ACM SIGGRAPH / Eurographics Symposium on Computer Animation 2018
T. Beeler and N. Thuerey
(Guest Editors)

Volume 37 (2018), Number 8

HairControl: A Tracking Solution for Directable Hair Simulation

A. Milliez[1,2] R. W. Sumner[1,2] M. Gross[1,2] B. Thomaszewski[1,3]

[1]Disney Research [2]ETH Zurich [3]Université de Montréal

Figure 1: *An unconstrained simulation of Celia's ponytail fails to meet the artistic intent of reaching the glass (*left*). To edit the initial simulation, the artist authors a guide animation as indicated in the center. Our method constrains the hair motion to follow the guide in an averaged sense, and the resulting simulation on the right knocks the glass over as desired (*right*).*

Abstract

We present a method for adding artistic control to physics-based hair simulation. Taking as input an animation of a coarse set of guide hairs, we constrain a subsequent higher-resolution simulation of detail hairs to follow the input motion in a spatially-averaged sense. The resulting high-resolution motion adheres to the artistic intent, but is enhanced with detailed deformations and dynamics generated by physics-based simulation. The technical core of our approach is formed by a set of tracking constraints, requiring the center of mass of a given subset of detail hair to maintain its position relative to a reference point on the corresponding guide hair. As a crucial element of our formulation, we introduce the concept of dynamically-changing constraint targets that allow reference points to slide *along the guide hairs to provide sufficient flexibility for natural deformations. We furthermore propose to regularize the null space of the tracking constraints based on variance minimization, effectively controlling the amount of spread in the hair. We demonstrate the ability of our tracking solver to generate directable yet natural hair motion on a set of targeted experiments and show its application to production-level animations.*

1. Introduction

Expressive characters are the beating heart of computer animated movies. Bringing these characters to life requires skilled artists who laboriously create posture and facial expressions through keyframing. In addition to these primary motions, the secondary motions of flesh, clothing, and hair are just as essential to making characters believable and compelling. Rather than passively following the primary motion, hair often takes a decidedly *active* role in defining the personality, style, and attitude of a character. But in order to do so successfully, hair animations must not only be complex and physically plausible, they also have to comply with the artistic intent of the animator, leading to the classical dilemma between physical simulation and art direction.

Hair simulations can, to some extent, be controlled by carefully selecting values for the various parameters that influence the resulting motion: material coefficients such as stretching and bending stiffness or damping, as well as restitution and friction parameters for collisions and contact. But even though this parameter space is already quite large for trial-and-error exploration, it is typically too restrictive to achieve the desired motion style or meet some other artistic intent. It is therefore common practice to introduce *target shapes* and corresponding control forces that attract the simulation towards desired states; see, e.g., Petrovic et al. [PHA05] for hair, or Thuerey et al. [TKRP06] for fluids. Control forces add another set of parameters, making the problem of finding the *right* values all but intractable for high-resolution simulations. For this reason, artists typically resort to lower-resolution simulations, where iterations are faster and manual edits possible. But unfortunately, the parameter values determined in this way can only serve as an initial guess for the full-resolution simulation, which often behaves very different from its coarse counterpart when the same parameters are used. To avoid expensive iterations in full resolution, an alternative is to up-sample the coarse simulation using geometric [CCK05] or data-driven [CZZ14] interpolation. However, the drawback of this approach is that, despite increased geometric complexity, the high-resolution motion does not exhibit more physical detail than the low-resolution input.

In this work, we propose a new approach for adding artistic control to physics-based hair simulation. Our method takes as input a coarse animation that captures the bulk motion of the hair. However, instead of trying to infer parameters, we constrain the full-resolution simulation to follow the coarse-scale motion in a spatially-averaged sense. In this way, the full-resolution animation strictly adheres to the artistic intent for the bulk motion, while enhancing it with detailed deformations and dynamics that only simulation can produce. Our approach draws great inspiration from TRACKS—a tracking solver for deformable surface animations by Bergou et al. [BMWG07]. TRACKS works on two discretizations of the same surface, one coarse, one fine, which are decomposed into patches with known coarse-to-fine correspondence. The central idea is then to introduce *tracking constraints* requiring that, for each surface patch, the generalized centers of mass in the high-resolution simulation should coincide with their coarse-scale counterparts.

While our method builds on the same conceptual basis, the problem of tracking hair simulations is very different. Due to its quasi-inextensibility, cloth exhibits strong local coupling between its degrees of freedom for any in-plane direction, leading to a high degree of spatial and temporal coherence in its motion. By contrast, hair exhibits such strong coupling in only one direction—along individual hairs. The coupling across hairs happens through collisions, which is a different and weaker mechanism that leads to less spatial and temporal coherence.

This difference in coupling leads to several challenges. When using simulations with different numbers of hairs, it is not obvious what exactly the coarse-to-fine correspondence between these simulations should be. Furthermore, center-of-mass type tracking constraints generally exhibit a high-dimensional null space, i.e., the set of particle motions that leave a given constraint unchanged. For the case of cloth, this null-space is strongly regularized by the resistance to in-plane stretching. Such strong coupling lacking, the regularization for hair is much weaker, requiring further control mechanisms to prevent hair from producing undesirable motion in the null-space. Finally, while a static map for coupling of low- and high-resolution simulations works well for deformable surfaces, this approach causes hair to stretch and compress, leading to unnatural motion or even failure to converge.

To address these challenges, we propose three novel concepts:

- we introduce a coarse-to-fine correspondence scheme for hair simulations and formulate tracking constraints on this basis,
- we propose a dedicated regularizer for the constraint null space using the concept of variance minimization, and
- we allow for dynamically-changing constraint targets that can slide along their guide hairs in order to provide more flexibility for natural deformations.

We combine these ideas with a variational formulation of implicit Euler integration into constrained optimization problem that we solve using sequential quadratic programming (SQP).

