

Augmented Mass-Spring Model for Real-Time Dense Hair Simulation

Supplementary Material

A. Biphasic Interaction

Despite various optimizations, DER-based methods remain computationally intensive when simulating strand dynamics, making them challenging to apply in complex interactive scenarios involving intricate hairstyles. Consequently, we focused on addressing the two primary limitations of the more efficient MS method: stability and the loss of global shape during initialization.

Based on the detailed study of Selle *et al.* [19], stability issues in MS arise from collapsed tetrahedra formed by springs between consecutive strand particles. Because of this, our approach introduces an angular interaction with the ghost rest-shape configuration, which prevents tetrahedron collapse by maintaining an augmented stable structure based on the connections between ghost and real particles, as shown in Figure 1.

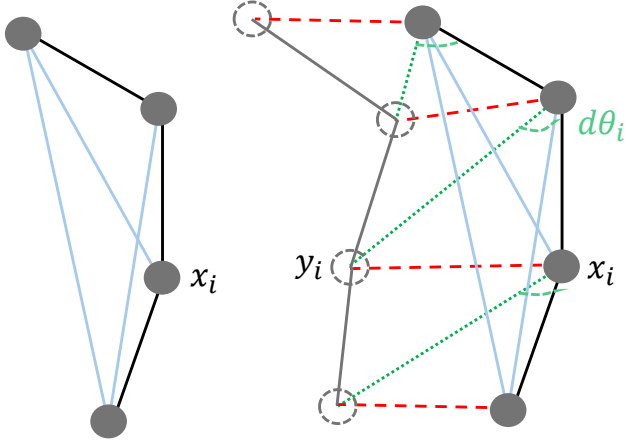


Figure 1. Schematic representation of the tetrahedra formed between consecutive real particles (left), and the additional real-ghost interaction in our formulation (right). The *angular* one-way force enhances stability by preventing tetrahedron collapse when particles deviate from their original dihedral angles.

Despite the enhanced numerical stability, the use of very stiff springs remains necessary to preserve global features, which, in practice, reintroduces instabilities unless extremely small time step sizes are employed. This constraint limits the feasibility of real-time applications. Moreover, while edge, bending, torsion, and *angular* interactions maintain local shape fidelity, they fail to encode the global hair structure. To address these challenges, we encode the global features of the hair through the *integrity* interaction with the rest shape, which establishes a relationship between each particle and its corresponding ghost based on

the total displacement of the strand. This mechanism introduces an additional force that mitigates sagging and preserves the global shape, independently of the particle count in the discretization, by counteracting the weight of consecutive particles.

It is important to note that the two couplings we introduce for the biphasic interaction function as force perturbations to prevent tetrahedral collapse and encode global features. However, there is a potential risk that these additional forces may interfere with the fidelity of the dynamics. To mitigate this, we typically set the biphasic coupling constants several orders of magnitude lower than those of the traditional local springs, ensuring that the necessary perturbations are introduced to enhance the MS model while preserving dynamic accuracy.

B. Numerical Integration

Our integration procedure, summarized in Algorithm 1, updates the particle dynamics on each iteration.

ALGORITHM 1: Time integration procedure of our framework.

Input: Current hair strands and mesh.

Output: Updated particles/mesh.

1 Procedure:

2 — Compute $\Delta t' = \Delta t / M$.

3 — Define $\mathbf{X}_0^{n+1} = \mathbf{X}^n$.

4 — Define $\mathbf{V}_0^{n+1} = \mathbf{V}^n$.

5 **for** $i = 1; i \leq M$ **do**

6 — Compute the intra-particle and biphasic interaction terms, as described in Section 3.1.3.

8 — Solve the implicit Euler step for velocity update

9 given by $\mathbf{V}_i^{n+1} = \mathcal{E} \cdot (\mathbf{X}_{i-1}^{n+1}, \mathbf{V}_{i-1}^{n+1}, \mathbf{F}^n, \Delta t')$

10 — Update Position $\mathbf{X}_i^{n+1} = \mathbf{X}_{i-1}^{n+1} + \Delta t' \mathbf{V}_i^{n+1}$.

