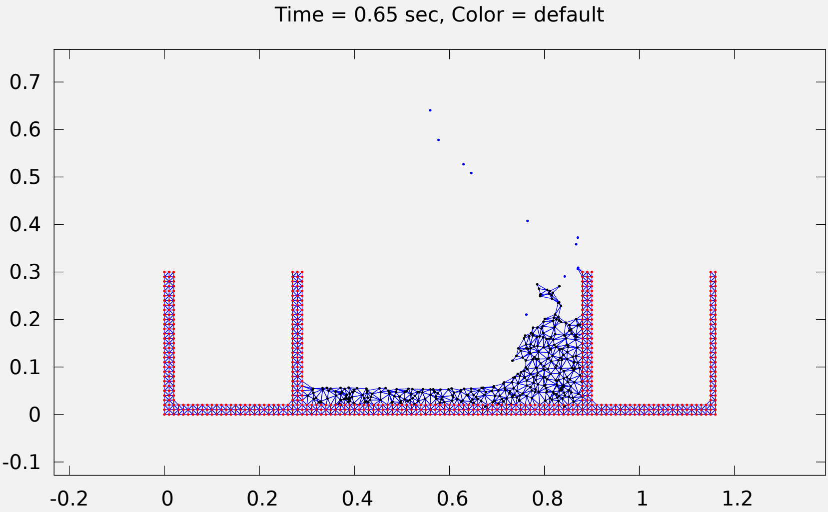
Water column collapse



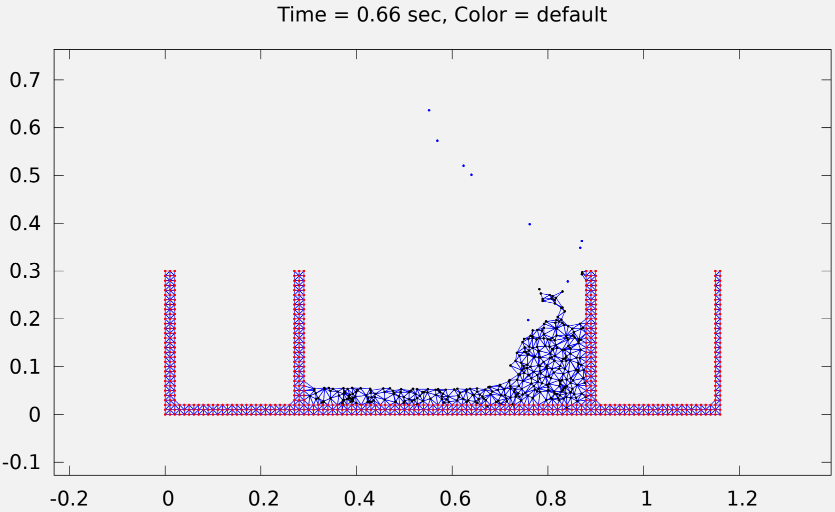
Water column collapse



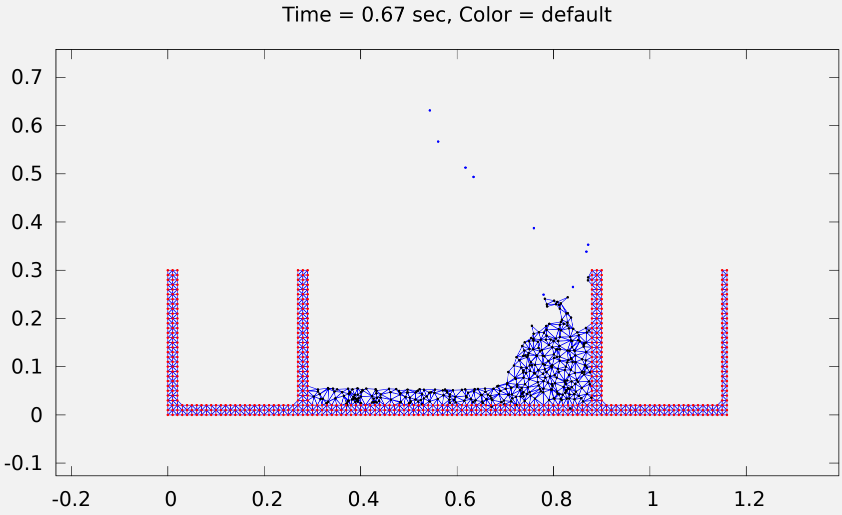
Water column collapse



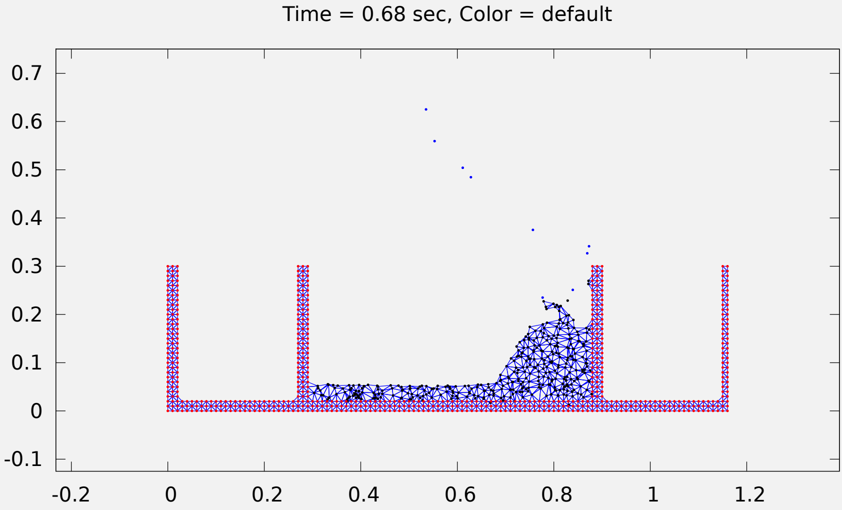
Water column collapse



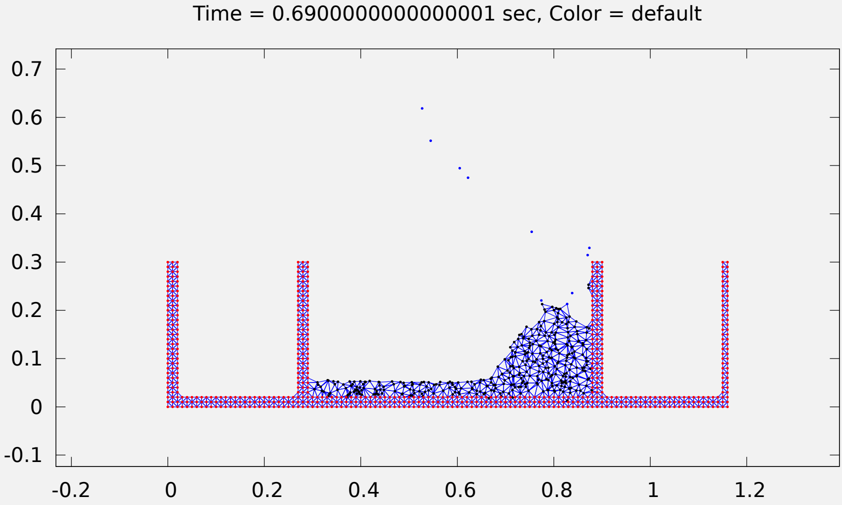
Water column collapse



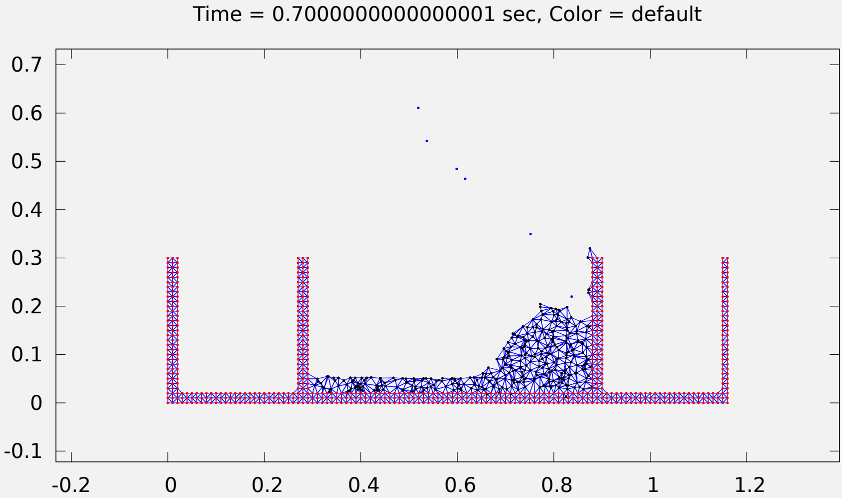
Water column collapse



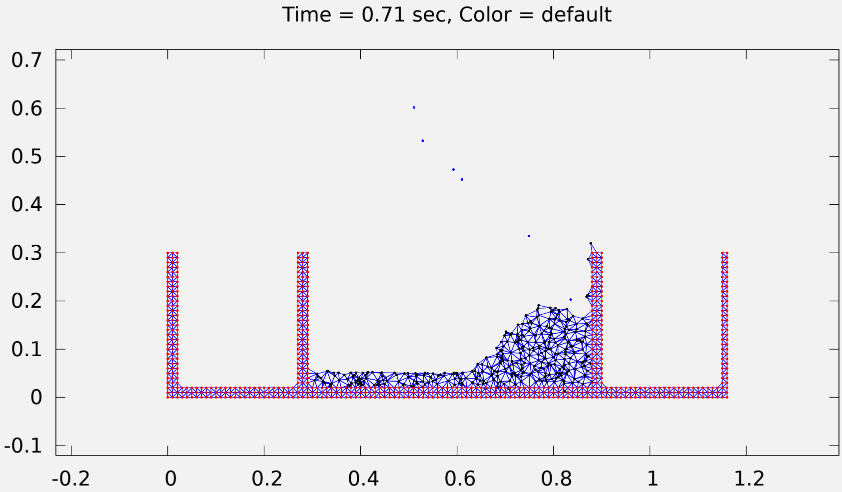
Water column collapse



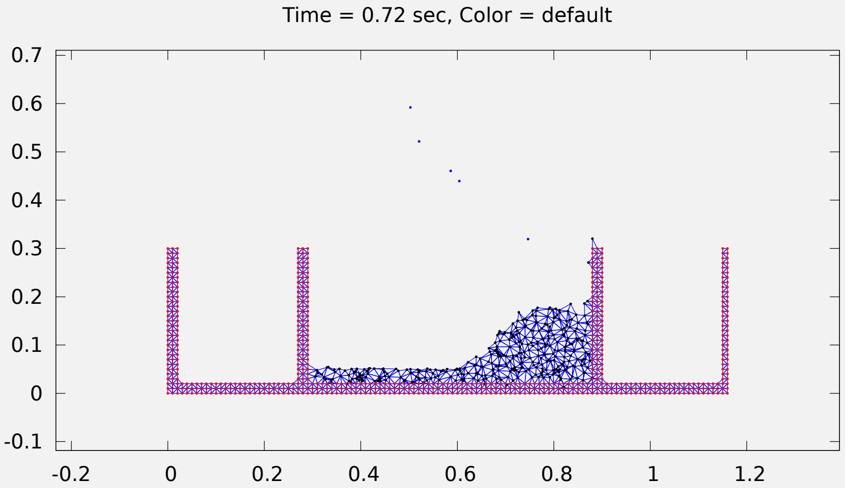
Water column collapse



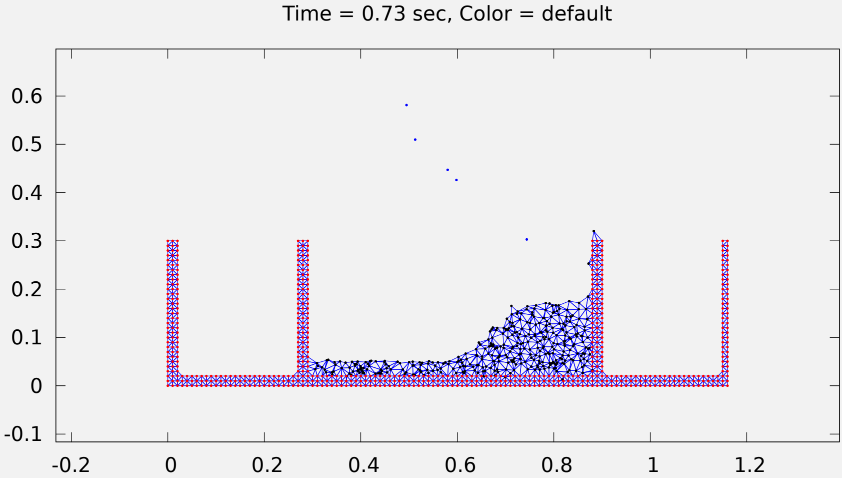
Water column collapse



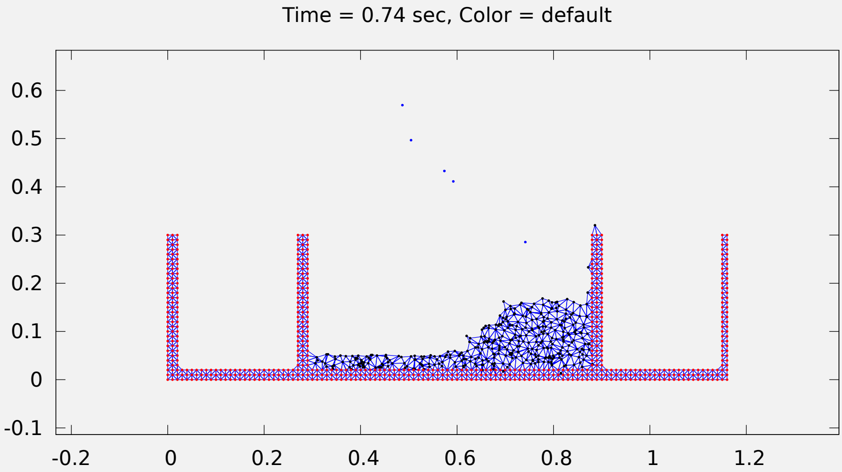
Water column collapse



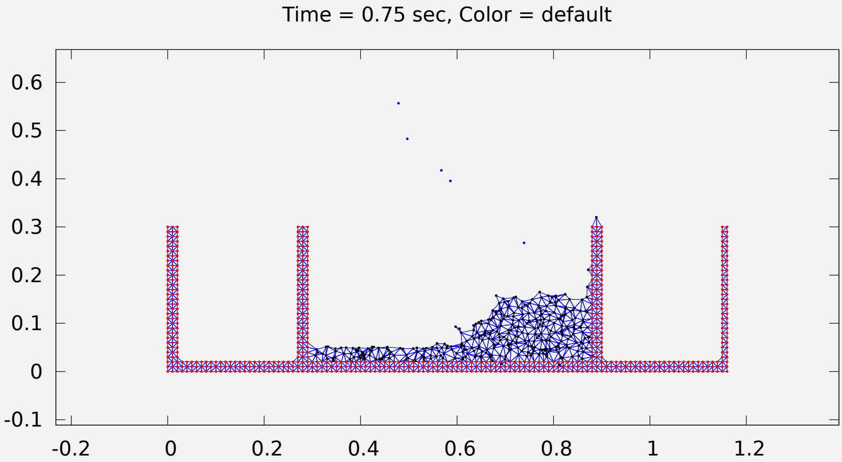
Water column collapse



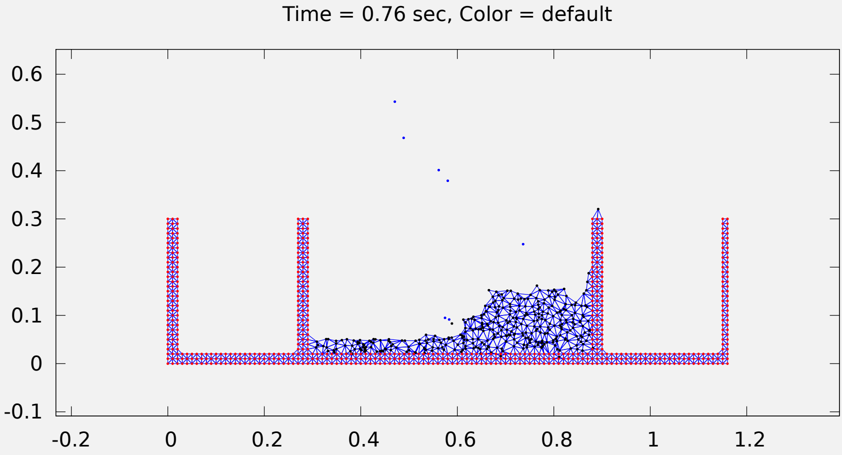
Water column collapse



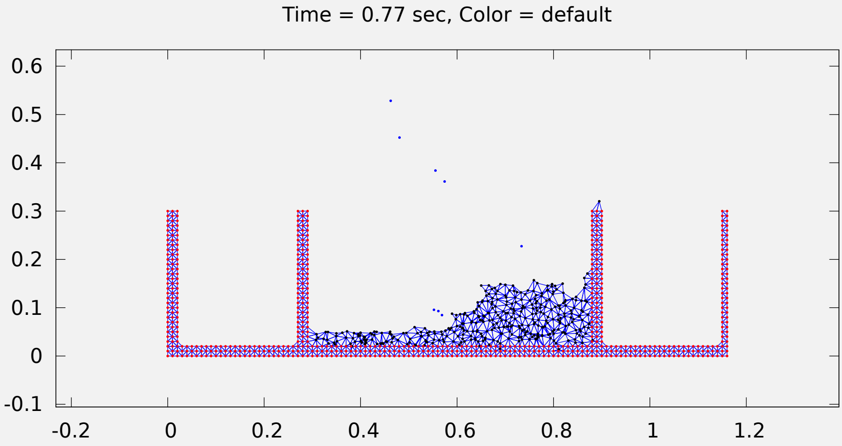
Water column collapse



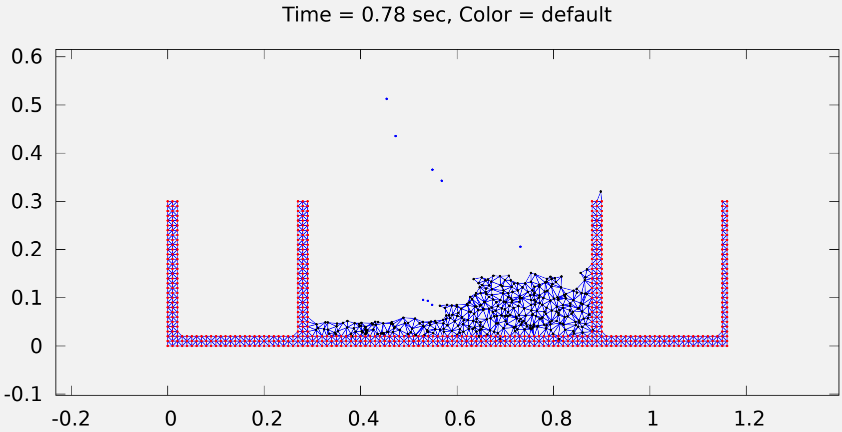
Water column collapse



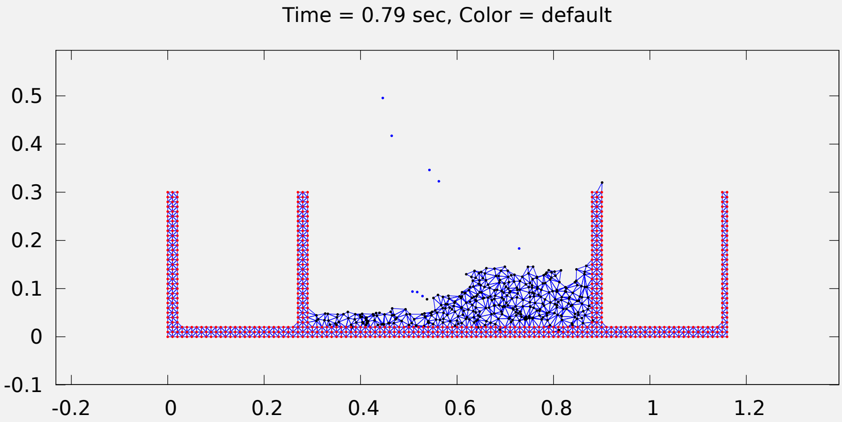
Water column collapse



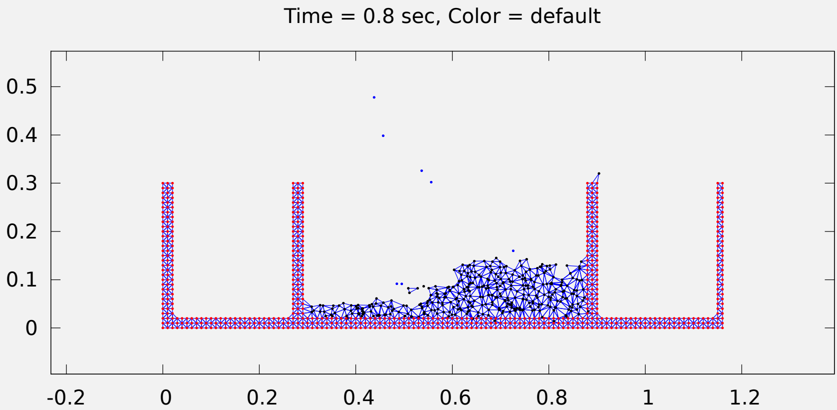
Water column collapse



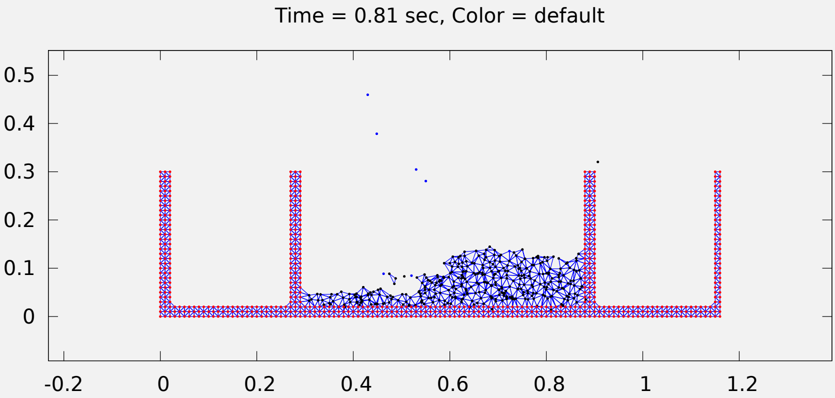
Water column collapse



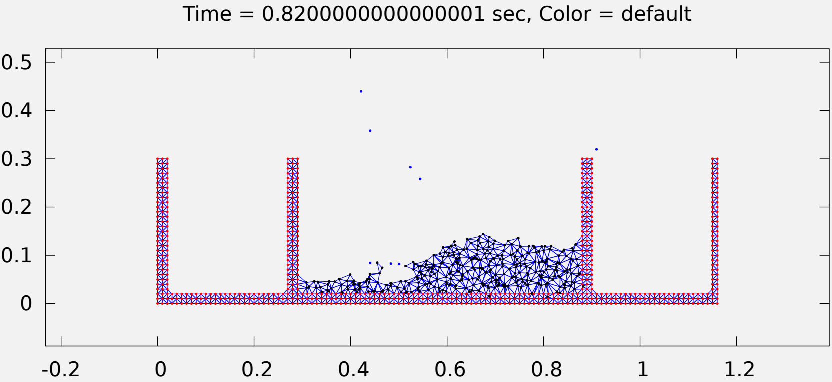
Water column collapse



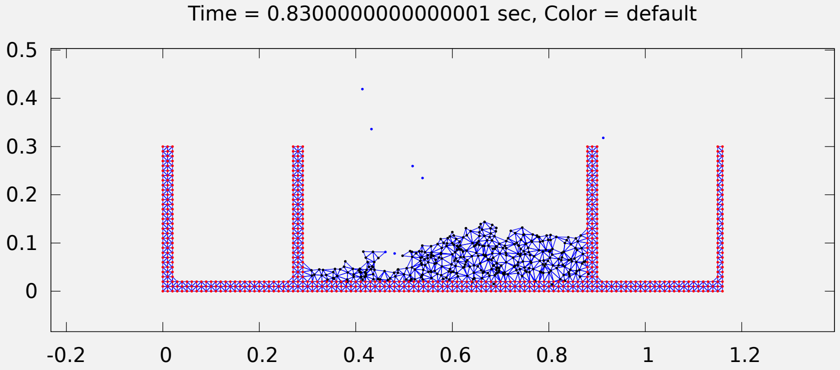
Water column collapse



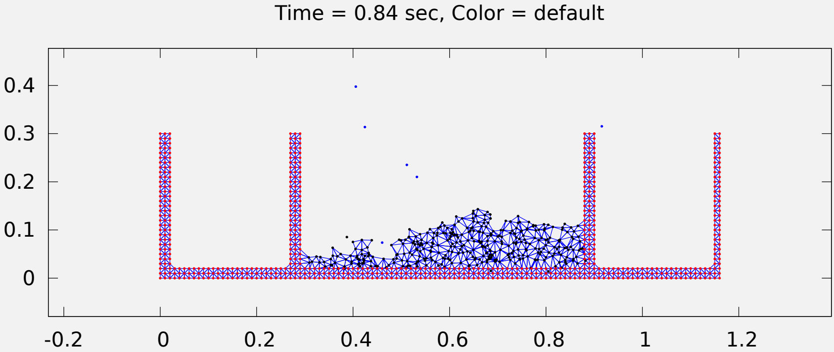
Water column collapse



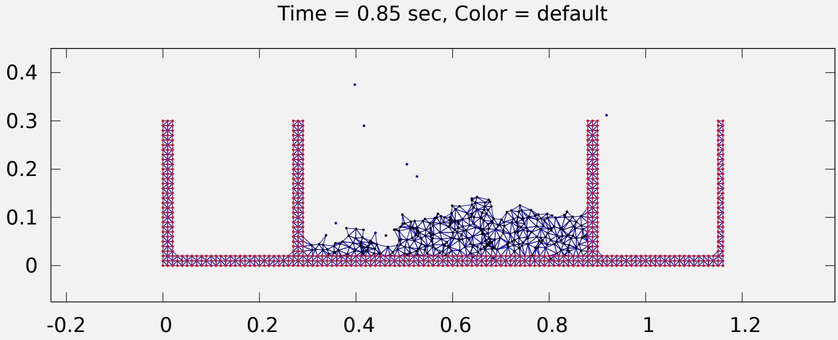
Water column collapse



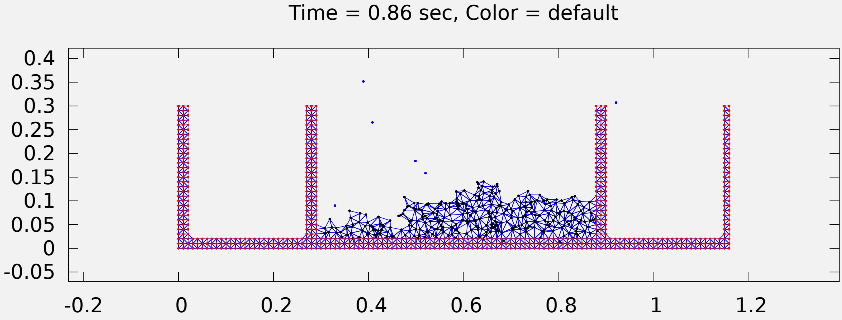
Water column collapse



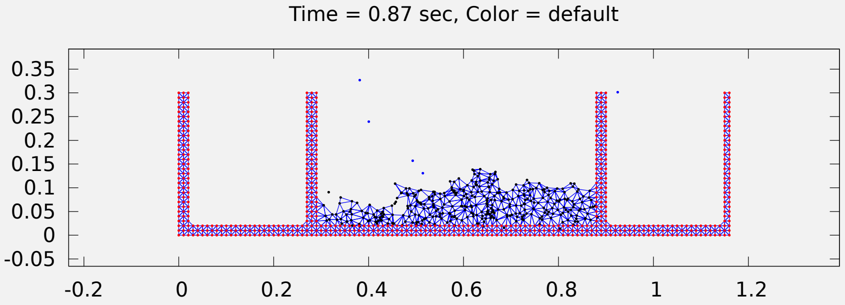
Water column collapse



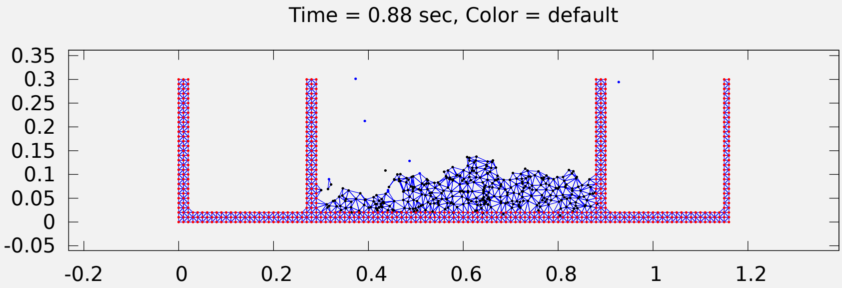
Water column collapse



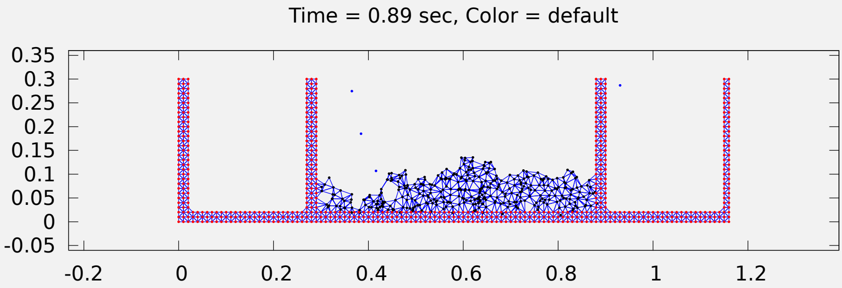
Water column collapse



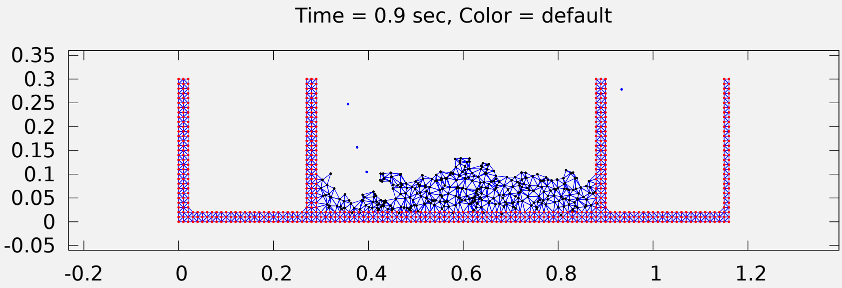
Water column collapse



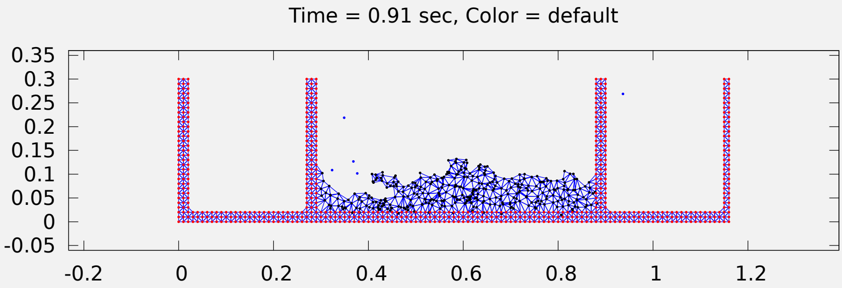
Water column collapse



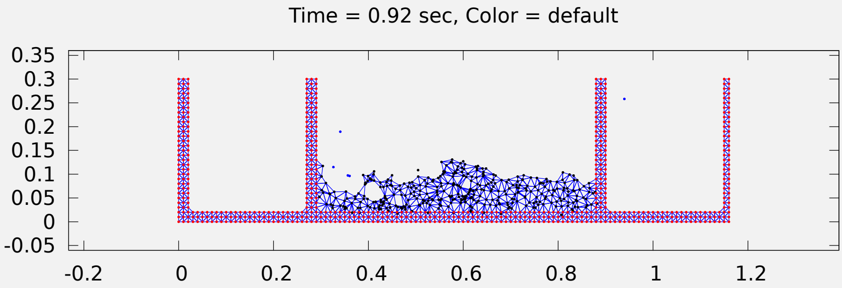
Water column collapse



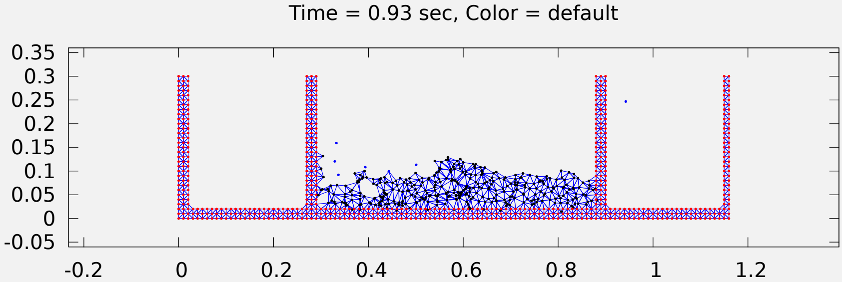
Water column collapse



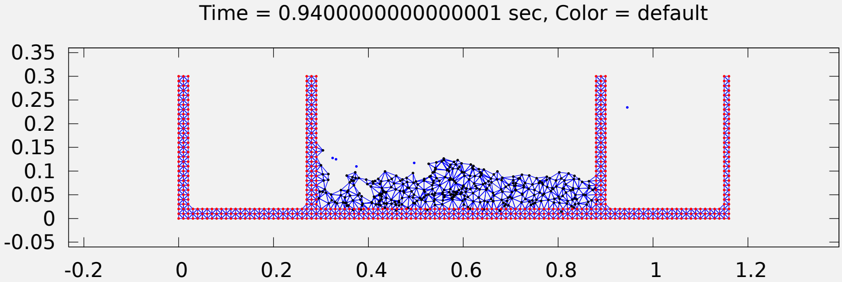
Water column collapse



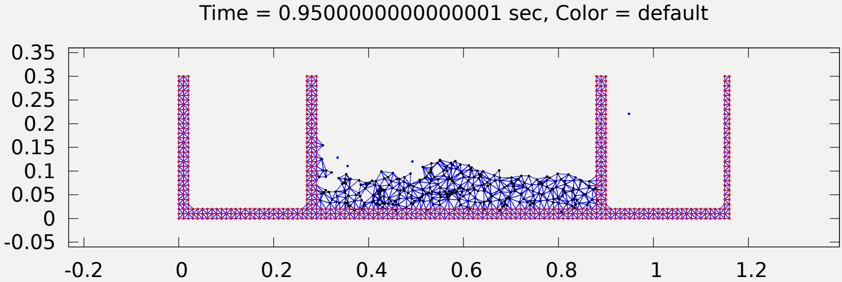
Water column collapse



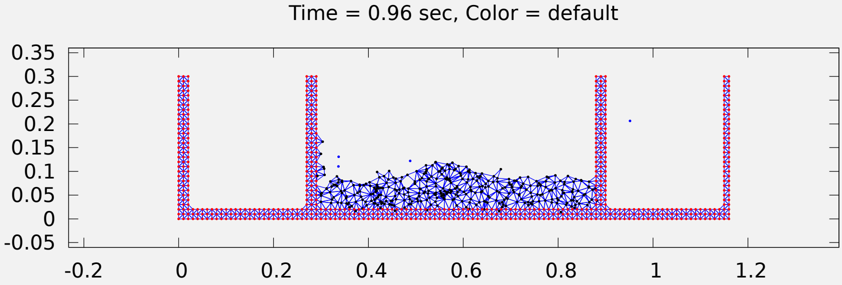
Water column collapse



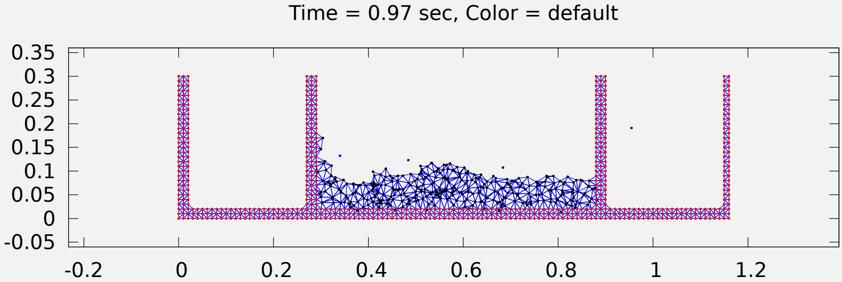
Water column collapse



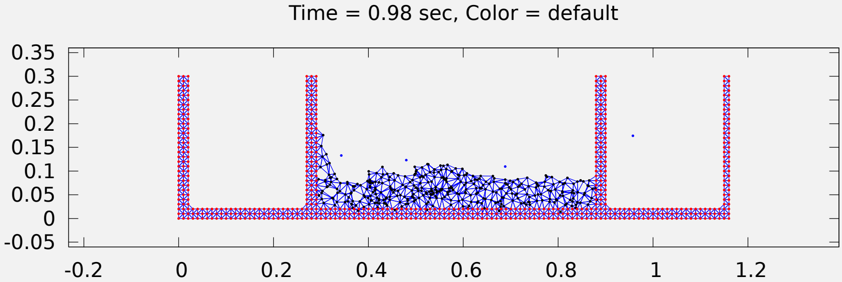
Water column collapse



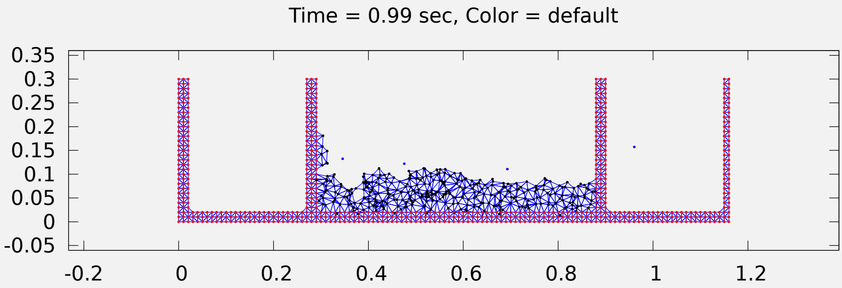
Water column collapse



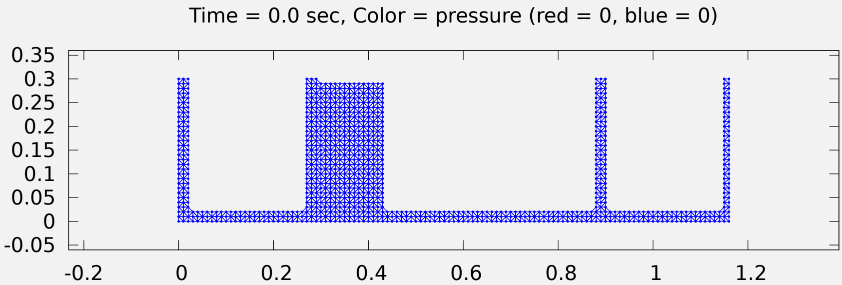
Water column collapse



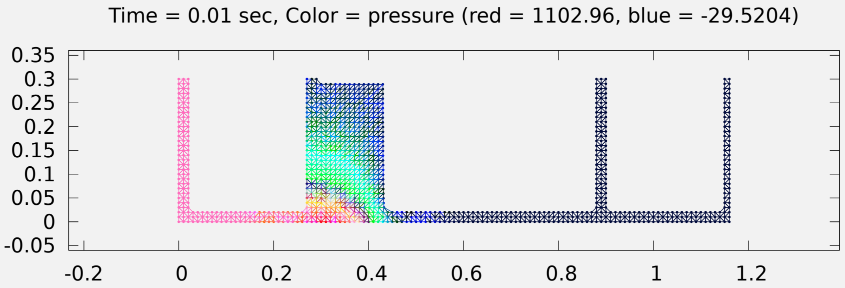
Water column collapse



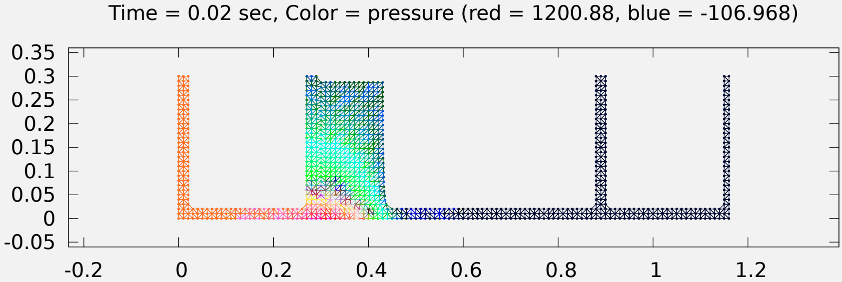
Water column collapse : pressure distribution



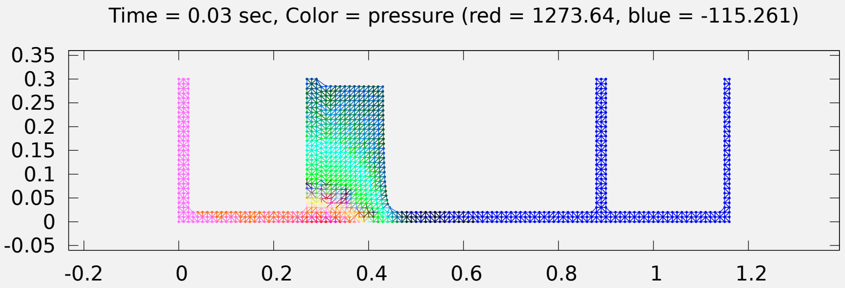
Water column collapse : pressure distribution



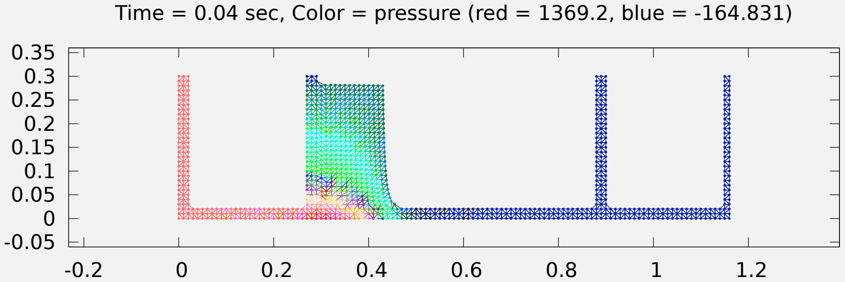
Water column collapse : pressure distribution