The results that we show in this work were generated with a prototype implementation based on a simplified model of hair mechanics (straight hair without resistance to twisting) and simple impulse-based collision handling (no coupling with hair mechanics). These design choices currently limit the complexity of our results and the combination of our method with state-of-the-art hair solvers such as the one by Kaufman et al. [KTS*14] is left for future work. Nevertheless, our preliminary results demonstrate the ability of our tracking solver to generate directable yet natural hair motion.

2. Related Work

Hair Mechanics Animating hair is a central problem in computer animation and, consequently, this topic has received a lot ot attention from the graphics community; see, e.g., Ward et al. [WBK*07] for an overview. Among the simplest approaches are mass-spring systems [RCT91] which, in their most basic form, model the resistance of hair to bending and stretching, but ignore twisting. This assumption is generally considered adequate for straight hair, but curly hair requires proper modeling of direction-dependent rest curvatures and bending stiffness—qualities offered by rod models based on Kirchhoff [BAC*06, BWR*08, BAV*10, IMP*13] or Cosserat [Pai02,ST07] theories. For simplicity, we focus on straight hair in this work and adopt a mass-spring model similar to Rosenblum et al. [RCT91]. However, our tracking solver is largely independent of the underlying energy terms and could, in principle, be combined with more sophisticated rod simulation models supporting curly hair.

Collisions & Contact Apart from the mechanics of individual hair strands, proper handling of collisions and contact is essential for high-quality animations. Many existing works use penalty forces [CJY02,SLF08,IMP*13] or impulses [SLF08] adapted from cloth animation [BFA02]. However, recent work has also started to explore more accurate ways of modeling frictional contact [BDCDA11, DBDB11] and capturing nonlinearities in the elastic response [KTS*14].

Continuum and Clump Models Simulating every individual hair is computationally expensive, both because of the large number of degrees of freedom and the combinatorial complexity of handling collisions between individual hairs. In seeking alternative solutions, one line of research has investigated so-called *clump* models that simulate only a number of guide hairs, each representing many detail hairs. The motion for the detail hairs is then generated using geometric [CCK05] or data-driven [CZZ14, CZZ16] interpolation, or a statistical model for synthesis [CCK05]. Adaptive clump models [BKCN03, WL03] support splitting and merging clumps in order to focus computations on regions in which detail is required most. Another approach for reducing complexity is to resort to a volumetric hair representation and model the bulk motion based on fluid dynamics [HMT01]. This approach allows for efficient treatment of hair-hair interactions, but the purely Eulerian representation leads to a loss of detail motion. To address this drawback, McAdams et al. [MSW*09] propose a hybrid approach that takes advantage of the Eulerian representation while retaining detail through Lagrangian hair-hair collisions.

Although clump and continuum models have their advantages, they cannot produce the level of detail that full simulation can—a fact that was impressively demonstrated by the high-resolution simulations of Selle et al. [SLF08] and Kaufman et al. [KTS*14]. However, the sheer complexity of these simulations renders a trial-and-error approach for artistic control infeasible. Our tracking solver aims at filling this gap by enabling high-resolution simulations that reliably follow coarse-scale, artist-controlled input animations while exhibiting the desired degree of detail.

Simulation Control The overall look and feel of simulated deformable materials is governed by their material model and its parameters. While tweaking the coefficients of a standard constitutive model can be very tedious, recent works by Xu et al. [XSZB15] and Li et al. [LB15] have proposed artist-friendly material models that provide more intuitive control over the resulting static and dynamic behavior. Aiming at stylized artistic materials, Martin et al. [MTGG11] describe an example-based approach for designing context-sensitive deformation behavior. In addition to custom-tailored materials, however, animations typically require more direct ways of control in order to meet the artistic intent.

There is a large number of different methods for directing simulations by virtue of control forces. As one particular example, Kondo et al. [KKA05] animate the rest shape of an elastic character in order to compute control forces that induce naturally looking motion. Coros et al. [CMT*12] and Tan et al. [TTL12] use similar mechanisms to induce control forces, but automatically compute changes in rest state such that the resulting simulation tracks some high-level motion objectives. Aiming at real-time applications with user interaction, Barbic et al. [BP08] introduce a feedback controller to generate forces that gently guide the simulation back to its input animation. Finally, real-time interfaces such as the work by Xing et al. [XKG*16] offer users to add external forces into simulated 2D scenes through painting gestures, letting them create highly stylized elements.

Rather than providing a dense animation as input, an alternative approach is to use spacetime constraints [WK88] that describe the desired motion through a temporally-sparse set of keyframes. The essence of this approach is to minimize the amount of control forces required to meet the keyframes over the entire animation, leading to large coupled optimization problems. Previous work has demonstrated that the adjoint method can, to some extent, alleviate the computational complexity when controlling smoke [TMPS03], liquids [MTPS04], or cloth [WMT06] animations. Another way of accelerating optimal control of elastic objects is by using subspace simulation [BdSP09, HSvTP12, SvTSH14, LHdG*14].

In contrast to solid materials, hair typically exhibits rich dynamic detail that is difficult to capture with a low-dimensional subspace. But even when leaving aside questions of computational tractability, creating keyframes for complex hair geometries that lead to detailed yet natural looking motion is a non-trivial problem in itself.

Instead of interpolating sparse keyframes, the tracking solver by Bergou et al. [BMWG07] relies on a temporally-dense but spatially-coarse input animation that captures the bulk motion of a deformable surface. The coarse animation is then enhanced with physics-based detail by virtue of a high-resolution simulation that is constrained to follow the input in a spatially-averaged sense. A conceptually similar approach was recently presented by Li et al [YL16], who enrich triangle mesh animations with physics-based detail created through solid finite element simulations. While our approach shares the same conceptual basis, hair exhibits much less spatial and temporal structure than deformable surfaces and solids. As we show in this work, a naive application of this tracking paradigm leads to excessive compression and buckling in the detail hair, making the tracked simulation slow and unstable. We solve this problem by introducing dynamically changing constraint targets that can slide on their corresponding guide hairs in order to accommodate the natural resistance of the detail hair to stretching and compression. As another central difference, our tracking constraints exhibit a large null-space, but unlike for deformable surfaces and solids, relative motion between hairs is regularized only by momentum, not elasticity. We show that a regularizer based on variance minimization not only eliminates numerical problems, but can be used to effectively control the desired amount of spreading.