11 **end**

12 — Apply inextensibility constraints to modify $\mathbf{V}_M^{n+1}$ and $\mathbf{X}_M^{n+1}$.

13 — Rasterize velocities into dynamic background volume.

14 — Solve equivalent system through FLIP/PIC routine.

15 — Transfer velocity back to particles and resolve detailed collisions.

16 — Resolve hair-solid collisions as described in Section 3.3.

First, we embed the head mesh S (or other solid meshes in the scene) within two 3D volumes $\Omega_{\text{Int}, \text{SDF}} \in \mathbb{R}^3$ which we use for hair-hair interactions and SDF computation, respectively. Depending on the specific use of altitude springs and ghost configuration, the mass-spring model of Selle *et*

al. [19] forms a banded matrix with seven to nine non-zero entries per particle, which represent the local connectivity of the system. Since we do not use two-way coupled ghosts or altitude springs, the resulting numerical system in our framework is strictly heptadiagonal, which means the LU decomposition can be solved exactly using only two iterations, in a similar fashion as the solvers used in [9] and [24]. In general, the implicit system for a strand will have the form $\mathbf{A}\mathbf{V} = \mathbf{b}$, where the biphasic interaction is incorporated into $\mathbf{b}$, and, considering the edge, bending, and torsional degrees of freedom, the only non-zero elements in row i are those at $j = i - 3, \dots, i + 3$. In turn, we can write the system as

$$\begin{aligned} \mathbf{A}_{i,j} &= \begin{cases} -\Delta t^2 M_i^{-1} \kappa_{i,j} \mathbf{D}_{i,j}, & |i-j| \leq 3, \\ \mathbf{0}, & \text{otherwise.} \end{cases} \\ \mathbf{A}_{i,i} &= (1 + \Delta t M_i^{-1} \mathbf{G}_i) \mathbf{I} + \sum_{k \in \mathcal{N}(i)} \Delta t^2 M_i \kappa_{i,k} \mathbf{D}_{i,k}. \\ \mathbf{b}_i &= \mathbf{V}_i^n + \Delta t M_i^{-1} (\mathbf{F}^n + \hat{\mathbf{S}}^n). \end{aligned}$$

This represents a linear equation in $\mathbb{R}^3$ and can be solved using a single forward and backward pair of sweeps. The first sweep corresponds to the decomposition $\mathbf{A} = \mathbf{L}\mathbf{U}$, where the strict band size of $\mathbf{A}$ implies that $\mathbf{L}_{i,j} = \mathbf{U}_{i,j} = \mathbf{0}$ for $i - j < 3$ and $j - i > 3$, respectively. For the other entries in the decomposition, we first do the forward sweep to compute

$$\begin{aligned} \mathbf{L}_{i,j} &= \mathbf{A}_{i,j} - \sum_{k=\max(1,i-3)}^{j-1} \mathbf{L}_{i,k} \mathbf{U}_{k,j}, \\ \mathbf{V}'_i &= (\mathbf{L}_{i,i})^{-1} \left[\mathbf{b}_i - \sum_{j=\max(0,i-3)}^{i-1} \mathbf{L}_{i,j} \mathbf{V}'_j \right], \end{aligned}$$

with the intermediate vector $\mathbf{V}' = \mathbf{L}^{-1}\mathbf{b}$. Next, the backward sweep yields

$$\begin{aligned} \mathbf{U}_{i,j} &= (\mathbf{L}_{i,i})^{-1} \left[\mathbf{A}_{i,j} - \sum_{k=\max(1,j-3)}^{i-1} \mathbf{L}_{i,k} \mathbf{U}_{k,j} \right], \\ \mathbf{V}_i &= \mathbf{V}'_i - \sum_{j=i+1}^{\min(i+3,N)} \mathbf{U}_{i,j} \mathbf{V}_j, \end{aligned}$$

where the final vector $\mathbf{V}$ is given by the relation $\mathbf{U}\mathbf{V} = \mathbf{V}'$.

