Eurographics/ ACM SIGGRAPH Symposium on Computer Animation (2008)
M. Gross and D. James (Editors)

Globally coupled collision handling using volume preserving impulses

Eftychios Sifakis[†]
University of California, Los Angeles
Disney Animation Studios

Sebastian Marino[†]
Makani Power Inc.

Joseph Teran[†]
University of California, Los Angeles
Disney Animation Studos

Abstract

We present a novel algorithm for collision processing on triangulated meshes. Our method robustly maintains a collision free state on complex geometries while resorting to collision resolution at time intervals often comparable to the frame rate. Our approach is motivated by the behavior of a thin layer of fluid inserted in the empty space between nearly-colliding parts of the simulated surface, acting as a cushioning mechanism. Point-triangle or edge-edge pairs on a collision course are naturally resolved by the incompressible response of this fluid buffer. This response is formulated into a globally coupled nonlinear system which we solve using Newton iteration and symmetric, positive definite solvers. The globally coupled treatment of collisions allows us to resolve up to two orders of magnitude more collisions than traditional greedy algorithms (e.g. Gauss-Seidel collision response) and take substantially larger time steps without compromising the visual quality of the simulation.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Physically based modeling I.3.7 [Computer Graphics]: Animation

1. Introduction

The simulation of clothing and other cloth materials is ubiquitous in the feature animation and visual effects industry. The most confounding cloth behaviors for digital costuming are external and self contact. Real and computer generated actors typically wear layer upon layer of clothing resulting in the frequent and difficult nature of this problem. We introduce a novel technique for reducing the simulation cost of elaborate layered cloth in complex collision scenarios.

Cloth is efficiently represented with a Lagrangian mesh (e.g. masses and springs on a triangulated mesh). Such approaches were pioneered in graphics by [TPBF87, TF88] and are used today almost without exception. The Lagrangian mesh model naturally allows for accurate computation of internal elastic response to deformation. However, collision and contact are not resolved as easily, since these phenomena may cause interactions between topologically distant regions in the Lagrangian mesh. In contrast to this, stationary grid Eulerian methods, typically used for computational fluid dynamics, have difficulty computing an elastic response but naturally resolve contact, collision and topological change. For example in a free surface incompressible flow simulation, fluid regions naturally collide and interact by simply enforcing the incompressibility of the fluid. In fact, some Lagrangian elastic solid/Eulerian incompressible fluid coupling algorithms like Peskin's immersed boundary method do not require any special collision handling at all [Pes02]. The incompressibility and structure of the fluid is enough to guarantee that collisions never happen. That is, if the velocity defined throughout the fluid and solid is always divergence free, there is no way any two particles (or other mesh facets) can collide because doing so would require a locally divergent velocity field. Of course, when simulating clothing for visual effects, it is not practical to consider the surrounding air as a fluid and we do not recommend this here. However, we do draw inspiration from the natural preclusion of collision phenomena in incompressible velocity fields.

Our approach to the cloth collision/contact problem is motivated by the aforementioned observations. We detect point-triangle and edge-edge pairs in a mesh that are on collision trajectory over a time step, and formulate a global system of equations for their collective response to this imminent collision. This globally coupled system models the response of an incompressible fluid trapped between the colliding surfaces which acts as a cushioning mechanism to prevent interpenetration of its lateral surfaces. In lieu of an Eulerian discretization, we use the tetrahedra defined by the vertices of a point-triangle or an edge-edge pair to define the incompressible response of this fluid buffer. We solve this nonlinear system using an iterative Newton method. In the course

[†] email: {sifakis|jteran}@math.ucla.edu,sebastian@makanipower.com

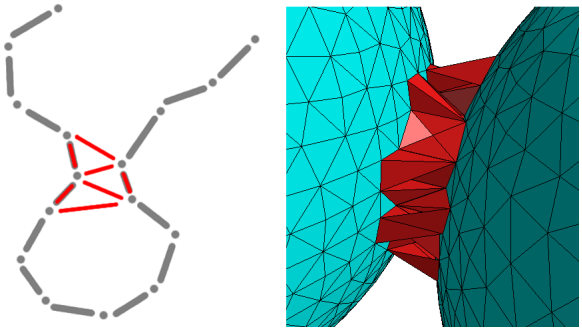

Figure 1: *The spheres depicted on the right are moving towards each other and are about to collide. Edge-edge and point-face pairs determined to collide during the collision time interval are used to define a number of (possibly overlapping) tetrahedra used in our response scheme, drawn in red. The leftmost image depicts the 2D analogue of this, depicting imminently colliding point-segment pairs in red.*

of these iterations we recompute the set of colliding point-triangle and edge-edge pairs as to release any such primitives from the incompressibility constraint if they are no longer in a collision course. In an analogous fashion, points that have been brought into a collision trajectory as a result of previous corrections will be included in the incompressibility constraint for subsequent iterations of the Newton method.

