DOI: 10.1111/cgf.15187
ACM SIGGRAPH / Eurographics Symposium on Computer Animation 2024
M. Skouras and H. Wang
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 43 (2024), Number 8

Robust and Artefact-Free Deformable Contact with Smooth Surface Representations

Y. Du, Y. Li, S. Coros and B. Thomaszewski

ETH Zürich, Switzerland

Abstract

Modeling contact between deformable solids is a fundamental problem in computer animation, mechanical design, and robotics. Existing methods based on C^0-discretizations—piece-wise linear or polynomial surfaces—suffer from discontinuities and irregularities in tangential contact forces, which can significantly affect simulation outcomes and even prevent convergence. In this work, we show that these limitations can be overcome with a smooth surface representation based on Implicit Moving Least Squares (IMLS). In particular, we propose a self collision detection scheme tailored to IMLS surfaces that enables robust and efficient handling of challenging self contacts. Through a series of test cases, we show that our approach offers advantages over existing methods in terms of accuracy and robustness for both forward and inverse problems.

CCS Concepts

• ***Computing methodologies*** → ***Physical simulation; Continuous simulation;***

1. Introduction

Modeling contact between deformable objects is a fundamental problem in science and engineering. It is at the heart of many applications in computer animation, mechanical design, and robotics. The computer graphics community has made great strides in contact detection and response over the past two decades. The state-of-the-art is perhaps best reflected in the Incremental Potential Contact (IPC) method by Li *et al*. [LFS*20] that combines fail-safe collision detection with implicit integration of contact forces based on log-barrier penalty functions. This algorithm is able to generate compelling and stable animations for highly challenging contact scenarios. Nevertheless, there are still limitations with existing formulations that demand further investigation. Here we draw attention to a particular and fundamental problem, *i.e*., the discontinuities and irregularities in tangential contact forces induced by non-smooth surface discretization.

Li et al. [LFS*20] rely on per-primitive potentials that prevent too close approach between vertex-triangle and edge-edge pairs. To avoid discontinuities in contact forces, IPC allows for multiple per-primitive potentials to be active simultaneously. While this strategy yields smooth forces when contact points migrate from one triangle to another, our analysis shows that the superposition of per-primitive potentials leads to energy walls at element transitions that resist tangential motion—even for the perfectly planar, frictionless case. These erroneous forces degrade the accuracy of forward simulations and can even prevent convergence for inverse problems.

IPC addresses this problem using smoothly-clamped penalty functions that allow multiple collision pairs for the same nodes to

Figure 1: *We propose a new approach for robust and efficient contact handling between deformable objects. Using smooth surface representations, our method enables artefact-free contact forces during sliding motion.*

be active simultaneously. While there are no contact force discontinuities with this approach, our analysis shows that the superposition of per-primitive potentials leads to energy walls at element transitions that resist tangential motion, even for the perfectly planar, frictionless case. These erroneous forces degrade the accuracy of forward simulations and can even prevent convergence for inverse problems.

In this work, we show that an alternative approach for deformable contact modeling based on smooth surface representations eliminates contact force discontinuities and irregularities. Our method builds on Implicit Moving Least Squares (IMLS), a meshless representation that describes smooth surfaces as the zero levelset of an implicit function defined through position and normal data. While IMLS surfaces have been explored for contact handling before [LFP21], we propose a novel self collision detection scheme tailored to IMLS surfaces that enables robust and efficient handling of challenging self contacts. As we show through a series of test cases, our smooth representation offers advantages over existing methods based on C^0-discretizations for both forward and inverse problems.

In summary, our work makes the following contributions:

- We identify a fundamental limitation of contact handling methods for deformable bodies that rely on C^0-discretizations. In particular, we show that spurious tangential forces can significantly affect the relative motion between contacting surfaces.
- We propose a self collision detection scheme for IMLS-based surfaces that eliminates false positives using geodesic carving.
- We introduce a two-stage continuous collision detection scheme for IMLS-based surfaces that combines rapid quadratic distance tests with 1D nonlinear minimization for robust and efficient intersection computation.
- We demonstrate that contact handling with our IMLS-based approach eliminates force discontinuities and other irregularities.

We furthermore propose a series of forward and inverse problems to analyze and quantify contact force irregularities. Using these tests, we show that existing methods based on C^0-discretizations can significantly alter simulation outcomes and may even prevent convergence while our smooth formulation eliminates these problems.

2. Related Work

2.1. Contact Modeling in Graphics

Handling contact is a fundamental problem in graphics. Early research primarily focused on contacts between rigid bodies [Bar91, MW88, MC95, KEP05, KSJP08, KP03]. To handle contacts between deformable bodies, most approaches relied on impulses, i.e., velocity-level corrections for contacting primitive pairs [BFA02, HVTG08]. However, as a post process to time stepping, these impulses do not result in equilibrium states. Instead of using quadratic penalty potentials, Harmon *et al.* [HVS*09] proposed an asynchronous time stepping strategy with layered collision potentials, which are infinitely stiff in the limit. While accurate and robust, their explicit time stepping can lead to extremely long computation times. Another way of coupling contact forces and dynamics is through constraint-based approaches, which often rely on linear complementarity problems (LCP) [CPS09]. Kaufman *et al.* [KSJP08] reduce the formulation for rigid bodies and reduced-order deformable objects to a pair of coupled quadratic program problems and solve them iteratively using staggered projections. More recently, Li et al. [LFS*20] introduce a fully implicit treatment of contact using smoothly clamped barrier functions. This incremental potential contact (IPC) paradigm has been generalized to handle co-dimensional objects [LKJ21], rigid bodies [FLS*21, LKL*22], and reduced models [LYK*21]. While IPC enjoys significantly improved robustness, we show that the superposition of per-element potentials leads to spurious tangential forces that can significantly affect simulation outcomes.

2.2. Frictional Contact

Accurate modeling of friction is of paramount importance in many applications and has been extensively studied in the graphics community for both forward modeling [Dav20, VJ19, LDN*18, LFP21] and inverse design [LDW*22, GHZ*20]. However, the inherently non-smooth nature of the governing Maximal Dissipation Principle [GRP91, Mor11] poses significant challenges for conventional Newton-type integrators. Consequently, a variety of customized second-order solvers have been developed, such as the Non-smooth Newton solvers [AC91, DBDB11, MEM*19] and different iterative strategies [JM92, KSJP08, OTSG09, KTS*14]. Our method incorporates friction using a smooth model with a lagged update of contact data similar to IPC. We show that even for frictional contact, the spurious tangential forces from IPC can induce macroscopic artifacts and lead to less accurate behavior.

2.3. Contact Problems in Engineering

Contact problems have been extensively studied by the engineering community [WZ04, PHW19]. Perhaps the most widely used method is the Node-to-segment (NTS) approach [HTS*76, ZD09]. However, the contact force discontinuities arising from simple NTS methods are well-known and alternatives have been explored. Using an integral formulation that extends over the entire region of contact, Mortar methods [MMP88, Ber89] are arguably the most accurate approaches that exist today. We refer to the work by De Lorenzis *et al.* [DLWW17] for an overview and comparisons. For all its accuracy and robustness, the complexity of Mortar methods is substantial. Integrating this approach into an inverse problem solver seems all but intractable. A more recent work by Elandt *et al.* [EDSR19] computes contact forces between nominally rigid bodies using continuous pressure fields. These fields are precomputed for each object and used at run time to determine approximate contact surfaces. While this approach offers fast computation of continuous contact forces, it requires interpenetration between objects and cannot properly capture the mechanical properties of elastic materials.

2.4. Contacts with Implicit Representations

Implicit representations have found widespread applications in various areas of computer graphics [TO02, SOS04] and have emerged as a valuable tool for handling contact. Previous work has leveraged signed distance fields (SDFs) [JBS06, FPRJ00, KDBB17]

for contact handling between deformable objects [FL01, Gas93] and skinning techniques [MZS*11, VBG*13, VGB*14]. Macklin *et al*. [MEM*20] further improve the robustness of this approach by replacing point-based sampling with a method that finds the closest point between the SDF iso-surface and a given mesh vertex. However, one limitation of SDFs defined on polygonal meshes is their potential discontinuity across element boundaries, unless parallel edges are specifically considered [LKJ21]. Inspired by Larionov *et al*. [LFP21], we use Implicit Moving Least Squares (IMLS) [Lev04, Kol08, OGG09] to construct C^2-continuous surfaces whose smooth normal fields eliminate contact force irregularities. We show through experiments that this approach leads to higher accuracy for both forward simulation and inverse design.

