Minimum Displacements For Cloth-obstacle Penetration Resolving

Liming Sun[1] and Timo R. Nyberg[2,5] and Gang Xiong[3,4] and Juntao Ye[†1]

[1]NLPR-LIAMA, Institute of Automation, CAS, China. [2]Cloud Computing Center, CAS, Dongguan, China.
[3]SKL-MCCS, Institute of Automation, CAS, China. [4]School of Automation, Foshan University, China.
[5]Department of Industrial Engineering and Management, Aalto University, Finland.

Abstract

Pre-existing penetrations often show up in many applications, particularly in garments fitting. The popular continuous collision detection (CCD) based methods are incapable of handling them, as there is no history information to rely on. On the other hand, surfaces of human bodies have normals defined to designate their orientation (i.e. front- and back-face), which are totally overlooked by CCD methods (thus they are orientation-free). In this paper we present a history-free method for separating two penetrating meshes, given that one of them represents a rigid object and has clarified surface orientation. This method computes all edge-face (E-F) intersections with discrete collision detection, and identifies illegal vertices with the help of surface orientation, and then builds a number of penetration stencils. On response, the stencil vertices are relocated into a penetration-free state, via a global displacement minimizer. The proposed algorithm outperforms existing methods for handling solid/cloth collisions, thus is an effective tool for applications like virtual-try-on and example-based garment animation synthesis.

1. Introduction

In garment fitting applications, pre-existing penetrations exist between human model and garment model. Resolving deformable collision has been extensively studied in the context of physically based simulation. Yet the majority of proposed approaches [Pro97, BFA02, BEB12, HVS*09, AVGT12, HVTG08] are history based and are not capable of handling pre-existing penetrations in garment fitting. They exploit continuous collision detection (CCD) to predict impending collisions, then attempt to prevent them by altering particles' velocities. The success of CCD-based response method relies on a hard constraint: a collision-free state for not only the initial configuration but also the start of every time interval in the simulation. As a matter of fact, the history is used to derive the orientation on-the-fly in order to decide whether a geometric primitive is located on the incorrect side of a surface or not. CCD methods are totally blind to existing orientation information. For pre-existing penetrations there is no *a priori* collision-free status for reference, so they can only be tackled by history-free methods. These methods require orientation information of the surface, either specified explicitly or implicitly. Fortunately, in garment fitting a human model is a usually watertight surface, and body surface normals always point to the outside so that no cloth vertices are allowed to go inside the body. Separating inter-penetrations is also one of the key issues considered in the domain of garment animation synthesis. Wang *et al.* [WHRO10] mentioned an idea of using triangle normals to determine if a vertex is in penetrating status. Yet formulation details were not given. Guan *et al.* [GRWB12] presented a much detailed algorithm on how to move penetrating cloth vertices out of the human body. As in [BMF03] only penetrating vertices but not edges are corrected, these methods are not robust and often relies on an extra relaxation factor to work.

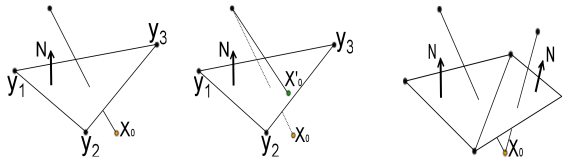
In the case where no explicit orientation is available, heuristic rules are often exploited. Baraff *et al.* [BWK03] addressed the necessity of history-free collision correction in a complex simulation environment, and put forward a solution for resolving one type of intersection – the contour being a closed curve. Volino and Magnenat-Thalmann [VM06] proposed an intersection contour minimization (ICM) method, intending to rely on neither history nor orientation, to handle any type of contours. Yet they overlooked a fact that without history and orientation, separating two penetrating surfaces is an ambiguous and ill-defined problem [WLG06]. Not surprisingly, the ICM method often exhibits unpredictable convergence behavior, even for cases where surface orientation is available, as pointed by Ye and Zhao [YZ12]. Narain *et al.* [NSO12] encountered the interpenetration problem while remeshing cloth to enrich details, and the penetrating vertices were relocated according to the previous penetration-free state. Yet this method does not suit applications that lack *a priori* collision-free states as reference.

