

# Geometric Contact Potential

ZIZHOU HUANG, New York University, USA  
MAXWELL PAIK, New York University, USA  
ZACHARY FERGUSON, Massachusetts Institute of Technology, USA  
DANIELE PANOZZO, New York University, USA  
DENIS ZORIN, New York University, USA

Barrier potentials gained popularity as a means for robust contact handling in physical modeling and for modeling self-avoiding shapes. The key to the success of these approaches is adherence to geometric constraints, i.e., avoiding intersections, which are the cause of most robustness problems in complex deformation simulation with contact. However, existing barrier-potential methods may lead to spurious forces and imperfect satisfaction of the geometric constraints. They may have strong resolution dependence, requiring careful adaptation of the potential parameters to the object discretizations.

We present a systematic derivation of a continuum potential defined for smooth and piecewise smooth surfaces, starting from identifying a set of natural requirements for contact potentials, including the barrier property, locality, differentiable dependence on shape, and absence of forces in rest configurations. Our potential is formulated independently of surface discretization and addresses the shortcomings of existing potential-based methods while retaining their advantages.

We present a discretization of our potential that is a drop-in replacement for the potential used in the incremental potential contact formulation [Li et al. 2020], and compare its behavior to other potential formulations, demonstrating that it has the expected behavior. The presented formulation connects existing barrier approaches, as all recent existing methods can be viewed as a variation of the presented potential, and lays a foundation for developing alternative (e.g., higher-order) versions.

CCS Concepts: • **Computing methodologies** → **Physical simulation**.

Additional Key Words and Phrases: Finite element method, Elastodynamics, Contact dynamics

## ACM Reference Format:

Zizhou Huang, Maxwell Paik, Zachary Ferguson, Daniele Panozzo, and Denis Zorin. 2025. Geometric Contact Potential. *ACM Trans. Graph.* 44, 4 (August 2025), 24 pages. <https://doi.org/10.1145/3731142>

## 1 INTRODUCTION

Contact modeling is a critical component of simulation tools for many domains, including computer graphics, robotics, mechanical design, and biomedical engineering. A representative (friction-less)

Authors' addresses: Zizhou Huang, [zizhou@nyu.edu](mailto:zizhou@nyu.edu), New York University, USA; Maxwell Paik, [mp6569@nyu.edu](mailto:mp6569@nyu.edu), New York University, USA; Zachary Ferguson, [zy.fergus@gmail.com](mailto:zy.fergus@gmail.com), Massachusetts Institute of Technology, USA; Daniele Panozzo, [panozzo@nyu.edu](mailto:panozzo@nyu.edu), New York University, USA; Denis Zorin, [dzorin@cs.nyu.edu](mailto:dzorin@cs.nyu.edu), New York University, USA.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
0730-0301/2025/8-ART \$15.00  
<https://doi.org/10.1145/3731142>

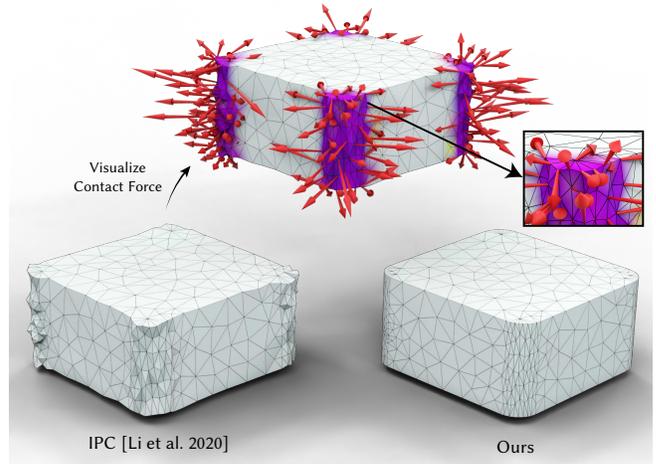


Fig. 1. We introduce a novel geometric barrier potential satisfying a set of natural properties. None of the other potentials in the literature satisfy all these properties simultaneously, leading to inaccurate results, penetrations, or other undesired artifacts. In this example, we show that the contact potential introduced in [Li et al. 2020] is not zero at the rest pose when the potential extent is larger than the length of an edge, introducing spurious forces (top) and deformation (bottom left). Our potential is zero in the rest pose by construction (bottom right), as this is one of the natural properties of a geometric contact potential.

contact problem can be viewed as a geometrically constrained optimization:

$$\min_u E(u), \text{ subject to } g(u) \geq 0$$

where  $u = u(x)$  is a deformation of a surface parametrized by  $x$ , and the function  $g$  is entirely geometric, measuring the *distance to contact* for a solution  $u$ .  $E(u)$  may be physical energy for static problems, a time integrator formulated in a variational way for dynamic problems, or a deformation objective for geometric modeling.

Overwhelmingly, robustness issues in contact modeling are related to handling these geometric constraints; once a contact constraint is violated, it may be very difficult for the solver to recover. For this reason, a natural approach to increasing the robustness of contact solvers is to ensure that *all solution updates*, including intermediate steps of iterations in a solver, are *contact-free*, i.e., satisfy  $g(x) > 0$ .

Solvers based on barrier potentials are particularly suitable to maintain a contact-free state, and in recent years, were demonstrated to handle complex, large-scale, and long-duration contact problems reliably. Conceptually, these solvers are based on viewing the contact

problem as an equivalent unconstrained optimization [Kane et al. 1999]:

$$\min_u E(u) + b(u)$$

where  $b(u)$  is an ideal barrier, equal to  $\infty$  for configurations with interpenetration, and 0 for contact-free.

If we take  $b(u)$  to be a potential that increases smoothly to infinity as  $u$  approaches a contact configuration, then the problem is converted to a smooth problem that can be solved with contact-free iterates, e.g., using a nonlinear solver with line search, equipped with continuous collision detection (CCD), as it is done in [Li et al. 2020].

All existing contact barrier potentials are defined based on aggregating repulsion terms between pairs of points or, in the discrete case, between pairs of elements (e.g., a face and a point). The key question then becomes how to choose the strength of repulsion between two points so that it is high when these points are close to contact, vanishes when these points are far, and has a number of other desirable properties (e.g., depends smoothly on the deformation, has controlled locality), as discussed in Section 4.

One important difficulty is distinguishing true contact, arising from different objects or parts of the surface of the same object moving toward each other, and points that remain close because they are close on the undeformed rest shape (Figure 2). Answering this question for a general class of surfaces requires careful geometric consideration. In Section 2, we review how it is addressed for existing potential types and problems associated with these approaches (possibility of geometric constraint violations, spurious forces, and strongly discretization-dependent behavior).

We undertake a systematic derivation of a *continuum barrier potential* defined as a surface integral, and its discretization, which retains key features of previously proposed approaches, IPC in particular, but satisfies several additional, natural requirements, elaborated in Section 5:

- **Resolution independence:** The potential definition is independent of discretization.
- **Finiteness:** The total barrier potential over the surface is defined and finite for any collection of piecewise-smooth surfaces not in contact. (The use of piecewise-smooth, rather than smooth, surfaces is critical for many applications, as detailed in Section 5.2).
- **Barrier:** The potential grows to infinity as the objects in the simulation approach (self-)contact, with respect to a suitably defined *distance-to-contact*.
- **Localization:** The potential has a localization parameter  $\hat{d}$ , which may vary over the object surface. The potential vanishes if the objects are further away than  $\hat{d}$  from contact, with respect to the same measure of contact.
- **No spurious forces:** In the undeformed configuration, the potential is zero.

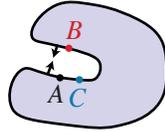


Fig. 2. Contact potentials must distinguish points nearing contact (A and B) from nearby non-contacting points (A and C).

- **Differentiability:** The potential depends smoothly on the surface configuration (e.g., for piecewise-linear surfaces, on mesh vertex positions).

The foundation of our approach is a *new geometric definition of the distance-to-contact* that allows us to define contact potentials for arbitrary piecewise-smooth surfaces without assuming a discretization, and in a way that all requirements are satisfied, leading to artifact-free simulation with a flexible choice of potential locality. The key features of our approach include:

- A systematic way of defining points close to contact for smooth and piecewise-smooth surfaces, based on *interaction sets*.
- An approach for defining a contact potential for piecewise-smooth surfaces that meets the requirements enumerated above.
- A discrete version of this potential for piecewise-linear surfaces that satisfies the requirements enumerated above and has similar efficiency to standard IPC.

We demonstrate that the new formulation eliminates spurious forces inherent in other formulations, converges to a limit under refinement, and decouples the barrier extent from the discretization choice, offering needed flexibility for applications such as shape optimization.

## 2 BARRIER POTENTIALS

To motivate the proposed approach, we consider several representative versions of contact potentials proposed in the past. We focus on four problems that each construction addresses differently:

- **Finiteness.** How the potentials ensure that true contact points are distinguished from close material points; without this distinction, they would always be infinite.
- **Spurious repulsion.** Relatedly, how spurious forces for close material points are eliminated or reduced; if applied to the wrong pairs of points, these forces have a large impact.
- **Mesh dependence.** How these potentials behave under remeshing or refinement; mesh adaptation, refinement, or remeshing are common, especially for large/complex deformations.
- **Discretization properties.** If a potential is defined in continuous form, even if it prevents contact exactly, its discretization may do so only approximately.

### 2.1 The IPC barrier potential [Li et al. 2020]

This potential is defined in a purely discrete way, based on vertex-face and edge-edge repulsion. To ensure it is *finite*, interactions of a face with its vertices and between edges sharing one vertex are excluded. Note that as the surface is refined, the strength of interaction increases, as the excluded part gets smaller.

*Spurious forces* are partly avoided by defining a potential that does not extend beyond the shortest mesh edge length in the undeformed state. However, during deformation, the edge length can change arbitrarily, and spurious forces may easily appear under compression (Section 6.1). The IPC potential is strongly *mesh dependent*: e.g., simply refining locally forces the maximum extent of the potential to be decreased by a large factor, resulting in a change in the results of simulation/deformation.

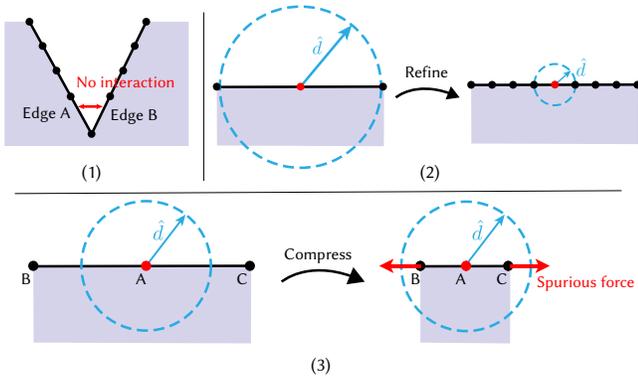


Fig. 3. IPC potential. The extent of the potential is shown in blue. (1) IPC potential is finite, as interactions between edges sharing vertices, and faces with their vertices are dropped. (2) Refinement forces the maximal potential extent  $\hat{d}$  to decrease. (3) Spurious forces (red arrows) arise if the surface is compressed horizontally and nearby vertices are closer than the potential extent  $\hat{d}$ .

