Aalto University
School of Engineering

GEO – E1050
Finite Element Method in Geoengineering

Lecture 11-12. Other numerical methods

To learn today & next time…

The lectures should give you overview of other numerical methods

1. Discrete element method (DEM, also distinct element method)
   - assumptions
   - solutions
   - problmes & accuracy

2. Smoothed particle hydrodynamics (SPH)
3. Material Point Method (MPM)
4. Particle Finite Element Method in Geoengineering (PFEM)
5. XFEM – eXtended Finite Element Method in Geoengineering (XFEM)
6. ALE , CLE – Coupled Lagrangian – Eurlerian FEM
7. Meshfree methods

Methods on continuous – discontinuous scale

FEM

CLE, ALE
Coupled Lagrangian – Eurlerian FEM

MPM

Smoothed Particle Hydrodynamics (SPH)

Discrete element method (DEM)

Continuous

Classical Meshfree methods

PFEM

XFEM

Discontinuous

Based on continuum mechanics

Not based on continuum mechanics

Methods on continuous – discontinuous scale

FEM

CLE, ALE: Coupled / Arbitrary Lagrangian – Eurlerian FEM

MPM

Smoothed Particle Hydrodynamics (SPH)

Discrete element method (DEM)

Continuous

Classical Meshfree methods

PFEM

XFEM

Discontinuous

Based on continuum mechanics

Not based on continuum mechanics

Discrete Element Method (DEM)

Also known as distinct element method

Idea: we model each grain of soil separately

We need to model all the contacts and contact behaviour

Each time step – we evaluate forces and velocities of all particles

Contact & contact forces are essential

Normally used for granular materials and atoms

Discrete Element Method (DEM)

Also known as distinct element method

Idea: we model each grain of soil separately

We need to model all the contacts and contact behaviour

Each time step – we evaluate forces and velocities of all particles. Method is time-step dependent

Contact & contact forces are essential

Due to simplified shapes of particles and simplified contact, method is known to be **problem and size dependent** (i.e. requires different parameters for different problems with same material)

Discrete Element Method (DEM)

Also known as distinct element method

Few simulations: Hannover, group of Prof. Wriggers

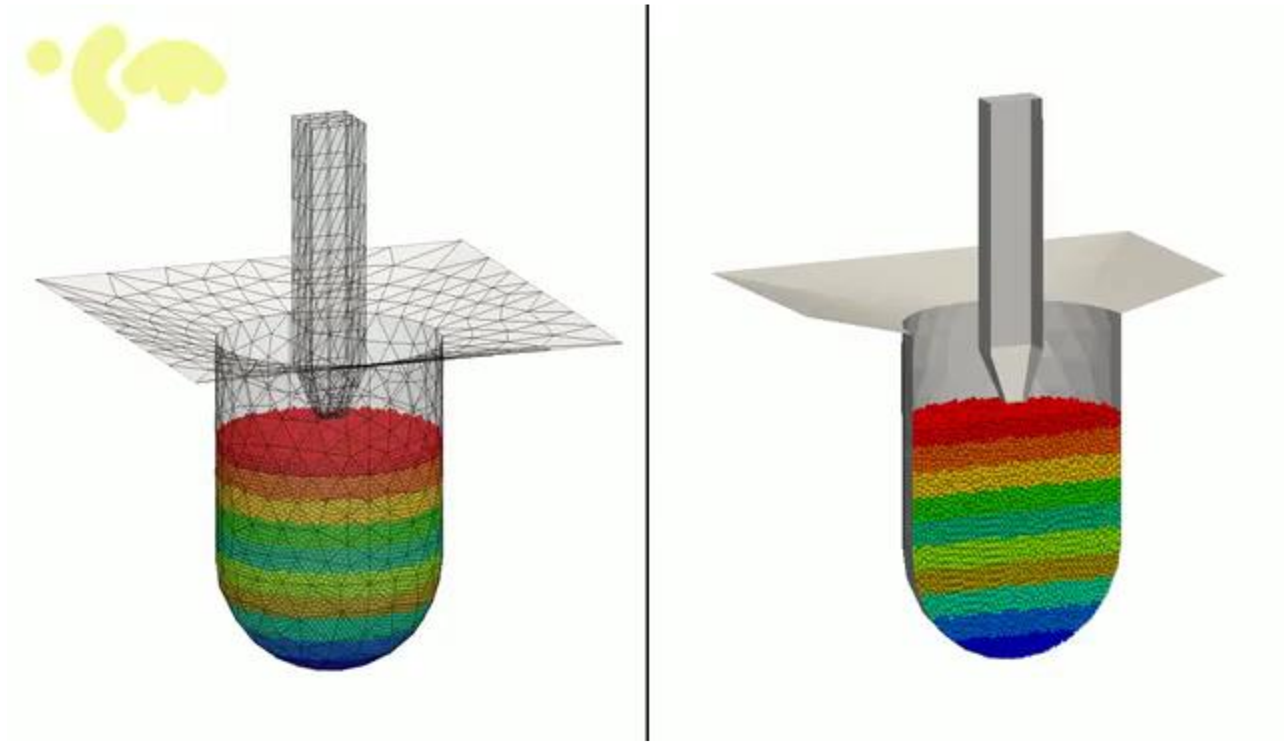
Discrete Element Method (DEM)

Also known as distinct element method

Few simulations: Hannover, group of Prof. Wriggers

Discrete Element Method (DEM)

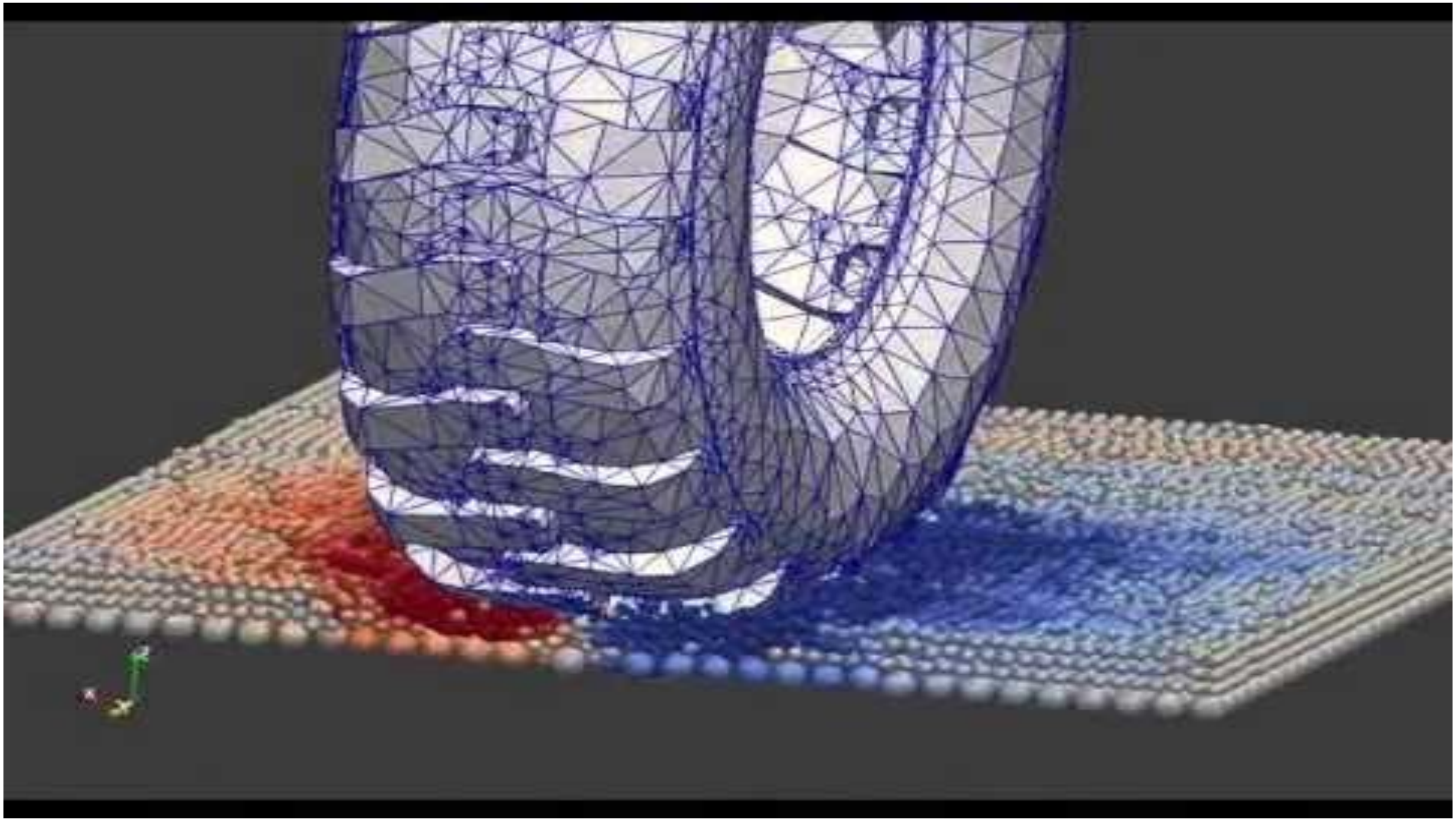
Also known as distinct element method



Few simulations: Hannover, group of Prof. Wriggers

Discrete Element Method (DEM)

Also known as distinct element method



DEM FEM coupling

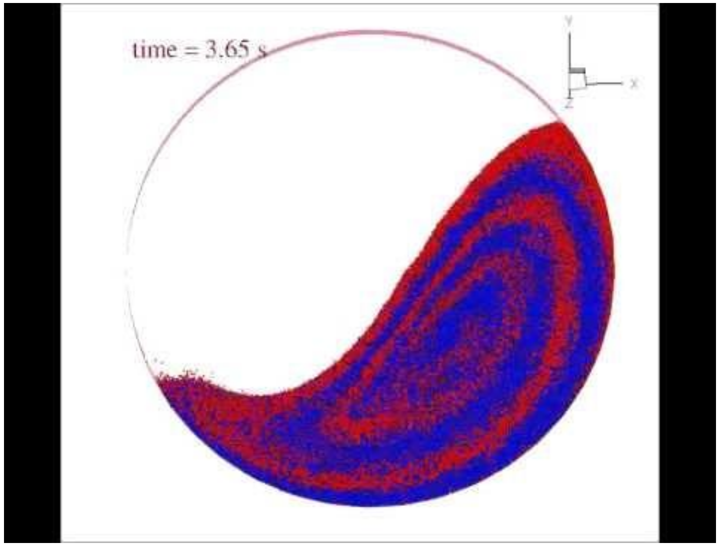
Discrete Element Method (DEM)

