

Animating Hair with Loosely Connected Particles

Yosuke Bando[†]

Bing-Yu Chen[‡]

Tomoyuki Nishita

The University of Tokyo
{ybando, robin, nis}@nis-lab.is.s.u-tokyo.ac.jp

Abstract

This paper presents a practical approach to the animation of hair at an interactive frame rate. In our approach, we model the hair as a set of particles that serve as sampling points for the volume of the hair, which covers the whole region where hair is present. The dynamics of the hair, including hair-hair interactions, is simulated using the interacting particles. The novelty of this approach is that, as opposed to the traditional way of modeling hair, we release the particles from tight structures that are usually used to represent hair strands or clusters. Therefore, by making the connections between the particles loose while maintaining their overall stiffness, the hair can be dynamically split and merged during lateral motion without losing its lengthwise coherence.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism, I.3.3 [Computer Graphics]: Picture/Image Generation

1. Introduction

The ability to represent realistic-looking hair plays a crucial role in the synthesis of life-like human models. However, the enormous number (a human scalp typically has 100,000 strands) and thin nature of hair strands complicate and slow down all of the processes for hair image generation, including modeling, rendering and animation. Moreover, when we want to animate hair based on a physically plausible simulation, the situation is even worse, because we have to take into account interactions among the hair strands (so-called hair-hair interactions) in order to reproduce realistic behavior of the hair, as many researchers have already pointed out^{4, 10, 13, 16, 20, 22}. The complex behavior of hair results from the characteristics of individual hair strands and the interactions between them such as collisions, friction, repulsion due to static electricity and cohesion/adhesion due to lipids or hair-dressings²¹. Hair-hair interactions are therefore essential when animating hair. However, computing the interactions among a large number of individual hair strands is still expensive with the computer power currently available.

In this paper, a practical method is proposed for animating hair while taking into account hair-hair interactions at

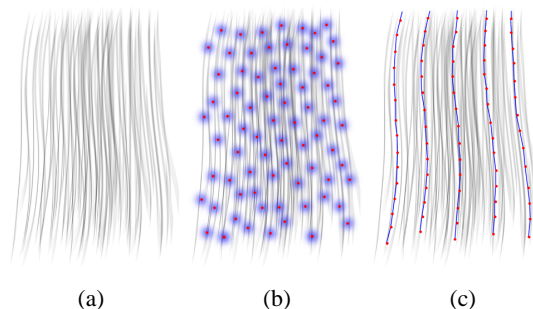


Figure 1: (a) The hair that we wish to represent. (b) The particles are distributed as sampling points for the volume of the hair. (c) Traditional methods often represent the hair as serial chains of connected particles (the same number of particles as (b)).

a reasonable cost. Traditionally, hair is modeled using fixed structures that represent hair strands or clusters, such as serial chains of connected points or particles. In contrast, we use unordered particles that have only loose connections to those nearby. Thus we call them *loosely connected particles*, or *LCP*. The particles serve as sampling points that track the motion of the volume of the hair, and the dynamics of the hair, including hair-hair interactions, is simulated us-

[†] In Toshiba Corporation since April 2003.

[‡] In National Taiwan University since August 2003.

ing the interacting particles. Each particle represents a certain amount of the hair medium around it, and it might be viewed as a volume density. This notion, inspired by Hadap and Magnenat-Thalmann¹⁰, is illustrated in Figure 1.

Representing hair by using sampling points can be regarded as an approximation or simplification of a collection of individual hair strands in order to reduce the high computational cost of checking and evaluating hair-hair interactions. From this point of view, one might think of a more typical approach, i.e., to bring several strands together to form a cluster or wisp and to then model each cluster as some structured element, such as a generalized cylinder. This is justifiable because the hair strands tend to clump together, which is one of the frequently observed phenomena resulting from hair-hair interactions, and clustering effects are therefore important for producing natural-looking hair images. However, as Kim and Neumann¹⁴ pointed out, clustering is not a static phenomenon. When hair moves, the hair-hair interactions cause clusters to split and merge dynamically. To simulate this, we could subdivide the structured elements to construct a hierarchical tree structure of clusters, as in their *multi-resolution hairstyle modeling (MHM)* system. However, it is unclear how we can split and merge the clusters at different levels in different sub-trees. Although we can animate the highest resolution clusters individually, this severely limits the number of clusters available in order to handle hair-hair interactions with a reasonable computation time.

For this reason, we do not arrange the particles to form a fixed set of clusters, but distribute them in an unordered manner throughout the hair volume. The hair is rendered by placing billboards with hair texture at the positions of the particles, oriented by a hair direction vector maintained with each particle. To simulate dynamic clustering effects, the neighboring particles should have coherence and move together to some extent. Moreover, the neighboring particles should also be able to draw apart, since they do not represent portions of the hair volume that are occupied by exactly the same set of hair strands. Therefore, some looseness in the connections between the particles is essential. Results show that our LCP approach can successfully reduce the computational cost of animating hair without losing most of the characteristics resulting from hair-hair interactions.

