Simulation of paintball shot

A project by Kevin Wallimann, Andri Schmidt and Marc Maetz

December 18, 2013

- Initial project idea:
  Paintball shot on wall
- Final project:
  Paintball shot on simple objects

Video

Expected challenges

- OpenGL
  - showing simple objects
- SPH solver
  - time to make it work as expected

Approach

- SPH solver
- Collision handling with axis aligned boundary boxes

Why?

- SPH solver
  - splashes and droplets
- Collision handling with axis aligned boundary boxes
  - lately attacket
  - risk of heavy time investition

Low level

- C++ 11
- Eigen
- OpenGL
  - GLSL (OpenGL Shading Language)
  - GLM (OpenGL Mathematics)

High level

1. Shot with gravitation
2. Collision detection
3. On impact the SPH simulation starts

Difficulties

General difficulties

- general code debugging
- parameter tuning
- coordinates: simulation space $\leftrightarrow$ window space

Difficulties

General difficulties

- general code debugging
- parameter tuning
- coordinates: simulation space $\leftrightarrow$ window space

Specific difficulties

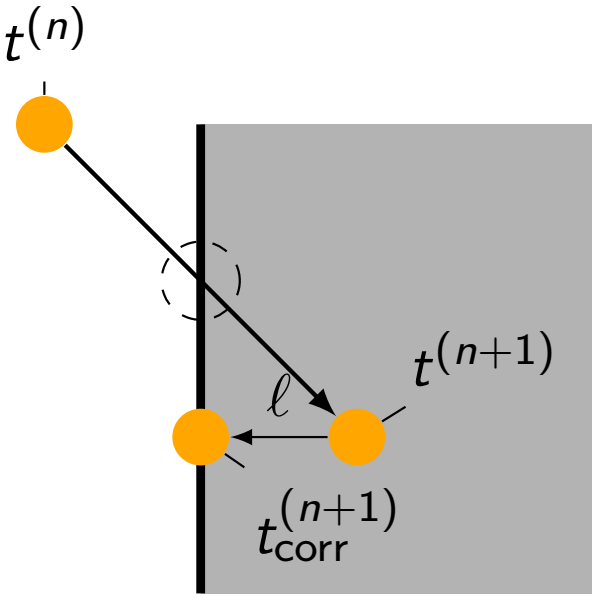
- many particles behaving like a ball
- particles pushed and trapped inside object
- sticky paint on object with given resolution

Tricks

- many particles behaving like a ball
  - $\rightarrow$ particle-particle forces ignored before collision
- particles pushed and trapped inside object
  - $\rightarrow$ projection further than on surface
- sticky paint on object
  - $\rightarrow$ velocity dependent trace on object

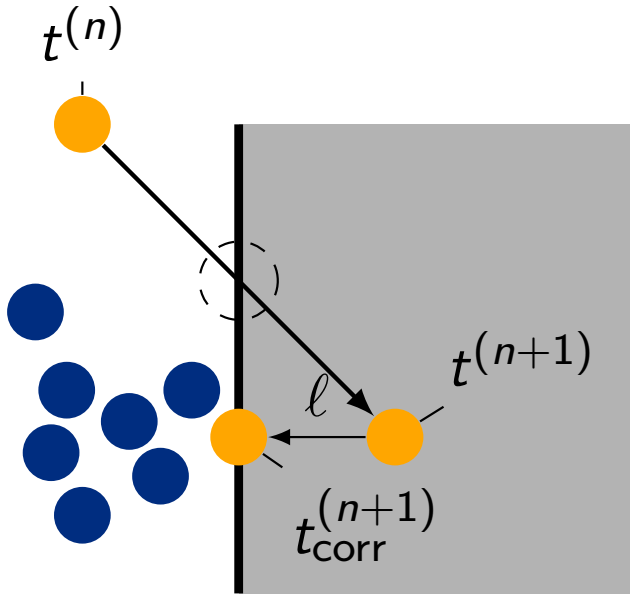
Particles pushed and trapped inside object