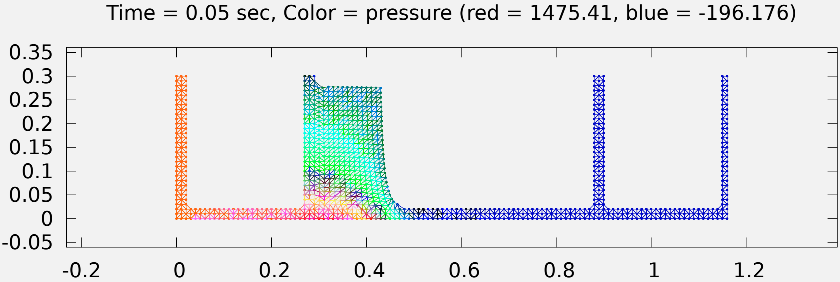
Water column collapse : pressure distribution



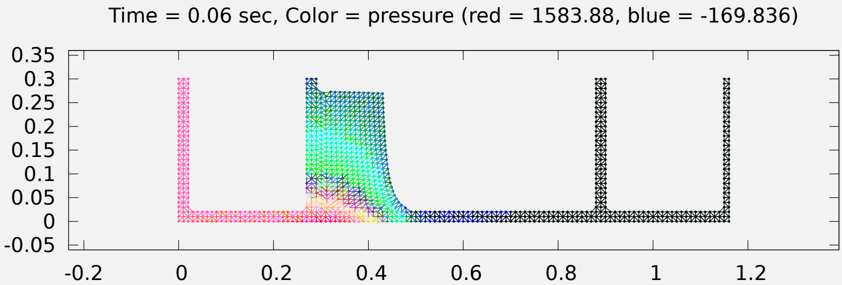
Water column collapse : pressure distribution



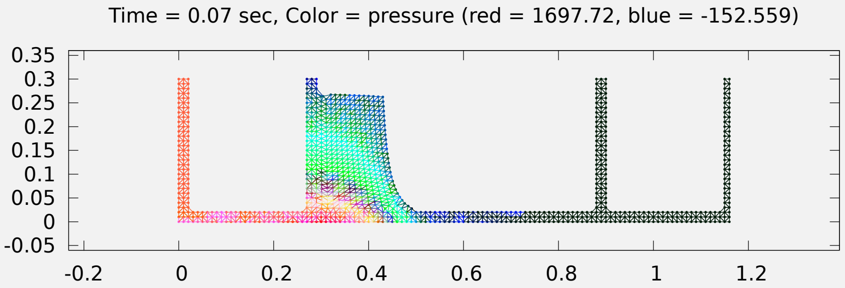
Water column collapse : pressure distribution



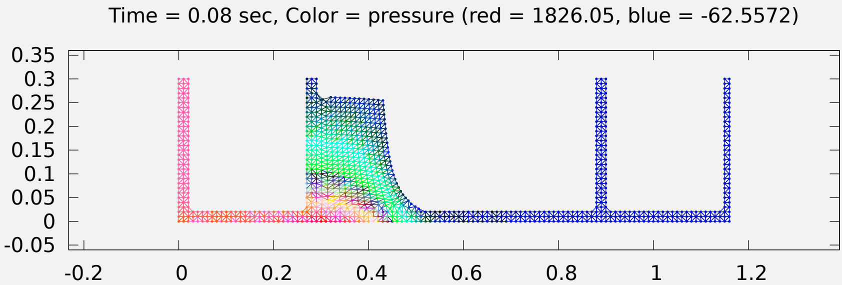
Water column collapse : pressure distribution



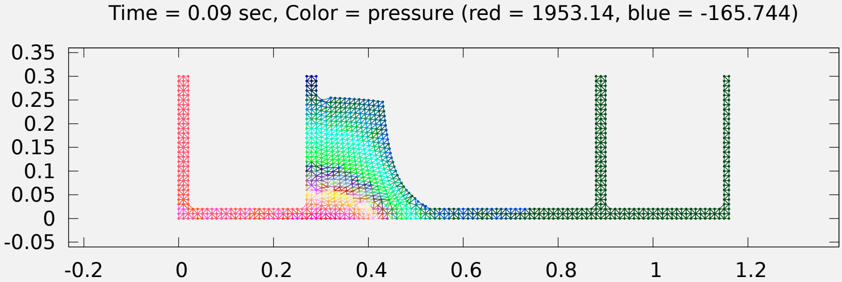
Water column collapse : pressure distribution



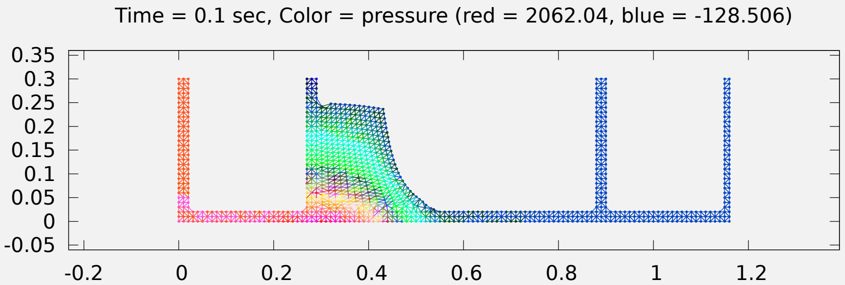
Water column collapse : pressure distribution



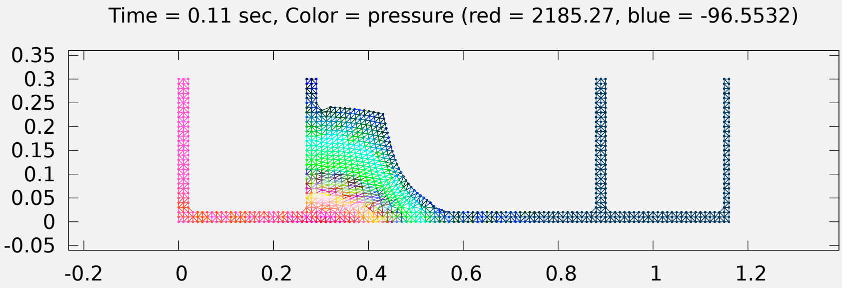
Water column collapse : pressure distribution



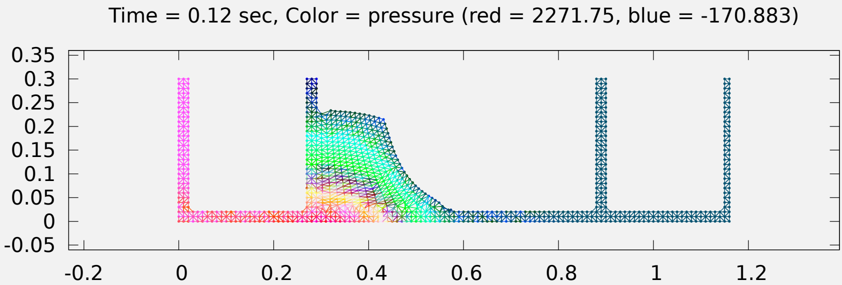
Water column collapse : pressure distribution



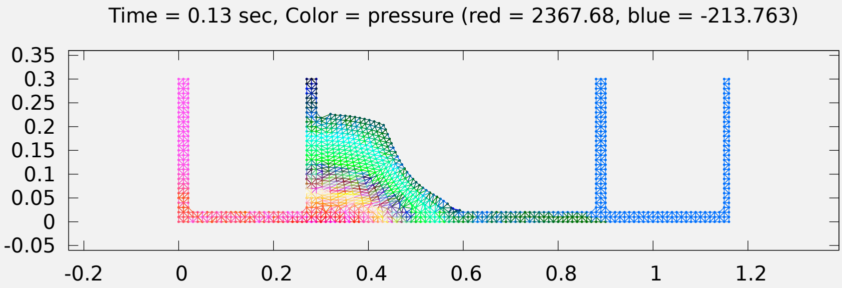
Water column collapse : pressure distribution



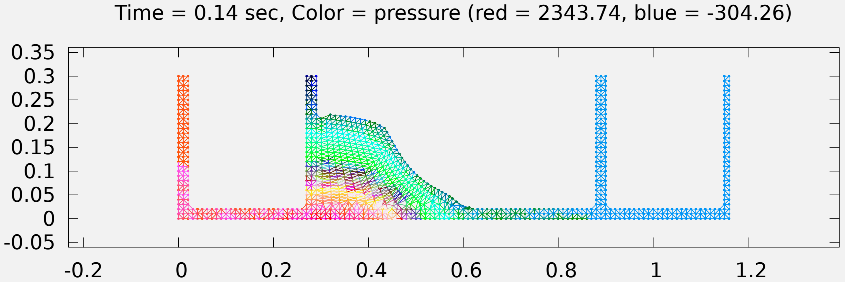
Water column collapse : pressure distribution



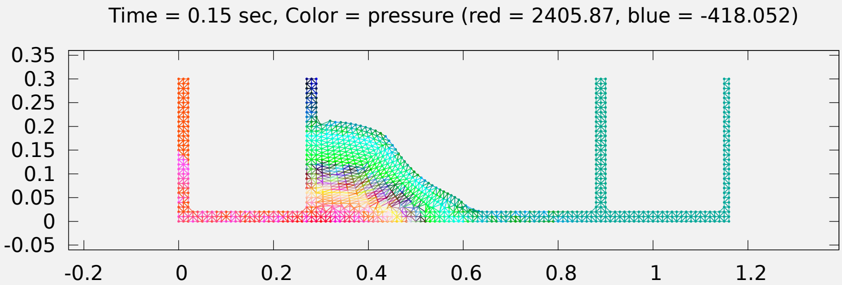
Water column collapse : pressure distribution



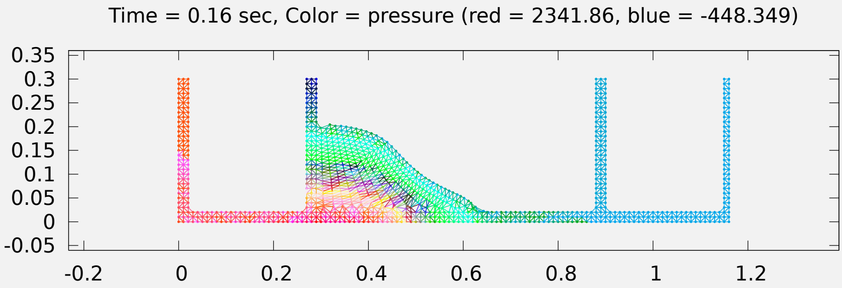
Water column collapse : pressure distribution



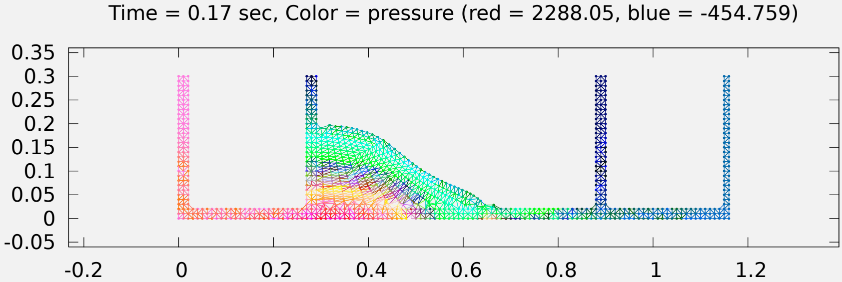
Water column collapse : pressure distribution



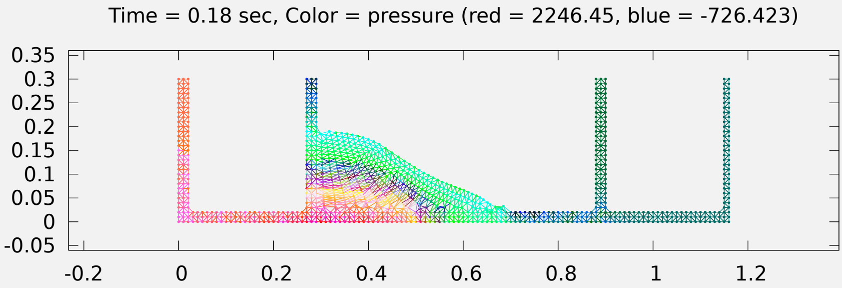
Water column collapse : pressure distribution



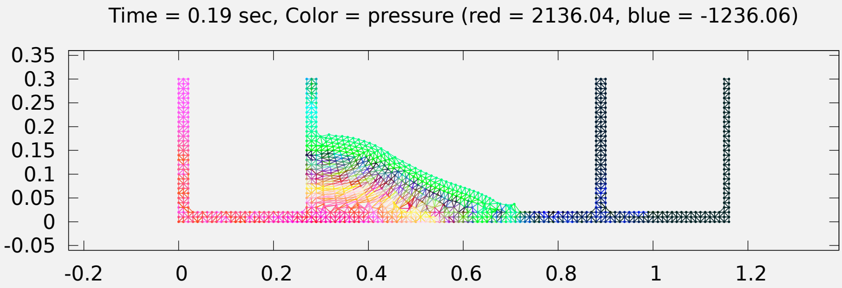
Water column collapse : pressure distribution



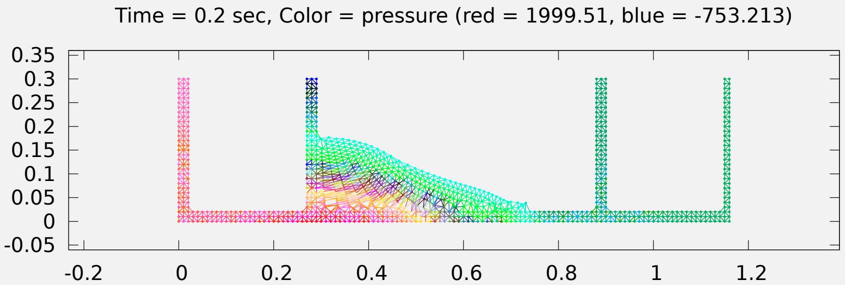
Water column collapse : pressure distribution



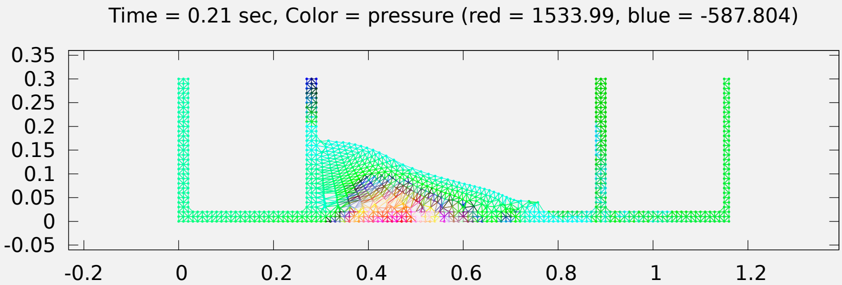
Water column collapse : pressure distribution



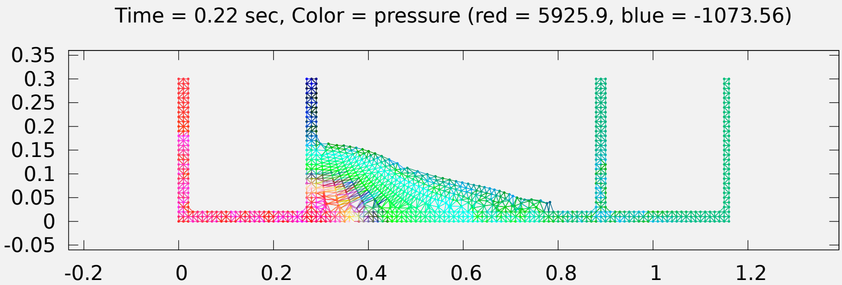
Water column collapse : pressure distribution



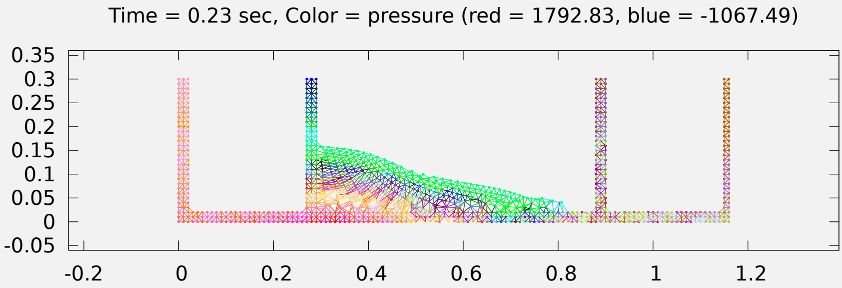
Water column collapse : pressure distribution



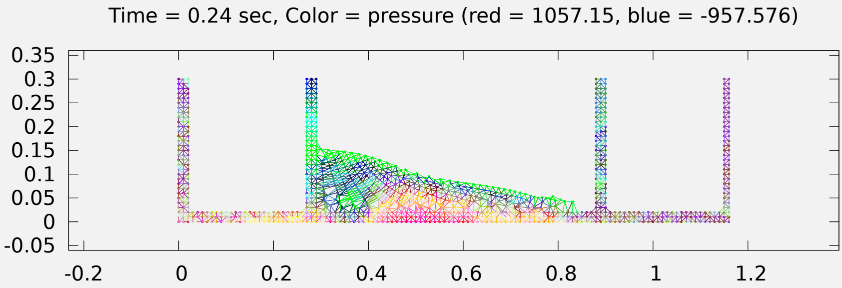
Water column collapse : pressure distribution



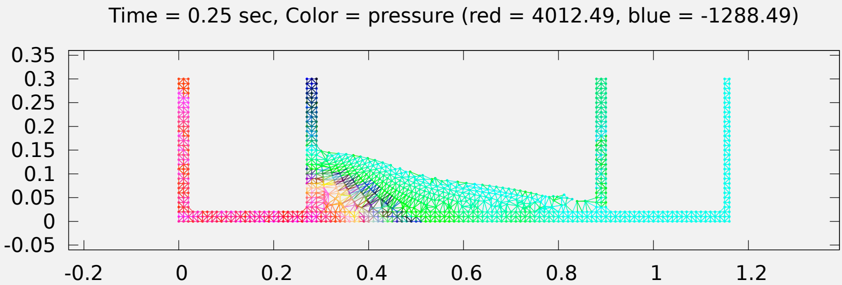
Water column collapse : pressure distribution



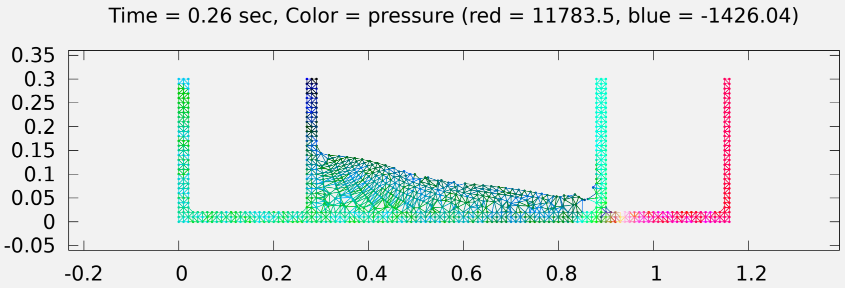
Water column collapse : pressure distribution



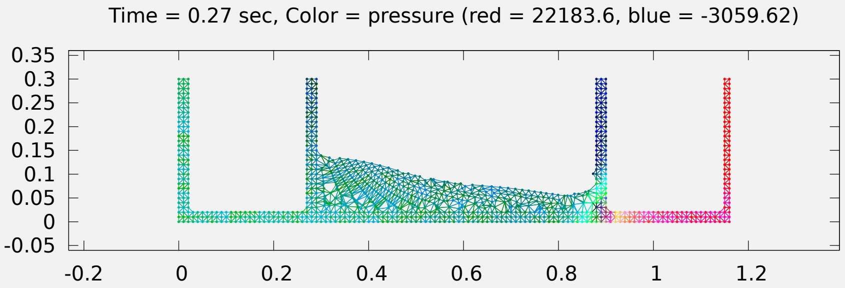
Water column collapse : pressure distribution



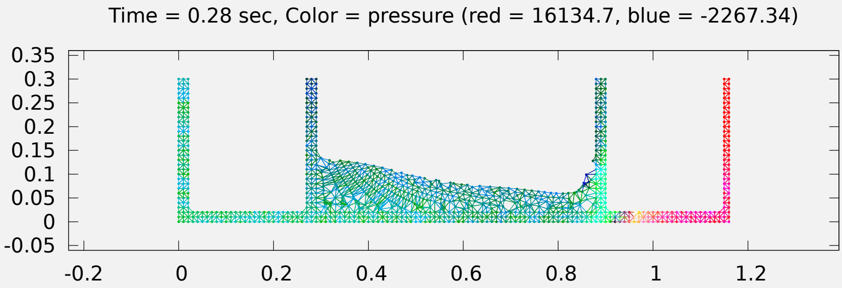
Water column collapse : pressure distribution



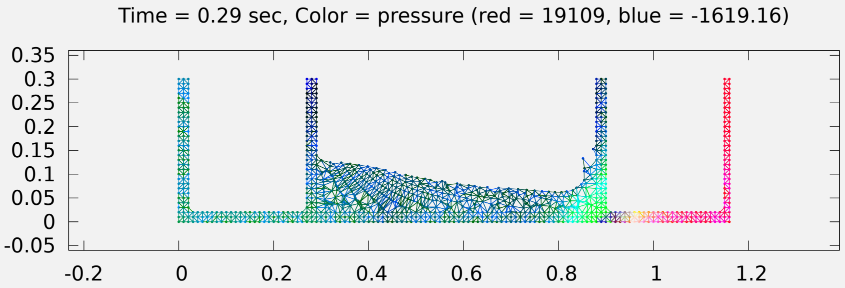
Water column collapse : pressure distribution



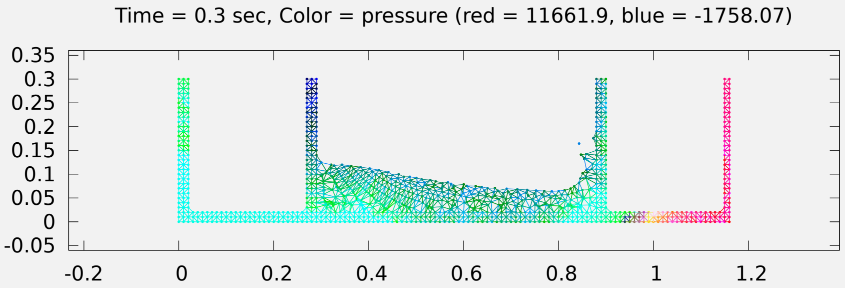
Water column collapse : pressure distribution



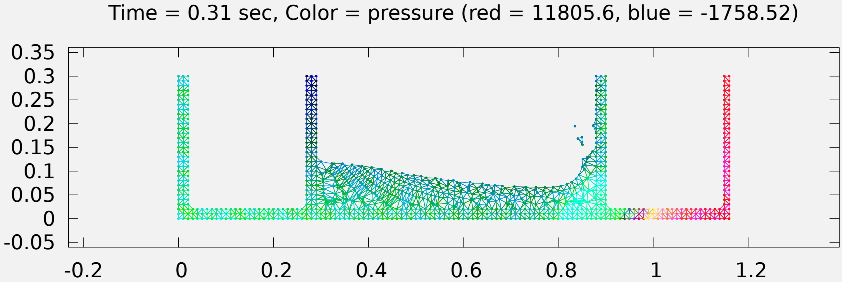
Water column collapse : pressure distribution



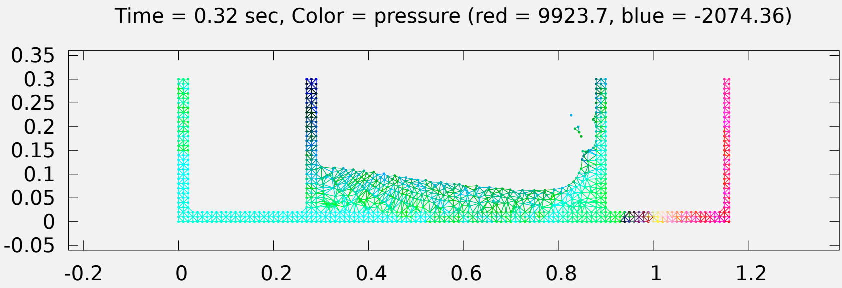
Water column collapse : pressure distribution



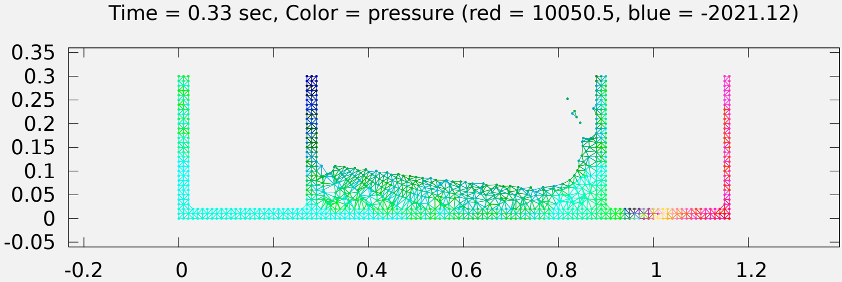
Water column collapse : pressure distribution



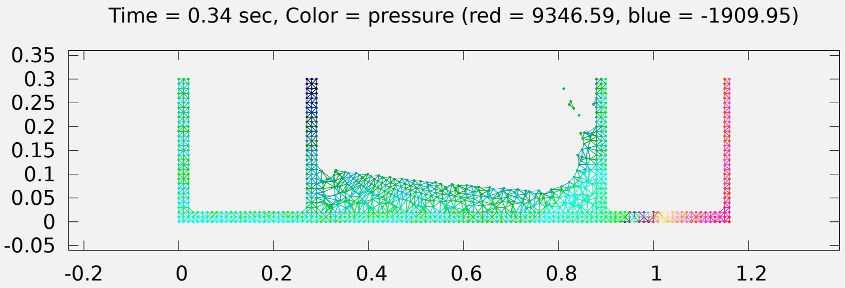
Water column collapse : pressure distribution