3. Deformable Contact with Piece-wise Linear Representations

We consider contact problems between deformable bodies $\mathcal{B}^k$ whose surfaces are given as triangle meshes. Let $\mathbf{x}_t$ denote the vector of nodal positions describing the configuration $\mathcal{B}^k$ for a given time t. We use implicit Euler for stable time integration and minimize the corresponding per step potential [MTGG11]

$$\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}} E(\mathbf{x}) = \tfrac{1}{h}(\mathbf{x}-\mathbf{x}_t)^{\mathsf{T}}\mathbf{M}\tfrac{1}{h}(\mathbf{x}-\mathbf{x}_t) + E_{\text{elastic}}(\mathbf{x}) + \mathbf{x}^{\mathsf{T}}\mathbf{f}_e + E_{\text{contact}}(\mathbf{x}), \tag{1}$$

where $\mathbf{M}$ is the mass matrix, E_{elastic} and E_{contact} are elastic and contact potentials, respectively, and h is the step size.

We follow standard practice and use linear tetrahedron finite elements with a Neo-Hookean material to model the elastic potential of solid bodies [KE22]. For modeling contact between deformable objects, piece-wise linear surfaces are arguably the most commonly used representation. However, constructing contact forces with sufficient smoothness and regularity proves particularly challenging in this discrete setting. We first demonstrate why the simple Node-to-Segment approach fails to satisfy the smoothness requirements of gradient-based optimization methods (Sec. 3.1). Then we discuss the strengths and limitations of IPC (Sec. 3.2).

3.1. Node-to-Segment Method

The Node-to-Segment (NTS) method is a simple approach for contact handling between deformable bodies that is widely used in engineering [HTS*76, ZD09]. One should note that NTS is a 2D method and in 3D both Edge-to-Edge and Node-to-Surface cases should be considered. For simplicity we only discuss the 2D method here but the problems we show remains in 3D. For each node, the NTS method first identifies the closest segment and computes the corresponding distance. Whenever the distance is sufficiently small, a penalty function is instantiated such that the corresponding contact force, collinear with the segment's normal, prevents intersection. Since the normals for piece-wise linear surfaces are piece-wise constant, the contact forces defined in this way will be discontinuous as nodes move from one segment to the next. To see this, consider the example shown in Fig. 2, where two objects $\mathcal{B}^1$ and $\mathcal{B}^2$ are in close contact, with $\mathcal{B}^1$ moving in a given direction as indicated. To compute contact forces, a discrete decision must be made to select the closest segment for a given node (shown in red). As the vertex enters or exits the shaded region where it is equidistant to both edges, the closest edge on surface $\mathcal{B}^2$ changes. Consequently, the contact force experiences an abrupt change in direction. This force discontinuity occurs whenever a given node is

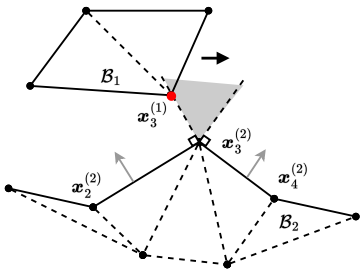


Figure 2: *Contact force discontinuity across element boundaries. As the red vertex moves in the indicated direction, stepping into or out of the gray region leads to discontinuities in contact forces.*

at the same distance to multiple segments. The discrete choice in contact primitive pairs thus presents a fundamental problem of the NTS method.

3.2. Incremental Potential Contact

The incremental potential contact method uses smoothly-clamped barrier functions to enforce non-intersection constraints between close primitive pairs, *i.e*., vertex-triangle and edge-edge pairs. Whereas the NTS method selects a single close segment for each vertex, IPC allows multiple primitive pairs for the same vertex to be active simultaneously. The collision potential for any given vertex is then obtained as the superposition of per-primitive potentials,

$$E_{\text{contact}} = \sum_{(ij,kl)\in\mathcal{E}} b(d_{ee}(\mathbf{x}_i\mathbf{x}_j,\mathbf{x}_k\mathbf{x}_l)) + \sum_{(i,jkl)\in\mathcal{T}} b(d_{vt}(\mathbf{x}_i,\mathbf{x}_j\mathbf{x}_k\mathbf{x}_l)), \tag{2}$$

where $\mathcal{E}$, and $\mathcal{T}$ are index sets for potentially colliding edge-edge, and vertex-triangle pairs. The values returned by the corresponding inter-primitive distance functions d_{vv}, d_{ee}, and d_{vt} are passed into the smoothly-clamped log-barrier function,

$$b(d) = \begin{cases} 0 & d > \hat{d} \\ -(d-\hat{d})^2 \log(d/\hat{d}) & d < \hat{d} \end{cases}, \tag{3}$$

that activates when the distance d falls below a threshold value $\hat{d}$.

Removing the need to make discrete decisions eliminates an important source of non-smoothness. However, the superposition principle also induces energy variations when multiple per-primitive potentials are active. We illustrate this problem using a simple examples in which we examine the contact potential from IPC between a vertex v and a triangulated plane (Fig.3). We assume that the initial distance between vertex and plane is smaller than the

support radius of the barrier function, resulting in a nonzero contact potential. As v moves parallel to the plane, the set of potentially colliding primitive pairs evolves from a single element (face F_1) to three elements (faces F_1, F_2, and F_3), and to two elements (faces F_3 and F_4). As can be seen from Fig. 3, since IPC superimposes contact potentials from all elements in the constraint set, the resulting energy is not constant. Rather, we observe energy peaks at $x = x_2 = 10$ and $x = x_3 = 20$ where contributions from multiple elements amplify to a maximum degree. The resulting *energy walls* give rise to spurious tangential forces that affect simulation outcomes and may stall progress for inverse problems as we show in Sec. 5.

The problems encountered with NTS and IPC approaches are ultimately different manifestations of an inherent limitation of piecewise linear surfaces. It is worth noting that using elements with higher polynomial degree alone does not resolve this problem unless inter-element continuity is explicitly enforced. Rather, addressing these problems necessitates a fully C^1-continuous surface representation, which we introduce next.

4. Deformable Contact with Smooth Representations

To avoid the inherent limitations of piece-wise polynomial surfaces for contact modeling, we resort to smooth surface representations based on Implicit Moving-Least Squares (IMLS).

IMLS surfaces have already been explored for modeling frictional contact between deformable objects by Larionov *et al.* [LFP21]. Our method largely follows this approach, but we make two crucial extensions that enable self collision detection for IMLS-based surfaces with robust continuous collision detection. Before we describe these extensions in detail, we start with a brief summary of IMLS-based contact modeling.

4.1. IMLS-based Contact Modeling

Let $\mathbf{x}^i$ denote the set of mesh vertices of body $\mathcal{B}^i$ with associated normals $\mathbf{n}^i$. The corresponding IMLS surface is defined implicitly as the zero-level set of the signed distance function

$$f[\mathbf{x}^i](\mathbf{x}) = \frac{\sum_j {\mathbf{n}_j^i}^\mathsf{T}(\mathbf{x}-\mathbf{x}_j^i)\phi_j^i(\mathbf{x})}{\sum_j \phi_j^i(\mathbf{x})}\,, \tag{4}$$

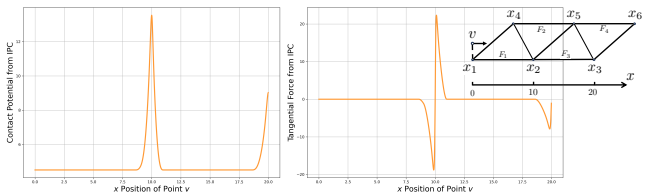


Figure 3: *Energy walls. We plot the contact potential (left) and the tangential force (right) from IPC as a single vertex slides on a plane discretized with triangle elements (top right). IPC allows for multiple per-primitive potentials to be active concurrently* (e.g. *at $x = x_2$ or $x = x_3$), leading to energy walls,* i.e., *local maxima in contact energy whose gradients resist tangential motion.*

where $\phi_j^i(\mathbf{x})$ are locally-supported, smoothly-clamped radial basis functions that can be evaluated at any spatial location $\mathbf{x}$ [Kol08].

To enforce contact constraints, we ask that all points $\mathbf{x}_i^1$ of a given object $\mathcal{B}^1$ must have non-negative distance from the IMLS surface of another body $\mathcal{B}^2$, i.e.,

$$f[\mathbf{x}^2](\mathbf{x}_i^1) \geq 0 \quad \forall i\,. \tag{5}$$

With sufficiently smooth basis functions ϕ, the signed distance function f is continuously differentiable and can therefore be used for constructing a smooth contact potential,

$$E_{\text{contact}} = \sum_{\mathbf{x}_i^1 \in \mathcal{B}^1} b\left(f[\mathbf{x}^2](\mathbf{x}_i^1)\right) + \sum_{\mathbf{x}_j^2 \in \mathcal{B}^2} b\left(f[\mathbf{x}^1](\mathbf{x}_j^2)\right)\,, \tag{6}$$

where the barrier function b is the same as in Eq. 3. While this formulation for *handling* contact is conceptually sound and simple, *detecting* collisions for IMLS surfaces proves more challenging as explained next.