A modified formulation of this potential is introduced in Li et al. [2023], with a convergent discretization which, however, requires refinement in both spatial discretization and the potential extent  $\hat{d}$ . We provide further discussion and comparisons in Section 6 and Table 1.

## 2.2 Surface/volume double-integral potentials [Kamensky et al. 2018; Sauer and De Lorenzis 2013]

Barrier potentials defined in continuum setting in [Sauer and De Lorenzis 2013] are adapted to self-contact in [Alaydin et al. 2021; Kamensky et al. 2018]. To include self-contact, *finiteness* is achieved by excluding a fixed-size area or sphere in the undeformed state from the integral. However, this exclusion eliminates the guarantee that under all deformations there is no penetration (Figure 4(1)).

To avoid *spurious repulsion*, the potential is localized to a distance less than that of the self-contact exclusion distance. In this way, in the undeformed state, there is no repulsion. However, if compression is high enough, spurious forces will appear (Figure 4(2)).

As these potentials are defined as integrals on smooth surfaces and then discretized, there is a high degree of *mesh independence*, especially for finer meshes, but may lead to self-intersection for extreme deformations, as described above. Furthermore, as pointed out in [Kamensky et al. 2018], the discretization based on pairwise potentials between quadrature points for extreme deformation may also result in failure to satisfy contact constraints (Figure 4(3)).

## 2.3 Repulsive shells [Sassen et al. 2024a]

Building on, e.g., [Strzelecki and von der Mosel 2013], [Sassen et al. 2024a] introduced a repulsive potential, whose minimization can be used to define self-avoiding surfaces. An important feature of this potential in our context is that it handles the problem of near-point interactions gracefully for smooth surfaces, as the potential depends not only on positions but also on the normals. It vanishes in the limit of points approaching a common position on a smooth surface,

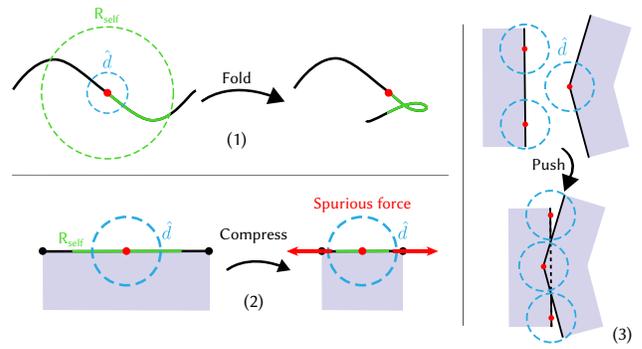


Fig. 4. Surface barrier of [Kamensky et al. 2018] (1) The potential has a self-contact exclusion zone (green), outside the potential extent (blue). For extreme deformations (folding), this allows self-intersections. (2) Extreme compression may push points into the potential zone, leading to spurious forces (red arrows) (3) In quadrature point-to-point discretization, the surface may not remain contact-free, a point may “push through” between other [Kamensky et al. 2018].

ensuring *finiteness* for smooth surfaces, but not for surfaces with sharp features, as the normal alignment of close points is needed. As these potentials are designed to ensure that surfaces minimizing these are  $C^1$  (i.e., precisely to eliminate any possible sharp features), they use high inverse powers of distance, resulting in divergence for piecewise smooth surfaces.

As the interaction potential for any curved surface does not vanish, *spurious repulsion* is present for an undeformed surface (in the context of the target application of [Sassen et al. 2024a], defining surfaces as extrema of the repulsive potentials, this repulsion is not spurious). As the discretization used in [Sassen et al. 2024a] is based on sampling at quadrature point pairs similar to [Sauer and De Lorenzis 2013], it also does not guarantee in general that the contact constraints are satisfied exactly for the discretization, which is particularly important for coarse discretizations.

We further note that all these potentials depend smoothly on the surface deformation; IPC and [Kamensky et al. 2018] are local, and [Sassen et al. 2024a] is global, although it decays rapidly.

## 2.4 Barriers based on the gap functions [Wriggers and Laursen 2006]

We briefly mention the possibility of constructing barriers based on gap functions, which can be viewed as solving a standard inequality-constrained contact problem formulation using an interior point method (e.g., [Kloosterman et al. 2001], although this work shifts the barrier towards the interior of an object).

Most contact papers use one of three gap functions: distance along the normal direction (DND) [Belgacem et al. 1998; Benson and Hallquist 1990; Christensen et al. 1998; Hübner and Wohlmuth 2006; Kloosterman et al. 2001; Laursen and Love 2002; Popp et al. 2012; Taylor and Papadopoulos 1993; Vola et al. 1998; Wang et al. 2024], closest point (CP) [Alart and Curnier 1991; Armero and Petócz 1998; Carpenter et al. 1991; Pietrzak and Curnier 1999; Simo and Laursen

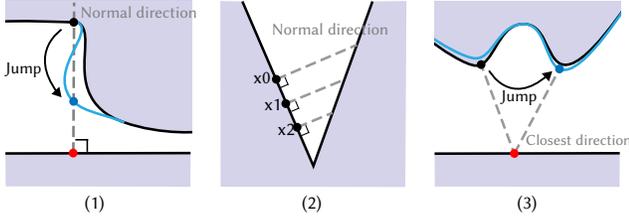


Fig. 5. (1) The DND gap function is discontinuous when the deformation progresses from the black to the blue curve. (2) The DND gap function can be arbitrarily small for p.w. smooth surfaces as  $x_i$  approaches the sharp corner in a continuous setting. (3) The CP gap function changes non-smoothly, as the closest point switches discontinuously (from black to blue).

1992; Taylor and Wriggers 1999; Temizer et al. 2014, 2012; Wriggers 1995], and contact-pair closest distance (CPCD) [Chen et al. 2024; Daviet 2020; Huang et al. 2024a; Kim and Eberle 2022; Li et al. 2020, 2023; Macklin et al. 2020, 2019; Otaduy et al. 2009; Razon et al. 2023; Sassen et al. 2024b; Shen et al. 2024; Verschoor and Jalba 2019], the latter defined directly on the discrete surface (e.g., vertex-face and edge-edge distances for PL surfaces). Most importantly, the CP gap function does not allow for self-contact without explicit exclusion of nearby points, and, while continuous, is not differentiable with respect to surface deformations (Figure 5(3)). [Konyukhov and Schweizerhof 2013] provides a generalized procedure for closest point projection that addresses the issue of uniqueness; however, they don’t handle self-contact or complex geometries. The distance along the normal direction resolves the self-interaction problem, but is discontinuous, hindering the application of efficient numerical methods (Figure 5(1)). While the closest point distance is well-defined for distinct objects if they are not smooth, the distance along the normal direction requires a well-defined surface normal to exclude contact precisely, making it non-trivial to apply for piecewise smooth surfaces.

A potential obtained by applying a logarithmic or inverse power function to these gap functions inherits the properties of these functions.

Our construction addresses these problems by design, without the need for parameter tuning per simulation: for sufficiently close points, the potential vanishes exactly; no amount of compression leads to potential activation, unless surface points get close to contact; it is defined in a continuum form, so its behavior for different meshes is consistent; its strength is chosen so that it is sufficiently strong for smooth surfaces and does not blow up for piecewise smooth. Finally, we choose the discretization in a way that guarantees that the discretized surfaces are contact-free.

### 3 RELATED WORK

We present a detailed analysis of most of the references discussed here in the supplementary material, summarizing the features of the algorithms and their performance. We briefly summarize the key aspects here.

There are many textbooks and reviews on contact mechanics; we focus on the aspects of algorithms most directly related to robustness. [Li et al. 2020] supplementary material presents in-depth

testing of representative implementations of a number of techniques on a common benchmark, providing empirical evidence of the higher robustness of the barrier-based approach.

#### 3.1 Barrier/interior-point methods

We group barrier and interior-point methods together, as these are closely related. These methods, while known for a long time (cf. [Wriggers and Laursen 2006]), only recently became popular in the context of complex deformable contact. In an early work, [Christensen et al. 1998], a direct application of an interior-point method with a non-localized logarithmic potential is compared to a non-smooth Newton method, primarily from the performance perspective. A modified barrier method, with the barrier domain shifted towards the surface interior, and thus not preventing inadmissible configurations, is described in [Kloosterman et al. 2001].

[Sauer and De Lorenzis 2013] introduced a systematic description of contact methods as surface potentials of several types, including potentials based on repulsion depending on the distance between points, which was extended in [Kamensky et al. 2018] and [Alaydin et al. 2021] to handle self-contact (see Section 2). [Temizer et al. 2014] uses an interior point method to solve the contact problem, but in the presented weak formulation, exact enforcement of the contact is avoided. In [Wang et al. 2024], the Ipopt interior point software is used to solve the mortar contact formulation.

The barrier method introduced in Li et al. [2020], to which the discrete version our method is closest, has been used and extended by numerous follow-up works to include support for codimensional elements [Li et al. 2021], rigid/affine body dynamics [Ferguson et al. 2021; Lan et al. 2022], medial elastics [Lan et al. 2021], solid-fluid interactions [Xie et al. 2023], higher-order finite element analysis [Ferguson et al. 2023], etc. Most recently, Huang et al. [2024a]; Shen et al. [2024] proposed accelerated preconditioned solvers that are amenable to GPU data parallelism to accelerate [Li et al. 2020]. These methods sacrifice the exact guarantees of the original IPC to improve performance.

Du et al. [2024], observes that “Node-to-Segment” methods including IPC [Li et al. 2020]) produce spurious tangential contact forces (cf. [Puso and Laursen 2004]). To address this, Du et al. [2024] proposes to use a fully  $C^1$ -continuous surface representation in combination with piecewise-linear meshes, computing distances used in the contact potential between the discretization nodes and smooth surface, extending the method of Larionov et al. [2021]. Similar to [Kamensky et al. 2018], Du et al. [2024] supports self-collision by ignoring the collision of a vertex with its geodesic neighborhood. The method does not guarantee contact-free iterations, a feature critical for IPC’s robustness [Li et al. 2020]. Our work is complementary; we do not address this type of artifact in our discretization, a higher-order discretization of our continuum formulation is needed for this.

Another possible continuum barrier approach to avoiding intersections is to use *tangent-point energies* [Buck and Orloff 1995; Strzelecki and von der Mosel 2013] designed to produce self-avoiding smooth surfaces as minimizers. These potentials are global and need to be localized for efficiency and to avoid artificial long-range forces in a physical simulation context. These methods were used in a

shape modeling context in [Sassen et al. 2024b; Yu et al. 2021] with limitations discussed in Section 2.