2. Related Work

In this section, we limit our review to previous work on the animation of hair that takes into account hair-hair interactions. Interested readers may refer to the survey by Magnenat-Thalmann et al.¹⁷ and Kim's dissertation¹².

Several works have modeled specific aspects of hair-hair interactions. Kim and Neumann¹³ added some constraints to hair strands by enclosing a hair surface within a thin bounding volume. Lee and Ko¹⁶ gave hair body by prohibiting hair strands from penetrating inside layers defined around the head.

Hadap and Magnenat-Thalmann¹⁰ animated hair with generalized hair-hair interactions. They assumed that hair is a continuum and elegantly handled hair-hair interactions using particle-based fluid dynamics. However, the computational cost was still high because they modeled individual hair strands explicitly with serial chains of rigid segments (10,000 strands with 30 segments each) and glued several particles to each of the segments, so that the amount of computation was huge (2 minutes per frame). The results were excellent, but not to the extent that they captured the discontinuities of hair. Our LCP approach can be viewed as freeing the particles from these serial chains.

Koh and Huang¹⁵ modeled hair clusters as strips of spline surfaces and avoided collisions by introducing springs among them. Plante et al.²⁰ simulated interactions among the hair clusters, each of which was modeled as a skeleton and its envelope. The skeleton captures the lengthwise shape of the cluster, whereas its envelope captures the cross-sectional shape deformation. Chang et al.⁴ simulated only a small number of hair strands called guide hairs, and interpolated the rest. They handled hair-hair interactions with auxiliary triangle strips spanning the nearby strands. These three methods are similar in spirit. They reduce the number of hair strands to be computed (in the form of strips, skeletons, and guide hairs, respectively) in order to speed up the simulation and also to capture the discontinuities of hair. Moreover, the gaps among the sparse hair strands are compensated for by additional structures such as springs, envelopes and auxiliary triangle strips, respectively. We also take a similar approach, but since the particles are distributed over the hair volume as shown in Figure 1 (b), the gaps between the particles are small. Besides, compensation for these gaps, which actually models hair-hair interactions, is performed through particle dynamics rather than by introducing additional structures. The method by Koh and Huang¹⁵ achieved real-time animation, but the springs accounting for hair-hair interactions were simplistic and imposed strong restrictions on free lateral motion of the hair. The methods proposed by Plante et al.²⁰ and Chang et al.⁴ took up to tens of seconds to obtain the final rendered image.

3. Modeling Hair with LCP

3.1. Overview

Hair is modeled as a set of particles that serve as sampling points for the volume of the hair. Particle i has mass m_i , position \mathbf{x}_i and velocity \mathbf{v}_i , just like a standard particle system. The mass indicates the amount of hair medium that the particle represents and the distribution of the particle mass in space determines the density of the hair. The density here indicates the amount of hair medium in a unit volume, as distinguished from the density of the hair material. As described by Hadap and Magnenat-Thalmann¹⁰, we borrow an idea from *smoothed particle hydrodynamics (SPH)*^{18, 6}, and

the density ρ_i at position \mathbf{x}_i is computed as follows:

$$\rho_i = \sum_j m_j W(\|\mathbf{x}_j - \mathbf{x}_i\|, h), \quad (1)$$

where j runs through all of the particle indices and W represents an interpolating kernel called a *smoothing kernel*. We assume that the particles are smeared out in space so that each particle represents the volume density around it. h is called the *smoothing length* and indicates the extent of the influence. We used the following smoothing kernel¹⁸:

$$W(r, h) = \frac{1}{4\pi h^3} \begin{cases} 4 - 6\left(\frac{r}{h}\right)^2 + 3\left(\frac{r}{h}\right)^3 & 0 \leq r \leq h \\ \left(2 - \frac{r}{h}\right)^3 & h \leq r \leq 2h \\ 0 & 2h \leq r \end{cases}. \quad (2)$$

Thus only nearby particles within a distance of $2h$ contribute to the summation of Equation (1).

In addition, particle i has a unit vector \mathbf{t}_i that represents the direction of hair, and it also has constant coordinates (u_i, v_i, s_i) indicating the initial state of the hair, where (u_i, v_i) are parameters on the scalp and s_i is the arc length of hair from the scalp. Section 3.2 describes how to determine the initial hair directions and assign these coordinates.

The dynamics of hair comes from a combination of the characteristics of hair strands and hair-hair interactions. As described in Section 3.3, the dynamics of hair strands can be accounted for by forces acting between a fixed set of particle pairs that are initially near neighbors, because two particles in each pair share a portion of some hair strands and should have lengthwise coherence. We determine this set of pairs using smoothing length h_1 in the initialization process described in Section 3.2. On the other hand, forces due to hair-hair interactions act among current neighboring particles with smoothing length h_2 , as described in Section 3.4. Thus, at each time step we have to search for pairs of particles less than $2h_2$ apart. This can be performed efficiently by a grid of voxels of size $2h_2$. Assuming that the distribution of particles is fairly even, the time complexity bears a linear relationship to the number of particles⁶. We also apply other forces such as gravity and air friction as described in Section 3.5, and update the particle positions and velocities by integrating the equations of motion. In summary, after initializing particles, we repeat the following sequence of steps per time step.