Our goal is to give a solution that takes advantage of surface orientation if it is available, and no longer relies on the history information. Different from prevention-based CCD methods, our method embraces the *repair* strategy – penetrations are allowed to

† Corresponding: juntao.ye@ia.ac.cn

DOI: 10.2312/egsh.20161013

occur but will be detected and penetrating objects will be separated. It employs *discrete collision detection* (DCD) so it is history-free. DCD only finds out edge-face (E-F) intersections for two given meshes, thus is faster than CCD for two points: bounding volumes are more tight-fitting and no cubic solver is needed. Please note that at the time being our method is capable of handling collisions in-between two different meshes, at least one of them being oriented. How to extend our method to handle self-collision is still under investigation.

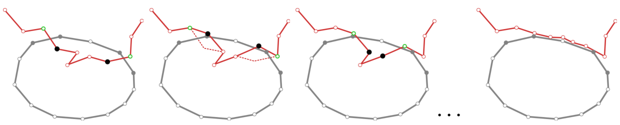


(a) (b) (c)

Figure 2: (a) four vertices $\{\mathbf{x}_0, \mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3\}$ form a V-F penetration stencil; (b) penetration is resolved by relocating $\mathbf{x}_0$ to $\mathbf{x}'_0$ which is above the triangle plane; (c) a configuration of a two-stencils.

2. Cloth-Body Penetration Handling

We present a robust algorithm for resolving body/garment penetrations. It is based on an assumption that the body mesh is an oriented surface with its face normals pointing to the positive side (or front side). Each penetrating region is formulated as a series of edge-face (E-F) intersections. Our method iteratively detects E-F intersections, and identifies and relocates penetrating vertices, until a collision-free status is obtained. In each iteration, the vertex of an intersecting edge located on the negative side of an oriented face is identified as illegal. The new positions of the illegal vertices are computed so that the vertex-to-face signed distances are zeroed. In the next round of iteration, the detector checks the new geometries, and reclassifies the legal, illegal and un-determined vertices. We repeat this process until there are no more E-F intersections (see Fig. 1 for this pipeline).



(a) (b) (c) ... (d)

Figure 1: An analogues 2D penetration configuration being resolved by our method step by step. The gray curve represents an oriented closed surface, and the red poly-line denotes an un-oriented surface. Vertex of an intersecting red edge is marked as *green circle* if it is in legal state, or marked as *black dot* if it is in illegal state; Vertices of the intersecting face are marked as *gray dots*. All other vertices are specified as *undetermined*.

We denote vertices of an n-vertex deformable mesh $\mathbf{x} = [\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_{n-1}] \in \mathbb{R}^{3n}$, and similarly $\mathbf{y} = [\mathbf{y}_0, \mathbf{y}_1, \cdots, \mathbf{y}_{m-1}] \in \mathbb{R}^{3m}$ for a solid mesh. When resolving penetration between a deformable mesh and a solid mesh, the latter remains unchanged after the resolution. Therefore, only $\mathbf{x}$ is treated as unknown. In Fig. 2(a), the face $\mathbf{y}_1\mathbf{y}_2\mathbf{y}_3$, from the body mesh, has a normal designating its front-face orientation. Vertex $\mathbf{x}_0$, from the garment mesh, is located illegally on the back-face side of the face. If we could relocate these vertices so that $\mathbf{x}_0$ goes to the other side of the face (Fig 2(b)), this penetration will be resolved. We name the quadruple $\{\mathbf{x}_0, \mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3\}$ a *penetration stencil*. A stencil has three vertices coming from an oriented face, and one vertex from an edge, be it from an oriented surface or not.