In addition to the barrier methods mentioned above, most notably, Harmon et al. [2009] and Vouga et al. [2011] utilize a set of layered discrete penalty barriers that grow unbounded as the configuration approaches contact. However, this incremental construction makes it unsuitable for optimization-based implicit time integration and therefore requires small time steps for stability. Other notable works include Kaldor et al. [2008], which simulates knitted cloth at the yarn level where yarn-yarn collisions are handled through a continuous potential integrating a barrier function over two disjoint spline segments. This approach, similar to surface potentials described above, ignores self-collisions within a single segment and between neighboring segments. Similarly, this work uses quadrature points fixed in the parameter space to compute interactions.

### 3.2 Other approaches

The simplest contact response is *penalty-based* [Armero and Petöcz 1998; Benson and Hallquist 1990; Chen et al. 2024; Kim and Eberle 2022; Laursen and Love 2002; Meier et al. 2017; Temizer et al. 2012; Wriggers 1995]. Similarly to the barrier methods, the problem is converted to an unconstrained optimization problem with an extra term, but in contrast to barrier methods, a penalty potential is added only if penetration occurs, and no forces are introduced otherwise; by design, penalty methods allow constraint violation, leading to difficult-to-untangle configurations for complex contact.

*Augmented Lagrangian methods*, widely used in engineering [Alart and Curnier 1991; Daviet 2020; Fernandez et al. 2020; Hiermeier et al. 2018; Konyukhov and Schweizerhof 2013; Pietrzak and Curnier 1999; Puso et al. 2008; Puso and Laursen 2004; Puso et al. 2024; Simo and Laursen 1992; Wriggers 1995], considerably improve convergence and numerical stability compared to the penalty methods, but also have to go through infeasible configurations before a solution is obtained. Accurate *mortar methods* [Belgacem et al. 1998; Hübner and Wohlmuth 2006; Puso and Laursen 2004] are typically formulated in the augmented Lagrangian form, and do not enforce geometric constraints exactly.

*“Active set” methods* (which we define broadly as methods that rely in an essential way on identifying a contact set for which the geometric constraints may become equalities, and imposing equality constraints for these v.s. more specific class of active-set optimization methods) include [Belytschko and Neal 1991; Carpenter et al. 1991; Hübner and Wohlmuth 2006; Popp et al. 2012; Taylor and Papadopoulos 1993; Vola et al. 1998]. As these methods seek solutions exactly at the boundary of the admissible solution space, having intermediate infeasible solutions is hard to avoid. Many also approximate the exact geometric constraints in the discretization.

Methods based on reducing the problem to *sequential quadratic programming (SQP)* or *linear complementarity problems (LCP/MLCP)* are widely used [Deuffhard et al. 2008; Kaufman et al. 2008; Otaduy et al. 2009; Verschoor and Jalba 2019; Youett et al. 2019]. In these approaches the non-linear constrained problem is typically converted to a problem with linear constraints. As a consequence, these algorithms go through inadmissible configurations as a part of the solution process, which affects robustness. Several works aim to

have contact-free states at the end of each time step in a dynamic simulation by resolving interpenetration, but there is no guarantee that this can be achieved [Kaufman et al. 2008; Otaduy et al. 2009].

## 4 CONTACT POTENTIAL REQUIREMENTS

We consider a collection of deformable objects defined on a material domain  $\text{Int}(\Omega)$ , whose boundary is  $\Omega$ . We assume that normals on  $\Omega$  are chosen to point outwards.  $\text{Int}(\Omega)$  may have multiple connected components (Figure 6).

An admissible deformation  $f : \text{Int}(\Omega) \mapsto \mathbb{R}^n$ ,  $n = 2, 3$  is non-injective on the boundary only, i.e.,  $f$  is injective in the interior of the domain, but may have *contact points*  $x \neq y$  on the boundary  $\Omega$ , for which  $f(x) = f(y)$ . We assume there is an initial contact-free map  $f_0$  corresponding to the undeformed shape of the object (typically an identity map if  $\text{Int}(\Omega) \subset \mathbb{R}^n$ ).

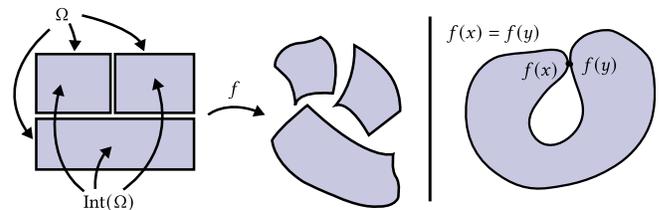


Fig. 6. A collection of objects is transformed using a deformation  $f$  (left), which can cause the objects to intersect at a contact point (right).

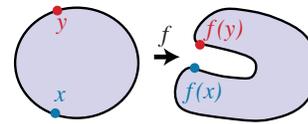


Fig. 7. The distance from  $x$  to real contact point  $y$  may be arbitrarily close, depending on the deformation  $f$ .

We say that an admissible  $f$  is *in contact* if it has contact points, otherwise, we call it *contact-free*. We assume that for a point  $x \in \Omega$ , we have a metric,  $d_c(x, f)$ , of how far it is from being a contact point. Defining this metric is a key aspect of our construction, and is necessary to formulate our contact requirements.

While for rigid objects the distance to the closest point on *another* object is adequate to use as  $d_c(x, f)$ , as points on the same object cannot move into contact, this is not the case for deformable objects. Due to the possibility of self-contact, we cannot exclude points on the same object and, in this case, there are always points on  $\Omega$  arbitrarily close to  $x$  in Euclidean distance that need to be considered far from being in contact with  $x$ . At the same time, an arbitrarily small, if measured by Euclidean distance, perturbation of a surface may create an actual contact, so the distance from  $x$  to a real contact point  $y$  along the surface can be arbitrarily small (Figure 7).

For smooth surfaces, the solution is to take the normals at the points into account. For points in contact, the normals are pointing in opposite directions. So if we include the distance between  $n(x)$  and the normals at other points in the definition of  $d_c(x, f)$ , we

Table 1. The table columns correspond to six comparison characteristics: (1) Support self-contact, (2) guarantee intersection-free with conservative CCD, (3) is defined in the continuous setting and discretized after, or directly in the discrete setting, (4) may have spurious rest forces, (5) support types of geometry discretization. We refer to Section 3 and Section 6 for detailed discussions. A more complete comparison with other methods can be found in the supplementary material.

	Self-contact	Intersection-free	Discrete/Continuous	Localization	Spurious rest forces	Surface types
Sauer and De Lorenzis [2013]	Not descr.	No	C	Yes	Yes	PL
Christensen et al. [1998]	No	No	D	Yes	No	PL
Temizer et al. [2014]	No	No	C	Yes	No	smooth
Kamensky et al. [2018]	Yes	No	C	Yes	Yes	smooth
Li et al. [2020]	Yes	Yes	D	Yes	Yes	PL
Alaydin et al. [2021]	Yes	No	C	Yes	Yes	smooth
Li et al. [2023]	Yes	Yes	C	Yes	Yes	PL
Wang et al. [2024]	Not descr.	No	C	Yes	No	PL
Sassen et al. [2024b]	Yes	No	C	No	Yes	PL
Huang et al. [2024a]	Yes	Yes	D	Yes	Yes	PL
Shen et al. [2024]	Yes	No	D	Yes	Yes	PL
Du et al. [2024]	Yes	No	D	Yes	Yes	PL + Implicit
Ours	Yes	Yes	C	Yes	No	p.w. smooth

can eliminate the nearby points unless the curvature is high, and folds over onto itself, as discussed more precisely in Section 5.1. For piecewise smooth surfaces, the situation is more complex.

Before describing our approach to the distance to contact, we make the properties we stated in the introduction more precise.

**Requirement 1 (Finiteness).** A total contact potential integral

$$\Psi(f) := \iint_{\Omega \times \Omega} \psi_\epsilon(x, y; f) \, dx dy$$

is finite, i.e., the pointwise potential  $\psi_\epsilon(x, y; f)$  is integrable on  $\Omega \times \Omega$  for piecewise-smooth surfaces  $\Omega$  for any  $f$  not in contact.  $\psi_\epsilon(x, y; f)$  is repulsive, meaning it must be monotonically non-increasing with  $d_c(x, f)$  for any  $f$ .

**Requirement 2 (Barrier).** For a time-varying  $f_t$ ,  $\Psi(f_t)$  increases to infinity for  $t \rightarrow t_0$  if the distance to contact  $d_c(x, f_t)$  goes to zero for any  $x$ . Combined with incremental potential time-stepping and continuous collision detection (CCD), this can be used to guarantee that all configurations remain contact-free.

**Requirement 3 (No spurious forces).** Suppose  $f_0$  is the initial configuration of the simulation. For any transformation  $f$  differing from  $f_0$  by a rigid transformation, both  $\Psi(f)$  and  $\nabla \Psi(f)$  are zero. This requirement is necessary to ensure that the potential does not create artificial forces that would cause motion/deformations if no external forces are acting on the object.

**Requirement 4 (Localization).**  $\psi_\epsilon(x, y; f)$  has a locality parameter  $\epsilon_{\text{trg}} > 0$  ( $\hat{d}$  in the notation of Li et al. [2020]), with no restrictions on its magnitude, on which it depends at least continuously, and the potential vanishes if  $d_c(x, f) > \epsilon_{\text{trg}}$ . If we solve a sequence of problems with decreasing  $\epsilon_{\text{trg}}$ , we approach a solution of the standard inequality-constrained formulation of contact problems.

**Requirement 5 (Differentiability).** If  $f$  is defined by a finite number of parameters (in the simplest case, vertex positions of a mesh) then  $\Psi(f)$  depends differentiably, and piecewise twice differentiably, on the parameters of  $f$ . Then the potential leads to a force with piecewise continuous Jacobian, allowing for second-order methods for implicit time-stepping.

Importantly, the formulation of most of these properties, except for the first one, can be applied to either the continuum potential or its discretization. The robustness advantages of using a barrier disappear if these are enforced only approximately in the discrete case. For this reason, we add an additional requirement:

**Requirement 6 (Discretization).** The discretization of the contact potential satisfies requirements *Finiteness to Differentiability* (rather than just in the limit of refinement).

In particular, this requires constructing a special type of quadrature that ensures the barrier property.

**Remark.** The *Finiteness* requires the integral to be well-defined. Even in the absence of contact, the potential may not be bounded (see Section 4.2 in the supplemental document). In the discrete case, the integral in *Finiteness* is replaced by a finite sum.

## 5 FORMULATION

Our approach to defining contact potentials is conveniently formulated in terms of *interaction sets*  $C(x, f)$ . This set is a subset of  $\Omega$  away from  $x$ , for which the barrier potential at  $x$  does not vanish.

If we define the pointwise contact potential as

$$\psi_\epsilon(x, y; f) := \gamma(x, y) p_{\epsilon(x)}(\|f(y) - f(x)\|) \quad (1)$$

where  $\gamma$  is a factor vanishing outside of  $C(x, f)$ ,  $p_{\epsilon(x)}$  vanishes at a distance  $\epsilon(x)$ . Note that  $\gamma$  and  $p_{\epsilon(x)}$  have different supports, the intersection of which forms the support of  $\psi_\epsilon$ . If  $p_{\epsilon(x)}$  tends to infinity sufficiently quickly as the argument tends to zero, and  $\epsilon(x)$  is chosen so that the potential vanishes for the rest shape, then such interaction sets satisfy Requirements 2 to 4. In addition, Requirement 5 may be satisfied by choosing appropriately smooth  $p_{\epsilon(x)}$  and  $\gamma$ .