1. Perform neighbor search.
2. Compute the particle density using Equation (1).
3. Activate the forces described in Sections 3.3, 3.4 and 3.5.
4. Update the particle positions and velocities.
5. Update the particle directions using Equation (15).

3.2. Particle Initialization

This subsection describes the method for distributing particles over a polygonal head model prior to animation.

First, as shown in Figure 2 (a), we trim the scalp surface

from the head model with four Catmull-Rom spline curves³ by specifying their control points interactively. Then we embed the scalp surface in a unit square domain $D: [0, 1] \times [0, 1]$ with axes u and v as shown in Figure 2 (b), using a piecewise linear approximation of *harmonic mapping*⁸. If we denote this mapping from world space to the parameter domain by $\phi: \mathbb{R}^3 \rightarrow D$, the scalp surface in the world space is fully parameterized by coordinates (u, v) as $\phi^{-1}(u, v)$. We add a third dimension s over D , which represents the arc length of hair, and then distribute the particles within the volume of this square pillar, as shown in Figure 2 (c). Its lower bound is the plane $s = -s_r$, where s_r is the depth of the hair root beneath the scalp. The particles with negative arc length are referred to as *root particles*, and are fixed to the head during animation. The upper bound of the parameter volume is given as the hair length map $L(u, v)$, which is given as a user-defined grayscale image scaled by the maximum length of hair. As we want the particles to evenly sample the hair volume, we distribute the particles based on the Poisson disc distribution. In order to compensate for the distortion of mapping ϕ , we use the distance between the corresponding points on the scalp in the world space for the (u, v) coordinates. That is, we define the distance between two points $P_1(u_1, v_1, s_1)$ and $P_2(u_2, v_2, s_2)$ in the parameter domain as:

$$d(P_1, P_2) = \sqrt{\|\phi^{-1}(u_1, v_1) - \phi^{-1}(u_2, v_2)\|^2 + (s_1 - s_2)^2}. \quad (3)$$

Finally, particle i with parameters (u_i, v_i, s_i) other than the root particles is mapped back to the world space using the cantilever beam simulation method proposed by Anjyo et al.¹. Starting from the root position $\phi^{-1}(u_i, v_i)$ in the hair root direction $\mathbf{t}_r(u_i, v_i)$ specified over the scalp, we integrate the beam model of length s_i to account for the bending of the hair due to gravity, and we determine the particle position \mathbf{x}_i and the direction \mathbf{t}_i by those of the endpoint of the beam. On the other hand, the root particles are placed inside the scalp as $\mathbf{x}_i = \phi^{-1}(u_i, v_i) + s_i \mathbf{t}_r(u_i, v_i)$. Figure 2 (d) shows the resultant initial state of the particles.

Specifying the root direction $\mathbf{t}_r(u, v)$ manually is a rather tedious task, and we can determine it simply by the following steps. The user first selects a point on the scalp as a whorl center of the hair. We denote the normal vector at this point by \mathbf{n}_w . Then the root direction is computed using the unit normal vector $\mathbf{n}(u, v)$ of the scalp by the following equation:

$$\mathbf{t}_r(u, v) = -w_t \{\mathbf{n}_w - (\mathbf{n}_w \cdot \mathbf{n}(u, v))\mathbf{n}(u, v)\} + w_n \mathbf{n}(u, v), \quad (4)$$

where w_t and w_n are user-specified global weighting constants for the tangent and normal components of the hair root direction, respectively. A large value for w_t relative to w_n makes the hair lie down. The obtained root direction is then normalized.

To simulate the dynamics of hair strands, we establish the connections between nearby particles. As shown in Figure 3 (a), the neighbor search is performed with a smoothing length h_1 for the positions of the particles initialized in the

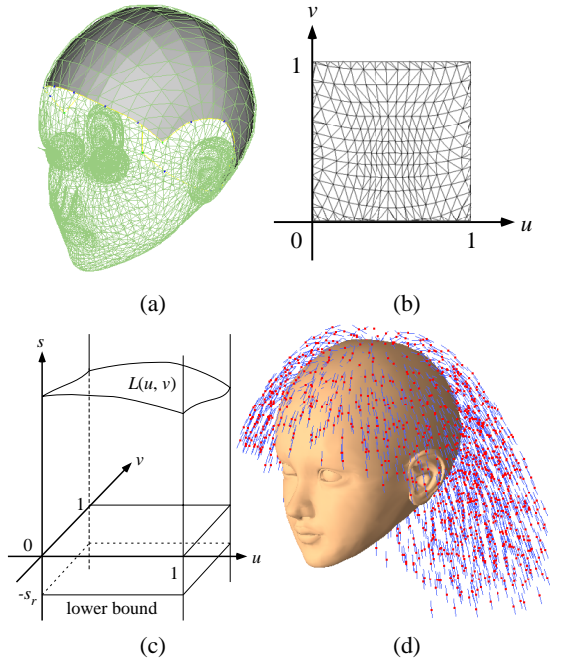


Figure 2: Initializing particles over the scalp. (a) A trimmed scalp surface. (b) The scalp surface embedded in a unit square. (c) The volume of the parameter domain (u, v, s) . (d) The particles mapped back to the world space. The line segments indicate the hair directions.

