Recent developments on MPM and its application in impact and explosion simulation

Xiong ZHANG
School of Aerospace
Tsinghua University
Email: xzhang@tsinghua.edu.cn
Web: http://xzhang.comdyn.cn

Acknowledgement: S Ma, P Huang, Z T Ma, Y P Lian
PF Yang, JG Li, Y T Zhang, X J Wang

Outline

- **Introduction**
- Our improvements on MPM
  - Efficient implementation of MPM
  - Improved contact method
  - Adaptive material point method
  - Parallel based on OpenMP
  - Material point finite element method (MPFEM)
  - Hybrid FE-MP method
- MPM3D – A 3D explicit MPM code
- Numerical Examples
- Concluding remarks

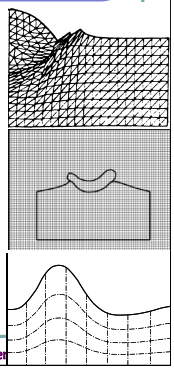
Introduction

- Intense impulsive loading (high velocity impact and blast)
  - Wave propagation
  - Strain rate
- Highly nonlinear
  - Boundary condition nonlinearity
  - Geometrical nonlinearity
  - Material nonlinearity
- Deformation is highly localized
  - Material in the impact region will be liquefied.



Introduction

- Numerical methods
  - Lagrangian method
  - Eulerian method
  - Hybrid Lagrangian-Eulerian methods
    - Arbitrary Lagrangian Eulerian (ALE)
    - Particle In Cell (PIC)
  - Meshfree methods
    - SPH
    - EFG
    - RKPM
    - MLPG
    - … …



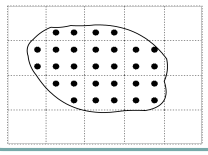
Introduction

- PIC (Particle In Cell, Harlow, 1955)
  - Represents a fluid by Lagrangian particles moving through a Eulerian computational grid
  - Highly successful
    - in tracking contact discontinuities
    - in modelling highly distorted flow
  - Disadvantages
    - too noisy
    - had high numerical viscosity (momentum diffusion)
    - large heat conduction (energy diffusion)

➔FLIP (Brackbill, 1986) — a low dissipation PIC method

Material Point Method

- FLIP ➔ MPM (Sulsky, Chen et. al. 1994)
  - An extension of the FLIP PIC method to solid mechanics
    - Invoke the constitutive equation at particles
    - Weak formulation consistent with the FE method
  - Fully Lagrangian particle method
    - Particles carry all state variables
    - Grid carries no permanent information



MPM discretizes a material domain by particles moving through an Eulerian background mesh.

Material Point Method

- In each time step
  - Particles are attached rigidly to a predefined regular grid
  - State variables of particles are mapped to the grid nodes to establish their momentum equations

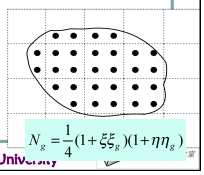
FEM

$$p_{ig} = \sum_p m_p v_{ip} N_{gp} \qquad f_{ig}^{\text{int}} = -\sum_p N_{gp,j} \sigma_{ijp} \frac{m_p}{\rho_p}$$

  - After solving momentum equations on grid nodes, results are mapped back to the particles to update their states.

$$\bar{\boldsymbol{v}}_p^{k+1} = \sum_{g=1}^{g} \frac{\boldsymbol{p}_g^{k+1}}{m_g^k} S_{gp}^k$$

  - The deformed grid is discarded, and a new regular grid is used
    - Mesh distortion is avoided



$$N_g = \frac{1}{4}(1+\xi\xi_g)(1+\eta\eta_g)$$

Material Point Method

- Combines the advantages of Lagrangian and Eulerian methods and avoid their disadvantages
  - History-dependent constitutive models
  - Free surfaces or multiple materials
  - No mesh distortion and element entanglement
  - low dissipation
  - Promising in modeling extremely large deformation and material discontinuities

S Ma, X Zhang, et al. ***IJIE***, 36: 272-282, 2009

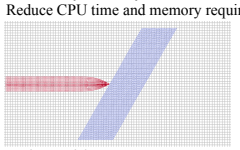
- Much more efficient than SPH
  - The critical time step size depend on the cell size
  - No neighboring particles search

