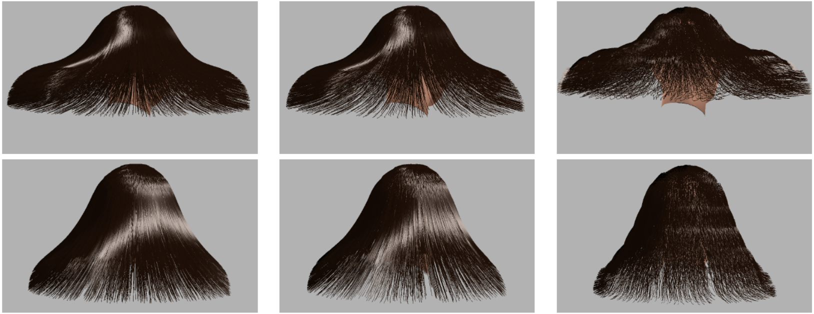
Hair Simulation Model for Real-Time Environments (review)

Petr Kadleček, Martin Kahoun

MFF UK

14.11.2011

Hair Simulation Model for Real-Time Environments
Kmoch, P., Bonanni, U. and Magnenat-Thalmann, N.



Proceedings of the 2009 Computer Graphics International Conference

Table of contents

Introduction

- Head is a natural focal point
- Realistic hair animation is a crucial part of presenting virtual humans
- Hair properties: bends, twists, unstretchable, unshearable, anisotropic, ... (Kirchhoff's hypotheses)
- Typical head = 100,000 hair strands

Goals and motivation

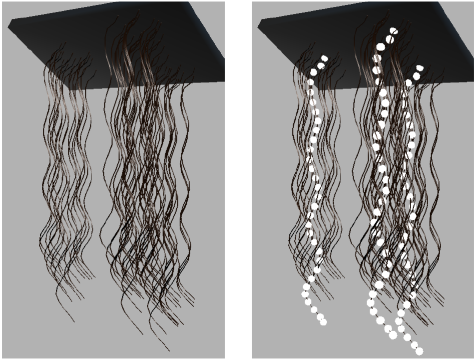
- Dynamic hair animation method designed for use in real-time virtual environments
- Physically plausible technique which utilizes specific properties of hair strands
- Enhanced stability due to decoupling major sources of dynamic equation stiffness into a separate post-integration step
- Smooth results even in frequency-sensitive areas such as haptics-based hair modelling

Methods for animating hair

- Volume based
  - Volume of "hair matter", individual strands retained
  - Free-form lattice, strands attached as viscoelastic springs
  - Smoothed particles loosely connected by springs (no notion of strands)
- Strand based
  - Mass-spring systems
  - Rigid multi-body chains
  - Cosserat theory of elastic rods (helix as a simulation primitive)

Hair simulation algorithm outline

- Per-strand basis (elastic rods)
- Leader strands and follower strands



Algorithm outline

Algorithm 1 Hair simulation outline

```
precompute rest-state values;
while simulation running do
    compute forces;
    integrate equations of motion;
    detect hair-head collisions;
    while constraints or collisions unsolved do
        perform one constraint enforcement step;
    end while
    if positions changed then
        update velocities;
    end if
    update Bishop frame;
    compute twist;
end while
```

Kirchhoff's rods

rod = deformable body whose one dimension (length) is significantly larger than the other two (cross section)

$$\Gamma(s) = \{\mathbf{x}(s), \mathbf{m}_1(s), \mathbf{m}_2(s)\} \tag{1}$$

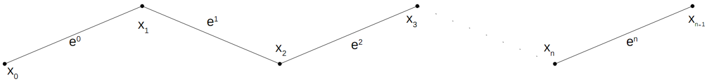
- $\mathbf{x}(s)$ is centreline position
- $\mathbf{m}_{1,2}(s)$ are axes of the cross section s runs from 0 to the rod's length L