above procedure, where h_1 is a little larger than the initial average inter-particle distance. Moreover, we store the initial state of each particle pair (i, j) by computing the distance l_{ij} between the two particles, and the angle θ_{ij} between the hair direction \mathbf{t}_i and the direction of the particle pair $\mathbf{x}_{ij} = \mathbf{x}_j - \mathbf{x}_i$, as shown in Figure 3 (b).

We assign a strength of connection c_{ij} to the particle pair using the following equation, which is used for the spring coefficients described in the next subsection:

$$c_{ij} = a_{ij}l_{ij}, \quad (5)$$

where a_{ij} indicates the degree to which two particles are aligned in a row and thus how strong their lengthwise coherence is. This is defined as:

$$a_{ij} = \begin{cases} |\cos\theta_{ij} - \cos\theta_{ji}|/2 & \cos\theta_{ij} \cdot \cos\theta_{ji} < 0 \\ 0 & \text{otherwise} \end{cases}, \quad (6)$$

which has the maximum value of 1 when the three vectors \mathbf{t}_i , \mathbf{t}_j and \mathbf{x}_{ij} are oriented in the same direction. Even when \mathbf{t}_i and \mathbf{t}_j are colinear, the two particles are not considered to be well aligned if \mathbf{x}_{ij} is perpendicular to them. Because we activate the spring forces that account for the stiffness of the hair strands, connecting particles in close proximity with stiff springs makes the system unstable. Therefore, the second factor l_{ij} in Equation (5) is used for stability. For

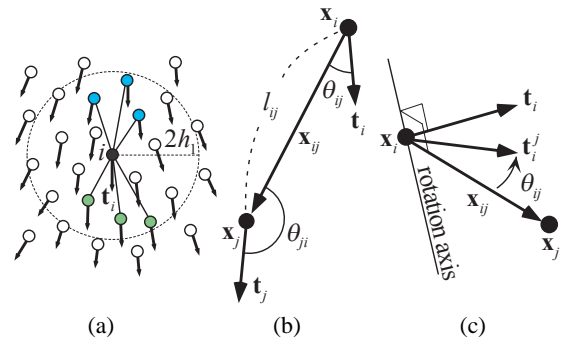


Figure 3: (a) The neighbor search is performed with a smoothing length h_1 (so the radius is set as $2h_1$), and the particles connected with the center particle i are divided into two groups: $N_r(i)$ (the blue nodes) and $N_t(i)$ (the green nodes). (b) The initial state of a particle pair. (c) A diagram for updating the hair direction.

computational efficiency, we first omit the pairs which have a_{ij} smaller than a threshold a_0 . Then, from the pairs which have smaller values for $c_{ij}W(l_{ij}, h_1)$, which is the initial spring coefficient between the two particles as will be described in Equation (7) in the next subsection, we omit them recursively until the number of connections for each particle is around the specified value N_n . Experimentally we use $a_0 = 0.5$ and $N_n = 12$. In Figure 3 (a), $N_n = 6$.

3.3. Dynamics of Hair

In this subsection, we focus on the dynamics of hair that is due to the characteristics of hair strands, and hair-hair interactions are explained in the next subsection.

The hair volume can be considered as a deformable body, which can be animated by connecting the neighboring particles with damped springs. The particle pairs are established using smoothing length h_1 in the initialization process described in the previous subsection and this set of pairs is fixed during the animation phase. Since the hair volume deforms anisotropically, i.e., it is hard to stretch but free to move laterally as Chang et al.⁴ pointed out, we model large tensile stiffness and small bending stiffness of the hair, though we do not consider the twist of hair and neglect the torsional stiffness. To realize free lateral motion of the hair, we make the connections between the particles loose by using the following spring coefficient, which is a function of the current distance between a pair of particles:

$$k_{ij} = c_{ij}W(\|\mathbf{x}_{ij}\|, h_1), \quad (7)$$

which gradually decreases as the two particles draw apart until they become free from each other at a distance of $2h_1$.