Looking past the fluids paradigm which provided the original inspiration, our method is supported by the following facts which are further discussed in later sections:

- In the limit of a small time step, our treatment is equivalent to an inelastic collision impulse normal to the collision surface.
- Collision response is formulated as a nonlinear constraint, which *guarantees* the resolution of all collisions when satisfied. Using an iterative Newton method allows for continuous improvement of the computed solution, taking into consideration the configuration created by previous corrections.
- The linear systems arising from our formulation are *symmetric and positive definite* enabling the use of efficient solvers
- Our method does not require any vector normalization, which could lead to robustness issues in degenerate configurations. In fact, forming the linear systems needed by our method requires absolutely no division.

Finally, our method integrates naturally into existing collision processing pipelines. In particular, we demonstrate how our method can be combined with the time integration scheme of [BFA02] as a self-contained replacement of a specific module in their algorithm (i.e. geometric collision resolution using inelastic impulses). Additionally, our algorithm is independent and does not affect their treatment of other behaviors (i.e. collisions with rigid bodies, friction,elasticity).

2. Previous Work

Early examples of cloth constitutive modeling include [Wei86, TPBF87, TF88, TC92, CYMTT92, OITN92, BHW94]. A good summary of cloth modeling is provided in [HB00]. Our method works with any temporal integration scheme including semi-implicit methods like that introduced in [BMF03] and implicit methods like those of [BW98, MDDB01, PF02, BA04, VMT05, OAW06]. Though not considered here, much investigation has gone into cloth constitutive behavior, including bending models by [CK02, BMF03, GHDS03, BWH*06, TWS06, VMT06, GHF*07]. Adaptive resolution approaches have also been investigated by e.g. [GKS02]. For collision detection, we use straightforward algorithms and extensions based on well known work but refer the interested reader to [GKJ*05, SGG*06] and the references therein. We also note the work on improving efficiency in low curvature regions [VMT94, VCMT95]. Though not in the context of cloth, [ISF07] investigated the enforcement of incompressibility on a tetrahedron mesh. Pressures were defined at nodes rather than tetrahedra to avoid locking. We found the tetrahedral meshes used in our method tended to be much less prone to locking than the volumetric meshes used in solid mechanics (due to a closer ratio of number of tetrahedra to number of nodes in our meshes).

3. Simulation methodology

Our time integration and collision handling loop is derived from the model proposed in [BFA02]. Time integration is interleaved with collision handling, which is invoked at intervals whose lengths are adjusted in accordance with the recent rate of success of previous collision handling attempts. Time evolution proceeds in the following loop:

1. Integrate positions $\mathbf{x}^n$ and velocities $\mathbf{v}^n$ for a time step Δt^n to obtain *candidate* positions $\mathbf{x}_*^{n+1}$ and velocities $\mathbf{v}_*^{n+1}$.
2. Compute the effective velocity $\mathbf{v}_*^{n+\frac{1}{2}} = (\mathbf{x}_*^{n+1} - \mathbf{x}^n)/\Delta t^n$.
3. Proccess $\mathbf{v}_*^{n+\frac{1}{2}}$ for repulsions/collisions to obtain $\mathbf{v}^{n+\frac{1}{2}}$.
   a. Adjust $\mathbf{v}$ for elastic/inelastic repulsions and friction
   b. Adjust $\mathbf{v}$ for inelastic response to geometric collisions
4. Compute the new positions $\mathbf{x}^{n+1} = \mathbf{x}^n + \Delta t \mathbf{v}^{n+\frac{1}{2}}$.
5. - If $\mathbf{x}^{n+1}$ is self-intersecting: Restore the collision free state $(\mathbf{x}^n, \mathbf{v}^n)$ and retry from step 1 using a smaller time step Δt^n. If Δt^n has reached a user-specified minimum value, group colliding pairs into rigid impact zones eliminating all collisions and return to step 1.
   - If $\mathbf{x}^{n+1}$ is intersection-free: Use $\mathbf{v}^{n+\frac{1}{2}}$ to update $\mathbf{v}^{n+1}$ and continue, optionally using a larger step Δt^{n+1} at the next iteration.