We view each E-F intersection as the violation of a scalar-valued function, $D(\mathbf{x},\mathbf{y})$, with $D(\mathbf{x},\mathbf{y}) < 0$ whenever $\mathbf{x}$ is in a penetration configuration with $\mathbf{y}$. For V-F penetration the function can be considered as a *signed distance* from vertex $\mathbf{x}_0$ to triangle $\mathbf{y}_1\mathbf{y}_2\mathbf{y}_3$:

$$D(\mathbf{x},\mathbf{y}) = \mathbf{n} \cdot [\mathbf{x}_0 - (\beta_1\mathbf{y}_1 + \beta_2\mathbf{y}_2 + \beta_3\mathbf{y}_3)] \tag{1}$$

where $\mathbf{n}$ is the triangle normal, and $\beta_1, \beta_2, \beta_3$ are the barycentric coordinates of the intersection point with respect to triangle vertices. Each stencil can be differentiated with respect to $\mathbf{x}$ to yield the *stencil gradient*, $D_\mathbf{x}$, a vector in configuration space. For the V-F penetration above, the stencil gradient expressed in the local indices of the stencil is $D_\mathbf{x} = \mathbf{n}$.

We are typically faced with k penetration stencils, *i.e.*, k V-F penetrations with possibly non-disjoint stencils. To elevate the stencil gradient in the $\mathbb{R}^{3n}$ space, we simply map the local indices in Equ. 1 to their global position and re-write $D_\mathbf{x}$ as a $3n \times 1$ column vector, with zeros everywhere else, in the style of finite element stiffness matrix assembly. Now let $D_\mathbf{x} = [D_1, \cdots, D_k]$ be a $3n \times k$ matrix whose columns span the possible displacement directions (i.e., face normals). As in the single-stencil case, we require that $\lambda_1 \cdots \lambda_k$ be non-negative, since displacements are to cancel the negative distance, and lead to a non-negative post-response distance.

We propose an algorithm for approximating λ by assuming the post-response distance to be exactly zero: $D(\mathbf{x}',\mathbf{y}) = \mathbf{0}$. We relax the conditions on a response being valid to allow both positive and negative entries in λ, then $\mathbf{x}'$ is the minimizer of

$$||\mathbf{x}' - \mathbf{x}||^2, \quad \text{subject to} \quad D(\mathbf{x}',\mathbf{y}) = \mathbf{0}. \tag{2}$$

This minimization seeks new vertex positions that are as close as possible to the old positions, by projecting the illegal vertices onto the orthogonal complement of the span of the rows of $D_\mathbf{x}$. We may repose the above as an extremization of the augmented functional

$$W(\mathbf{x}',\lambda) = \frac{1}{2}||\mathbf{x}' - \mathbf{x}||^2 - \lambda D(\mathbf{x}',\mathbf{y}), \tag{3}$$

with respect to $(\mathbf{x}',\lambda)$, where $\lambda \in \mathbb{R}^{3n}$ is a vector of Lagrange multiplier. The corresponding stationary equations are

$$\mathbf{0} = \frac{\partial W}{\partial \mathbf{x}'} = \mathbf{x}' - \mathbf{x} - \lambda D_{\mathbf{x}'}, \tag{4}$$

$$\mathbf{0} = \frac{\partial W}{\partial \lambda} = D(\mathbf{x}',\mathbf{y}). \tag{5}$$

The first stationary equation guarantees that the response acts only along the $D_{\mathbf{x}'}$ direction, and the second one ensures vanishing negative distance. Please note there is $D_{\mathbf{x}'} = D_\mathbf{x}$. Substituting the first into the second yields

$$D_\mathbf{x}^T D_\mathbf{x} \lambda = -D(\mathbf{x},\mathbf{y}) \tag{6}$$

for the unknown λ, and then substituting it into the first stationary equation to obtain positional displacement for a set of k penetration stencils. This process can be interpreted as projecting the illegal vertices onto the surface they penetrate. To ensure illegal vertices are sufficiently outside the faces, the post-response distances can be relaxed from zero to certain small positive values, i.e., $D(\mathbf{x}',\mathbf{y}) = \mathbf{d}$, with $\mathbf{d} = (d_1, d_2, \cdots, d_k)^T$. Equ. 6 becomes

$$D_{\mathbf{x}}^T D_{\mathbf{x}} \lambda = -D(\mathbf{x},\mathbf{y}) + \mathbf{d}. \tag{7}$$

