Sofia University "St. Kliment Ohridski"
Faculty of Mathematics and Informatics
Department of Numerical Methods and Algorithms

Cloth Simulation

Master's thesis
in
Mathematical Modeling and Computational Mathematics

Author:
Dimitar V. Trendafilov
FN: 25612

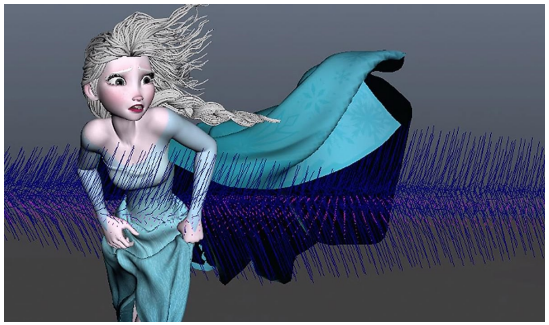
Supervisor:
Assist. Prof. Dr.
Tihomir B. Ivanov

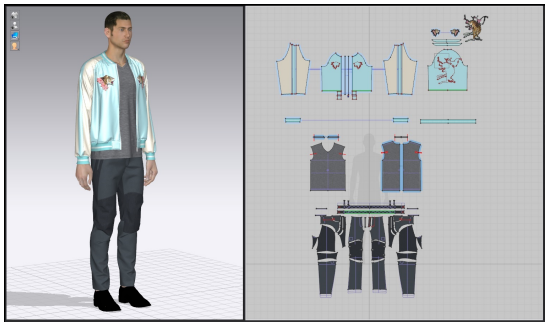
March 31, 2021

Introduction

Cloths are everywhere!

- In featured films.
- In video game.
- The textile and fashion industries.



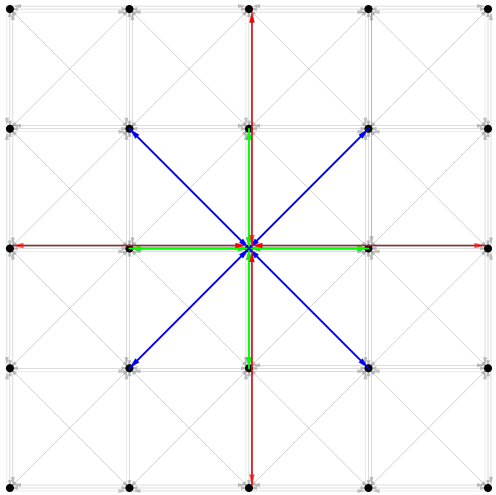


Particle Mass-Spring Model

- Considers the fabric as a mesh of $m \times n$ masses (particles).
- Each connected to 12 others through massless springs.
- The evolution of the system is governed by Newton's second law of motion $\vec{F}_{i,j} = \mu \vec{a}_{i,j}$

Particle Mass-Spring Model

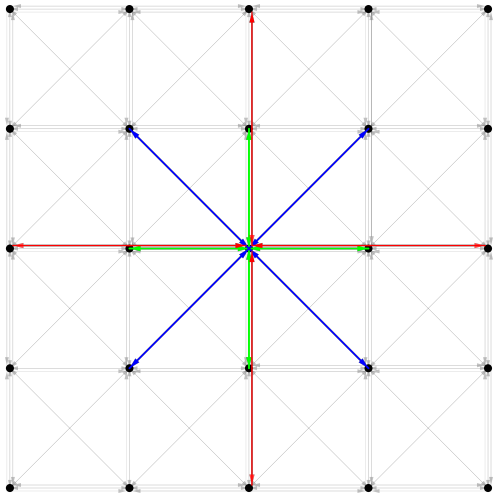
- Considers the fabric as a mesh of $m \times n$ masses (particles).
- Each connected to 12 others through massless springs.
- The evolution of the system is governed by Newton's second law of motion $\vec{F}_{i,j} = \mu \vec{a}_{i,j}$



- Structural springs.
- Shear springs.
- Flexion springs.