The actions taken in step 3 determine how the final velocities $\mathbf{v}^{n+1}$ will be updated in step 5. If collisions were detected

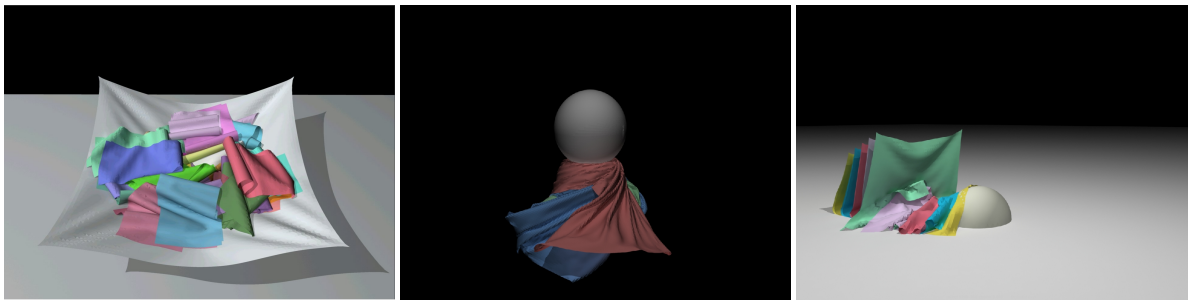

Figure 2: *Thirty cloth napkins are piled in a hammock (Left). Layered cloth trapped between rotating kinematic spheres (Middle). Kinematic sphere shooting through five layered drapes (Right).*

and processed, $\mathbf{v}^{n+1}$ is set to the value of $\mathbf{v}^{n+1/2}$ for all particles processed for repulsion or collision. Unaffected particles retain their value in $\mathbf{v}_*^{n+1}$. If no collisions were found in step 3 (although repulsions may still have been applied) we instead compute the velocity change $\Delta\mathbf{v} = \mathbf{v}^{n+1/2} - \mathbf{v}_*^{n+1/2}$ and update the final velocity as $\mathbf{v}^{n+1} = \mathbf{v}_*^{n+1} + \Delta\mathbf{v}$. Finally, collisions of the cloth surface with kinematic collision objects are typically handled in step 1 via penalty forces or a projection scheme [BFA02].

In the context of this time integration scheme, our algorithm deals precisely with step (3.b) above, replacing the Gauss-Seidel impulse scheme of [BFA02] with our globally coupled scheme which mimics an incompressible fluid response in the contact region. As a result, the treatment of elasticity (Step 1), collisions with kinematic bodies (Step 1), or friction (Step 3.a) remains unaffected. We followed the implementation choices of [BFA02] for all components of the time integration scheme other than our replacement for step (3.b). Finally, we observed that with our new globally coupled collision scheme the benefit of the rigid impact zone failsafe in step (5) was of no practical use; for all examples in this paper our algorithm never needed to resort to rigid impact zones. In more extreme settings than illustrated in this paper (e.g. performing collision processing just once after several frames) our algorithm would resort to this failsafe only in scenarios where the unmodified algorithm of [BFA02] was condemned to severe artifacts after the rigid impact zone step.

4. Collision response properties

Our algorithm targets step (3.b) in the time integration scheme of Section 3, treating the integrator used in step 1 as a black box. Since this step only affects the half-timestep velocity $\mathbf{v}_*^{n+1/2}$, its objective can be formalized as the determination of a velocity update $\Delta\mathbf{v}$ such that $\mathbf{v}^{n+1/2} = \mathbf{v}_*^{n+1/2} + \Delta\mathbf{v}$ and the final positions

$$\mathbf{x}^{n+1} = \mathbf{x}^n + \Delta t \mathbf{v}_*^{n+\frac{1}{2}} + \Delta t \Delta\mathbf{v} \qquad (1)$$

are intersection free. Additionally, $\Delta\mathbf{v}$ needs to adhere to the following physical restrictions:

- **Conservation of momentum.** The velocity update $\Delta\mathbf{v}$ is equivalent to the application of an impulse $\mathbf{j} = \mathbf{M}(\Delta\mathbf{v})$ to the particles affected by collision processing (where $\mathbf{M}$ is the mass matrix). This impulse needs to be conservative both *globally* (i.e. conservation of linear momentum implies $\sum j_i = 0$) and *locally*. In the context of our approach, we define a response to be locally conservative if the total momentum in each contact zone is conserved. Thus we require any exchange of momentum to be mediated by collision events, as opposed to a transfer of momentum between spatially remote parts of the cloth that are not connected by a chain of collision pairs.
- **Inelasticity.** Even in cases where the computed impulse $\mathbf{j}$ conserves momentum, it could nevertheless increase the kinetic energy in our system. In fact, a number of authors have argued that the collision response should ideally be *inelastic* [BFA02], thus minimizing the kinetic energy of the system after collision handling. Note that an elastic response may be used as part of a repulsion stage (Step 3.a in Section 3), which is a completely independent process than the inelastic response described in Step (3.b).
- **Consistency with normal collision response.** Similar to the approach of [BFA02] and others, our method processes collisions at the end of a time interval, rather than resolving collision events in a continuous, sequential fashion. However, when the interval between collision attempts is sufficiently small, the geometry of the cloth at either the beginning or the end of this time interval provides a good approximation of the geometry of the cloth at the exact time of collision. In this limit case, we require the collision impulses (and associated velocity adjustments $\Delta\mathbf{v}$) to be *normal* to the collision surface. This is necessary to ensure that, at the limit of small collision intervals, our method converges to the ideal response computed at the exact time of collision.