3. Tracking Solver

Given a coarse animation of guide hairs as input, the tracking solver should produce a higher-resolution simulation of detail hairs that follow the input in a spatially-averaged sense, preserving its overall characteristics while enhancing the motion with physically-simulated detail. In the following, we first describe how to map between guide and detail hairs (3.1), then introduce tracking constraints that preserve coarse-scale motion (3.2), and a regularizer for the constraint null-space based on variance control (3.3). Finally, we combine these concepts into constrained optimization problem and describe its solution (3.4).

3.1. Coarse-To-Fine Correspondence

The input to our method are a coarse hair set $\mathcal{G}$ of guide hairs $\mathbf{g}_j$ and a higher-resolution set $\mathcal{D}$ of detail hairs $\mathbf{d}_j$. We represent all guide and detail hairs as piecewise linear curves, whose vertices we refer to as $\mathbf{g}_{j,i}$ and $\mathbf{d}_{j,i}$, respectively. In order to establish a correspondence between $\mathcal{G}$ and $\mathcal{D}$, we assign detail hairs to guide hairs based on their integrated Euclidean distance, which we quantify as

$$d(\mathbf{d}_j, \mathbf{g}_i) = \sum_k ||\mathbf{d}_{j,k} - \text{prox}(\mathbf{d}_{j,k}, \mathbf{g}_i)||^2 , \quad (1)$$

where $\text{prox}(\mathbf{d}_{j,k}, \mathbf{g}_i)$ returns the interpolated point on the guide hair $\mathbf{g}_i$ closest to a given detail hair node $\mathbf{d}_{j,k}$. We then assign each detail hair $\mathbf{d}_j$ to the guide hair with the smallest distance value.

With this horizontal correspondence established, we proceed to a vertical decomposition of the detail hairs. To this end, we sample each guide hair $\mathbf{g}_j$ equidistantly with *tracking points* $\mathbf{p}_{j,k}$. Each detail hair $\mathbf{d}_i$ assigned to a given guide $\mathbf{g}_j$ is then decomposed into a number of segments $\mathbf{s}_{i,k}$ containing the set of nodes $\mathbf{d}_{i,l}$ that are closer to $\mathbf{p}_{j,k}$ than to any other tracking point on $\mathbf{g}_j$. The union of segments $\mathbf{s}_{i,k}$ from all detail hairs corresponding to a given tracking point $\mathbf{p}_{j,k}$ defines a cluster $\mathcal{C}_{j,k} = \cup_i \mathbf{s}_{i,k}$ around the guide, which is used to define tracking constraints as explained next.

3.2. Tracking Constraints

Having established the basic structure for coarse-to-fine correspondence, we formulate constraints that couple the simulation of $\mathcal{D}$ to the coarse-scale motion of $\mathcal{G}$. In principle, we would like the detail hairs to remain close to their corresponding guides throughout the entire animation. However, asking that each individual detail hair should strictly maintain a given relative position with respect to its guide would overly restrict their motion and suppress dynamic detail altogether. Following Bergou et al. [BMWG07], we therefore ask that the detail hairs should track the motion of their guides not in a strict sense, but in a spatially averaged way. Applied to the coarse-to-fine structure described above, this notion of spatially-averaged tracking translates naturally into conditions on the center of mass of a given set of hair segments. To this end, we define the center of mass of a given cluster as

$$\mathbf{c}_{j,k} = \frac{1}{M_{j,k}} \sum_{(i,l)\in\mathcal{C}_{j,k}} m_{l,i}\mathbf{d}_{l,i} \ , \qquad (2)$$

where $M_{j,k}$ is the summed mass of all nodes belonging to the cluster. The center of mass for a given cluster will generally not coincide with its corresponding tracking point $\mathbf{p}_{j,k}$ on the guide. In order for the cluster to track the motion of its guide, we therefore ask that the relative position of a cluster's center of mass, as measured in a local reference frame, be preserved at all times as described next.

Updating Offsets Let $\mathbf{r}_{j,k} = \mathbf{c}_{j,k} - \mathbf{p}_{j,k}$ denote offset vectors describing the relative position of the center of mass $\mathbf{c}_{j,k}$ of a given cluster with respect to its corresponding tracking point $\mathbf{p}_{j,k}$. In principle, we would like the cluster to maintain its relative position to its guide, but asking that $\mathbf{r}_{j,k}$ be preserved with respect to a global frame of reference would obviously lead to artifacts as guide hairs move and deform during animation. In order for the detail hairs to track this motion, we have to update the offset vectors to reflect these transformations. For this purpose, we encode offset vectors using local coordinates with respect to a *twist-free* reference frame, computed by applying the discrete parallel transport method of Bergou et al. [BWR*08] to the rigid body frame at the hair's attachment point (see Fig. 3).

At the beginning of the simulation, we assign each guide $\mathbf{g}_j$ an orthonormal reference frame $\mathbf{F}_j = [\mathbf{t}_j, \mathbf{n}_j, \mathbf{b}_j]$, with $\mathbf{t}$ parallel to the first edge of the guide. We then parallel-transport this frame along the guide and encode the offset vectors for all tracking points as $\hat{\mathbf{r}}_{j,k} = (\theta_{j,k}, d_{j,k})$, where $\theta_{j,k}$ is the angle between $\mathbf{r}_{j,k}$ and $\mathbf{n}_j$, and $d_{j,k}$ is the distance from the centerline, i.e., $d_{j,k} = ||\mathbf{r}_{j,k}||$. For each frame of the animation, we update the reference frames for all guides using the transformation matrix of the mesh primitive that they attach to. The updated reference frame is propagated along the guide hair curve, and the offset vectors are reconstructed from the local coordinates using the updated frames.

