Capturing the Complexity of Hair Motion

Eric Plante

Taarna Studios, inc., and University of Montreal[1]

Marie-Paule Cani

iMAGIS/GRAVIR-IMAG[2]

Pierre Poulin

University of Montreal

E-mail: (plante | poulin) @iro.umontreal.ca, Marie-Paule.Cani@imag.fr

Modelling the highly inelastic interactions inside hair, and between the hair and the character body is a determinant feature for achieving realistic hair animation. However, the extremely large number of hair strands has so far prevented animators from handling these interactions. It is also unclear if they should be treated at the level of individual hair strands.

This paper proposes an approach to handle these interactions, based on a new clustering paradigm. The high friction forces between neighboring hair strands, yielding a high motion coherence, is simulated by clustering these strands into an anisotropic viscous volume representing a hair wisp. All other interactions between hair strands are approximated at the wisp level, thanks to an anisotropic behavior that allows pairs of wisps to interpenetrate or to collide depending on their relative orientations. The wisps also offer the necessary structure to efficiently simulate the collisions with the character body. The model results in visually-realistic animations, where complex wisp interactions capture both continuities and discontinuities easily observed in thick, long hair.

Key Words: hair, collision detection and response, layered models, physically-based animation, natural phenomena

1. INTRODUCTION

Impressive advances have been achieved in modelling and animating virtual creatures within the past few years (see for instance [30, 5, 14, 13]). These techniques cover mod-

[1] current affiliation: discreet

[2] a joint research project of CNRS/INRIA/UJF/INPG, INRIA Rhône-Alpes, ZIRST, 655 avenue de l'Europe, Montbonnot, 38334 Saint Ismier Cedex, France

elling the creature's body, clothes, motion, and behavior. Meanwhile, hair animation did not seem to attract as much attention. Due to the number of primitives to animate (about 100,000 hair strands), most previous approaches only simulated the motion of a few "guide strands", and relied on interpolation for generating the others, yielding extremely smooth animated hair. Collisions with the character body were only processed for the guide strands, causing possible penetration of the interpolated strands into the model. Moreover there was no attempt to model the complex way long hair interacts with itself. Without these interactions – namely the modelling of inelastic collisions and friction between hair strands – the use of hair models for general animation purposes seems doubtful: the kinetic energy is not sufficiently dissipated. Moreover the hair does not tend to come back to its rest density after compression. This explains why computer generated hair often seems too light and lacks volume.

The aim of this paper is to present a general methodology for modelling and animating long hair.[3]

Due to the extremely large number of hair strands, computing hair animation at a reasonable amount of time has to rely, somehow, on calculations at a coarser level. Most previous techniques assumed that this coarser level was still a set of hair strands : they interpolate the position of most hair strands from those of a few animated "guide strands". The approach we develop in this paper breaks away from these techniques. Our coarser level is composed of a set of anisotropic viscous volumes representing wisps of hair, i.e., clusters of neighboring hair strands.

The number of wisps being about two orders of magnitude smaller than the number of hair strands, all wisps can be animated. This avoids the unnatural hair smoothness inherent to the use of interpolation. During animation, wisp deformations model the relative motion of neighboring hair strands, for which friction plays a key role. Both collisions between hair and the character's body and inter-collisions between non-neighboring hair strands are computed at the wisp level. An anisotropic collision model is used for wisps inter-collisions, since two perpendicular colliding wisps should compress to avoid intersection, while interpenetration may be allowed between neighboring wisps of similar orientations and velocities. As our results show, this careful modelling of interactions achieves to capture both the continuities and the discontinuities that can be observed in thick, long hair.

Previous work in hair modelling and animation are reviewed in Section 2. Section 3 presents our model for hair wisps. Section 4 explains how wisps are used for processing interactions. Our implementation of the resulting animation algorithm is detailed in Section 5. Rendering of hair is addressed in Section 6, followed by results of our technique in Section 7. We finally conclude and discuss future directions.

2. PREVIOUS WORK

2.1. Dynamic Animation of Hair

Most techniques to animate hair are based on dynamics. One of the first attempts was presented by Rosenblum *et al.* [25]. The movement of each individual hair strand is approximated using a chain of point-masses linked by stiff springs and hinges. The enormous

[3] A preliminary version of this work appeared in [24]. It can be accessed from www.iro.umontreal.ca/labs/infographie/papers/Plante-2001-LWM.

amount of calculation required for their examples limited the number of strands to about 1000, which might explain why subsequent research has not followed this approach.

Instead, several researchers [10, 2, 27, 20] have been inspired by an approach introduced by Anjyo *et al.* [3]. The initial position of each strand is determined by a cantilever beam simulation, and its movement is simulated with a set of rigid sticks, only from root to tip.

2.2. Hair Interactions

Anjyo *et al.* [3] were the first to integrate collision with the character's body to their simulations. During the root-to-tip calculation of hair strands motion, penetrations of strand segments (sticks) into the character body are detected and avoided. However, since a given stick has no influence on the sticks closer to the root, the motion of a hair strand cannot be adequately modified during a collision. The problem can be avoided by animating only short hair that do not reach the shoulders.

Starting from a similar framework, Lee and Ko [20] try to model the emerging effect of hair-hair interactions by adding friction forces between each hair strand and a set of head hull layers. This restores a convincing hair volume as long as the head stays near the vertical position, but would not work for large head motions as the ones in Figure 10.

Another solution for modelling hair strands that occupy an adequate volume and avoid obstacles is to attach them to streamlines of a fluid flow [15]. However, this technique generates neighboring hair that always have the same orientations, and thus fails capturing the case when two wisps of different orientations collide.

A very interesting method for handling hair interactions was recently proposed by Hadap and Magnenat-Thalmann [16]. It consists in embedding animated hair strands, modelled as chains of rigid links, into a fluid that represents the hair volume. Both hair-hair interactions and hair-air interactions are handled through the additional fluid dynamic forces. However, the idea of modelling the hair volume as a continuum still seems to prevent the generation of highly discontinuous hair configurations, as in the photograph of real hair in the left image of Figure 11.