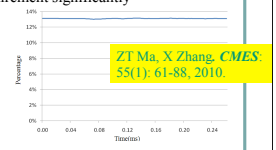
Outline

- Introduction
- **Our improvements on MPM**
  - Efficient implementation of MPM
  - Improved contact method
  - Adaptive material point method
  - Parallel based on OpenMP
  - Material point finite element method (MPFEM)
  - Hybrid FE-MP method
- MPM3D – A 3D explicit MPM code
- Numerical Examples
- Concluding remarks

Efficient implementation of MPM

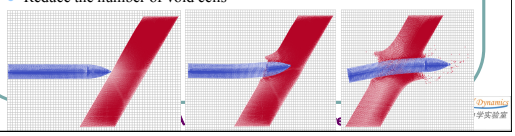
- Dynamic grid
  - Cells are dynamically created when occupied by at least one particle
  - Reduce CPU time and memory requirement significantly





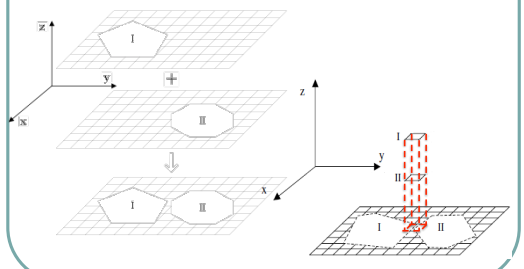
ZT Ma, X Zhang, ***CMES***: 55(1): 61-88, 2010.

- Moving grid
  - Adjust grid according to the position of particles
  - Reduce the number of void cells



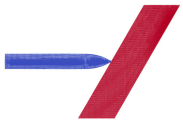
Efficient implementation of MPM

- Local multi-mesh
  - Multi-meshes are only used at contacted grid nodes



Efficient implementation of MPM

- Projectile: Steel
  - L: 88.9 mm
  - D: 12.9 mm
  - v: 575 m/s
  - N_p: 13,314
- Target: Aluminum
  - t: 26.3 mm
  - Angle of incidence: 30°
  - N_p: 187,550



Contact	N_p / N_g	CPU time per step (s)			Memory used (MB)		
		original	new	Reduced	original	new	Reduced
off	200,864/	0.5455	0.4064	34.2%	73.18	54.25	25.9%
on	170,688	0.8146	0.4313	88.8%	109.36	55.02	49.7%

- 3 big disks and 207 small disks
- Total particles number: 52,730



Improved contact algorithm

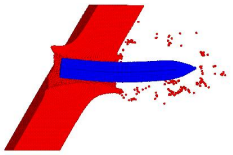
- Unit normal outward Bardenhagen(2000)

$$\hat{\mathbf{n}}_{bi} = \frac{1}{\left|\sum_p \mathbf{G}_{ip} m_{bp}\right|} \sum_p \mathbf{G}_{ip} m_{bp}$$

P Huang, X Zhang, et al. *IJNME*, 2010, in press

Problem in simulation:

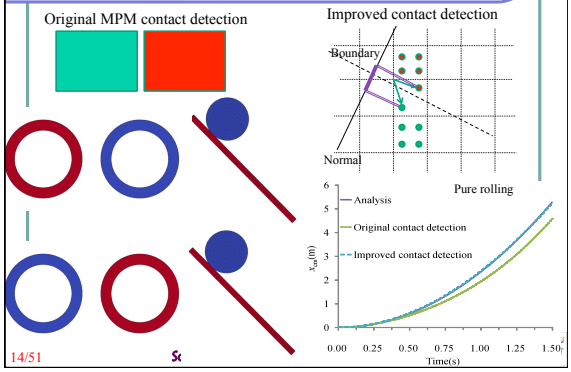
- Normal vectors at contact surface are not collinear;
- Cause non-conservation of momentum and penetration phenomena;
- The calculated normal vectors is wrong for crushing target;



❖ A new method for normal vector calculation

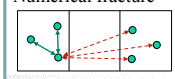
Satisfy the collinearity condition which insures the momentum conservation