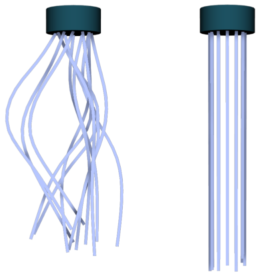

Figure 2: *Artifacts resulting from using static tracking points on a straight shrinking guide (*left*). Allowing tracking points to dynamically slide along the guide keeps the hair at rest (*right*).*

Dynamic Constraint Targets The center of mass constraints are designed to allow for bounded deviation of detail hairs from their guides. However, we found that, when using static tracking points at fixed locations along the guides, detail hairs are forced to stretch and compress, depending on the deformation of their guide (see Fig. 2). While stretching is detrimental to realism, compressions additionally give rise to ill-conditioned linear systems, deteriorating or even preventing convergence. To avoid these problems, we further relax the constraints and allow tracking points to dynamically *slide* along their guide hairs. In order to enable sliding with sufficient continuity, we adopt a spline-based representation for the guide hairs using piecewise cubic Catmull-Rom splines $\mathbf{g}_j(t)$ that are arc-length parameterized with respect to their (straight) rest configuration. We then endow each tracking point $\mathbf{p}_{j,k}$ with a degree of freedom $t_{j,k}$ corresponding to its parametric coordinate along the guide, i.e., $\mathbf{p}_{j,k} = \mathbf{g}_j(t_{j,k})$. By exposing these coordinates as variables to the optimizer, tracking points can slide along their guide in order to provide sufficient flexibility for the detail hairs to deform.

As tracking points $\mathbf{p}_{j,k}$ slide along their guides, the corresponding offset vectors $\mathbf{r}_{j,k}$ have to be updated accordingly. Rather than modeling both offset vectors and tracking points as functions of the sliding parameter $t_{j,k}$, we create an equivalent target spline curve $\hat{\mathbf{c}}_{j,k}(t)$ that directly describes admissible positions for the center of mass $\mathbf{c}_{j,k}$ (Fig. 3, right). To this end, we parallel-transport the initial offset vector $\hat{\mathbf{r}}_{j,k}$ from the base of the hair along the deformed guide hair and interpolate the resulting world-space positions to obtain $\hat{\mathbf{c}}_{j,k}(t)$. In this way, we define a target spline curve for each cluster and each frame of the input animation. It is important to note that only the sliding parameters are variables in the optimization: although changing from frame to frame, the geometry of the target curves is fixed for any given simulation step, since it is directly defined by the shape of the prescribed guide hair.

Constraint Formulation Putting everything together, we formulate for each cluster three tracking constraints, one for each spatial

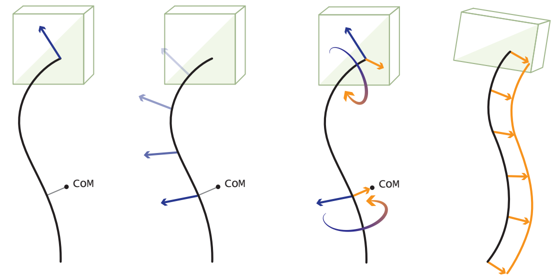

Figure 3: *Registration and propagation of constraints. A direction orthogonal to the guide base (blue, left) is parallel transported to the guide point closest to the center of mass (left, center). The rotation around the guide and the scaling that point the direction to the center of mass are applied to the base orientation (center, right). The new orientation (orange) is registered for this constraint. During simulation (right), the registered direction is transformed using deformation gradients and parallel transported to define the locus of this constraint.*

dimensions, as

$$\mathbf{C}_{j,k} = \mathbf{c}_{j,k}(t_{j,k}) - \hat{\mathbf{c}}_{j,k}(t_{j,k}) = \frac{1}{M_{j,k}} \sum_{(i,l)\in\mathcal{C}_{j,k}} m_{l,i}\mathbf{d}_{l,i} - \hat{\mathbf{c}}_{j,k}(t_{j,k}) \;, \tag{3}$$

whose free variables are the nodal positions $\mathbf{d}_{l,i}$ of all detail hair segments belonging to cluster $\mathcal{C}_{j,k}$, and one parametric coordinate $t_{j,k}$ describing the corresponding position along the target curve.

3.3. Spread Control

Each tracking constraint involves multiple nodes from a number of detail hairs, but only eliminates three degrees of freedom. We therefore have substantially more variables than constraints and, consequently, a high-dimensional null-space. In the language of statistical analysis, our tracking constraints are based on the first moment of the particle distribution in a given cluster but leave higher-order moments unconstrained. In particular, any symmetry-preserving spreading of vertices will change the *variance*, i.e., the second moment without affecting the tracking constraint. This situation is not problematic by itself, since our goal is to constrain only certain aspects of the motion while leaving much of it free to evolve according to physics. But although the momentum term in (5) regularizes motion in the null-space to some extent, a spread-out configuration might deviate from unconstrained physics just as much as a configuration with lower variance.

In order to regularize this null-space in an art-directable way, we introduce an additional energy term $P_{j,k} = k_v(V_{j,k} - \bar{V}_{j,k})^2$, where

$$V_{j,k} = \sum_{(i,l)\in\mathcal{C}_{j,k}} ||\mathbf{c}_{j,k} - \mathbf{d}_{l,i}||^2 \tag{4}$$

is the variance within a given cluster $\mathcal{C}_{j,k}$ in the current configuration and $\bar{V}_{j,k}$ a corresponding target value. By default, we set $\bar{V}_{j,k}$ to the rest state variance, but as shown in Fig. 4, both the target variance and the corresponding penalty coefficient k_v can be set by an artist in order to achieve various degrees of hair spreading.