Also known as distinct element method

Convergence:

UoM presentation - DEM convergence.pdf

More simulations:



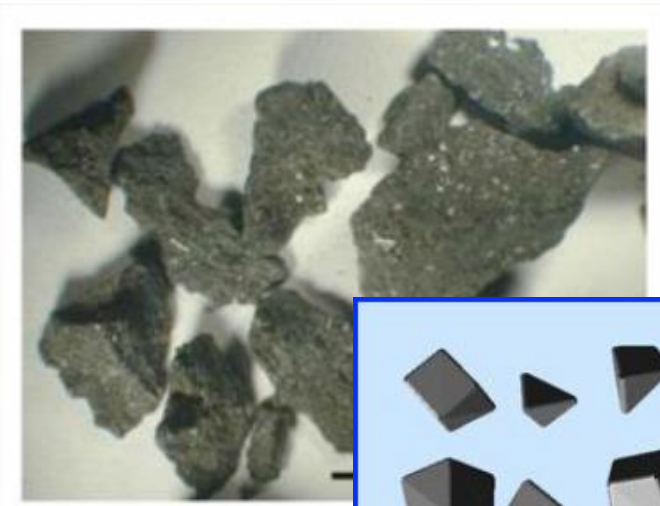
In short, 2D quite does not work and 3D is very expensive… and even then it may not work…

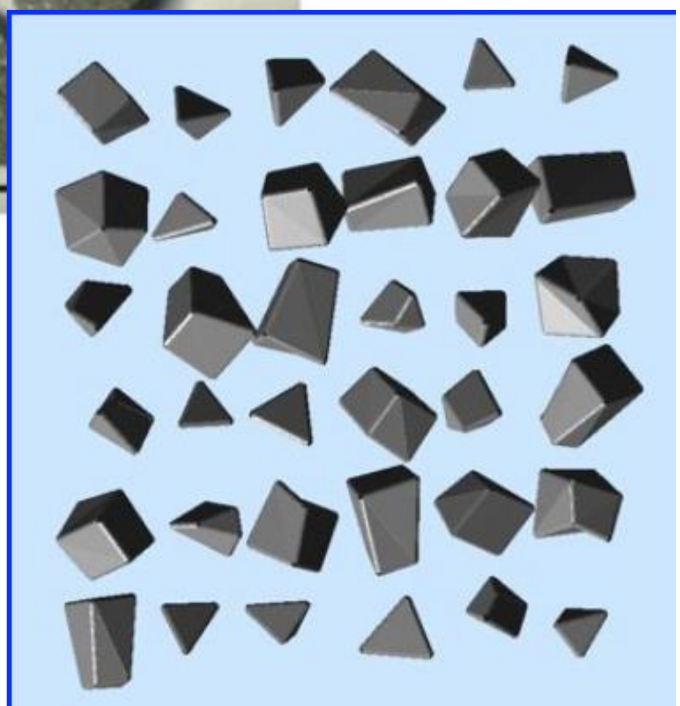
Discrete Element Method (DEM)

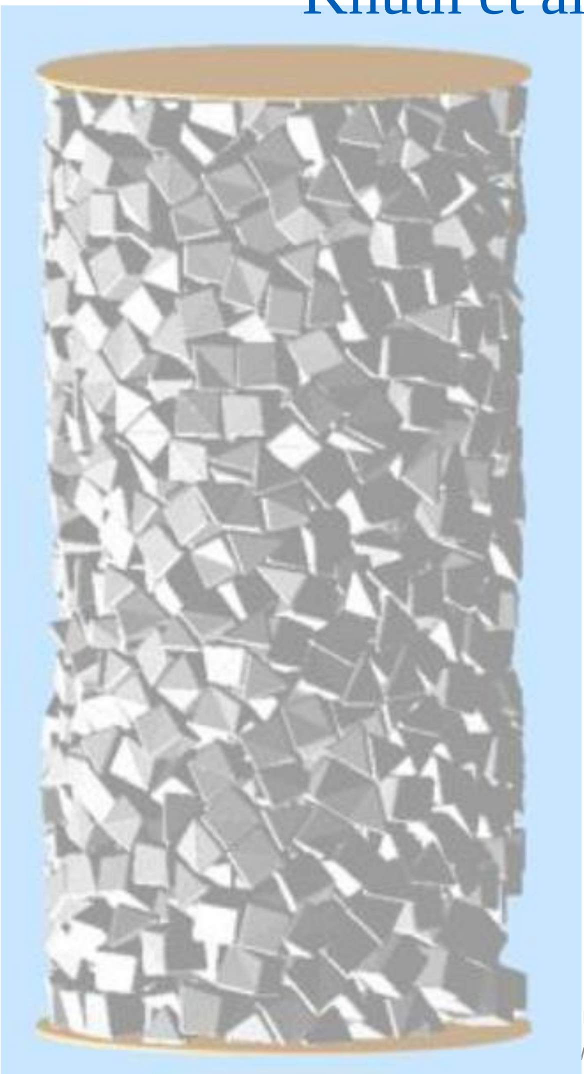
Also known as distinct element method

Knuth et al. 2010

Shape effects…



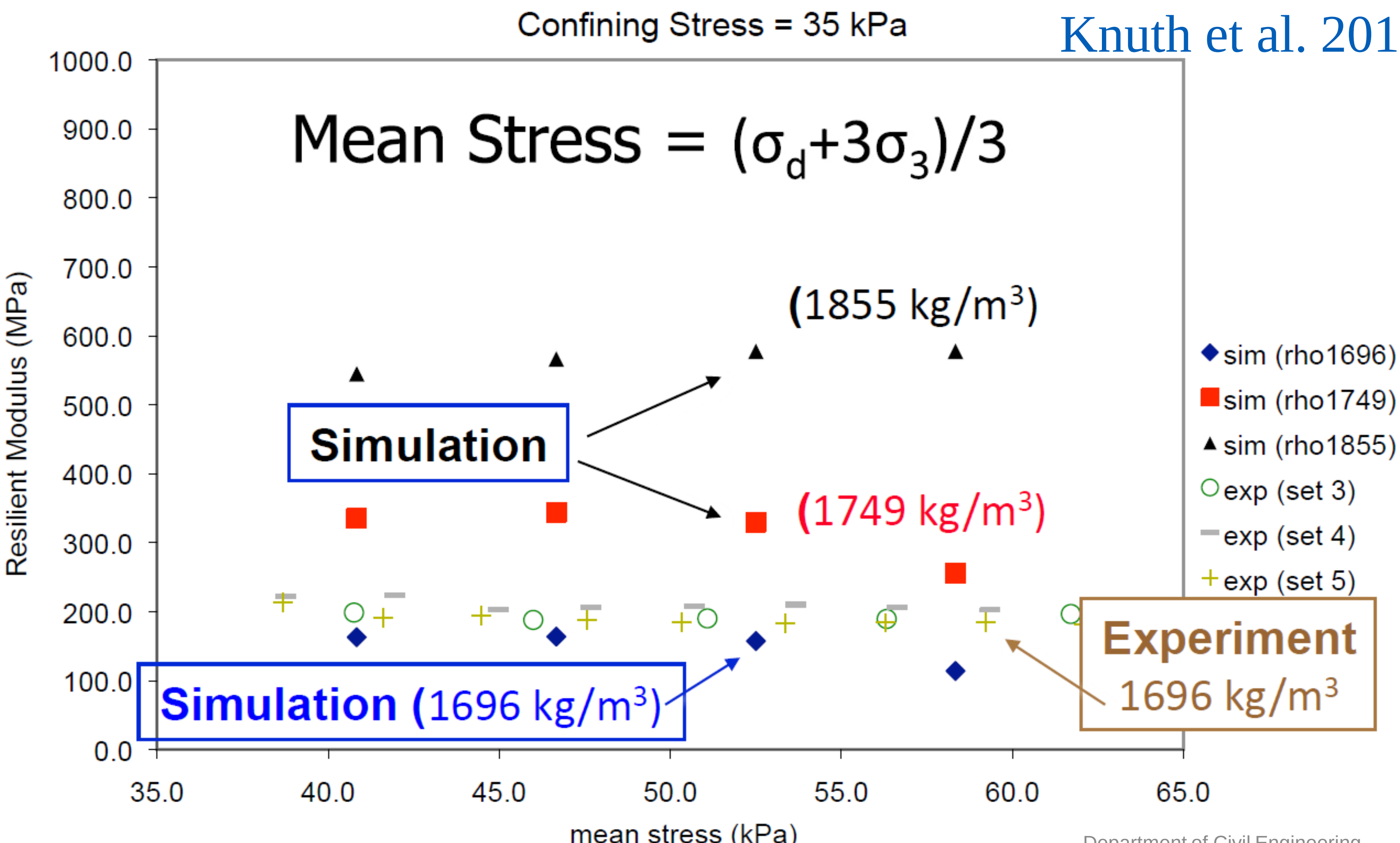




Discrete Element Method (DEM)

Also known as distinct element method

Knuth et al. 2010



Size effects…

Discrete Element Method (DEM)

Also known as distinct element method

Software:

YADE (free & open source)

- generally non-cohesive materials, or materials with some cohesion

Quite a lot of other software…

3DEC – by ITASCA, now popular in mining industry

Discrete Element Method (DEM)

Also known as distinct element method

Wait, using DEM for non-granular materials???