The values of **d** can be chosen to be a small fraction of averaged edge length of the mesh. In dynamics simulations, we set **d** to be the proximity threshold, which is often the cloth thickness. Pushing the geometries to be that distance apart has a good chance to add less burden to the proximity check in the next timestep.

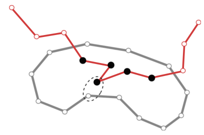
In implementing the above algorithm we follow the concept of *impact zone* (IZ) of [Pro97,BFA02,HVTG08] and compute the displacements globally. Each penetration region, consisting of multiple stencils, is called an impact zone. As each stencil gradient depends on four vertices, an impact zone is defined as a set of overlapping stencils. The projection operation (Equ. 2) is performed per IZ, and each IZ corresponds to one linear system of Equ. 6. With impact zone, the case that an edge intersects multiple faces can be handled without any special treatment. Multi-intersection means multiple stencils. Please note that when an edge intersects even number of faces, two end-vertices are on the same side of the surface. These stencils can be fed into the IZ simultaneously to solve for displacements; the E-F intersections disappear after a number of iterations.

3. Results and Discussion

Comparisons. There exist some work that rely on surface orientation for separating inter-penetrations. The ICM method [VM06], claimed to be both history- and orientation-independent, is actually less effective for configurations involving oriented surfaces. Even the improvement work of [YZ12] still has the following issues:

- The displacement vector is computed per edge basis and then assigned to its end-vertices. For a vertex shared by multiple intersecting edges, these edge displacements may conflict with each other. Moreover, how much contributions an edge assign to its vertices are not clear.
- The magnitude of the displacement vector in ICM is determined by an artificial arctangent function, which has neither geometric nor physical support.
- The global scheme of ICM does a brutal averaging of all displacements for vertices along an intersection contour. This disrespects the distinct displacement for each individual vertex, and therefore some vertices may be over-shot and some may be under-shot.

In DRAPE [GRWB12], another method was proposed to separate deformable penetrations. For each penetrating cloth vertex a closest body vertex is found, and the cloth vertex is then moved along the normal of that body vertex until it is outside of the body surface. For a very deep penetrating cloth vertex, its closest body vertex may not be a good destination reference and moving towards it may cause discontinuities (see inset figure), as suggested by Heidelberger *et al.* [HTK*04]. Our method guarantees a consistent penetration resolving. Second, DRAPE merely checks vertex-face but not edge-face penetrations from one mesh against the other. This is not sufficient to catch all collisions; sparsely sampled cloth model often produces intersections with high curvature objects that can not be processed. Therefore, DRAPE easily fails for scenarios as shown in the accompany video.



Fitting Examples. To evaluate the proposed approach, we fit several garment models onto a human body (Fig. 3). In an example-based clothing synthesis system, the machine learned model can only generate a roughly matched garment shape, and interpenetrations have to be resolved at the refining stage. And the same requirement applies to a virtual try-on system after a garment model is aligned with the body. This experiment emulates the latter scenario and four garments are fit onto a human while being physically simulated. In some cases we attribute severe interpenetrations to the initial position of the garment; in others we tight-fit the garment to the body.