However, these loose connections also allow the hair to stretch. Therefore, in order to prevent the tensile stiffness of

hair from changing, we keep the tensile elastic modulus of each particle constant during animation. The elastic modulus (also known as Young's modulus) E is unique to a material, and we can relate it to the rest length L and a cross-sectional area A of the material by the following equation:

$$E = \frac{L}{A}K, \quad (8)$$

where K corresponds to the spring coefficient when we view the material as the lengthwise spring. In the following, we express E in terms of the parameters of the particles and the springs connected between them. The cross-sectional area A_i of the hair which particle i represents (not of individual hair strands) can be written as the following equation using the density δ of the hair material:

$$A_i = \left(\frac{m_i}{\delta}\right)^{\frac{2}{3}}. \quad (9)$$

Since we are concerned with the tensile stiffness of the hair, we divide the neighboring particles of particle i into two groups, as shown in Figure 3 (a): one includes those closer to the root of the hair, and the other includes those closer to the tip of the hair. They are denoted by $N_r(i)$ and $N_t(i)$, respectively. For each group, we sum up all of the contributions of the springs to the elastic modulus of particle i , which is kept constant by solving the following equations with unknown scaling factors α_{ij} for the spring coefficients k_{ij} :

$$E = \frac{1}{A_i} \sum_{j \in N_r(i)} l_{ij} \alpha_{ij} k_{ij} \quad (10)$$

$$E = \frac{1}{A_i} \sum_{j \in N_t(i)} l_{ij} \alpha_{ij} k_{ij}, \quad (11)$$

where l_{ij} is the rest length of the spring between particles i and j , which is their initial distance as described in Section 3.2. The number of unknowns α_{ij} is the number of particle pairs, and it is much larger than the number of equations, which is twice the number of particles. Thus, Equations (10) and (11) form an underdetermined sparse linear system. To keep the computational cost as low as possible, we directly (non-iteratively) obtain an approximate solution to this system in linear time in the number of particles, which cannot be accomplished by the least square methods for underdetermined linear systems⁵. Since the spring coefficients k_{ij} already include weighting factors through Equation (7), we make α_{ij} independent of j as:

$$\alpha_{ij} = \alpha_{r,i} \text{ for } \forall j \in N_r(i), \quad (12)$$

and we solve Equation (10) for each i individually:

$$\alpha_{r,i} = \frac{EA_i}{\sum_{j \in N_r(i)} l_{ij} k_{ij}}. \quad (13)$$

Similarly Equation (11) is solved to obtain $\alpha_{t,i}$, and we take the average of the two solutions as follows:

$$\alpha_{ij} = \begin{cases} (\alpha_{r,i} + \alpha_{t,j})/2 & i \in N_r(j) \wedge j \in N_r(i) \\ (\alpha_{r,j} + \alpha_{t,i})/2 & i \in N_r(j) \wedge j \in N_t(i) \end{cases}. \quad (14)$$

In summary, at each time step, we first compute the raw spring coefficients k_{ij} using Equation (7), then we compute the scaling factors α_{ij} using Equations (13) and (14), and finally we activate spring forces with spring coefficients $\alpha_{ij}k_{ij}$. As the particles draw apart and several connections between them become weak or have no effect, the remaining connections become strong to maintain the tensile stiffness. Even though the particles have no direct connections to the scalp, the hair neither stretches nor comes apart.

Based on the current spring coefficient $\alpha_{ij}k_{ij}$, we update the hair direction \mathbf{t}_i of particle i . As described in Section 3.2, particle i has the initial angle θ_{ij} between the hair direction \mathbf{t}_i and the direction of the connection \mathbf{x}_{ij} . Thus, we can compute the updated hair direction \mathbf{t}_i^j by rotating the normalized vector of \mathbf{x}_{ij} by angle θ_{ij} around the axis normal to both \mathbf{t}_i and \mathbf{x}_{ij} , as shown in Figure 3 (c). Taking the weighted sum, the final updated hair direction is:

$$\mathbf{t}_i^* = \sum_j \alpha_{ij} k_{ij} \mathbf{t}_i^j, \quad (15)$$

where \mathbf{t}_i^* is subsequently normalized.

3.4. Hair-hair Interactions

We model attraction/repulsion, collision, and friction to account for hair-hair interactions. The forces due to these interactions act upon current nearby particles, and can be modeled by using a smoothing length h_2 that is a little smaller than the initial average inter-particle distance.

Attraction/repulsion forces can be caused by lipids, hair-dressings or static electricity. Macroscopically, the effect of these interactions is to preserve the average density of hair. The force on particle i due to particle j is given by the following equation, which is the pressure force in SPH¹⁸:

$$\mathbf{f}_{a,ji} = -m_i m_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} \right) \frac{\partial}{\partial \mathbf{x}_i} W(\|\mathbf{x}_{ij}\|, h_2), \quad (16)$$

where the pressure P is modeled as $P = k_a(\rho - \rho_0)$ and k_a and ρ_0 control the magnitude of the force and the average density, respectively⁶. A small value for ρ_0 makes the hair repulsive, whereas a large value makes it cohesive.