Particle Mass-Spring Model

- Considers the fabric as a mesh of $m \times n$ masses (particles).
- Each connected to 12 others through massless springs.
- The evolution of the system is governed by Newton's second law of motion $\vec{F}_{i,j} = \mu \vec{a}_{i,j}$



- Structural springs.
- Shear springs.
- Flexion springs.

Internal Forces

$$\vec{F}_{internal}(P_{i,j}) = - \sum_{(k,l) \in R_{i,j}} F_{Hooke}(i,j,k,l)$$

Particle Mass-Spring Model

External Forces

$$\vec{F}_{external}(P_{i,j}) = \vec{F}_{gravity}(P_{i,j}) + \vec{F}_{damp}(P_{i,j}) + \vec{F}_{wind}(P_{i,j})$$

Particle Mass-Spring Model

External Forces

$$\vec{F}_{external}(P_{i,j}) = \vec{F}_{gravity}(P_{i,j}) + \vec{F}_{damp}(P_{i,j}) + \vec{F}_{wind}(P_{i,j})$$

- $\vec{F}_{gravity}(P_{i,j}) = \mu\vec{\mathbf{g}}$

Particle Mass-Spring Model

External Forces

$$\vec{F}_{external}(P_{i,j}) = \vec{F}_{gravity}(P_{i,j}) + \vec{F}_{damp}(P_{i,j}) + \vec{F}_{wind}(P_{i,j})$$

- $\vec{F}_{gravity}(P_{i,j}) = \mu\vec{\mathbf{g}}$
- $\vec{F}_{damp}(P_{i,j}) = -C_{damp}\vec{v}_{i,j}$

Particle Mass-Spring Model

External Forces

$$\vec{F}_{external}(P_{i,j}) = \vec{F}_{gravity}(P_{i,j}) + \vec{F}_{damp}(P_{i,j}) + \vec{F}_{wind}(P_{i,j})$$

- $\vec{F}_{gravity}(P_{i,j}) = \mu\vec{\mathbf{g}}$
- $\vec{F}_{damp}(P_{i,j}) = -C_{damp}\vec{v}_{i,j}$
- $\vec{F}_{wind}$ is discussed later.

Particle Mass-Spring Model

External Forces

$$\vec{F}_{external}(P_{i,j}) = \vec{F}_{gravity}(P_{i,j}) + \vec{F}_{damp}(P_{i,j}) + \vec{F}_{wind}(P_{i,j})$$

- $\vec{F}_{gravity}(P_{i,j}) = \mu\vec{\mathbf{g}}$
- $\vec{F}_{damp}(P_{i,j}) = -C_{damp}\vec{v}_{i,j}$
- $\vec{F}_{wind}$ is discussed later.

Initial Value Problem

$$\begin{cases} \vec{F}_{i,j} & = \mu\vec{a}_{i,j} \\ \frac{d\vec{v}_{i,j}}{dt} & = \vec{a}_{i,j} \\ \frac{d\vec{P}_{i,j}}{dt} & = \vec{v}_{i,j} \end{cases}, \text{ where } i = \overline{1,m} \text{ and } j = \overline{1,n}$$

Simple Explicit Method

Numerical Scheme

$$\begin{cases} \vec{a}_{i,j}^{n} & = \frac{1}{\mu}\vec{F}_{i,j}^{n} \\ \vec{v}_{i,j}^{n+1} & = \vec{v}_{i,j}^{n} + \frac{\Delta t}{\mu}\vec{F}_{i,j}^{n} \\ \vec{P}_{i,j}^{n+1} & = \vec{P}_{i,j}^{n} + \Delta t\vec{v}_{i,j}^{n} + \frac{\Delta t^2}{\mu}\vec{F}_{i,j}^{n} \end{cases}$$

Simple Explicit Method

Numerical Scheme

$$\begin{cases} \vec{a}_{i,j}^{n} &= \frac{1}{\mu}\vec{F}_{i,j}^{n} \\ \vec{v}_{i,j}^{n+1} &= \vec{v}_{i,j}^{n} + \frac{\Delta t}{\mu}\vec{F}_{i,j}^{n} \\ \vec{P}_{i,j}^{n+1} &= \vec{P}_{i,j}^{n} + \Delta t\vec{v}_{i,j}^{n} + \frac{\Delta t^2}{\mu}\vec{F}_{i,j}^{n} \end{cases}$$

Pros:

- Extremely easy to implement.
- Low storage complexity.