2.3. Exploiting Coherence in Hair Motion

Even with nowadays computers, animating each individual hair strand results in very low frame rates. Parallelism may be exploited to reach acceptable performances; hence Hadap and Magnenat-Thalmann achieve the animation of 10,000 strands at 2 minutes per frame [16].

Another methodology consists in taking advantage of the coherence in hair motion. The simplest way is to animate only a few strand guides, and to interpolate the motion of the strands in between [10]. However, it results in very smooth hair, which may lack volume and cannot adequately handle collisions with the character's body since collisions are only detected for the guide strands.

Another way to exploit local hair coherence is to cluster hair strands into a smaller number of hair wisps. Watanabe and Suenaga [29] were the first ones to introduce this idea. They were followed by Daldegan *et al.* [10] (wisps as cylinders), Chen *et al.* [9] (trigonal prisms), Yang *et al.* [31] (generalized cylinders), Kim and Neumann [17] (thin-shell volumes), and Koh and Huang [18] (semi-transparent polygon strips). These methods focussed on improving the aspect of hair, and easing the modelling of hair styles. None of them used physically-based animation of wisps.

Koh and Huang [19] recently extended their real-time hair model [18] to a simplified physically-based animation framework : vertical rows of points controlling the wisp polygon strips are animated using chains of rigid links. Wisp deformations are controlled by adding internal springs within a wisp. Moreover, wisp interpenetrations are avoided by adding extra springs between neighboring wisps. However, this fails modelling wisp interactions in a general situation, since the hair almost behaves as a smooth surface. In contrast with this method, our physically-based hair animation will rely on volumetric wisps and fully handle 3D interactions between them.

3. VOLUMETRIC WISPS

Similarly to previous techniques, our hair motion is derived from physically-based animation. In doing so, our goal is not one of exact physical correctness, but instead, of increased visual realism obtained within a reasonable amount of computation time. We believe that modelling the complex interactions that occur both inside the hair and between the hair and obstacles is a key step towards this goal.

As mentioned earlier, correctly simulating interactions such as collisions and friction between 100,000 individual hair strands is prohibitively expensive. Fortunately, hair motion is locally very coherent. The extreme lightness of hair strands allows friction forces to play a key role in hair dynamics. These forces yield a coherent motion for groups of neighboring hair strands with similar orientations. In order to model this local coherence at low cost, we propose to cluster hair strands into a number of deformable hair wisps.

3.1. Modelling a Hair Wisp

Intuitively, a wisp is a group of hair strands of similar orientations, originating from the same neighborhood on the scalp. Our wisp model captures the effect of friction forces between these strands. At first sight, the global motion of a wisp under gravity is similar to the motion of a single, but heavier hair strand. However, the wisp cannot be reduced to this single strand. The shape of its cross-sections may vary over time, since applied forces may locally either compress or expand the hair strands composing the wisp. Moreover, contrary to individual hair strands, the wisp axial curve (skeleton) is not always inextensible. For instance, the waviness of hair strands inside the wisp may decrease during rapid motion, thus causing a global elongation of the wisp.

We therefore model a wisp as an anisotropic volume (a form of generalized cylinder) whose skeleton may be extensible and whose cross-sections offer a very viscous and slightly elastic response to both compression and stretching. Wisps being a coarse approximation of what occurs at a finer scale, trying to set a finite element formulation for this problem would not be appropriate. We rather structure a wisp into a hierarchy of simple sub-models:

1. A *skeleton curve* that defines the large-scale motion of the wisp.
2. A deformable virtual *envelope* that defines the thickness of the wisp around the skeleton (thus delimiting the wisp volume).
3. *Hair strands* which are rendered inside the wisp volume according to its deformations.

While designing this model, we did not try to handle the dynamic twisting of hair wisps. Hair offers a relatively strong resistance to torsion, so hair wisps rarely start to twist when a person moves. Modelling wisps that never twist, whatever the way they bend, allows

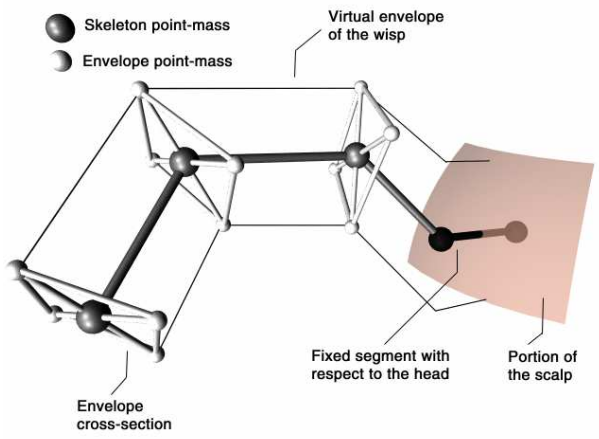

FIG. 1. The elements defining the skeleton and the envelope of a wisp, and their configuration.

us to rely on a simple method, namely constrained point dynamics, for animating the two first sub-models. In contrast, solid dynamics would have been required for animating dynamic twisting effects. A possible extension of our method incorporating twisting will be discussed in Section 8.

The remainder of the section details the first two sub-models, animated using a physically-based approach. The third sub-model is a geometric representation added for display. It will be discussed in Section 6.

3.2. Wisp Skeleton

In order to provide a good approximation of the global wisp motion, the wisp skeleton is represented as a deformable curve whose mass is the sum of the masses of the hair strands within the wisp. This curve should resist slightly to bending, and be more or less extensible, depending on the waviness of the hair inside the wisp.

In our implementation we use a chain of point-masses for modelling each wisp skeleton. These point-masses are linked by damped linear springs, which allows us to experiment with both wavy and almost straight hair (although a chain of rigid links would probably be more efficient for straight hair). Angular damped springs model a slight resistance to bending.