4.2. Self Contact

Using the basic IMLS formulation as described above for handling self contact leads to an obvious problem: the distance of any vertex $\mathbf{x}_i \in \mathcal{B}^k$ to the level set of $\mathcal{B}^k$ is always (close to) zero. While this distance value is formally correct, it does not provide any indication of whether a given vertex $\mathbf{x}_i^k$ is close other parts of the surface. To rule out this trivial case, we must devise a way of *carving out* the region of the distance field around $\mathbf{x}_i^k$ when testing it for self-collisions. To this end, we observe that the Euclidean neighborhood

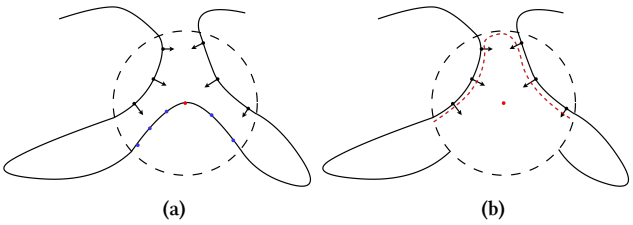


Figure 4: *Geodesic carving for self collision detection. (a): The Euclidean neighborhood of a given vertex (red) contains points that are geodesically close (blue) and points whose geodesic distance is large (black). (b): To detect self contact for the given vertex, we remove geodesically close points when constructing the IMLS representation (dashed curve).*

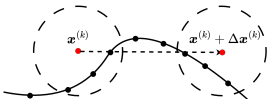
of a given vertex generally contains points that are geodesically close, and points whose geodesic distance to the vertex is large (see Fig.4). It is highly unlikely that $\mathbf{x}_i^k$ will intersect with its immediate geodesic neighborhood, since this would require excessive local curvature. Nevertheless, this immediate geodesic neighborhood is overriding the distance of $\mathbf{x}_i^k$ to other parts of the surface. To solve this problem, we propose to carve out the region of points around from the distance function that is used to test $\mathbf{x}_i^k$ for self collisions. To this end, we pre-compute pairs of vertices whose geodesic distance d_{gd} in the initial configuration is smaller than a threshold value ε_{gd}. We store this geodesic proximity data in a sparse matrix structure $\mathcal{G}$ such that $\mathcal{G}(i,j) = 1$ if $d_{gd}(\mathbf{x}_i, \mathbf{x}_j) < \varepsilon_{gd}$. Note that

$\mathcal{G}$ is computed only once at the start of simulation such that the runtime overhead is negligible. With these definitions in hand, the self contact potential for a given object $\mathcal{B}^k$ is constructed as

$$E_{\text{selfcontact}}(\mathcal{B}^k) = \sum_i b\left(f[\mathbf{x}^{k\setminus i}](\mathbf{x}_i^k)\right) . \tag{7}$$

where $\mathbf{x}^{k\setminus i} = \{\mathbf{x}_j^k | \mathcal{G}(i,j) = 0\}$. The modified penalty terms for self and external contacts are used in the time-stepping potential (1), which we minimize using Newton's method with intersection aware line search.

4.3. Continuous Collision Detection



While IMLS surfaces provide signed distance information, ensuring that distances are positive at the beginning and end of the time step is not sufficient, as tunneling might occur during a time step (see inset figure). To guarantee that no such intersections are introduced, we rely on continuous collision detection. Let $\mathbf{x}^{(l)}$ denote the current estimate for end-of-step positions in the l-th Newton iteration. Given a new search direction $\Delta\mathbf{x}^{(l)}$ from the linear solver, we must compute a step size $\alpha \in (0,1]$ to positions as $\mathbf{x}^{(l+1)} = \mathbf{x}^{(l)} + \alpha\Delta\mathbf{x}^{(l)}$. To prevent intersections, we must ensure that no vertex passes through any IMLS surface at any time in the interval $(0,\alpha]$.

Our continuous collision detection is divided into two phases: in the broad phase, we detect for each vertex $\mathbf{x}_i$ whether it approaches any other vertex $\mathbf{x}_j$ more closely than the kernel radius R. Each such test involves solving a simple quadratic root finding problem

$$||\mathbf{x}_i^{(l)} + t\Delta\mathbf{x}_i^{(l)} - \mathbf{x}_j^{(l)} - t\Delta\mathbf{x}_j^{(l)}||^2 = R^2 , \tag{8}$$

with respect to t. If the solutions t_1 and t_2 exist and overlap with the interval $(0,1]$, we add $\mathbf{x}_i$ to an active set $\mathcal{A}$ of potentially colliding vertices. To bootstrap the narrow phase solver, we also store minimum and maximum values for t_1 and t_2. After the broad phase has identified potentially colliding vertices and their intersection intervals, our narrow phase solver determines the maximum possible step size for each node. This requires finding the earliest root of its signed distance function within the interval determined by the broad phase detection. Mathematically speaking, this means finding the smallest root for the function

$$g(t) = \left(f[\mathbf{y}^{k\setminus p}(t)](\mathbf{y}_p(t))\right)^2 = 0 , \tag{9}$$

for each point $\mathbf{x}_p$ in the active set $\mathcal{A}$, where the distance f is evaluated based on the *new* vertex positions and $\mathbf{y}(t) = \mathbf{x}^{(l)} + t\Delta\mathbf{x}^{(l)}$ is the updated vertex positions when the step size is t . We cast this root finding problem as a 1-dimensional minimization problem for the *squared distance*, allowing us to leverage robust minimization algorithms. Since multiple roots can exist, we successively shrink the search interval (by a constant factor $c = 0.9$) until no further zero-distance solution can be found. The complete procedure is listed in Algorithm 1. It should be noted that solving the 1D minimization problems is very fast and easily parallelized. Indeed, the cost of these operations is negligible compared to the time required to assemble the Hessian of the total energy.

Our CCD guarantees penetration-free state of the simulation meshes. The broad phase stage can filter out states where no intersections can happen during this interval. Potential penetrations and tunneling will be caught in the narrow phase, which finds (if exists) the smallest t such that the unsigned distance $g(t)$ in Eq.9 evaluates to zero. This guarantees that there is no intersection in the interval $(0,t]$.

5. Results

We compare the behavior of IPC and our IMLS-based formulation on a set of experiments that include both qualitative and quantitative benchmarks for forward and inverse problems (Sec. 5.1). Finally, we show that our method can handle challenging scenarios with self-contact and friction (Sec. 5.3).

5.1. Comparison with IPC

For comparisons with IPC, we use the publicly available code base[†] from Li *et al.* [LFS*20], with modifications made solely to the scene configurations. The scene description files can be found in the supplemental material accompanying this paper.

5.1.1. Qualitative Comparison.

We begin our analysis by examining a forward simulation task where the spurious tangential forces generated by IPC become evident. To this end, we consider a deformable cube that is pressed down onto a frictionless surface and subjected to a constant horizontal force. The desired outcome is for the cube to move in the direction of the applied force. However, as indicated in Fig. 5 and shown in the accompanying video, IPC produces undesirable tangential forces making the cube grind to a halt. This behavior is explained by the presence of energy walls arising from the superposition of contact potentials (see also Fig. 3). These energy walls generate artificial friction-like forces that prevent the cube from moving along the desired trajectory. In contrast, our approach accurately reproduces the expected linear motion.

In the second example, we consider a cube falling onto a frustum whose apical face has the same area as the faces of the cube. The basal surface of the cube and the apical surface of the frustum are supposed to align perfectly. While our approach successfully captures this behavior, IPC produces lateral motion. This is due to the energy walls located at mesh edges leading to an unstable local energy maximum for perfect alignment, which induces tangential forces upon slight (numerical) perturbations.

As shown in the accompanying video, these artifacts are observed consistently for different hyper-parameters, including the threshold $\hat{d}$ that controls when contact potentials are activated.

5.1.2. Quantitative Comparison.

We now shift our focus to an inverse design example where spurious tangential forces prevent convergence to the desired solution.