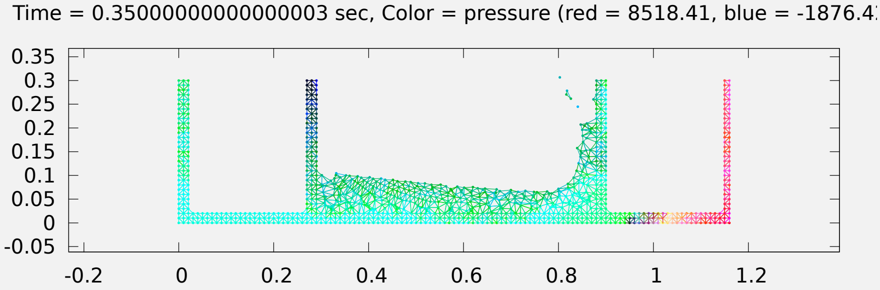
Water column collapse : pressure distribution



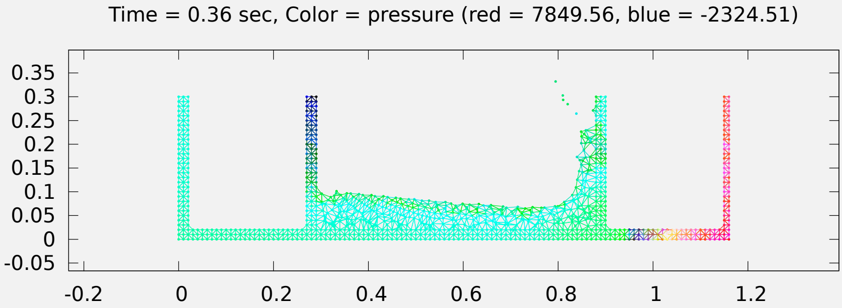
Water column collapse : pressure distribution



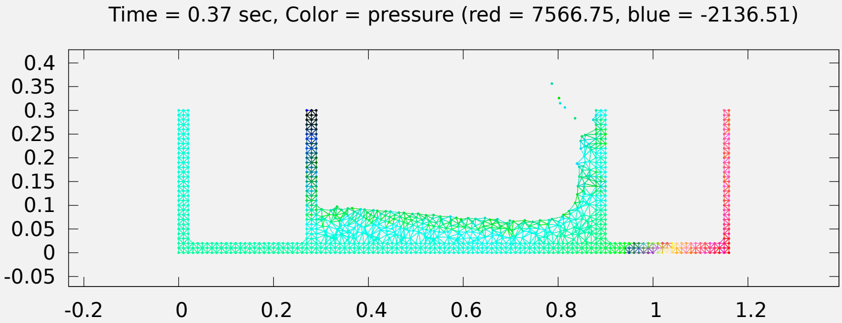
Water column collapse : pressure distribution



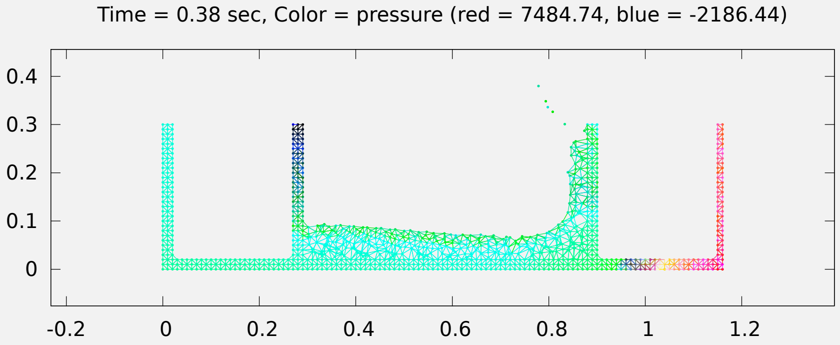
Water column collapse : pressure distribution



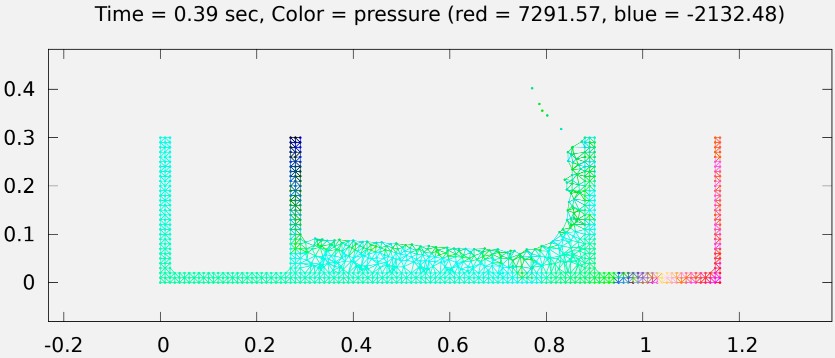
Water column collapse : pressure distribution



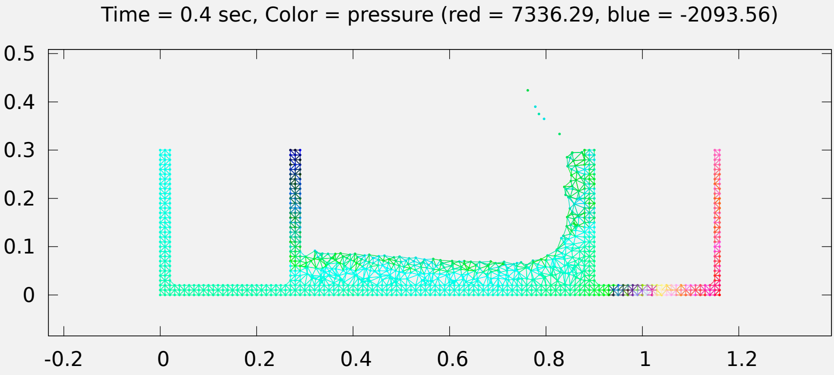
Water column collapse : pressure distribution



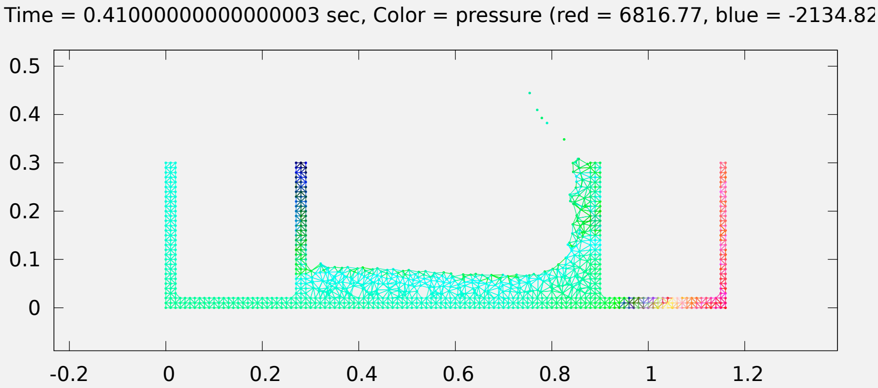
Water column collapse : pressure distribution



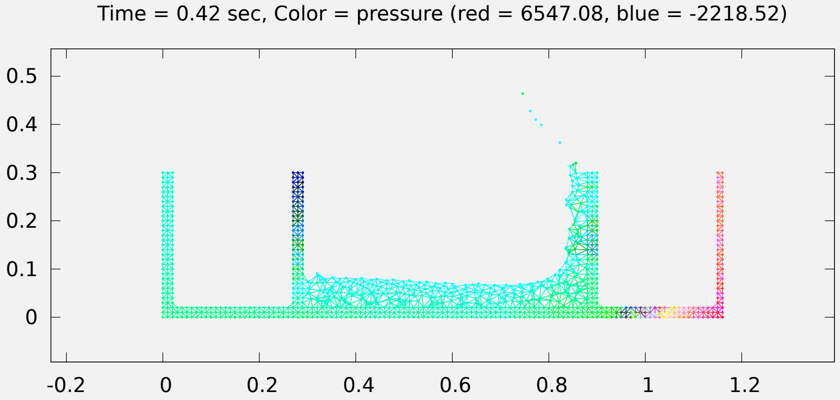
Water column collapse : pressure distribution



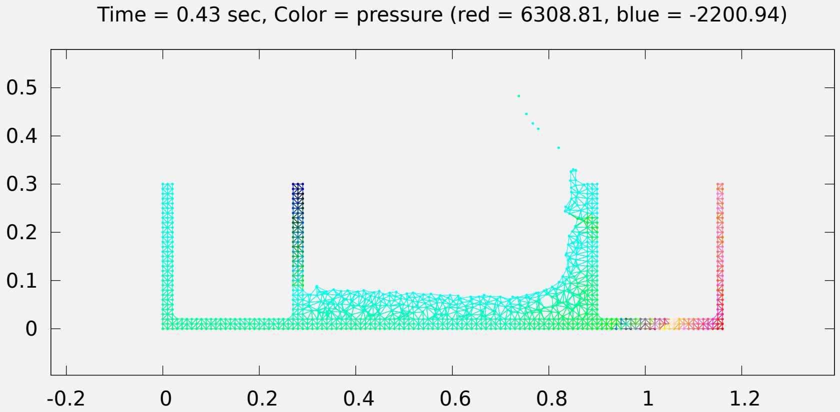
Water column collapse : pressure distribution



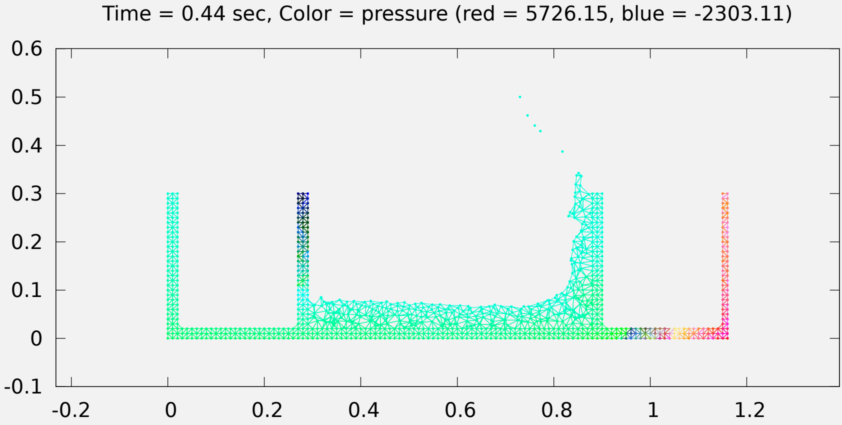
Water column collapse : pressure distribution



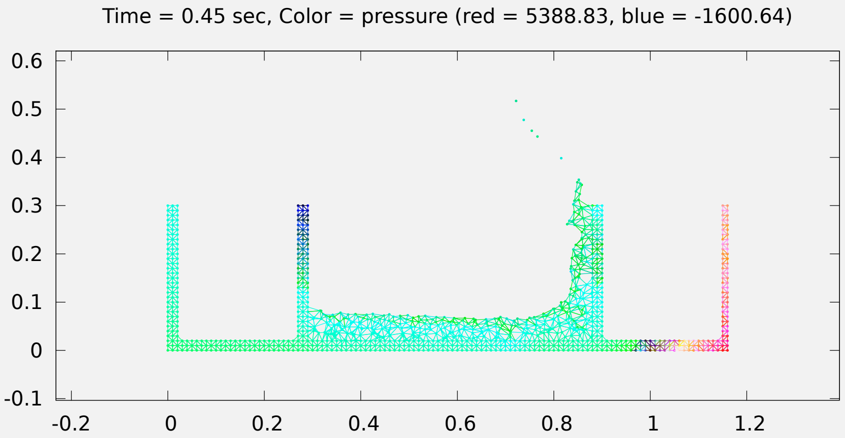
Water column collapse : pressure distribution



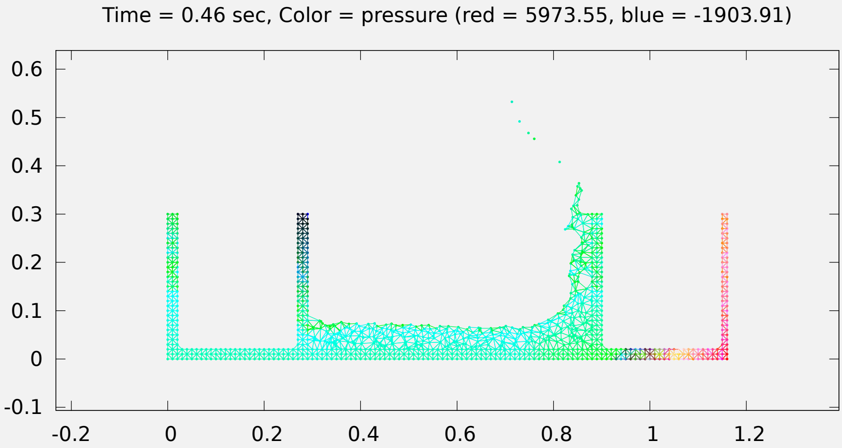
Water column collapse : pressure distribution



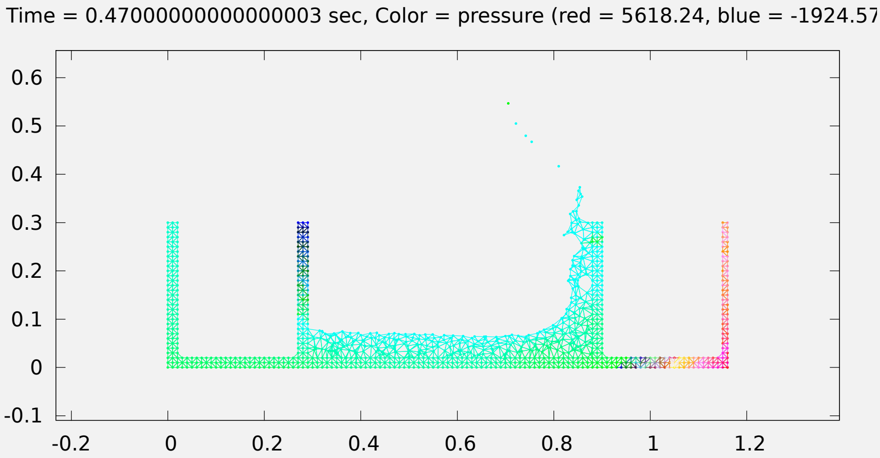
Water column collapse : pressure distribution



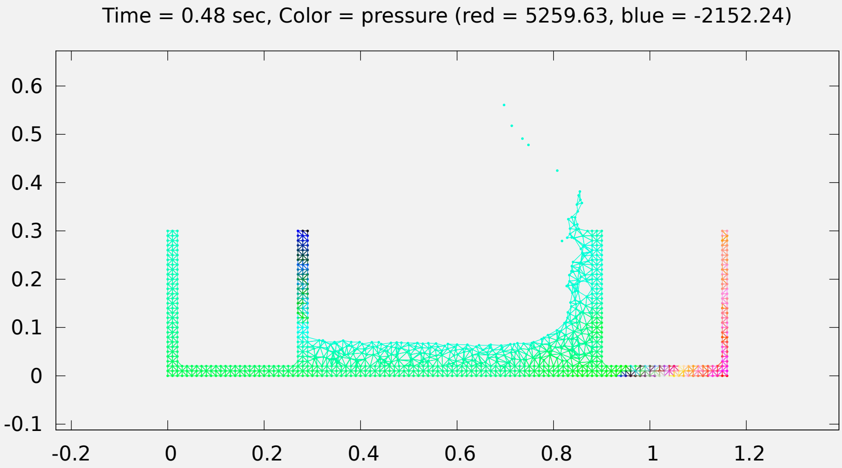
Water column collapse : pressure distribution



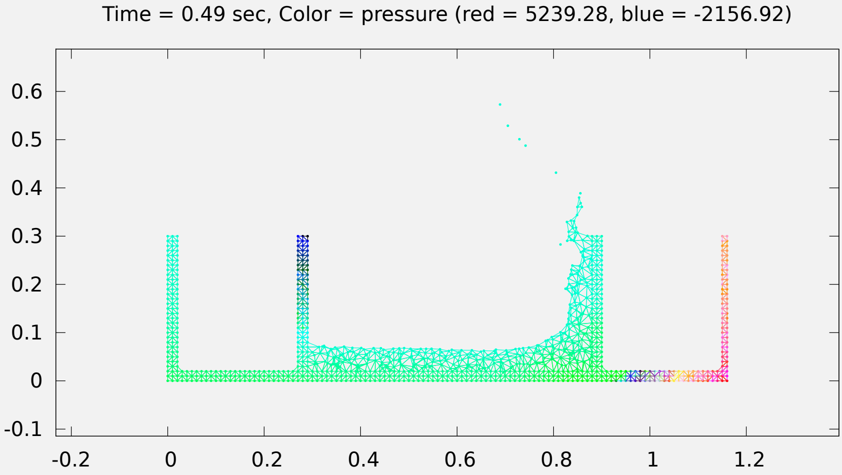
Water column collapse : pressure distribution



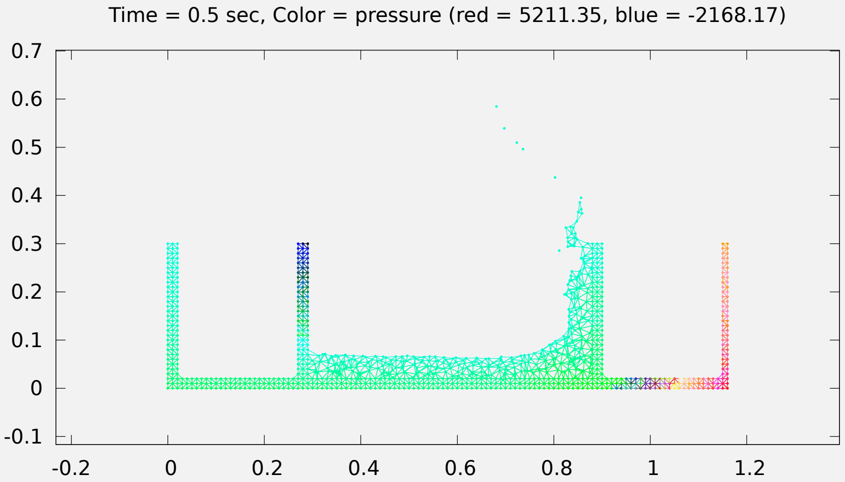
Water column collapse : pressure distribution



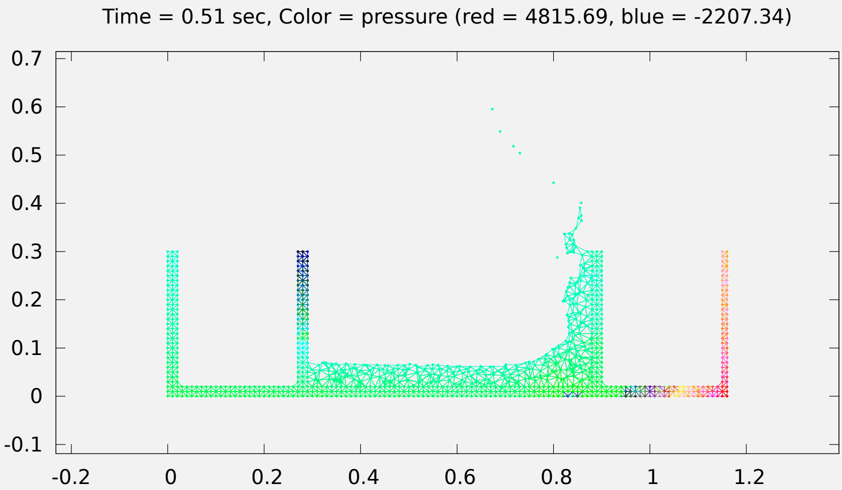
Water column collapse : pressure distribution



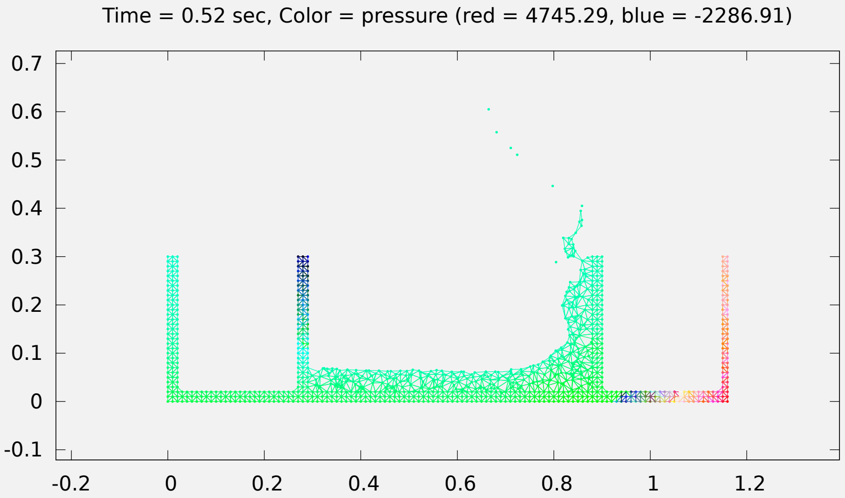
Water column collapse : pressure distribution



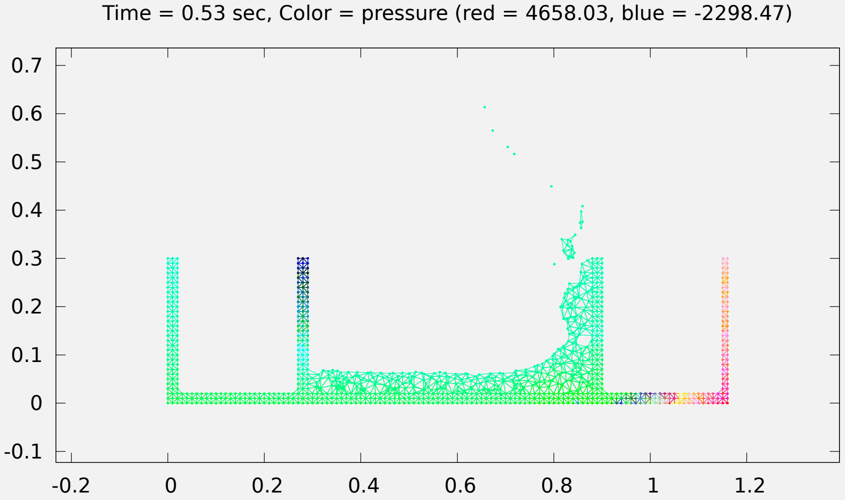
Water column collapse : pressure distribution



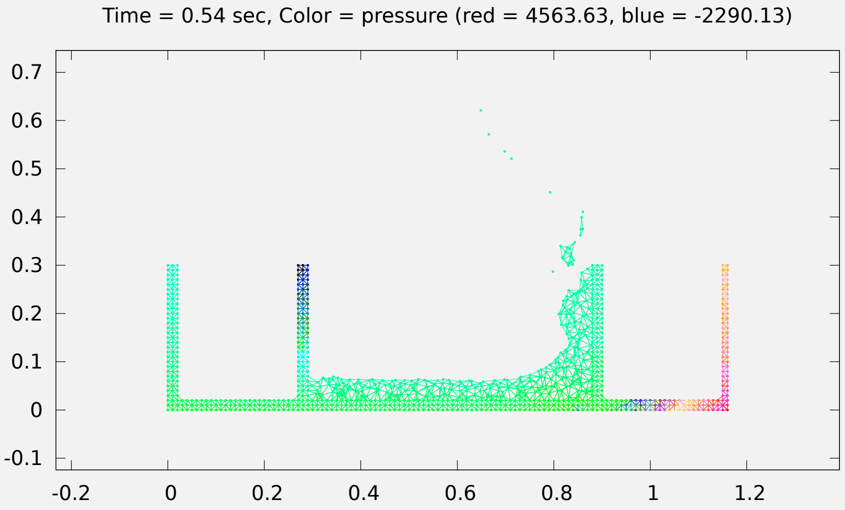
Water column collapse : pressure distribution



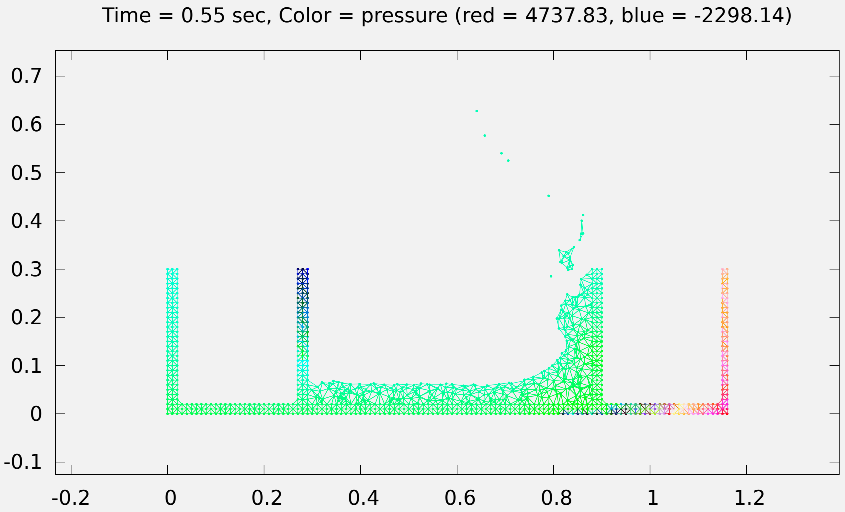
Water column collapse : pressure distribution



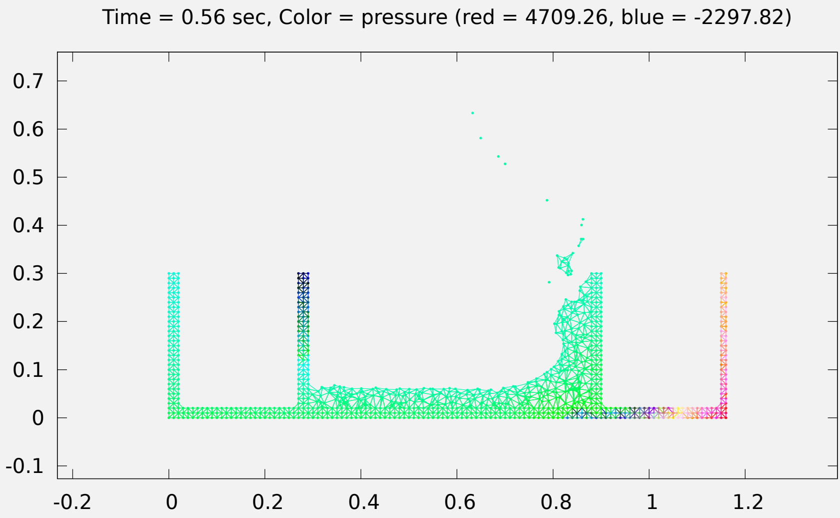
Water column collapse : pressure distribution



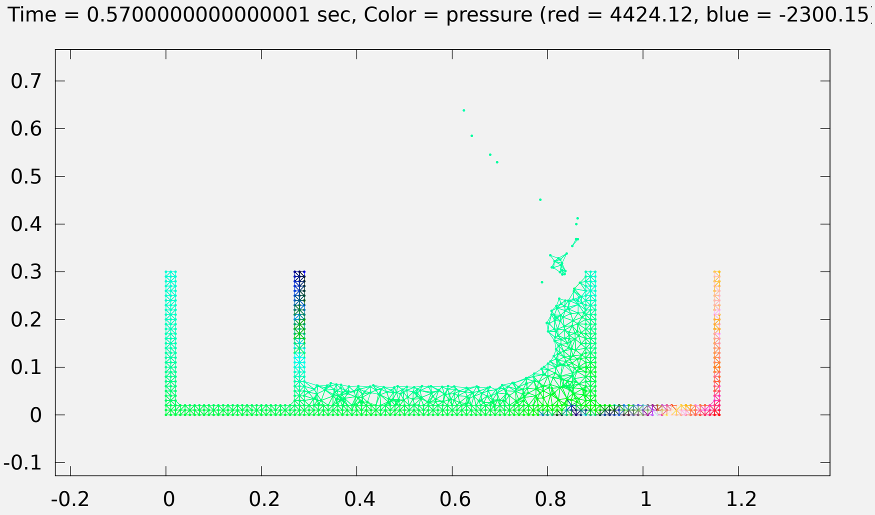
Water column collapse : pressure distribution



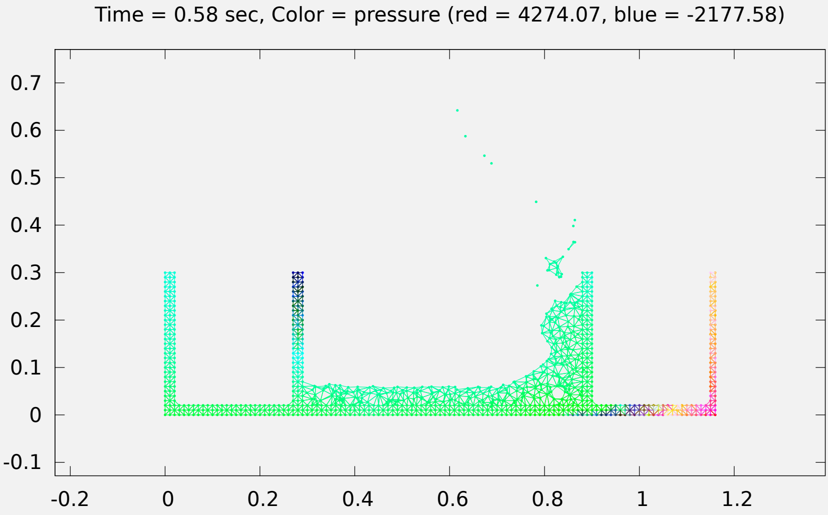
Water column collapse : pressure distribution



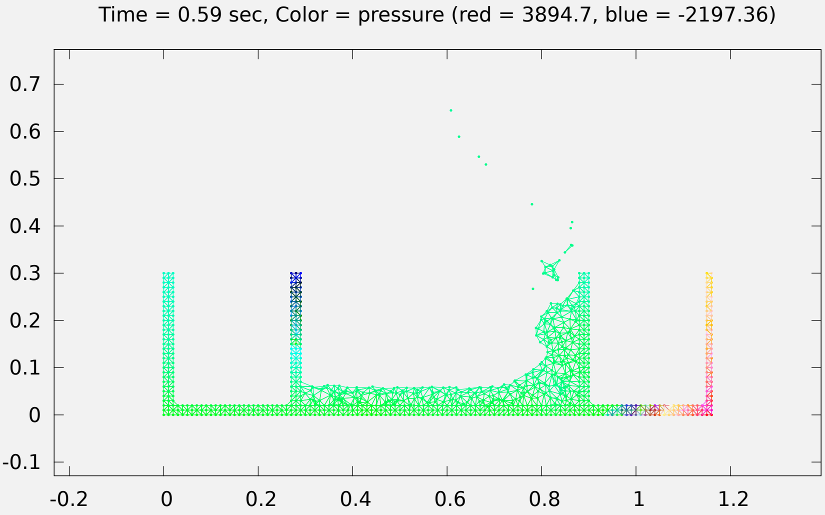
Water column collapse : pressure distribution



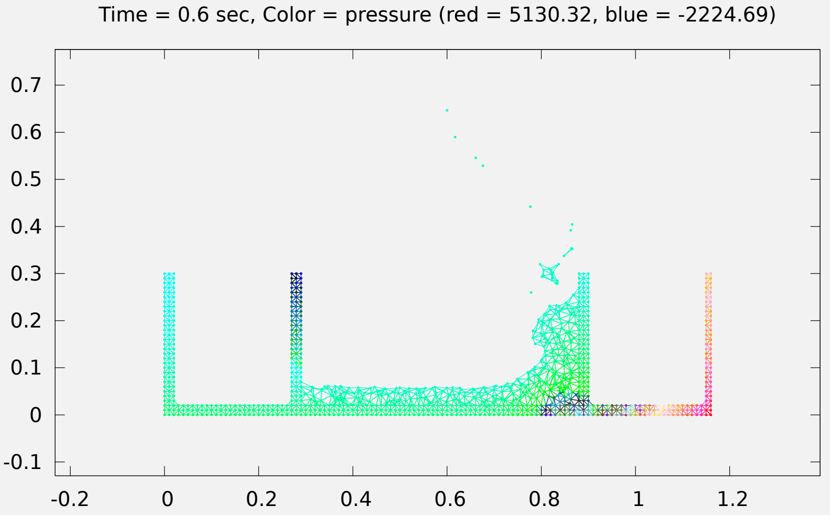
Water column collapse : pressure distribution



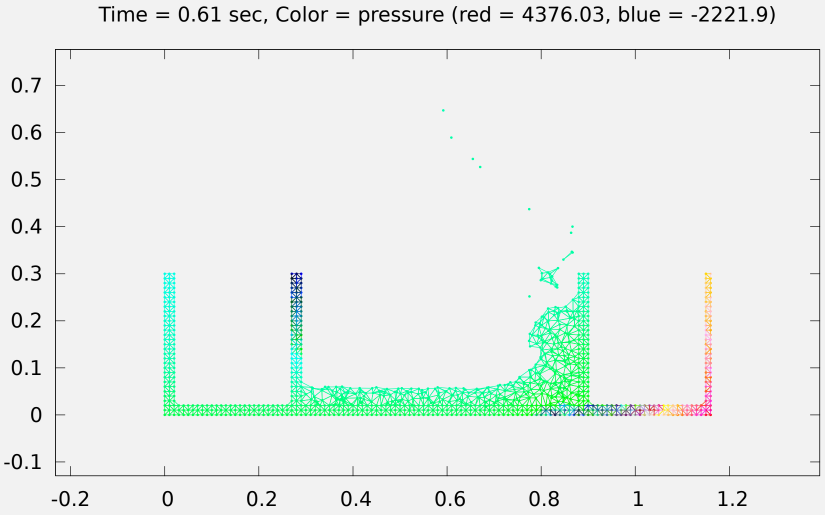
Water column collapse : pressure distribution



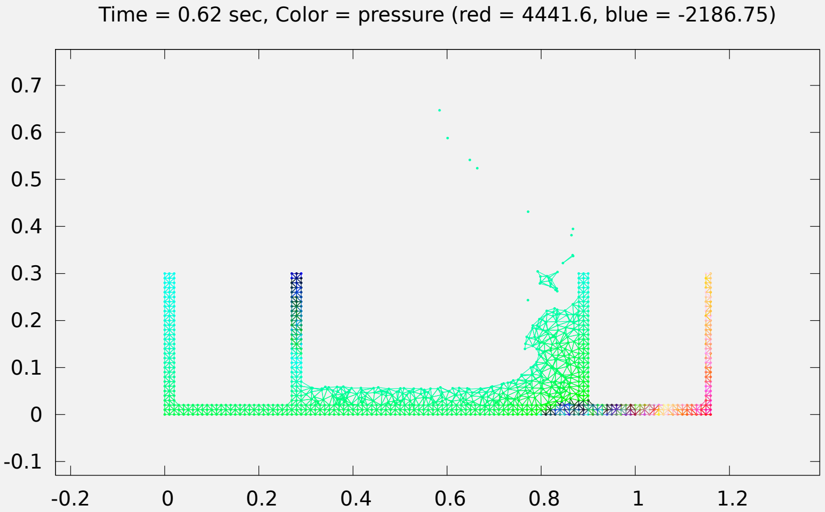
Water column collapse : pressure distribution