The first segment of the wisp skeleton is not part of the simulation. It is set to penetrate inside the character's head, and thus provides a reference point that lies always outside the head. This will prove useful in collision detection.

3.3. Wisp Envelope

Due to the very nature of the wisp (a set of hair strands), the behavior of its envelope is quite different from standard elastic models. A wisp should resist only slightly to radial compression or stretching. This can be observed as dynamic deformations of its envelope when the wisp accelerates, or as deformations of the hair strands near the envelope when an object slowly collides with the wisp. If the object moves further, the entire wisp itself should move. Moreover, wisp cross-sections are highly viscous. Hair within the wisp even-

tually comes back to its rest density after deformation, but does it in a perfectly damped way.

The wisp envelope models the thickness of the wisp around the skeleton. We represent it by a set of disconnected cross-sections subjected to only radial deformations, as in [7, 26]. Each cross-section is attached to a corresponding point-mass of the skeleton curve. It lies in the plane that bisects the angle between the two line segments linking the skeleton point-mass to its neighbors.

A coherent set of 2D coordinate frames for the cross-sections along a wisp skeleton will be needed to position the points modelling the wisp envelope, and the hair strands within a wisp (see Section 6). Since we rely on point dynamics only for animating the skeleton, such local frames are not provided by the skeleton structure. However, a simple method can be used for recursively generating these frames, since we are not modelling twisting effects. We attach to the head the first local frame associated with the first point of the wisp skeleton. Then, we successively define the other local frames using the quaternion that minimally rotates a given cross-section plane onto the next one. This quaternion is the one whose axis of rotation is the cross-product of the orientations of the two neighboring skeleton segments, and whose angle brings them one onto the other. This solution, as mentionned earlier, completely avoids twisting effects when a wisp moves.

A number of point-masses reside on the envelope in each cross-section plane. Since we are modelling only radial deformations, these points perform a 1D motion along a half line originating at its skeleton point-mass, and fixed with respect to the cross-section local coordinate system. An envelope point-mass always models the thickness changes of the same part of the envelope. We model the deformations of the wisp envelope by connecting envelope point-masses to their corresponding skeleton point-mass using soft, highly damped springs. The mass associated with an envelope point-mass represents the mass of the strands in its neighborhood, and therefore is much lighter (about 1/100) than its skeleton counterpart. Envelope point-masses of the same cross-section are also linked together by soft, highly damped springs. This allows for a certain conservation of volume, encouraging the wisp to stretch in one radial direction if it is compressed in the other. Figure 1 illustrates the structures of a wisp.

3.4. Wisp Animation

Despite the similarities between the structure of an animated wisp (i.e., an envelope coating a skeleton) and layered representations [8, 7, 26, 6], standard solutions for animating layered models do not apply to our problem. In previous layered models, the skeleton defines a new position for the envelope [8, 26], and some retroaction forces from the envelope to the skeleton can be added to alter the global motion in specific situations such as collisions [7, 6]. This framework enables to decouple the skeleton simulation from the envelope animation, using two different animation steps.

The situation we have to handle is different : we would like to capture the dynamic deformations of the envelope due to skeleton acceleration, together with the reaction forces these deformations apply on the skeleton. Modelling these deformations is essential in the case of hair wisps. For instance, the tip of a wisp can experience large radial deformations when the wisp swings. We achieve this by animating both skeleton and envelope point-masses within the same loop, processing a wisp from root to tip. During this process, we take care of always animating the associated skeleton point-masses before the associated envelope point-masses. This enables to handle the geometric constraint stating

that an envelope point-mass should always move along a half line defined with respect to neighboring skeleton point-masses. The detailed animation algorithm will be given in Section 5.

In collision situations, the virtual envelope of a wisp still does not behave as "a skin" over a skeleton. Although it should prevent penetrations inside an obstacle or inside another wisp of a different orientation, neighboring wisps of similar orientations should be allowed to interpenetrate. This specific anisotropic behavior is essential for capturing both the continuous aspect of hair at rest, and the highly discontinuous aspect after fast head motions. Section 4 explains how we deal with hair interactions.

4. HAIR WISP INTERACTIONS

Precisely modelling the complex interactions that occur inside long hair is a challenge, due to the extremely large number of interactions that may take place at the same time. Structuring hair into wisp volumes brings a partial solution to the problem since interactions between adjacent hair strands lying inside each wisp do not need to be modelled. However, since we wish to animate long chaotic hair, wisp interactions must be detected and processed locally, for each portion of the wisp envelope that coats a given segment of the skeleton. Wisp interactions may result from three distinct phenomena:

- Interpenetration between adjacent wisp segments whose skeletons locally move in the plane they form. Then, the two interacting wisps should locally react as a single thicker wisp.
- Collisions between two wisp segments that do not have similar orientations, and between which no interpenetration should be allowed.
- Collision between a wisp segment and the character's head or body.

Before discussing the models we use in these three different situations, we describe the way we deal with the main bottleneck of our approach, i.e., collision detection between pairs of wisp segments.

4.1. Collision Detection

Data structure. A naive implementation of wisp interactions would lead to $O(n^2)$ relatively expensive intersections tests, where n is the total number of wisp segments in the hair model. To reduce computations, we rely on a regular 3D grid data structure, built around the character and its hair. Each grid cell contains a list of wisp segments whose bounding box intersects the cell. At each animation step, the grid is used for quickly determining a shorter list of wisp segments susceptible to intersect. A mailbox parameter [1], which indicates the last time step when a given pair of such segments has been tested, ensures that each pair is tested only once.

Collisions between two wisps. Collision detection between two wisp segments is achieved first by comparing their bounding boxes. If they intersect, a collision is detected when at least one of the skeleton or envelope point-masses of a wisp segment penetrates inside the volume of the other wisp. This volume is defined by its two cross-sections. Because our wisp segments are mostly as wide as they are long, this simple and efficient intersection scheme has appeared sufficient.