[†] https://github.com/ipc-sim/IPC

ALGORITHM 1: Continuous Collision Detection for IMLS Surfaces

Data: Search direction $\Delta\mathbf{x}^{(k+1)}$, number of vertices n, kernel support radius R, scaling parameter $0 < c < 1$, threshold value ε
Result: Maximum step size α that is intersection-free
$\alpha \leftarrow 1$;
$\mathcal{A} \leftarrow \{\}$;
/* Broad Phase: Detect all IMLS surface points x_j whose kernels are intersected by trajectory of point x_i */
for $i \leftarrow 1$ **to** n **do**
  $\mathcal{A}[\mathbf{x}_i] \leftarrow \varnothing$;
  for $j \leftarrow 1$ **to** n **do**
    if $\mathcal{G}(i,j) == 0$ **then**
      $a \leftarrow ||\Delta\mathbf{x}_i - \Delta\mathbf{x}_j||^2$;
      $b \leftarrow -2\langle \Delta\mathbf{x}_i - \Delta\mathbf{x}_j, \mathbf{x}_i - \mathbf{x}_j \rangle$;
      $c \leftarrow ||\mathbf{x}_i - \mathbf{x}_j||^2 - R^2$;
      if $b^2 - 4ac \geq 0$ **then**
        $t_1 \leftarrow \frac{-b-\sqrt{b^2-4ac}}{2a}$;
        $t_2 \leftarrow \frac{-b+\sqrt{b^2-4ac}}{2a}$;
        $\mathcal{A}[\mathbf{x}_i] \leftarrow \mathcal{A}[\mathbf{x}_i] \cup ([t_1,t_2] \cap (0,1])$
      end
    end
  end
end
/* Narrow Phase: For each point in $\mathcal{A}$ with non-empty interval, find the first α that makes unsigned distance value zero */
for $p \leftarrow 1$ **to** n ***and*** $\mathcal{A}[\mathbf{x}_p] \neq \varnothing$ **do**
  /* Define 1D function to minimize, i.e., the unsigned distance of point $\mathbf{x}_p$ to IMLS surface as a function of t */
  $g(t) \leftarrow EvaluateIMLSSquared(\mathbf{x}^{(l)} + t \cdot \Delta\mathbf{x}^{(l)}, \mathbf{x}_p)$;
  $lo \leftarrow \mathcal{A}[\mathbf{p}].low$; $hi \leftarrow \mathcal{A}[\mathbf{p}].high$;
  $\alpha_p \leftarrow 1$;
  while *True* **do**
    if $lo \geq \alpha$ **then**
      break;
    end
    $hi \leftarrow min(hi, \alpha_p)$;
    $(t_{min}, f_{min}) \leftarrow 1DMinimizer(g(t), lo, hi)$;
    if $f_{min} \leq \varepsilon$ **then**
      $\alpha_p \leftarrow c \cdot t_{min}$;
    else
      break;
    end
  end
  $\alpha \leftarrow min(\alpha, \alpha_p)$
end

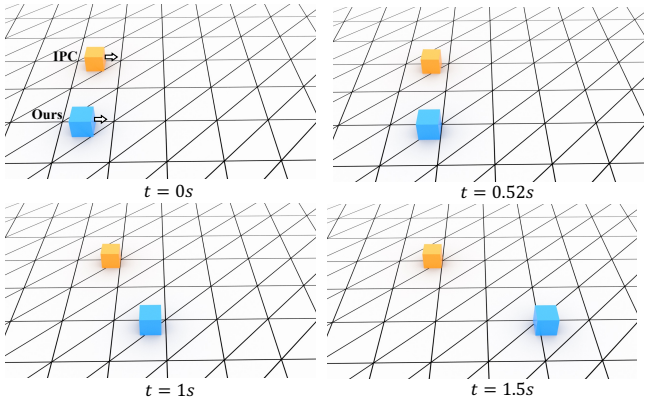


Figure 5: *Comparison with IPC. A deformable cube is pressed down onto a frictionless floor and subjected to a horizontal force (shown with arrows). Whereas the energy wall from IPC induces spurious tangential forces that put an early stop to the cube's trajectory (orange), our approach produces the expected linear motion (blue).*

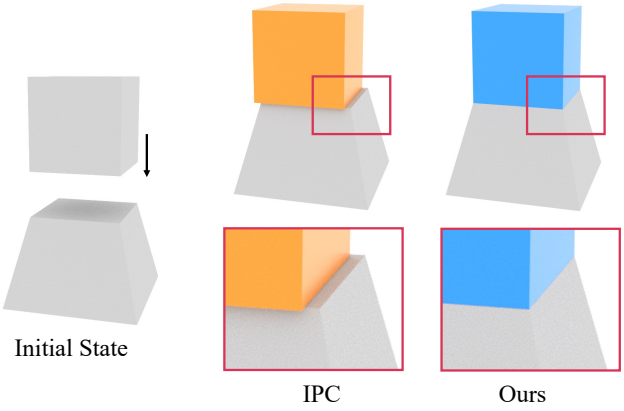
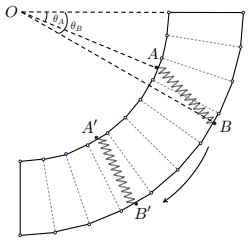


Figure 6: *Comparison with IPC. A cube falls onto a frustum under gravity (left). The superposition of per-element potentials in IPC induces lateral motion whereas our approach produces no such artifacts.*



As shown in the inset figure, we examine a case where two vertices slide along a rigid body in the form of a quarter annulus. Two vertices, denoted as A and B, are placed on the inner and outer tracks of the annulus and connected by a zero-length spring. In the absence of friction forces, moving vertex B should lead to an analogous motion for vertex A such as to minimize their distance. We first examine the accuracy of both approaches on the forward problem, where we seek the equilibrium states of A when B is moved to a prescribed location. The comparison between IPC

and our approach with the ground truth solution is shown in Fig. 7. As can be seen from this figure, our approach achieves accurate matching of the targets throughout the entire range of motion. In contrast, IPC exhibits poor matching behavior. By examining the insets in Fig. 7 (targets shown in red), we observe that this failure is once again due to spurious tangential forces stemming from energy walls. In particular, point A comes to a halt at its closest vertex and fails to reach its target location.

Next, we consider an inverse version of this problem, where the goal is to control the position of vertex B such that vertex A assumes a given target location A^*. We formulate this task as a constrained optimization problem where the constraint enforces static equilibrium,

$$\min_{\theta_B} \quad O(\mathbf{x}_B(\theta_B)) = ||\mathbf{x}_A(\theta_A)(\mathbf{x}_B(\theta_B)) - \mathbf{x}_A^*(\theta_A)||_2^2, \\ \text{s.t.} \quad \frac{\partial E}{\partial \mathbf{x}_A} = \frac{\partial E}{\partial \mathbf{x}_B} = \mathbf{0}\,, \tag{10}$$

where E is the total energy of the system and $\mathbf{x}_A^*(\theta_A)$ is the target location prescribed by a clock-wise rotation θ_A w.r.t. the horizontal axis in the image plane. To minimize (10) with respect to θ_B we use gradient descent and compute the simulation derivative $\frac{d\mathbf{x}_A}{d\mathbf{x}_B}$ using sensitivity analysis. The convergence plot shown in Fig. 8 reveals that, while IPC fails to converge to the desired accuracy, our approach converges rapidly. This discrepancy is again explained by the energy walls introduced at segment transitions when using IPC, which induce undesirable local minima that the gradient-based solver struggles to escape. We note that increasing mesh resolution for IPC changes the frequency of energy walls without removing them.

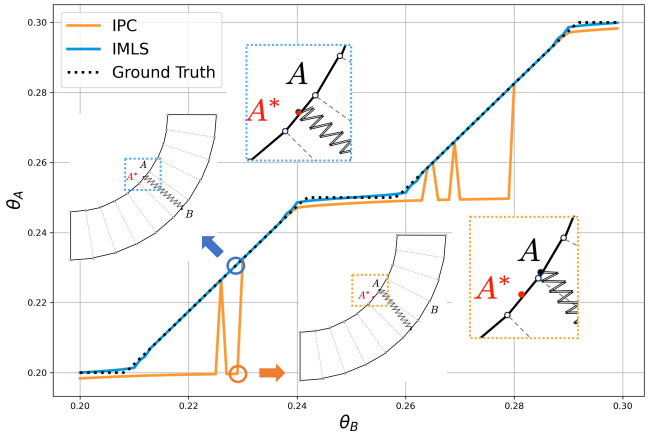


Figure 7: *Accuracy of the forward problem. Our approach (solid blue line) consistently arrives at locations that closely match the ground truth values (dotted line). IPC (solid orange line) struggles to overcome energy walls at segment transitions as illustrated in the inset.*

5.2. 3D Examples

We showcase the effectiveness of our approach through a series of 3D examples with deformable bodies represented using IMLS surfaces. In the first example (Fig. 9), an armadillo lands on a soft

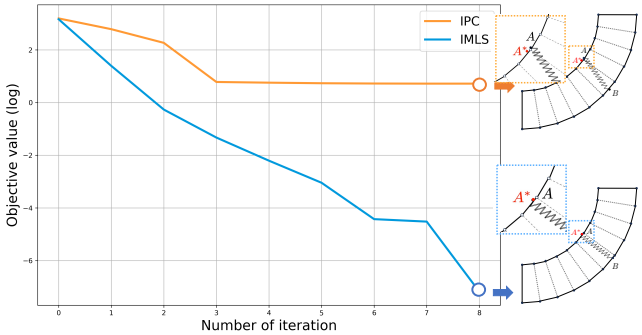


Figure 8: *Accuracy of the inverse problem. The convergence plots for IPC and IMLS are shown in orange and blue, respectively. IPC fails to find the true solution to this inverse design problem due to the presence of energy walls that introduce local optima. In contrast, our approach demonstrates robust convergence, reaching the solution in just 8 steps.*

plane. The second example (Fig. 10) shows a soft bunny passing through a narrow deformable torus. Finally, the interactions between complex geometries can be seen in Fig. 11, in which a deformable dolphin collides with a soft armadillo. As can be seen from these sequences, our approach reliably handles complex contact scenarios between deformable bodies undergoing large deformations. We refer to the accompanying video for the complete simulation sequences.

