Review 3.3 - Efficient Simulation of Inextensible Cloth

Shivam Thukral

October 14, 2019

1 Description

Clothes do not stretch under their own weight. For efficiency, many cloth simulations allow 10% or more strain. This paper presents a new cloth algorithm, which excels the current methods by treating the cloth model as inextensible material with a permissible strain of 1% or less. The solver builds on a framework of Constrained Lagrangian Mechanics (CLM) in which warp and weft are prohibited from stretching by enforcing constraint equations.

Contributions:

- Summary of relevant literature.
- Describe discrete cloth model
- Propose a novel CLM formulation that is implicit on constraint gradients.
- Motivates a fast projection method for enforcing inextensibility and describes how this fast projection method can be used for inextensibility in cloths. This approach uses gradient at the end of the time step which is geometrically robust and equivalent to a minimization problem.
- This approach can be easily incorporated within existing bending, damping and collision models to yield accelerated performance.

Coth Model: is an interleaving structure of yarn which resists stretch along the warp and weft directions. Treats cloth as a quad-dominant mesh with edges along the warp and weft directions.Inextensibility is introduced by adding edge preserving constrains along the edges.
Inextensibility Constraint: A warp and weft-aligned quad-edge (X_a, X_b) maintains an undeformed length, L, be enforcing:

$$C(X_a, X_b) = ||X_b - X_a||^2 - L^2 = 0$$

This can be rewritten as, $C(X_a, X_b) = \frac{||X_b - X_a||^2}{L} - \frac{L}{2}$.

Taking the constraint gradient w.r.t X_b :

$$\nabla_{X_b} C(X_a, X_b) = \frac{(X_b - X_a)}{L}$$

A constraint is modelled as an implicit scalar function $C(x)$, whose zero value is when the constraint is met. So, we have 1 constraint per warp and weft.

$C(x) = [C_0(x), C_1(x), ..., C_{m-1}(x)]$ where, m = number of constraints or edges

To introduce constraints, we can add the constraint function to the Lagrangian equation as follows:

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

Where,
$x(t)$: the time-varying 3n-vector of vertex position,
$V(t)$: time derivative,
M : $3n * 3n$ mass matrix,
$U(x)$: the stored energy (e.g. bending, shear, and gravity),
$C(x)$: the m-vector of constraints (i-th edge),
λ : the m-vector of Lagrange multipliers,

And, the corresponding Euler Lagrangian equations are given by,

$$M\dot{v} = -\nabla U(x) - \nabla C(x)^T \lambda,$$

$$C(x) = 0,$$

For simulation we discretize last two equations using the following methods:

Explicit Constraint Direction (ECD) :
SHAKE and RATTLE which extends Verlet scheme that considers the force direction, $-\nabla C(x)^T$, evaluated at the beginning of the time step. Here we compute only the forces magnitude to get the particle back on

the constraint manifold. However, in large time steps or rapidly changing constraints this approach is known to fail.

Implicit Constraint Direction (ICD):
Evaluating the constraint direction, $-\nabla C(x)^T$, at the end of the time step. Requires the position at the end of the time step to allow computing forces direction and magnitude. Solving for an ICD step is expensive, because there are many unknowns and each Newton iteration requires the solution of an indefinite linear systems.

Step and Project(SAP):
It's a different viewpoint of the same ICD technique. This involves two steps:

- Perform unconstrained step that takes away from constraint manifold.
- Project the particle to the closest point on the constraint manifold.