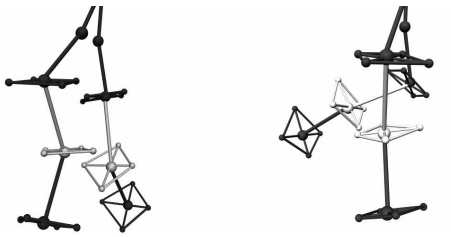

FIG. 2. On the left, the two wisps will interpenetrate, producing viscous friction, while on the right, whey will collide.

Wisp self-collisions. The previous collision detection method applies as well for non-adjacent segments of the same wisp. Wisp self-collisions are thus handled exactly the same way than collisions between two different wisps of hair.

Collision with the character model. The 3D grid data structure used for optimizing collision detection between wisps also references the polygons of the character for quickly detecting if a point-mass is close to the character.

4.2. Anisotropic Interactions between Wisps

Computing interactions between hair wisps is fairly different from modelling collisions between standard deformable bodies. Wisps are highly anisotropic, since they are just a virtual representation for a group of hair strands. While two perpendicular colliding wisps should be compressed in order to avoid intersection, interpenetration has to be allowed between neighboring wisps moving roughly in the plane they form (see Figure 2). Another major feature of hair self-interactions is that they are highly dissipative, since the extreme lightness of hair strands causes friction forces to play a very important role.

4.2.1. Interpenetration and friction between quasi-parallel wisps

If the two wisp segments lie in the same plane, and if their relative velocity is also roughly in this plane, wisps are allowed to interpenetrate. Since the hair strands of the wisps should not go through each other during interpenetration, we eliminate the residual relative velocity lying outside the plane. This is done with a velocity modification.

Suppose a point-mass at position $\mathbf{p_p}$, moving at velocity $\mathbf{v_p}$, has penetrated in a wisp segment whose velocity is $\mathbf{v_s}$ at this same position $\mathbf{p_p}$. Let $\sigma_\mathbf{p}$ and $\sigma_\mathbf{s}$ be the direction of the two wisp skeletons near $\mathbf{p_p}$. The direction $\sigma_\mathbf{p} \times \sigma_\mathbf{s}$ is normal to the plane containing both wisps, and the relative velocity to eliminate is $(\mathbf{v_p} - \mathbf{v_s}) \cdot (\sigma_\mathbf{p} \times \sigma_\mathbf{s})$. We do this by applying to the point-mass the velocity modification:

$$\mathbf{\Delta v_p} = -\left(\frac{(\mathbf{v_p} - \mathbf{v_s}) \cdot (\sigma_\mathbf{p} \times \sigma_\mathbf{s})}{2}\right)(\sigma_\mathbf{p} \times \sigma_\mathbf{s}).$$

Since similar modifications will be applied to point-masses of the other wisp at the same time step, this operation sets the velocities of the two wisps outside the plane that contains them, to the average of their velocities at the collision time.

In addition, the relative velocity of the two wisps in the plane that contains them is submitted to a viscous drag, approximating the friction of the strands against each other. This drag is again implemented through a direct velocity modification, proportional to the hair strand density at position $\mathbf{p_p}$ and to the time step.

4.2.2. Collision between wisps of different orientations

If viscous interpenetration conditions are not met, a very dissipative collision is modelled. We simply eliminate the relative velocities of the point-masses located in the contact area. Similarly to the viscous friction case, we use velocity modifications to set the velocities to the average of their initial values.

4.3. Collisions with the Character

Due to the extreme lightness of hair, the character reacts as a rigid body when it interacts with hair, and the effects of these interactions on hair itself are again, very dissipative.

Collision detection and response with the character has been implemented through two mechanisms, described in the next subsections. The first one avoids a collision using velocity modifications, therefore modelling sliding contact. When this mechanism has failed and a point-mass moves inside the character, a reaction to interpenetration triggers a position modification.

4.3.1. Sliding Contact

If a point-mass moves closer to the character's surface than a user-specified threshold, then the portion of the relative velocity approaching the point-mass to the surface is eliminated. A viscous drag is added between the tangential velocity of the wisp point-mass with respect to the surface and the velocity of the polygon, in order to model viscous friction.

4.3.2. Penetration Reaction

If the character moves fast, the sliding contact mechanism may not be sufficient for preventing some wisp point-masses from penetrating inside the model. Detecting these points is simplified by the specific order in which we process each wisp: A wisp is traversed from root to tip, each skeleton point-mass being processed before its associated envelope point-masses. Since the segment at the beginning of the wisp skeleton is guaranteed to be penetrating the head of the character, any subsequent intersection between a polygon of the character and a radial or skeletal spring of the wisp model means that the point-mass at the extremity of the current spring has moved into the character. Section 7 will show that even this simplified test results in about 30% processing time spent detecting collisions with the model (made of about 10,000 polygons).

If a collision occurs for a skeleton point-mass, it is moved out from the surface at a distance slightly under the sliding contact threshold. To avoid an undesired spring elongation along the skeleton curve, the new position is set at the current length of the spring from the skeleton point-mass that precedes the current one along the wisp. The same translation is applied to all the following point-masses of the wisp. This prevents the reaction from introducing instabilities in the system.

If penetration occurs for an envelope point-mass, it is moved out to the surface of the character along the half line it is constrained to lie on. Since the skeleton point-masses are moved out first, we are guaranteed that a portion of this half line is outside of the character. The sliding contact mechanism is then engaged with the nearest polygon, resulting in a completely inelastic collision.

5. IMPLEMENTATION

5.1. Animation Scheme for Hair Wisps

Due to our interaction processing mechanism, constraints yielding velocity and position modifications have to be allowed. We thus use the following animation algorithm for each time step:

1. Compute the set of applied forces; these forces include gravity, air resistance, and forces generated by springs.
2. Detect interactions between pairs of wisps, and between wisps and the character's model.
3. Process velocities, using the current velocity values and applied forces; then apply the velocity modifications.
4. Process positions using the current position and the new velocity; then apply the position modifications.