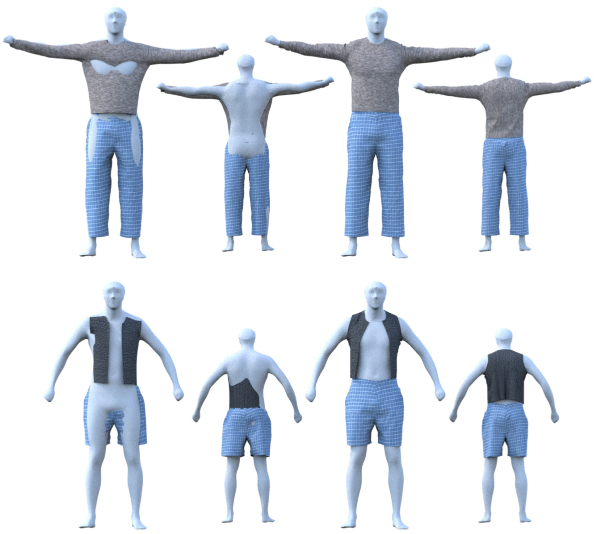


Figure 3: Fitting various garment models on a human.

Integration with physical simulation. The proposed method can also be used in context of physical simulation, handling cloth/obstacle collisions in a correction-based manner, as opposed to the prevention-based manner in CCD methods. A popular CCD algorithm described in [HVTG08] was implemented in the cloth simulator ARCSim [NSO12], handling both cloth/body and cloth/cloth collisions. We integrated our method into ARCSim and let it handle cloth/body collisions, leaving the simulator's built-in resolver to handle cloth/cloth collisions. The timing, in Table 1, was collected with single-threading on an Intel Core2 Quad CPU @ 2.83GHz. This example has more cloth/body collisions than cloth/cloth collisions, and our method decreases the cost of cloth/body handling dramatically. Therefore the entire collision handling time of ARCSim is 60% more than ours. The result (see

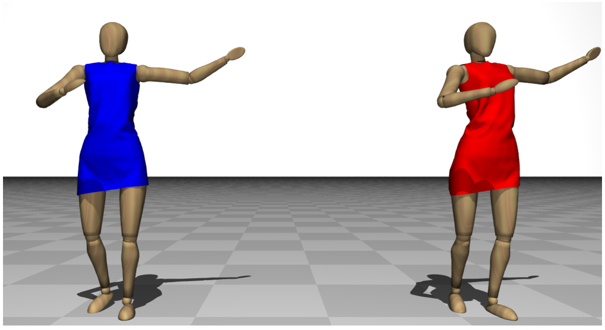


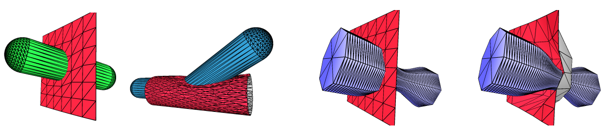
Figure 4: Replacing the solid/cloth collision handling with our method, our simulation runs faster and the result (right) does not suffer any quality loss compared to the result of ARCSim (left). The dummy model has 13,481 vertices and the dress model has 3,984 vertices.

Fig. 4 and the accompanying video) shows that our method does not introduce any quality loss.

	Physics	Proximity	Collision handling		Total
ARC-Sim	1923.4s (48.9%)	701.9s (17.9%)	1284.6s (32.7%)		3931.8s
			C/B	C/C	
Ours	1900.1s (56.8%)	598.9s (17.9%)	15.1s (0.44%)	791.1s (23.7%)	3344.2s

Table 1: The timings of simulation two experiments for 500 frames each. C/B means cloth/body collision handling and C/C means cloth/cloth handling.

Limitations. The major limitation of the proposed algorithm is that it is not applicable to self-collision of a single mesh. For cloth/solid penetrations, our method may fail for some extreme cases. Fig. 5 shows several examples. In the first two cases, the orientation of the colliding portion of the cylinder surface can not determine the moving the direction for the illegal cloth vertices. Therefore, an initial configuration like Fig. 5b between a human arm and a sleeve fails, even if the arm mesh contracts completely into its bone. In the case of Fig. 5c, the normal of the closed surface makes the cloth to be stuck at the local minimum as in Fig. 5d. Please note that these are all extreme cases that are unlikely to happen in real applications, unless they are set up intentionally.