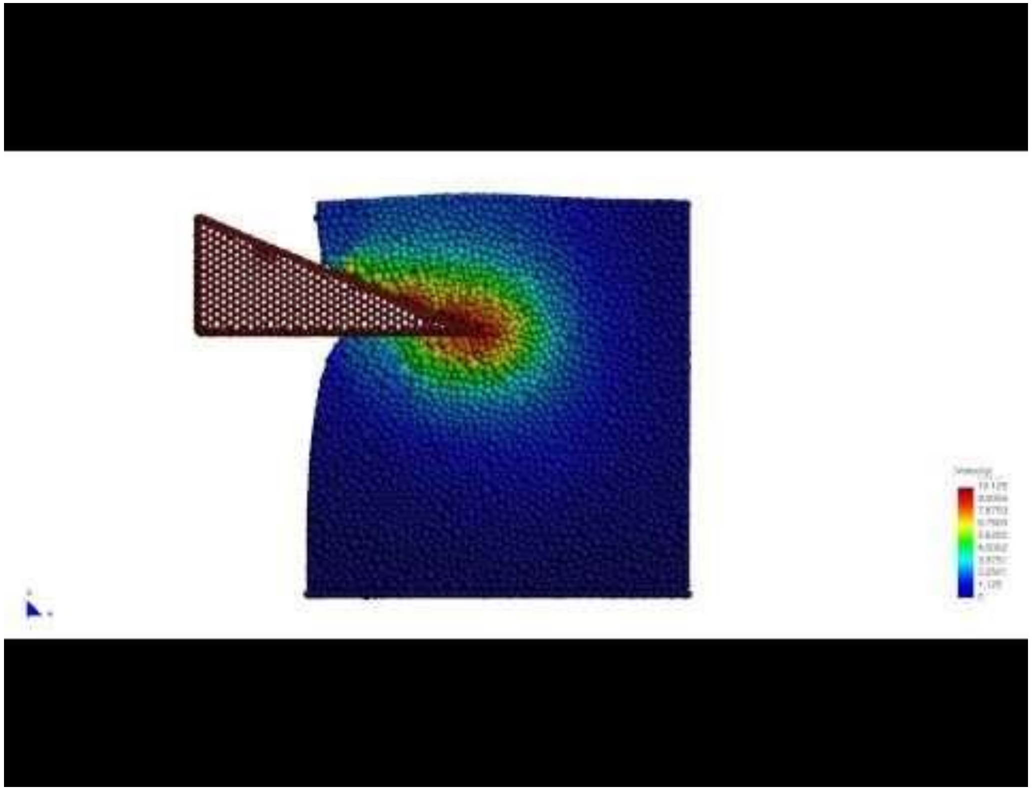
Of course leads to lots of problems and issues:

- in reality to get decent results cohesion between grains should be scale dependent
- contact between assemblies of non – smooth particles (i.e. when crack is formed) is again problematic (generally, to get real surface with required roughness, **MANY** particles are needed, and other solutions do not work well
- currently used rather for flow than anything else…

Discrete Element Method (DEM)

Also known as distinct element method

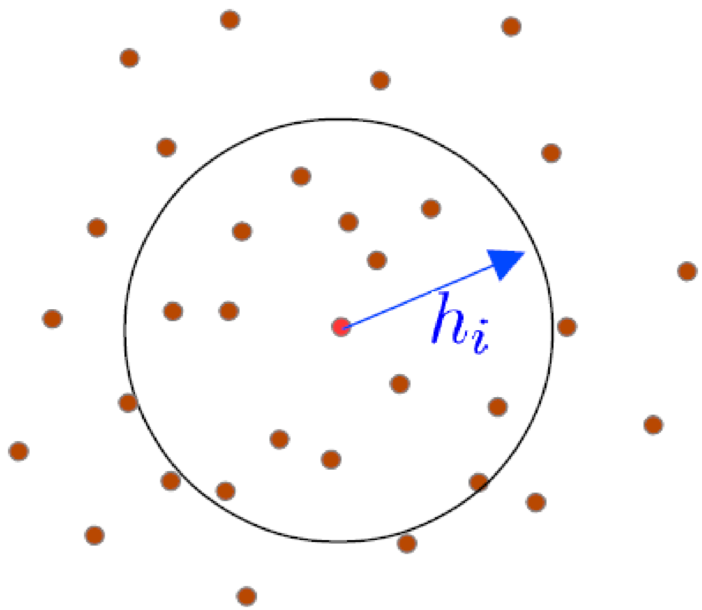
Wait, using DEM for non-granular materials???



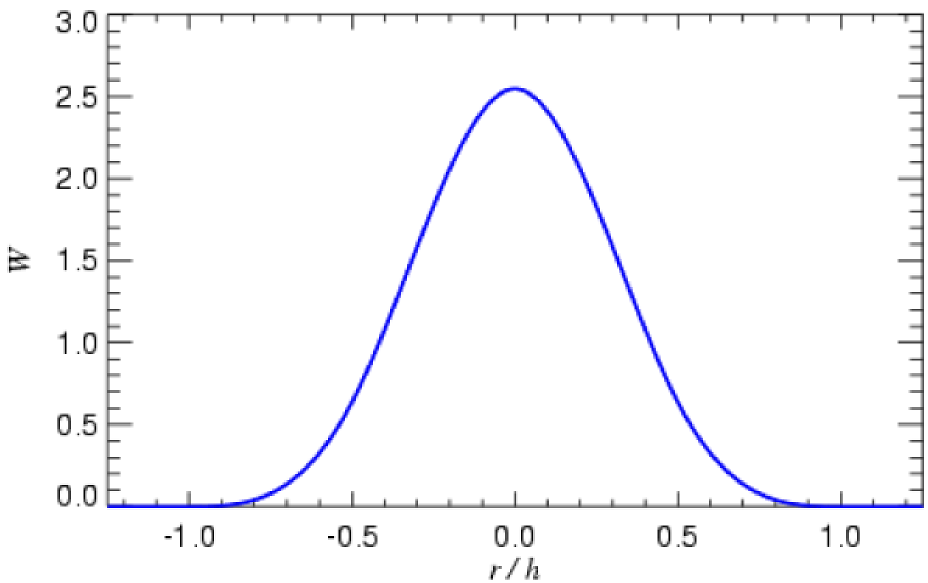
Aalto University
School of Engineering

Smoothed particle hydrodynamics (SPH)

Smoothed Particle Hydrodynamics (SPH)



Density computed via weighted sum over neighbouring particles…



Smoothed Particle Hydrodynamics (SPH)

Problems:

- boundary conditions
- numerical noises
    - sometimes – velocity noise of few percent of local sound speed…
    - instabilities over contact discontinuities

- requires high artificial viscosity (to mute errors), giving high viscosity of the system, leading to errors

Benefits:

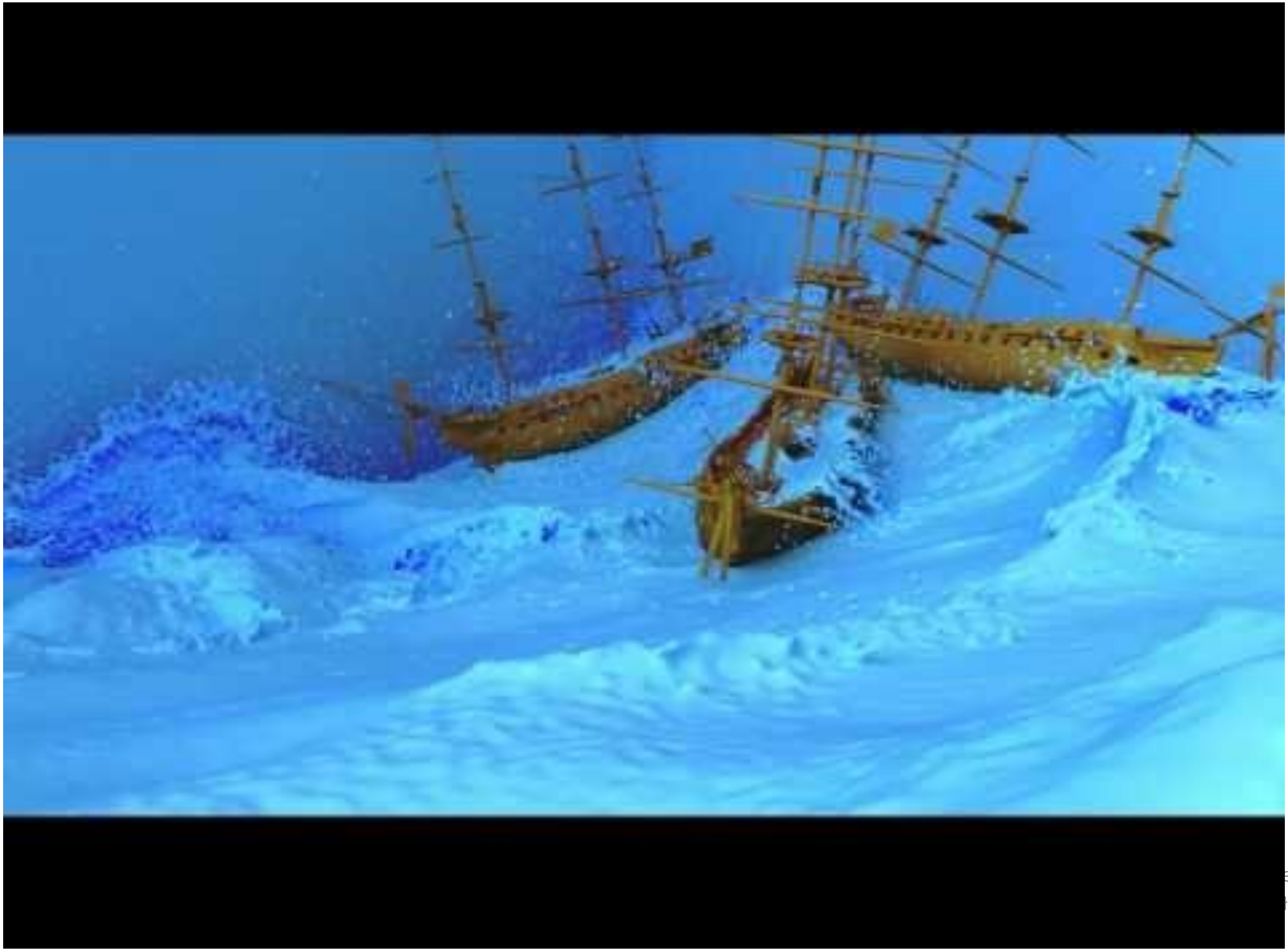
- versatile, simple, good conservation properties
- quite robust

Smoothed Particle Hydrodynamics (SPH)