For items 1, 3, and 4, all point-masses are processed before going to the next time step. A given wisp of hair is processed from root to tip, the envelope point-masses being processed after their associated skeleton point-mass. For envelope points, all operations are projected onto the permitted axis, defined in the newly computed cross-section plane. This holds for forces, velocity modifications, and position modifications.

In practice, interactions are not processed at every evaluation of the system, since wisp motion is relatively slow with respect to the integration step. This yields a different time step for some of the velocity modifications, which is easily handled by the algorithm above. The new velocity modification $\mathbf{\Delta v}'_{\mathbf{p}}$ is obtained by:

$$\mathbf{\Delta v}'_{\mathbf{p}} = \mathbf{\Delta v}_{\mathbf{p}} \left(\frac{\Delta t_c}{\Delta t} \right)$$

where Δt_c is the time step used for the interaction, and Δt is the main integration step.

5.2. Numerical Integration

Some techniques use implicit integration [4, 11] for enabling larger time steps in mass-spring simulations. However, this integration method is hard to combine with constraints such as speed and position modifications produced by interaction processing. Our current implementation thus relies on a classical Euler explicit scheme, although trying to use recent solutions for combining implicit schemes with constraints [28, 12] would probably be a better solution.

As much as stiffness, the amount of damping applied to a spring may yield instabilities in an explicit integration scheme. Indeed, too large damping forces could inverse the particles relative speeds instead of setting them to zero. Since our hair wisp model is mostly based on soft, highly-damped spring, this problem may occur in our case. We thus formulate the damping of springs as a velocity modification rather than as a damping force.

Consider the case in which point-masses at positions $\mathbf{p_1}$ and $\mathbf{p_2}$, of masses m_1 and m_2, and of velocities $\mathbf{v_1}$ and $\mathbf{v_2}$ respectively, are linked by a damped spring. The force applied on the point-mass at $\mathbf{p_1}$ is generally written as

$$\mathbf{f_1} = [-k(\|\mathbf{p_1} - \mathbf{p_2}\| - r) - \nu((\mathbf{v_1} - \mathbf{v_2}) \cdot \mathbf{u})]\,\mathbf{u}$$

where k is the stiffness of the spring, r is its rest length, ν is the damping coefficient, and $\mathbf{u}$ is the normalized vector in the direction from $\mathbf{p_2}$ to $\mathbf{p_1}$. The damping term involving ν

should always tend to reduce the relative velocities of the two point-masses. However, if the integration step is too large, it may change the sign and even increase the magnitude of the relative point-mass velocity along the spring direction, thus yielding divergences. We solve the problem by formulating viscosity as a velocity modification rather than as a damping force. The damping term above yields the following velocity modifications:

$$\Delta \mathbf{v_1} = -\left(\frac{\nu dt}{m_1}\left((\mathbf{v_1} - \mathbf{v_2}) \cdot \mathbf{u}\right)\right)\mathbf{u}.$$

In order to avoid that the effect of damping reverses the direction of the point-masses relative velocities, we change this term into:

$$\Delta \mathbf{v_1} = -\left(\min\left(\nu dt, 1\right)\frac{m_2}{m_1 + m_2}\left((\mathbf{v_1} - \mathbf{v_2}) \cdot \mathbf{u}\right)\right)\mathbf{u}.$$

With this formulation, the relative velocities of the point-masses along $\mathbf{u}$ decrease in a similar way than before if the time step is small enough ($dt < \frac{1}{\nu}$). If this condition is not met, the values of velocity modifications for the two point-masses set their relative speed to zero instead of inverting its direction, which is the correct solution. The factor $\frac{m_2}{m_1+m_2}$ ensures that the action-reaction laws are still obeyed, since the velocity modifications correspond to the application of opposite forces of value $\frac{1}{dt}\frac{m_1 m_2}{m_1+m_2}\left((\mathbf{v_1} - \mathbf{v_2}) \cdot \mathbf{u}\right)$ along $\mathbf{u}$ to each of the point-masses.

6. RENDERING DIFFERENT HAIR STYLES

Once the position and shape of a wisp have been determined, hair strands must be rendered in this wisp. For faster previewing, we simply use OpenGL segments with a shading model similar to Watanabe and Suenaga [29]. The resulting colors at the vertices are linearly interpolated along its segment. For antialiased results appearing in the finer images and animations, we used a similar shading model, but with Renderman. The large number of hair strands prevents us from storing much related data for each strand. We rely instead on the pseudo-random number generator for defining the hair strand position inside a wisp. Because the number of random values required for modelling the hair is always the same, we can reset the seed of the pseudo-random number generator at the beginning of every frame, thus providing the necessary coherence.

In practice, we use only four envelope point-masses for modelling the cross-sections of wisps. A cross-section defines a 2D coordinate system whose origin is the skeleton point-mass, and the four envelope point-masses determine the orthogonal axes and scaling factors. A set of 2D points, one per hair strand in the wisp, are generated in a disk according to a given distribution (see Figure 3). This set is scaled and positioned in 3D for each cross-section of the wisp. The 3D points resulting from a 2D point from this set are then linked to form the control points of a Catmull-Rom piecewise cubic spline defining a hair strand. The first and last control points are duplicated so the curve goes through all the control points. In order to reduce the apparition of patterns in the strands of a wisp, a user-specified jittering is applied to each control point, moving it towards its skeleton point-mass. Jittering is also applied to reduce the total length of the strand. Any strand therefore always remains within the wisp envelope.