5.3. Self-Contact and Friction

We demonstrate the ability of our method to handle tight self-contacts with friction using two knot-tightening examples. In the first example, an elastic rope is tied into a single knot. In the second and more complex example, we tie a rope into a double knot. The knot is tightened by pulling the rope around a rigid cylinder. We refer to the accompanying video for the complete simulation sequences. Both of these examples involve frictional contacts with initial sliding that eventually converge into tight, static self-contacts. Our IMLS-based method handles these challenging scenarios robustly without any artifacts. The implementation detail of the frictional forces is included in Appx.B.

5.4. Erleben Tests

Erleben [Erl18] proposed a set of degenerate test cases for mesh-based contact problems. While our approach circumvents these issues by construction by leveraging smooth representations of the contact geometries, these unit tests nonetheless serve as a challenging benchmark. As can be seen from Fig. 14, our approach generates intersection-free trajectories in all cases.

5.5. Implementation Details

Our framework is implemented in C++ with Eigen [GJ*10] for linear algebra operations. The Eigen wrapper for the CHOLMOD solver [CDHR08] is used for solving linear systems. The total

Figure 9: *Free falling. A soft armadillo is falling onto a deformable plane. Both the plane and the armadillo's surface are represented using IMLS.*

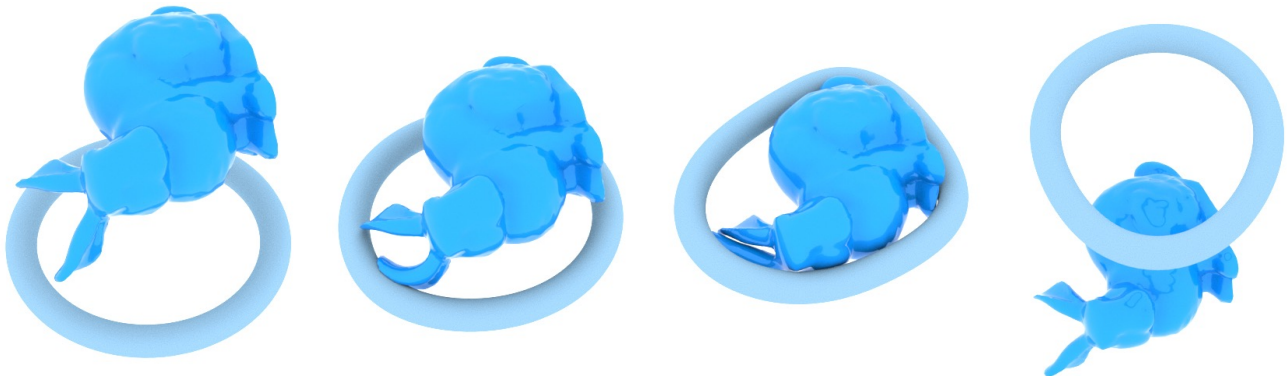


Figure 10: *Torus tunneling. A soft bunny falls under gravity through a torus, with both geometries represented by implicit moving least square surfaces for contact resolution.*

energy is minimized using Newton's method augmented with a back-tracking line search and adaptive regularization. We use linear tetrahedron elements for deformable solids and discrete shell elements [GHDS03] for surface meshes. A standard Neo-Hookean material [BW97] is used for all of our examples. We use the 1D minimizer provided by ROOT [BR97] in our continuous collision detection. The timings are obtained on a workstation with a *AMD Ryzen Threadripper PRO 5995WX* CPU and summarized in Table 1. More details about our robust IMLS implementation and comparisons with other formulations can be found in Appx.A.

5.6. Performance

From the timing information shown in Table 1, we observe that a single Newton iteration of IPC is generally faster than for our approach. This is explained by the fact that the contact Hessian of IPC is relatively simple for each primitive pair, while our distance computation involves a larger (and varying) number of points as well as their normals. On the other hand, Table 1 indicates that IPC generally requires more iterations to converge. We hypothesize that the slower convergence might be due to local minima introduced by the superposition of per-primitive potentials for IPC. As a result, our method is overall comparable to IPC in terms of run time.

Although in the narrow phase of our continuous collision detection, we perform an exhaustive quadratic test on all pairs of mesh vertices, it still saves a significant amount of time since the 1D minimizer requires additional overhead to set up. We observe a 5x to 20x speed up among our experiments depending on the ratio between the number of vertices in the active set $\mathcal{A}$ and the number of all vertices. However, they are both comparably much faster than the RIMLS hessian computation.

6. Conclusions

We have investigated a fundamental limitation of existing methods for handling contact between deformable bodies based on C^0-discretizations. Our analysis showed that spurious tangential forces generated by IPC-type approaches can significantly affect simulation outcomes and prevent inverse problems from converging to the correct solution. We furthermore showed that using smooth surface representations based on IMLS effectively resolves this problem, leading to robust behavior even for challenging contact scenarios and complex geometries.

6.1. Future Work

While we have demonstrated that our smooth representation enables robust convergence for inverse design on a 2D example, applications of our approach to inverse design tasks in 3D deserve further investigation.

Figure 11: *Catch-of-the-day sequence. Our approach robustly handles intricate contacts between deformable bodies with complex geometries.*

Table 1: *Statistics for 3D examples. We report the number of simulation degrees of freedom, the average computational time per Newton iteration, the maximum number of vertices under contact, and the average number of Newton iterations per time step.*

Example	# Vertices	Δt [s]	μ	Max # contact	Avg Time per Timestep [s] (IPC/Ours)	Avg Time per Newton Iter [s] (IPC/Ours)
Single Knot	10,184	0.01	0.05	1327	287.485 / 92.535	1.656 / 10.170
Double Knot	13,258	0.01	0.05	2325	101.162 / 98.731	3.167 / 9.943
Armadillo Plane	12,641	0.0025	0	1734	28.492 / 48.774	1.444 / 8.634
Bunny Torus	20,371	0.01	0	1600	195.674 / 135.858	3.478 / 15.189
Armadillo Dolphin	31,705	0.01	0	1015	312.992 / 393.324	4.530 / 14.671

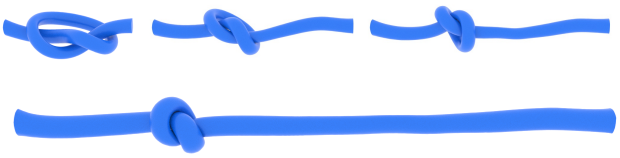


Figure 12: *Knot Tightening. We anchor one end of a knot while pulling on the opposite end, effectively tightening the knot without self-intersections.*

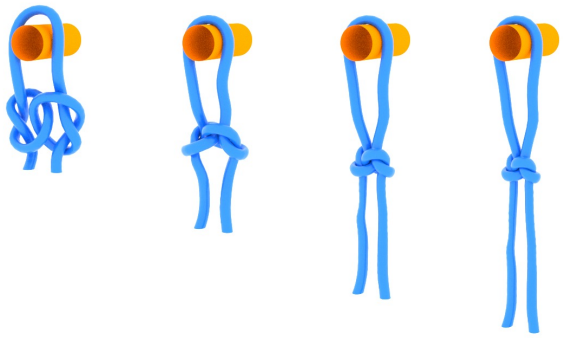


Figure 13: *Double-knots. We tighten two interweaving knots by pulling the open ends while having the upper end anchored to a stationary cylinder. The double knots fasten without self-intersections.*

We have introduced a continuous collision detection scheme that combines rapid quadratic distance tests for broad-phase detection with robust 1D minimization. While the broad-phase tests are significantly faster than solving narrow-phase minimization problems, we still have to perform pairwise checks between all nodes. Bounding volume hierarchies could further accelerate the average time for broad-phase detection. Our method uses the vertices of the input mesh for constructing a corresponding smooth implicit surface. While this is sufficient in most cases, keeping sampling points fixed during simulation might lead to under-sampled regions for extreme deformations. A possible improvement would be to develop adaptive sampling strategies that offer constant sample density while maintaining intersection-free states.

Acknowledgements

We are grateful to the anonymous reviewers for their valuable comments. This work was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 866480), and the Swiss National Science Foundation through SNF project grant 200021_200644. Open access funding provided by Eidgenossische Technische Hochschule Zurich.