The definition involves four main components: the interaction set, the adaptive locality parameter  $\epsilon(x)$ , the barrier potential  $p_{\epsilon(x)}(x, y)$ , and the factor  $\gamma(x, y)$ , supported on  $C(x, f)$ . As it will be clear from the discussion in the next section,  $\gamma(x, y)$  can be viewed as a *direction localization factor*, supplementing the distance localization, to address the problems described in Section 2.

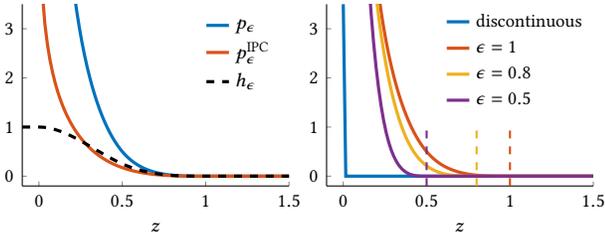


Fig. 8. **Barriers.** Left: plot of the cubic spline  $h_\epsilon(z)$  and barrier function  $p_\epsilon(z) = h_\epsilon(z)/z^{n-1}$  for  $\epsilon = 1$ . We also include a plot of the log barrier  $p_\epsilon^{\text{IPC}}(z)$  of Li et al. [2020] for comparison. Right: our barrier improves approximation to the discontinuous function as  $\epsilon$  goes to 0.

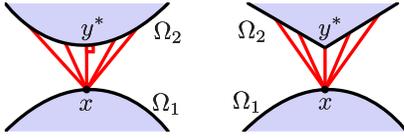


Fig. 9. Contact points as a local minimum of distance on the surface.

We start with contact handling for smooth deformable surfaces, then generalize to piecewise smooth surfaces, in each case explaining how these three potential components are defined.

### 5.1 Deformable smooth surfaces

We first consider the case where both  $\Omega$  and  $f(\Omega)$  are smooth surfaces, to describe the main ideas without considering the many cases needed for piecewise smooth surfaces.

Two points of an admissible smooth surface  $f$  are in contact, if (a) they coincide in space,  $f(x) = f(y)$ , and (b) the normals of the point have the opposite orientation,  $n(x) = -n(y)$ . We define the interaction sets to include points close to contact, i.e., points for which  $f(x)$  and  $n(x)$  are both close to  $f(y)$  and  $-n(y)$ , respectively, and exclude points for which at least one of these requirements is not satisfied. Taking the normals into account addresses the problem of distinguishing between close material points and points close to contact.

Rather than measuring the distances between normals, we use a different approach that generalizes naturally to non-smooth points of piecewise smooth surfaces but behaves similarly in the smooth case.

In Figure 9, observe that for a fixed point  $x \in \Omega_1$ , if we consider  $d_x(y, f) = \|f(y) - f(x)\|$  with  $y$  varying over  $\Omega_2$ ,  $d_x(y, f)$  has a local minimum at  $y^*$ , the closest point on the surface.

If we also consider points  $y \in \Omega_1$ , i.e., include self-contact,  $d_x(y)$  has a local minimum at  $x$ , but not at any point in a neighborhood of  $x$ . This leads to the following idea, which is the key to our generalization to piecewise smooth surfaces, as it does not use normals or surface smoothness:

*Define the interaction set  $C(x, f)$  as points that are close to being local minima of the distance function  $d_x(y, f)$  distinct from  $x$ .*

Additionally, we require that the vector  $f(y) - f(x)$  points towards the exterior of the surface at  $f(x)$  and towards the interior at  $f(y)$ , which is critical for handling thin objects. Next, we derive expressions based on this to define  $C(x, f)$  and the factor  $\gamma(x, y)$ .

**5.1.1 Interaction sets.** For smooth surfaces, the local minima of the distance function from  $f(x)$  to  $f(y)$  for a contact-free  $f$  satisfy the condition

$$\Phi^m(x, y) := \|(f(y) - f(x))_+ \times n(y)\| = 0, \quad (2)$$

where  $(\cdot)_+$  denotes normalization to a unit vector. The inequality  $\Phi^m(x, y) \leq \alpha$ , identifies the set of points close to satisfying this condition. We refer to this as *the local minimum constraint*.

To exclude points for which the vector  $(f(y) - f(x))$  points towards object exterior at  $y$  define

$$\Phi^e(x, y) := -n(y) \cdot (f(y) - f(x))_+ \quad (3)$$

Then  $f(x)$  is on the side of the outward pointing normal from  $f(y)$ , if  $\Phi^e(x, y) > 0$ . We refer to this as *the exterior direction constraint*.

This leads to the following definition of an interaction set for a smooth surface:

**Definition 1.** The interaction set for a smooth deformation  $f(x)$  with normals  $n(x)$  is defined as

$$C(x, f) := \left\{ y \in \Omega \mid \begin{array}{l} \Phi^m(x, y) \leq \alpha, \Phi^e(x, y) \geq -\alpha, \\ \Phi^m(y, x) \leq \alpha, \Phi^e(y, x) \geq -\alpha \end{array} \right\} \quad (4)$$

where the local minimum constraint  $\Phi^m$  and exterior direction constraint  $\Phi^e$  are given by (2) and (3), where  $\alpha$  satisfies  $0 < \alpha < 1$ . We show the impact of  $\Phi^m$  and  $\Phi^e$  in Figure 10.

The choice of  $\alpha$  is discussed in Section 6; the method works for any choice of  $\alpha < 1$ , the only impact is on performance, as larger values include more points into the contact set. If we define the *distance to contact*  $d_c(x, f)$  of a point  $x$  under deformation  $f$  as the minimum (Euclidean) distance to the interaction set of  $x$ , which is positive for contact-free surfaces,  $C(x, f)$  has the following properties:

- If  $f$  is in contact,  $C(x, f)$  contains all contact points of  $x$ ;
- If  $f$  is not in contact,  $d_c(x, f)$  is positive;
- $d_c(x, f_t)$  tends to zero, as  $t \rightarrow t_0$  when a contact-free, time-dependent  $f_t$  approaches contact at  $t_0$ .

Please see the supplementary for details.

**Remark.** In the case of smooth surfaces  $(\Phi^m)^2 = 1 - (\Phi^e)^2$ , so if  $\Phi^e$  is close to 1,  $|\Phi^m|$  is close to zero, i.e., the condition of  $\Phi^m$  can be ensured by the choice of  $\Phi^e$  in Equation (4). However, the relationship is more complex for piecewise smooth surfaces since both normal and tangent are not unique at joints of multiple faces, so we will treat these separately.

The complete potential for smooth surfaces is given by

$$\Psi(f) = \iint_{(x, y) \in \Omega^2} \gamma^S(x, y) p_{\epsilon(x)}(\|f(x) - f(y)\|) dx dy. \quad (5)$$

where  $\gamma^S(x, y)$  and  $\epsilon(x)$  are defined based on the interaction sets.

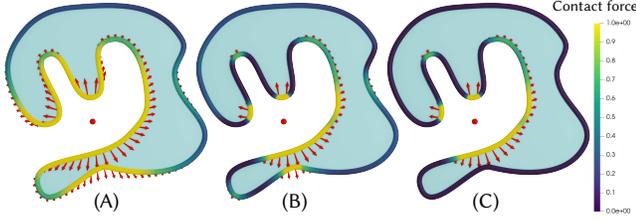


Fig. 10. Contact forces (arrows) and potential distribution on the 2D object surface with respect to the red dot. From left to right are: (A) The potential in Li et al. [2020] distributes spherically around the red dot regardless of the surface shape. (B) With local minimum constraints, only surfaces that are close to the local minima or maxima of distance from the red dot have high values. (C) With both local minimum and exterior direction constraints, only surfaces on the closer side of the volumetric object have high values.

**5.1.2 Barrier function.** We use  $p_\epsilon(z) := h_\epsilon(z)z^{-p}$  (Figure 8), where the choice of the parameter  $p$  is discussed in the supplementary document, we use  $p = n - 1$ , where  $n = 2, 3$  is the dimension of the scene. Define  $h_\epsilon(z) := \frac{3}{2}B^3(2z/\epsilon)$ , a cubic  $C^2$  spline basis function with support  $|z| \leq \epsilon$  (Appendix C). Then  $h_\epsilon$  vanishes if  $d_c(x, f)$  exceeds  $\epsilon$ , and as a consequence,  $\psi_\epsilon(x, y; f)$  is zero. By construction,  $\Psi(f)$  becomes infinite if  $d_c(x, f) \rightarrow 0$ , i.e., this potential satisfies Requirements 2, 4, and 5, as long as the potential grows fast enough.

**5.1.3 Factor  $\gamma^S$ .** Restricting the potential to the interior of  $C(x, f)$  while keeping it differentiable, requires mollification. To construct a suitable  $\gamma^S(x, y)$ , we use a cubic spline basis function for  $\Phi^m$ , and the smoothed Heaviside step function  $H^\alpha(z) := H(z/\alpha) \in C^2(\mathbb{R})$  (Figure 11, Appendix C) for  $\Phi^\epsilon$ ,

Then we define the directional factor  $\gamma^S$  as

$$\begin{aligned} \delta^\alpha(z) &:= \frac{2}{\alpha}B^3\left(\frac{2z}{\alpha}\right), \\ g^S(x, y) &:= \delta^\alpha(\Phi^m(x, y))H^\alpha(\Phi^\epsilon(x, y)), \\ \gamma^S(x, y) &:= g^S(x, y)g^S(y, x) \end{aligned} \quad (6)$$

which ensures that the potential is supported within the contact set  $C(x, f)$ . Note the similarity between  $h_\epsilon(z)$  and  $\delta^\alpha(z)$ : one provides localization in distance w.r.t. position of  $f(x)$ , the other in direction with respect to  $n(x)$  for the interaction set points.

**5.1.4 Adaptive barrier localization.** To satisfy Requirement 3 (No spurious forces), we choose  $\epsilon(x)$  for each point  $x$  to be equal to

$$\epsilon(x) := \min(d_c(x, f_0)/2, \epsilon_{\text{trg}}),$$

where  $\epsilon_{\text{trg}}$  is the global parameter determining the maximal distance at which the potential may be nonzero and  $f_0$  is the rest deformation. Note that  $\epsilon(x)$  is not necessarily smooth or even continuous as a function of  $x$ . This may affect the convergence of the outer integral (under mesh refinement) in computing  $\Psi$ ; however, it does not affect the requirements, in particular, Requirement 5 (Differentiability) of the total potential's dependence on the shape parameters of  $f$  is still satisfied.