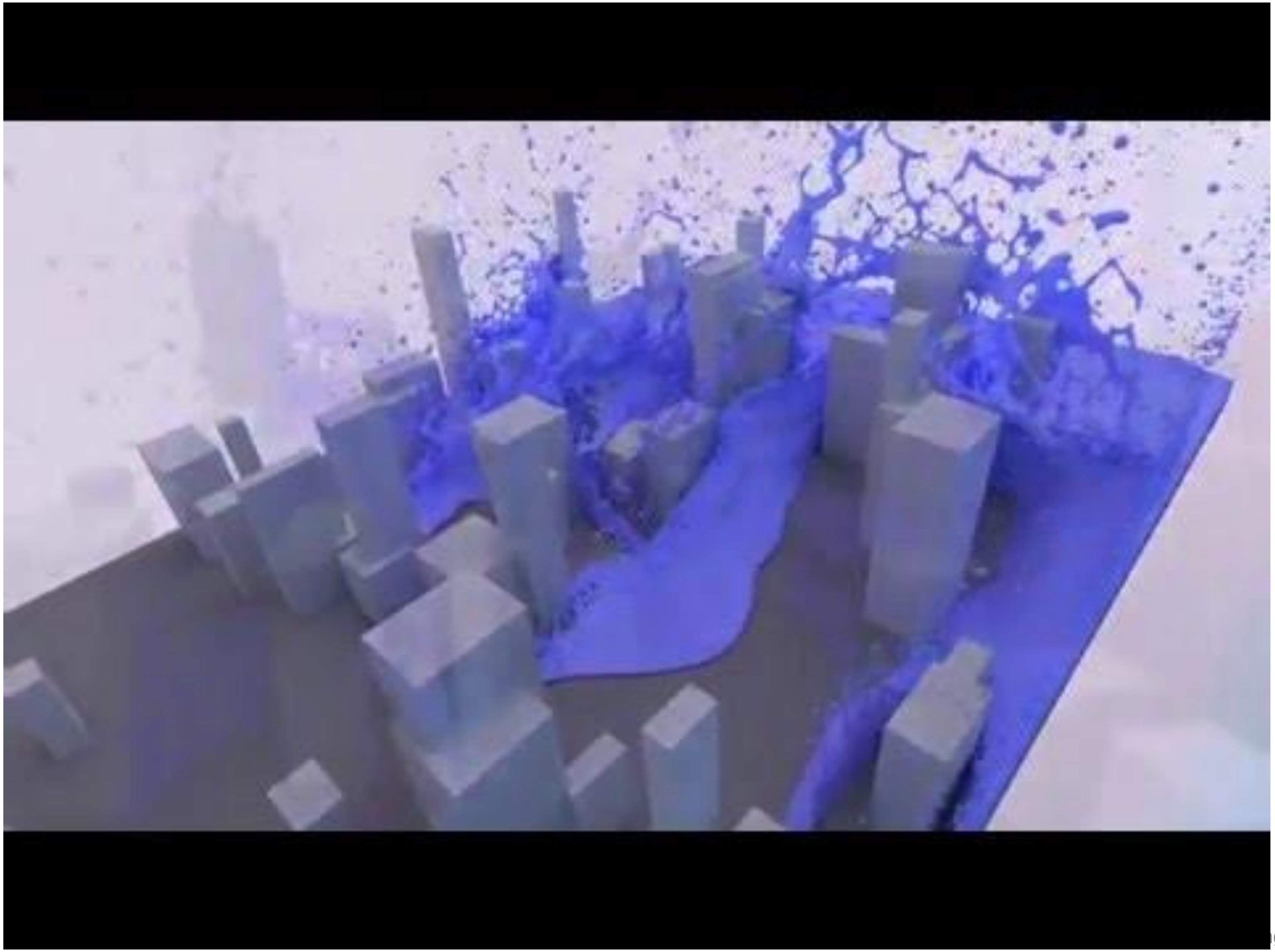
Barcelona supercomputing centre

Smoothed Particle Hydrodynamics (SPH)

Smoothed Particle Hydrodynamics (SPH)



Smoothed Particle Hydrodynamics (SPH)



Material Point Method MPM

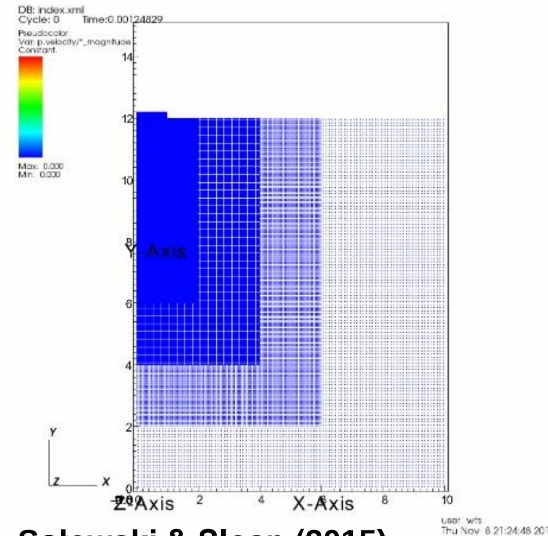
Material Point Method MPM

Ken Kamrin's group, MIT

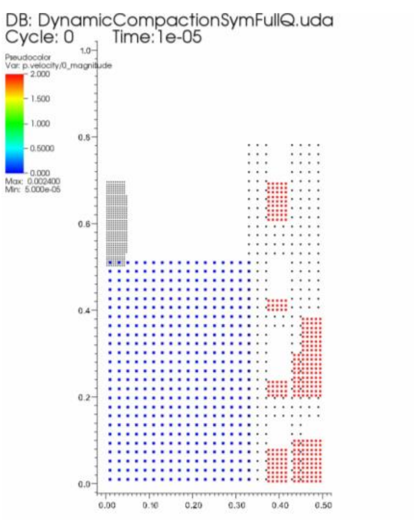
Some more movies ;)

Material Point Method:

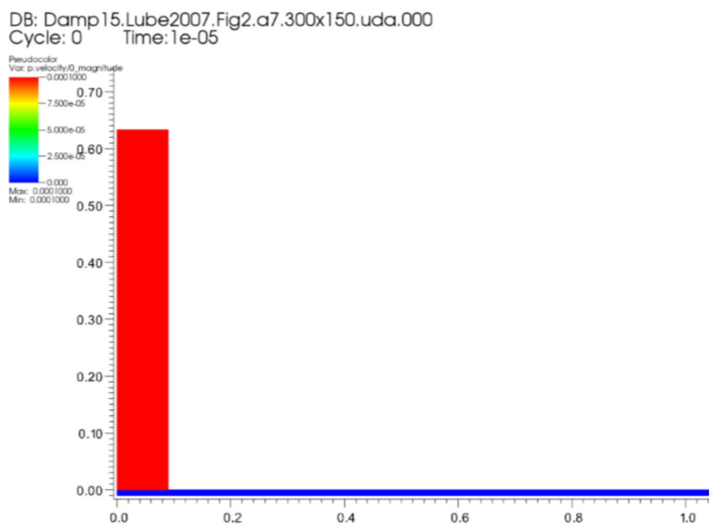
- good for very large deformations
- explicit
- dynamics
- continuum method (like FEM)



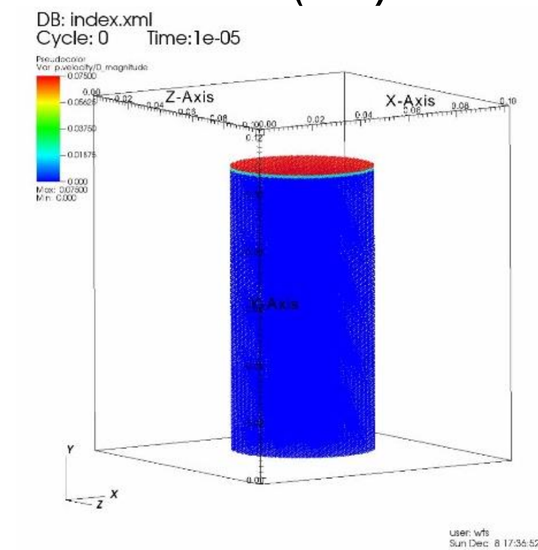
Sołowski & Sloan (2015)



Sołowski et al. (2013)

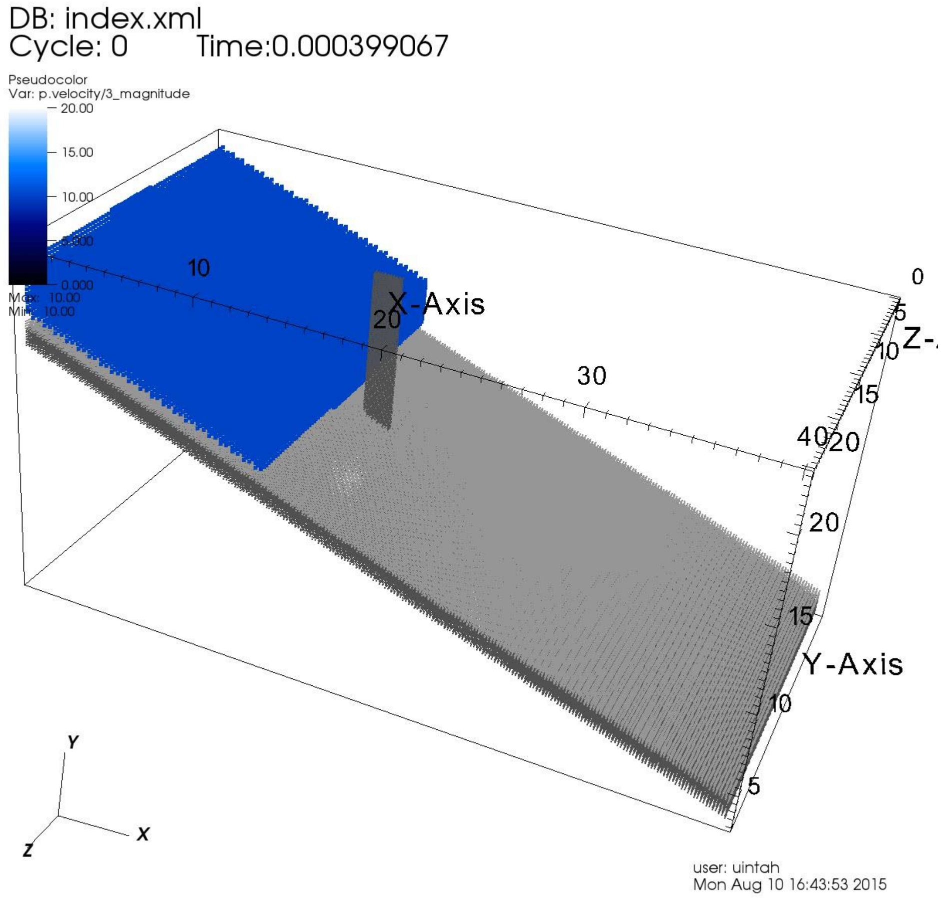


Sołowski & Sloan (2013)