To model inelastic collisions of hair, we simply reduce the relative velocity $\mathbf{v}_{ij} = \mathbf{v}_j - \mathbf{v}_i$ of two particles if they are approaching. The direction of the collision, \mathbf{d}_n , is normal to the hair directions \mathbf{t}_i and \mathbf{t}_j of both particles. Thus, $\mathbf{d}_n = \mathbf{t}_i \times \mathbf{t}_j$. If $\|\mathbf{d}_n\| \ll 1$, the two hair directions are almost colinear, and in this case we can compute the direction of the collision as $\mathbf{d}_n = \mathbf{x}_{ij} - (\mathbf{x}_{ij} \cdot \mathbf{t}_i)\mathbf{t}_i$. After normalizing \mathbf{d}_n , we check the sign of $(\mathbf{x}_{ij} \cdot \mathbf{d}_n)(\mathbf{v}_{ij} \cdot \mathbf{d}_n)$. If this is negative, the two particles are approaching, and we apply the following force:

$$\mathbf{f}_{c,ji} = d_c W(\|\mathbf{x}_{ij}\|, h_2) (\mathbf{v}_{ij} \cdot \mathbf{d}_n) \mathbf{d}_n, \quad (17)$$

where d_c is the collision damping constant.

Friction is modeled similarly. The direction \mathbf{d}_f of the frictional force is defined as $\mathbf{v}_{ij} - (\mathbf{v}_{ij} \cdot \mathbf{d}_n)\mathbf{d}_n$, which is normal to the direction of the collision \mathbf{d}_n . After normalizing it, the force is computed as:

$$\mathbf{f}_{f,ji} = d_f W(\|\mathbf{x}_{ij}\|, h_2) (\mathbf{v}_{ij} \cdot \mathbf{d}_f)\mathbf{d}_f, \quad (18)$$

where d_f is the frictional damping constant.

3.5. Other Forces

We model gravity, air friction, wind, inertia due to head movement, and collisions against the body (the head and the torso). Gravity is a downward force whose magnitude is proportional to the particle mass, and air friction is modeled as a damping force whose magnitude is proportional to the particle velocity.

The wind is also represented by a set of particles, and is simulated using SPH¹⁸. The interactions between the hair particles and the air particles are modeled as drag forces between them¹⁰, and thus the wind is also affected by movement of the hair. Though these interactions include the effect of air friction, the damping force described above is still necessary to prevent abrupt motion of the hair. Since our LCP approach models hair solely with particles, other particle-based methods are easily incorporated with a slight modification to the programming code.

The reason to model inertia is that it is numerically more stable to modify the particle accelerations than their positions. We describe the equations of motion of the particles with respect to the local coordinate system attached to the head, and we exert inertial forces on the particles, avoiding directly altering their positions relative to the head.

After applying all of the forces described so far, collisions between the particles and the body are detected efficiently by again utilizing the voxel grid structure. And each particle in contact with the body suffers a repulsive force that dissipates its relative velocity normal to the body, and a frictional force is also exerted according to Coulomb's model².

4. Rendering Hair

Since the particles represent the volume density of hair, we can render hair by using volume rendering techniques, just as Kajiya and Kay did for their furry teddy bear¹¹. Unlike the standard voxel grid structure, the density field in our model is represented by the distribution of particles, and in this case we can benefit greatly from the splatting method with billboards by using the hardware accelerated method presented by Dobashi et al.⁷. We assume a parallel light source, and in the first pass, we project the particles in the direction of the light source in order to ascertain how much of the light intensity is attenuated. In the second pass, we can then render the hair with complex self-shadows that enhance the volumetric appearance of the hair. Figure 4 shows the rendered images of our hair model without and with shadows.

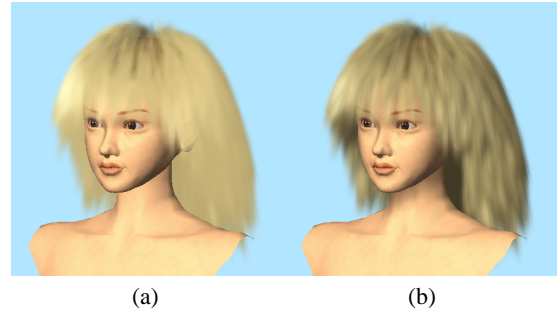


Figure 4: The hair rendered (a) without (b) with shadows.

To simulate the highly anisotropic phase function of hair we use the Goldman model⁹, which adds a directionality factor to the Kajiya-Kay model¹¹ that controls the reflection and transmission ratios of light. However, as the number of particles is relatively small for computational efficiency, the phase function alone is not sufficient to produce the appearance of hair strands. We compensate this loss by mapping hair texture to the billboards instead of the simple texture of a Gaussian distribution. Figure 5 (a) shows an example of hair texture. We match the hair direction of the texture (usually the vertical axis) to the particle direction \mathbf{t}_i as shown in Figure 5 (b). As opposed to the standard billboard, we can only place it so that its horizontal axis is parallel to the screen. Hence, we scale the opacity of the billboard by $1/\sin\theta$, where $\cos\theta = \mathbf{t}_i \cdot \mathbf{d}_p$ and \mathbf{d}_p is the direction of projection. We can further augment the directionality by scaling the size of the billboards as shown in Figure 5 (c).