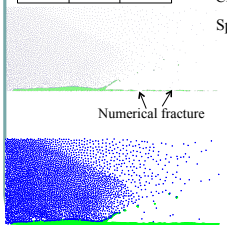
Improved contact algorithm



Adaptive Material Point Method

Numerical fracture





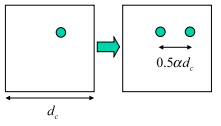
- Adaptive particle split scheme

Accumulated strain: $\varepsilon_{xx} = \sum_k \Delta\varepsilon_{xx}^k = \sum_k D_{xx}^k \Delta t$

Characteristic particle length: $L_{px} = (1+\varepsilon_{xx})\sqrt[3]{V_0}$

Splitting criterion: $L_{px} > \alpha d_c$

Split: mass, volume, internal energy

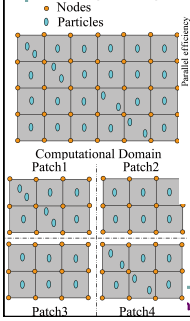


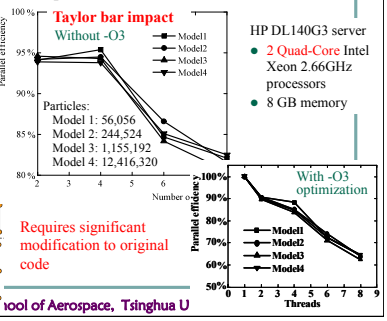
S Ma, X Zhang, et al. *CMES* 2009; 39(2):101-123

Parallel MPM based on OpenMP

- Symmetric multiprocessing (SMP)
- Background grid decomposition

P Huang, X Zhang, et al. *CMES* 2008; 38(2):119-148





HP DL140G3 server

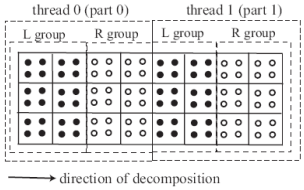
- 2 Quad-Core Intel Xeon 2.66GHz processors
- 8 GB memory

Requires significant modification to original code

Parallel MPM based on OpenMP

Alternated grid updating method to avoid complex manipulation of subgrid

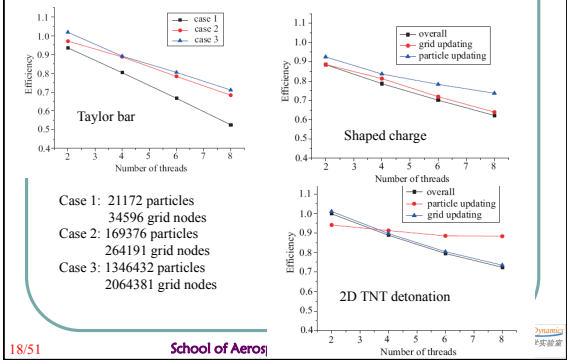
1. divide the computing domain along the direction selected
2. each part is further divided into two parts
3. each thread update its **L** group
4. a barrier is set
5. each thread upate its **R** group



Requires minor modification to original code

MPI + OpenMP hybrid parallel algorithm is in progress.

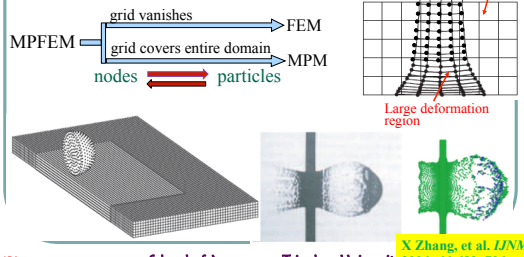
Parallel MPM based on OpenMP



Case 1: 21172 particles
34596 grid nodes
Case 2: 169376 particles
264191 grid nodes
Case 3: 1346432 particles
2064381 grid nodes

Material Point Finite Element Method

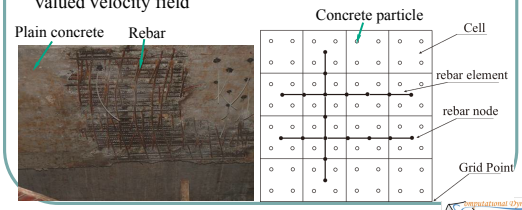
- Material domain is discretized by a mesh of finite elements
- Momentum equations — integrated explicitly on
  - Background grid — large deformation zone
  - FE mesh — elsewhere