Sołowski et al. (2015)

Some more movies ;)

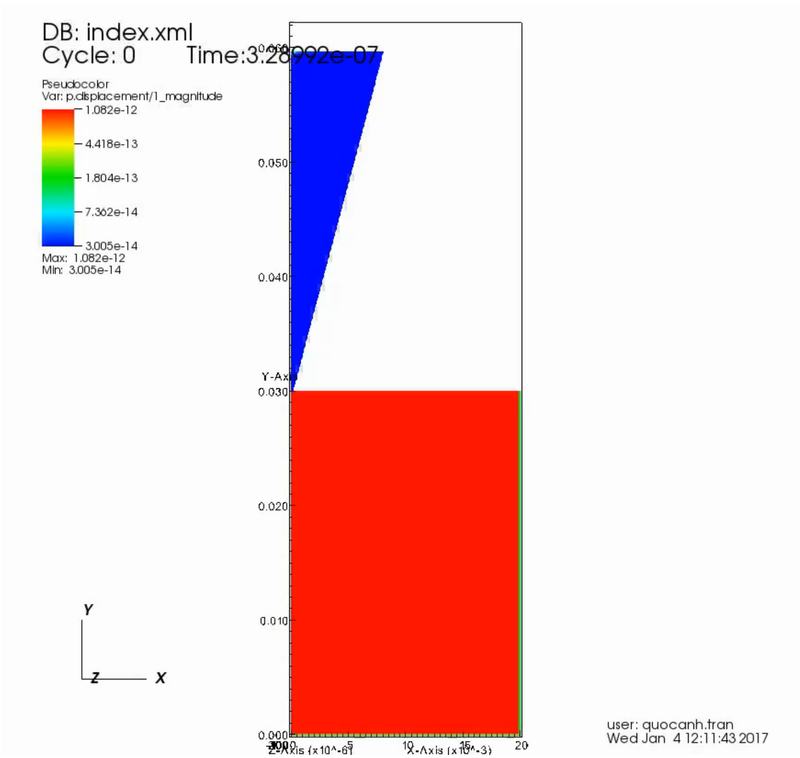


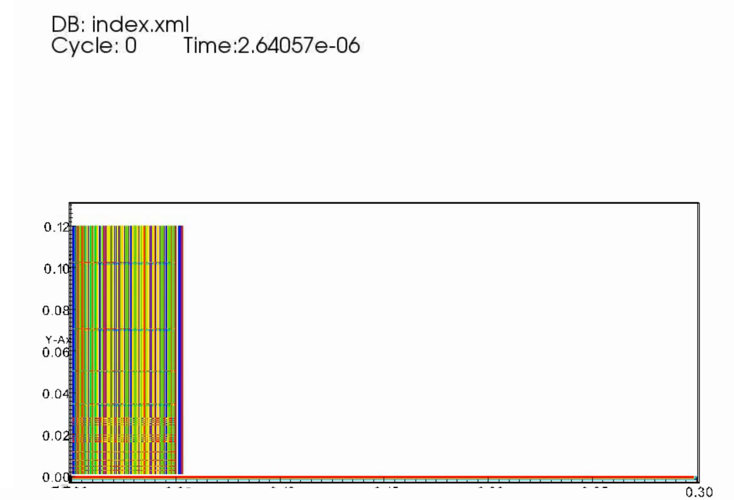
Material Point Method:

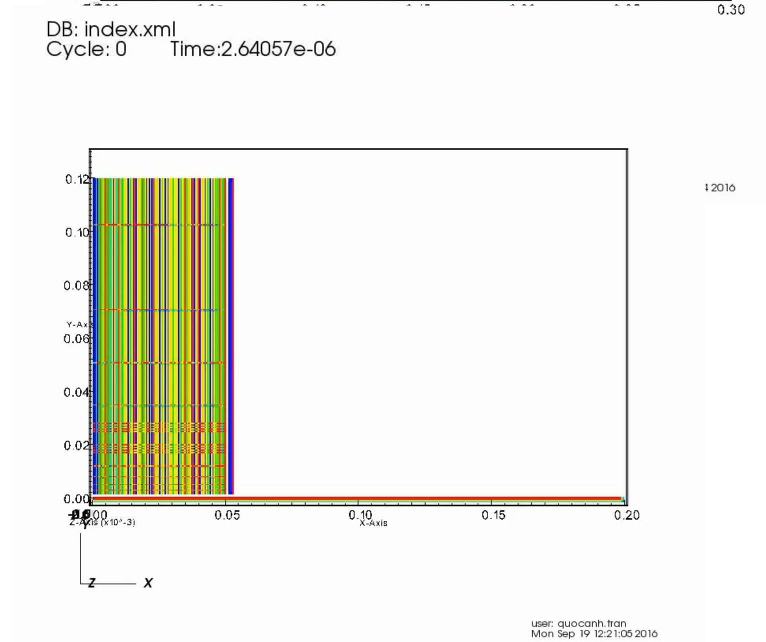
- **good for very large deformations**
- **explicit**
- **dynamics**
- **continuum method (like FEM)**

Sołowski et al. (2015)

Some more movies ;)

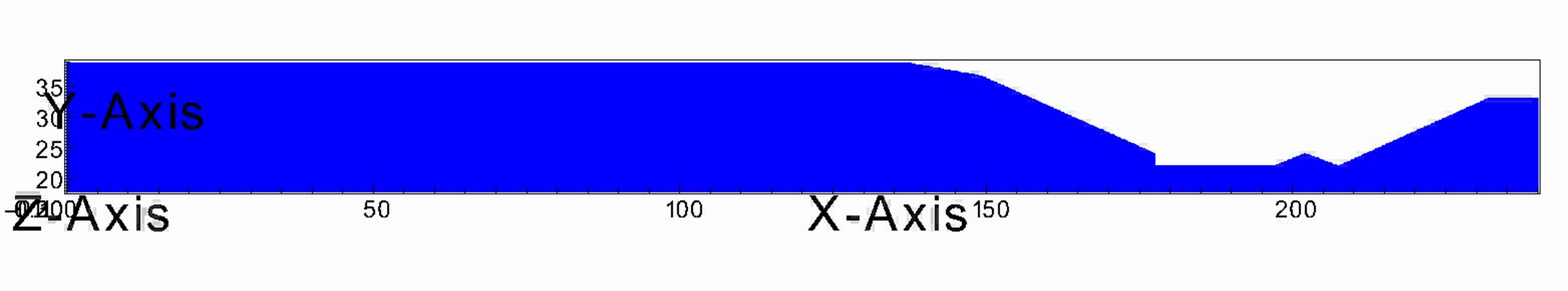


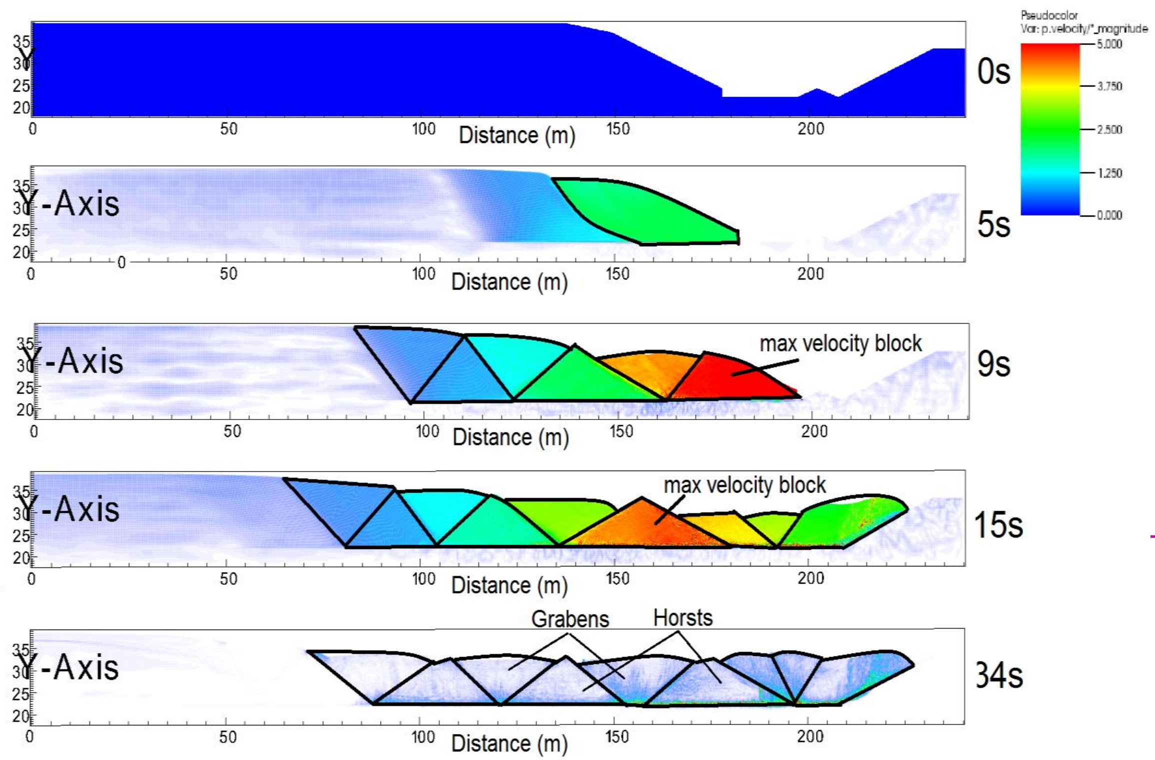




Tran et al. (2017)