Twist and bend: material frame vs Bishop frame

- frame (linear algebra) = a certain type of ordered set of vectors that spans a space
- Let $\mathbf{t}(s)$ be a unit vector tangent to the centreline, $t(s) \parallel x'(s)$
- $\mathbf{t}(s)$ form a $\{\mathbf{t}, \mathbf{m}_1, \mathbf{m}_2\}$ *material frame* (orthonormal frame)

- Express the material frame as a rotation of a twist-free reference frame $\rightarrow$ *Bishop frame* $\{\mathbf{t}(s), \mathbf{u}(s), \mathbf{v}(s)\}$
- Twist representation using scalar function Θ measuring the angle (around the tangent) between the material frame and the Bishop frame
- Rod's elastic energy expressed using 4 dimensions ($\mathbf{x}(s)$ and Θ)

Polyline hair approximation



- Rod $\Gamma(s)$ as $n+2$ nodes $x_0, x_1, \ldots, x_{n+1}$ and $n+1$ segments $e^0, e^1, \ldots, e^n$
- Material frame assigned to each segment

Integration step

- Twist treated quasistatically, only elastic force, gravity and friction applied
- Elastic force tries to minimize elastic energy across nodes
- Concept of holonomy used to express energy derivatives between frames
- Equations integrated using sympletic Euler method

Constraint enforcement

- Using fast manifold projection by Goldenthal et. al. (2007)
- Post-integration step, removes numerical stiffness

Constraint types

$\mathrm{CI^j} = \mathbf{e}^j \cdot \mathbf{e}^j - \hat{\mathbf{e}}^j \cdot \hat{\mathbf{e}}^j$	inextensibility ($j = 1, 2, \ldots n$)
$\mathrm{CR_0} = \hat{\mathbf{x}}_0 - \mathbf{x}_0$	rigid body coupling
$\mathrm{CH_i} = (\mathbf{x}_i - \mathbf{h}) \cdot (\mathbf{x}_i - \mathbf{h}) - \hat{r}$	hair head collisions ($i \in P$)

Fast manifold projection

- Fast manifold projection method based on Constrained Lagrangean Mechanics
- Finding a "nearby" constrained configuration for an uncostrained one
- "Nearby" based on the manifold's natural metric
- In our case in terms of kinetic energy: $\frac{1}{2}\mathbf{v}^T \mathrm{M} \mathbf{v}$

Energy functional

$$L(\mathbf{x}, \mathbf{v}) = \frac{1}{2}\mathbf{v}^T \mathrm{M}\mathbf{v} - \mathrm{C}(\mathbf{x})^T \cdot \lambda$$

$$\mathrm{M}\dot{\mathbf{v}} = -\nabla \mathrm{C}(\mathbf{x})^T \cdot \lambda, \ \mathrm{C}(\mathbf{x}) = 0$$

Fast manifold projection (cont.)

- For elaborate derivation refer to *Efficient Simulation of Inextensible Cloth* by Goldenthal et. al. (2007)
- Iterative Newtonian minimalization

Discrete linear system

$$\delta \mathbf{x}_{i+1} = -h^2 \mathrm{M}^{-1} \nabla \mathrm{C}(\mathbf{x}_i)^T \delta \lambda_{i+1} \tag{2}$$

$$\nabla \mathrm{C}(\mathbf{x}_i) \delta \mathbf{x}_{i+1} = -\mathrm{C}(\mathbf{x}_i) \tag{3}$$

$$h^2 (\nabla \mathrm{C}(\mathbf{x}_i) \mathrm{M}^{-1} \nabla \mathrm{C}(\mathbf{x}_i)^T) \delta \lambda_{i+1} = \mathrm{C}\mathbf{x}_i \tag{4}$$

Results

- Model linear with number of nodes (strands)
- Twisting computed 2x faster than using Newtonian methods
- Collisions introduce negligible overhead
- Medium sized scenes reach real-time performance (1kHz)
- Further improvements possible via GPU parallel

Q&A