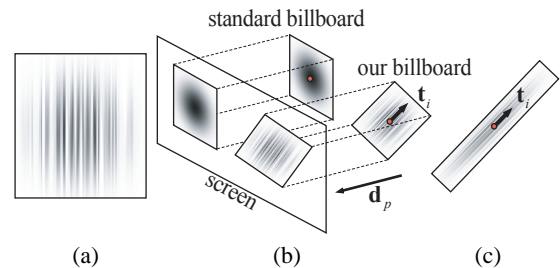


Figure 5: (a) An example of hair texture. (b) The hair direction of the texture is matched to the particle direction. This figure shows the case of orthogonal projection. The case of perspective projection is similar. (c) The size of the billboard is scaled.

5. Results

We successfully animated the hair with a head and torso model at an interactive frame rate. Figures 6 (a) – (c) show a few frames from an animation of back hair. The hair is forced to split into two, and then merges again because of gravity. Hair-hair interactions prevent the hairs from penetrating

each other as shown in Figure 6 (c), which is a contrast to Figure 6 (d) where hair-hair interactions are not considered. Figure 7 shows an animation of hair when the head is shaken. The head turns around and the hair follows this motion with delay because of inertia. Dynamic clustering effects are visible. Figure 8 shows an animation of hair blowing in wind. Interactions between the hair and the air are also simulated.

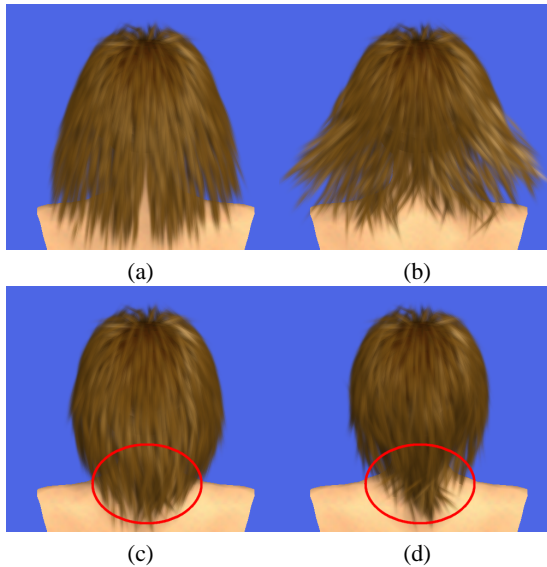


Figure 6: (a) – (c) A few frames from an animation of back hair. (d) The hairs penetrate each other when hair-hair interactions are not considered, which is a contrast to (c). The difference is clearly seen in the red circles.



Figure 7: Animation of hair when the head is shaken. Left to right, top to bottom.

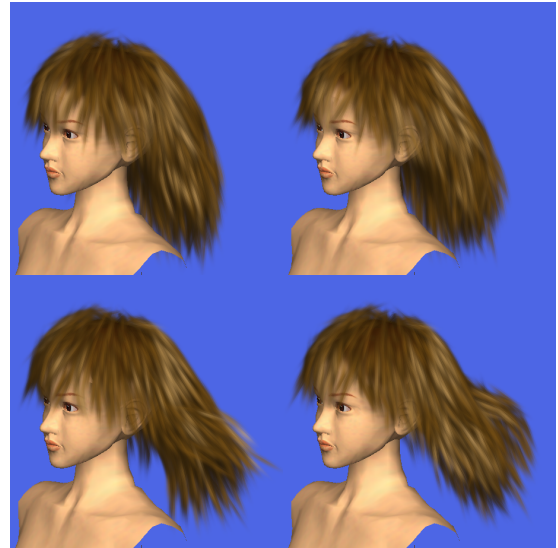


Figure 8: Animation of hair blowing in wind. Left to right, top to bottom.

We used about 2,000 particles to represent the hair for all of the animations presented above. The number of particles is determined by a compromise between accuracy and efficiency, and we chose the number so that the animation, including both simulation and rendering with shadows, can be performed at an interactive frame rate. The hair was animated at 6.7 fps on a PC with an Intel Pentium 4 2.8 GHz CPU and an NVIDIA Quadro4 900 XGL GPU. The simulation took 33% of the total computation time, and the rendering took the rest. The size of the image and the buffer for computing shadows is 512×512 .

6. Conclusions and Future Work

This paper has presented a practical method for animating hair with a new model called *loosely connected particles (LCP)*. The hair is modeled as a set of unordered particles that serve as sampling points for the volume of the hair. The loose connections between the particles make the hair free to move laterally, nevertheless providing lengthwise coherence by maintaining their tensile stiffness. The hair is rendered by placing billboards with hair texture at the positions of the particles, utilizing graphics hardware accelerations.

Our LCP approach has the following advantages:

- It reduces the high computational cost of simulating the dynamics of the hair including hair-hair interactions, and realizes interactive hair animation.
- The simulation is solely based on the particle dynamics, which simplifies the implementation. Other particle-based methods can be easily incorporated.