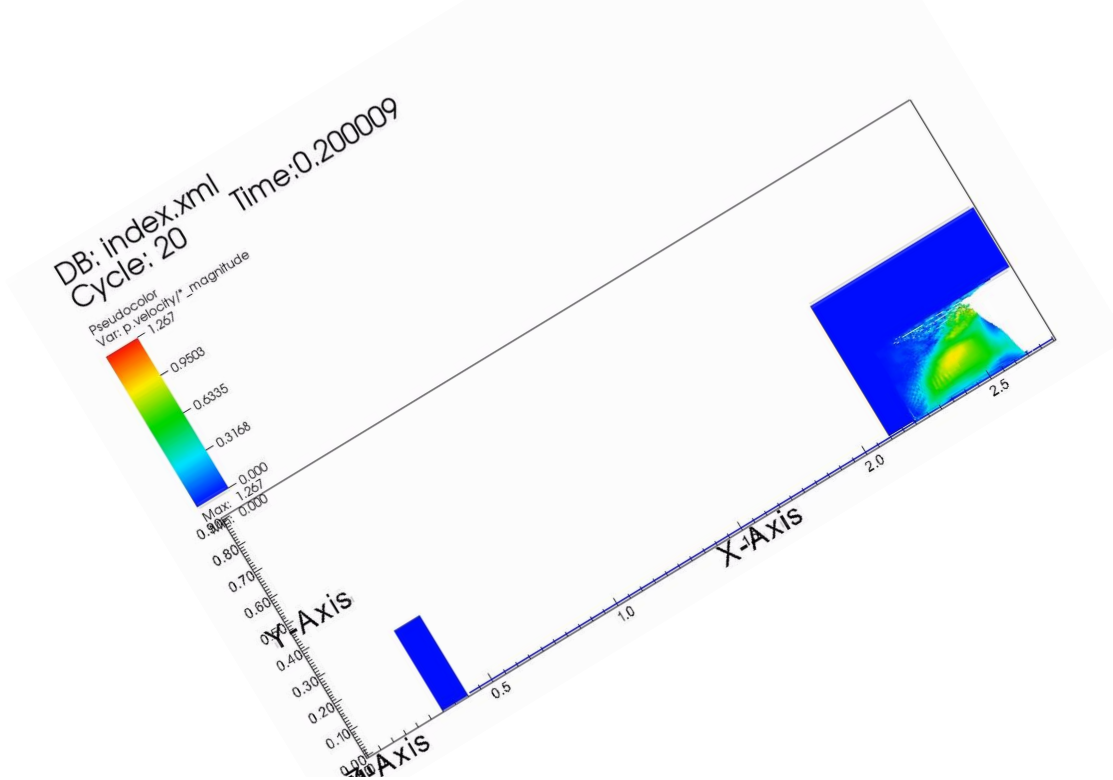
Some more movies ;)

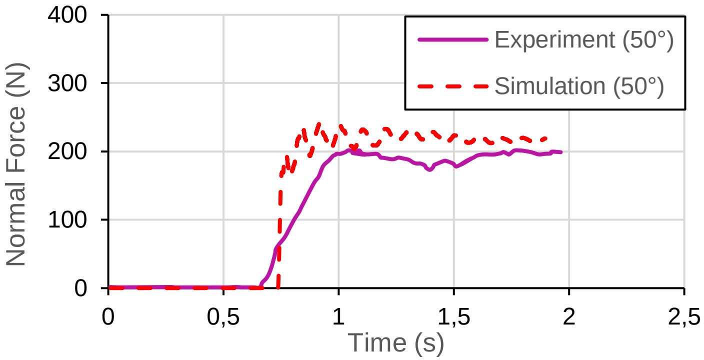


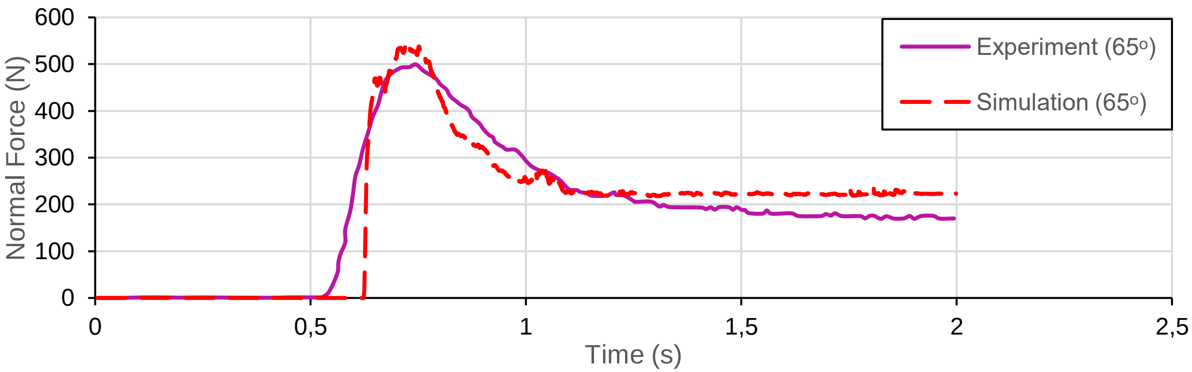


Tran & Sołowski (2018?)

Some more movies ;)







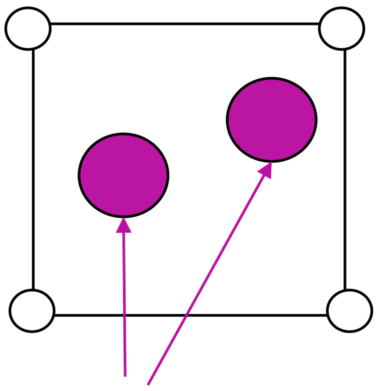
Seyedan & Sołowski (2017)

Material Point Method MPM

Initially:

In the material point method **all the data is stored with the material points**. There is no information on the grid, other than grid nodes position.

The constitutive models & laws, as well as the theory follows the **general framework of continuum mechanics**.



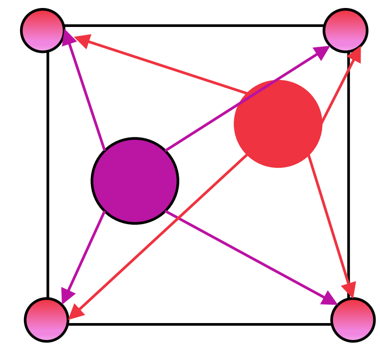
these folks keep all the data

Material Point Method MPM

Interaction happens on a background grid in each time-step.

Quantities are moved from material points to the grid, solved, and moved back…

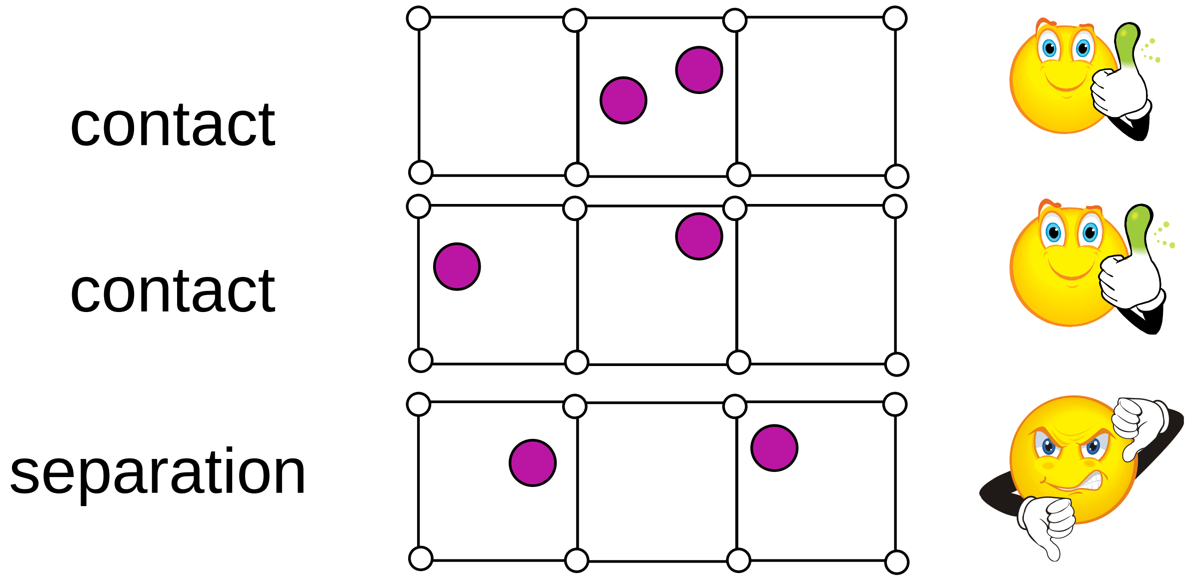
Very simple algorithm…



interaction thanks to the grid nodes

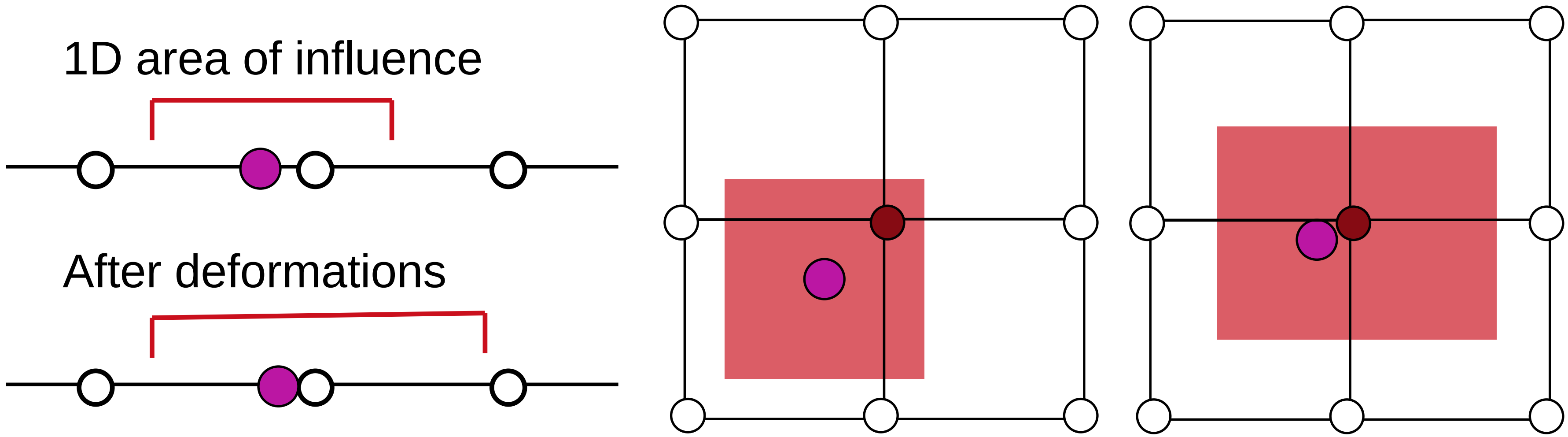
Important: material points need to share grid nodes to interact