X Zhang, et al. *IJNME* 2006; 66:689–706

Hybrid FEMP method

- Plain concrete is discretized into particles
- Rebars are discretized into truss elements (rebar elements)
  - Rebar nodes carry kinematic variables
  - Rebar elements carry history variables
- All the rebar nodes and particles move in the same single-valued velocity field



Outline

- Introduction
- Material point method
- Our improvements on MPM
  - Material point finite element method (MPFEM)
  - Adaptive material point method
  - Efficient implementation of MPM
  - Parallel based on OpenMP
  - Improved contact method
  - Hybrid FE-MP method
- **MPM3D – A 3D explicit MPM code**
- Numerical Examples
- Concluding remarks

MPM3D - 3D explicit MPM code

- MPM3D solver
  - Developed using C++
  - Material models
    - Elasticity, plasticity (hardening, temperature, strain rate)
    - Fluid, High explosive
    - Concrete, Ceramic, rubber, Soil …
  - Equation of states
    - Solid
    - Fluid
    - High explosive
  - Failure models
    - Hydrodynamic tensile failure; Effective plastic strain
    - Principal stress failure; Principal strain failure

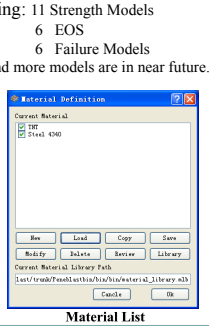
MPM3D - 3D explicit MPM code

- Graphical User Interface (GUI)
  - Preprocessor
    - Create 3D solid models & Particle discretization
    - Setup solution parameters
    - Generate input data for solver
  - Solution process monitor
    - Call MPM3D solver to solve the model
    - Real Time Monitoring (Energy, momentum, animation)
    - Interactive steering (Pause/Resume/Restart/Stop)
    - Auto-testing
  - Postprocessor
    - Solver ➔ ParaView (http://www.paraview.org)
  - Cross platform (Windows, Linux, Mac OS X)

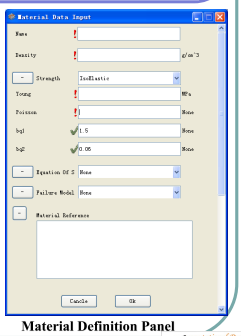
Preprocessor
Material Definition

MPM3D - 3D explicit MPM code

Including: 11 Strength Models
6 EOS
6 Failure Models
More and more models are in near future.



