Position Based Dynamics

Matthias Müller et al.
VRIPHYS 2006

Presented by Eunki Hong
2014. 10. 21

Computer Graphics @ Korea University
Index

• Introduction
• Related Work
• Position Based Simulation
  ▪ Algorithm Overview
  ▪ Constraint Projection
  ▪ Collision Response
• Cloth Simulation
  ▪ Representation of Cloth
  ▪ Self Collision
  ▪ Cloth Balloons
• Result
Introduction

• Traditional approach to simulation dynamic objects had been work with forces
  \[ F = ma \rightarrow \text{Calculate accelerations} \rightarrow \text{Error} \]

• This paper use only constraint and position

• Main Feature of PBD
  - Control over explicit integration.
  - Removes the typical instability problems.
  - Positions of vertices can directly be manipulated during the simulation.
  - Easy to understand and implement.
Related Work

- Large Steps in Cloth Simulation
  - [Baraff D. and Witkin A. / Proceedings of ACM SIGGRAPH 1998]

- Using constraint function

- Instead of computing constraint function energy, this paper solve for the equilibrium configuration and project positions
Related Work

- Meshless Deformations Based on Shape Matching
  - [Matthias Müller et al. / Proceedings of ACM SIGGRAPH 2005]

- Closest method with this paper

- Only treat one specialized global constraint. Therefore, do not need a position solver.
Position Based Simulation
Algorithm Overview

1. forall vertices $i$
2. initialize $x_i = x_i^0$, $v_i = v_i^0$, $w_i = 1/m_i$
3. endfor
4. loop
5. forall vertices $i$ do $v_i \leftarrow v_i + \Delta t w_i f_{ext}(x_i)$
6. dampVelocities($v_1$, ..., $v_N$)
7. forall vertices $i$ do $p_i \leftarrow x_i + \Delta t v_i$
8. forall vertices $i$ do generateCollisionConstraints($x_i \rightarrow p_i$)
9. loop solverIterations times
10. projectConstraints($C_1$, ..., $C_{M+M_{coll}}$, $p_1$, ..., $p_N$)
11. endloop
12. forall vertices $i$
13. $v_i \leftarrow (p_i - x_i)/\Delta t$
14. $x_i \leftarrow p_i$
15. endfor
16. velocityUpdate($v_1$, ..., $v_N$)
17. endloop

- Initialize
- Move Position (external forces)
- Solver (internal forces)
Algorithm Overview - Solver

(9) loop solverIterations times
(10) projectConstraints($C_1, \ldots, C_{M+M_{coll}}, p_1, \ldots, p_N$)
(11) endloop

projectConstraints($C_1, \ldots, C_{M+M_{coll}}, p_1, \ldots, p_N$)
for ($C = C_1$ to $C_M$)
    q $\leftarrow$ C.get_using_vertices_set();
    ConstraintProjection(C,q);
endfor
for ($C = C_{M+1}$ to $C_{M+M_{coll}}$)
    q $\leftarrow$ C.get_using_vertices_set();
    CollisionResponse(C,q);
endfor
end
Define Constraint

- Constraint is a function
  - \( C : \mathbb{R}^{3n_j} \rightarrow \mathbb{R} \)
  - \( n_j \): the number of vertices that constraint use

- Constraints expression internal energy

- \( C(p) = 0 \) mean **System is stable**
Constraint Example

- $C(p_1, p_2) = ||p_1 - p_2|| - d$

\[d + \Delta p_2 = m_2 - p_2\]

\[C(p_1, p_2) = ||\Delta p_1|| + ||\Delta p_2||\]

\[d = m_2 - m_1\]

\[C(p_1, p_2) = 0\]
Constraint Projection

- Let define terms like
  - $C$: Current constraint
  - $n$: The number of vertices that constraint use
  - $p_i$: i’s vertex position vector (size is 3)
  - $w_i$: Inverse of i’s vertex mass ($= 1/m_i$)
  - $p$: $(p_1, p_2, ..., p_n)$ Connect of vertices position vector (size is 3n)
  - $\Delta p$: next timestep vertices position (size is 3n)

- We want to find $\Delta p$ s.t. $C(p + \Delta p) = 0$
- and $\sum_i m_i \Delta p_i = 0$ (momentum conserved)
Constraint Project Equation (1/2)

- Taylor expansion
  \[ C(p + \Delta p) \approx C(p) + \nabla_p C(p) \cdot \Delta p = 0 \]
- Direction of \( \Delta p \)
  \[ \Delta p = \lambda \nabla_p C(p) \] (only if all vertices mass are equal)
- Putting together
  \[ \Delta p = -\frac{C(p)}{\|\nabla_p C(p)\|^2} \nabla_p C(p) \]
- For individual position
  \[ \Delta p_i = -s \nabla_{p_i} C(p), \]
  \[ s = \frac{C(p)}{\sum_j \|\nabla_{p_j} C(p)\|^2} \] (is scaling factor)
Constraint Project Equation (2/2)