**Proposition 1.** *The potential (5) satisfies Requirements 1 to 5 if  $f$  is a curvature-continuous surface, with  $C(x, f)$  given by Definition 1,*

*distance-to-contact defined as  $d_c(x, f) := \min_{y \in C(x, f)} \|f(y) - f(x)\|$ , and  $\gamma(x, y)$  given by Equation (6).*

The details can be found in the supplementary document.

**5.1.5 Relation to other potentials.** Our potential eliminates many shortcomings of the previously proposed potentials described in Section 2.

This potential remains finite for admissible smooth surfaces, for our choice of  $p = n - 1$  (see the supplementary), without excluding a self-interaction radius as in Kamenskyy et al. [2018], at the same time retaining the barrier property exactly, thus preventing the possibility of configurations of the type shown in Figure 4.

Spurious repulsion under compression (Figure 3 and 4) is eliminated, as these points cannot be in the interaction set, as their normals are parallel or close to parallel. Similarly, the adaptive choice of  $\epsilon$  eliminates spurious forces in the rest state.

Unlike the IPC potential, the extent of the barrier  $\epsilon$  (as well as all other properties) is independent of discretization.

Compared to a barrier constructed from common gap functions as described in Section 2, our potential depends smoothly on the deformation  $f$ , unlike potentials based on the discontinuous DND or non-smooth CP gap functions. Thus, our potential supports efficient numerical methods.

Our smooth-surface potential is closest to the tangent energies of [Strzelecki and von der Mosel 2013; Yu et al. 2021]. Note that the factor  $\Phi_m$  can be interpreted as the magnitude of the projection of  $f(y) - f(x)$  to the tangent plane of the surface, i.e., exactly the denominator of the energy [Yu et al. 2021, Section 2.1]. As a consequence, it can be viewed as a form of our potential without the mollification in the factor  $\gamma$ , i.e., with  $\gamma$  set to  $\Phi^m$  and the part related to  $\Phi^\epsilon$  dropped, and localization in space. As a consequence, there are repulsive forces for arbitrarily close points on curved surfaces, vanishing only in the limit, since the tangent-point distance is finite for  $x$  and  $y$  with non-parallel normals. There is also interaction between two sides of, e.g., a very thin object.

Another important difference is the choice of the power of the potential  $p$ . In the setting of [Yu et al. 2021], the power  $p$  is set high to ensure that the surfaces determined by the energy are  $C^1$ , while in our case, these are not determined by the energy minimization, and their smoothness is ensured by other forces (e.g., elasticity). Moreover, finiteness for p.w. smooth surfaces imposes an upper bound on  $p$ .

However, we emphasize that in the case of piecewise smooth surfaces (as a more general case of piecewise linear meshes, which are widely used in physics simulations), normals are not uniquely defined, and significant adjustments are needed to the repulsive potentials to apply in this case. Moreover, for all desired properties to carry over to the discrete case, the discretization needs to be constructed in a particular way (Section 5.3).

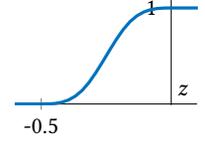


Fig. 11.  $H^\alpha(z)$  with  $\alpha = \frac{1}{2}$ .

## 5.2 Piecewise smooth contact

We now extend the potential to piecewise smooth surfaces. There are two reasons for this extension: (1) piecewise smooth surfaces are ubiquitous in graphics and scientific computing as they can represent shapes with sharp features, and (2) smooth surfaces are often approximated by piecewise linear surfaces for which efficient and robust continuous collision detection is available.

We consider surfaces consisting of patches  $\Omega_i$ , which form a possibly curved manifold mesh satisfying the standard definition [Do Carmo 2016]. We assume that each patch has continuously varying normals defined everywhere, including boundary (i.e., no cones are allowed), the edge curves meeting at a vertex, and faces meeting at an edge have distinct tangents and tangent planes in the undeformed shape. We refer to the curves separating patches as *edge curves*, or simply edges, and points shared by more than two patches as vertices.

In this case, there are six possibilities for a contact point, corresponding to the possible pairs of contacts between any two element types: Face-Face, Face-Edge, Face-Vertex, Edge-Edge, Edge-Vertex, Vertex-Vertex. We handle all these cases together in a uniform way.

**5.2.1 Interaction sets for piecewise smooth surfaces.** We use the same general definition for interaction sets  $C(x, f)$  for piecewise smooth surfaces. Specifically,  $C(x, f)$  is the set of points  $y$  close in the sense defined below to local minima of the distance function  $\|f(y) - f(x)\|$ , and with the vector  $f(y) - f(x)$  pointing to the exterior of the surface at  $x$  and to the interior at  $y$ .

However, more complex machinery is needed to convert this to a mathematical definition compared to  $\Phi^m$  and  $\Phi^e$  functions for smooth surfaces. Most of the effort required is to express the simple definition above as smooth functions of  $f$ .

**Local minimum constraint.** Consider the distance  $\|f(y) - f(x)\|$  as a function of  $y$ . If the closest point is in the interior of some patch  $\Omega_i$ , i.e., the contact point is a face point, then for the gradient w.r.t.  $y$ , we obtain

$$\nabla_y f(y)^\top (f(y) - f(x))_+ = 0 \quad (7)$$

at the closest point, where the gradient  $\nabla_y$  is computed with respect to a parametrization of  $\Omega_i$ . The columns of the matrix  $\nabla_y f$  are two tangents at  $y$ , i.e., this condition is equivalent to the condition that  $n(y)$  is parallel to  $f(y) - f(x)$  in the definition of  $C(x, f)$ .

To generalize the minimum condition to edge and vertex contact points on piecewise smooth surfaces, consider the set of patches  $\Omega_i$  containing a point  $y$  (one patch for face points, two for edge points, and any number for vertex points). For each  $\Omega_i$  and each parametric direction  $p \in \mathbb{R}^2$  at  $y$ , there is a well-defined unit tangent direction  $t := \partial_p f(y)$  of  $\Omega_i$ . The directional derivative  $\partial_p$  is one-sided if  $y$  is on the boundary of  $\Omega_i$ .

The condition for a local minimum is that for any  $\Omega_i$ , the distance does not decrease along any tangent  $t$ , i.e.,

$$t^\top (f(y) - f(x))_+ \geq 0 \quad (8)$$

It turns out to be sufficient to enforce this condition along *three* directions for each patch incident at a point, as shown in Figure 12, and explained in more detail in Appendix D.

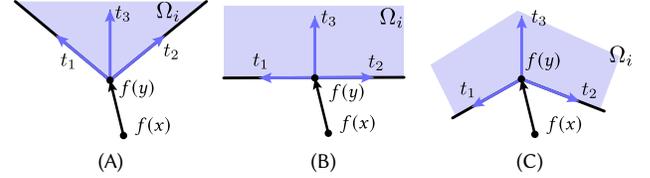


Fig. 12. Notation for tangents in Definition 2 in 3D.  $\Omega_i$  is a face (not necessarily triangle) in 3D colored with light purple,  $t_1$  and  $t_2$  are the two tangent vectors at  $f(y)$ ,  $t_3$  is the angle bisector of the angle between  $t_1$  and  $t_2$ . The face is viewed from the normal direction.

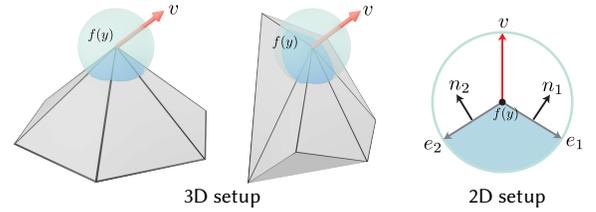


Fig. 13. Problem setup of determining if a vector  $v$  (red) is pointing inside of the piecewise linear surface. Point  $f(y)$  and its 1-ring neighbors on the piecewise linear surface (gray) intersect with a small sphere (green) centered at  $f(y)$ , their intersection is shown in blue. The intersection in 3D is not necessarily convex (middle).

**Exterior direction constraint.** The analog of  $\Phi^e(x, y) > 0$  requires a criterion for determining that a unit vector  $v := (f(y) - f(x))_+$  at a point  $f(y)$  of a closed piecewise smooth surface points inside. For piecewise linear surfaces, as in Figure 13, the point  $f(y)$  and its 1-ring neighbors form a cone. Consider a small sphere centered at  $f(y)$ , its surface is partitioned into two parts by the cone, and  $v$  intersects the sphere surface at one point. The problem is identifying the part of the sphere containing the point. For high-order surfaces, determining if  $v$  points inside only requires checking the cone formed by the tangent planes at  $f(y)$ , which is equivalent to considering the same problem for piecewise linear surfaces.

We are going to define a function  $\Phi^e(x, y)$  that is positive if and only if  $v$  points inwards at  $f(y)$ . Consider the problem in 2D (Figure 13). The two incident edges at  $f(y)$  are  $e_1, e_2$ , with corresponding normals  $n_1, n_2$ . We assume all vectors have unit length since we only care about the directions. Suppose  $e_1 \times n_1 > 0$  (otherwise swap  $e_1$  and  $e_2$ ), to determine if  $v$  points outside the surface we only need to check if  $e_1, v, e_2$  are in counter-clockwise order, so we can define

$$\Phi^e(x, y) := (v - e_1) \times (v - e_2). \quad (9)$$

Note that  $\Phi^e(x, y) = 0$  if and only if  $v, e_1, e_2$  are on the same line, i.e.  $v$  overlaps with  $e_1$  or  $e_2$ .

In 3D, we first discuss  $\Phi^e(x, y)$  for edges in Edge-Vertex and Edge-Edge contact. As shown in Figure 14, to determine if a vector  $v$  points inwards, we can project everything onto the plane perpendicular to edge  $e_j$  and apply Equation (9) on the projected vectors. For vertices, however, defining  $\Phi^e(x, y)$  becomes more complicated. A detailed description of how to define  $\Phi^e(x, y)$  in a robust fashion for vertices

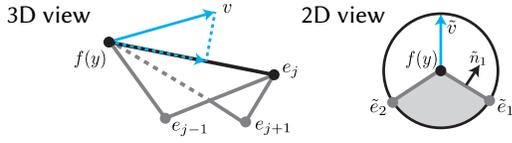


Fig. 14. 3D (left) and 2D (right) views of the closest point on an edge. The 2D view is the projection of the 3D view onto the plane perpendicular to the edge  $e_j$ .  $\tilde{v}, \tilde{e}_1, \tilde{e}_2$  are the projections of  $v, e_{j-1}, e_{j+1}$  respectively.  $\tilde{n}_1$  is the projected normal of the face bounded by  $e_{j-1}, e_j$ . Deciding if  $v$  points inside the cone in 3D is equivalent to deciding if  $\tilde{v}$  is inside the sector bounded by  $\tilde{e}_1$  and  $\tilde{e}_2$  in 2D.