Material Point Method MPM



Important: material points need to share grid nodes to interact

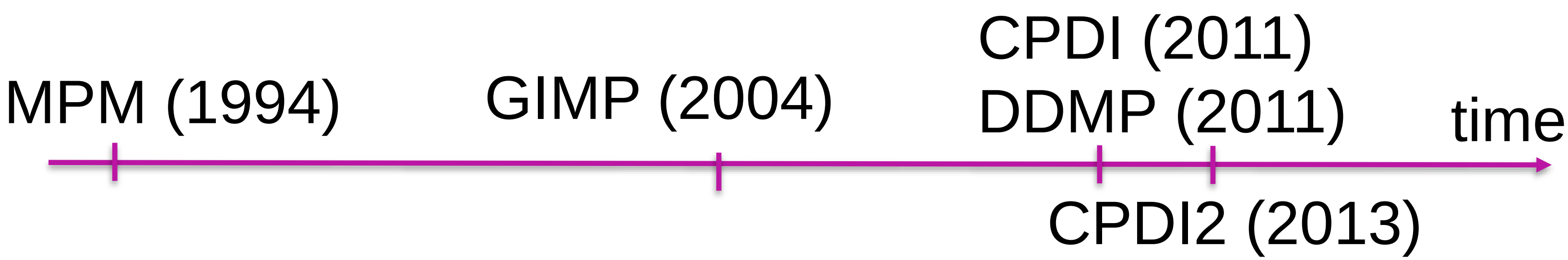
Material Point Method MPM



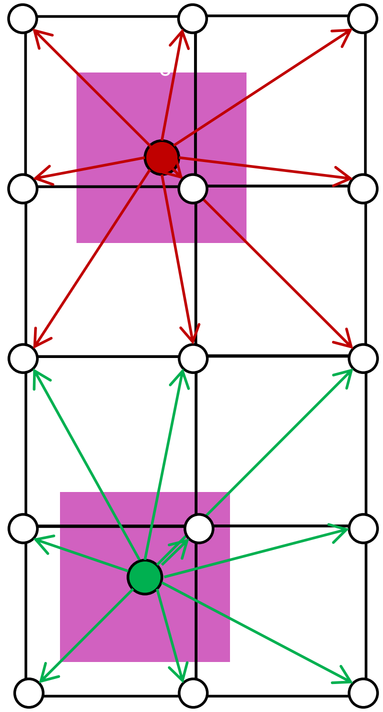
We can improve contact by introducing material point domains…

MPM – few remarks

- despite discretising material using material points, it is a continuous method
- due to the way material points interact with each other, discontinuities / fractures can happen in all the formulations but CPDI. In CPDI 2 the material separation is controlled, in other method is not controlled and very grid dependent.
- when the discontinuities appear in what is a continuous method, that leads to major mathematical challenges to prove convergence
- CPDI / CPDI 2 are used for convergence studies (Sulsky & Gong 2016)
    - arbitrary high order of convergence is the goal



Algorithm: Generalized Interpolation Material Point Method (GIMP)



Data transferred from the material points to the nodes:

- Material points cover some defined space
- Nodes „gather" information about material points within their domain
- Momentum (mass and velocity) transferred from the material points to the nodes
- Forces acting on nodes computed
- Nodal acceleration computed and velocity updated

numerical damping

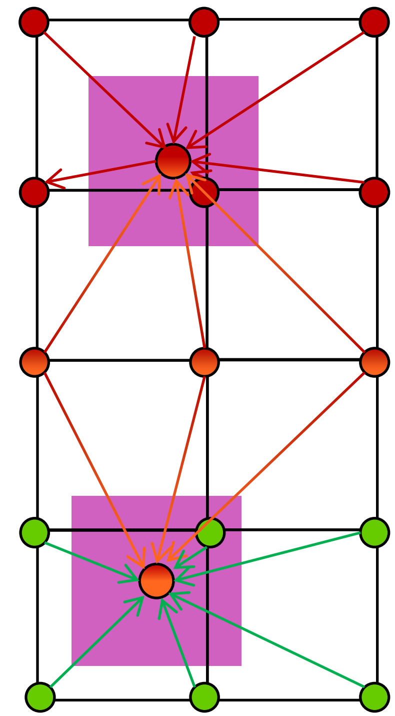
$$\mathbf{p}_n = \sum_p \mathbf{p}_p S_{np} \qquad \mathbf{v}_n = \frac{\mathbf{p}_n}{m_n} \qquad \mathbf{a}_n = \frac{\mathbf{f}_n^{\text{int}} + \mathbf{f}_n^b + \mathbf{f}_n^{ext}}{m_i} - c_d \cdot \mathbf{v}_n$$

$$\mathbf{v}_n^{new} = \mathbf{v}_n + \mathbf{a}_n \Delta t \qquad \mathbf{f}_n^{\text{int}} = -\sum_p \nabla S_{np} \cdot \boldsymbol{\sigma}_p V_p \qquad \mathbf{f}_n^b = \sum_p m_p \mathbf{b} S_{np}$$

Algorithm: Generalized Interpolation Material Point Method (GIMP)

Data in the nodes added, and transferred back to the material points

- Data in nodes is not kept
- Material points carry the data for the next step
- Material point velocities and positions updated using nodal acceleration and velocity respectively
- Velocities used to compute increments of strains and stresses
- Grid position usually not updated after time step



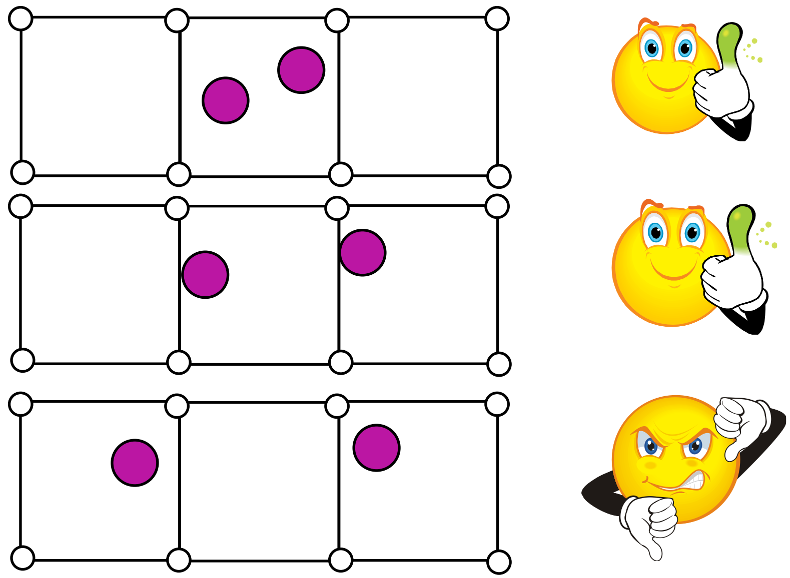
$$\mathbf{x}_p^{new} = \mathbf{x}_p + \sum_n S_{np}\mathbf{v}_n^{new}\Delta t \qquad \mathbf{v}_p^{new} = \mathbf{v}_p + \sum_n S_{np}\mathbf{a}_n\Delta t$$

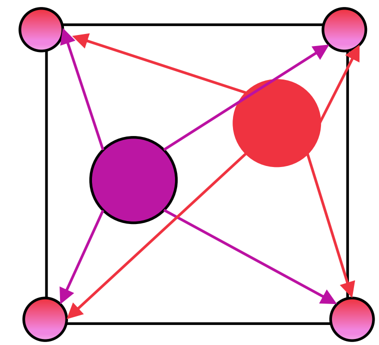
MPM - discontinuities

contact

contact

separation



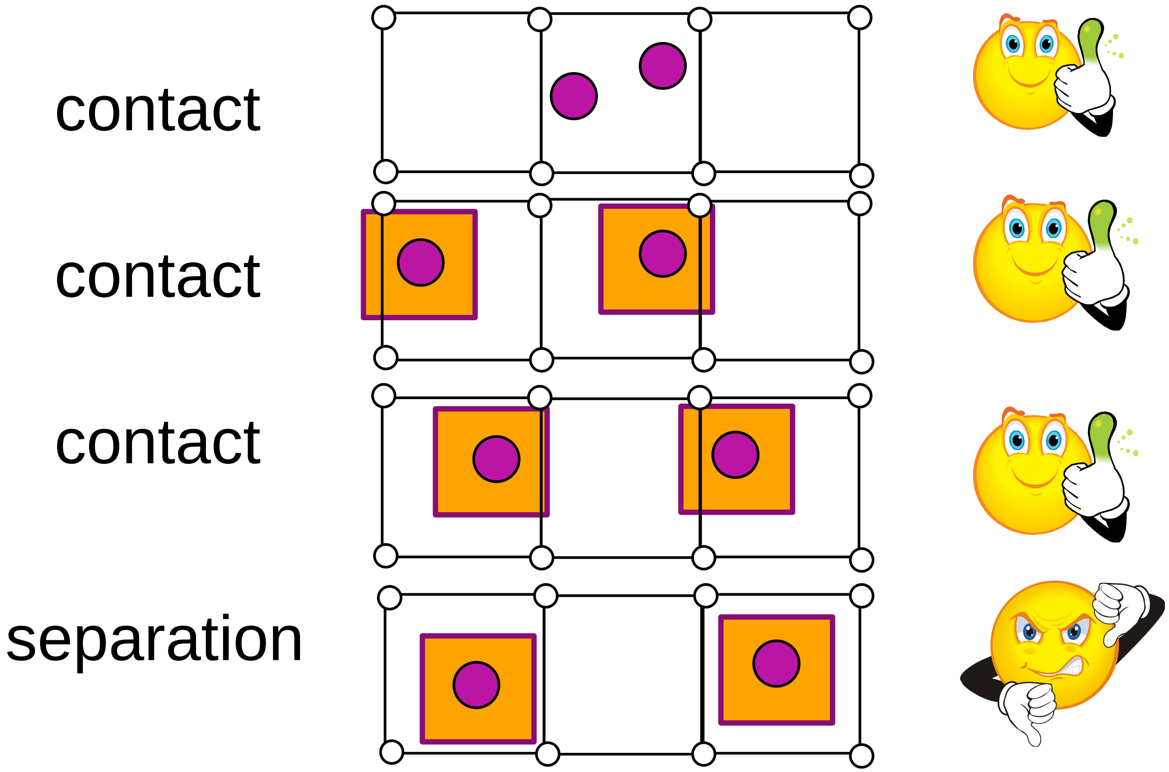


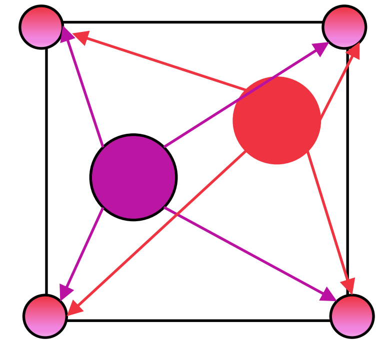
interaction thanks to the grid nodes

classical MPM – all data in a material point, no domain

The points interact only when they are in the same grid cell or at neighbouring grid cells. As such, the final results (and separation) is grid size and time step dependent.