Simple Explicit Method

Numerical Scheme

$$\begin{cases} \vec{a}_{i,j}^{n} & = \frac{1}{\mu}\vec{F}_{i,j}^{n} \\ \vec{v}_{i,j}^{n+1} & = \vec{v}_{i,j}^{n} + \frac{\Delta t}{\mu}\vec{F}_{i,j}^{n} \\ \vec{P}_{i,j}^{n+1} & = \vec{P}_{i,j}^{n} + \Delta t\vec{v}_{i,j}^{n} + \frac{\Delta t^2}{\mu}\vec{F}_{i,j}^{n} \end{cases}$$

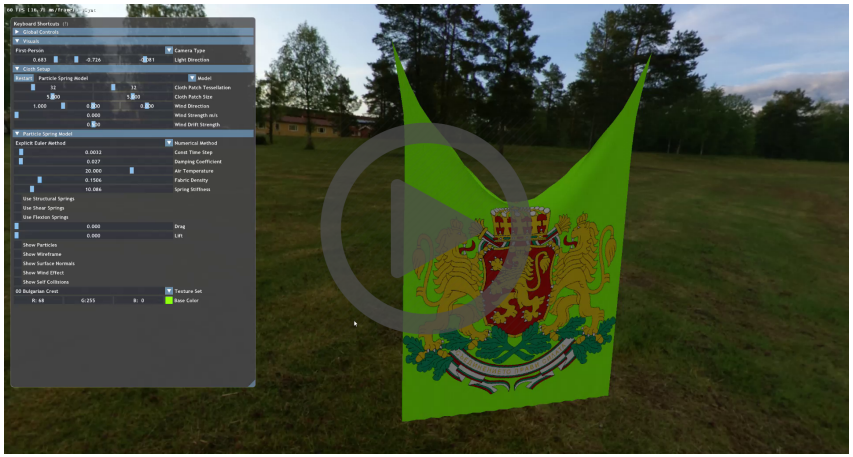
Pros:

- Extremely easy to implement.
- Low storage complexity.

Cons:

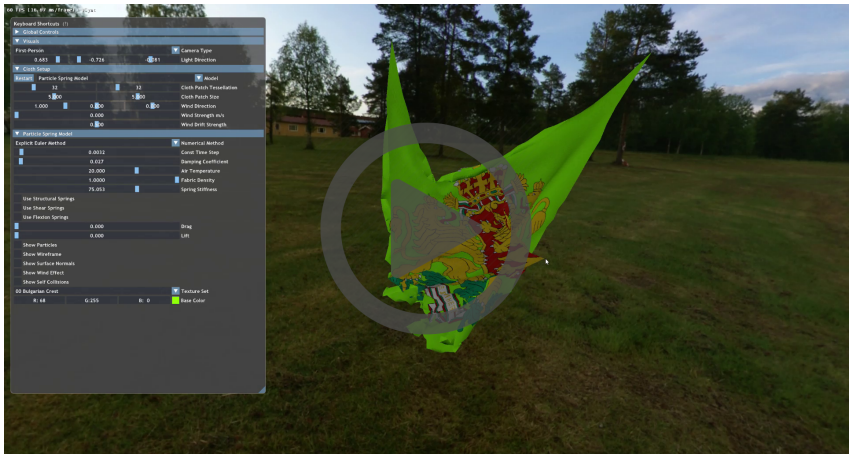
- Has numerical stability issues.
- Due to the previous point it needs small time steps which in turn leads to performance decline.