Wavy hair can be produced by specifying the number of waves the wisp should contain as well as its amplitude when the wisp's radius is at rest. Then, as the wisp cross-sections

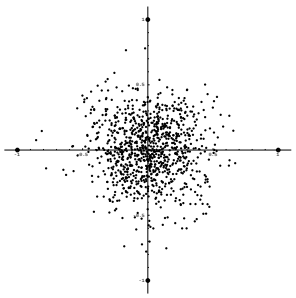

FIG. 3. A typical distribution of hair strand control points in a wisp cross-section of unit radius

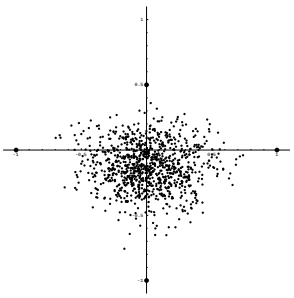

FIG. 4. Deformation of this distribution as the wisp cross-section deforms

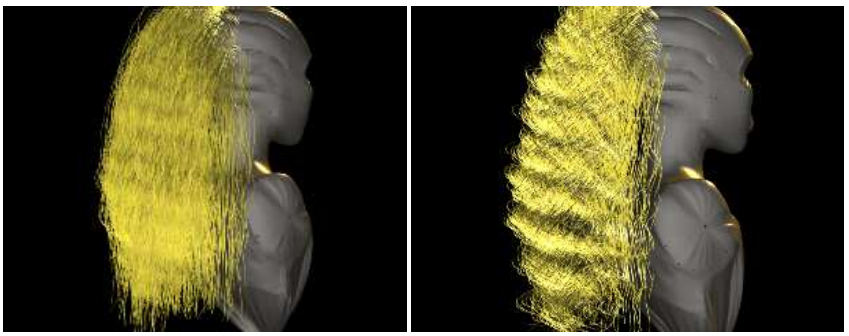

FIG. 5. Increasing the waviness of hair

are more or less compressed during the animation, the amplitude at those cross-sections is adjusted accordingly. Likewise, if a wisp segment stretches due to gravity for example, the amplitude is again scaled, along with the frequency for this segment so that the number of waves in this segment is maintained. Figure 5 shows how the waviness parameter can be used to create different hair styles. Our wisp model uses the same wisp sections, simply reducing the stiffness of the skeletal springs to allow stretching of the wisp lengths.

In Figure 6, two hair styles are created by reducing the number of hair strands in a wisp or increasing the thickness of each strand within a wisp.

In Figure 7, the shadows created from a simple shadow buffer demonstrate how important shadows are to create an impression of depth within hair. Antialiasing and semi-transparency are two issues that deep shadow maps [21] should improve at reasonable cost.

7. ANIMATION RESULTS

7.1. Two Wisps Balancing under Gravity

A few selected frames from animations of two wisps balancing under gravity are provided in Figure 8. They illustrate the modifications due to the wisp interactions that we modelled. In the first row (left to right), no interactions are simulated, and the two wisps simply go through each other. In the second row, a viscous friction reduces the relative speed of the wisps when they penetrate each other and when they separate. In the third

FIG. 6. Hair styles

FIG. 7. Shadows cast by hair

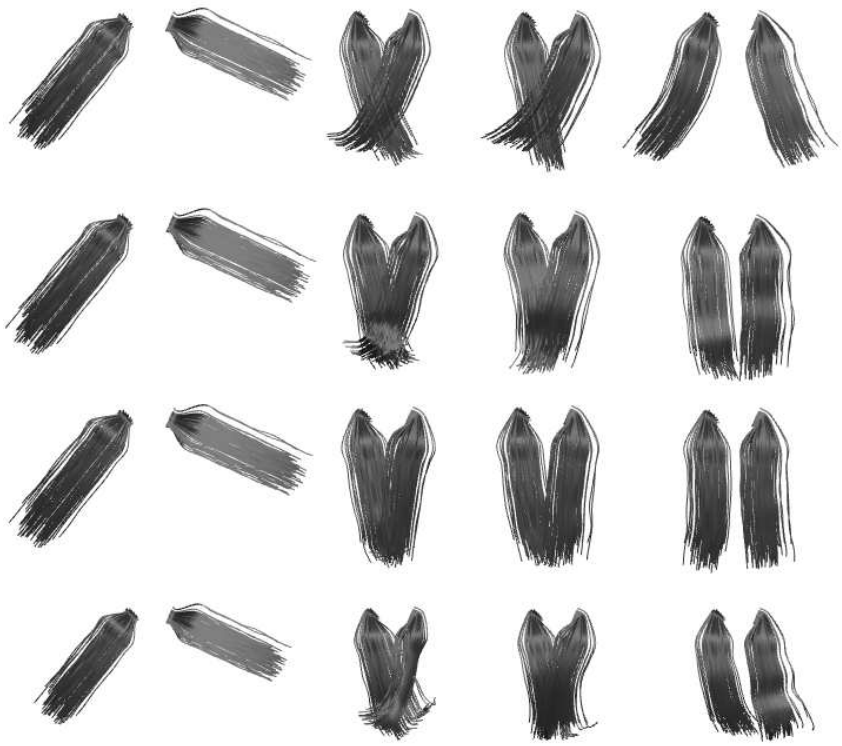

FIG. 8. The first row shows frames from an animation of two wisps without any interaction; the second row with friction enabled; the third row with collision enabled; the fourth row with both collision and friction enabled.

row, a collision is simulated when the relative speed of the segments forms an angle superior at 25^o from a plane defined by the two segments. In the fourth row, both collisions and viscous frictions are simulated.

Figure 9 illustrates the effect of varying the viscosity (0.5 in the first row and 50 in the second row) as the two already interpenetrating wisps separate at different speeds under gravity. With a large viscosity value, the two wisps tend to remain attached together during most of the motion.