3.4. Implementation

Hair Simulation We implement our tracking solver on top of a conventional hair simulation model based on mass-spring chains that resist stretching and bending, but not twisting [RCT91]. To prevent intersections between hair, we follow the framework of Bridson et al. [BFA02] and use a combination of instantaneous and continuous collision detection. After each time step, we detect collisions and filter the end-of-time-step velocity by applying correcting impulses to colliding edge pairs.

Following Martin et al. [MTGG11], we use a variational formulation of implicit Euler for time stepping and minimize the functional

$$H(\mathbf{x}_{n+1}) = \frac{1}{2}\Delta\mathbf{v}_{n+1}^T\mathbf{M}\Delta\mathbf{v}_{n+1} + E(\mathbf{x}_{n+1}) + P(\mathbf{x}_{n+1}) \;, \tag{5}$$

where $\Delta\mathbf{v}_{n+1} = 1/dt(\mathbf{x}_{n+1} - \mathbf{x}_n) - \mathbf{v}_n$, E collects stretching and bending energies for all hairs as well as gravity, and P is the potential for variance penalization. The derivative of the above expression with respect to end-of-time step positions $\mathbf{x}_{n+1}$ yields the well-known update rules for the backward Euler method. The advantage of the variational formulation is that the potential H can be used as objective function in the optimization problem as described next.

Optimization We cast simulation tracking as a constrained optimization problem whose Lagrangian

$$\mathcal{L}(\mathbf{X}_{n+1}, \lambda_{n+1}) = H(\mathbf{X}_{n+1}) + \lambda_{n+1} \cdot \mathbf{C}(\mathbf{X}_{n+1}) \tag{6}$$

is composed of an objective function H modeling the discrete equations of motion (5) and a set of tracking constraints $\mathbf{C}$, scaled by Lagrange multipliers λ_{n+1}. Note that we use the notation $\mathbf{X}$ to denote the concatenation of $\mathbf{x}$ and $\mathbf{t}$, the latter being the vector of sliding parameters defined in our dynamic constraint targets, which is also solved for. The first order optimality conditions require that the derivative of (6) vanish with respect to all parameters,

$$\begin{bmatrix} \nabla_{\mathbf{X}}\mathcal{L} \\ \nabla_{\lambda}\mathcal{L} \end{bmatrix} = \begin{bmatrix} \nabla_{\mathbf{X}}H(\mathbf{X}_{n+1}) + \nabla_{\mathbf{X}}\mathbf{C}(\mathbf{X}_{n+1})^T\lambda_{n+1} \\ \mathbf{C}(\mathbf{X}_{n+1}) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} . \tag{7}$$

We solve this set of nonlinear equations using sequential quadratic programming. Since our formulation only involves equality constraints, doing so amounts to solving saddle point problems of the form

$$\begin{bmatrix} \mathbf{H} & \nabla\mathbf{C}^T \\ \nabla\mathbf{C} & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{X} \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} -\nabla_{\mathbf{X}}\mathcal{L} \\ -\nabla_{\lambda}\mathcal{L} \end{bmatrix} , \tag{8}$$

where $\mathbf{H}$ is an approximate Hessian that includes second derivatives of all terms in (5) except for the variance penalty. Since the latter couples many degrees of freedom, including its Hessian would lead to a drastically increased number of non-zero entries in $\mathbf{H}$ and thus significantly deteriorate performance. Fortunately, since the forces resulting from the variance penalty are not stiff, we can neglect its second derivative without adversely affecting convergence. We solve the resulting system using Pardiso and, using a standard merit function, perform line search on the solution to determine a feasible step size as described in [NW06].

Figure 4: *Our variance penalty term allows for controlled hair spreading using different target variance values at the tip of the hair. From left to right:* v_0, $2.5v_0$, $3.5v_0$, $5v_0$, $7.5v_0$, $10v_0$, *where* v_0 *is the variance in the rest state. Note that the penalty is keyframed in and out by the artist as the hair reaches its apex.*

4. Results

In order to analyze the behavior of our method we performed a set of experiments that emphasize different aspects and use cases. We report our findings in the following and refer to the accompanying video for the corresponding animations.

Goaling The artist's intent often requires hair to hit a specific pose at a given time, but tweaking material parameters or force field coefficients to obtain the desired motion can be a laborious and frustrating process. Fig. 5 shows an example in which a wisp of hair is attached to a scripted rigid body that is first accelerated forward, then suddenly stopped such as to propel the hair forward. An artist performs an initial simulation on a coarse set of guide hairs using best-guess parameters, but the hair fails to hit the spinning wheel. She then edits the motion of one guide hair as desired using conventional keyframing to create the desired effect. Taking this edited motion as input, our method creates a higher-resolution simulation that produces the desired behavior while exhibiting natural motion and physical detail.

Splitting and Merging When hair collides with the body or other objects, it is often important to control not only its motion but also how it splits and merges in the process. Fig. 6 shows an example aimed at demonstrating these effects on a wisp of hair that is pulled over a spherical object. The uncontrolled simulation exhibits the expected even spreading, but using our method, the artist can create a variety of different splitting and merging behaviors by controlling the motion of several guide hairs. Again, the resulting tracked simulation adheres to the artistic intent in all cases, introducing compelling detail motion due to collisions.