Material List

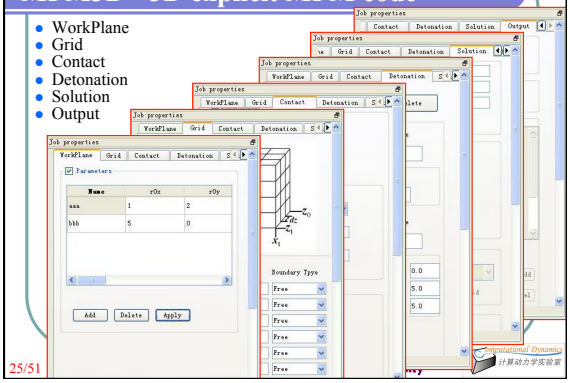


Material Definition Panel

Preprocessor – properties setting

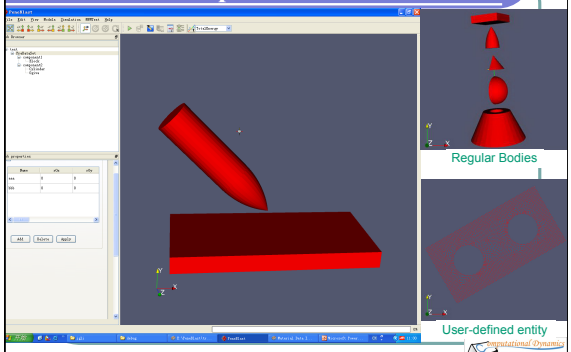
MPM3D - 3D explicit MPM code

- WorkPlane
- Grid
- Contact
- Detonation
- Solution
- Output



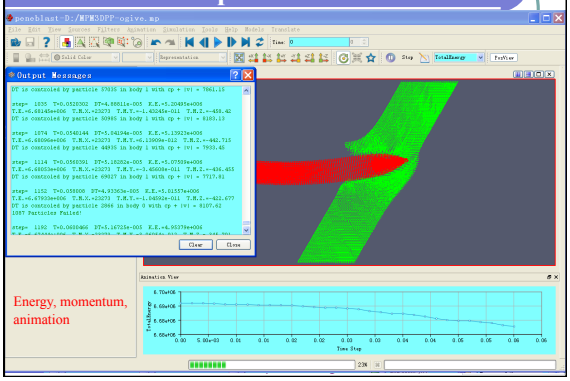
Preprocessor - Modeling

MPM3D - 3D explicit MPM code



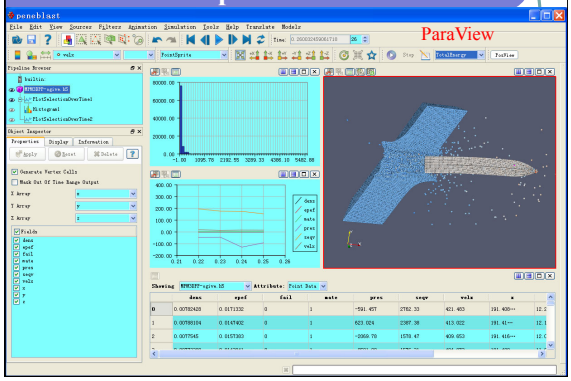
Real time monitoring

MPM3D - 3D explicit MPM code



Postprocessor - ParaView

MPM3D - 3D explicit MPM code



Outline

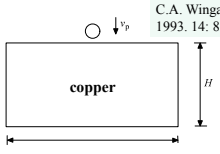
- Introduction
- Material point method
- Our improvements on MPM
  - Efficient implementation of MPM
  - Improved contact method
  - Adaptive material point method
  - Parallel based on OpenMP
  - Material point finite element method (MPFEM)
  - Hybrid FE-MP method
- MPM3D – A 3D explicit MPM code
- **Numerical Examples**
- Concluding remarks

Numerical examples

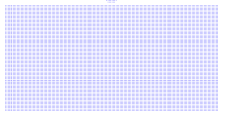
Crater forming

- Impact velocity：6.0 km/s
- Projectile mass：0.5 g
- L = 80mm, H = 40mm
- 1,024,224 particles
- Run 1h 55m in P4 2.8G PC for simulation time of 32ms
- Time step size = 46.2 μs
- Steps: 692 steps

C.A. Wingate et. al. ***IJIE***, 1993. 14: 819-830

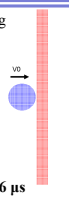


	Deepness (mm)	Crater diameter (mm)	Deepness/ Diameter
Experiment*	14	25.4	55
EPIC	18	24	75
MESA	15.9	28	57
SPH	17.3	26	67
CALE	15.1	24.4	62
MPM3D	14	25	56



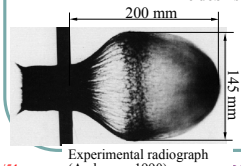
Numerical examples — Debris cloud

- Projectile: Lead sphere, r =7.5mm, m = 20g
- Target: Lead plate, t = 6.35 mm
- Impact velocity: v_0 = 6.58 km/s
- Material model
  - Elastic-plastic constitutive model
  - Mie-Grüneisen EOS
- Material failure model
  - Tensile Failure (P_{min} = −1500 MPa)
  - Plastic Strain Failure (ε_{pmax} = 3.0)

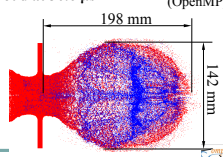


2 Quad-Core Xeon 2.66 GHz processors (OpenMP): 1.93h

The debris cloud at 30.6 μs



Experimental radiograph (Anderson, 1990)



13,542,030 particles

X Zhang, xzhang@tsinghua.edu.cn

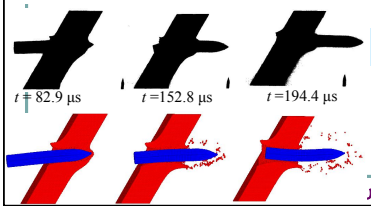
Numerical examples — Penetration

Projectile: Steel
Length: 88.9 mm
Diamete: 12.9 mm
Velocity: 575 m/s
Particles: 13,314