7.2. Hair Animations on Captured Motion

The first animated sequence on the web site associated with this paper[4] shows the movement of long hair recorded during a motion tracking session. It illustrates the complex nature of hair motion, where wisps are easily observed. One can notice also how certain wisps stretch in width depending on the head motion, and how certain wisps slide on top of each other. The data acquired from this tracking session was then applied on a synthetic character with three different hair lengths (short, medium, and long). The short hair style is composed of 199 wisps with 4.0 segments per wisp. The medium hair style is composed

[4] www.iro.umontreal.ca/labs/infographie/papers/Plante-2002-CCHM

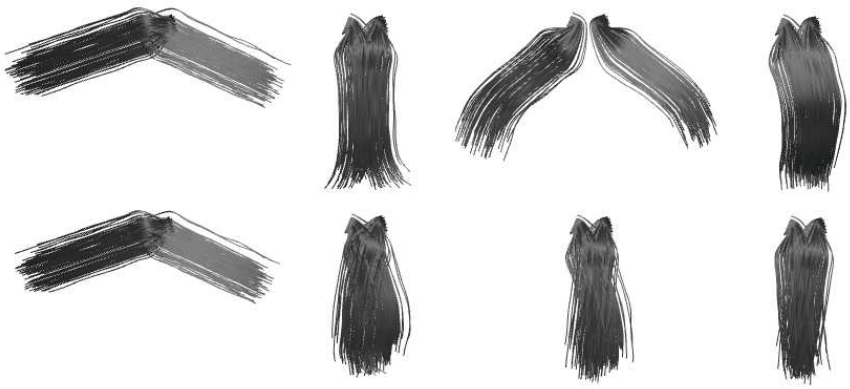

FIG. 9. Two interpenetrating wisps separate under the action of gravity. Viscosities are set to 0.5 (top row) and 50 (bottom row).

of 73 wisps with 9.9 segment per wisp on average. The long hair style is composed of 56 wisps with 7.7 segments. In the three hair styles, wisps are usually fairly wide. On average, a wisp segment intersects 17.5 other segments in the short hair style, 21.0 in the medium hair style, and 12.8 in the long hair style. To initiate the hair styles, each wisp is first stretched perpendicularly to the scalp. It is then simply relaxed under gravity. [5]

All three simulations were computed at 1000 Hz, but collision detection was computed at 120 Hz, and wisp self-interactions at 24 Hz. Even with this choice, 92 to 95% of the total computation time was spent on detecting wisp self-interactions (64 to 70% of this percentage) and wisp collisions with the character (30 to 36%). The total animation computations [6] (264 frames) took 3.2 hours for the short hair style, and 2.9 hours for the medium and long hair style (which gives an average rate of 0.68 minutes per frame).

These statistics confirm that the complete treatment of wisp interactions is the most significant portion of the total computation time in our simulation of hair motion. Avoiding the simulation at 1000 Hz thanks to implicit integration would probably improve performances, but not as much, since the collision processing cost would remain.

The benefits of processing interactions are however illustrated by the improved visual aspect of the animations. Figure 11 presents two frames extracted from the tracking session and animated sequence. Although the movements of the real person and of the virtual character were similar, they were not identical, as the magnetic trackers would have influenced the movement of the hair. Moreover, we did not attempt to reproduce live conditions, such as initial hair configuration and actual friction coefficients. Nonetheless, several similarities in the motion and the distribution of the wisps can be observed, as illustrated in these two frames.

8. CONCLUSION

As our results show, offering a good modelling of interactions is essential for achieving realistic hair animations. This paper has presented the first approach for modelling

[5] In the three animated sequences, the small hair motion at the top of the head is due to the wisps that did not have enough time to settle before we started applying the recorded character's movements.

[6] All statistics were acquired on an SGI O2 with one R12000 processor at 270 MHz.

FIG. 10. A few frames from an animated sequence with various hair lengths

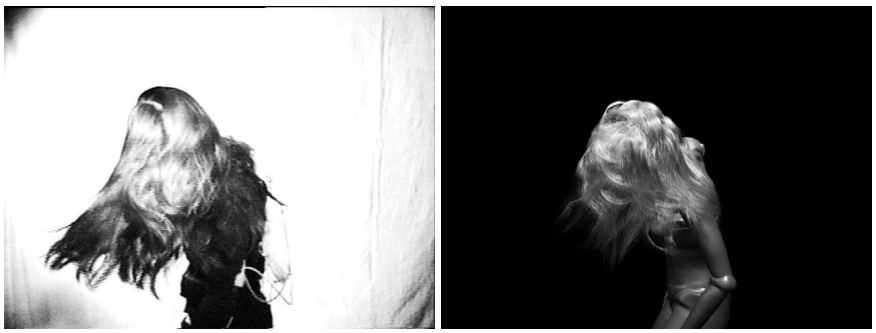

FIG. 11. Comparing real and synthetic results

the complex self-interactions and the multiple collisions with obstacles that occur during the motion of thick, long hair. This is achieved thanks to a hierarchical structure for hair (i.e., a number of hair wisps representing clusters of hair strands) which allows the modelling of the interaction phenomena at two different levels of detail. Friction between neighboring hair strands of similar orientations is modelled by associating a highly viscous anisotropic behavior to the virtual volume of a wisp. Other interactions are modelled at the wisp level. They include viscous friction between wisps of similar orientations allowed to interpenetrate, collisions between other wisps, and collisions with the character body. The simulation of all these interactions greatly enhances the visual realism of animations. In particular, both continuous and highly discontinuous hair configurations are generated during motion, as can be observed in real life.

We have chosen a very simple representation for modelling wisps of hair, relying only on point dynamics. Two coupled sub-models have been used: a skeleton curve modelled as a chain of springs to capture both straight and wavy hair wisps, and a visco-elastic envelope for which only radial deformations are allowed. During animations, an algorithm generates a consistent set of coordinate systems for the envelope cross-sections, considering the fact that we are neglecting the effects of dynamic twisting of hair wisps. This is not too restrictive, since we have shown that the method handles both straight and wavy hair. Similarly, curly hair could be modelled using a static twisting of the hair strands drawn inside each wisp. The coordinate systems determining hair positions would be progressively rotated along the wisp, with respect to the original coordinate systems of wisp cross-sections. Interpenetration between curly wisps should not be allowed, even when their skeleton curves have similar orientations. Alternately, dynamic twisting of the wisps could be implemented by using oriented particles [22] for the skeleton point-masses.

Other essential directions for future work include developing a simpler and faster approximative simulation to allow the user to set the various wisp parameters more efficiently, the design of a hair style editor to create more realistic initial hair styles, and the use of volumetric texels [23] to improve the illumination and shadowing of our wisps. We are currently investigating how an adaptive subdivision and merging of wisps could improve on these various aspects.