On the other hand, there are limitations to our method:

- It is applicable to only a simple hairstyle, and extension to various hairstyles is a major remaining task.
- The simulation is not physically rigorous. The nonlinear spring forces accounting for the stiffness of the hair are designed to obtain the desirable motion of the hair, and the bending stiffness is not modeled explicitly.
- The hair is still rendered coarsely. Interpolating between the particles to place many smaller billboards would produce finer images, but this will slow down the system below an interactive frame rate on current computers.

In addition to addressing the above limitations, the future direction of this work would be to take further advantage of the particle representation, for example, adaptive sampling of the hair volume and incorporating the *simulation level of detail* technique¹⁹.

Acknowledgments

We thank Prof. Nelson Max (University of California) and Dr. Yoshinori Dobashi (Hokkaido University) for many useful suggestions, and Ryoichi Mizuno for naming our hair model “loosely connected particles.” This project is partially supported by Toshiba Corporation.

References

1. K. Anjyo, Y. Usami, and T. Kurihara. A simple method for extracting the natural beauty of hair. *ACM Computer Graphics (Proc. of SIGGRAPH '92)*, **26**(2):111–120, 1992.
2. R. Bridson, R. Fedkiw, and J. Anderson. Robust treatment of collisions, contact and friction for cloth animation. *ACM Transactions on Graphics (Proc. of SIGGRAPH 2002)*, **21**(3):594–603, 2002.
3. E. Catmull and R. Rom. A class of local interpolating splines. *Computer Aided Geometric Design (Proc. of International Conference on Computer Aided Geometric Design '74)*, 317–326, 1974.
4. J. T. Chang, J. Jin, and Y. Yu. A practical model for hair mutual interactions. *Proceedings of ACM SIGGRAPH Symposium on Computer Animation 2002*, 73–80, 2002.
5. R. E. Cline and R. J. Plemmons. l_2 -solutions to underdetermined linear systems. *SIAM Review*, **18**(1):92–106, 1976.
6. M. Desbrun and M.-P. Gascuel. Smoothed particles: a new paradigm for animating highly deformable bodies. *Proceedings of Eurographics Workshop on Computer Animation and Simulation '96*, 61–76, 1996.
7. Y. Dobashi, K. Kaneda, H. Yamashita, T. Okita, and T. Nishita. A simple, efficient method for realistic animation of clouds. *Proceedings of ACM SIGGRAPH 2000*, 19–28, 2000.
8. M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbury, and W. Stuetzle. Multiresolution analysis of arbitrary meshes. *Proceedings of ACM SIGGRAPH '95*, 173–182, 1995.
9. D. Goldman. Fake fur rendering. *Proceedings of ACM SIGGRAPH '97*, 127–134, 1997.
10. S. Hadap and N. Magnenat-Thalmann. Modeling dynamic hair as a continuum. *Computer Graphics Forum (Eurographics 2001 Proc.)*, **20**(3):329–338, 2001.
11. J. Kajiya and T. Kay. Rendering fur with three dimensional textures. *ACM Computer Graphics (Proc. of SIGGRAPH '89)*, **23**(4):271–280, 1989.
12. T.-Y. Kim. *Modeling, Rendering and Animating Human Hair*. Ph.D dissertation, University of Southern California, 2002.
13. T.-Y. Kim and U. Neumann. A thin shell volume for modeling human hair. *Proceedings of Computer Animation 2000*, 121–128, 2000.
14. T.-Y. Kim and U. Neumann. Interactive multiresolution hair modeling and editing. *ACM Transactions on Graphics (Proc. of SIGGRAPH 2002)*, **21**(3):620–629, 2002.
15. C. K. Koh and Z. Huang. A simple physics model to animate human hair modeled in 2D strips in real time. *Proceedings of Eurographics Workshop on Computer Animation and Simulation 2001*, 127–138, 2001.
16. D.-W. Lee and H.-S. Ko. Natural hairstyle modeling and animation. *Graphical Models*, **63**(2):67–85, 2001.
17. N. Magnenat-Thalmann, S. Hadap, and P. Kalra. State of the art in hair simulation. *Proceedings of International Workshop on Human Modeling and Animation 2000*, 3–9, 2000.
18. J. J. Monaghan. Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics*, **30**:543–574, 1992.
19. D. O'Brien, S. Fisher, and M. C. Lin. Automatic simplification of particle system dynamics. *Proceedings of Computer Animation 2001*, 210–219, 2001.
20. E. Plante, M.-P. Cani, and P. Poulin. A layered wisp model for simulating interactions inside long hair. *Proceedings of Eurographics Workshop on Computer Animation and Simulation 2001*, 139–148, 2001.
21. C. R. Robbins. *Chemical and Physical Behavior of Human Hair (4th ed.)*. Springer-Verlag, 2002.
22. R. Rosenblum, W. Carlson, and E. Tripp. Simulating the structure and dynamics of human hair: modeling, rendering and animation. *The Journal of Visualization and Computer Animation*, **2**(4):141–148, 1991.