Non-Hookean Effects In order to simulate the progressive loss of hair shape features under extreme forces, we introduce non-linear tension responses in AMS by parametrizing an elongation curve for the *integrity* tension T_I which accounts for non-Hookean behavior, as demonstrated in Figure 2.

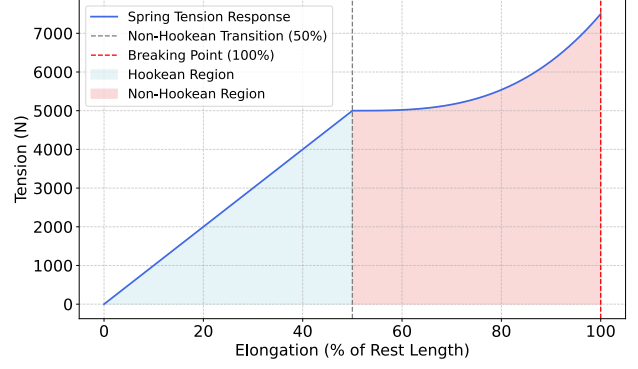


Figure 2. Parametrization plot for incorporating non-Hookean responses in the T_I term of the biphasic coupling.

C. Integrity Preprocessing

Although our system effectively preserves global hair features, it still exhibits minor sagging effects during initialization. To mitigate this, we apply a technique similar to the *gravity pre-loading* method proposed by [8], adapted to the specific interactions relevant to our model. First we note that, at initialization, all the strand springs as well as biphasic terms are at equilibrium, so the only force acting on each particle is due to its own weight $\mathbf{w} = m\mathbf{g}$. Because of this, the initial sag stops until all the spring forces reach a new equilibrium with the total hair weight. Moreover, all of the internal Dofs and the angular interaction are given by the input configuration and then evolve dynamically. However, we can pre-process the *integrity* coupling of the interaction T_I in terms of the ghost configuration to achieve an equilibrium.

Specifically, we consider each particle i with position $\mathbf{x}_i$ and its corresponding ghost at position $\mathbf{y}_i = \mathbf{x}_i + \Delta\mathbf{r}_i$, where $\Delta\mathbf{r}_i$ is the vector joining both particles. Originally, $\Delta\mathbf{r}_i = \mathbf{0}$ at initialization. However, we pre-process this value to account for sagging by setting

$$\mathbf{T}_I - \mathbf{w}_i = \mathbf{0}. \quad (1)$$

Developing this equation we get

$$\kappa_I d(\mathbf{x}_i, \mathbf{y}_i) \hat{\mathbf{r}}_i - \mathbf{w}_i = \kappa_I \|\Delta\mathbf{r}_i\| \hat{\mathbf{r}}_i - \mathbf{w}_i = \mathbf{0}. \quad (2)$$

Solving this equation element-wise we finally get

$$\Delta\mathbf{r}_i = \frac{m}{\kappa_I} \mathbf{g}. \quad (3)$$

Translating the initial position of ghost particles to $\mathbf{y}_i = \mathbf{x}_i + \Delta\mathbf{r}_i$ enables us to eliminate sagging at initialization.