• Consider mass
  \[ \Delta p_i = -s \, w_i \, \nabla p_i C(p), \]
  \[ s = \frac{C(p)}{\sum_j w_j \| \nabla p_j C(p) \| 2} \]

• Why?
  \[ \sum_i m_i \Delta p_i = 0 \text{ (linear momentum conserved)} \]
  \[ = -s \sum_i \nabla p_i C(p) = 0 \text{ (translation invariance)} \]
Constraint Project Example

- \( C(p_1, p_2) = \|p_1 - p_2\| - d \)

\[ \nabla_{p_1} C(p) = \frac{p_1 - p_2}{\|p_1 - p_2\|} \]
\[ \nabla_{p_2} C(p) = -\frac{p_1 - p_2}{\|p_1 - p_2\|} \]

- \[ \therefore s = \frac{C(p)}{\sum_j w_j \|\nabla_{p_j} C(p)\|^2} = \frac{\|p_1 - p_2\| - d}{w_1 + w_2} \]

\[ \therefore \Delta p_1 = -\frac{\|p_1 - p_2\| - d}{w_1 + w_2} \]
\[ \Delta p_2 = +\frac{\|p_1 - p_2\| - d}{w_1 + w_2} \]
Stiffness parameter of Constraint

- \( p_i \leftarrow p_i + k \Delta p_i \) (\( k \in [0...1] \))

(9) \text{loop} \text{ solverIterations times}
(10) \text{projectConstraints}(C_1, \ldots, C_M + M_{coll}, p_1, \ldots, p_N)
(11) \text{endloop}

- Distance between prev point and result point is \( \Delta p_i(k) \)
- After \( n_s \) iteration, distance become \( \Delta p_i(1 – (1 – k)^{n_s}) \)
- So we need to change \( k' \) dependent on iteration time

- \( k' = 1 – (1 – k)^{1/solverIterations} \)
- \( p_i \leftarrow p_i + k' \Delta p_i \)
Collision Constraint

• If $x_i \rightarrow p_i$ ray enters an object,
  • $q_c$ : entry point
  • $n_c$ : surface normal

• Make Constraint
  $C(p_i) = (p_i - q_c) \cdot n_c$
  Stiffness $k=1$

• This constraint goal is $C(p_i + \Delta p_i) \geq 0$, not $C(p_i + \Delta p_i) = 0$
Collision Response

- Collision constraint will work only $C(p_i) < 0$

$$
\Delta p_i = \begin{cases} 
0 & C(p_i) \geq 0 \\
-s \, w_i \nabla p_i C(p_i) & C(p_i) < 0 
\end{cases}
$$
Cloth Simulation
 Representation of Cloth

- There are two types of internal force
  - Stretch
    - $C_{\text{stretch}}(p_1,p_2) = \|p_1 - p_2\| - l_0$
    - Stiffness $k = k_{\text{stretch}}$
  - Bending
    - $C_{\text{bend}}(p_1,p_2,p_3,p_4) = \cos\left(\frac{(p_2 - p_1) \times (p_3 - p_1)}{\| (p_2 - p_1) \times (p_3 - p_1) \|} \cdot \frac{(p_2 - p_1) \times (p_4 - p_1)}{\| (p_2 - p_1) \times (p_4 - p_1) \|}\right) - \varphi_0$
    - Stiffness $k = k_{\text{bend}}$
Bending Constraint

With Bending

Without Bending
Self Collision

• If vertex q move through a triangle $p_1$, $p_2$, $p_3$

\[ C(q, p_1, p_2, p_3) = (q - p_1) \cdot n - h \quad \text{and} \]
\[ C(q, p_1, p_2, p_3) = -(q - p_1) \cdot n - h \]

• When
  - $n$ : normal vector of triangle $p_1$, $p_2$, $p_3$
  - $h$ : cloth thickness
Cloth Balloons

\[ C(p_1,\ldots,p_N) = \left( \sum_i (p_{t_1}^i \times p_{t_2}^i) \cdot p_{t_3}^i \right) - k_{\text{pressure}} V_0 \]
\[ \triangleq \text{(Volume of mesh)} \]

- If \( k_{\text{pressure}} > 1 \) mesh will get overpressure

\[ \nabla p_i C(p) = \sum_{j:t_1=i} (p_{t_2}^j \times p_{t_3}^j) + \sum_{j:t_2=i} (p_{t_3}^j \times p_{t_1}^j) + \sum_{j:t_3=i} (p_{t_1}^j \times p_{t_2}^j) \]
\[ = \sum \text{(normal vector who contain } p_i) \times \text{(area of flat)} \]
Result

Position Based Dynamics

paper #11

All sequences captured in real time