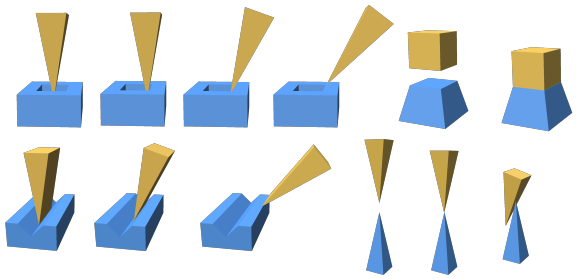


Figure 14: *Erleben tests. We perform simulations between elementary meshes with simple geometries and sharp features such as spikes, wedges, and cubes. Our IMLS-based method generates penetration-free trajectories for all challenging test cases.*

References

[AC91] ALART P., CURNIER A.: A mixed formulation for frictional contact problems prone to newton like solution methods. *Computer methods in applied mechanics and engineering 92*, 3 (1991), 353–375. 2

[ADVDI03] ALLIEZ P., DE VERDIRE E. C., DEVILLERS O., ISENBURG M.: Isotropic surface remeshing. In *2003 Shape Modeling International.* (2003), IEEE, pp. 49–58. 13

[Bar91] BARAFF D.: Coping with friction for non-penetrating rigid body simulation. *ACM SIGGRAPH computer graphics 25*, 4 (1991), 31–41. 2

[Ber89] BERNARDI C.: A new nonconforming approach to domain decomposition: the mortar element method. *Nonlinear partial equations and their applications* (1989). 2

[BFA02] BRIDSON R., FEDKIW R., ANDERSON J.: Robust treatment of collisions, contact and friction for cloth animation. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques* (2002), pp. 594–603. 2

[BR97] BRUN R., RADEMAKERS F.: Root—an object oriented data analysis framework. *Nuclear instruments and methods in physics research section A: accelerators, spectrometers, detectors and associated equipment 389*, 1-2 (1997), 81–86. 8

[BW97] BONET J., WOOD R. D.: *Nonlinear continuum mechanics for finite element analysis*. Cambridge university press, 1997. 8

[CDHR08] CHEN Y., DAVIS T. A., HAGER W. W., RAJAMANICKAM S.: Algorithm 887: Cholmod, supernodal sparse cholesky factorization and update/downdate. *ACM Transactions on Mathematical Software (TOMS) 35*, 3 (2008), 1–14. 7

[CPS09] COTTLE R. W., PANG J.-S., STONE R. E.: *The linear complementarity problem*. SIAM, 2009. 2

[Dav20] DAVIET G.: Simple and scalable frictional contacts for thin nodal objects. *ACM Transactions on Graphics (TOG) 39*, 4 (2020), 61–1. 2

[DBDB11] DAVIET G., BERTAILS-DESCOUBES F., BOISSIEUX L.: A hybrid iterative solver for robustly capturing coulomb friction in hair dynamics. *ACM Transactions on Graphics 30* (12 2011), 1–12. doi:10.1145/2070781.2024173. 2

[DLWW17] DE LORENZIS L., WRIGGERS P., WEISSENFELS C.: *Computational Contact Mechanics with the Finite Element Method*. John Wiley & Sons, Ltd, 2017, pp. 1–45. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119176817.ecm2033, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781119176817.ecm2033, doi:https://doi.org/10.1002/9781119176817.ecm2033. 2

[EDSR19] ELANDT R., DRUMWRIGHT E., SHERMAN M., RUINA A.: A pressure field model for fast, robust approximation of net contact force and moment between nominally rigid objects. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2019), IEEE, pp. 8238–8245. 2

[Erl18] ERLEBEN K.: Methodology for assessing mesh-based contact point methods. *ACM Transactions on Graphics (TOG) 37*, 3 (2018), 1–30. 7

[FL01] FISHER S., LIN M. C.: Deformed distance fields for simulation of non-penetrating flexible bodies. In *Computer Animation and Simulation 2001: Proceedings of the Eurographics Workshop in Manchester, UK, September 2–3, 2001* (2001), Springer, pp. 99–111. 3

[FLS*21] FERGUSON Z., LI M., SCHNEIDER T., GIL-URETA F., LANGLOIS T., JIANG C., ZORIN D., KAUFMAN D. M., PANOZZO D.: Intersection-free rigid body dynamics. *ACM Transactions on Graphics 40*, 4 (2021). 2

[FPRJ00] FRISKEN S. F., PERRY R. N., ROCKWOOD A. P., JONES T. R.: Adaptively sampled distance fields: A general representation of shape for computer graphics. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (2000), pp. 249–254. 2

[Gas93] GASCUEL M.-P.: An implicit formulation for precise contact modeling between flexible solids. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques* (1993), pp. 313–320. 3

[GHDS03] GRINSPUN E., HIRANI A. N., DESBRUN M., SCHRÖDER P.: Discrete shells. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2003), Citeseer, pp. 62–67. 8

[GHZ*20] GEILINGER M., HAHN D., ZEHNDER J., BÄCHER M., THOMASZEWSKI B., COROS S.: Add: Analytically differentiable dynamics for multi-body systems with frictional contact. *ACM Transactions on Graphics (TOG) 39*, 6 (2020), 1–15. 2

[GJ*10] GUENNEBAUD G., JACOB B., ET AL.: Eigen v3. http://eigen.tuxfamily.org, 2010. 7

[GRP91] GOYAL S., RUINA A., PAPADOPOULOS J.: Planar sliding with dry friction part 2. dynamics of motion. *Wear 143*, 2 (1991), 331–352. 2

[HTS*76] HUGHES T. J., TAYLOR R. L., SACKMAN J. L., CURNIER A., KANOKNUKULCHAI W.: A finite element method for a class of contact-impact problems. *Computer methods in applied mechanics and engineering 8*, 3 (1976), 249–276. 2, 3

[HVS*09] HARMON D., VOUGA E., SMITH B., TAMSTORF R., GRINSPUN E.: Asynchronous contact mechanics. *ACM Transactions on Graphics 28* (7 2009). doi:10.1145/1531326.1531393. 2

[HVTG08] HARMON D., VOUGA E., TAMSTORF R., GRINSPUN E.: Robust treatment of simultaneous collisions. In *ACM SIGGRAPH 2008 papers*. 2008, pp. 1–4. 2

[JBS06] JONES M. W., BAERENTZEN J. A., SRAMEK M.: 3d distance fields: A survey of techniques and applications. *IEEE Transactions on visualization and Computer Graphics 12*, 4 (2006), 581–599. 2

[JM92] JEAN M., MOREAU J. J.: Unilaterality and dry friction in the dynamics of rigid body collections. In *1st Contact Mechanics International Symposium* (1992), pp. 31–48. 2

[KDBB17] KOSCHIER D., DEUL C., BRAND M., BENDER J.: An hp-adaptive discretization algorithm for signed distance field generation. *IEEE transactions on visualization and computer graphics 23*, 10 (2017), 2208–2221. 2

[KE22] KIM T., EBERLE D.: Dynamic deformables: Implementation and production practicalities (now with code!). In *ACM SIGGRAPH 2022 Courses* (New York, NY, USA, 2022), SIGGRAPH '22, Association for Computing Machinery. URL: https://doi.org/10.1145/3532720.3535628, doi:10.1145/3532720.3535628. 3

[KEP05] KAUFMAN D. M., EDMUNDS T., PAI D. K.: Fast frictional dynamics for rigid bodies. In *ACM SIGGRAPH 2005 Papers* (New York, NY, USA, 2005), SIGGRAPH '05, Association for Computing Machinery, p. 946–956. URL: https://doi.org/10.1145/1186822.1073295, doi:10.1145/1186822.1073295. 2

[Kol08] KOLLURI R.: Provably good moving least squares. *ACM Transactions on Algorithms (TALG) 4*, 2 (2008), 1–25. 3, 4

[KP03] KRY P. G., PAI D. K.: Continuous contact simulation for smooth surfaces. *ACM Transactions on Graphics (TOG) 22*, 1 (2003), 106–129. 2

[KSJP08] KAUFMAN D. M., SUEDA S., JAMES D. L., PAI D. K.: Staggered projections for frictional contact in multibody systems. In *ACM SIGGRAPH Asia 2008 papers*. 2008, pp. 1–11. 2

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

[LDN*18] LI J., DAVIET G., NARAIN R., BERTAILS-DESCOUBES F., OVERBY M., BROWN G. E., BOISSIEUX L.: An implicit frictional contact solver for adaptive cloth simulation. *ACM Transactions on Graphics (TOG) 37*, 4 (2018), 1–15. 2

[LDW*22] LI Y., DU T., WU K., XU J., MATUSIK W.: Diffcloth: Differentiable cloth simulation with dry frictional contact. *ACM Transactions on Graphics (TOG) 42*, 1 (2022), 1–20. 2

[Lev04] LEVIN D.: Mesh-independent surface interpolation. In *Geometric modeling for scientific visualization* (2004), Springer, pp. 37–49. 3

[LFP21] LARIONOV E., FAN Y., PAI D. K.: Frictional contact on smooth elastic solids. *ACM Trans. Graph. 40*, 2 (Apr. 2021). URL: https://doi.org/10.1145/3446663, doi:10.1145/3446663. 2, 3, 4