D. Procedural Growth

We use an heuristic approach for hair growth that is divided into two stages. First, given a pre-selected set of triangles in

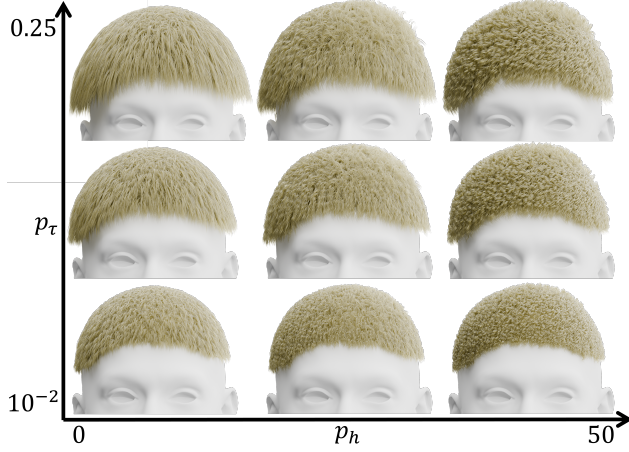


Figure 3. Parameter space exploration showing the impact of increasing values of helix radius p_h and step size p_τ in our procedural hair growth module. We can control the *curliness* and length of generated hair with these two parameters.

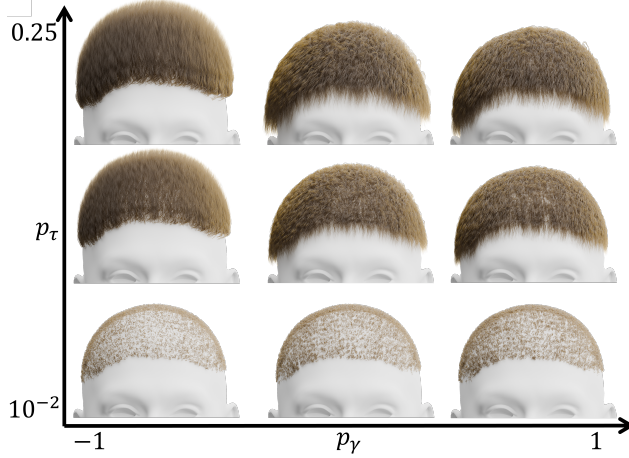


Figure 4. Parameter space exploration showing the impact of increasing values of the gravity influence parameter p_γ and step size p_τ in our procedural hair growth module. We can control the hair deviation in the y direction using different values for p_γ .

the mesh, we sample p_n random root positions per triangle. Then, we compute the initial strand direction $\mathbf{p}_{\text{dir}}^0$ on each position by weighting the per-vertex normal vectors of the root using its barycentric coordinates, and adding a noise vector with entries from the distribution $\mathcal{U}(-1, 1)$. During a second stage, we add sequential vertices to the strand, starting from the root. Specifically, we compute

$$\mathbf{p}_{\text{dir}}^{i'} = \mathbf{p}_{\text{dir}}^{i-1} + \mathbf{p}_{\text{grav}}^{i-1} \max(p_\Gamma, 1 - \|\mathbf{p}_{\text{dir}}^{i-1} \cdot (0, 1, 0)\|), \quad (4)$$

where p_Γ fixes the maximum particle deviation, and the procedural vector $\mathbf{p}_{\text{grav}}^i$ accounts for strand changes in the vertical direction, and is defined as

$$\mathbf{p}_{\text{grav}}^i = (0, -ip_\gamma, 0), \quad (5)$$

with gravity influence parameter p_γ . Then, to incorporate curls into our procedural growth module, we perform an additional update step

$$\mathbf{p}_{\text{dir}}^i = \mathbf{p}_{\text{dir}}^{i'} + p_\Omega (\mathbf{p}_{\text{dir}}^{i'} - \mathbf{H}^i), \quad (6)$$

with spiral impact factor p_Ω , and helix vector $\mathbf{H}^i$ described by

$$\mathbf{H}^i = ((p_h \cos(ip_{\text{freq}}), 1, p_h \sin(ip_{\text{freq}})), \quad (7)$$

with helix radius p_h . We demonstrate the generation capabilities of our procedural growth scheme by performing two parameter space explorations, as shown in Figures 3 and 4. In both cases, we set $p_\Gamma = 0.2$, $p_{\text{freq}} = 1$, and $p_\Omega = 0.017$.