ACKNOWLEDGMENT

We would like to thank Arash Habibi of Université Louis Pasteur, for his help in the early stages of this work, and François Faure, from iMAGIS, for rereading an early version of the paper. This research has been made possible through a NSERC Industry Partnership Grant with Taarna Studios, inc. It also benefited from a grant from the “Centre Jacques Cartier”.

REFERENCES

1. J. Amanatides and A. Woo. A fast voxel traversal algorithm for ray tracing. In *Eurographics ’87*, pages 3–10, August 1987.
2. M. Ando and S. Morishima. Expression and motion control of hair using fast collision detection methods. In *Image Analysis Applications and Computer Graphics. Third International Computer Science Conference. ICSC ’95 proceedings*, pages 463–470, 1995.
3. K. Anjyo, Y. Usami, and T. Kurihara. A simple method for extracting the natural beauty of hair. In *Computer Graphics (SIGGRAPH ’92 Proceedings)*, volume 26, pages 111–120, July 1992.
4. D. Baraff and A. Witkin. Large steps in cloth simulation. In *SIGGRAPH 98 Conference Proceedings*, Annual Conference Series, pages 43–54. ACM SIGGRAPH, July 1998.
5. D. Baraff and A.P. Witkin. Large steps in cloth simulation. *Proceedings of SIGGRAPH 98*, pages 43–54, July 1998.

6. M.-P. Cani-Gascuel and M. Desbrun. Animation of deformable models using implicit surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 3(1):39–50, January – March 1997.
7. M.-P. Cani-Gascuel, A. Verroust, and C. Puech. Animation and collisions between complex deformable bodies. In *Graphics Interface '91*, pages 263–270, June 1991.
8. J.E. Chadwick, D.R. Haumann, and R.E. Parent. Layered construction for deformable animated characters. In *Computer Graphics (SIGGRAPH '89 Proceedings)*, volume 23, pages 243–252, July 1989.
9. L.-H. Chen, S. Saeyor, H. Dohi, and M. Ishizuka. A system of 3d hair style synthesis based on the wisp model. *The Visual Computer*, 15(4):159–170, 1999.
10. A. Daldegan, N. Magnenat-Thalmann, T. Kurihara, and D. Thalmann. An integrated system for modeling, animating and rendering hair. In *Eurographics '93*, volume 12, pages 211–221, 1993.
11. M. Desbrun, P. Schroeder, and A. Barr. Interactive animation of structured deformable objects. In *Graphics Interface '99*, pages 1–8, June 1999.
12. B. Eberhardt, O. Etzmus, and M. Hauth. Implicit-explicit schemes for fast animation with particle systems. In *Computer Animation and Simulation '00*, pages 137–154, August 2000.
13. J. Funge, X. Tu, and D. Terzopoulos. Cognitive modeling: Knowledge, reasoning and planning for intelligent characters. *Proceedings of SIGGRAPH 99*, pages 29–38, August 1999.
14. M. Gleicher. Retargeting motion to new characters. *Proceedings of SIGGRAPH 98*, pages 33–42, July 1998.
15. S. Hadap and N. Magnenat-Thalmann. Interactive hair styler based on fluid flow. In *Computer Animation and Simulation '00*, pages 87–100, August 2000.
16. S. Hadap and N. Magnenat-Thalmann. Modeling dynamic hair as a continuum. In *Eurographics 2001 Proceedings*, volume 20, September 2001.
17. T-Y. Kim and U. Neumann. A thin shell volume for modelling human hair. In *Computer Animation '00*, pages 121 – 128, May 2000.
18. C. Koh and Z. Huang. Real-time animation of human hair modeled in strips. In *Computer Animation and Simulation '00*, pages 101–112, August 2000.
19. C. Koh and Z. Huang. A simple physics model to animate human hair modeled in 2D strips in real time. In *Computer Animation and Simulation '01*, pages 127–138, September 2001.
20. D.-W. Lee and H.-S. Ko. Natural hairstyle modeling and animation. *Graphical Models*, 63(2):67–85, March 2000.
21. T. Lokovic and E. Veach. Deep shadow maps. *Proceedings of SIGGRAPH 2000*, pages 385–392, July 2000.
22. J.-C. Lombardo and C. Puech. Oriented particles: A tool for shape memory objects modelling. In *Graphics Interface '95*, pages 255–262, 1995.
23. F. Neyret. Modeling, animating, and rendering complex scenes using volumetric textures. *IEEE Transactions on Visualization and Computer Graphics*, 4(1):55–70, January-March 1998.
24. E. Plante, M.-P. Cani, and P. Poulin. A layered wisp model for simulating interactions inside long hair. In *Computer Animation and Simulation '01*, September 2001.
25. R.E. Rosenblum, W.E. Carlson, and E. Tripp. Simulating the structure and dynamics of human hair: Modelling, rendering and animation. *The Journal of Visualization and Computer Animation*, 2(4):141–148, 1991.
26. J. Shen and D. Thalmann. Interactive shape design using metaballs and splines. In *Implicit Surfaces '95*, pages 187–196, April 1995.
27. Z.-C. Shih and H.-D. Guo. The modeling and animation of human hair. *The Journal of Information Science and Engineering*, 11(3):465–488, 1995.
28. P. Volino and N. Magnenat-Thalmann. Accurate collision response on polygonal meshes. In *Computer Animation '00*, pages 179–188, May 2000.
29. Y. Watanabe and Y. Suenaga. A trigonal prism-based method for hair image generation. *IEEE Computer Graphics and Applications*, 12(1):47–53, January 1992.
30. J. Wilhelms and A. Van Gelder. Anatomically based modeling. *Proceedings of SIGGRAPH 97*, pages 173–180, August 1997.
31. X.D. Yang, Z. Xu, J. Yang, and T. Wang. The cluster hair model. *Graphical Models*, 62(2):85–103, March 2000.