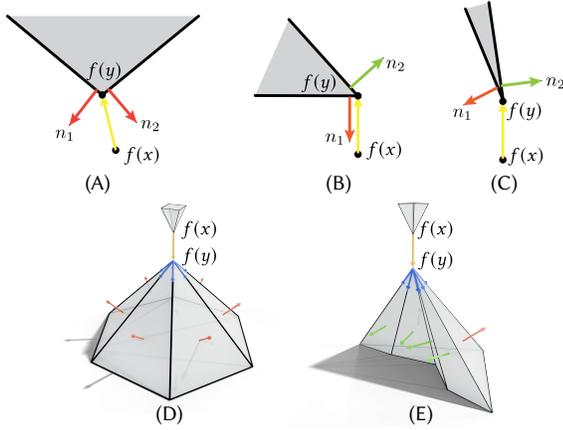


Fig. 15. 2D (top) and 3D (bottom) normal directions for contact points.  $f(y) - f(x)$  is shown in yellow;  $n_i(y)$  are shown in red if Equation (16) is satisfied, otherwise green. Tangent directions are shown in blue in 3D. In simple contact cases, e.g. (A) and (D), the normals all satisfy Equation (16), but in other cases, not all normals satisfy the inequality.

is included in Appendix B. This leads to the following definition of an interaction set:

**Definition 2** (Interaction sets for p.w. smooth surfaces). For each point  $y$  of  $\Omega$ , let  $I$  be the set of indices of patches  $\Omega_i$  containing  $y$ , and for each patch, define  $t_i^k(y)$ ,  $k = 1, 2, 3$  as above. Then  $C(x, f)$  consists of points  $y$  satisfying

$$\begin{aligned} \Phi_{ik}^m(x, y) &:= t_i^k(y)^\top (f(y) - f(x))_+ \geq -\alpha, \\ \Phi_{ik}^m(y, x) &\geq -\alpha, \quad \Phi^e(x, y) \geq -\beta, \quad \Phi^e(y, x) \geq -\beta, \end{aligned} \quad (10)$$

for  $i \in I$ ,  $k = 1, 2, 3$ ,  $0 < \alpha \leq 1$ , and  $0 < \beta \leq 1$  is a separate parameter that controls the smoothness of  $\Phi^e$ . We use  $\beta = 0.1$ ; the specific choice of  $\beta$  has little impact on the method's behavior and numerical stability.

*Remark.* The size of the index set  $I$  depends on the type of primitive being considered: In 2D, edge curves have size  $|I| = 1$ ; vertices have  $|I| = 2$ , including the two edges joined at this vertex. In 3D, faces have  $|I| = 1$ , edge curves have  $|I| = 2$ , and vertices have  $|I| \geq 2$ .

Using mollification, we define factors  $g^e(x, y)$  and  $g^m(x, y)$  from  $\Phi^e$  and  $\Phi_{ik}^m$  as described in Appendix D.

**5.2.2 Contact potential for piecewise smooth surfaces.** Using the new definition of interaction sets  $C(x)$  and combining the local minimum and exterior direction constraints, we get, similar to the smooth case,

$$\begin{aligned} g^{PS}(x, y) &:= g^e(x, y)g^m(x, y) \\ \gamma^{PS}(x, y) &:= g^{PS}(x, y)g^{PS}(y, x) \end{aligned} \quad (11)$$

However, one cannot simply replace  $\gamma^S(x, y)$  with  $\gamma^{PS}(x, y)$  in Equation (5) to obtain the potential for the piecewise smooth case. Recall that contact may happen to measure-zero sets on the piecewise smooth surface, e.g., vertex-vertex and edge-edge contact. Consequently, the contact set in these configurations has zero measure (edge curves and vertices in 3D), and direct surface integration of  $\psi(x)$  will lead to a zero potential.

For this reason, we treat low-dimensional elements separately. Informally, we can think about the interaction of lower-dimensional elements, as expanding them into areas: e.g., an area of width  $L$  along curves, and an area of size  $L^2$  assigned to vertices, with the potential constant along the direction in the area orthogonal to the curve, or on the whole area assigned to a vertex. More precisely, this corresponds to adding line integrals for edge curves weighted by  $L$ , and point sums for vertices, weighted by  $L^2$ . The potentials we integrate for a pair of elements  $G$  and  $H$ , possibly of different dimensions, e.g., with  $G$  being a vertex and  $H$  a face, can be written in the form

$$P(x, y; G, H) = \gamma^{PS}(x, y; G, H)p_{\epsilon(x)}(x, y)$$

where  $\gamma^{PS}(x, y; G, H)$  is the factor defined in Equation (11), for  $x$  considered as a point on element  $G$ , in other words, if  $G$  is a face, and  $x$  happens to be an edge or vertex point, the factor  $\gamma^{PS}(x, y)$  is computed using formulas for face points  $x$ .

Then the total potential can be written as

$$\Psi(f) = \sum_{(g,h)} \sum_{\substack{i \in I_g, j \in I_h \\ G_i \cap H_j = \emptyset}} L^{4-\dim g - \dim h} \int_{G_i} \int_{H_j} P(x, y; G_i, H_j) dx dy \quad (12)$$

where  $g, h$  are one of the element types {Face, Edge, Vertex}, with the first sum is over 9 unordered pairs,  $I_g$  and  $I_h$  are sets of indices of elements of types  $g$  and  $h$ , and  $G$  and  $H$  element indices. The integrals are area, line, or 0-dimensional, i.e. simple evaluations for vertices.

**Proposition 2.** *The potential (12) satisfies Requirements 1 to 5 if  $f$  is piecewise-curvature-continuous surface as defined in this section, with  $C(x, f)$  given by Definition 2, distance-to-contact defined as  $d_c(x, f) := \min_{y \in C(x, f)} \|f(y) - f(x)\|$ , and  $\gamma(x, y)$  given by Equation (11).*

Please see the supplementary document for details.

**5.2.3 Parameters of the potential.** Whether the potential satisfies contact potential requirements does not depend on the choice of  $L > 0$ ,  $\epsilon_{\text{trg}} > 0$ , and  $0 < \alpha < 1$ , as long as these are positive. There is no strong impact on robustness, as long as the values are not too close to the bounds, but it may affect performance.

- $\alpha$  determines the size of the interaction sets, and as a consequence, how smoothly the potential depends on  $f$ . The closer  $\alpha$  is to zero, the less smooth the dependence is, but the more localized it is.
- $\epsilon_{trg}$  determines the upper bound on how far the potential extends ( $\hat{d}$  in IPC), but especially at concave corners,  $\epsilon_{trg}$  as well as  $f_0$ , affect this.
- $L$  determines the strength of potential for low-dimensional contact. In our examples, for vertices, we set it to the average edge length around the vertex; for edges, we set it to the length of the edge; for faces,  $L$  is not needed.

### 5.3 Discretization

For a barrier method to guarantee that the geometric constraints are not violated, and maintains the attractive features of the potential, we must ensure that the discrete version satisfies the same conditions, most importantly:

- requirement 2 (the exact barrier property with respect to the discretized geometry);
- requirement 5 (differentiability).

It turns out that these requirements are, to an extent, conflicting, and additional effort is needed to satisfy both. We consider the lowest order discretization for piecewise-linear surfaces, leaving extensions to higher orders as future work. Designing a discretization that, on the one hand, converges to the underlying continuum potential and, on the other hand, satisfies all requirements is nontrivial already in this case.

**5.3.1 Ensuring the barrier property.** If we use standard quadrature points as it is done in [Kamensky et al. 2018; Yu et al. 2021], the integral approximations in Equation (12) may have optimal accuracy, but the barrier property is satisfied approximately, and interpenetration is possible, as discussed in [Kamensky et al. 2018] and Section 3.

To address this problem, we choose the closest points on element pairs as our quadrature points, possibly reducing integration precision, but still satisfying the requirements.

In this case, the lowest-order discretization of the potential integral is

$$\Psi^D(f) = \sum_{(g,h)} \sum_{(i,j)} L^{4-\dim g - \dim h} p(x_i, y_j; G_i, H_j) A(G_i) A(H_j) \quad (13)$$

where  $(x_i, y_j)$  is a pair of closest points on elements  $G_i$  and  $H_j$ , respectively, and  $(G_i, H_j)$  are pairs of non-adjacent elements, and  $A(G_i)$  and  $A(H_j)$  are measures of elements (area for faces, length for edges and 1 for vertices).

Different from Equation (12), the sum above over  $(g, h)$  only includes four types: Vertex-Face, Edge-Edge, Vertex-Edge, Vertex-Vertex, since in the case of piecewise linear surfaces, other types are reduced to these types when only the closest points are considered. [Li et al. 2020] only considers Vertex-Face and Edge-Edge, since it is sufficient to satisfy Requirement 2 in their case. In our work, due to the mollification discussed below, we include Vertex-Edge and Vertex-Vertex. In practice, we do not have more contact pairs than [Li et al. 2020] (Table 2), since our contact set is more narrow due to our local minimum and exterior direction constraints.

As the closest distances between points are used, the terms in Equation (13) approximating corresponding terms in Equation (12), bound them strictly from above, ensuring the exact barrier property. One exception is the terms for adjacent elements, for which the closest distance is zero, and the bound would be infinity.

From the perspective of convergence of the potential to the limit under refinement, this is not an issue, as the fraction of adjacent pairs among all pairs converges to zero under refinement. Excluding interactions between these elements does not affect the barrier requirement either: for piecewise linear meshes, if adjacent elements are in contact, other, nonadjacent elements will be in contact (see Appendix E for a more detailed discussion), i.e., Requirement 2 still holds.

However, in the piecewise linear case, the contact between these elements is possible only if a non-adjacent pair is in contact, so these can be safely omitted.

However, using the closest points as quadrature points creates a new complication: 3D positions of quadrature points with a fixed position in the material space naturally depend smoothly on the deformation  $f$ , but this is not the case for the positions of closest points.

**5.3.2 Ensuring differentiability.** For the discrete potential to depend differentially on the shape parameters (vertex positions in the piecewise linear case), we need an additional modification. First, observe that the distance between two elements is already  $C^1$  with the only exception being the distance between two parallel edges, which can be mollified as explained in [Li et al. 2020], so the term  $p_{\epsilon(x)}$  in  $P$  does not require modifications. However, the directional factor  $\gamma^{PS}$  Equation (11) depends on the direction between closest points, which may be only  $C^0$  continuous with respect to the deformation  $f$ , when the closest point is on the boundary of the face or edge (Figure 39). We introduce a mollification  $M(x, y)$  for the closest point pair direction, as an extra multiplicative factor for  $\gamma^{PS}$ :

$$\gamma^{PS}(x, y) := g^{PS}(x, y) g^{PS}(y, x) M(x, y), \quad (14)$$

Importantly, introducing this factor does not affect other requirements for the potential. The details of the construction of this factor can be found in Appendix A.

**5.3.3 Directional factor: local minimum terms.** In the case of only the closest points being used as quadrature points, we can simplify the local minimum terms discussed in Section 5.2. If a face point  $y \in \Omega_i$  is the closest point on  $\Omega_i$  to a fixed point  $x$ , then  $y$  is a local minimum of the distance function  $d_x(y) = \|f(y) - f(x)\|$ , i.e. Equation (7) is satisfied, so no local minimum constraint is needed. Similarly, if  $y$  is on an edge, then  $y$  is a local minimum of  $d_x(y)$  on the edge, so the local minimum term along the edge can be ignored (Figure 16).