GIMP - discontinuities

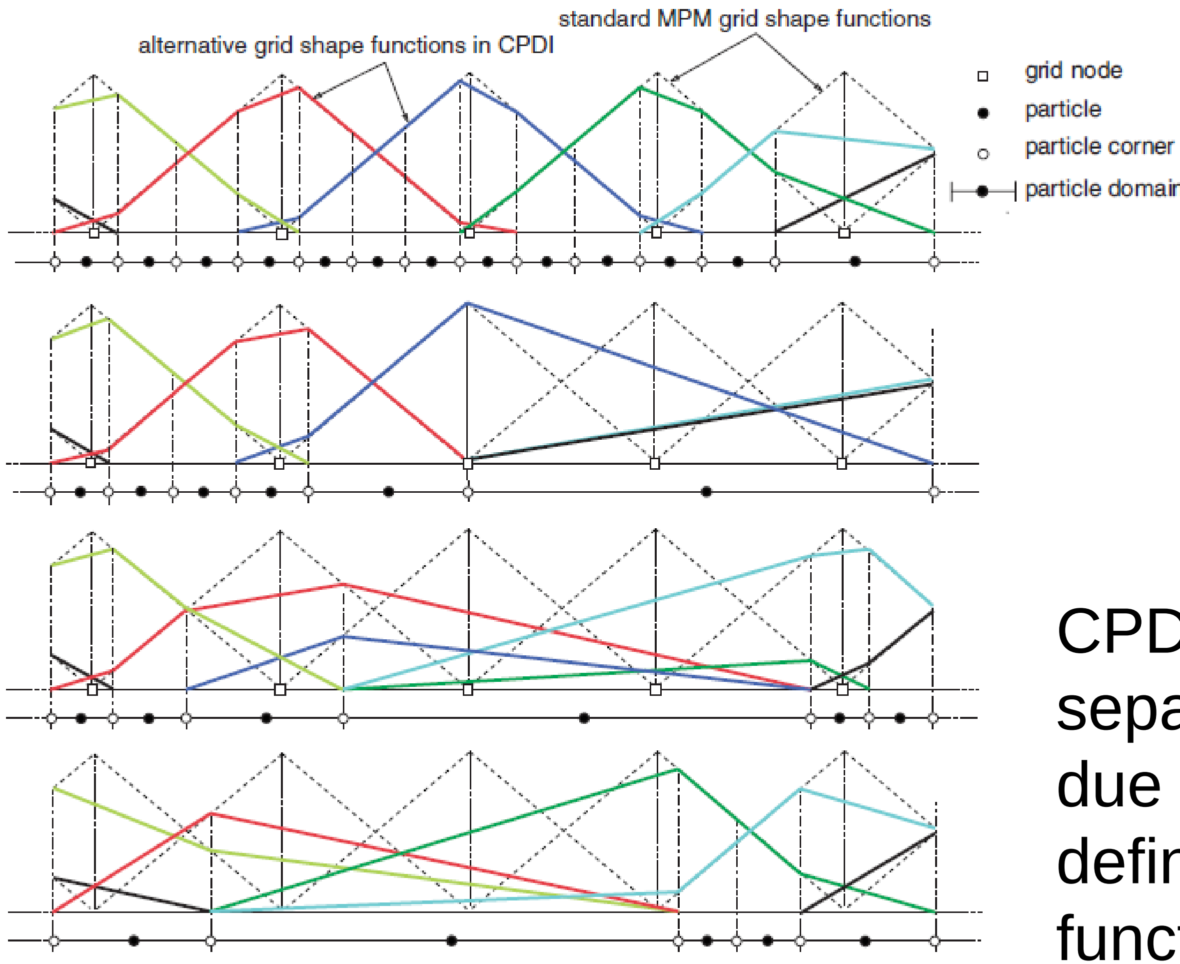


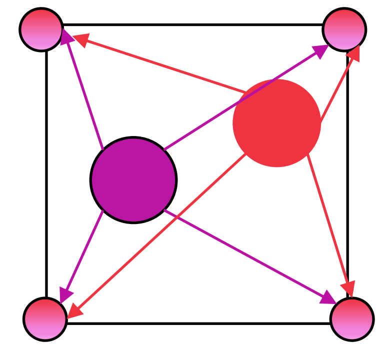


interaction thanks to the grid nodes

GIMP– all data in a material point, defined domain by a particle characteristic function

CPDI – no discontinuities





interaction thanks to the grid nodes

CPDI –no unwanted separation possible due to specially defined grid shape functions.

© Sadeghirad et al. 2013

MPM – grid shape functions

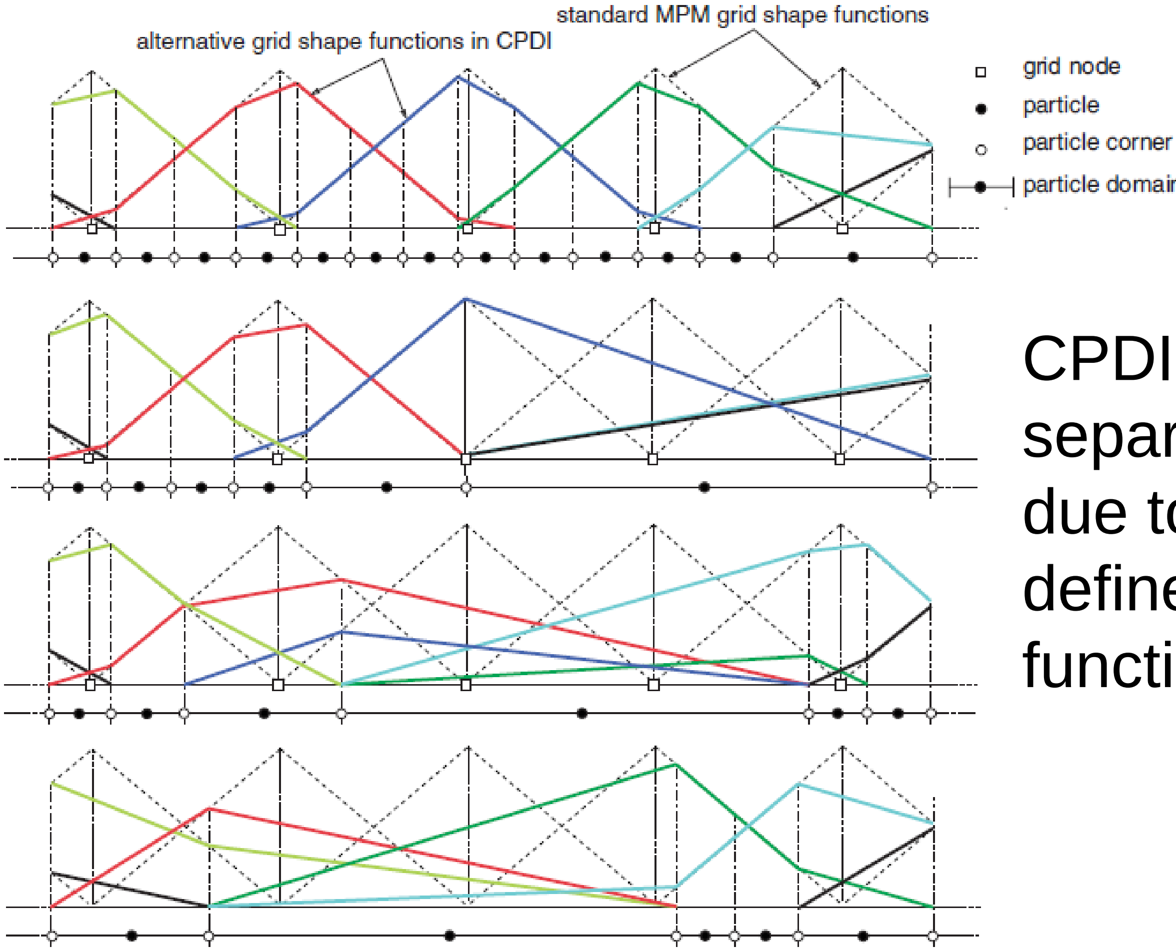
$$\int_{\Omega} \rho \mathbf{a} \cdot \delta \mathbf{v} d\mathbf{x} + \int_{\Omega} \boldsymbol{\sigma} : \nabla \delta \mathbf{v} d\mathbf{x} = \int_{\Omega} \rho \mathbf{b} \cdot \delta \mathbf{v} d\mathbf{x} + \int_{\Omega_\tau} \boldsymbol{\tau} \cdot \delta \mathbf{v} d\Omega_\tau$$

rate of change of momentum | internal force | body force | external force

The equation above is solved on the grid. Each computational step the data from the material points is transferred to the grid using the grid shape functions.

Grid shape function depends on the domain of the material point as well as on the chosen implementation of MPM

CPDI – no unwanted discontinuities



CPDI 2 –no unwanted separation possible due to specially defined grid shape functions.

© Sadeghirad et al. 2013

Thank you