Fast Projection method[2]:
Instead of finding the exact point on x, we settle for a point, that is close but not necessarily close to the unconstrained position. It is basically a sequence of short steps, starting at X_o(unconstrained position). At each step we try to get close to the constraint manifold while moving as little as possible.
For a single particle this can written as minimizing the objective function:

$$W(x_{j+1}, \lambda) = \frac{1}{2} \|X_{j+1} - X_j\|^2 + C(X_{j+1})^T \lambda$$

The pseudo code for this method is as follows:
$input = X_o$
$whileC(X_j) \neq 0$
{
$solve : [\nabla C(X_j) \nabla C(X_j)^T]\lambda = -C(X_j)$
$update : X_{j+1} = X_j + \nabla C(X_j)^T \lambda$
}

Fast Projection in Cloth Simulation:

- Compute the unconstrained position, X_o, using unconstrained time integration method and treat forces in the usual manner.
- Applying inextensibility using Fast Projection method in which we solve for all constraints simultaneously.
- Apply collision response using the velocity filter [Bridson et al. 02]

2 Analysis

Strengths:

- This approach has less computation time and many benefits can be obtained from the inextensibility property of the cloth.
  - Cloth generates more folds and wrinkles therefore looks more realistic.
  - [Bergou et al. 06] and [Garg et al. 07] use this property to accelerate bending computation.
- This approach is simple and modular- making it very practical and maintainable implementation.
- This approach doesn't cause any artifacts since:
  - To avoid glaring collision artifacts authors implement collision handling in the end.
  - Errors are not accumulated during simulation as they are being filtered out in the next iteration.
- We can add other constraints to this approach (like sharp creases on pants) using the Constraint Lagrangian Mechanics framework.
- Using this approach damping and strain are independent parameters unlike penalty spring where increasing stiffness will result in increasing damping.
- The main strength of CLM framework is that constraints are enforced exactly without making the underlying system stiff.

Weakness:

- There is no way to apply both ideal inextensibility and ideal collision handling together.
- To enforce this, the authors tried to combine both in a single pass but this did not give them satisfactory results.
- We can oscillate between contact and velocity filters but this approach may not converge.

- It is not clear whether this approach is more efficient than simply taking smaller time steps, as compared to performing few iterations of each filter allowing to taking large time steps

- In order to use this approach we need a quad-dominant mesh. By using a triangulated mesh the cloth would look rigid.

Impact of the paper:

According to Semantic Scholar this paper has 196 citations. In this, 23 are highly influenced papers and 63 of them cite methods, with an average citation of 13 per year from 2017 through 2019.

3 Clarity of Exposition and Reproducibility

The author clearly sets a background for this paper. In the related work section, the author highlights how to apply the constraints on edges using the existing methods:

- Stiff Springs [Terzopoulus et al. 87], [House et al. 94], [Baraff and Witkin 98], [Choi and Ko 02] : Here we set springs between the particles. The extensibility is achieved by making the spring stiffer and stiffer. Such methods are slow for simulations as springs become stiffer.

- Direct Manipulation [Provot 95], [Desbrun et al. 99] [Bridson et al. 02], [Muller et al. 06] : Given a strained edge directly move the vertices to reduce the strain. Here multiple iterations might be required.

- Constrained Dynamics [House et al. 96] [Hong et al. 05] [Tsiknis and Bridson 06] : Authors build on this to explain Explicit Constraint Direction, Implicit Constraint Direction, Step and Project and finally a novel method: Fast Projection Method.

For a person who doesn't have a strong background in computer graphics or in the fields like constrained dynamics this paper can be a bit difficult to understand. I had to go through Adrian R. Goldenthal's PhD thesis[1] ("Implicit Treatment of Constraints for Cloth Simulation") inorder to understand concepts like: proofs of robustness to snake fail cases and details about Newtons method. Personally, chapter 2: Physical Simulation of Cloth was very helpful in making me understand concepts discussed in the paper. However, since all pseudo code is provided we can implement this approach.

References

[1] Adrian R. Goldenthal. *Implicit Treatment of Constraints for Cloth Simulation.* PhD thesis, Hebrew University, 2010.

[2] Rony Goldenthal, David Harmon, Raanan Fattal, Michel Bercovier, and Eitan Grinspun. Efficient simulation of inextensible cloth. *ACM Trans. Graph.*, 26(3), July 2007.