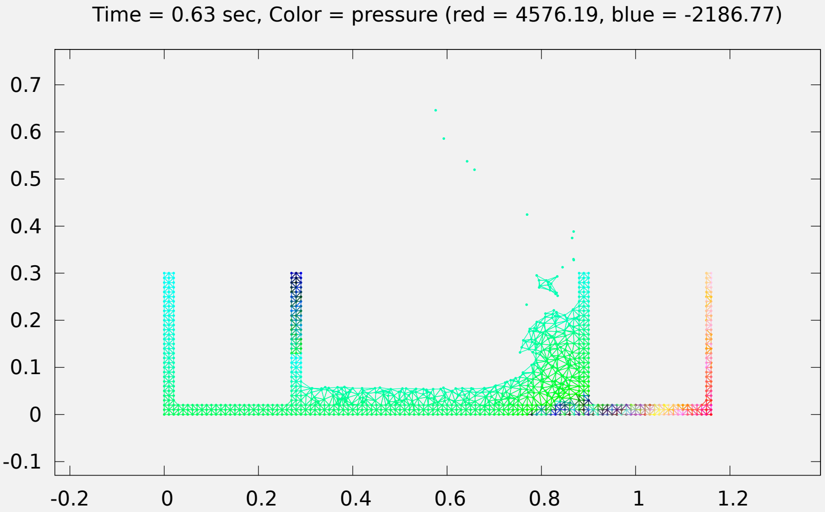
Water column collapse : pressure distribution



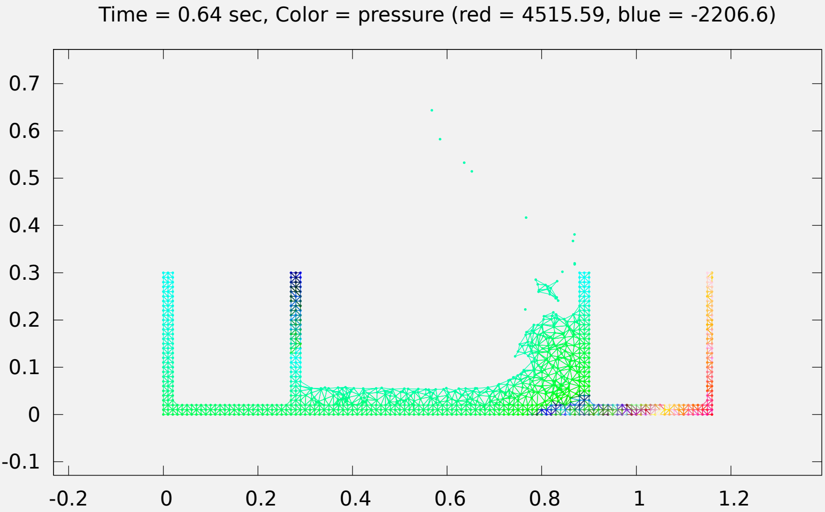
Water column collapse : pressure distribution



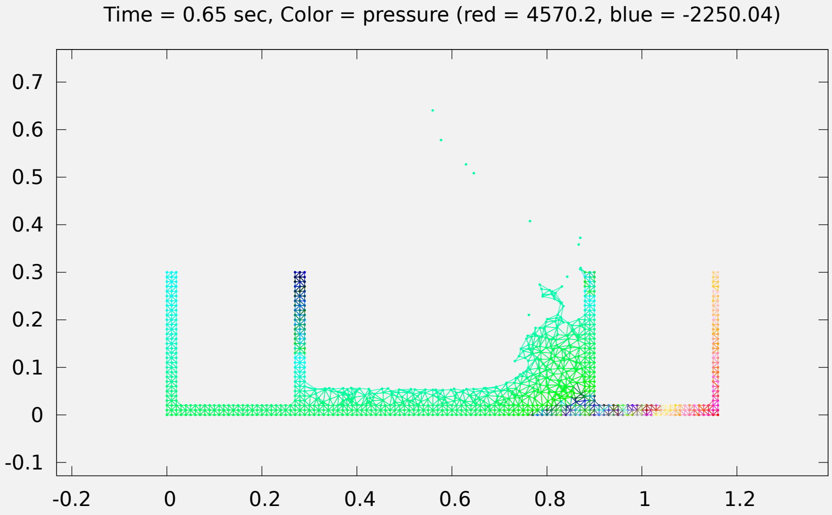
Water column collapse : pressure distribution



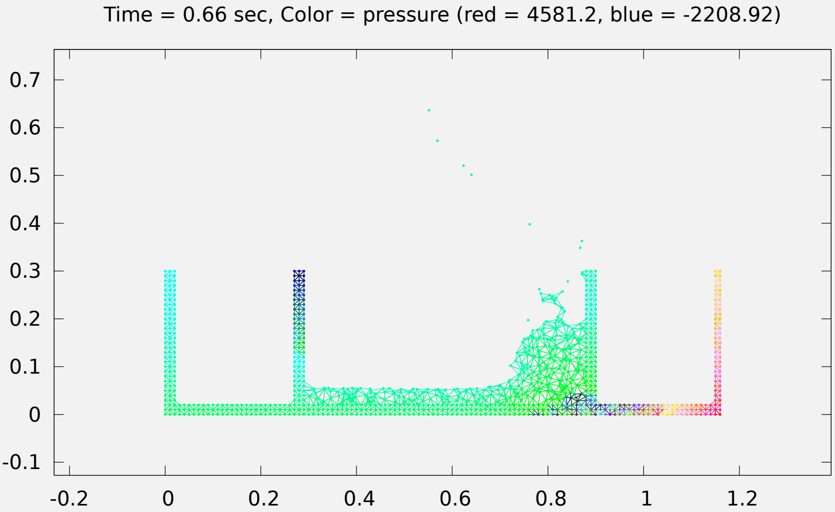
Water column collapse : pressure distribution



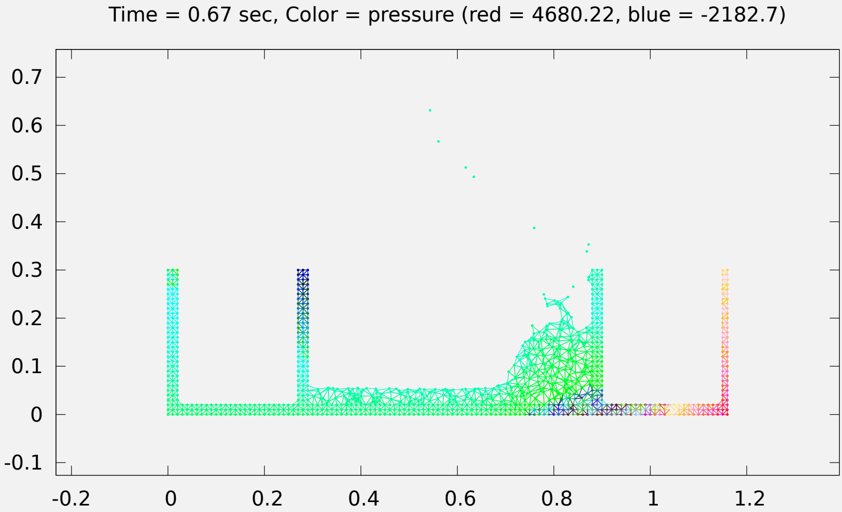
Water column collapse : pressure distribution



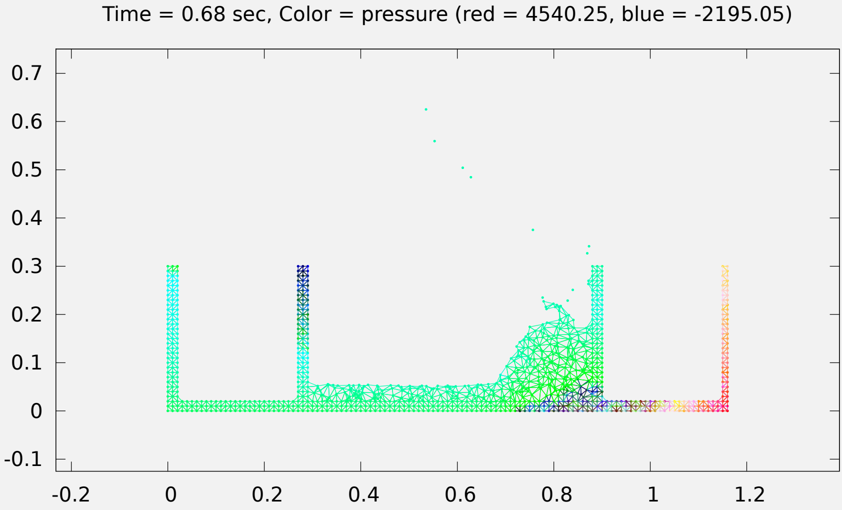
Water column collapse : pressure distribution



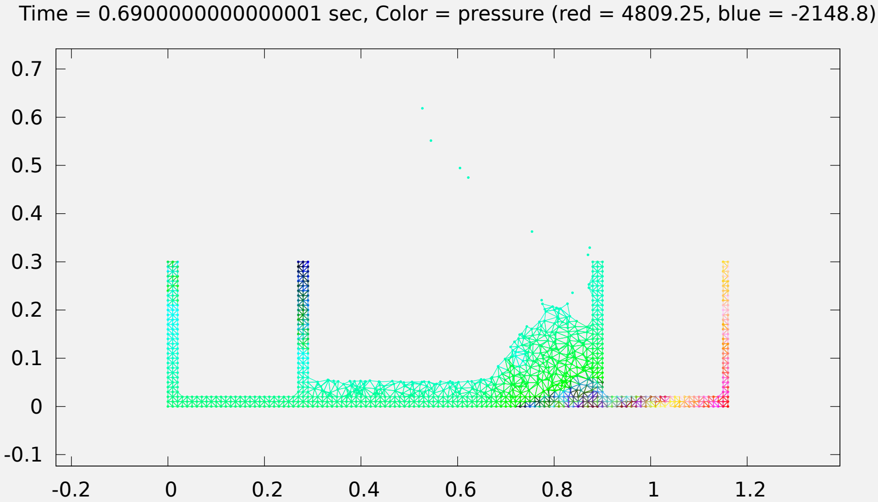
Water column collapse : pressure distribution



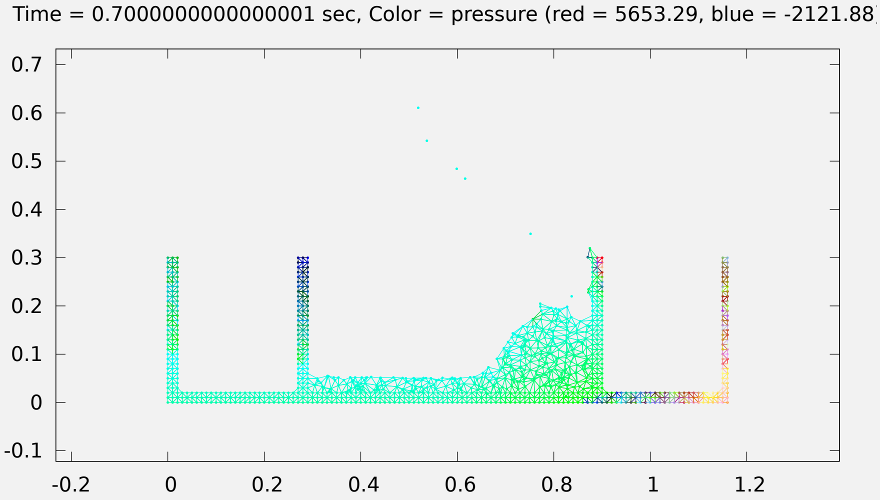
Water column collapse : pressure distribution



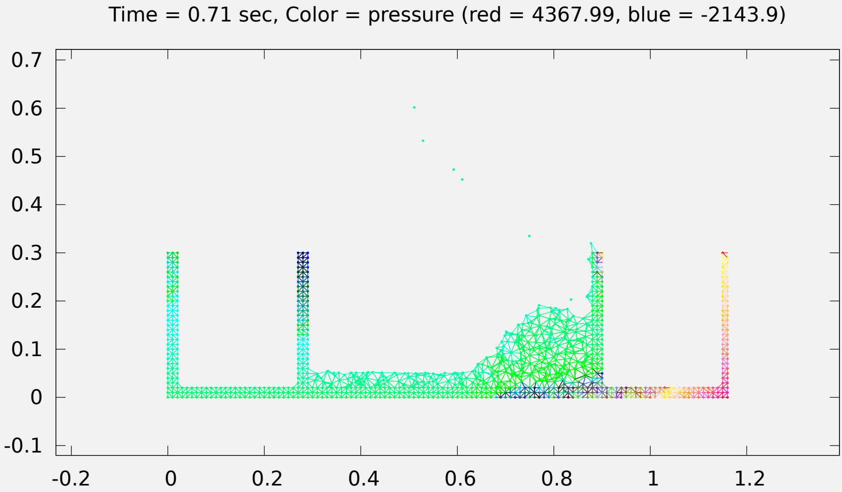
Water column collapse : pressure distribution



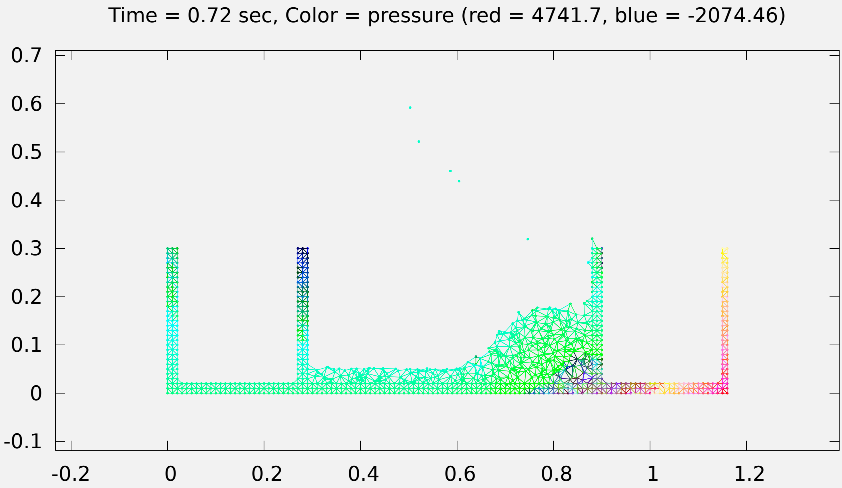
Water column collapse : pressure distribution



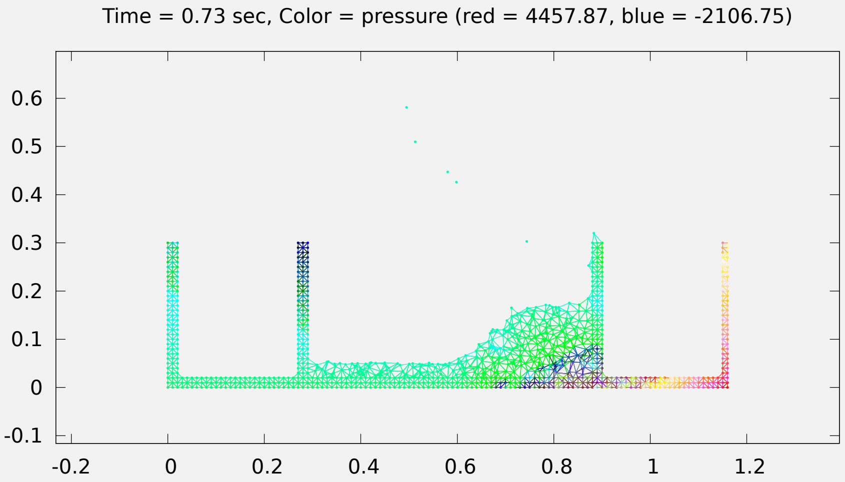
Water column collapse : pressure distribution



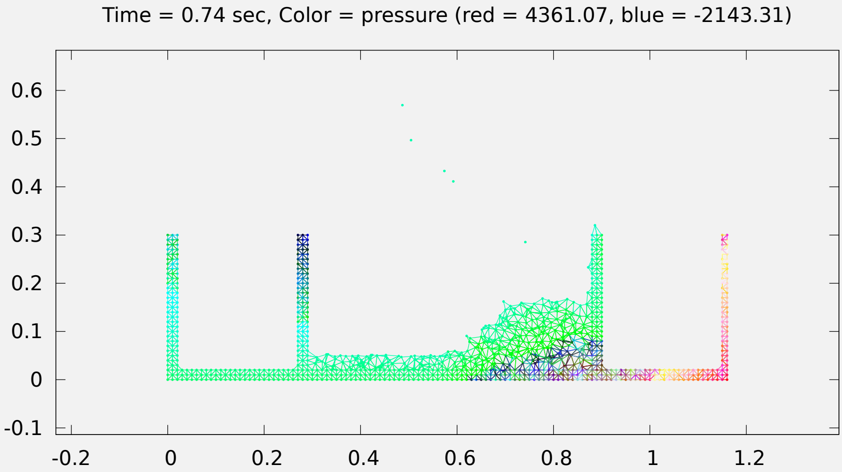
Water column collapse : pressure distribution



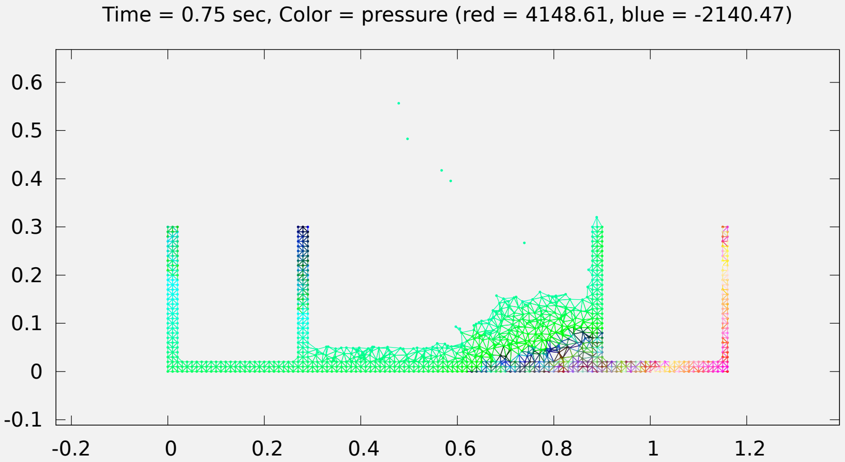
Water column collapse : pressure distribution



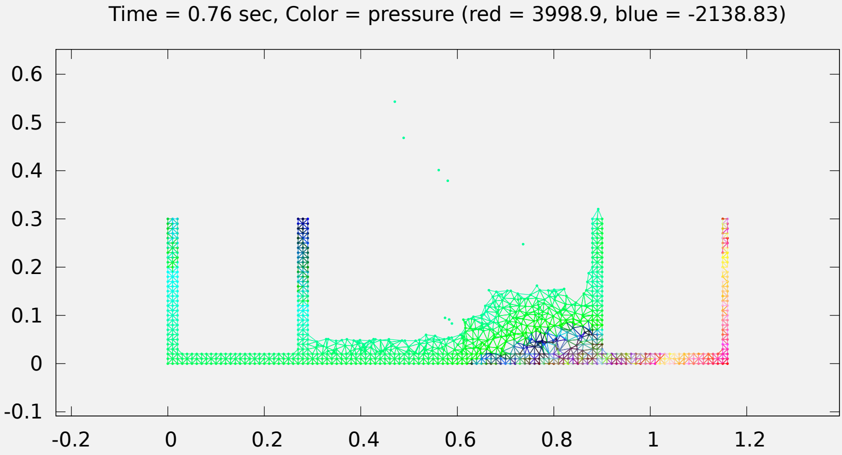
Water column collapse : pressure distribution



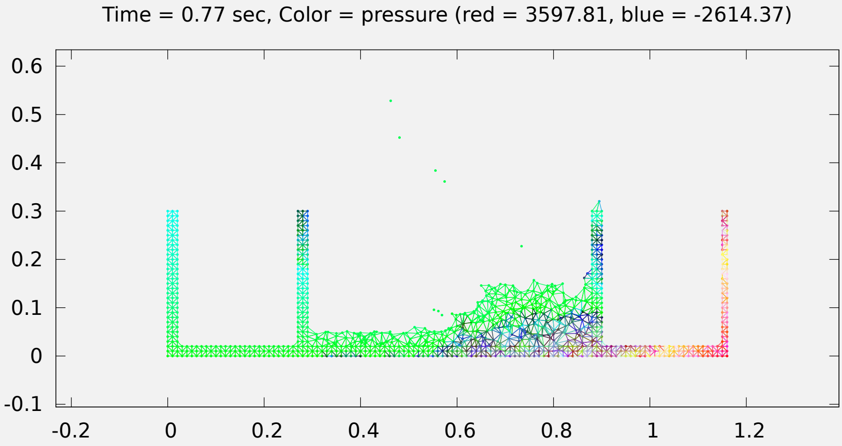
Water column collapse : pressure distribution



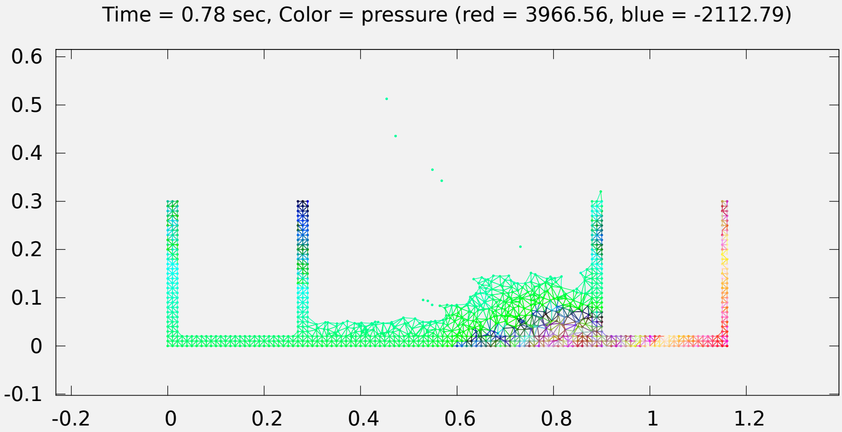
Water column collapse : pressure distribution



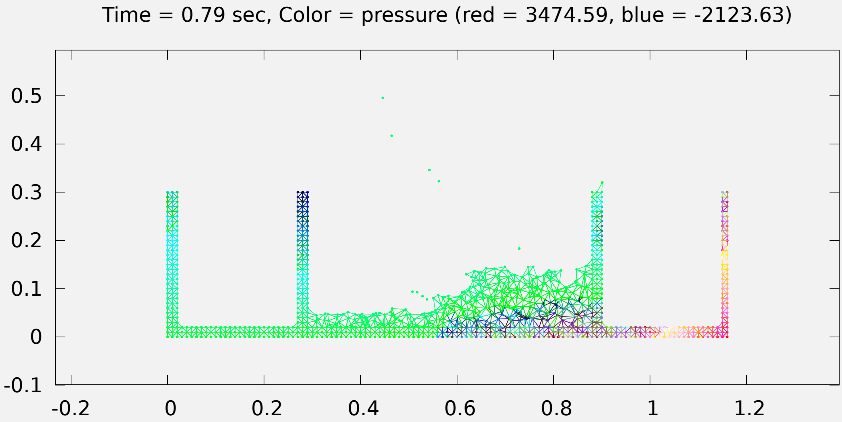
Water column collapse : pressure distribution



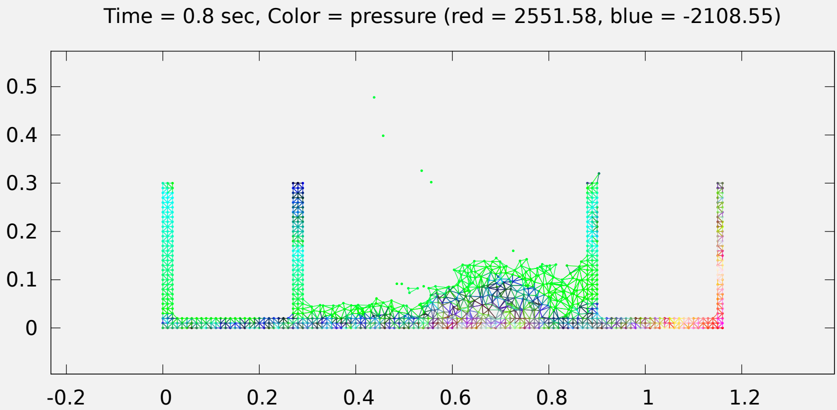
Water column collapse : pressure distribution



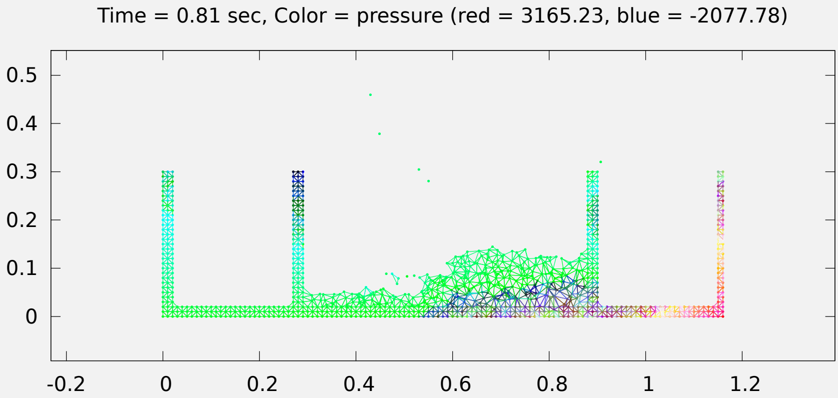
Water column collapse : pressure distribution



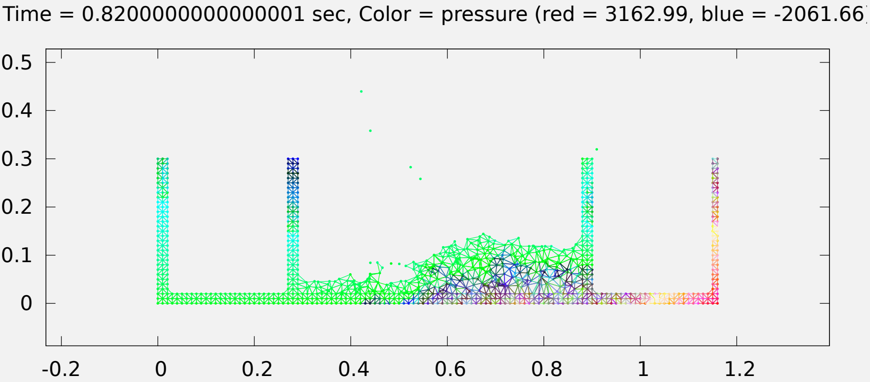
Water column collapse : pressure distribution



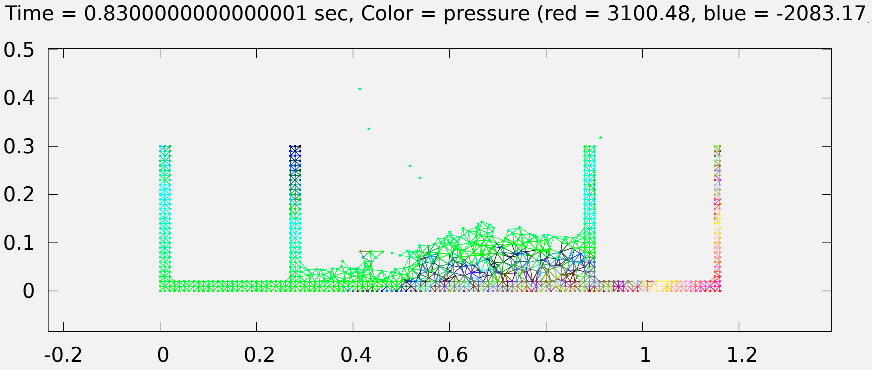
Water column collapse : pressure distribution



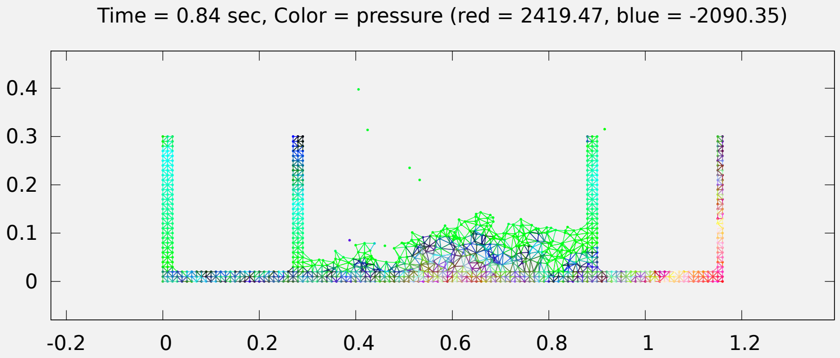
Water column collapse : pressure distribution



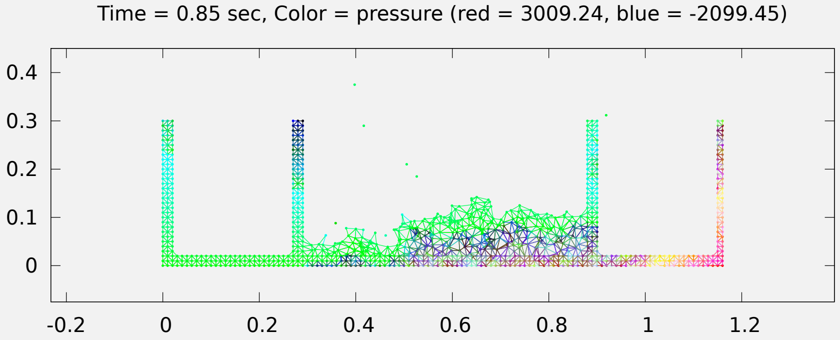
Water column collapse : pressure distribution



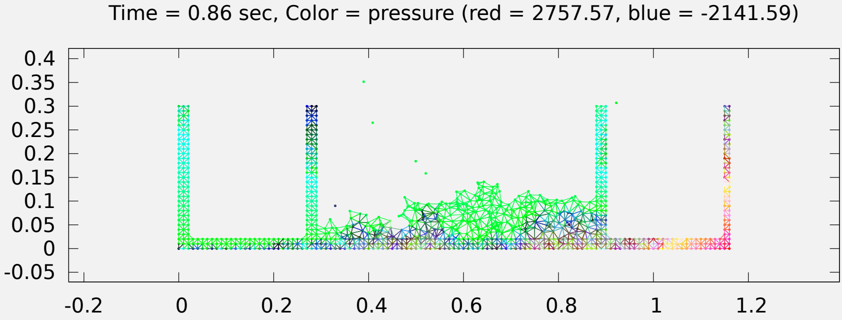
Water column collapse : pressure distribution



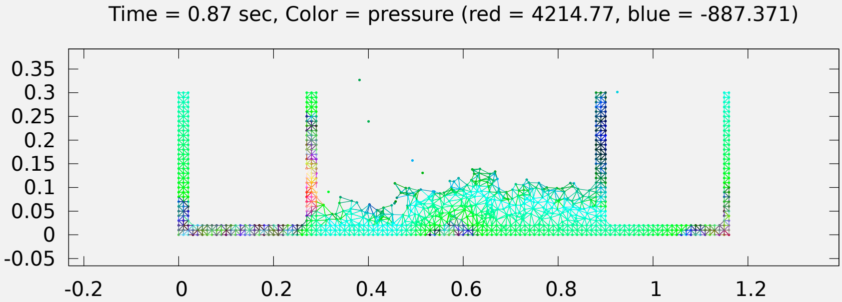
Water column collapse : pressure distribution



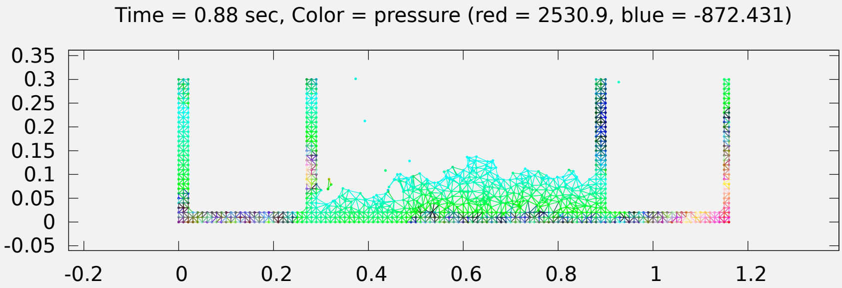
Water column collapse : pressure distribution



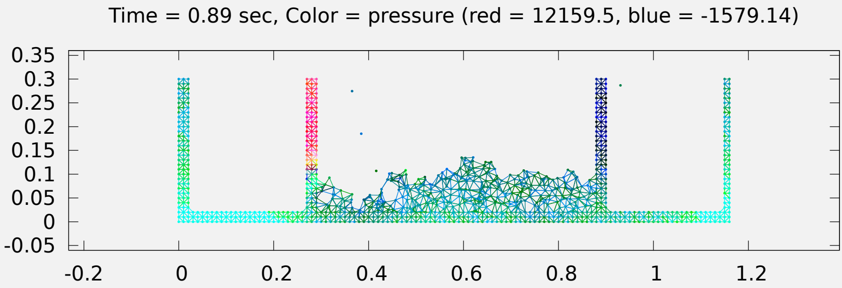
Water column collapse : pressure distribution



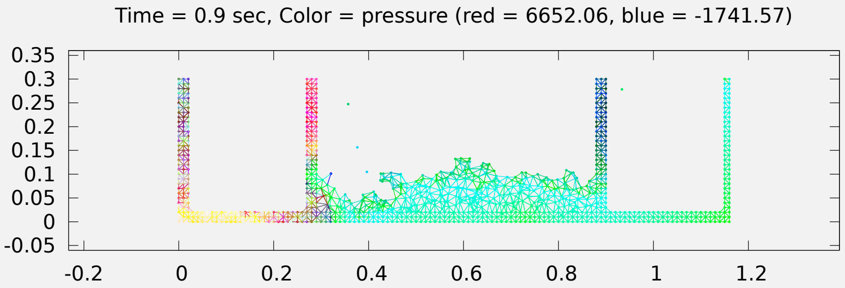
Water column collapse : pressure distribution