[LFS*20] LI M., FERGUSON Z. A. H., SCHNEIDER T., LANGLOIS T., ZORIN D., PANOZZO D., JIANG C., KAUFMAN D. M.: Incremental potential contact: Intersection- and inversion-free, large-deformation dynamics. *ACM Transactions on Graphics 39* (7 2020). doi:10.1145/3386569.3392425. 1, 2, 5

[LKJ21] LI M., KAUFMAN D. M., JIANG C.: Codimensional incremental potential contact. *ACM Transactions on Graphics 40* (7 2021). doi:10.1145/3450626.3459767. 2, 3

[LKL*22] LAN L., KAUFMAN D. M., LI M., JIANG C., YANG Y.: Affine body dynamics: Fast, stable & intersection-free simulation of stiff materials. *arXiv preprint arXiv:2201.10022* (2022). 2

[LYK*21] LAN L., YANG Y., KAUFMAN D., YAO J., LI M., JIANG C.: Medial ipc: accelerated incremental potential contact with medial elastics. *ACM Transactions on Graphics 40*, 4 (2021). 2

[MC95] MIRTICH B., CANNY J.: Impulse-based simulation of rigid bodies. In *Proceedings of the 1995 symposium on Interactive 3D graphics* (1995), pp. 181–ff. 2

[MEM*19] MACKLIN M., ERLEBEN K., MÜLLER M., CHENTANEZ N., JESCHKE S., MAKOVIYCHUK V.: Non-smooth newton methods for deformable multi-body dynamics. *ACM Transactions on Graphics (TOG) 38*, 5 (2019), 1–20. 2

[MEM*20] MACKLIN M., ERLEBEN K., MÜLLER M., CHENTANEZ N., JESCHKE S., CORSE Z.: Local optimization for robust signed distance field collision. *Proceedings of the ACM on Computer Graphics and Interactive Techniques 3*, 1 (2020), 1–17. 3

[MMP88] MADAY Y., MAVRIPLIS C., PATERA A.: *Nonconforming mortar element methods: Application to spectral discretizations*. Tech. rep., 1988. 2

[Mor11] MOREAU J. J.: On unilateral constraints, friction and plasticity. *New variational techniques in mathematical physics* (2011), 171–322. 2

[MTGG11] MARTIN S., THOMASZEWSKI B., GRINSPUN E., GROSS M.: Example-based elastic materials. In *ACM SIGGRAPH 2011 Papers* (New York, NY, USA, 2011), SIGGRAPH '11, Association for Computing Machinery. URL: https://doi.org/10.1145/1964921.1964967, doi:10.1145/1964921.1964967. 3

[MW88] MOORE M., WILHELMS J.: Collision detection and response for computer animation. In *Proceedings of the 15th annual conference on Computer graphics and interactive techniques* (1988), pp. 289–298. 2

[MZS*11] MCADAMS A., ZHU Y., SELLE A., EMPEY M., TAMSTORF R., TERAN J., SIFAKIS E.: Efficient elasticity for character skinning with contact and collisions. In *ACM SIGGRAPH 2011 papers*. 2011, pp. 1–12. 3

[OGG09] OZTIRELI C., GUENNEBAUD G., GROSS M.: Feature preserving point set surfaces based on non-linear kernel regression. In *Computer graphics forum* (2009), vol. 28, pp. 493–501. 3, 11, 12

[OTSG09] OTADUY M. A., TAMSTORF R., STEINEMANN D., GROSS M.: Implicit contact handling for deformable objects. *Computer Graphics Forum 28* (4 2009), 559–568. URL: https://onlinelibrary.wiley.com/doi/full/10.1111/j.1467-8659.2009.01396.xhttps://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8659.2009.01396.xhttps://onlinelibrary.wiley.com/doi/10.1111/j.1467-8659.2009.01396.x, doi:10.1111/J.1467-8659.2009.01396.X. 2

[PHW19] POPOV V., HESS M., WILLERT E.: *Handbook of Contact Mechanics: Exact Solutions of Axisymmetric Contact Problems*. Springer Berlin Heidelberg, 2019. URL: https://books.google.ch/books?id=OrhbvwEACAAJ. 2

[SOS04] SHEN C., O'BRIEN J. F., SHEWCHUK J. R.: Interpolating and approximating implicit surfaces from polygon soup. In *ACM SIGGRAPH 2004 Papers*. 2004, pp. 896–904. 2

[TO02] TURK G., O'BRIEN J. F.: Modelling with implicit surfaces that interpolate. *ACM Transactions on Graphics (TOG) 21*, 4 (2002), 855–873. 2

[VBG*13] VAILLANT R., BARTHE L., GUENNEBAUD G., CANI M.-P., ROHMER D., WYVILL B., GOURMEL O., PAULIN M.: Implicit skinning: Real-time skin deformation with contact modeling. *ACM Transactions on Graphics (TOG) 32*, 4 (2013), 1–12. 3

[VGB*14] VAILLANT R., GUENNEBAUD G., BARTHE L., WYVILL B., CANI M.-P.: Robust iso-surface tracking for interactive character skinning. *ACM Transactions on Graphics (TOG) 33*, 6 (2014), 1–11. 3

[VJ19] VERSCHOOR M., JALBA A. C.: Efficient and accurate collision response for elastically deformable models. *ACM Transactions on Graphics (TOG) 38*, 2 (2019), 1–20. 2

[WZ04] WRIGGERS P., ZAVARISE G.: *Computational Contact Mechanics*. John Wiley & Sons, Ltd, 2004, ch. 6. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/0470091355.ecm033, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/0470091355.ecm033, doi:https://doi.org/10.1002/0470091355.ecm033. 2

[ZD09] ZAVARISE G., DE LORENZIS L.: The node-to-segment algorithm for 2d frictionless contact: Classical formulation and special cases. *Computer Methods in Applied Mechanics and Engineering 198*, 41 (2009), 3428–3451. URL: https://www.sciencedirect.com/science/article/pii/S0045782509002278, doi:https://doi.org/10.1016/j.cma.2009.06.022. 2, 3

Appendix A

Robust Formulation of IMLS Surface

In order to address the limitations of Larionov *et al.*'s method described in 4.1, we integrate a robust IMLS formulation from Öztireli *et al.* [OGG09] into our contact handling pipeline. The idea

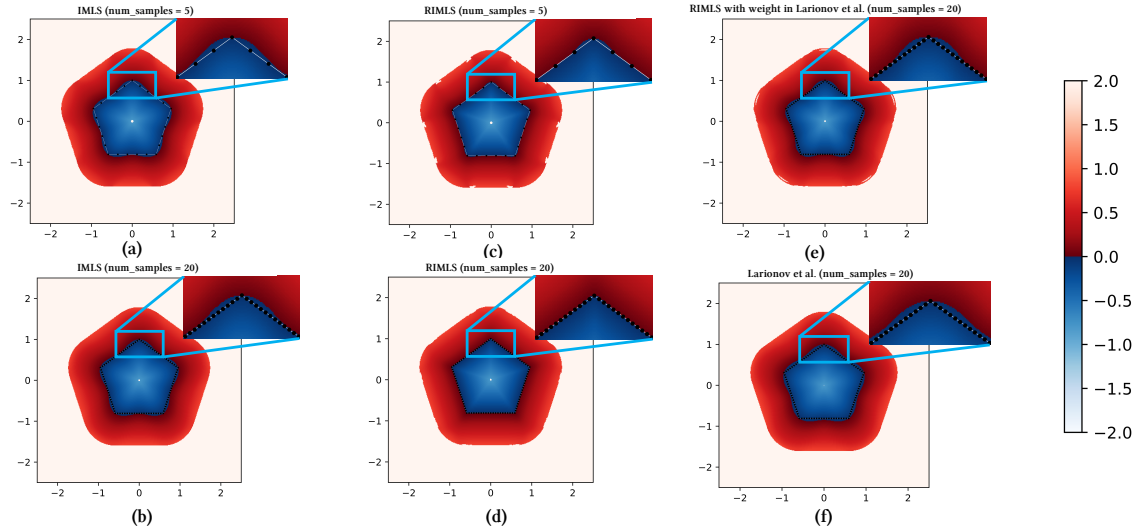


Figure 15: *Surface reconstruction from different formulations. We sample points (black dots) from a pentagon (white lines) with different resolutions (5 points or 20 points per side) and the signed distance values in the space are displayed with the color bar shown on the right. (a) and (b) show the reconstruction result from the basic IMLS formulation in Eq.4, where we can see the corners are overly smoothed and there is no significant improvement even when we increase the sample density. The robust formulation in Eq.11 generates sharper results even with 5 sample points (c) and it improves to negligible difference from the underlying pentagon with more samples (d). Larionov* et al. *use a different kernel function ϕ_i that generates the worst result in this example even with 20 samples (e) and Robust formulation (f).*

of this robust formulation is to treat the normal as the source of outliers of the IMLS surface. The sample points around a sharp corner have very different normals and more weight should be assigned to those who have a closer normal to the implicit surface. However, the signed distance function $f_{[\mathbf{p}]}(\mathbf{x})$ is unknown and an iterative least squared approach is adopted to refine $f_{[\mathbf{p}]}(\mathbf{x})$. This leads to a new formulation and the signed distance at k^{th} iteration is defined as

$$f^k_{[\mathbf{p}]}(\mathbf{x}) = \frac{\sum \mathbf{n}_i^{\mathsf{T}}(\mathbf{p}_i - \mathbf{x}^{\mathsf{T}})\phi_i(\mathbf{x})w(r_i^{k-1})w_n(\Delta\mathbf{n}_i^{k-1})}{\sum \phi_i(\mathbf{x})w(r_i^{k-1})w_n(\Delta\mathbf{n}_i^{k-1})}, \tag{11}$$

where $r_i^{k-1} = ||f^{k-1}(\mathbf{x}) - \mathbf{n}_i^T(\mathbf{x} - \mathbf{p}_i)||$ is the residual of the estimated function value and $\Delta\mathbf{n}_i^{k-1} = ||\nabla f^{k-1}(\mathbf{x}) - \mathbf{n}_i||$ measures the difference between the normal at sample point $\mathbf{n}_i$ and the gradient of the IMLS surface at query point $\mathbf{x}$.