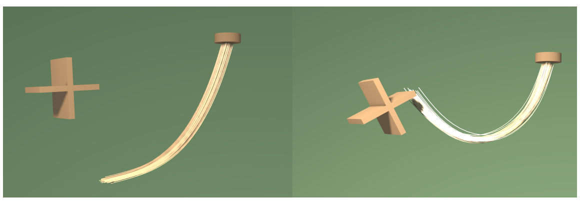

Figure 5: *Our method can be used to constrain this wisp of hair to hit the spinning wheel target (right) while an unconstrained simulation does not come close to reaching it (left).*

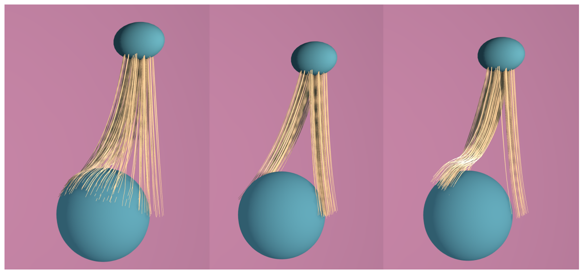

Figure 6: *While unconstrained hair splits evenly when colliding with this sphere (left), the splitting can be stylized by constraining the simulation to follow two (center) or three (right) guides.*

Spreading In order to analyze the ability of our variance penalty to control the spreading of hair, we create a sequence of wisp animations similar to the example shown in Fig. 5, but this time using different values for the target variance. We additionally keyframe the penalty coefficient to gradually increase to a maximum value when the hair reaches its apex, and decrease back to zero afterwards. As can be seen in Fig. 4 and the accompanying video, we achieve wildly different spreading behavior while retaining naturally-looking motion.

Stylization Rather than passively following the motion of a character, a talented artist can use our method to breathe life into hair and create highly stylized motion. Fig. 7 shows such an example in which a wisp of hair is attached to a rigid block resting on a rail. Without using simulation as input, the artist created a keyframe animation of a single guide hair that pulls itself off the rail by building up the necessary momentum. Using our method, the expressive nature of the input motion automatically translates into the tracked simulation, which exhibits the intended anticipation and follow-through enhanced with physically-simulated detail.

Celia Fig. 1 shows a more complex example in which Celia—a beautifully-stylized character—suddenly moves her head, causing her ponytail to accelerate upward. The initial simulation fails to hit the goblet, conspicuously placed at the edge of the table. However, after editing a single guide hair and tracking the motion using our solver, the hair knocks over the goblet as intended.

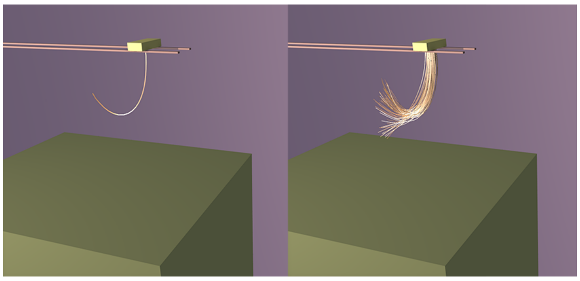

Figure 7: *This highly-stylized guide motion (left) can be tracked using our method to give life to a simulated wisp of hair (right).*

Performance The scenes used to create our results contain between 1360 hair vertices for the highly stylized guide and 7811 for Celia. They are simulated with timesteps ranging from $0.008s$ to $0.04s$. We have simulated between 56 frames (for the goaling example) and 230 frames (for the highly stylized guide). To indicate the performance of our tracking solver, we provide timing information and other statistics for the Celia animation, which is our most complex example. Celia's hair consists of 301 hairs, each comprising roughly 26 vertices, making for a total of 23,433 degrees of freedom that are controlled by 24 tracking constraints. Using a time step of 0.04s, the free simulation took about 11.3s in total or 0.045s per frame for 6s of output animation, whereas our tracking solver required 20.6s in total or 0.08s per frame, requiring about 20 iterations on average to solve the optimization problem. To break down computation times, our tracking solver spends roughly 40% of its time on computing and assembling the linear system, while another 30% is spent on its numerical solution. Between 10% and 20% of the time are spent on collision detection.

5. Conclusions

HairControl is a novel method to art-direct hair simulations, using a tracking solver in the lineage of TRACKS ([BMWG07]). Having established that existing constrained simulation methods either rely on consistent geometry for the simulated medium (cloth, solids) or control particles upon which no specific structure is imposed (fluids), we have presented a constraint formulation designed specifically for hair simulation. The results of our prototype implementation indicate that our approach is indeed effective at guiding and stylizing the coarse level motion of hair. Even for clearly non-physical guide hair animations, our method still produces plausible detail hair motion. Our results show animations edited in the context of targeting and stylization, and show robustness when handling animated non-physical guides.

Limitations & Future Work While we have designed our hair tracking method to be general, the hair model used in our prototype implementation does not account for twisting. Extending our method to curly hair types, potentially with explicit control over twist, is an interesting direction for future work.

For simplicity, our current implementation uses an impulse-based approach that resolves collisions after each time step. While serving as a proof of concept, handling collisions outside the optimization can induce large deformations that slow down convergence for the subsequent time step. This effect is exacerbated for larger step sizes, higher resolution per hair, and larger number of hairs. Addressing this limitation by including collision response directly within the optimization is another interesting challenge for future research, and would make the production of higher-resolution results easier.

In our method, each simulated hair is assigned to exactly one guide and, consequently, when guides move apart, clumps of hair move apart. We believe this behavior to be reasonable since our guide curves are meant to describe the coarse motion of the simulated hair. However, our method does not allow simulated hair to fan out around the guides as visualized in Fig. 8. In the future, we want to experiment with soft hair-to-guide assignments. Finally, providing artists with a tool to control the silhouette of simulated hair would bring our pipeline closer to the intuitive types of interaction 2D artists are accustomed to.