Simple Explicit Method



Video experiment showing material properties behavior:

- Changes in sagging with different spring stiffness coefficients.
- Changes in sagging with different material density.



Video experiment showing issues with large damping coefficients:

- Large fabric density can cause instabilities.
- Larger damping coefficient values mitigate the issue.
- Can give the feeling that the atmosphere is too viscose.

Basic Störmer–Verlet Method

Numerical Scheme

$$\begin{cases} \vec{a}^{n}_{i,j} & = \frac{1}{\mu}\vec{F}^{n}_{i,j} \\ \vec{P}^{n+1}_{i,j} & = 2\vec{P}^{n}_{i,j} - \vec{P}^{n-1}_{i,j} + \Delta t^2 \vec{a}^{n}_{i,j} \end{cases}$$

Basic Störmer–Verlet Method

Numerical Scheme

$$\begin{cases} \vec{a}_{i,j}^{n} & = \frac{1}{\mu}\vec{F}_{i,j}^{n} \\ \vec{P}_{i,j}^{n+1} & = 2\vec{P}_{i,j}^{n} - \vec{P}_{i,j}^{n-1} + \Delta t^2 \vec{a}_{i,j}^{n} \end{cases}$$

Pros:

- As simple as the simple explicit method.
- Low computation and storage complexity.
- Extremely stable.

Basic Störmer–Verlet Method

Numerical Scheme

$$\begin{cases} \vec{a}_{i,j}^{n} & = \frac{1}{\mu}\vec{F}_{i,j}^{n} \\ \vec{P}_{i,j}^{n+1} & = 2\vec{P}_{i,j}^{n} - \vec{P}_{i,j}^{n-1} + \Delta t^2 \vec{a}_{i,j}^{n} \end{cases}$$

Pros:

- As simple as the simple explicit method.
- Low computation and storage complexity.
- Extremely stable.

Cons:

- Global error of $\mathcal{O}(\Delta t)$.

Basic Störmer–Verlet Method

Video experiment demonstrating basic Störmer–Verlet method stability:

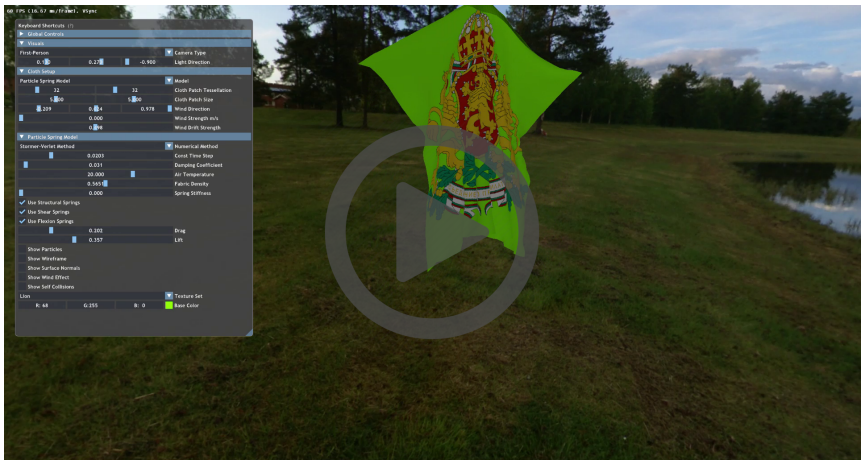
0:00 - 0:09 : Time step Δt is fixed at $0.01s$.

0:13 - 0:21 : Time step Δt is fixed at $0.02s$.

0:25 - 0:33 : Time step Δt is fixed at $0.03s$.

0:34 - 0:41 : Time step Δt is fixed at $0.04s$.

0:49 - 0:53 : Time step Δt is fixed at $0.005s$.