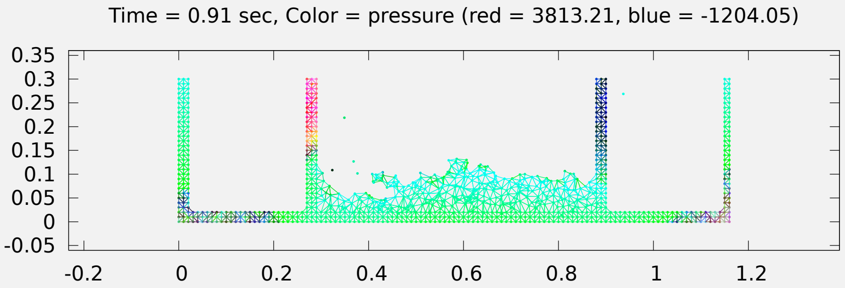
Water column collapse : pressure distribution



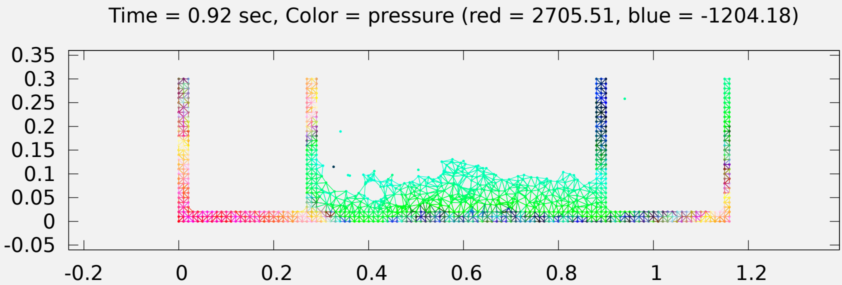
Water column collapse : pressure distribution



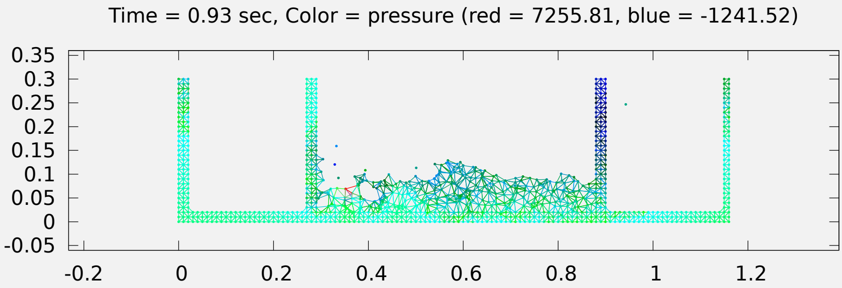
Water column collapse : pressure distribution



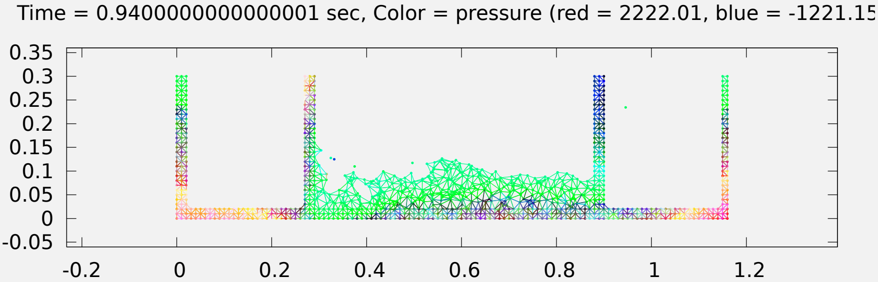
Water column collapse : pressure distribution



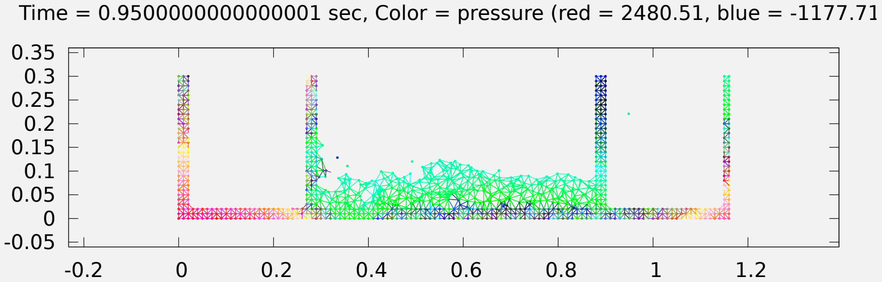
Water column collapse : pressure distribution



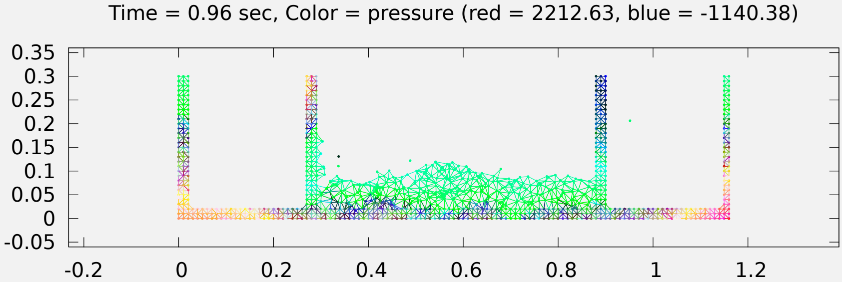
Water column collapse : pressure distribution



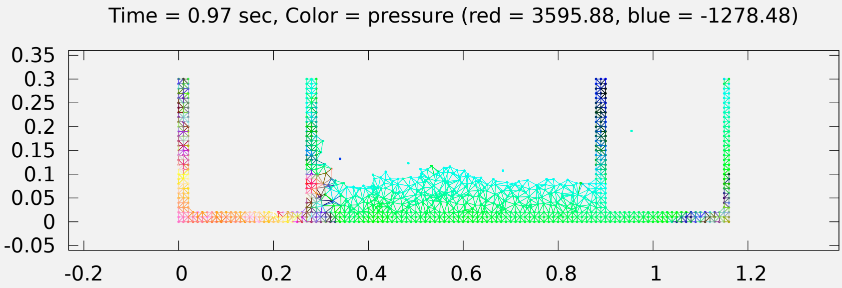
Water column collapse : pressure distribution



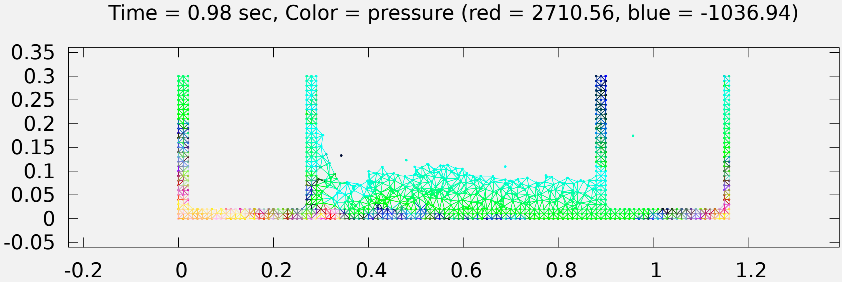
Water column collapse : pressure distribution



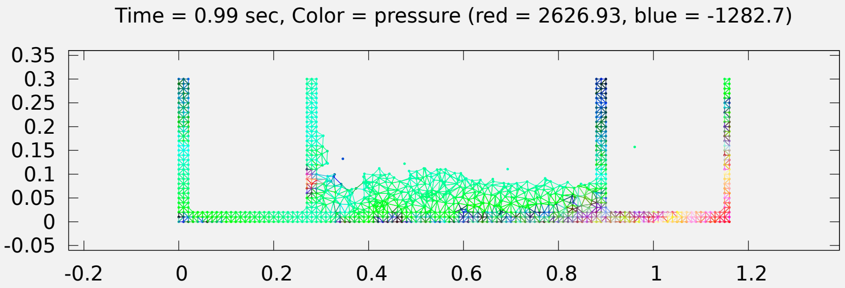
Water column collapse : pressure distribution



Water column collapse : pressure distribution



Water column collapse : pressure distribution



Mesh and Remesh

The mesh and remesh are based only on nodal positions and done by using the Delaunay Triangulation and Alpha Shape.

Mesh and Remesh

The mesh and remesh are based only on nodal positions and done by using the Delaunay Triangulation and Alpha Shape.

- Delaunay Triangulation of a set of nodes is a partition of the convex hull Ω of all the nodes into regions Ω_i such that $\Omega = \cup\Omega_i$. Each Ω_i is a triangle element defined by 3 nodes for 2D problem. The circumcircle of each triangle element contains no other node inside.

Mesh and Remesh

The mesh and remesh are based only on nodal positions and done by using the Delaunay Triangulation and Alpha Shape.

- Delaunay Triangulation of a set of nodes is a partition of the convex hull Ω of all the nodes into regions Ω_i such that $\Omega = \cup\Omega_i$. Each Ω_i is a triangle element defined by 3 nodes for 2D problem. The circumcircle of each triangle element contains no other node inside.
- Alpha Shape is a generalization of the convex hull of a set of nodes. Let α a real number with $0 \leq \alpha \leq \infty$. For $\alpha = \infty$, the Alpha Shape is identical to the Delaunay Triangulation. For $\alpha = 0$, the Alpha Shape shrinks to the set of nodes. Setting α to a appropriate value can remove the unnecessary triangles to recover the real boundaries.

Meshing in Tcl

A Tcl script `delaunayTri.tcl` is used to mesh the domain,

- In each time step, the current coordinates of all nodes are passed to procedure `delaunayTri`

Meshing in Tcl

A Tcl script `delaunayTri.tcl` is used to mesh the domain,

- In each time step, the current coordinates of all nodes are passed to procedure `delaunayTri`
- External program `qhull` is executed in Tcl to do the Delaunay Triangulation based on the nodal positions

Meshing in Tcl

A Tcl script `delaunayTri.tcl` is used to mesh the domain,

- In each time step, the current coordinates of all nodes are passed to procedure `delaunayTri`
- External program `qhull` is executed in Tcl to do the Delaunay Triangulation based on the nodal positions
- The area, radius, size of each triangle in the triangulation are calculated

Meshing in Tcl

A Tcl script `delaunayTri.tcl` is used to mesh the domain,

- In each time step, the current coordinates of all nodes are passed to procedure `delaunayTri`
- External program `qhull` is executed in Tcl to do the Delaunay Triangulation based on the nodal positions
- The area, radius, size of each triangle in the triangulation are calculated
- Alpha shape test is performed on each triangle. For triangles failed on the test are removed from the triangulation

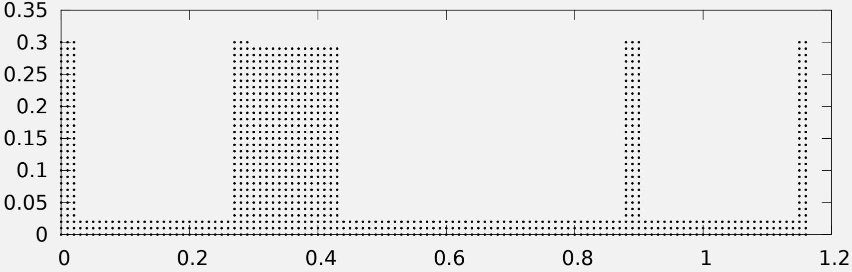
Meshing in Tcl

A Tcl script `delaunayTri.tcl` is used to mesh the domain,

- In each time step, the current coordinates of all nodes are passed to procedure `delaunayTri`
- External program `qhull` is executed in Tcl to do the Delaunay Triangulation based on the nodal positions
- The area, radius, size of each triangle in the triangulation are calculated
- Alpha shape test is performed on each triangle. For triangles failed on the test are removed from the triangulation
- Remaining triangles are elements to be created in OpenSees.

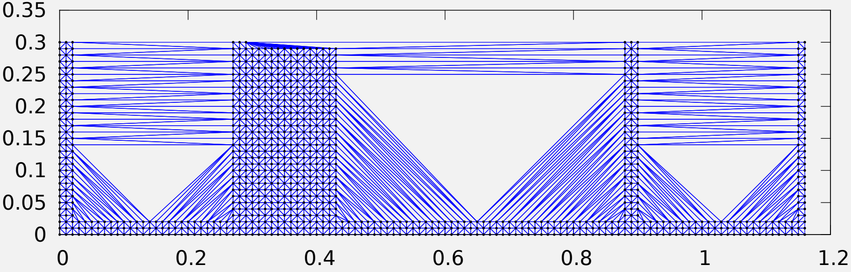
Example of meshing

At each time step, a set of nodes are given



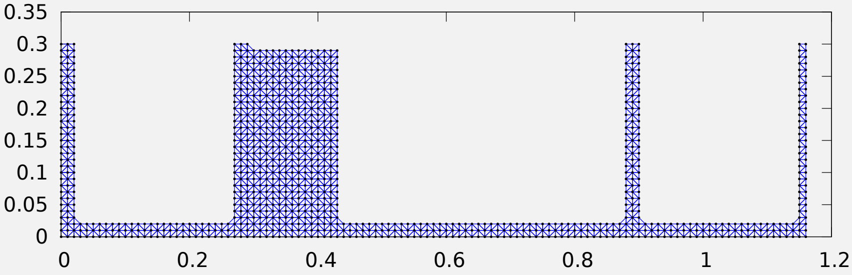
Example of meshing

Delaunay Triangulation of the set of nodes is calculated



Example of meshing

Alpha Shape of the set of nodes ($\alpha = 1.4$) is calculated



Outline

Sensitivity

Definition

The effects of assumed parameter values on computed response

Sensitivity

Subsequent applications of sensitivity include

- Reliability analysis
  - Quantify system performance under multiple limit state
- Optimization
  - Find best solution for some objective functions and constraints on the system
- System identification
  - Inverse modeling, find model parameters that match observed results
- and others ...

Sensitivity

Simply speaking, a sensitivity is a gradient,

$$\frac{\partial U}{\partial \theta}(x, y, z, t)$$

where,

Sensitivity

Simply speaking, a sensitivity is a gradient,

$$\frac{\partial U}{\partial \theta}(x, y, z, t)$$

where,

- U – response velocity, pressure, ...

Sensitivity

Simply speaking, a sensitivity is a gradient,

$$\frac{\partial U}{\partial \theta}(x, y, z, t)$$

where,

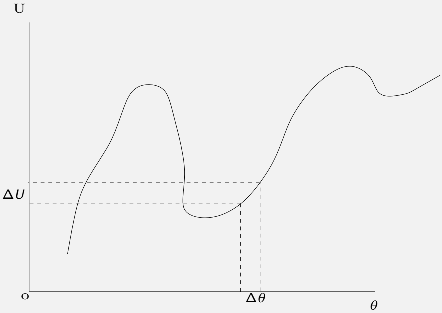
- U – response velocity, pressure, ...
- θ – modeling or physical property

Computing sensitivity

- A simple way (Finite Difference Method)

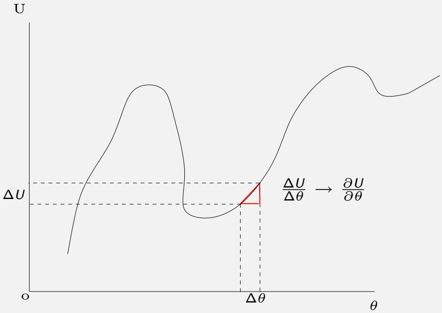
Computing sensitivity

- A simple way (Finite Difference Method)



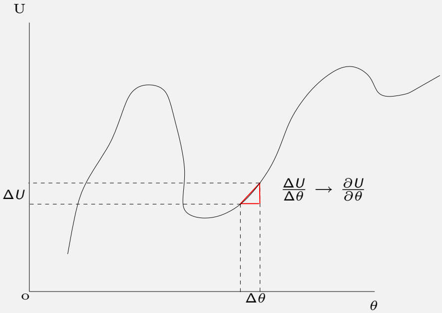
Computing sensitivity

- A simple way (Finite Difference Method)



Computing sensitivity

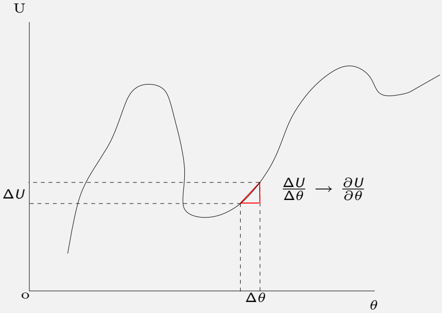
- A simple way (Finite Difference Method)



- Require to solve response equations one more time for each parameter
- Prone to round-off error for small $\Delta\theta$

Computing sensitivity

- A simple way (Finite Difference Method)



- Require to solve response equations one more time for each parameter
- Prone to round-off error for small $\Delta\theta$

FDM is not practical due to the computational cost

Computing sensitivity

- A direct way (Direct Differentiation Method – DDM)

Computing sensitivity

- A direct way (Direct Differentiation Method – DDM)

$$\mathbf{M}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}$$

Computing sensitivity

- A direct way (Direct Differentiation Method – DDM)

$$\mathbf{M}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}$$

$$\frac{\partial \mathbf{M}}{\partial \theta}\dot{\mathbf{U}} + \mathbf{M}\frac{\partial \dot{\mathbf{U}}}{\partial \theta} + \frac{\partial \mathbf{K}}{\partial \theta}\mathbf{U} + \mathbf{K}\frac{\partial \mathbf{U}}{\partial \theta} = \frac{\partial \mathbf{F}}{\partial \theta}$$

Computing sensitivity

- A direct way (Direct Differentiation Method – DDM)

$$\mathbf{M}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}$$

$$\frac{\partial \mathbf{M}}{\partial \theta}\dot{\mathbf{U}} + \mathbf{M}\frac{\partial \dot{\mathbf{U}}}{\partial \theta} + \frac{\partial \mathbf{K}}{\partial \theta}\mathbf{U} + \mathbf{K}\frac{\partial \mathbf{U}}{\partial \theta} = \frac{\partial \mathbf{F}}{\partial \theta}$$

$$\mathbf{M}\frac{\partial \dot{\mathbf{U}}}{\partial \theta} + \mathbf{K}\frac{\partial \mathbf{U}}{\partial \theta} = \frac{\partial \mathbf{F}}{\partial \theta} - \frac{\partial \mathbf{M}}{\partial \theta}\dot{\mathbf{U}} - \frac{\partial \mathbf{K}}{\partial \theta}\mathbf{U}$$

Computing sensitivity

- A direct way (Direct Differentiation Method – DDM)

$$\mathbf{M}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}$$

$$\frac{\partial \mathbf{M}}{\partial \theta}\dot{\mathbf{U}} + \mathbf{M}\frac{\partial \dot{\mathbf{U}}}{\partial \theta} + \frac{\partial \mathbf{K}}{\partial \theta}\mathbf{U} + \mathbf{K}\frac{\partial \mathbf{U}}{\partial \theta} = \frac{\partial \mathbf{F}}{\partial \theta}$$

$$\mathbf{M}\frac{\partial \dot{\mathbf{U}}}{\partial \theta} + \mathbf{K}\frac{\partial \mathbf{U}}{\partial \theta} = \frac{\partial \mathbf{F}}{\partial \theta} - \frac{\partial \mathbf{M}}{\partial \theta}\dot{\mathbf{U}} - \frac{\partial \mathbf{K}}{\partial \theta}\mathbf{U}$$

Computing sensitivity

- A direct way (Direct Differentiation Method – DDM)

$$\mathbf{M}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F}$$

$$\frac{\partial \mathbf{M}}{\partial \theta}\dot{\mathbf{U}} + \mathbf{M}\frac{\partial \dot{\mathbf{U}}}{\partial \theta} + \frac{\partial \mathbf{K}}{\partial \theta}\mathbf{U} + \mathbf{K}\frac{\partial \mathbf{U}}{\partial \theta} = \frac{\partial \mathbf{F}}{\partial \theta}$$

$$\mathbf{M}\frac{\partial \dot{\mathbf{U}}}{\partial \theta} + \mathbf{K}\frac{\partial \mathbf{U}}{\partial \theta} = \frac{\partial \mathbf{F}}{\partial \theta} - \frac{\partial \mathbf{M}}{\partial \theta}\dot{\mathbf{U}} - \frac{\partial \mathbf{K}}{\partial \theta}\mathbf{U}$$

where,

$$\frac{\partial \mathbf{F}}{\partial \theta} - \frac{\partial \mathbf{M}}{\partial \theta}\dot{\mathbf{U}} - \frac{\partial \mathbf{K}}{\partial \theta}\mathbf{U}$$

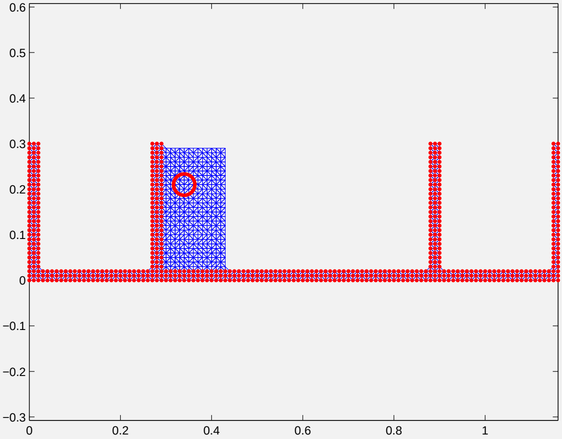
is the sensitivity "force" vector

- Only form RHS vectors and solve linear equations once for each parameter, LHS matrices don't change,
- No additional round-off error as the FDM

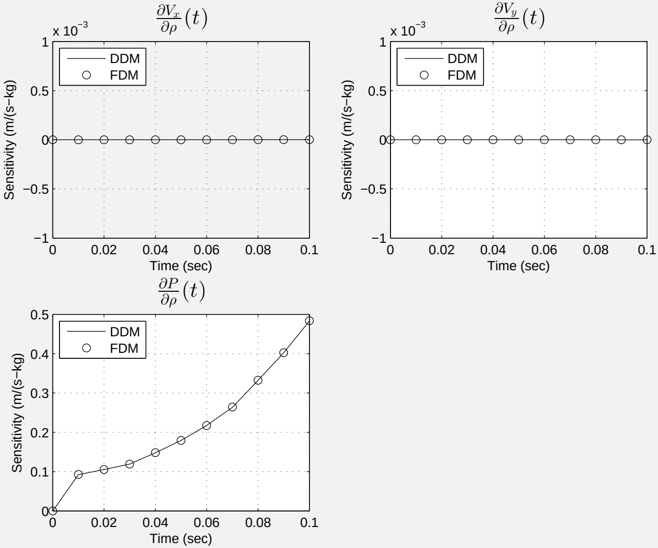
Example of sensitivity analysis

Example of sensitivity analysis

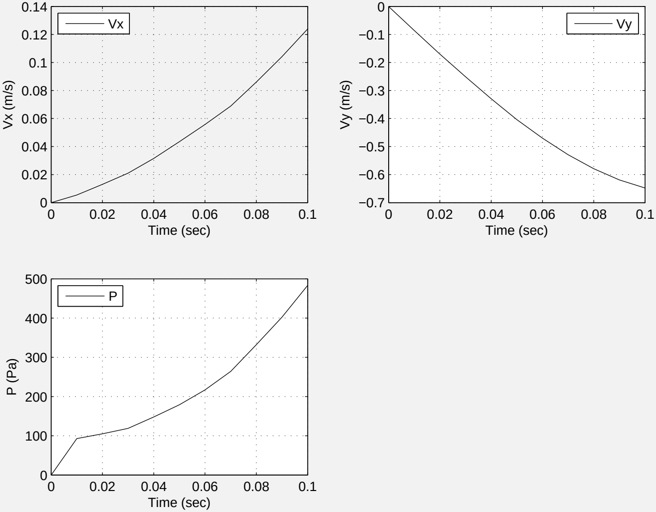
- Find $\frac{\partial V_x}{\partial \rho}(t)$, $\frac{\partial V_y}{\partial \rho}(t)$, $\frac{\partial P}{\partial \rho}(t)$ for particle 400, the sensitivity of velocities and pressure to the density of the fluid.



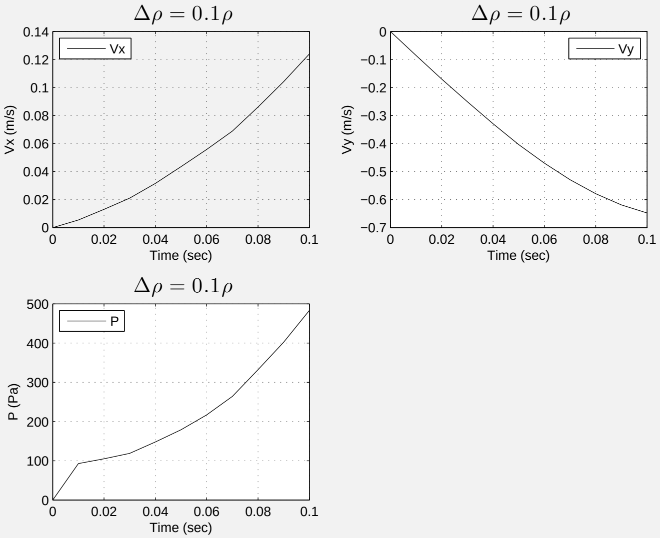
Example of sensitivity analysis



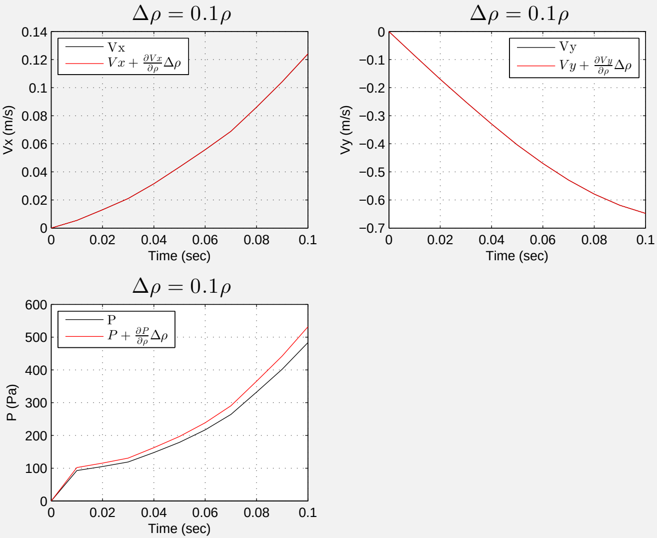
Example of sensitivity analysis



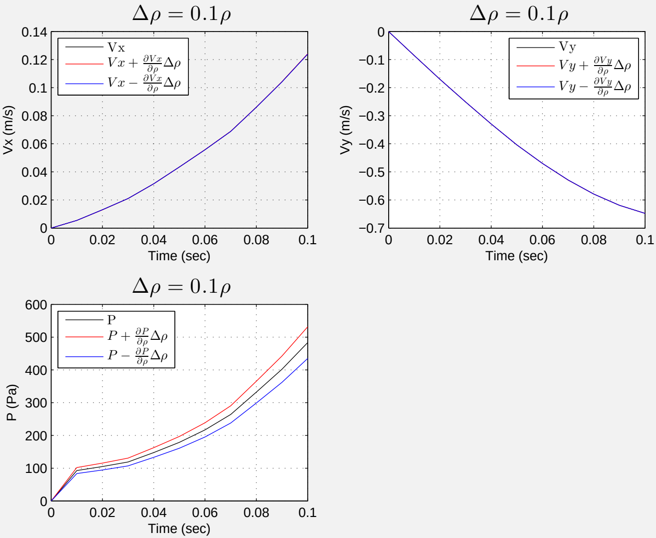
Example of sensitivity analysis



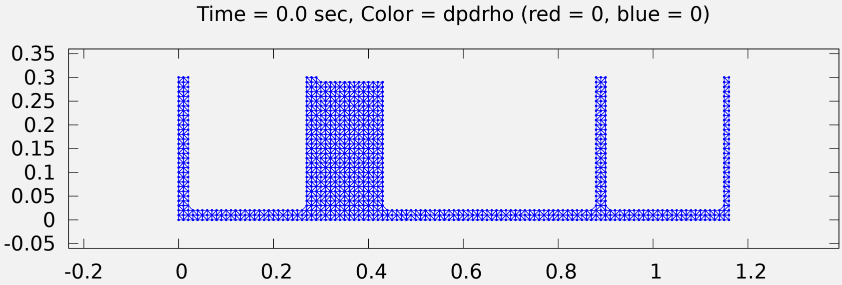
Example of sensitivity analysis



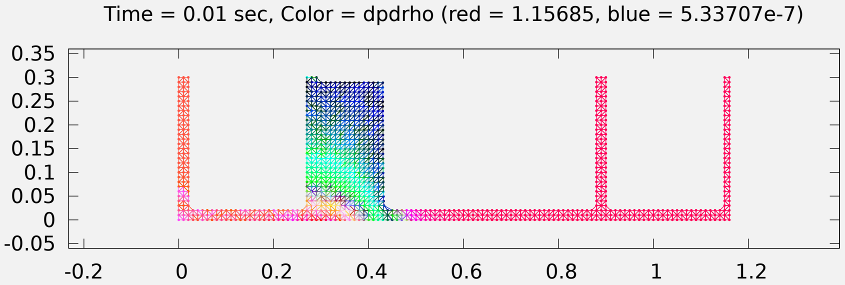
Example of sensitivity analysis



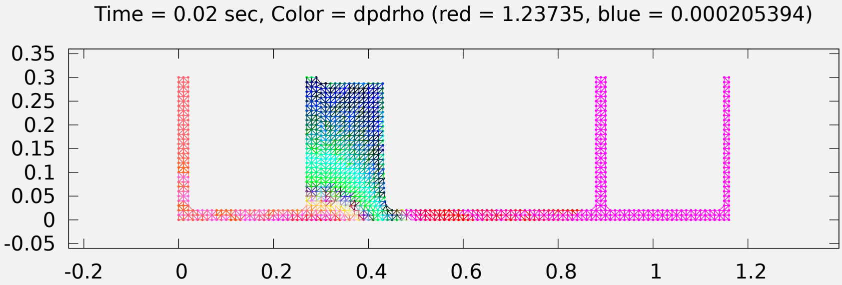
Water column collapse : pressure sensitivity distribution



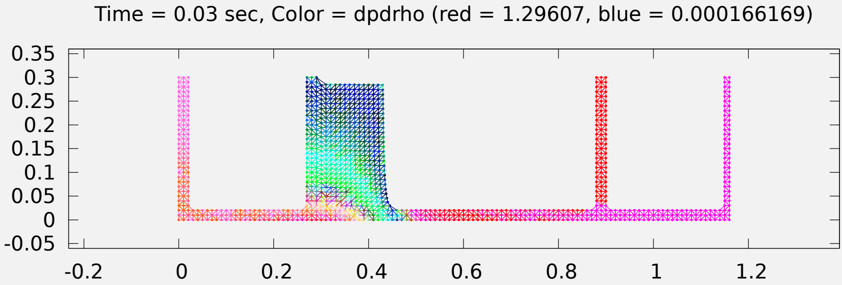
Water column collapse : pressure sensitivity distribution



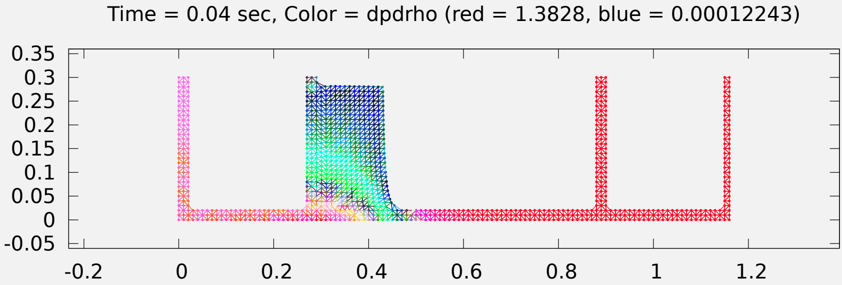
Water column collapse : pressure sensitivity distribution



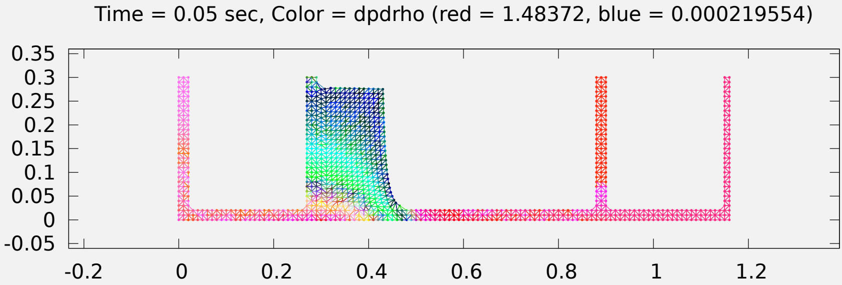
Water column collapse : pressure sensitivity distribution



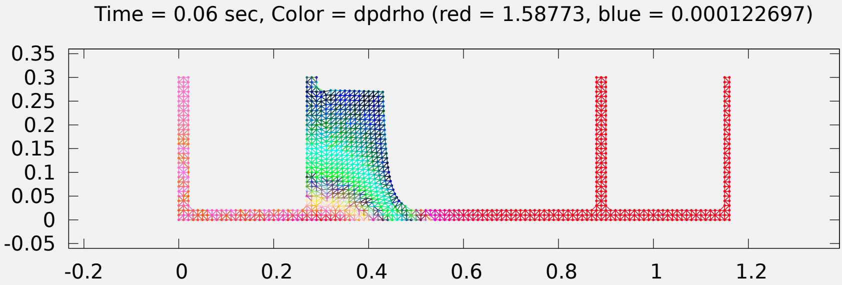
Water column collapse : pressure sensitivity distribution