Beyond the tracking solver itself, our experiences with directable hair simulation have indicated opportunities for future work on related problems. Our method, as in similar art-directed simulation works, relies on the quality of the provided guide hairs' animation. Artists can simulate sets of few guide hairs and use these physically-plausible guides to drive a realistic hair simulation, or manually edit guide animations. The absence of satisfying tools to manually author such animated 3D curves has been a source of frustration when creating the results presented in this paper. We believe that geometric tools for 3D curve animations would benefit users of our method and assist them in authoring plausible guide motions. One particular option would be to extend the method of Whited and colleagues [WNS*10] for 2D curve interpolation based on artistic principles to 3D. Moreover, future work could target more automated approaches based, e.g., on space-time optimization with sparse keyframes, to guarantee that the authored guide motions do not contradict physical principles.

6. Acknowledgements

The authors sincerely thank Toby Jones, Alex McAdams, Maryann Simmons and Brian Whited at Walt Disney Animation Studios for the inspiring and fruitful research discussions, and for their continued feedback throughout this work.

References

[BAC*06] Bertails F., Audoly B., Cani M.-P., Querleux B., Leroy F., Lévêque J.-L.: Super-helices for predicting the dynamics of natural hair. In *Proc. of ACM SIGGRAPH '06* (2006). 2

[BAV*10] Bergou M., Audoly B., Vouga E., Wardetzky M., Grinspun E.: Discrete viscous threads. *ACM Trans. Graph. (Proc. SIGGRAPH) 29*, 4 (2010). 2

[BDCDA11] Bertails-Descoubes F., Cadoux F., Daviet G., Acary V.: A nonsmooth newton solver for capturing exact coulomb friction in fiber assemblies. *ACM Trans. Graph. 30*, 1 (Feb. 2011), 6:1–6:14. 2

[BdSP09] Barbič J., da Silva M., Popović J.: Deformable object animation using reduced optimal control. *ACM Trans. Graph. 28*, 3 (July 2009), 53:1–53:9. 3

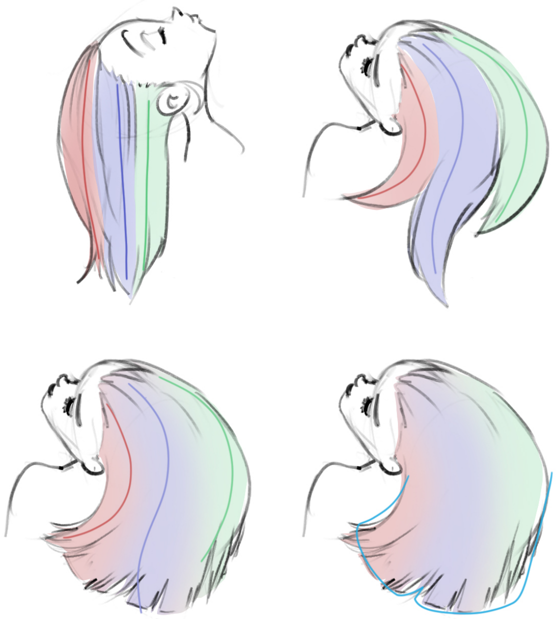

Figure 8: *The discrete hair-guide association performed in the rest state (top left) results in separate clumps of simulated hair (top right). Soft assignments could be useful to let the hair fan out (bottom left), while silhouette inputs (bottom right) would be a more intuitive shape depiction for such cases.*

[BFA02] BRIDSON R., FEDKIW R., ANDERSON J.: Robust treatment of collisions, contact and friction for cloth animation. *ACM Trans. Graph. 21*, 3 (July 2002), 594–603. 2, 5

[BKCN03] BERTAILS F., KIM T.-Y., CANI M.-P., NEUMANN U.: Adaptive wisp tree: A multiresolution control structure for simulating dynamic clustering in hair motion. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2003), SCA '03, Eurographics Association, pp. 207–213. 3

[BMWG07] BERGOU M., MATHUR S., WARDETZKY M., GRINSPUN E.: Tracks: Toward directable thin shells. *ACM Trans. Graph. 26*, 3 (July 2007). 2, 3, 4, 7

[BP08] BARBIČ J., POPOVIĆ J.: Real-time control of physically based simulations using gentle forces. *ACM Trans. Graph. 27*, 5 (Dec. 2008), 163:1–163:10. 3

[BWR*08] BERGOU M., WARDETZKY M., ROBINSON S., AUDOLY B., GRINSPUN E.: Discrete elastic rods. *ACM Trans. Graph. (Proc. SIGGRAPH) 27*, 3 (2008). 2, 4

[CCK05] CHOE B., CHOI M. G., KO H.-S.: Simulating complex hair with robust collision handling. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2005), SCA '05, ACM, pp. 153–160. 2, 3

[CJY02] CHANG J. T., JIN J., YU Y.: A practical model for hair mutual interactions. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2002), SCA '02, ACM, pp. 73–80. 2

[CMT*12] COROS S., MARTIN S., THOMASZEWSKI B., SCHUMACHER C., SUMNER R., GROSS M.: Deformable objects alive! *ACM Trans. Graph. 31*, 4 (July 2012), 69:1–69:9. 3

[CZZ14] CHAI M., ZHENG C., ZHOU K.: A reduced model for interactive hairs. *ACM Trans. Graph. 33*, 4 (July 2014), 124:1–124:11. 2

[CZZ16] CHAI M., ZHENG C., ZHOU K.: Adaptive skinning for interactive hair-solid simulation. *IEEE Transactions on Visualization and Computer Graphics PP*, 99 (2016), 1–1. 2

[DBDB11] DAVIET G., BERTAILS-DESCOUBES F., BOISSIEUX L.: A hybrid iterative solver for robustly capturing coulomb friction in hair dynamics. *ACM Trans. Graph. 30*, 6 (Dec. 2011), 139:1–139:12. 2

[HMT01] HADAP S., MAGNENAT-THALMANN N.: Modeling Dynamic Hair as a Continuum. *Computer Graphics Forum* (2001). 3