Implementation of Robust IMLS Surface

We stick to the choice of parameters from Öztireli *et al.* [OGG09] to have the weighting functions $\phi_i(\mathbf{x}) = \left(1 - \frac{||\mathbf{x}-\mathbf{x}_i||^2}{R^2}\right)^4$, $w(r) = e^{-\left(\frac{r}{\sigma_r R}\right)^2}$ and $w_n(\nabla f(\mathbf{x})) = e^{-\left(\frac{||\nabla f(\mathbf{x}) - \mathbf{n}_i||}{\sigma_n}\right)^2}$, where R is the search radius, σ_r is set to 0.5 and σ_n ranges from 0.5 to 1.5 depending on how much sharpness the user wants. We also set the number of iterations k to be 1 since Öztireli *et al.* observe that only 1 iteration of optimization can already create sharp enough reconstructed surfaces. This also means the surface gradient $\nabla f^k_{[\mathbf{p}]}(\mathbf{x})$ is only evaluated at the initial iteration where the additional weighting terms $w(r)$ and $w_n(\nabla f(\mathbf{x}))$ is set to 1. The gradient $\nabla f^0_{[\mathbf{p}]}(\mathbf{x})$ is therefore has a simpler analytical form

$$\nabla f^0_{[\mathbf{p}]}(\mathbf{x}) = \frac{\sum \mathbf{n}_i\phi_i(\mathbf{x}) + \sum \nabla\phi_i(\mathbf{x})(\mathbf{n}_i^{\mathsf{T}}(\mathbf{x} - \mathbf{p}_i) - f^0(x))}{\sum \phi_i(\mathbf{x})}. \tag{12}$$

Since we use mesh vertex as sample points, the normal associated with each point is the per-vertex normal, which is the average of the per-face normal for all the incident triangles. It should also be noted that they are computed in the deformed configuration in order to support large deformation. In order to leverage the efficiency of second-order solvers for unconstrained optimization problems, we need to compute the first and second-order derivatives of the signed distance function with respect to both sample points $\mathbf{x}_i$ and the evaluated point $\mathbf{x}$. This is done analytically without any auto-differential involvement to increase performance.

Comparison Among Different Implicit Surface Formulations

We also compared the reconstructed surfaces from Larionov *et al.*'s method, the IMLS method and the Robust IMLS method. The results can be found in Fig.15 where we sample points from a pentagon with different resolutions and fix the search radius. Comparing (a) and (c) we can see that Robust IMLS demonstrates better preservation of sharp features even with a small number of sample points. IMLS tends to smooth the sharp corners and such behavior doesn't improve even if you increase the resolution (see (c)). RIMLS, on the other hand, improves quite a lot when there are more sample points. This suggests that simply increasing cannot help solve the issues with the IMLS method. Larionov *et al.* use a different weighting function $\phi_i^{cubic}(\mathbf{x}) = 1 - 3\left(\frac{||\mathbf{x}-\mathbf{x}_i||_2}{R}\right)^2 +$

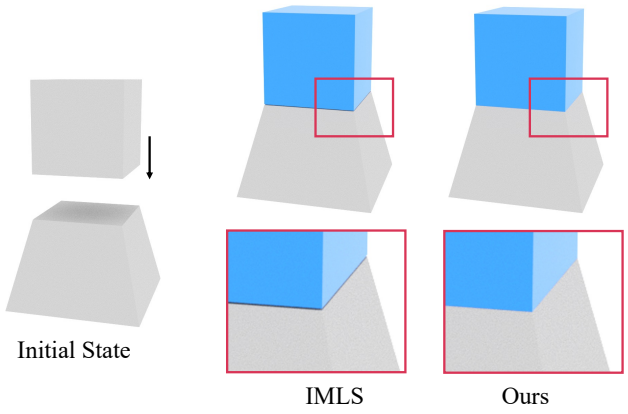


Figure 16: *Comparison with IMLS. The IMLS reconstruction failed to capture sharp features such as corners and edges. Consequently, the overly smoothed reconstruction leads to floating artifacts. We leverage the robust IMLS formulation to preserve these sharp features.*

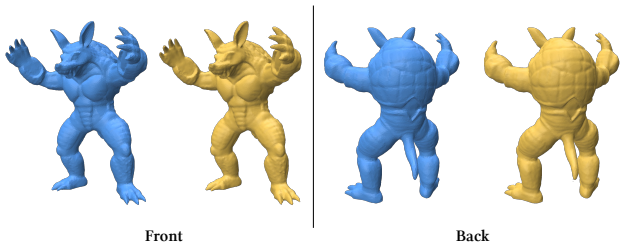


Figure 17: *Input mesh (blue) and its R-IMLS zero level set (yellow). As can be seen from the two viewpoints, using mesh vertices as sample points enables robust reconstruction of the input geometry.*

$2\left(\frac{||\mathbf{x}-\mathbf{x}_i||_2}{R}\right)^3$ for faster computation. However, this results in the worst reconstructed surfaces even with a large number of sample points (see (e) and (f)). The reconstructed surface doesn't converge to the true pentagon under refinement at the flat region. This could result in severe artifacts and make it harder to do continuous collision detections.

The inability to capture sharp features of IMLS formulation can lead to visual artifacts. As shown in the cube falling test in Fig. 16, IMLS leads to the over-smoothing of the frustum edges, thus, creating the artificial gap between the two contacting geometry. In contrast, using R-IMLS successfully addresses this limitation.

Point Sampling

We perform uniform sampling for the simple meshes in our experiments (such as the cube in Fig. 5 and the inverse design in Sec. 5.1.2). For the more complex 3D examples, we preprocess the simulation mesh using the isotropic surface remeshing algorithm [ADVDI03] such that the variance of the edge lengths is minimized. This allows us to directly use the mesh vertices as the sampling points and achieve good results in practice. While our approach can benefit from advanced sampling strategies, as we show in Fig. 17, using mesh vertices as sampling points can already faithfully represent the input geometry. We choose the search radius R to be 2 to 3 times the largest edge length such that there are enough points in the neighborhood.

As illustrated in Fig. 15, the sampling resolution will have an effect on the reconstruction quality of the RIMLS surfaces. Coarse sampling in general results in overly smoothed features. However, increasing sample points alone cannot capture sharp features (a,b,e,f). We therefore resort to RIMLS formulation which allows for detailed reconstruction with comparatively fewer samples.

Appendix B

Friction

We incorporate the friction force in our pipeline by using a lagged model proposed by IPC. This approach leverages the normal force from the preceding time step to calculate the friction force in the current time step. A mollifier is used to ensure a smooth transition from static friction to dynamic friction. To integrate this friction force into an optimization-based integrator, a friction potential is formulated on this basis. For one vertex i, given the contact force magnitude $\lambda_i^{(t-1)}$, direction $\mathbf{T}_i^{(t-1)}$, and position $\mathbf{x}_i^{(t-1)}$, the smooth friction potential is computed as

$$E_{\text{friction}}^{(t)} = \begin{cases} \mu\lambda_i^{(t-1)}\left(\frac{x^6}{\varepsilon^5} - \frac{3x^5}{\varepsilon^4} + \frac{5x^4}{2\varepsilon^3}\right) & x < \varepsilon \\ \mu\lambda_i^{(t-1)}\left(x - \frac{\varepsilon}{2}\right) & x \geq \varepsilon \end{cases}, \tag{13}$$

where $x = || \left(\mathbf{x}_i^{(t)} - \mathbf{x}_i^{(t-1)}\right) - \mathbf{T}_i^{(t-1)\mathsf{T}} \left(\mathbf{x}_i^{(t)} - \mathbf{x}_i^{(t-1)}\right) \mathbf{T}_i^{(t-1)} ||$ is the displacement on the tangential plane at the contact point and ε represents a threshold parameter used to distinguish between static and dynamic friction.