**5.3.4 Adaptive local factor  $\epsilon$ .** We introduce adaptive  $\epsilon$  for our formulation. We first specify a fixed  $\epsilon_{trg}$  for the simulation and collect contact pairs in the rest configuration within  $\epsilon_{trg}$  and with nonzero potential values. We pick  $\epsilon$  for every primitive (vertex/edge/face) so that none of the contact pairs are active in the rest configuration. Then the  $\epsilon$  for every edge (face) is chosen to be the minimum among its neighboring vertices (edges). Benefiting from local minimum and

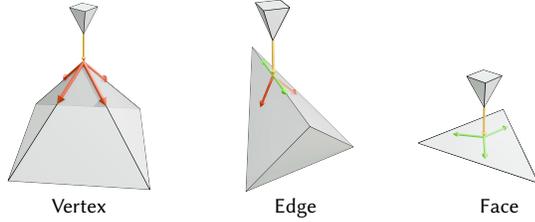


Fig. 16. Local minimum constraints for closest points in contact.  $f(y) - f(x)$  is shown in yellow, tangent directions included in  $\Phi^m$  are in red, otherwise green. By the property of the closest point, the distance function  $d_x(y)$  has zero derivatives along the green tangents, so Equation (10) is always satisfied.

exterior direction constraints, the adaptive  $\epsilon$  in our method can be much larger than in [Li et al. 2020] (Figure 18).

**5.3.5 Satisfying requirements.** Just as the smooth and piecewise smooth potentials, the discretized potential 13 satisfies our requirements:

**Proposition 3.** *The discrete potential 13, with the interaction sets and directional factors computed as described above, satisfies the Requirements 2 to 5.*

We note that there is no need for Requirement 1 for a discrete potential, as it is a finite sum: *any* potential that grows to infinity for  $\|f(y) - f(x)\| \rightarrow 0$  can be used, as this is a finite sum, and, unlike integrals, it is unbounded when any term is unbounded. As we use the closest points on elements as quadrature points, the barrier property is guaranteed in the discrete case.

However, if the growth rate of the potential is too low for the smooth or piecewise-smooth integral potentials, this means that under refinement, the potential will become progressively weaker (Figure 38).

The remaining properties are verified in a similar way to the continuum case (see the supplementary document).

**5.3.6 Convergence under refinement.** Above, the piecewise smooth formulation is used directly to discretize the potentials on piecewise linear meshes. A natural question is whether the potential for piecewise linear meshes sampled from a smooth (or piecewise smooth) surface will converge to the potential directly defined for this surface. Mathematical analysis of convergence is beyond the scope of this paper, but we expect that the piecewise linear mesh potentials converge to the potential defined on the smooth mesh if the parameter  $L$  is adjusted in the same way as the mesh edge lengths, i.e., decreased by a factor of two if the mesh is refined uniformly. Intuitively, the discretization corresponds to sampling the integrand at a set of points on faces, edges, and vertices and weighting by areas associated with them ( $L$  controls the size of areas assigned to edges and vertices), which suggests this scaling of  $L$ .

The situation is more complicated for the refinement of piecewise smooth surfaces. In this case, the limit surface integral contains the low-dimensional terms (Edge-Edge, Vertex-Face, etc.). These are computed on sharp feature curves and at vertices embedded in the surface, and in discretization corresponding to piecewise linear

feature curves and vertices embedded in a piecewise linear mesh. In this case, for convergence to the correct limit potential, the factors  $L$  for edges and vertices on these feature curves should *not* be adjusted, unlike factors for edges and vertices inserted on faces.

## 5.4 Friction

The friction formulation in [Li et al. 2020] can be reused in our setting with minor modifications, which we report here for completeness. In IPC [Li et al. 2020], the friction force for a contact pair is

$$F_k(x) = -\mu\lambda_k T_k f_1(\|u_k\|) \frac{u_k}{\|u_k\|},$$

where  $\mu$  is the friction coefficient,  $T_k(x)$  is the local tangential basis,  $\lambda_k$  is the contact force magnitude,  $u_k$  is the local relative sliding displacement at contact  $k$ ,

$$f_1(y) = \begin{cases} -\frac{y^2}{\epsilon_v^2 h^2} + \frac{2y}{\epsilon_v h}, & y \in (0, h\epsilon_v) \\ 1, & y \geq h\epsilon_v, \end{cases}$$

where  $h$  is the time step size,  $\epsilon_v$  is a velocity magnitude bound that controls how accurately the friction is approximated. Then the dependency of  $F_k(x)$  on  $T_k$  and  $\lambda_k$  is made explicit (or lagged), to obtain the friction potential

$$D_k(x) = \mu\lambda_k^n f_0(\|u_k\|).$$

where  $\lambda_k^n$  is the contact force magnitude from a prior nonlinear solve (or previous time step)  $n$ . Since the dependency of  $D_k(x)$  on  $\lambda_k^n$  is explicit,  $\lambda_k^n$  can be directly replaced by the contact force magnitude in our method.

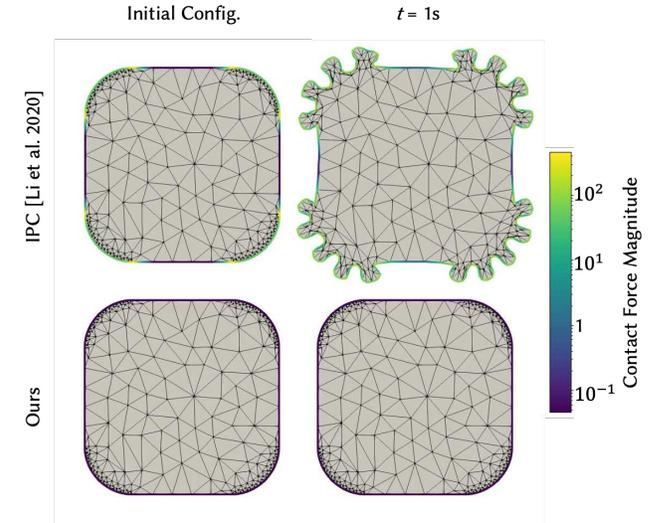


Fig. 17. **Spurious stresses: nonuniform squircle mesh.** Our method properly handles nonuniform meshes. In this case, we consider a 2D rounded block (of size 1 m x 1 m) with maximal edge length 0.21 m in its straight sections and minimal edge length 0.01 m at its corners. Top row: with  $\hat{d} = 0.1$  m, IPC introduces spurious contact forces in the refined corners, resulting in a deformation. Bottom row: our method avoids this by filtering based on tangent directions, without the need for small  $\epsilon(x)$ .

## 6 EVALUATION

We implement our algorithm by extending the IPC Toolkit [Ferguson et al. 2020] (details provided in the supplemental document). We use Eigen [Guennebaud et al. 2010] for linear-algebra operations, Pardiso [Alappat et al. 2020; Bollhöfer et al. 2019, 2020] for solving linear systems, and PolyFEM [Schneider et al. 2019] as the finite element simulation framework. All experiments are run on a system with an AMD Ryzen Threadripper PRO 3995WX 64-Cores (limited to 16 threads) and 440 GiB of memory. For comparisons against IPC [Li et al. 2020] and “Convergent IPC” [Li et al. 2023], we use the open-source implementations provided in the IPC Toolkit (together with PolyFEM). Please see our supplemental video for animations. All simulation parameters and statistics are summarized in Table 3. Our reference implementation, used to generate all results, will be released as an open-source project.

### 6.1 Resolving Spurious Forces

In this section, we consider a variety of unit tests where the original IPC formulation introduces spurious forces.

**6.1.1 Using larger  $\epsilon_{\text{trg}}$ .** Suppose the input mesh is relatively fine, either globally (a dense mesh) or locally (adaptively refined e.g. to resolve fine features). Having a large  $\epsilon_{\text{trg}}$  allows one to solve the implicit time-stepping problem faster because the barrier is less numerically stiff, and CCD needs to do less work in the line search if the objects are kept further apart by the potential. We provide an example of this scenario in Figure 17, where it is natural to have small edges around the rounded corners of a square and long edges along the sides. However, doing so restricts the range of usable  $\hat{d}$  (IPC’s notation for  $\epsilon_{\text{trg}}$ ) for the IPC barrier. Using a value ( $\hat{d} = 0.1$  m in this case) larger than the minimum edge length ( $h_{\text{min}} = 0.01$  m) results in spurious forces along the corners and artifacts upon simulation. In contrast, our utilization of an adaptive  $\epsilon$  allows us to choose a starting  $\epsilon_{\text{trg}}$  that results in zero initial contact force. These rest forces can be avoided while still using a large  $\epsilon$  for the non-refined regions (Figure 18). Further, in Figure 19, we run a simulation on the same armadillo model with IPC and our method. We choose  $\hat{d} = 0.001$  m for our method and  $\epsilon_{\text{trg}} = 0.0004$  m for IPC so that there is no contact force at rest shape. Both methods produce similar results, our method takes 512 iterations and 55 minutes (2/3 less than IPC), while IPC takes 1555 iterations and 161 minutes.

**6.1.2 Large Deformation.** In the presence of large deformations, elements may shrink a lot, and in this scenario, the original IPC barrier formulation adds spurious forces which make the material locally stiffer (Figure 21). Our formulation does not suffer from this issue.

**6.1.3 Spurious stress in the rest configuration.** Even if  $\hat{d} < h_{\text{min}}$  is satisfied, the IPC formulation may still have spurious stress in the rest configuration. We show two such scenarios in Figure 20.

In the first case (Figure 20 Top), two blocks in the initial configuration sit on the plane with the initial distance between blocks less than  $\hat{d}$ . With IPC, they incorrectly start sliding apart without external force applied where the blocks should stay still at all times.

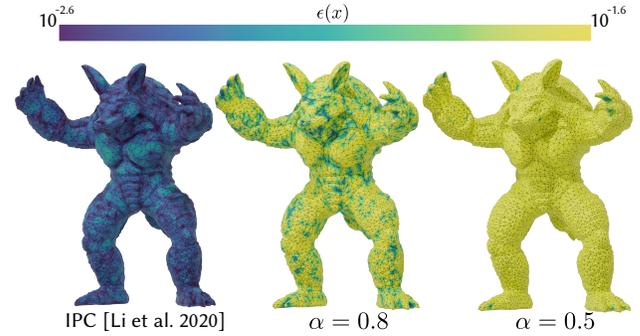


Fig. 18.  $\epsilon$  **distribution.** We implement an adaptive locality parameter with IPC and compare  $\epsilon$  after adaptation to avoid spurious rest forces with  $\epsilon_{\text{trg}} = 0.02$  m. IPC (left) requires small  $\hat{d}$  all over the surface due to the requirement that  $\hat{d}$  be smaller than the shortest adjacent edge. Due to local minimum and exterior direction constraints, our method (middle and right) is able to concentrate  $\epsilon$  refinement on the fingers and toes of the figure.