Target: Aluminium
Thickness: 26.3mm
Inclination: 30°
Particles: 187,550



Projectile's residual velocity
Experiment: 455 m/s
MPM3D: 453.4 m/s

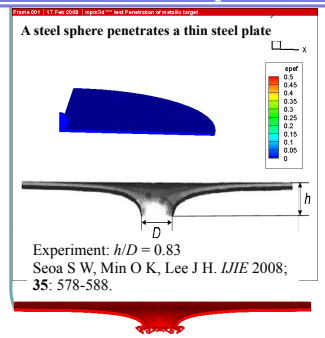


Piekutowski, Forrestal et al. ***IJIE*** 1996; 18: 877-887

X Zhang, xzhang@tsinghua.edu.cn

Numerical examples — Penetration

A steel sphere penetrates a thin steel plate



Experiment: h/D = 0.83
Seoa S W, Min O K, Lee J H. *IJIE* 2008; **35**: 578-588.

MPM3D: h/D = 0.87

Steel sphere
Diameter: 10mm
Velocity: 200m/s
Particles: 32,768

Steel plate:
Thickness: 1mm
Diameter: 178mm
Particles: 398,184

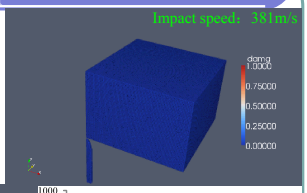
Effective plastic strain at failure: 0.5

X Zhang, xzhang@tsinghua.edu.cn

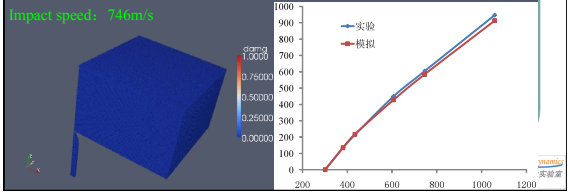
Numerical examples — Penetration: Plain Concrete

Projectile
Length: 112mm
Diameter: 15mm
45＃steel
Target
Thickness: 250mm
Concrete

Impact speed：381m/s



Impact speed：746m/s



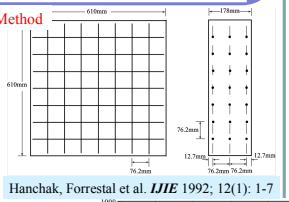
X Zhang, xzhang@tsinghua.edu.cn

Numerical examples — Penetration: C48 RC

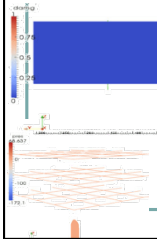
Hybrid Finite Element Material Point Method

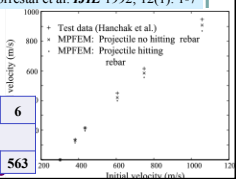
Projectile: Steel
Length: 143.1mm
Diamete: 25.4 mm
Velocity: 746 m/s

Target: Reinforced Concrete
Thickness: 178mm



Hanchak, Forrestal et al. ***IJIE*** 1992; 12(1): 1-7





Cell size(mm)	4	5	6
Experiment	615		
MPM3D	585	571	563

X Zhang, xzhang@tsinghua.edu.cn

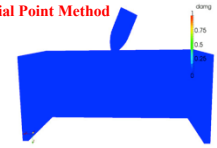
Numerical examples — Reinforced Concrete

Impact Velocity (m/s)	Residual Velocity (m/s)		Perforation Pore Diameter (mm)	
	Experiment	MPM3D	Experiment	MPM3D
257.5	100.7	100	780×710	900×1000
248.3	83.7	83	930×730	1000×1000

MPM3D: Hybrid Finite Element Material Point Method



Impact velocity: 257.5 m/s
Incident angle: 1°



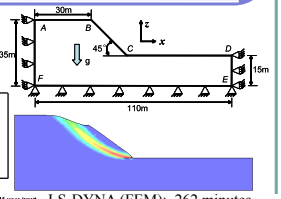
Impact velocity: 248.3 m/s
Incident angle: 24.5°

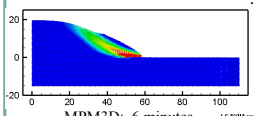
Numerical examples — Rock-soil slope failure