Water column collapse : pressure sensitivity distribution

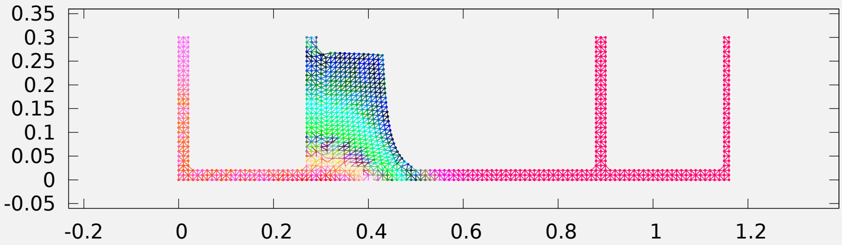


Water column collapse : pressure sensitivity distribution

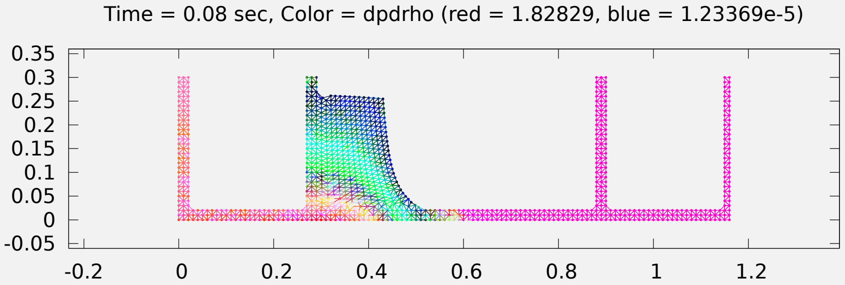


Water column collapse : pressure sensitivity distribution

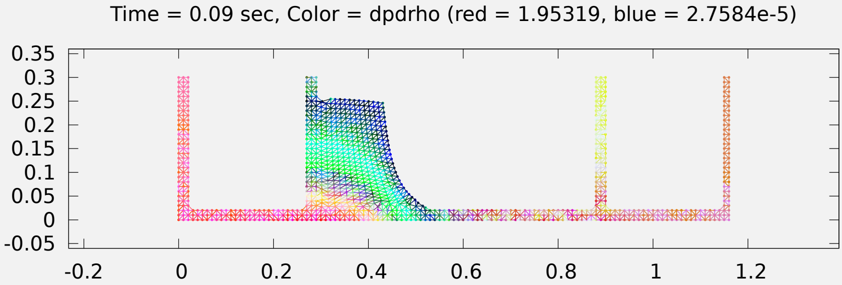
Time = 0.07 sec, Color = dpdrho (red = 1.70172, blue = 6.04305e-5)



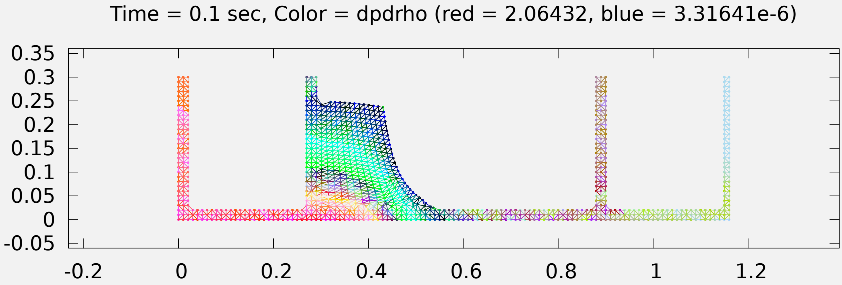
Water column collapse : pressure sensitivity distribution



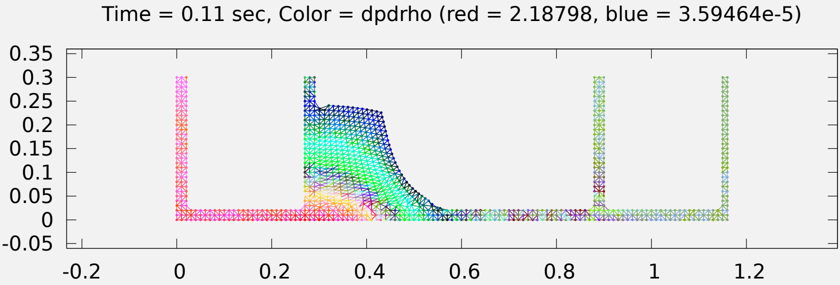
Water column collapse : pressure sensitivity distribution



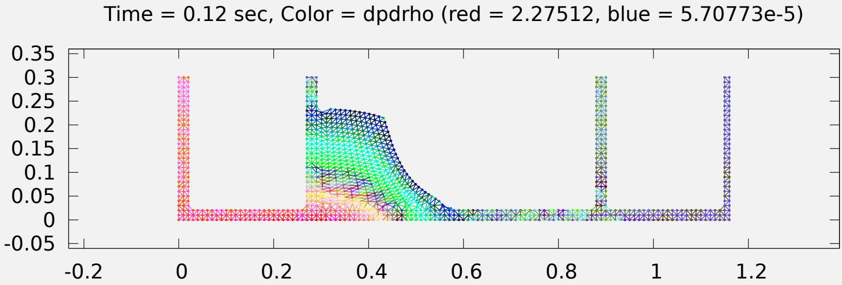
Water column collapse : pressure sensitivity distribution



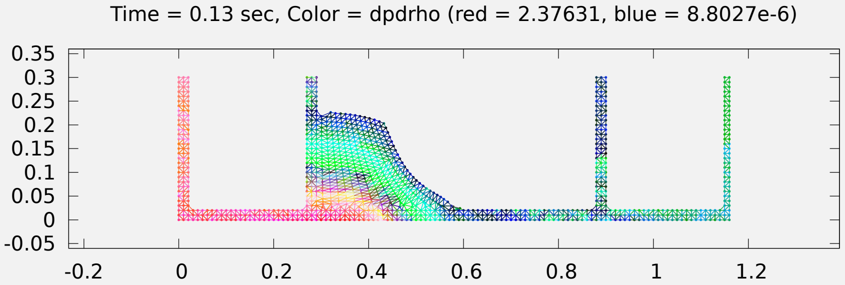
Water column collapse : pressure sensitivity distribution



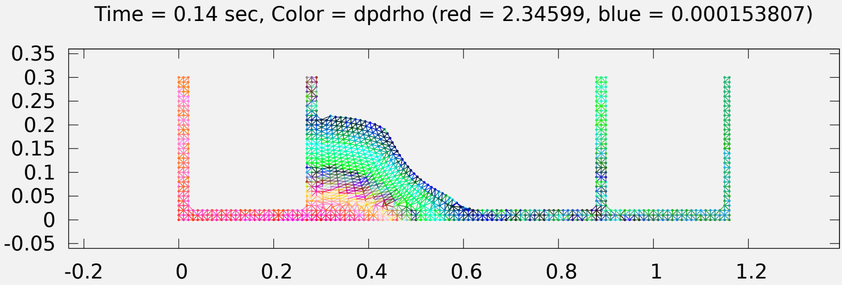
Water column collapse : pressure sensitivity distribution



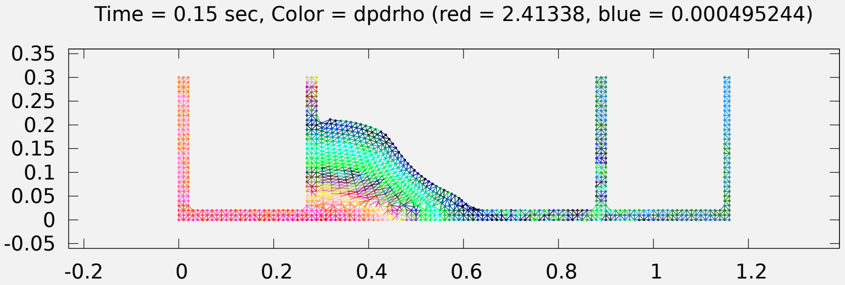
Water column collapse : pressure sensitivity distribution



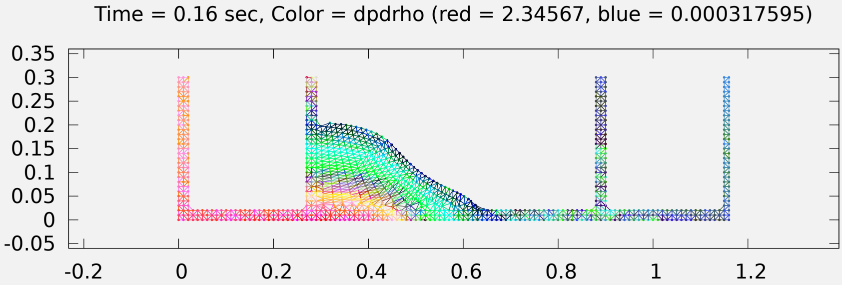
Water column collapse : pressure sensitivity distribution



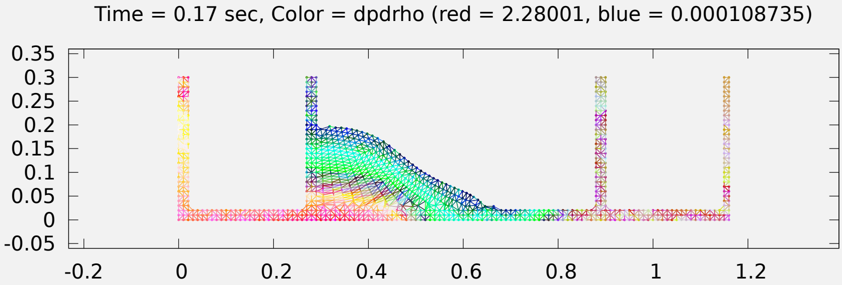
Water column collapse : pressure sensitivity distribution



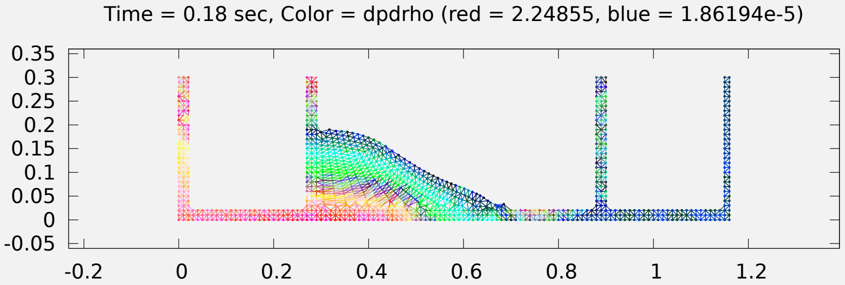
Water column collapse : pressure sensitivity distribution



Water column collapse : pressure sensitivity distribution

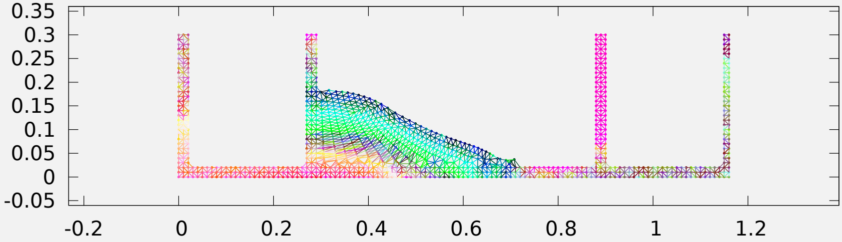


Water column collapse : pressure sensitivity distribution

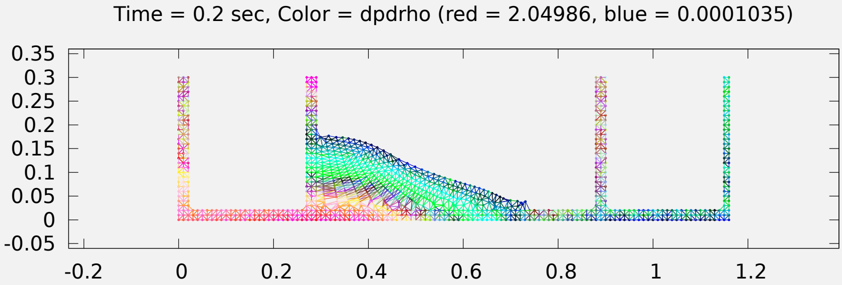


Water column collapse : pressure sensitivity distribution

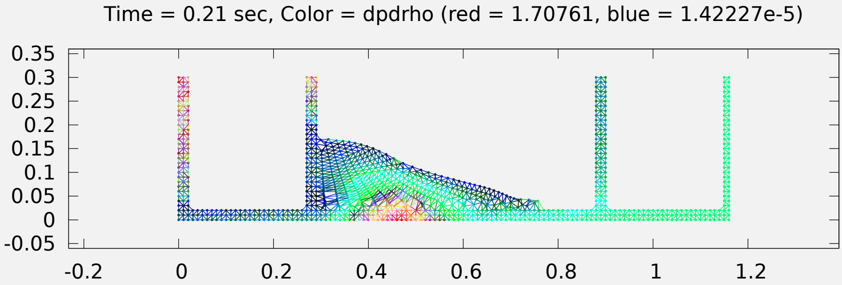
Time = 0.19 sec, Color = dpdrho (red = 2.13937, blue = 3.12969e-6)



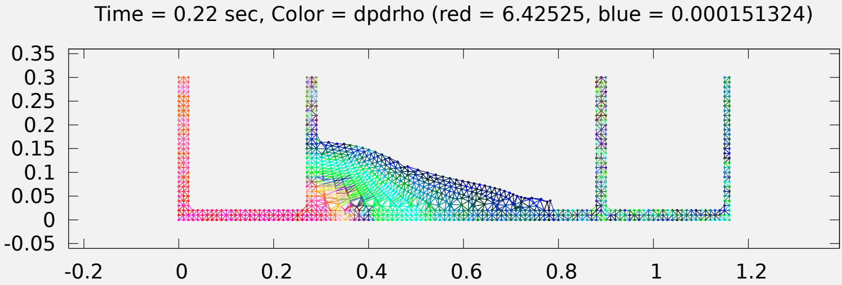
Water column collapse : pressure sensitivity distribution



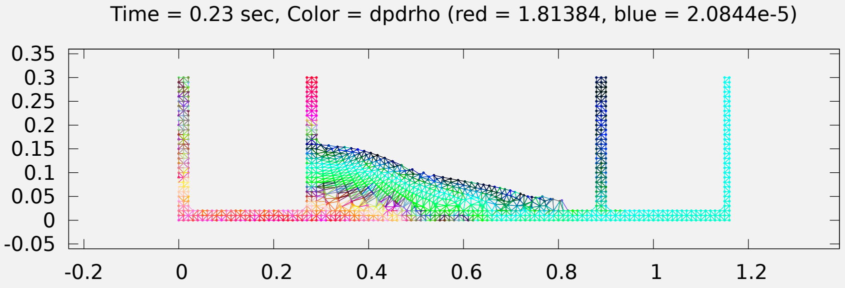
Water column collapse : pressure sensitivity distribution



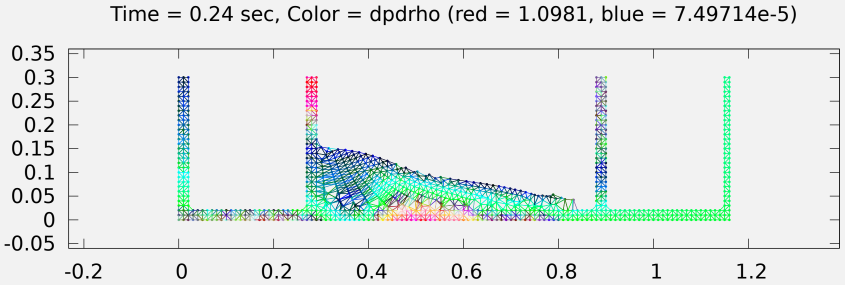
Water column collapse : pressure sensitivity distribution



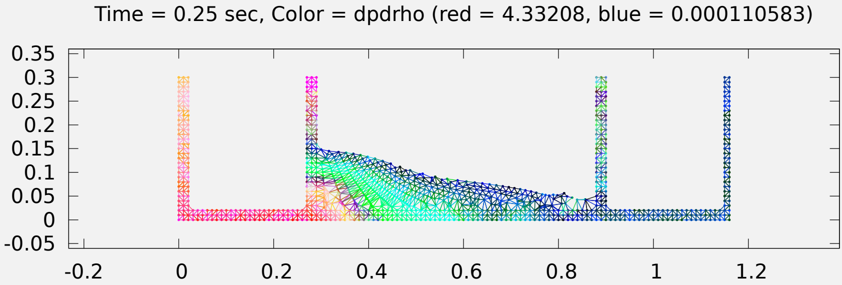
Water column collapse : pressure sensitivity distribution



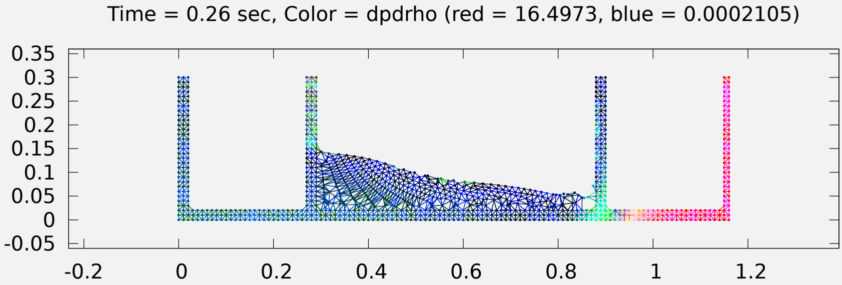
Water column collapse : pressure sensitivity distribution



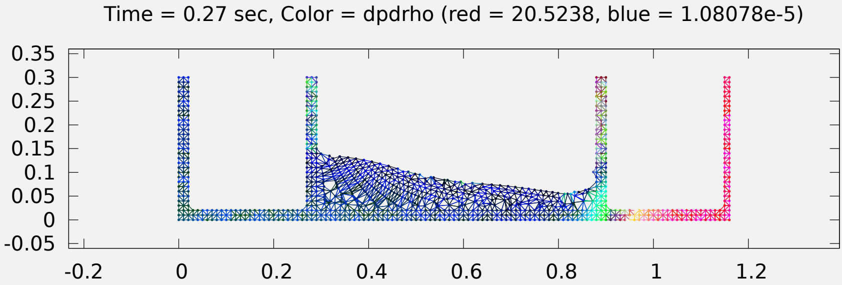
Water column collapse : pressure sensitivity distribution



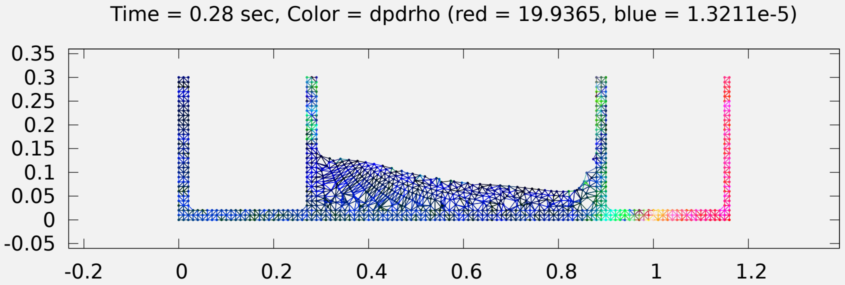
Water column collapse : pressure sensitivity distribution



Water column collapse : pressure sensitivity distribution

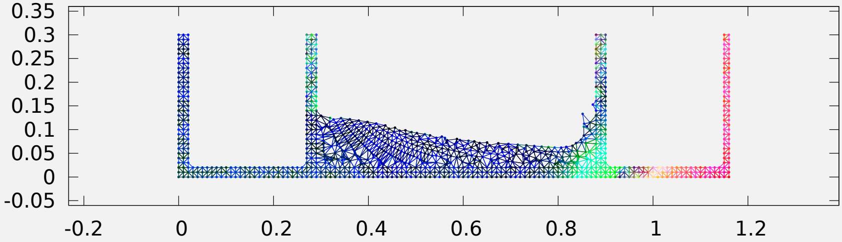


Water column collapse : pressure sensitivity distribution

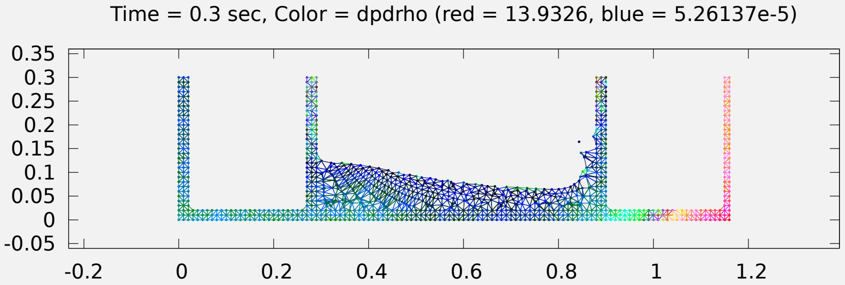


Water column collapse : pressure sensitivity distribution

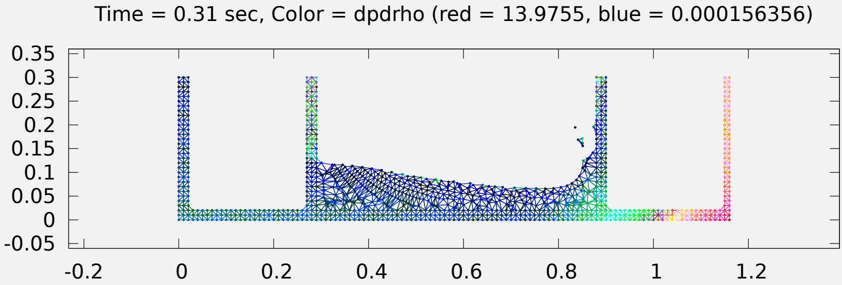
Time = 0.29 sec, Color = dpdrho (red = 22.4636, blue = 1.62874e-5)



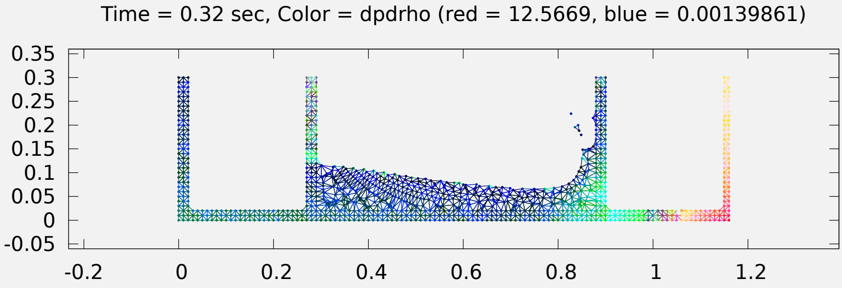
Water column collapse : pressure sensitivity distribution



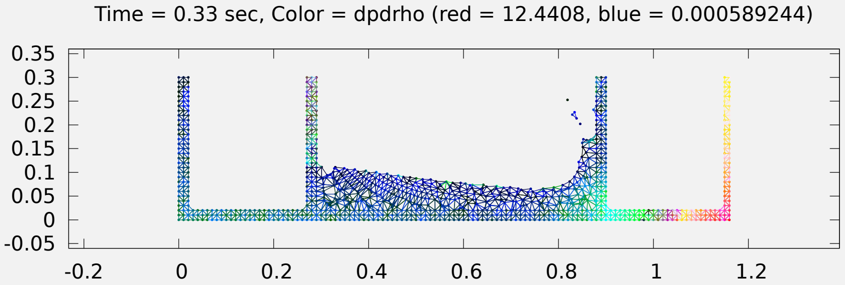
Water column collapse : pressure sensitivity distribution



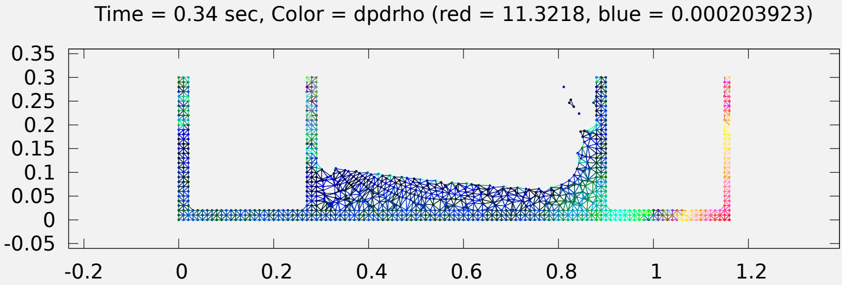
Water column collapse : pressure sensitivity distribution



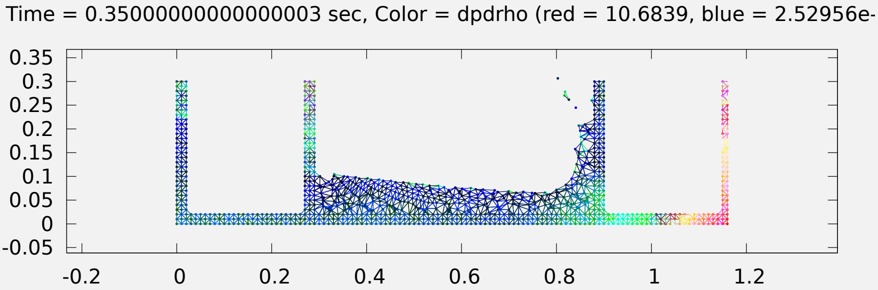
Water column collapse : pressure sensitivity distribution



Water column collapse : pressure sensitivity distribution

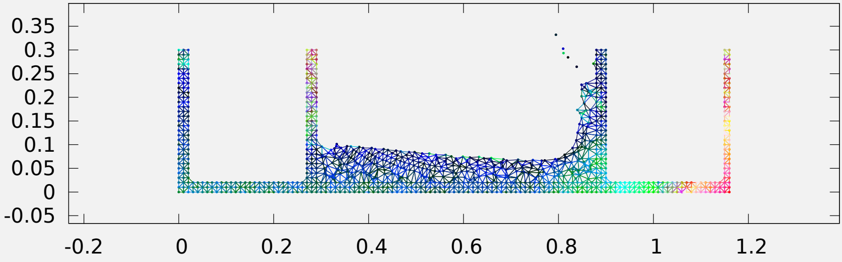


Water column collapse : pressure sensitivity distribution

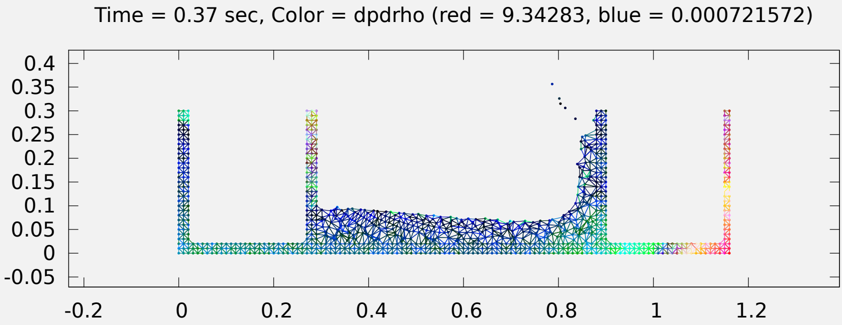


Water column collapse : pressure sensitivity distribution

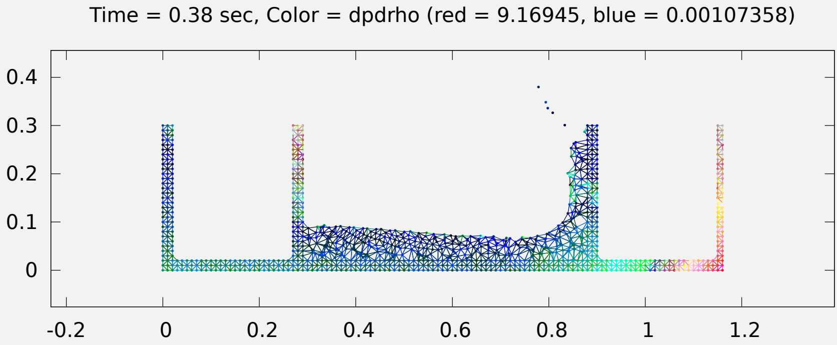
Time = 0.36 sec, Color = dpdrho (red = 9.76745, blue = 7.54104e-5)



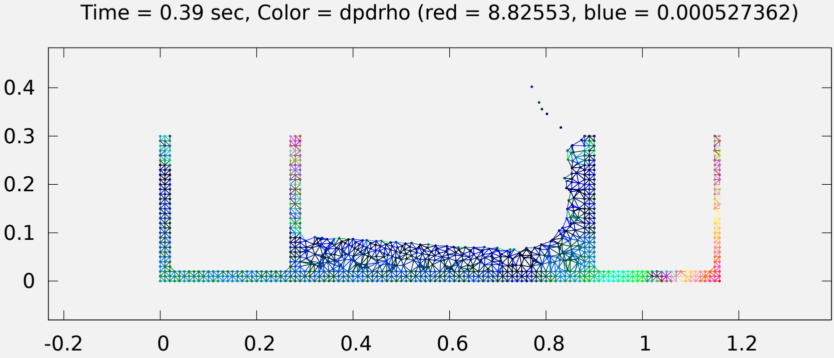
Water column collapse : pressure sensitivity distribution



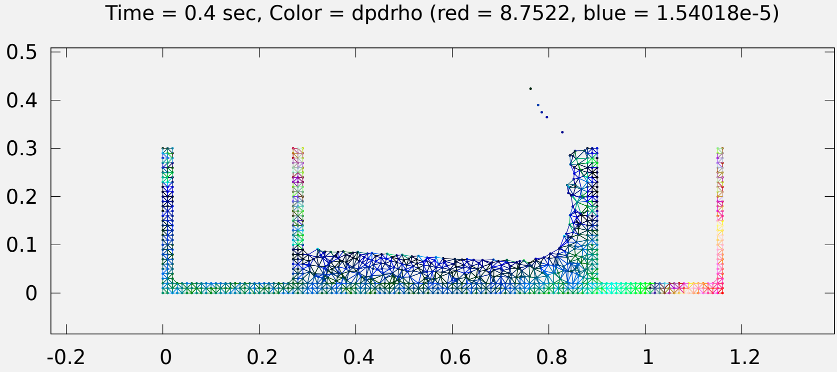
Water column collapse : pressure sensitivity distribution



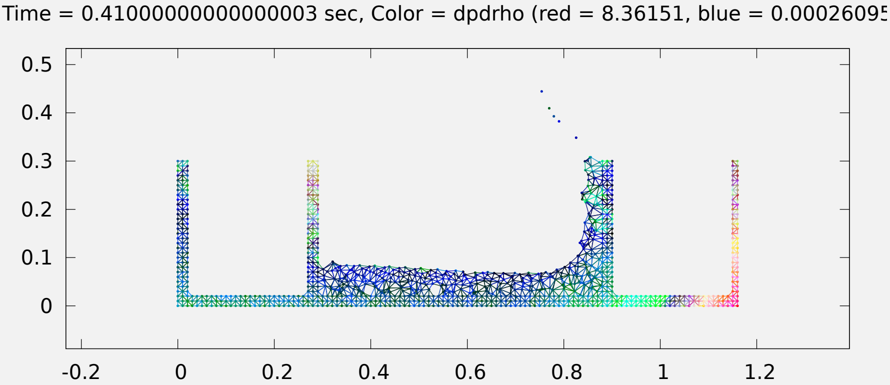
Water column collapse : pressure sensitivity distribution



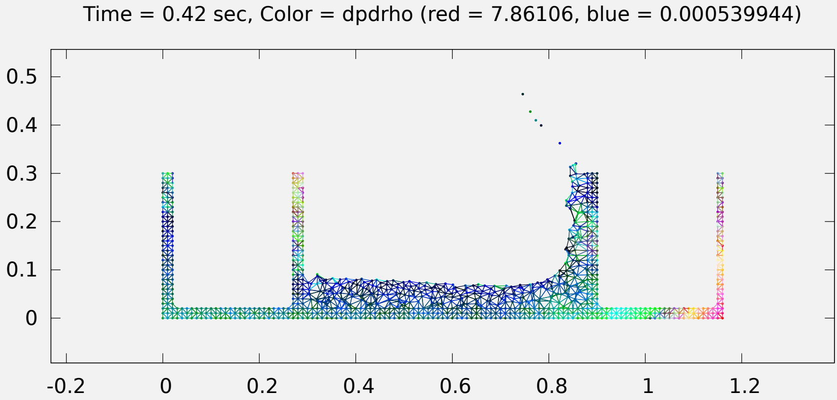
Water column collapse : pressure sensitivity distribution



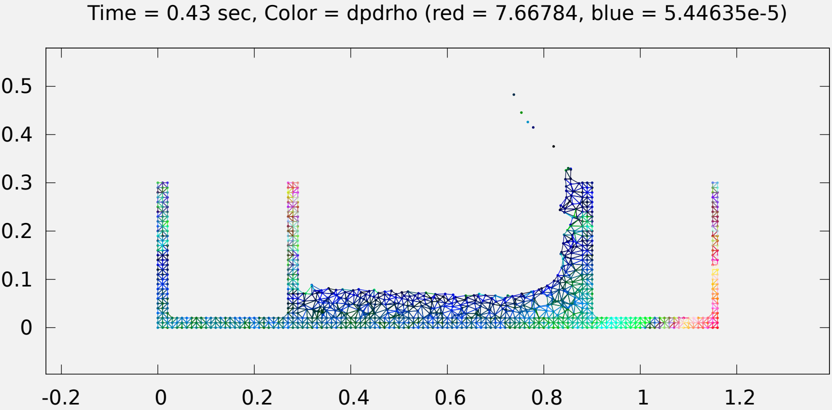
Water column collapse : pressure sensitivity distribution



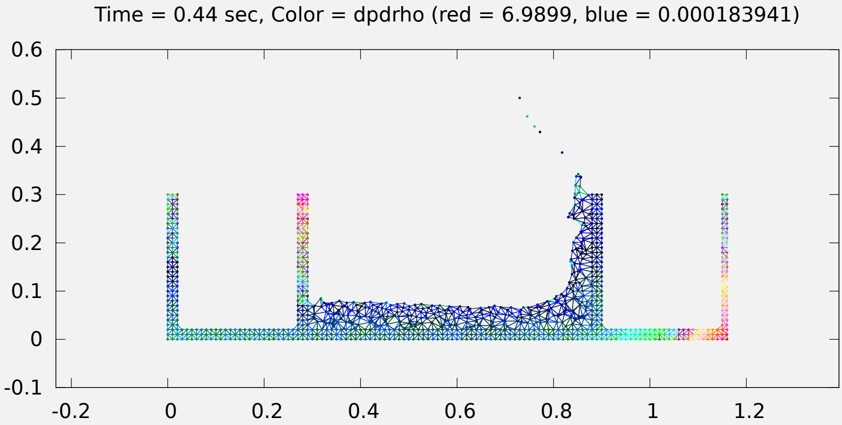
Water column collapse : pressure sensitivity distribution