5. Volume preserving impulse response model

We illustrate a natural and convenient mechanism to determine impulses that satisfy the requirements outlined in Section 4 and allow us to produce a collision-free state $\mathbf{x}^{n+1}$. We begin by identifying the pairs of cloth primitives (i.e.

point-triangle or edge-edge pairs) that are determined to be on a collision course in the duration of a time step Δt^n, assuming initial positions $\mathbf{x}^n$ and constant (effective) velocities $\mathbf{v}_*^{n+1/2}$. We note that this should be an *exact* collision test, rather than a conservative approximation using, for example, bounding boxes of each potentially colliding primitive. This is a well-documented problem; we use the approach of [BFA02] which determines colliding pairs by solving a cubic equation.

Once the pairs of colliding primitives have been determined, every point-triangle or edge-edge pair is used to introduce a *collision tetrahedron* defined by the four vertices of the colliding pair. We orient this tetrahedron so that it has a positive signed volume in the collision-free $\mathbf{x}^n$ configuration. We note that these tetrahedra may be overlapping in the general case, and span the space between the colliding surfaces. We formulate our collision constraint as to emulate the response of an incompressible fluid inserted in the gap between the surfaces. As a result of such a response, the volumes of the collision tetrahedra (which span the same gap as the hypothetical fluid) would remain constant, i.e. $\mathbf{V}(\mathbf{x}^{n+1}) = \mathbf{V}_{goal}$, where $\mathbf{V} = (V_1, V_2, \ldots, V_M)$ denotes the volumes of all M collision tetrahedra, and $\mathbf{V}_{goal} = \mathbf{V}(\mathbf{x}^n)$. Using (1), we get

$$\mathbf{V}(\mathbf{x}^n + \Delta t \mathbf{v}_*^{n+\frac{1}{2}} + \Delta t \Delta \mathbf{v}) = \mathbf{V}_{goal}$$

$$\Rightarrow \quad \mathbf{V}(\mathbf{x}_*^{n+1} + \Delta t \mathbf{M}^{-1}\mathbf{j}) = \mathbf{V}_{goal} \tag{2}$$

where $\mathbf{x}_*^{n+1}$ denotes the candidate positions at the end of the timestep, resulting from an effective velocity $\mathbf{v}_*^{n+1/2}$. We note that the nonlinear system (2) contains as many equations as collision tetrahedra, yet the unknown impulse $\mathbf{j}$ has the dimensionality of the particles incident on collision elements. As mentioned in Section 4 the impulse $\mathbf{j}$ needs to be momentum conserving and inelastic.

A natural way to satisfy these constraints is to consider a *pressure* $\mathbf{p}$ at each collision tetrahedron working to restore the tetrahedron to its original volume. Subsequently, the impulse vector $\mathbf{j}$ is computed as the result of the action of this pressure during the collision event. The resulting impulses can be computed using the finite volume method as:

$$\mathbf{j} = \mathbf{F}\mathbf{p}, \quad \vec{F}_{ij} = -\frac{1}{3} A_i^{(j)} \hat{n}_i^{(j)} \tag{3}$$

where $\hat{n}_i^{(j)}$ is the outward pointing face normal opposite node i in tetrahedron j, and $A_i^{(j)}$ is the area of the same face. We should emphasize that all values $\vec{F}_{ij}$ can be robustly computed without the need for division or normalization, as the area weighted normal $A_i^{(j)}\hat{n}_i^{(j)}$ is simply one half times the cross product of 2 edge vectors of the respective tetrahedron face. Therefore, collision tetrahedra that have near-zero volume or arbitrarily short edges do not pose a problem in our formulation.

It is important to observe that the impulses defined in equation (3) satisfy the requirements set forth in Section 4. The finite volume formulation guarantees that the four impulses applied to the vertices of the collision tetrahedron conserve momentum (e.g. they sum to zero, preserving linear momentum), thus local conservation of momentum as defined in Section 4 is automatic. In addition, in the limit of a small time step Δt^n the collision tetrahedra are expected to be asymptotically flat. In this case, equation (3) indicates that the computed impulses would be aligned with the normals to the faces of the collision tetrahedron; all such faces will be approximately coplanar and aligned with the collision surface. The last requirement to be satisfied is that of inelasticity. In fact, this requirement will be used to determine the exact values of the elemental pressures $\mathbf{p}$. We express the incompressibility condition in terms of the elemental pressures as follows:

$$\begin{aligned} \mathbf{V}_{goal} &= \mathbf{V}(\mathbf{x}_*^{n+1} + \Delta t \mathbf{M}^{-1}\mathbf{F}\mathbf{p}) \\ &\approx \mathbf{V}(\mathbf{x}_*^{n+1}) + \Delta t \frac{\partial \mathbf{V}}{\partial \mathbf{x}} \mathbf{M}^{-1}\mathbf{F}\mathbf{p} \end{aligned}$$

$$\text{or} \quad \left.\frac{\partial \mathbf{V}}{\partial \mathbf{x}}\right|_{\mathbf{x}_*^{n+1}} \mathbf{M}^{-1}\,\mathbf{F}|_{\mathbf{x}_*^{n+1}}\,(\mathbf{p}\Delta t) \approx \mathbf{V}_{goal} - \mathbf{V}(\mathbf{x}_*^{n+1}) \tag{4}$$

where we explicitly stated the dependence of $\partial\mathbf{V}/\partial\mathbf{x}$ and $\mathbf{F}$ on the candidate position $\mathbf{x}_*^{n+1}$ at the end of the time step. The most important property of equation (4), however is that the matrix $\partial\mathbf{V}/\partial\mathbf{x}$ can be shown to be the transpose of $\mathbf{F}$, leading to the symmetric, positive semi-definite system $\mathbf{F}^T\mathbf{M}^{-1}\mathbf{F}(\mathbf{p}\Delta t) = \mathbf{V}_{goal} - \mathbf{V}(\mathbf{x}_*^{n+1})$, which can be solved efficiently with a Krylov subspace method. Finally, one can show that the pressure jump $\mathbf{p}$ that minimizes the post-collision kinetic energy

$$K = \frac{1}{2}(\mathbf{v}_*^{n+\frac{1}{2}} + \mathbf{M}^{-1}\mathbf{F}\mathbf{p})^T\mathbf{M}(\mathbf{v}_*^{n+\frac{1}{2}} + \mathbf{M}^{-1}\mathbf{F}\mathbf{p})$$

satisfies the equation

$$\mathbf{F}^T\mathbf{M}^{-1}\mathbf{F}\mathbf{p} = -\mathbf{F}^T\mathbf{v}_*^{n+\frac{1}{2}} = \frac{1}{\Delta t}\frac{\partial \mathbf{V}}{\partial \mathbf{x}}(\mathbf{x}^n - \mathbf{x}_*^{n+1})$$

which is the same equation as (4) to first order, as revealed by a Taylor expansion of the right-hand side of equation (4). Thus, solving the nonlinear incompressibility constraint using pressure-based impulses yields a conservative, inelastic collision response, with global coupling of the collision constraint. This is of course commensurate with the idealization of volume preservation (as the pressure in an incompressible fluid can be shown to minimize kinetic energy).

Our algorithm is summarized in Table 1. The outer loop, repeated N times, exactly recomputes the pairs on a collision course using the most updated estimate of $\mathbf{v}_*^{n+1/2}$ by solving the cubic equation described in [BFA02]. As a result, previously colliding pairs that are no longer on a collision course are released from the incompressibility constraint, while pairs that are now colliding as a result of previous corrections are included in the system. The inner loop performs

```
for i=1 to N
    Compute_Collision_Pairs()
    for j=1 to M
        Compute F(x*^{n+1}) and V(x*^{n+1})
        Solve (4) to obtain p and Δv = M^{-1}Fp
        Update v*^{n+1/2} ← v*^{n+1/2} + γΔv
        Update x*^{n+1} ← x^n + Δt v*^{n+1/2}
```

Table 1: *Pseudocode for our collision response scheme*

M Newton iterations, keeping the set of active collision pairs fixed, but updating the values of $\mathbf{F}(\mathbf{x}_*^{n+1})$ and $\mathbf{V}(\mathbf{x}_*^{n+1})$ using the most updated estimate of the post-collision positions $\mathbf{x}_*^{n+1}$. The equation (4) is solved using a MinRes symmetric definite solver. The computed velocity correction $\Delta\mathbf{v}$ is scaled by a relaxation coefficient γ and added to the value of $\mathbf{v}_*^{n+1/2}$. A value of $\gamma = 1$ is equivalent to the standard Newton method, while a small positive value would lead to a behavior similar to that of steepest descent. Due to the highly nonlinear nature of the constraint (owing to its dependence on both $\mathbf{x}_*^{n+1}$ *and* the set of currently active pairs) we found that a value of $\gamma = 0.3$ leads to faster convergence in this damped Newton scheme rather than the standard Newton method ($\gamma = 1$) for cases involving challenging collision configurations. For our examples we used the values $N = 3$ and $M = 5$ for the outer and inner loop.