Variable Time Discretization Step Verlet Methods

Numerical Scheme

$$\begin{cases} \vec{a}_{i,j}^{n} & = \frac{1}{\mu}\vec{F}_{i,j}^{n} \\ \vec{P}_{i,j}^{n+1} & = \vec{P}_{i,j}^{n} + (\vec{P}_{i,j}^{n} - \vec{P}_{i,j}^{n-1})\frac{\Delta t_n}{\Delta t_{n-1}} + \vec{a}_{i,j}^{n}\frac{\Delta t_n + \Delta t_{n-1}}{2}\Delta t_n \end{cases}$$

Variable Time Discretization Step Verlet Methods

Numerical Scheme

$$\begin{cases} \vec{a}_{i,j}^{n} & = \frac{1}{\mu}\vec{F}_{i,j}^{n} \\ \vec{P}_{i,j}^{n+1} & = \vec{P}_{i,j}^{n} + (\vec{P}_{i,j}^{n} - \vec{P}_{i,j}^{n-1})\frac{\Delta t_n}{\Delta t_{n-1}} + \vec{a}_{i,j}^{n}\frac{\Delta t_n + \Delta t_{n-1}}{2}\Delta t_n \end{cases}$$

Pros:

- In terms of simplicity and precision they are similar to the Basic Störmer–Verlet Method.
- Add small computation overhead, but no storage overhead.

Variable Time Discretization Step Verlet Methods

Numerical Scheme

$$\begin{cases} \vec{a}_{i,j}^{n} & = \frac{1}{\mu}\vec{F}_{i,j}^{n} \\ \vec{P}_{i,j}^{n+1} & = \vec{P}_{i,j}^{n} + (\vec{P}_{i,j}^{n} - \vec{P}_{i,j}^{n-1})\frac{\Delta t_n}{\Delta t_{n-1}} + \vec{a}_{i,j}^{n}\frac{\Delta t_n + \Delta t_{n-1}}{2}\Delta t_n \end{cases}$$

Pros:

- In terms of simplicity and precision they are similar to the Basic Störmer–Verlet Method.
- Add small computation overhead, but no storage overhead.

Cons:

- Have precision issues with small time steps.

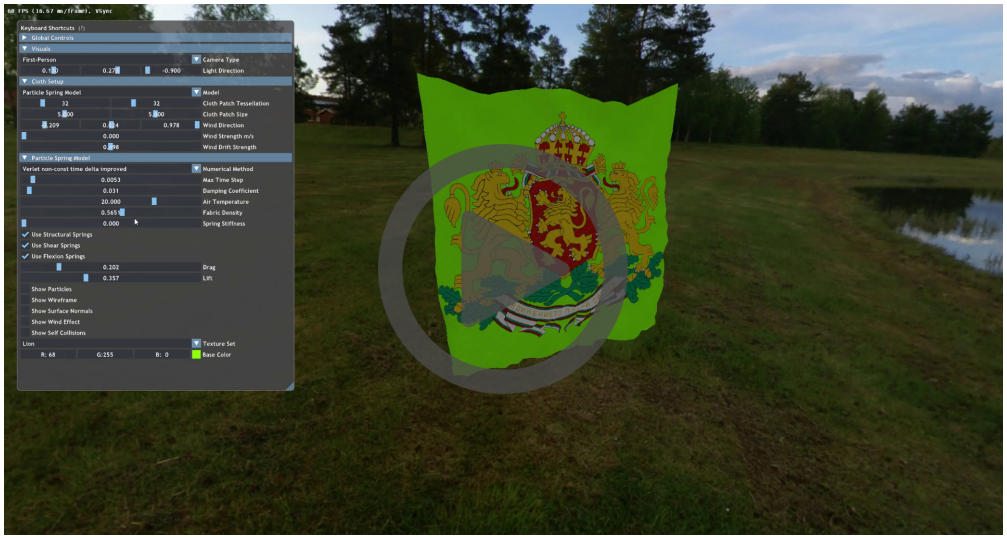
Variable Time Discretization Step Verlet Methods

Video experiment showing precision issues:

0:03 - 0:15 : At maximum $\Delta t = 0.0053s$ numerical errors cause vibrations.

0:16 - 0:21 : At maximum $\Delta t = 0.0113s$ vibrations stop.