Water column collapse : pressure sensitivity distribution

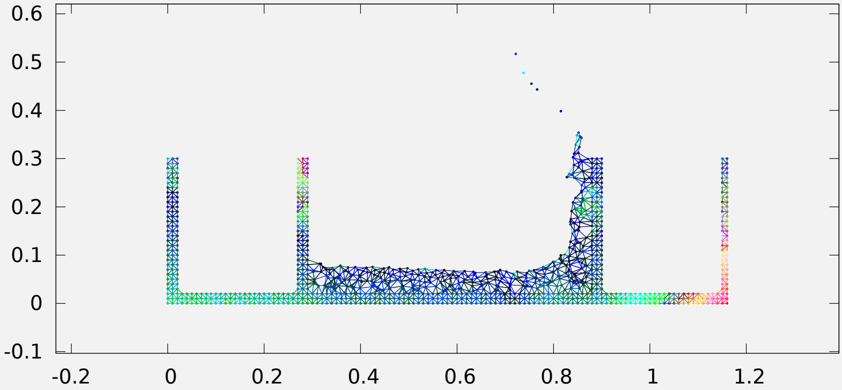


Water column collapse : pressure sensitivity distribution

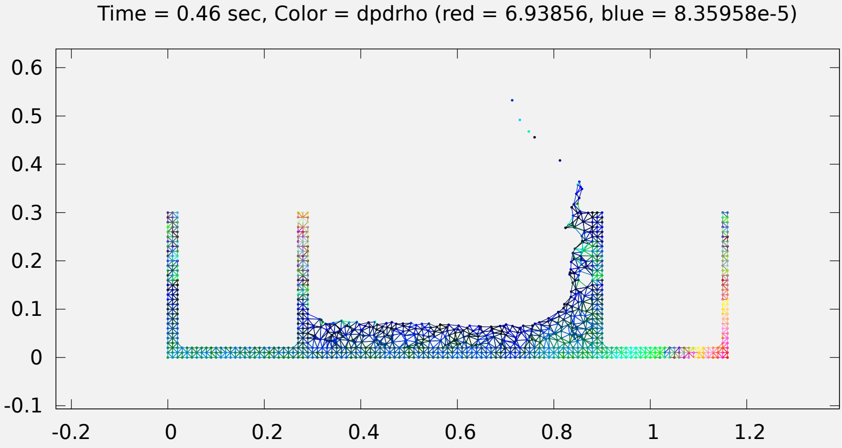


Water column collapse : pressure sensitivity distribution

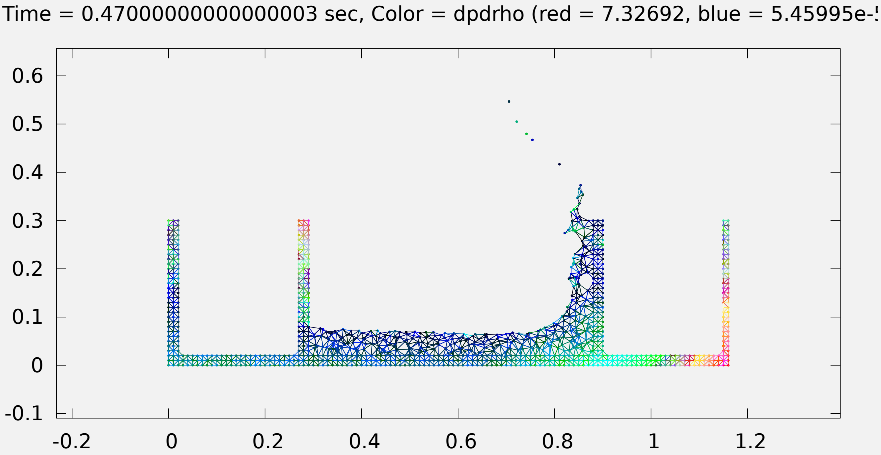
Time = 0.45 sec, Color = dpdrho (red = 6.6227, blue = 4.4567e-5)



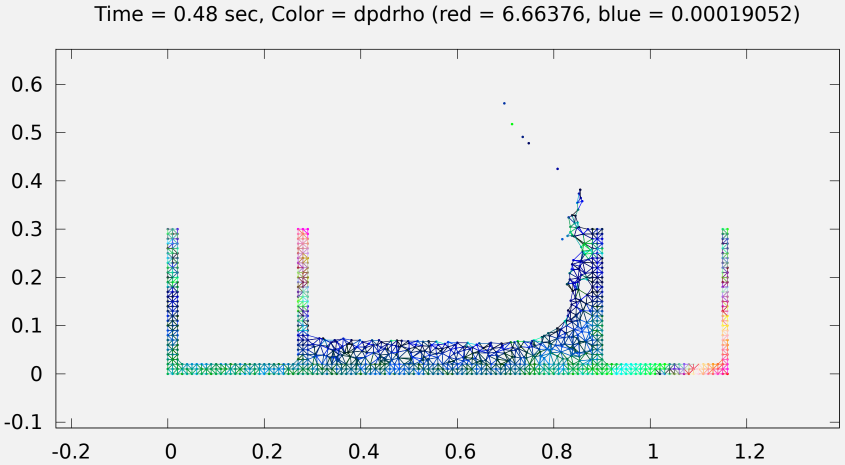
Water column collapse : pressure sensitivity distribution



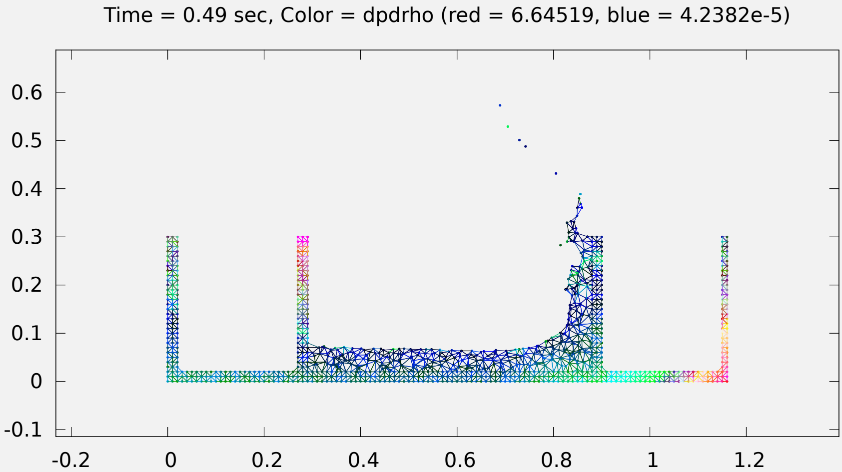
Water column collapse : pressure sensitivity distribution



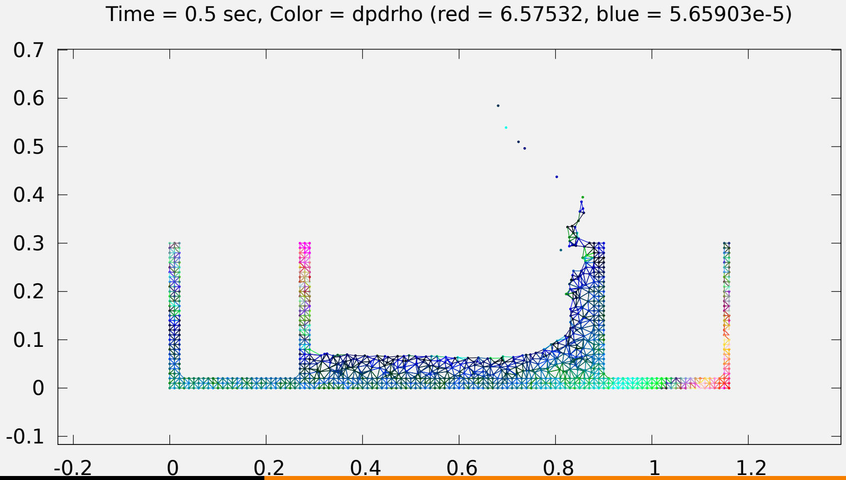
Water column collapse : pressure sensitivity distribution



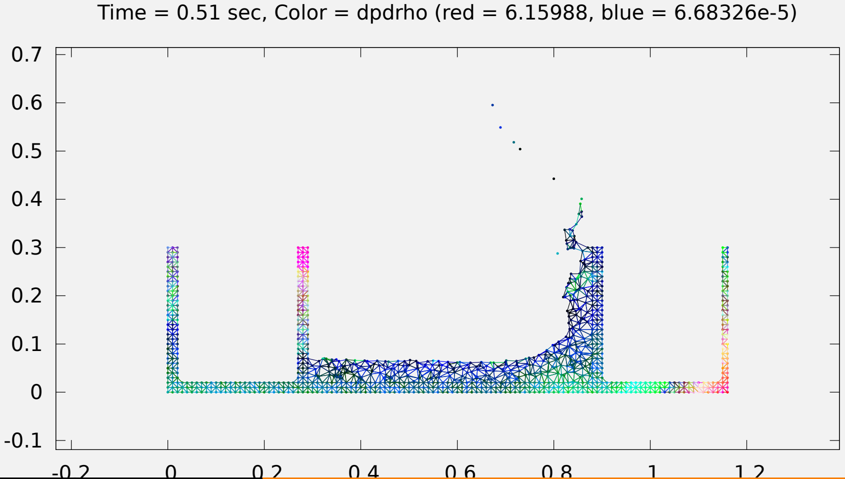
Water column collapse : pressure sensitivity distribution



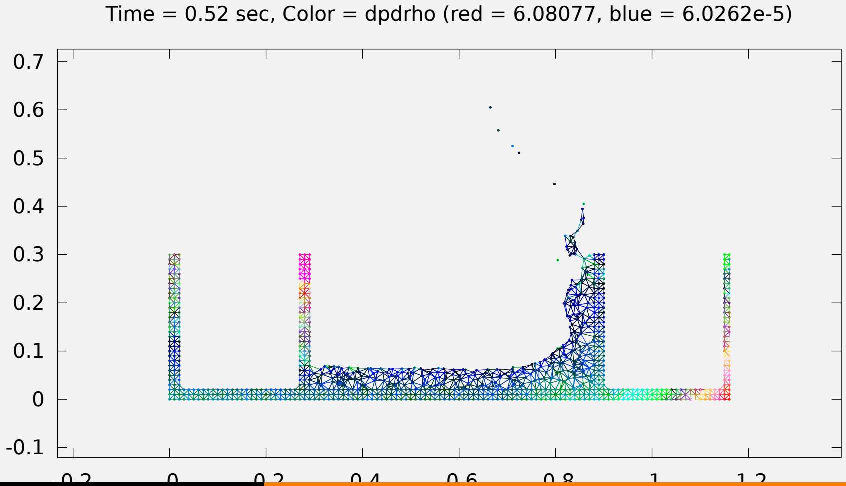
Water column collapse : pressure sensitivity distribution



Water column collapse : pressure sensitivity distribution

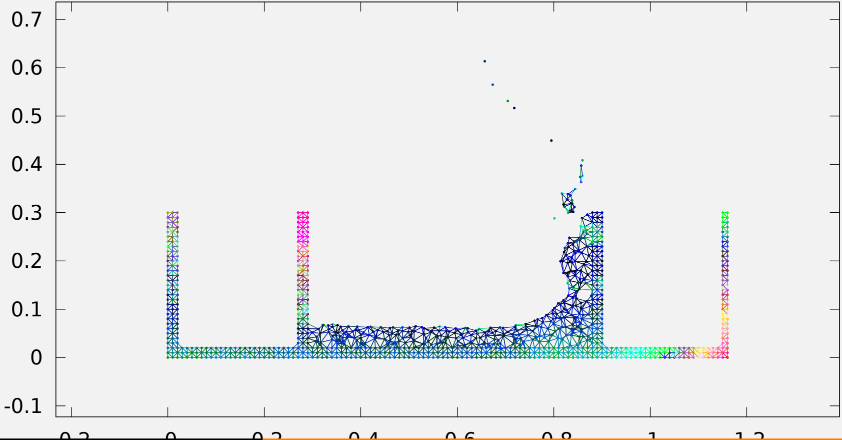


Water column collapse : pressure sensitivity distribution

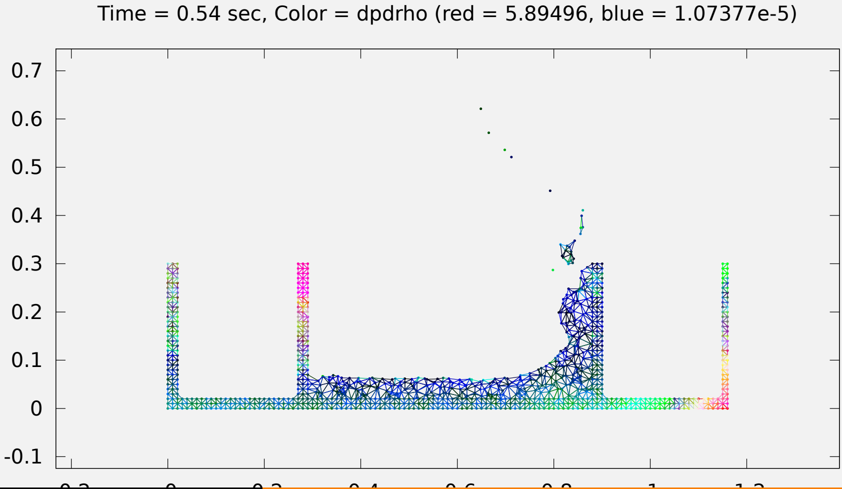


Water column collapse : pressure sensitivity distribution

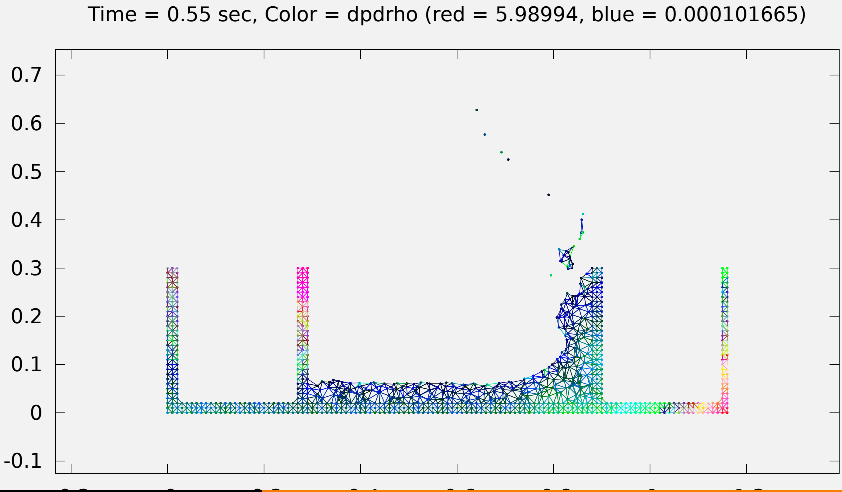
Time = 0.53 sec, Color = dpdrho (red = 5.95879, blue = 7.33743e-5)



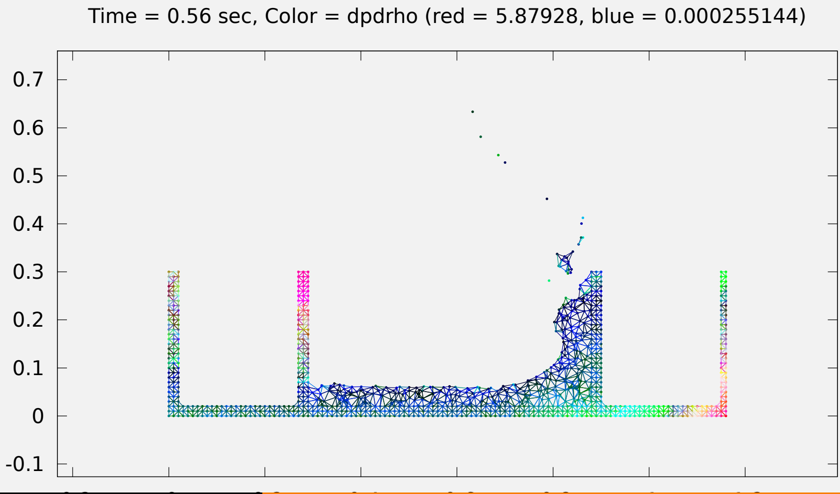
Water column collapse : pressure sensitivity distribution



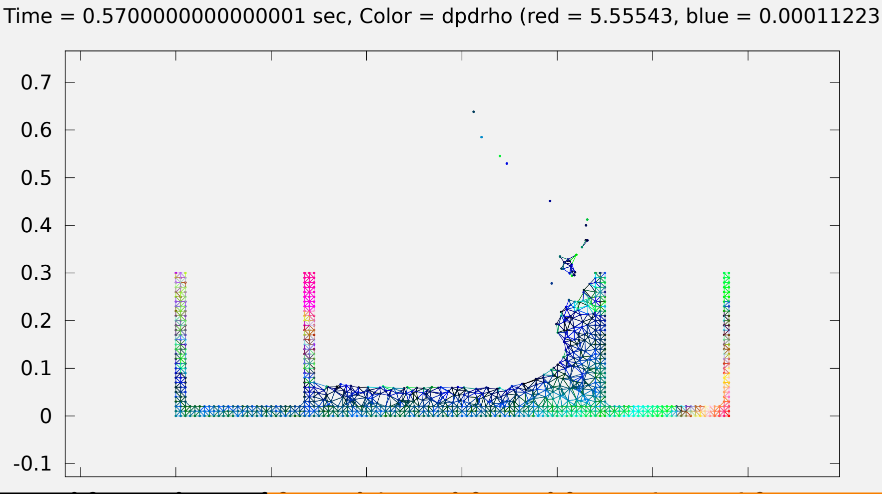
Water column collapse : pressure sensitivity distribution



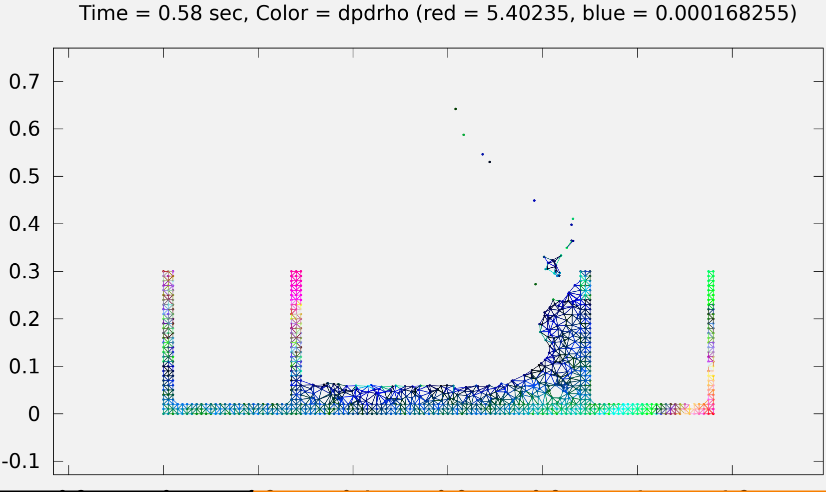
Water column collapse : pressure sensitivity distribution



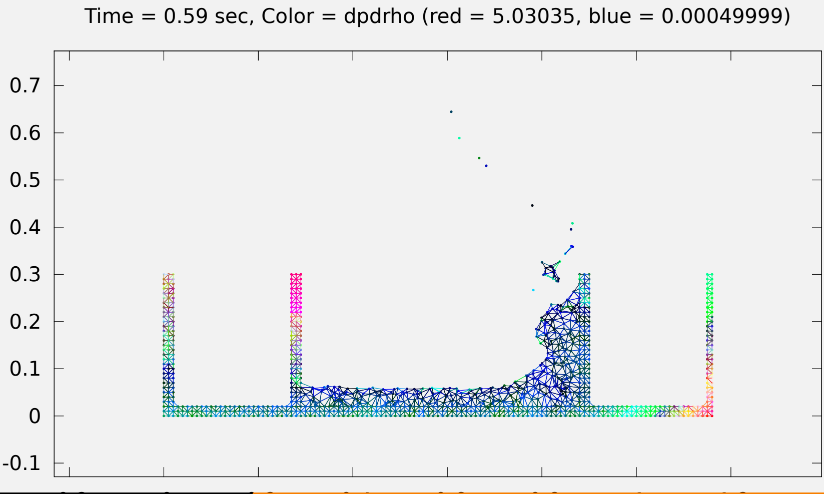
Water column collapse : pressure sensitivity distribution



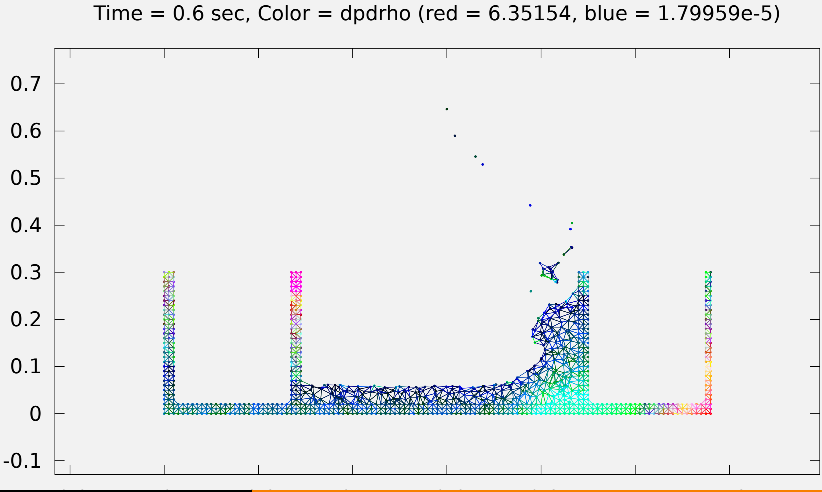
Water column collapse : pressure sensitivity distribution



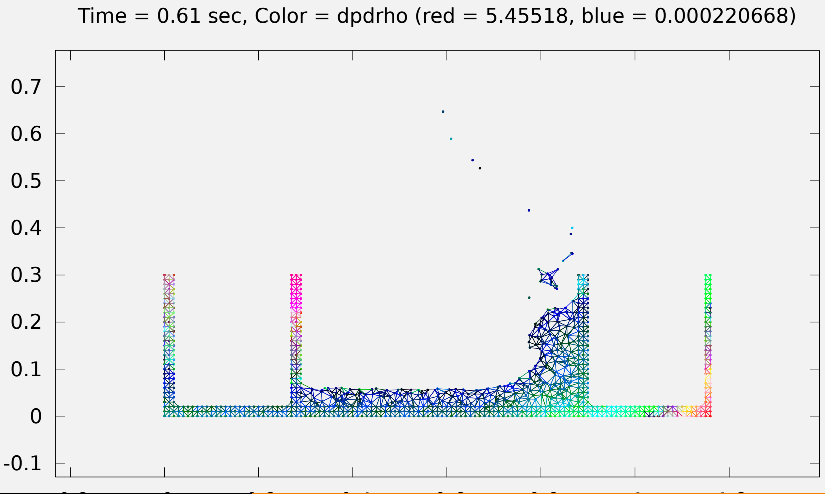
Water column collapse : pressure sensitivity distribution



Water column collapse : pressure sensitivity distribution

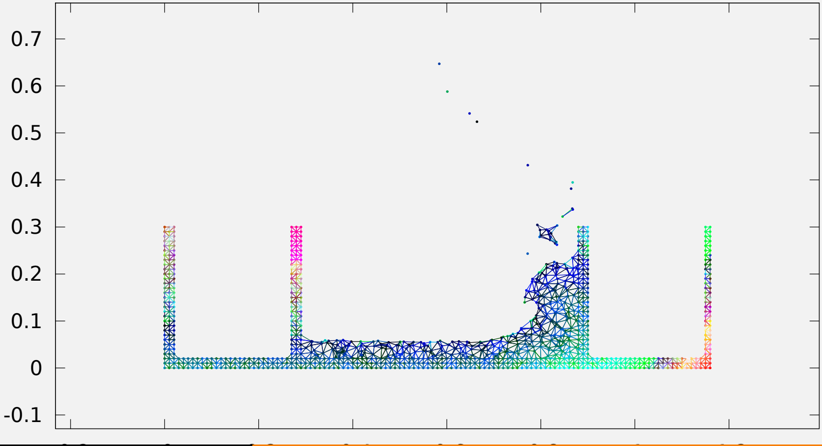


Water column collapse : pressure sensitivity distribution

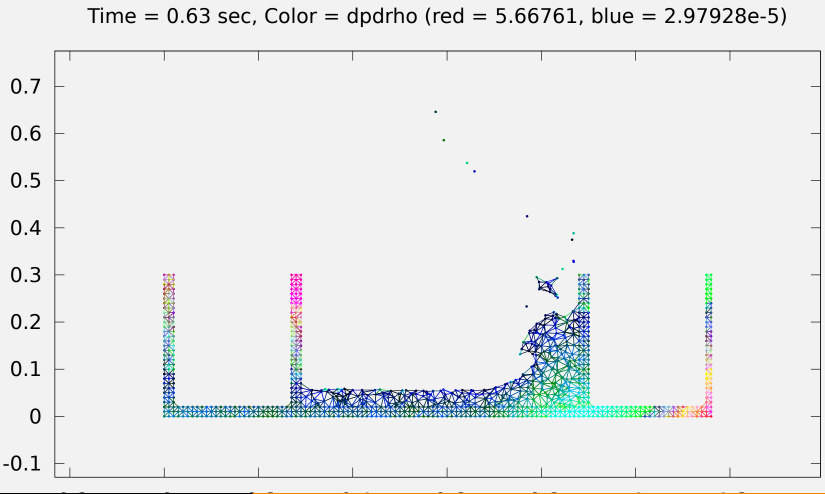


Water column collapse : pressure sensitivity distribution

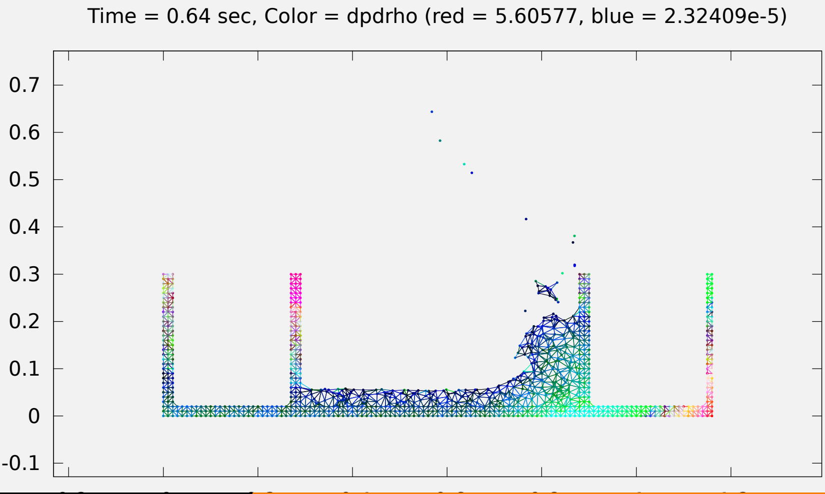
Time = 0.62 sec, Color = dpdrho (red = 5.43307, blue = 0.000260069)



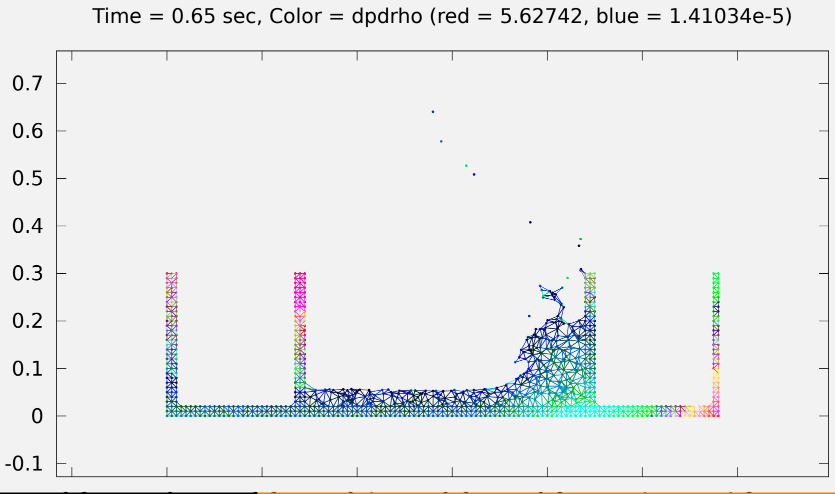
Water column collapse : pressure sensitivity distribution



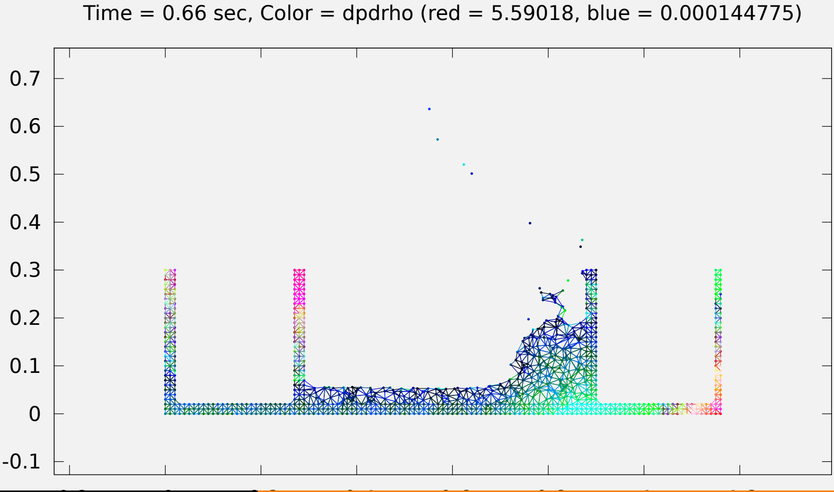
Water column collapse : pressure sensitivity distribution



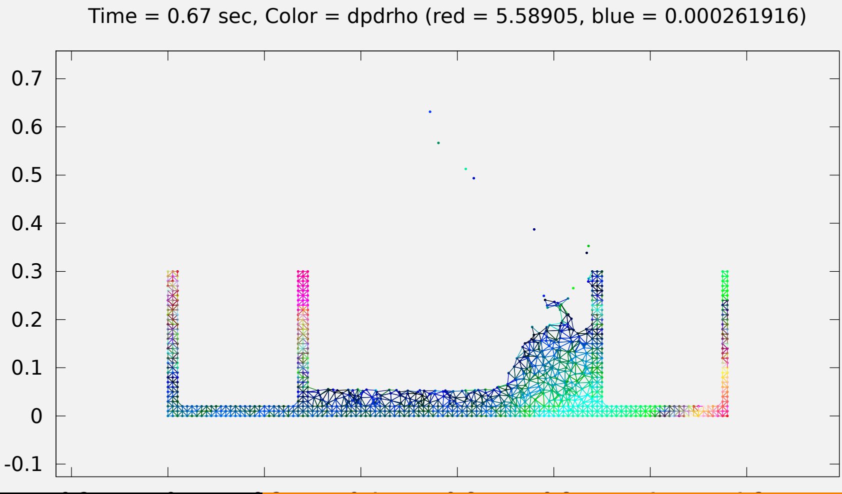
Water column collapse : pressure sensitivity distribution



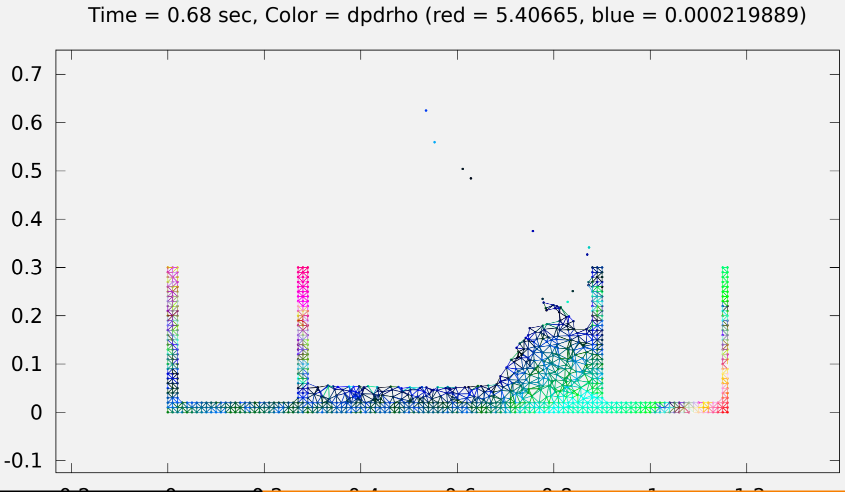
Water column collapse : pressure sensitivity distribution



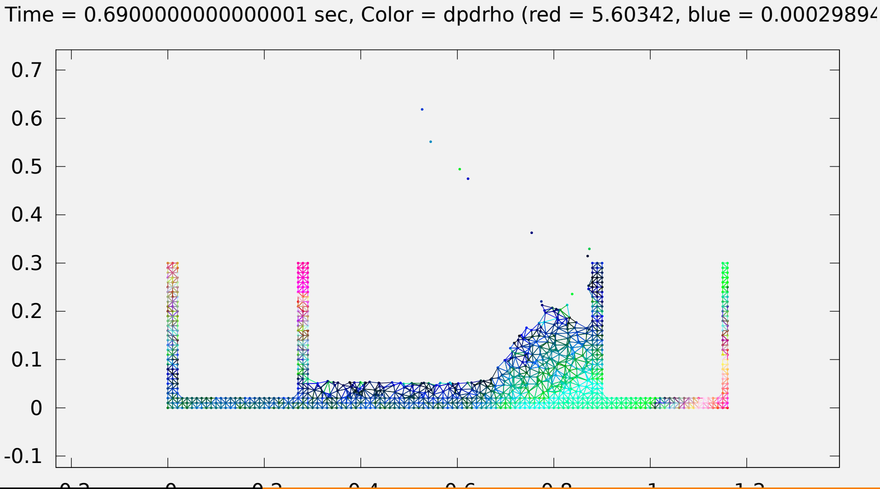
Water column collapse : pressure sensitivity distribution