6. Examples

We tested the speed and robustness of our global collision response algorithm in the context of a number of benchmark problems involving many complex collisions of cloth with itself and with harsh environments. The problems all include multiple layers of cloth and undergoing large deformation. Our objective was to demonstrate plausible and efficient simulation of highly complex collision scenarios. In that vein, we attempted to push the limits of our algorithm by processing collisions infrequently, often merely once per simulation frame.

The first example we considered was of an array of five curtains interacting with a sphere and plane (Figure 3). This example is similar to the original benchmark problem of [BFA02] only with more layers of cloth complicating collision response. We also used this example as a basis for a quantitative comparison between our approach, and the method of [BFA02]. We used the following setup:

- The cloth was simulated at 60fps, for a total of 280 frames.
- For both methods, we needed to specify the base time step used for the semi-implicit Newmark integrator. This is commonly specified as a multiple of the time step determined from the CFL condition. This multiplier is called the CFL number, and a value of 1 or less *guarantees* stability of the integrator. In practice higher numbers may be acceptable, and allow for better performance albeit at the risk of reduced stability. For the simulations using the method of [BFA02] we experimented with CFL numbers of 1 and 4. For our method, a CFL number of 4 was succesfully used throughout the simulation.
- A second parameter influencing the simulation is the time interval between successive collision processing instances. We specified this parameter in terms of the *maximum* number of Newmark integration loops that are performed before performing collision processing. We note that the exact number of integration loops per collision step is adjusted in response to a previous successful or failed collision handling attempt, as detailed in Section 3. For our method, we used a maximum of 16 integration loops per collision attempt, while for the method of [BFA02] we experimented with both 16 and 64 maximum loops.
- Although both methods could be combined with a failsafe that performs grouping into rigid impact zones, as described in Section 3, we chose not to exercise this option. This failsafe is invoked when repeated failures in collision resolution bring the number of integration loops below the user-specified minimum number of loops (which was one loop, for our tests). Our method never needed to resort to this failsafe. The unmodified method of [BFA02] often did resort to rigid impact zones, however at a cost of severe artifacts often leading to unavoidable failure at subsequent frames. Since our comparison was on a frame-by-frame basis, when the method of [BFA02] would not manage to resolve collisions even with a single integration loop per collision step we would consider this failed frame for their approach.
- All 280 frames were originally simulated with our new collision scheme. Subsequently, we would use each of these (collision-free) simulated frames as an initial configuration and use the method of [BFA02] to advance this configuration for one frame at a time. This guaranteed that both methods were supplied with the exact same configuration at the beginning of each frame.

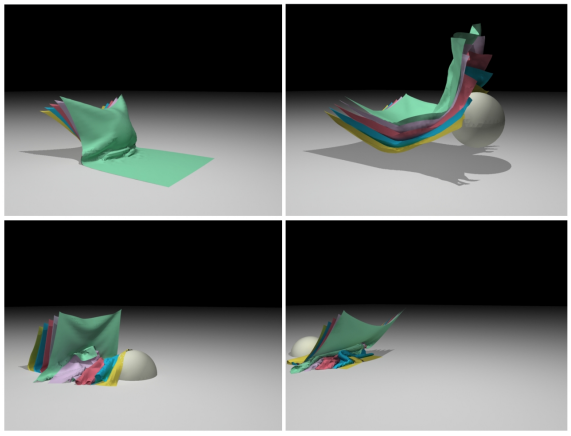

Figure 3: *Five layers of cloth interact with a spherical collision object, using our globally coupled impulse scheme*

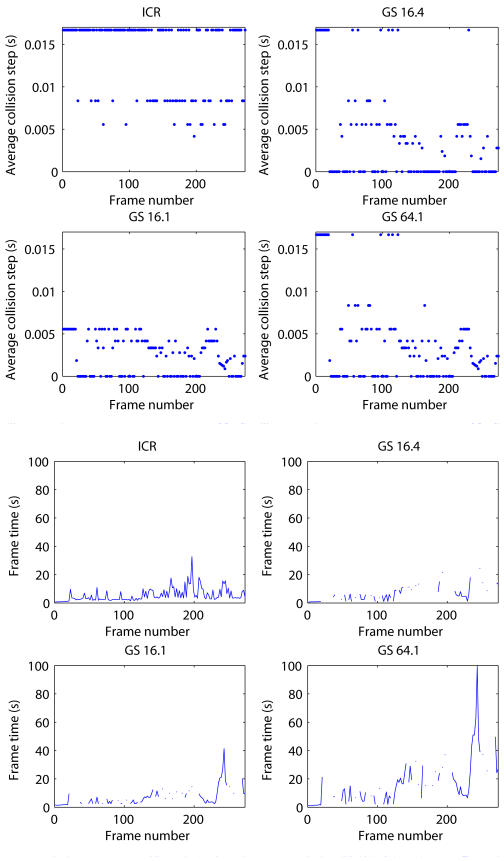

Figure 4: *Comparison of our incompressible collision response method (ICR) to the following variants of the Gauss-Seidel collision processing scheme of [BFA02]: GSx.y uses x integration loops between collision steps, and a CFL number of y. The top figures illustrate the average time interval between successful collision resolutions (note that our approach frequently operates at the frame rate of 60Hz). In the bottom, we provide a comparison of frame run times for the 4 algorithms described. Zero or missing values denote failure of the respective scheme to resolve collisions, even after reducing the collision attempt interval to one integration loop.*

