GPU Acceleration of the Material Point Method

Fabian Meyer

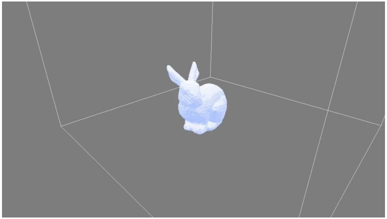
University of Koblenz

Kolloquium Computergrafik, 14 February 2019

A Brief MPM Overview: Do You Want to Build a Snowman?

A short historical summary of MPM:

- Belongs to family of particle-in-cell(PIC) techniques [EHB57].
- Initial application to solids [SZS95] → MPM
- From research to production in *Disney's* animation film *Frozen* [Sto+13].
- Avalanche research [Gau+18]



Video result of my bachelor thesis on the simulation of snow [Mey15].

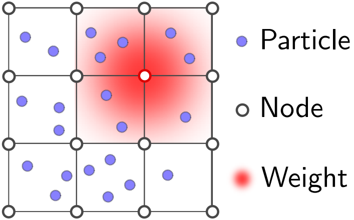
PIC ideas:

- Combine Lagrangian particles & Eulerian grid
- Particles store all information

Typical PIC/MPM roundtrip:

1. Particle-to-grid(P2G) transfer to an unmoving grid
2. Solve discretized governing equations on grid
3. Grid-to-particle(G2P) transfer back to particles & move them

$\Rightarrow$ **meshfree, non-empirical**



Transfers: Interpolation functions are defined over grid nodes.

GPGPU for performance enthusiasts

Why would('nt) you?

Drawbacks:

- Interactivity much easier on CPU, but slow PCI-Bus communication
- Code is mostly written against GPU architecture
- A lot of strain on the programmer

Benefits:

- Data is already on the GPU for rendering
- **Higher parallelization acceleration**

Governing Equations: Conservation of Mass & Momentum

Conservation of mass, continuum assumption holds.
Lagrangian (moving with a particle ${}_0\boldsymbol{x}$):

$$ {}_0^tJ\rho({}_0\boldsymbol{x}, t) = \rho({}_0\boldsymbol{x}, 0). \tag{1} $$

Eulerian (outside observer ${}_t\boldsymbol{x}$):

$$ \frac{\partial}{\partial t}\rho({}_t\boldsymbol{x}, t) = -\vec{\nabla} \cdot (\rho({}_t\boldsymbol{x}, t)\boldsymbol{v}({}_t\boldsymbol{x}, t)). \tag{2} $$

Lagrangian and Eulerian view measure differently but give same results. Equations are given in the strong form! [Jia+16][Abe12]

Conservation of momentum:
Lagrangian (moving with a particle ${}_0\boldsymbol{x}$):

$$\rho({}_0\boldsymbol{x}, 0)\boldsymbol{a}({}_0\boldsymbol{x}, t) = \vec{\nabla} \cdot \boldsymbol{P}({}_0\boldsymbol{x}, t) + \boldsymbol{f}^{\text{body}}({}_0\boldsymbol{x}, t){}_0^tJ. \tag{3}$$

Eulerian (outside observer ${}_t\boldsymbol{x}$):

$$\rho({}_t\boldsymbol{x}, t)\boldsymbol{a}({}_t\boldsymbol{x}, t) = \vec{\nabla} \cdot \boldsymbol{\sigma}({}_t\boldsymbol{x}, t) + \boldsymbol{f}^{\text{body}}({}_t\boldsymbol{x}, t) \tag{4}$$

Solving this equation will tell us how the velocity fields $\boldsymbol{v}({}_t\boldsymbol{x}), \boldsymbol{v}({}_0\boldsymbol{x})$ change on the whole domain due to acceleration $\boldsymbol{a}$. This is important to advect particles accounting for all forces. [Jia+16][Abe12]

The Pretty Strong but Mathematically Weak Formulation

$$\int_{\Omega^0} {}_0\boldsymbol{q} \cdot \left[({}_0\rho_0)({}_0\boldsymbol{a}) - {}_t\boldsymbol{f}^{\text{body}} \right] d_0\boldsymbol{x} = \int_{\partial\Omega^{t^n}} {}_t\boldsymbol{q} \cdot \boldsymbol{\sigma} \, d_t\boldsymbol{A} - \int_{\Omega^{t^n}} \nabla_t \boldsymbol{q} : \boldsymbol{\sigma} \, d_t\boldsymbol{x} \tag{5}$$

hi

1. [Jia+16]

hi

1. [Jia+16]

Rohan Abeyaratne. *Volume II of Lecture Notes on,The Mechanics of Elastic Solids: Continuum Mechanics*. http://web.mit.edu/abeyaratne/Volumes/RCA_Vol_II.pdf. [Online; accessed 08-November-2018]. MIT Department of Mechanical Engineering, 2012.

Martha W Evans, Francis H Harlow, and Eleazer Bromberg. *The particle-in-cell method for hydrodynamic calculations*. Tech. rep. LOS ALAMOS NATIONAL LAB NM, 1957.

Johan Gaume et al. „Unified modeling of the release and flow of snow avalanches using the Material Point Method“. In: Aug. 2018.

Chenfanfu Jiang et al. „The material point method for simulating continuum materials“. In: *ACM SIGGRAPH 2016 Courses*. ACM. 2016, p. 24.

Fabian Meyer. „Simulation von Schnee“. Bachelor's Thesis. Universität Koblenz-Landau, Institut für Computervisualistik, 2015.

Alexey Stomakhin et al. „A material point method for snow simulation“. In: *ACM Transactions on Graphics (TOG)* 32.4 (2013), p. 102.

Deborah Sulsky, Shi-Jian Zhou, and Howard L Schreyer. „Application of a particle-in-cell method to solid mechanics“. In: *Computer physics communications* 87.1-2 (1995), pp. 236–252.