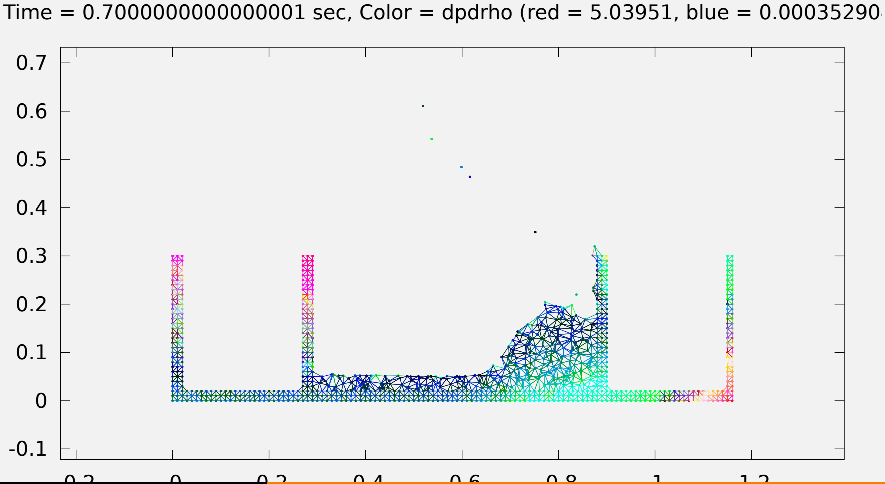
Water column collapse : pressure sensitivity distribution



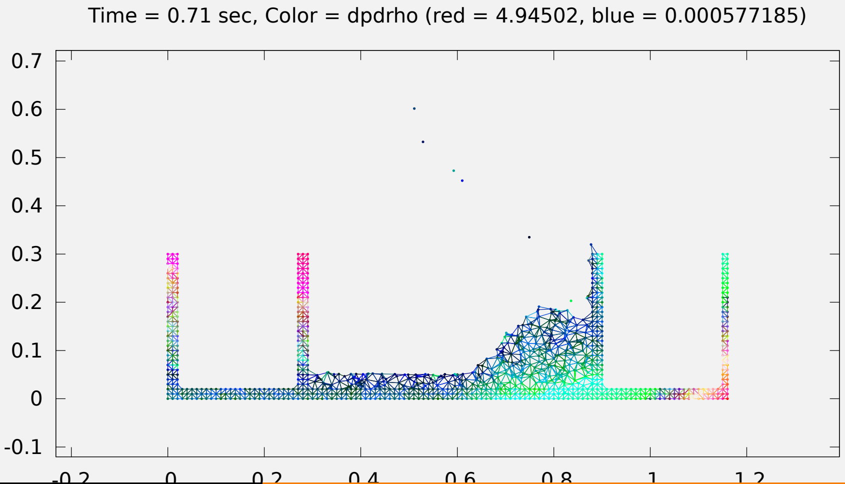
Water column collapse : pressure sensitivity distribution



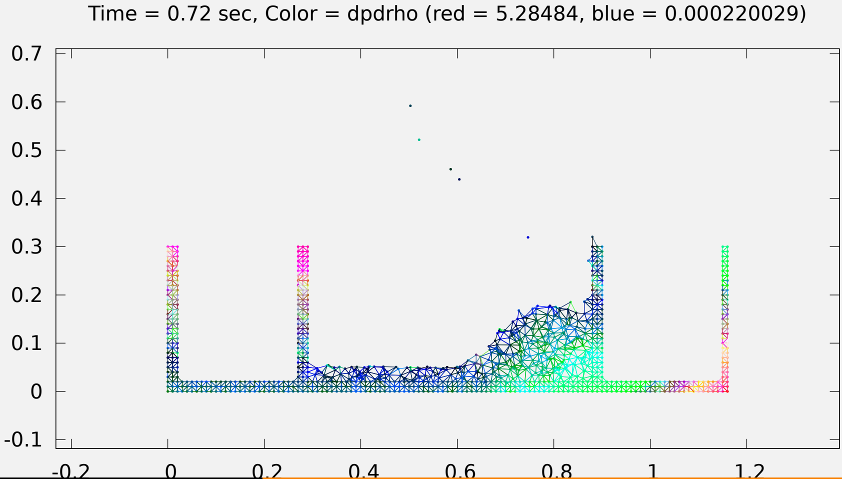
Water column collapse : pressure sensitivity distribution



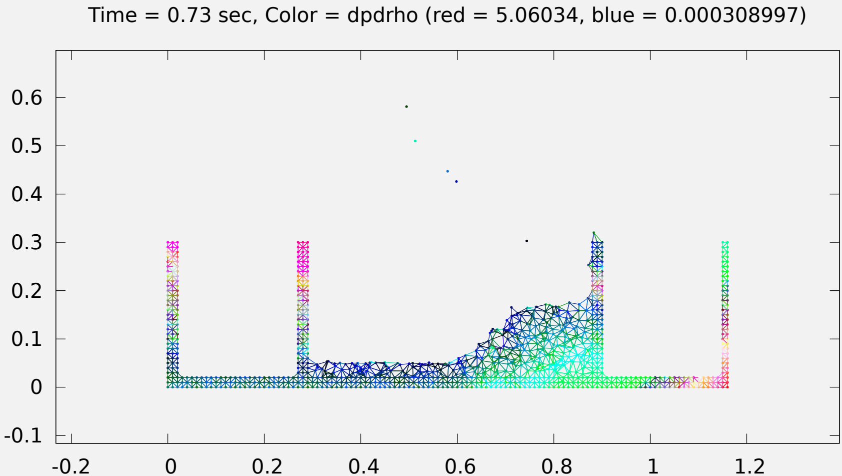
Water column collapse : pressure sensitivity distribution



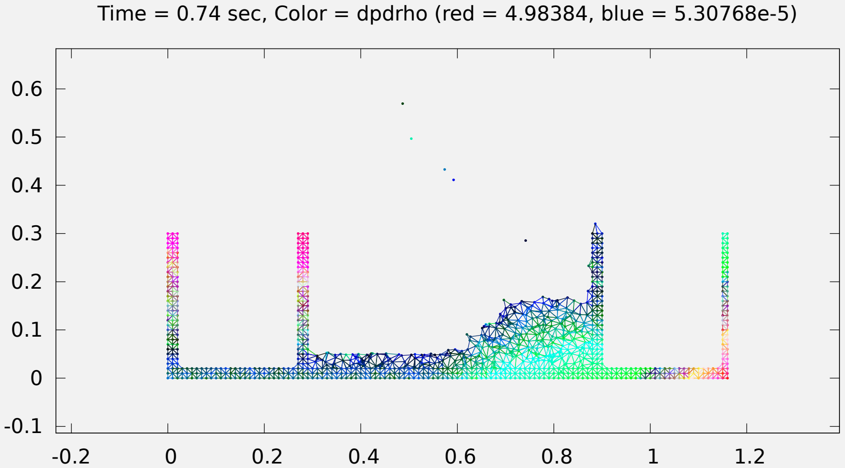
Water column collapse : pressure sensitivity distribution



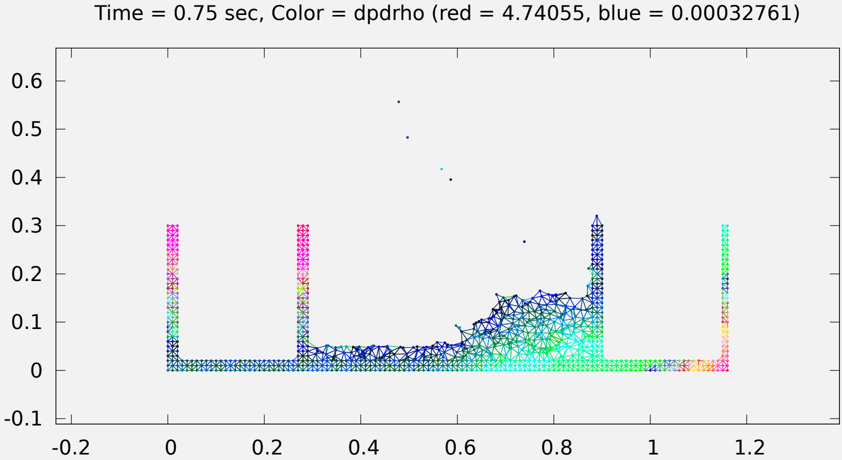
Water column collapse : pressure sensitivity distribution



Water column collapse : pressure sensitivity distribution

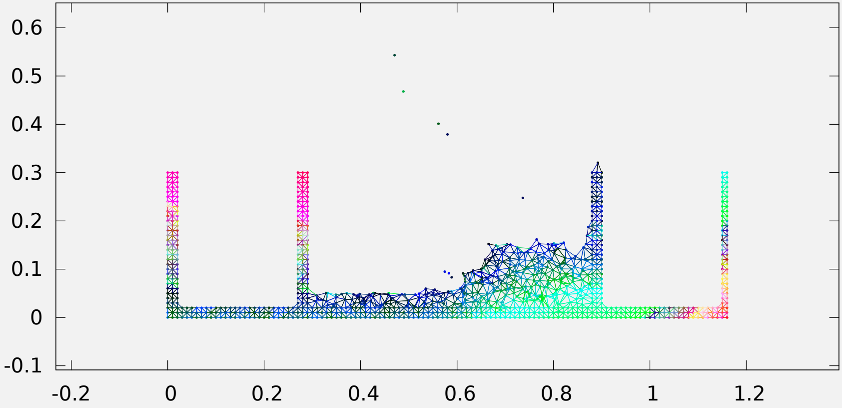


Water column collapse : pressure sensitivity distribution

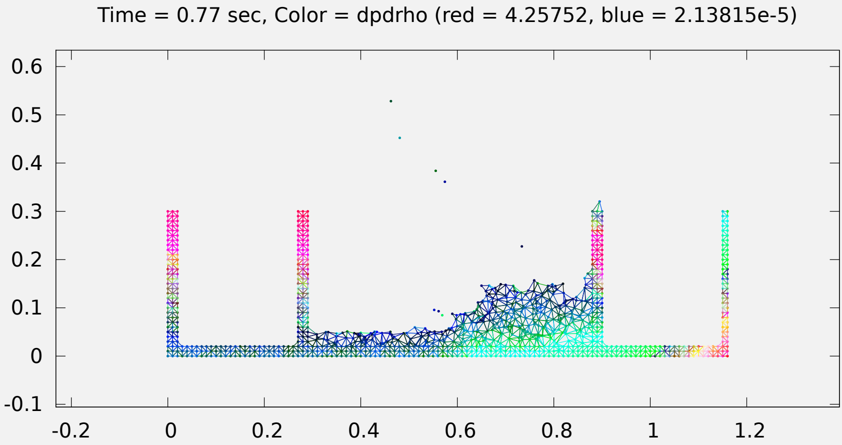


Water column collapse : pressure sensitivity distribution

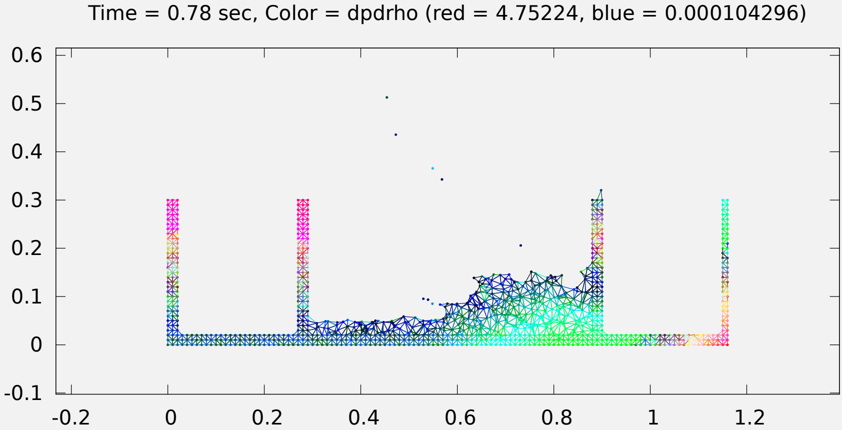
Time = 0.76 sec, Color = dpdrho (red = 4.59307, blue = 0.00188903)



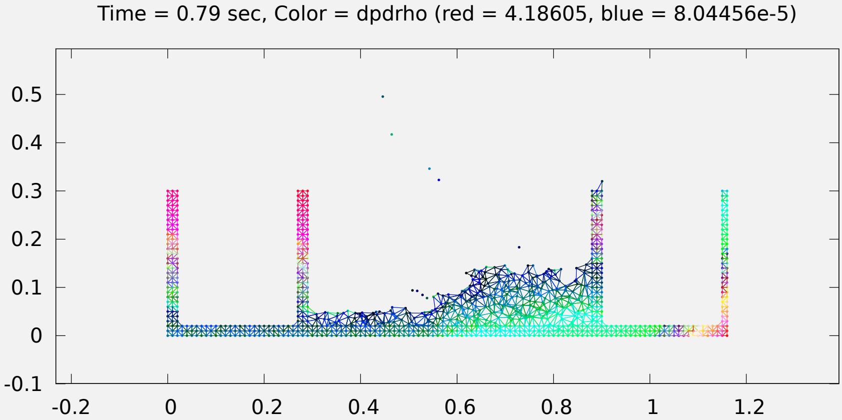
Water column collapse : pressure sensitivity distribution



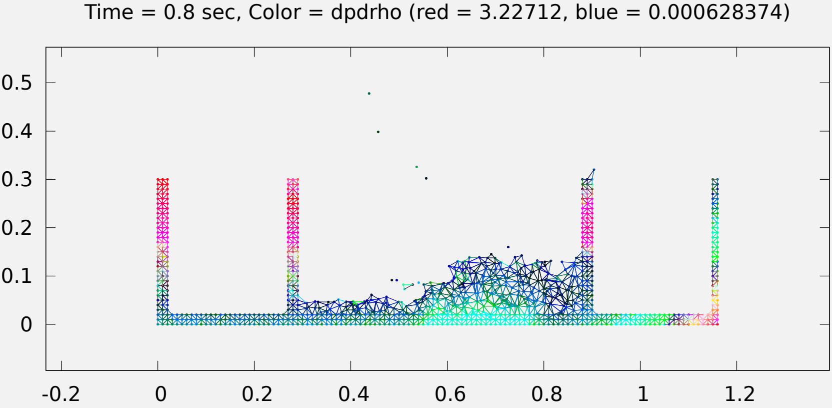
Water column collapse : pressure sensitivity distribution



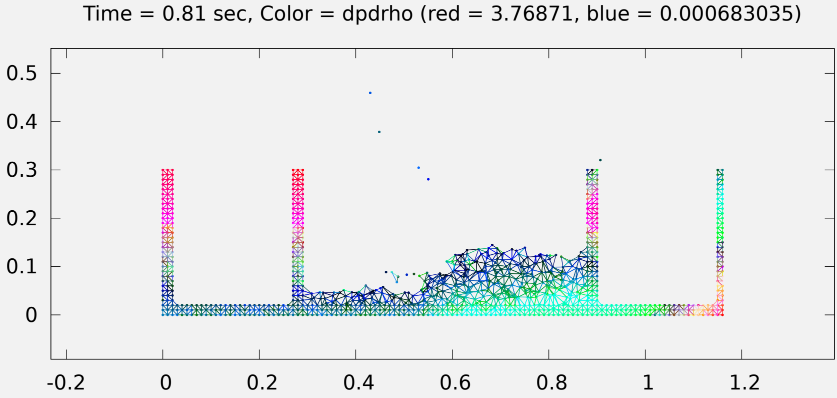
Water column collapse : pressure sensitivity distribution



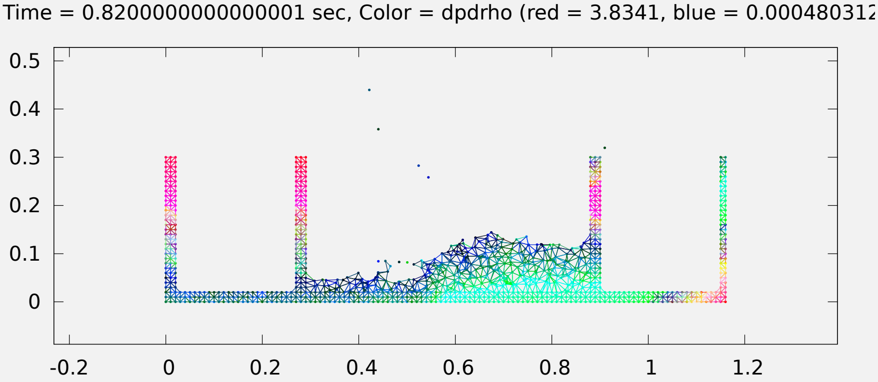
Water column collapse : pressure sensitivity distribution



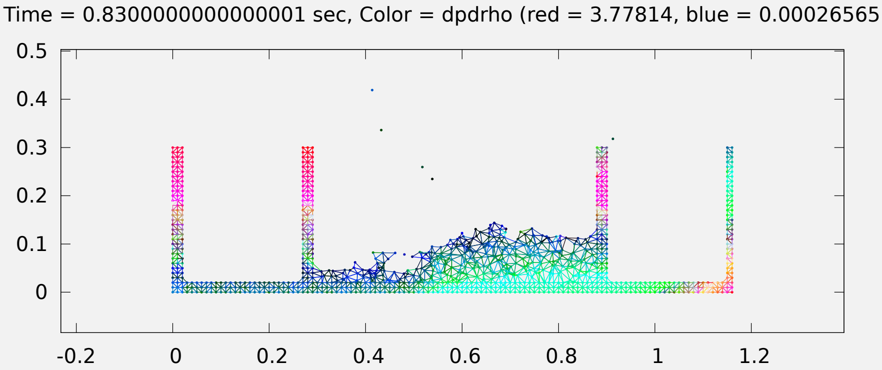
Water column collapse : pressure sensitivity distribution



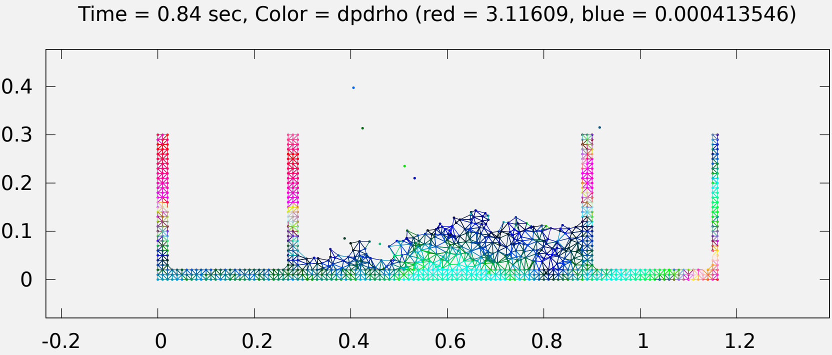
Water column collapse : pressure sensitivity distribution



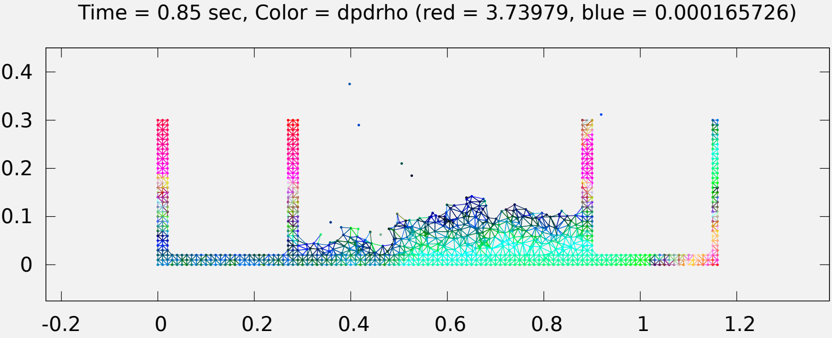
Water column collapse : pressure sensitivity distribution



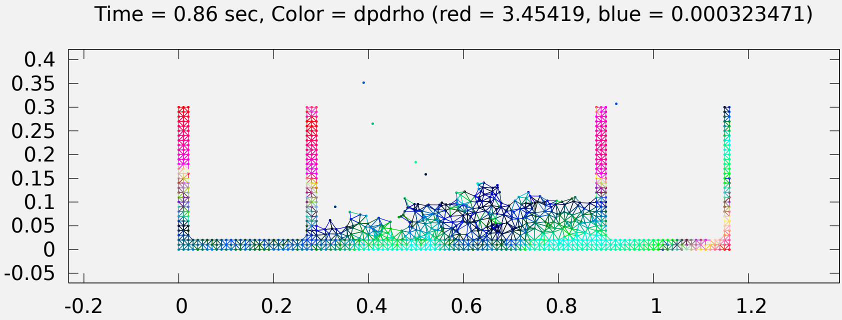
Water column collapse : pressure sensitivity distribution



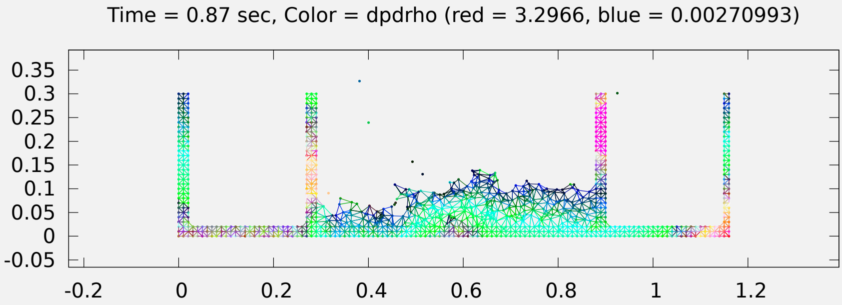
Water column collapse : pressure sensitivity distribution



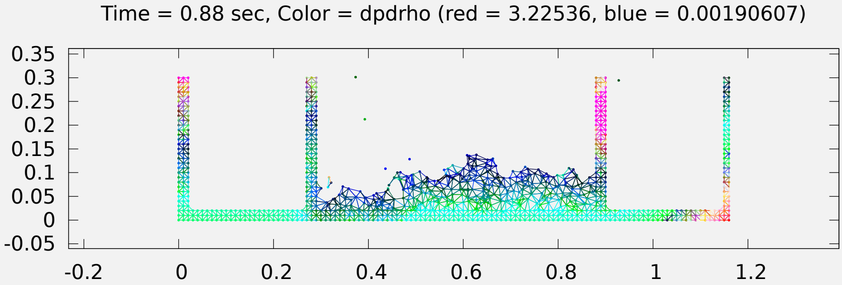
Water column collapse : pressure sensitivity distribution



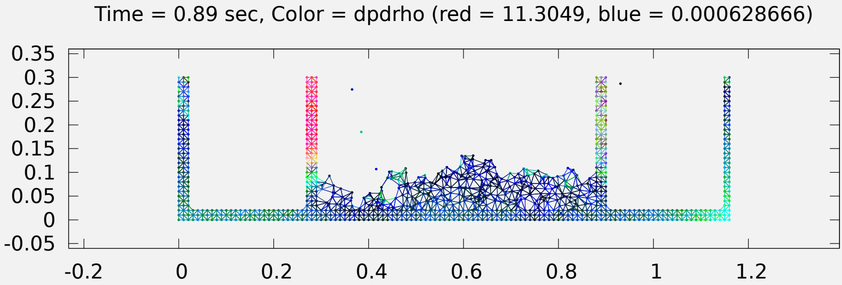
Water column collapse : pressure sensitivity distribution



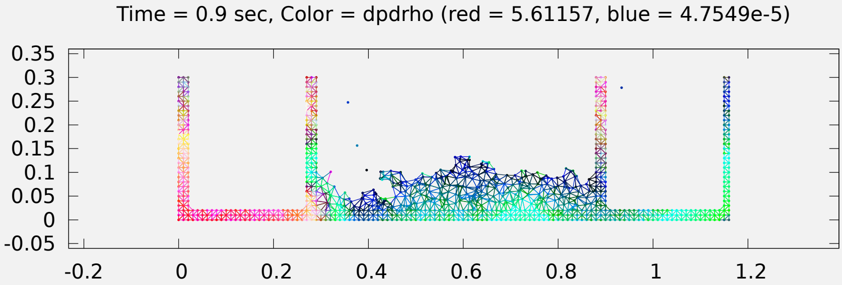
Water column collapse : pressure sensitivity distribution



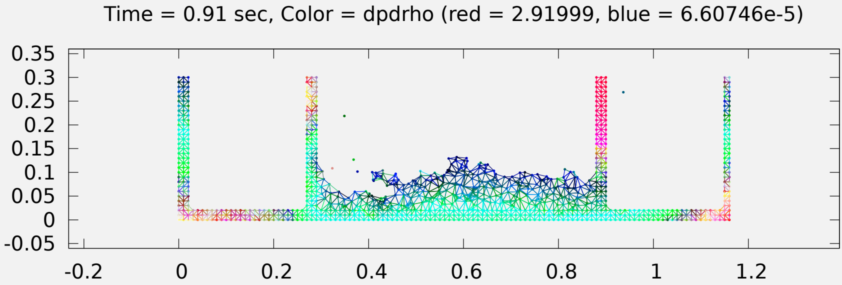
Water column collapse : pressure sensitivity distribution



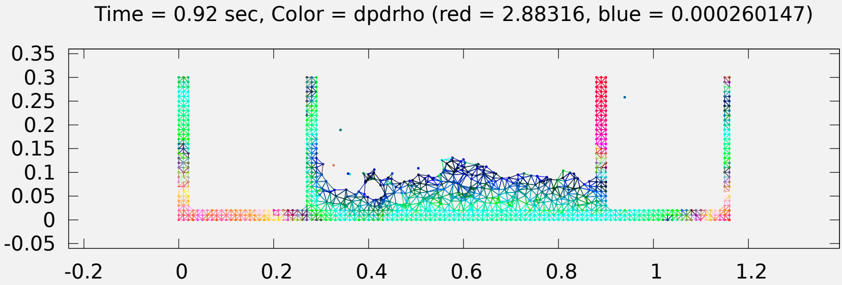
Water column collapse : pressure sensitivity distribution



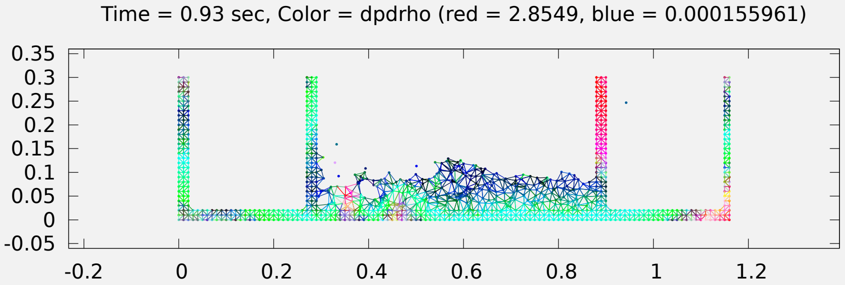
Water column collapse : pressure sensitivity distribution



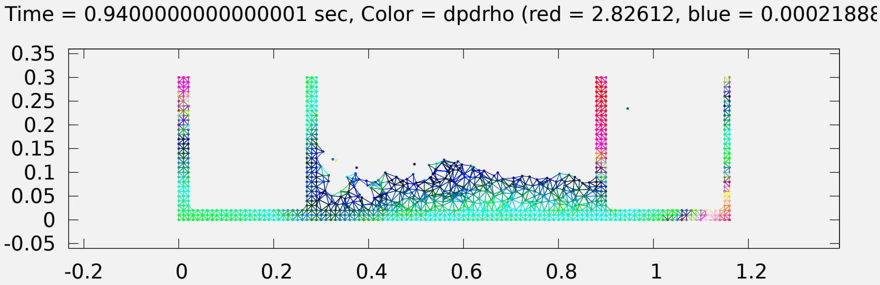
Water column collapse : pressure sensitivity distribution



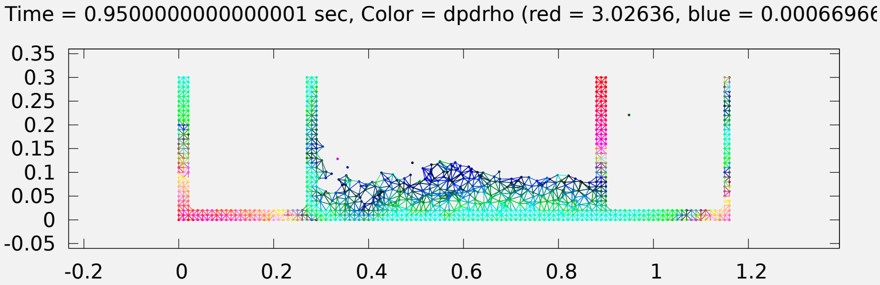
Water column collapse : pressure sensitivity distribution



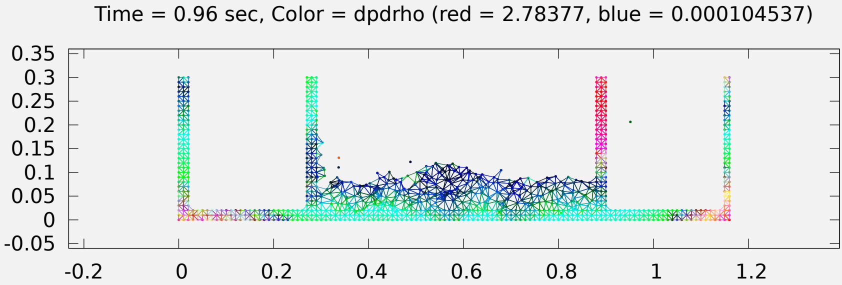
Water column collapse : pressure sensitivity distribution



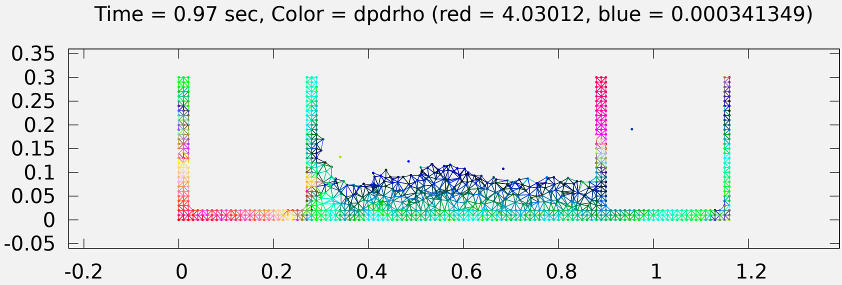
Water column collapse : pressure sensitivity distribution



Water column collapse : pressure sensitivity distribution

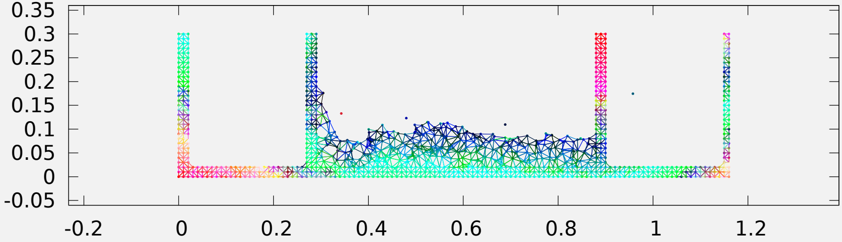


Water column collapse : pressure sensitivity distribution

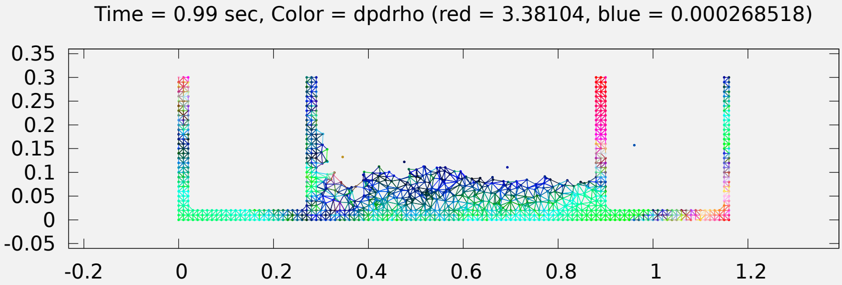


Water column collapse : pressure sensitivity distribution

Time = 0.98 sec, Color = dpdrho (red = 2.9524, blue = 4.2978e-5)



Water column collapse : pressure sensitivity distribution



Conclusions and future work

Conclusions and future work

- PFEM is very effective for free-surface flow problems

Conclusions and future work

- PFEM is very effective for free-surface flow problems
- DDM is fast and accurate for the evaluation of the sensitivity of variables to parameters of interests

Conclusions and future work

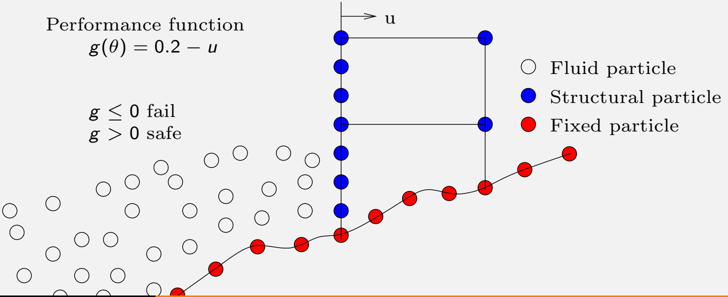
- PFEM is very effective for free-surface flow problems
- DDM is fast and accurate for the evaluation of the sensitivity of variables to parameters of interests
- Combining the two methods together allows us to compute the sensitivity of any problems that PFEM can solve

Conclusions and future work

- PFEM is very effective for free-surface flow problems
- DDM is fast and accurate for the evaluation of the sensitivity of variables to parameters of interests
- Combining the two methods together allows us to compute the sensitivity of any problems that PFEM can solve
- Future work includes fluid-structure interaction and reliability analysis,

Conclusions and future work

- PFEM is very effective for free-surface flow problems
- DDM is fast and accurate for the evaluation of the sensitivity of variables to parameters of interests
- Combining the two methods together allows us to compute the sensitivity of any problems that PFEM can solve
- Future work includes fluid-structure interaction and reliability analysis, ex. wave loading on structure with reliability analysis



M. Kleiber, H. Antunez, T.D. Hien, and P. Kowalczyk.
Parameter Sensitivity in Nonlinear Mechanics.
John Wiley & Sons, 1997.

Eugenio Onate, Sergio R. Idelsohn, Miguel A. Celigueta, and Riccardo Rossi.
Advances in the particle finite element method for the analysis of fluid-multibody interaction and bed erosion in free surface flows.
Computer Methods in Applied Mechanics and Engineering, 197(19-20):1777–1800, 2007.

E. Onate, S.R. Idelsohn, F. Del Pin, and R. Aubry.
The particle finite element method. an overview.
International Journal of Computational Methods, 1(2):267–307, 2004.

Acknowledgments

Support for this research has been provided by the National Science Foundation Grant #0847055 through the Hazard Mitigation and Structural Engineering program.

Acknowledgments

Support for this research has been provided by the National Science Foundation Grant #0847055 through the Hazard Mitigation and Structural Engineering program.

National Science Foundation
WHERE DISCOVERIES BEGIN

Questions?