0:22 - 0:24 : At maximum $\Delta t = 0.0165s$ numerical errors lead to catastrophic instabilities.



Wind Model

Equation

$$\vec{F}_{wind} = \frac{1}{2}\rho A \left[(C_D - C_L)(\vec{v}^r \cdot \vec{n})\vec{v}^r + C_L \left|\vec{v}\right|^2 \vec{n} \right]$$

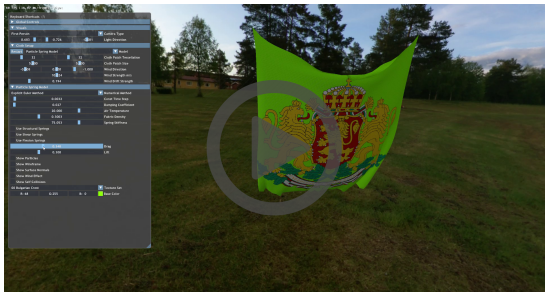
- ρ is air density.
- A is the cloth surface area.
- $\vec{n}$ is the geometry normal.
- $\vec{v}$ and $\vec{v}^r$ are the velocity and relative velocity.
- C_D and C_L are drag and lift coefficients

Wind Model

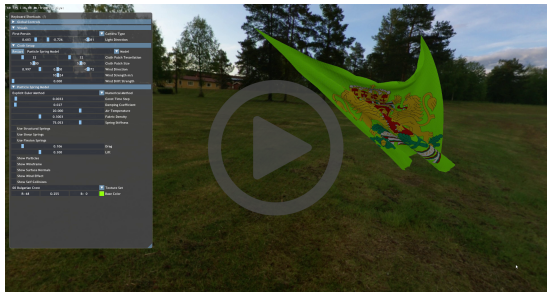
Equation

$$\vec{F}_{wind} = \frac{1}{2}\rho A \left[(C_D - C_L)(\vec{v}^r \cdot \vec{n})\vec{v}^r + C_L \left|\vec{v}\right|^2 \vec{n} \right]$$

- ρ is air density.
- A is the cloth surface area.
- $\vec{n}$ is the geometry normal.
- $\vec{v}$ and $\vec{v}^r$ are the velocity and relative velocity.
- C_D and C_L are drag and lift coefficients



Drag and lift



Turbulence

Continuum Mechanics Based Models

- Treating a cloth patch as a planar homogeneous orthotropic material.
- A finite element method for linear elastostatics in 2D is derived.
- Nonlinear effects due to large rotational deformations are handled using corotational formulation.

ODE System of the Model

$$M\ddot{x} + D\dot{x} + K(x - x_0) = f_{ext}$$

Continuum Mechanics Based Models

Mass Matrix

- The mass distribution is defined by a diagonal matrix M.
- Masses of the model are concentrated at the vertices of the mesh.
- Mass is the area "corresponding" to the respective vertex in the triangle mesh, times the material density.
- The area corresponding to a vertex is defined by the Voronoi area.

Continuum Mechanics Based Models

Mass Matrix

- The mass distribution is defined by a diagonal matrix M.
- Masses of the model are concentrated at the vertices of the mesh.
- Mass is the area "corresponding" to the respective vertex in the triangle mesh, times the material density.
- The area corresponding to a vertex is defined by the Voronoi area.