Our findings are illustrated in Figure 4. Notably, our new method managed to resolve collisions for the majority of frames without dropping below the maximum allowable number of integration loops per collision step (which led to collision resolution typically being performed at the frame rate of 60fps). The original algorithm of [BFA02] was not able to match this performance, and in many frames did not succeed in resolving collisions at all, even after performing just one integration loop per collision step, or when using a smaller CFL number of 1. We also demonstrated the scalability of our global response algorithm by dropping 30 towels onto a hammock-like cloth suspension (Figure 2–Left). Our method was able to simultaneously resolve collision for tens of thousands of collision pairs resulting in very efficient run time. Finally, we investigated the ability of our method to resolve a collision free state even in the context of pinching and self colliding kinematic environment geometry. We first draped three sheets between two rotating spheres (Figure 2–Middle). The spheres come together and overlap trapping the three layers of sheets inside. We allow conflicting kinematic constraints in this case by also simulating and colliding the kinematic simulation components as cloth.

7. Limitations and conclusion

Our method is motivated by the response of an incompressible fluid inserted between the colliding surfaces. Yet, we do not explicitly model this fluid buffer using a standard discretization, such as an Eulerian grid. Instead, we approximate this fluid using the tetrahedra defined by colliding point-triangle or edge-edge pairs. This in an imperfect approximation and as a result our response will deviate from the fluid behavior we intend to emulate. Additionally, there are certain behaviors of a fluid cushion placed between neighboring surfaces that are undesirable of a cloth simulation. For example, enforcing incompressibility of this fluid layer would cause sticking of two near-colliding surfaces that are on a separation trajectory. We prevent this behavior by continuously updating the set of active collision pairs, enforcing the incompressibility constraint only on those collision pairs that are *positively* determined to be on a collision course, and release them once their trajectories are corrected to be non-colliding. Of course, this entails the cost of repeated evaluation of the active collision pairs. In addition, our incompressibility condition (excluding the fluids motivation) may appear to be a less natural choice for cloth, where the prevention of interpenetration would appear to be a less restrictive constraint achieving the same goal. We showed that, for small time steps, the inelastic response computed from our scheme converges to the inelastic response derived from a criterion based on non-interpenetration. However, for larger time steps the two responses may differ. In fact, certain other methods employ comparable simplifications that only converge to the actual phenomenon under refinement. For example, [BFA02] only compute their impulses to zero out the relative velocity along the closest connecting segment of the two primitives in the *original* ($\mathbf{x}^n$) configuration. With large time steps there is no guarantee that this direction coincides with the normal to the collision surface at the *exact* moment of collision. In practice, such modeling errors are hardly the main factor leading to failure of a scheme.

We have presented a method that facilitates collision resolution in complex, layered cloth simulations. We derive the constraint leading to non-interpenetration from the idealized incompressible response of a fluid layer inserted in the gap between colliding surfaces. Our globally coupled formulation of the collision response affords significantly larger intervals between collision processing while maintaining visual plausibility of the simulation.

References

[BA04] BOXERMAN E., ASCHER U.: Decomposing cloth. In *Proc. ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.* (2004), pp. 153–161.

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

[BHW94] BREEN D. E., HOUSE D. H., WOZNY M. J.: Predicting the drape of woven cloth using interacting particles. *Comput. Graph. (SIGGRAPH Proc.)* (1994), 365–372.

[BMF03] BRIDSON R., MARINO S., FEDKIW R.: Simulation of clothing with folds and wrinkles. In *Proc. of the 2003 ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.* (2003), pp. 28–36.

[BW98] BARAFF D., WITKIN A.: Large steps in cloth simulation. In *Proc. SIGGRAPH 98* (1998), pp. 43–54.

[BWH*06] BERGOU M., WARDETZKY M., HARMON D., ZORIN D., GRINSPUN E.: A quadratic bending model for inextensible surfaces. In *Proc. of Eurographics Symp. on Geometry Processing* (2006), pp. 227–230.

[CK02] CHOI K.-J., KO H.-S.: Stable but responsive cloth. *ACM Trans. Graph. (SIGGRAPH Proc.) 21* (2002), 604–611.