[HSvTP12] HILDEBRANDT K., SCHULZ C., VON TYCOWICZ C., POLTHIER K.: Interactive spacetime control of deformable objects. *ACM Trans. Graph. 31*, 4 (July 2012), 71:1–71:8. 3

[IMP*13] IBEN H., MEYER M., PETROVIC L., SOARES O., ANDERSON J., WITKIN A.: Artistic simulation of curly hair. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2013), SCA '13, ACM, pp. 63–71. 2

[KKA05] KONDO R., KANAI T., ANJYO K.-I.: Directable animation of elastic objects. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2005), SCA '05, ACM, pp. 127–134. 3

[KTS*14] KAUFMAN D. M., TAMSTORF R., SMITH B., AUBRY J.-M., GRINSPUN E.: Adaptive nonlinearity for collisions in complex rod assemblies. *ACM Trans. Graph. 33*, 4 (July 2014), 123:1–123:12. 2, 3

[LB15] LI Y., BARBIČ J.: Stable anisotropic materials. *IEEE Trans. on Visualization and Computer Graphics 21*, 10 (2015), 1129–1137. 3

[LHdG*14] LI S., HUANG J., DE GOES F., JIN X., BAO H., DESBRUN M.: Space-time editing of elastic motion through material optimization and reduction. *ACM Trans. Graph. 33*, 4 (July 2014), 108:1–108:10. 3

[MSW*09] MCADAMS A., SELLE A., WARD K., SIFAKIS E., TERAN J.: Detail preserving continuum simulation of straight hair. *ACM Trans. Graph. 28*, 3 (July 2009), 62:1–62:6. 3

[MTGG11] MARTIN S., THOMASZEWSKI B., GRINSPUN E., GROSS M.: Example-based elastic materials. *ACM Trans. Graph. 30*, 4 (July 2011), 72:1–72:8. 3, 5

[MTPS04] MCNAMARA A., TREUILLE A., POPOVIĆ Z., STAM J.: Fluid control using the adjoint method. *ACM Trans. Graph. 23*, 3 (Aug. 2004), 449–456. 3

[NW06] NOCEDAL J., WRIGHT S. J.: *Numerical Optimization*, 2nd ed. Springer, New York, 2006. 5

[Pai02] PAI D. K.: Strands: Interactive simulation of thin solids using cosserat models. *Computer Graphics Forum 21*, 3 (2002), 347–352. 2

[PHA05] PETROVIC L., HENNE M., ANDERSON J.: *Volumetric methods for simulation and rendering of hair*. Tech. Rep. 06-08, Pixar, 2005. 1

[RCT91] ROSENBLUM R. E., CARLSON W. E., TRIPP E.: Simulating the structure and dynamics of human hair: modelling, rendering and animation. *J. Vis. and Comput. Anim.*, 2 (1991), 141–148. 2, 5

[SLF08] SELLE A., LENTINE M., FEDKIW R.: A mass spring model for hair simulation. *ACM Trans. Graph. 27*, 3 (Aug. 2008), 64:1–64:11. 2, 3

[ST07] SPILLMANN J., TESCHNER M.: Corde: Cosserat rod elements for the dynamic simulation of one-dimensional elastic objects. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2007), SCA '07. 2

[SvTSH14] SCHULZ C., VON TYCOWICZ C., SEIDEL H.-P., HILDEBRANDT K.: Animating deformable objects using sparse spacetime constraints. *ACM Trans. Graph. 33*, 4 (July 2014), 109:1–109:10. 3

[TKRP06] THUEREY N., KEISER R., RUEDE U., PAULY M.: Detail-Preserving Fluid Control. *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Jun 2006), 7–12. 2

[TMPS03] TREUILLE A., MCNAMARA A., POPOVIĆ Z., STAM J.: Keyframe control of smoke simulations. *ACM Trans. Graph. 22*, 3 (July 2003), 716–723. 3

[TTL12] TAN J., TURK G., LIU C. K.: Soft body locomotion. *ACM Trans. Graph. 31*, 4 (July 2012), 26:1–26:11. 3

[WBK*07] WARD K., BERTAILS F., KIM T.-Y., MARSCHNER S. R., CANI M.-P., LIN M. C.: A survey on hair modeling: Styling, simulation, and rendering. *IEEE Transactions on Visualization and Computer Graphics 13*, 2 (Mar. 2007), 213–234. 2

[WK88] WITKIN A., KASS M.: Spacetime constraints. *SIGGRAPH Comput. Graph. 22*, 4 (June 1988), 159–168. 3

[WL03] WARD K., LIN M. C.: Adaptive grouping and subdivision for simulating hair dynamics. In *Proceedings of the 11th Pacific Conference on Computer Graphics and Applications* (Washington, DC, USA, 2003), PG '03, IEEE Computer Society, pp. 234–. 3

[WMT06] WOJTAN C., MUCHA P. J., TURK G.: Keyframe control of complex particle systems using the adjoint method. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2006), SCA '06, Eurographics Association, pp. 15–23. 3

[WNS*10] WHITED B., NORIS G., SIMMONS M., SUMNER R., GROSS M., ROSSIGNAC J.: Betweenit: An interactive tool for tight inbetweening. *Comput. Graphics Forum (Proc. Eurographics) 29*, 2 (2010), 605–614. 7

[XKG*16] XING J., KAZI R. H., GROSSMAN T., WEI L.-Y., STAM J., FITZMAURICE G.: Energy-brushes: Interactive tools for illustrating stylized elemental dynamics. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (New York, NY, USA, 2016), UIST '16, ACM, pp. 755–766. 3

[XSZB15] XU H., SIN F., ZHU Y., BARBIČ J.: Nonlinear material design using principal stretches. *ACM Trans. Graph. 34*, 4 (July 2015), 75:1–75:11. 3

[YL16] YIJING LI HONGYI XU J. B.: Enriching triangle mesh animations with physically based simulation. *IEEE Transactions on Visualization and Computer Graphics* (2016). 3