(a) (b) (c) (d)

Figure 5: Configurations that can not be processed by our method.

4. Conclusion

We have presented an efficient history-free method to resolve cloth-body collisions. Being able to handling pre-existing penetrations, it is particularly useful for garment syntheses and the initialization of a physical simulation. The experiment results show that it overcomes drawbacks seen in other techniques for solid/cloth collision handling.

Acknowledgements

The authors appreciate garment models from UC Berkeley, and the human model shared by Nils Hasler. This work is partially supported by NSF China (Grants #61379096, #71232006, #61233001); Finnish TEKES's project "SoMa2020:Social Manufacturing" (2015-2017); Chinese Guangdong's S&T projects (2014B010118001, 2014B090902001, 2014A050503004).

References

[AVGT12] Ainsley S., Vouga E., Grinspun E., Tamstorf R.: Speculative parallel asynchronous contact mechanics. *ACM Trans. Graph. 31*, 6 (2012), 151:1–151:8. 1

[BEB12] Brochu T., Edwards E., Bridson R.: Efficient geometrically exact continuous collision detection. *ACM Trans. Graph. 31*, 4 (2012), 96:1–96:7. 1

[BFA02] Bridson R., Fedkiw R., Anderson J.: Robust treatment of collisions, contact and friction for cloth animation. *ACM Trans. Graph. 21*, 3 (2002), 594–603. 1, 3

[BMF03] Bridson R., Marino S., Fedkiw R.: Simulation of clothing with folds and wrinkles. In *Proc. Symp. Computer Animation* (2003), pp. 28–36. 1

[BWK03] Baraff D., Witkin A., Kass M.: Untangling cloth. *ACM Trans. Graph. 22*, 3 (2003), 862–870. 1

[GRWB12] Guan P., Reiss L.and Hirshberg D., Weiss A., Black M. J.: DRAPE: Dressnig any person. *ACM Trans. Graphics (Proc. SIGGRAPH) 31*, 4 (jul 2012). 1, 3

[HTK*04] Heidelberger B., Teschner M., Keiser R., Müller M., Gross M.: Consistent penetration depth estimation for deformable collision response. In *Proc. of Vision, Modeling, Visualization* (2004), pp. 339–346. 3

[HVS*09] Harmon D., Vouga E., Smith B., Tamstorf R., Grinspun E.: Asynchronous Contact Mechanics. *ACM Trans. Graph. 28*, 3 (2009). 1

[HVTG08] Harmon D., Vouga E., Tamstorf R., Grinspun E.: Robust treatment of simultaneous collisions. *ACM Trans. Graph. 27*, 3 (2008). 1, 3

[NSO12] Narain R., Samii A., O'Brien J. F.: Adaptive anisotropic remeshing for cloth simulation. *ACM Trans. on Graph. 31*, 6 (2012), 147:1–10. 1, 3

[Pro97] Provot X.: Collision and self-collision handling in cloth model dedicated to design garments. In *EG Workshop on Computer Animation and Simulation* (1997), pp. 177–189. 1, 3

[VM06] Volino P., Magnenat-Thalmann N.: Resolving surface collisions through intersection contour minimization. *ACM Trans. Graph. 25*, 3 (2006), 1154–1159. 1, 3

[WHRO10] Wang H., Hecht F., Ramamoorthi R., O'Brien J.: Example-based wrinkle synthesis for clothing animation. *ACM Trans. Graph. 29*, 4 (2010), 107:1–107:8. 1

[WLG06] Wicke M., Lanker H., Gross M.: Untangling cloth with boundaries. In *Proc. of Vision, Modeling, and Visualization* (2006), pp. 349–356. 1

[YZ12] Ye J., Zhao J.: The intersection contour minimization method for untangling oriented deformable surfaces. In *Proc. Symp. Computer Animation* (2012), pp. 311–316. 1, 3