- **Cohesive soil**

$\phi = 20°$; $\psi = 0^0$; $c = 10.0$ kPa

E=70MPa, ν=0.3, ρ=2.1 g/cm3





MPM3D: 6 minutes

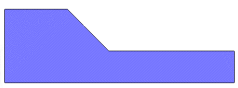
LS-DYNA (FEM): 262 minutes

- **Non-cohesive soil**

$\phi = 20°$; $c = 0.1$kPa; $\psi = 0^0$

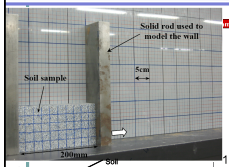


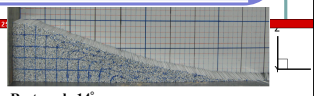
MPM：6 minutes



LS-DYNA (FEM): 19.8 hours

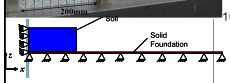
Numerical examples — Rock-soil slope failure





Rest angle 14°

Bui H H, Japan: Ritsumeikan University, 2006



K=0.7MPa，ρ=2.65 g/cm^3

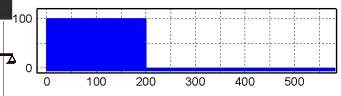
$\phi = 19.8°$; $c = 0$kPa; $\psi = 0°$

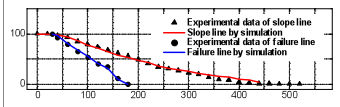
25, 600 particles

Rigid-soft contact algorithm

Foundation is a rigid body

Friction is considered





MPM, slide time 0.64s，rest angle14°

Numerical examples — Cohesive soil impacting

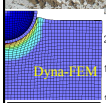


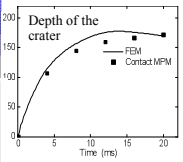
Half spherical shell:
Diameter: 0.408m
Mass: 12kg
Al alloy

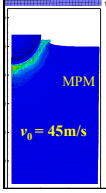
Fasanella E L, et al. AIAA, 2001

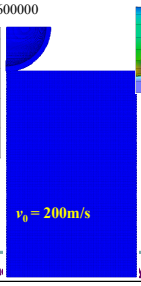
Particles for spherical shell: 26385
Particles for target: 1600000

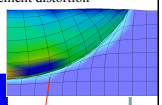
FEM calculation terminated at 0.62ms (98mm) due to element distortion

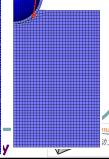




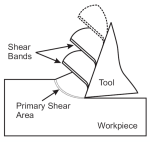


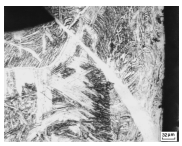


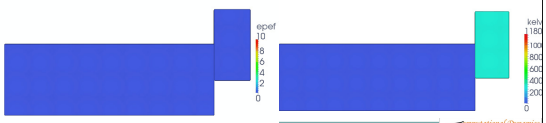




Numerical examples — Metal cutting process



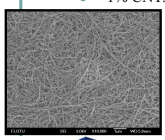


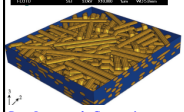


School of Aerospace, Tsinghua University

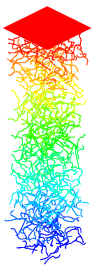
Numerical examples — Carbon Nanotube Composite

- Carbon nanotube
  - Exceptionally high stiffness, strength and resilience
  - Ideal to act as the reinforcing materials
    - 1% CNTs ➔ stiffness ↑40%, tensile strength ↑ 25%

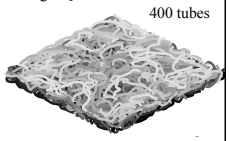




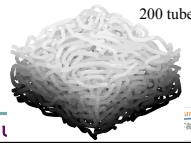
Pan, Iorga et al. ***Composites Science and Technology*** 68: 2792-2798, 2008



400 tubes



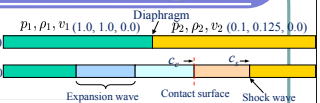
200 tubes

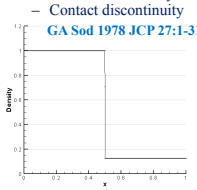


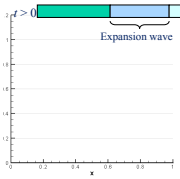
Numerical examples — Shock tube problem

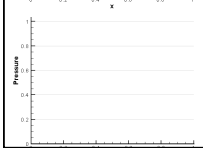
- Test code for compressible fluid
  - Shock discontinuity
  - Contact discontinuity