Damping Matrix

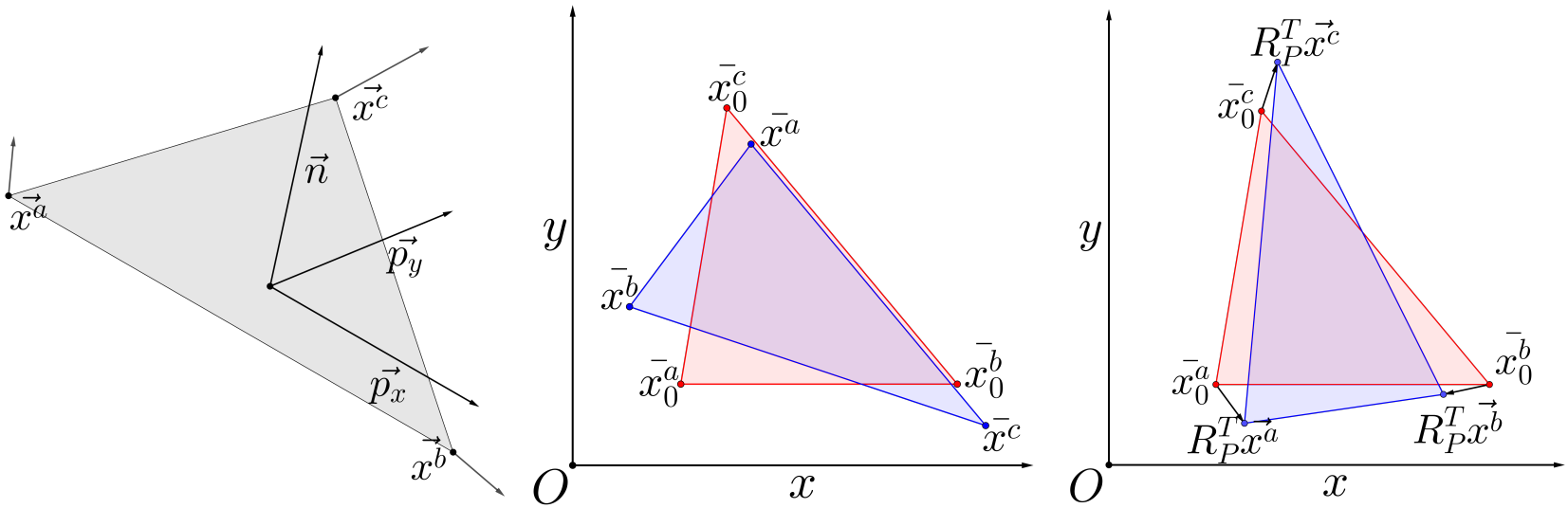
Rayleigh damping is used to simulate viscosity.

$$D = \alpha M + \beta K$$

where α and β are the damping coefficients for mass and stiffness damping, respectively.

Stiffness Matrix

$K = \left(\int\int_{\Omega} B^T C B d\Omega\right)$ is the stiffness matrix for the 2D problem. Using corrotational formulation it is transformed to $K^{CR} = R_P K R_P^T$ and $\hat{K}^{CR} = R_P K$. Using those the linear elastostatics problem becomes $F^{CR} = K^{CR} x - \hat{K}^{CR} \bar{x}_0$.



Continuum Mechanics Based Models

The ODE system of the model is numerically solved using an implicit method:

$$\left(M + \Delta t D + \Delta t^2 K^{CR}\right) \Delta v =$$

$$= -\Delta t \left(K^{CR} x_n - F_0^{CR} - f_{ext,n} + \Delta t K^{CR} v_n + D v_n\right)$$

$$\Delta x = \Delta t (v_n + \Delta v)$$

Results

Simulating wool with:

- $\Delta t = 0.003s$
- $\alpha = 0.2$, $\beta = 0.01$

Simulating viscose with:

- $\Delta t = 0.003s$
- $\alpha = 0.2$, $\beta = 0.01$

Results

Simulating wool with:

- $\Delta t = 0.099s$
- $\alpha = 0.2$, $\beta = 0.01$

Simulating wool with:

- $\Delta t = 0.003s$
- $\alpha = 0.2$, $\beta = 0.05$

Future Work

- Collision detection
  - Arbitrary geometry sweep tests
  - Bounding volume hierarchy
  - Friction forces
- Wind simulation
  - Use vector field generated by user-defined “wind sources”
  - Cloth affecting air currents
- Other models
  - Nonlinear hinge models
  - Geometry based models
  - Artificial intelligence based
  - Position based dynamics
- Computational performance optimizations
  - Parallelization
  - Conjugate gradient methods
  - Mesh subdivision

Questions

Thank you
for your attention!