[CYMTT92] CARIGNAN M., YANG Y., MAGNENAT-THALMANN N., THALMANN D.: Dressing animated synthetic actors with complex deformable clothes. *Comput. Graph. (SIGGRAPH Proc.)* (1992), 99–104.

[GHDS03] GRINSPUN E., HIRANI A., DESBRUN M., SCHRÖDER P.: Discrete shells. In *Proc. of the 2003 ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.* (2003), pp. 62–67.

[GHF*07] GOLDENTHAL R., HARMON D., FATTAL R., BERCOVIER M., GRINSPUN E.: Efficient simulation of inextensible cloth. *ACM Trans. Graph. (SIGGRAPH Proc.) 26*, 3 (2007).

[GKJ*05] GOVINDARAJU N., KNOTT D., JAIN N., KABUL I., TAMSTORF R., GAYLE R., LIN M., MANOCHA D.: Interactive collision detection between deformable models using chromatic decomposition. *ACM Trans. Graph. (SIGGRAPH Proc.)* (2005), 991–999.

[GKS02] GRINSPUN E., KRYSL P., SCHRÖDER P.: CHARMS: A simple framework for adaptive simulation. *ACM Trans. Graph. (SIGGRAPH Proc.) 21* (2002), 281–290.

[HB00] HOUSE D. H., BREEN D. E. (Eds.): *Cloth modeling and animation*. A. K. Peters, 2000.

[ISF07] IRVING G., SCHROEDER C., FEDKIW R.: Volume conserving finite element simulations of deformable models. *ACM Trans. Graph. (SIGGRAPH Proc.) (in press) 26*, 3 (2007).

[MDDB01] MEYER M., DEBUNNE G., DESBRUN M., BARR A. H.: Interactive animation of cloth-like objects in virtual reality. *The Journal of Visualization and Computer Animation 12*, 1 (2001), 1–12.

[OAW06] OH S., AHN J., WOHN K.: Low damped cloth simulation. *Visual Computer 22*, 2 (Feb. 2006).

[OITN92] OKABE H., IMAOKA H., TOMIHA T., NIWAYA H.: Three dimensional apparel CAD system. *Comput. Graph. (SIGGRAPH Proc.)* (1992), 105–110.

[Pes02] PESKIN C.: The immersed boundary method. *Acta Numerica 11* (2002), 479–517.

[PF02] PARKS D., FORSYTH D.: Improved integration for cloth simulation. In *Proc. of Eurographics* (2002), Comput. Graph. Forum, Eurographics Assoc.

[SGG*06] SUD A., GOVINDARAJU N., GAYLE R., KABUL I., MANOCHA D.: Fast proximity computation among deformable models using discrete Voronoi diagrams. *ACM Trans. Graph. (SIGGRAPH Proc.)* (2006), 1144–1153.

[TC92] THINGVOLD J. A., COHEN E.: Physical modeling with B-spline surfaces for interactive design and animation. *Comput. Graph. (SIGGRAPH Proc.)* (1992), 129–137.

[TF88] TERZOPOULOS D., FLEISCHER K.: Modeling inelastic deformation: viscoelasticity, plasticity, fracture. *Comput. Graph. (SIGGRAPH Proc.)* (1988), 269–278.

[TPBF87] TERZOPOULOS D., PLATT J., BARR A., FLEISCHER K.: Elastically deformable models. *Comput. Graph. (Proc. SIGGRAPH 87) 21*, 4 (1987), 205–214.

[TWS06] THOMASZEWSKI B., WACKER M., STRASSER W.: A consistent bending model for cloth simulation with corotational subdivision finite elements. In *Proc. ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.* (2006).

[VCMT95] VOLINO P., COURCHESNE M., MAGNENAT-THALMANN N.: Versatile and efficient techniques for simulating cloth and other deformable objects. *Comput. Graph. (SIGGRAPH Proc.)* (1995), 137–144.

[VMT94] VOLINO P., MAGNENAT-THALMANN N.: Efficient self-collision detection on smoothly discretized surface animations using geometrical shape regularity. In *Proc. of Eurographics* (1994), vol. 13 of *Comput. Graph. Forum*, Eurographics Assoc., pp. C–155–166.

[VMT05] VOLINO P., MAGNENAT-THALMANN N.: Implicit midpoint integration and adaptive damping for efficient cloth simulation. *Computer Animation and Virtual Worlds 16* (2005), 163–175.

[VMT06] VOLINO P., MAGNENAT-THALMANN N.: Simple linear bending stiffness in particle systems. In *Proc. of the ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.* (2006), pp. 101–105.

[Wei86] WEIL J.: The synthesis of cloth objects. *Comput. Graph. (SIGGRAPH Proc.)* (1986), 49–54.