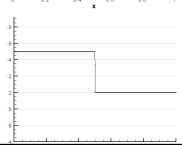
GA Sod 1978 JCP 27:1-31











GIMP
2000+1000 particles
1000 cells

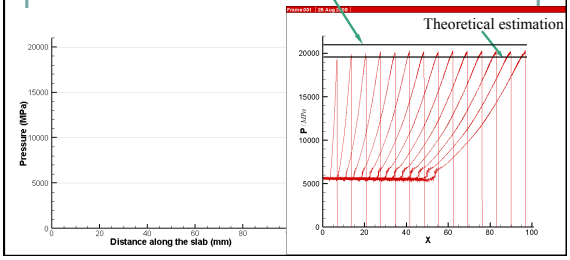
Numerical examples — 1-D TNT slab detonation

Slab length: 0.1m
MPM3D : 1000 particles, 500 cells
Detonation velocity: 6930m/s
JWL Equation of state

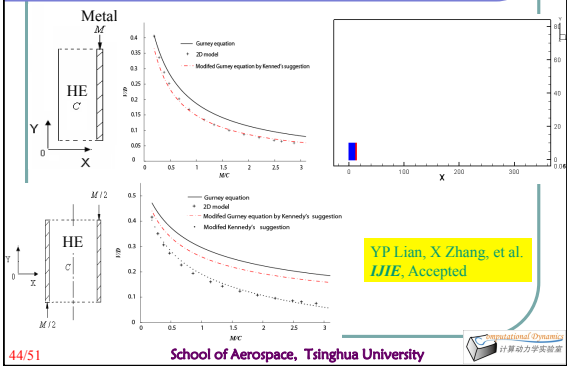
MPM3D (GIMP):
1000 particles, 500 cells
2000 particles, 1000 cells

Experiment

Theoretical estimation



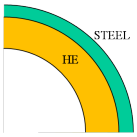
Numerical examples — Explosive driven Metal



YP Lian, X Zhang, et al.
IJIE, Accepted

Numerical examples — Fragmentation driven by explosion

Metal shell driven by High explosive



Experiment: 汤铁钢, 谷岩, 李庆忠. 爆轰加载下金属柱壳膨胀破裂过程研究. 爆炸与冲击, 2003, 23:529-533

	Experiment	MPM3D with Weibull	MPM3D with Gurson
Fragmentation time (μs)	15.4	15	16.3


Gurson model

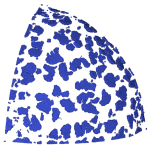

Weibull random failure

Numerical examples — Fragmentation driven by explosion

Weibull random failure

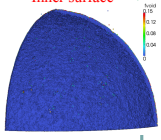


Gurson model



Void distribution at 15 μs

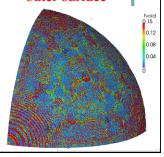
Inner surface



outer surface

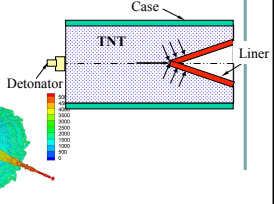


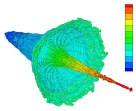




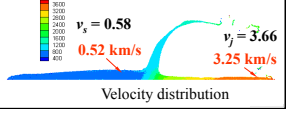
Numerical examples — Shaped charge

- Shaped charge
  - A cylinder of high explosive with a cone-shape thin metal liner in one end
  - Intense pressure ➔ Collapse of the liner ➔ A jet with extreme high velocity (> 10 km/s)
- Application
  - Armor penetration
  - Metal cutting
  - Oil well perforation









Velocity distribution

Outline

- Introduction
- Our improvements on MPM
  - Efficient implementation of MPM
  - Improved contact method
  - Adaptive material point method
  - Parallel based on OpenMP
  - Material point finite element method (MPFEM)
  - Hybrid FE-MP method
- MPM3D - 3D explicit MPM code
- Numerical Examples
- **Concluding remarks**

Concluding Remarks

- MPM combines the advantages of Lagrangian and Eulerian methods
- Very efficient and simple in solving high velocity impact and explosion problems
- Promising in the numerical simulation of material behaviour subject to severe deformation and distortion