naive method



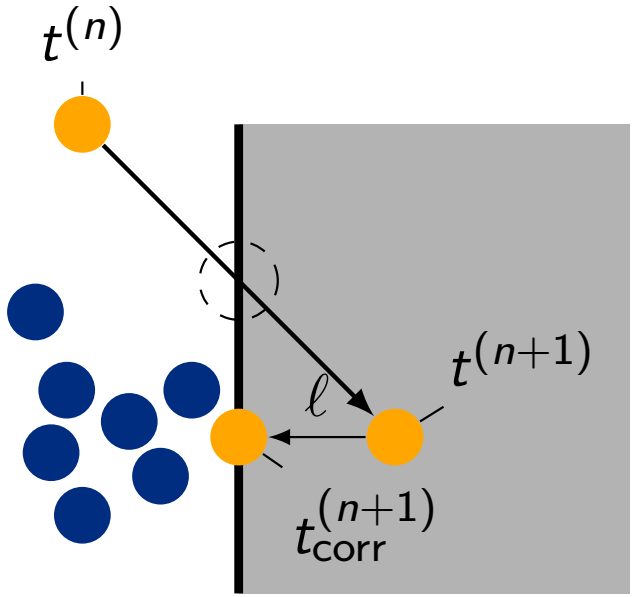
Particles pushed and trapped inside object

naive method

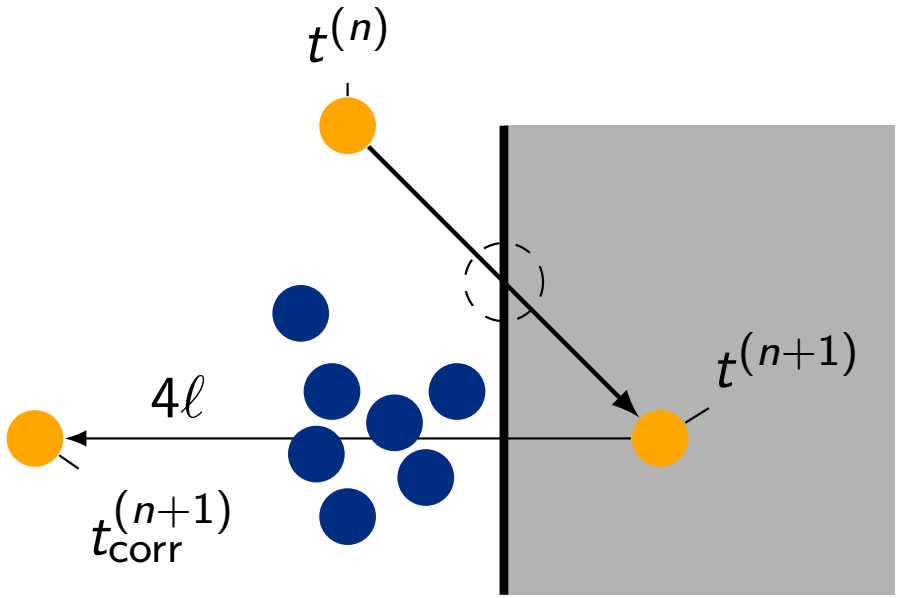


Particles pushed and trapped inside object

naive method



our method

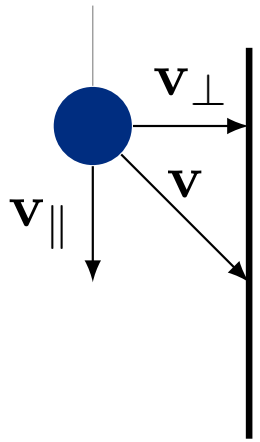


Sticky paint on object

Trace handling

Velocity split

SPH particle



Resulting trace

$\mathbf{v}_\parallel = 0$
$|\mathbf{v}|$ big

$\mathbf{v}_\parallel \neq 0$
$|\mathbf{v}|$ big

$\mathbf{v}_\parallel = 0$
$|\mathbf{v}|$ small

$\mathbf{v}_\parallel \neq 0$
$|\mathbf{v}|$ small

Demonstration

Performance

- Real time simulation for simple objects (500 vertices)
- For complex objects substantionally slower

Performance

- Real time simulation for simple objects (500 vertices)
- For complex objects substantionally slower

Limitations

- Resolution and number of collision objects
  - $\rightarrow$ bottleneck collision handling
- Number of particles
  - $\rightarrow$ bottleneck SPH simulation

What we learned

- OpenGL
- Blender

What we learned

- OpenGL
- Blender

What we would do different

- Start collision handling earlier
- Better time management