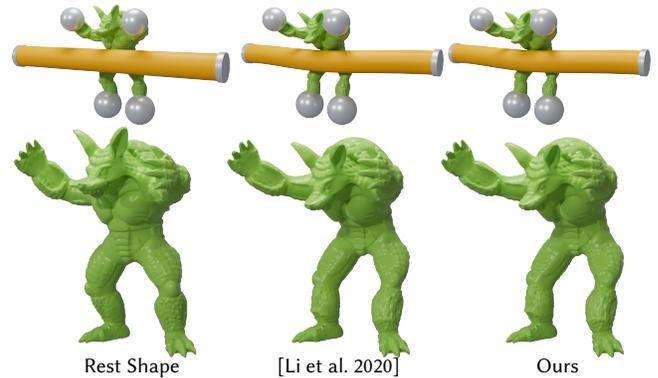


Fig. 19. **Armadillo-Bar.** An armadillo interacts with an elastic bar. The two ends of the bar are forced to move towards the back of the armadillo, and both the hands and feet of the armadillo are fixed. Both the armadillo and bar are elastic. We show the full scene on the top row, and only the armadillos on the bottom row.

In the second case (Figure 20 Bottom), IPC causes the slit to expand at the top without external forces and stress appears at the bottom.

Our approach avoids spurious forces between close objects by using an adaptive  $\epsilon_{\text{trg}}$ . For the contact between primitives on the same object, due to our local minimum and exterior direction constraints, we can use  $\epsilon_{\text{trg}}$  larger than the edge lengths without activating contact at rest configuration (with a wide range of  $\alpha$ ).

**6.1.4 Spurious stress under deformation.** Even for a well-chosen  $\hat{d}$  parameter and a benign rest configuration, the IPC barrier can still add spurious forces upon deformation.

In Figure 21, we compress a cube mesh to 33% of its original height. We assign a Poisson ratio of 0 to the cube to show an example where no bulging and/or folding of the surface occurs. The value for  $\hat{d}$  is initially chosen such that no points are in contact, but upon compression distances shrink and IPC introduces spurious

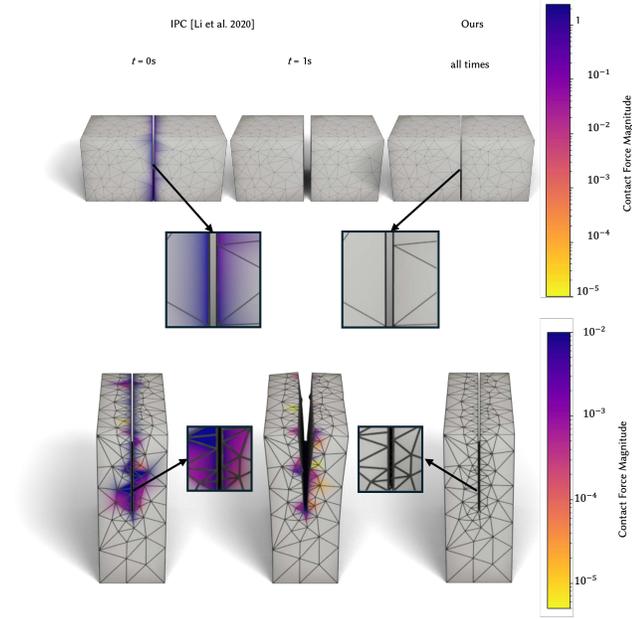


Fig. 20. **Spurious stresses: rest configuration 3D.** Top row: two cubes initially separated by less than  $\hat{d} = 0.025$  m. IPC artificially repels the two blocks while ours does not. Bottom row: a block with a slit of width less than  $\epsilon_{\text{trg}} = 0.0125$  m. Our method does not introduce spurious forces across the slit.

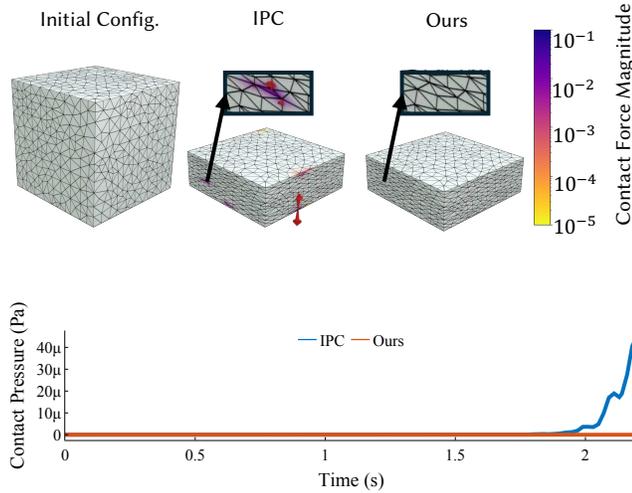


Fig. 21. **Spurious stresses: compression.** An initially valid choice for  $\hat{d}$  can lead to spurious contact (arrows and color) when using IPC. Left: cube at rest with a minimum edge length of 0.03 m. Center: upon compression with  $\hat{d} = 0.0125$  m, IPC introduces spurious contacts. Right: our method avoids this by considering the tangent directions when finding interaction sets. Bottom: IPC introduces artificial contact pressure.

contact forces on the sides of the cube (despite them being flat). Our formulation does not have spurious forces because we use the angle

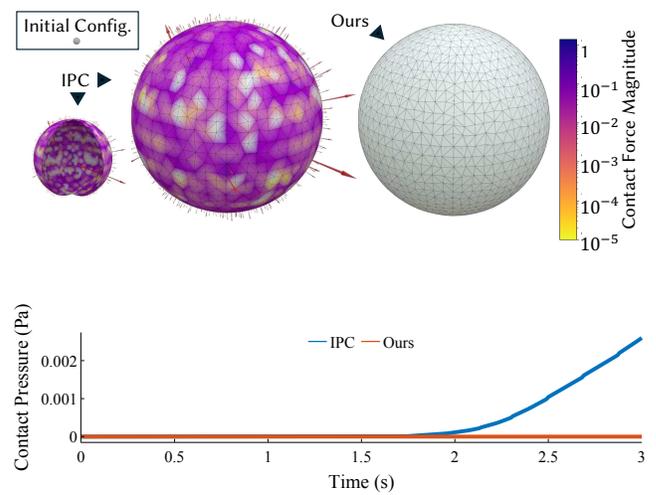


Fig. 22. **Spurious stresses: expansion.** Upper left: an initially deflated balloon with rest thickness of 0.1 m is expanded by an outward force of 5 kN. Center: with  $\hat{d} = 0.02$  m, the balloon eventually becomes thin enough that spurious contact forces (represented as red arrows) between its inner and outer layers appear. Lower left: a cross-section showing spurious forces on both the inner and outer layer of the balloon. Right: our method can inflate the balloon until it is arbitrarily thin without introducing artificial contact forces. Bottom: artificial contact pressure is introduced by IPC.

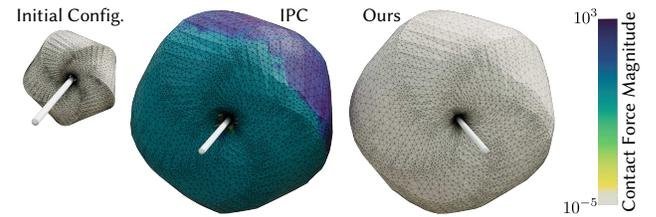


Fig. 23. **Spurious stresses: donut expansion.** Left: a donut with rest thickness of 0.02 m is inflated around a thin stick by a pressure boundary condition of 0.1t MPa, with simulation time  $t$  up to 0.6 s. Middle: with  $\hat{d} = 0.01$  m, the balloon eventually becomes thin enough that IPC introduces spurious contact forces between its inner and outer layers. Right: our method can inflate the balloon until it is arbitrarily thin without introducing artificial contact forces while still capturing contact forces between the rod and balloon.

between the direction to the point and surface normal to build our interaction sets.

In Figure 22, we inflate a spherical balloon modeled as a volumetric membrane of thickness 0.1 m, hanging on a rigid stick.<sup>1</sup> As the balloon inflates, its walls get thinner and eventually they become thinner than the initially chosen  $\hat{d} = 0.02$  m. At this point, IPC treats the inner and outer sides as in contact, introducing forces between the two sides. Our interaction set for one side does not include the

<sup>1</sup>While one could model the balloon using co-dimensional shell elements to avoid this issue, modeling the thickness may be important for analysis or design. For example, with constant outward pressure, the balloon oscillates in thickness, which a shell model would not capture.

other as it accounts for the angle of contact, avoiding this issue. As a more complex example (Figure 23), we inflate a donut-shape balloon around a thin stick. Our method only has contact forces around the center, while IPC has spurious contact forces everywhere. Although one can filter the interaction set by the surface connected component for IPC in these two simple examples (since the inner surface of the shell does not have contact with the outer surface), in more complex scenes (Figure 29) it does not work anymore, while our method avoids these issues by relying on our definition of interaction sets.

In Figure 24, we extrude the extended  $\chi$ -shaped structure from [Joodaky 2020] to 3D and simulate its compression. When the shape buckles it forms *cusp* contacts at the corners. Since the barrier potentials in different methods have very different scales, for a fair comparison, we pick  $\kappa$  for each method so that the minimum distance in the simulation is roughly the same ( $\kappa = 10^4$  for Convergent IPC and our method,  $\kappa = 10^5$  for IPC). We show that both Convergent IPC [Li et al. 2023] and IPC [Li et al. 2020] exhibit large contact forces in these regions, while our method has much less contact forces around the *cusp* because of the local minimum and exterior direction constraints – only when edges are close enough to parallel, i.e. the angle of the *cusp* becomes small enough, the contact is activated. Without this constraint, spurious stresses appear in both IPC and Convergent IPC.

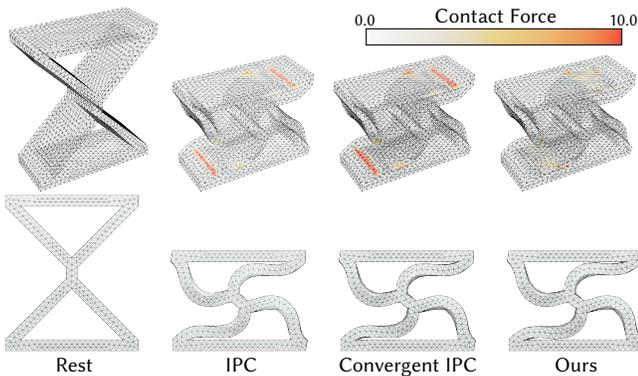


Fig. 24. **Cusp compression.** Compression of a  $\chi$ -shaped structure. We visualize the contact force distribution on the top row and the front view on the bottom row. Our method significantly reduces the contact forces around the cusp compared with IPC and convergent IPC.

## 6.2 3D Examples

We reproduce challenging 3D simulation examples of [Li et al. 2020]. First, we validate our method on the 3D unit tests proposed by [Erleben 2018] in Figure 25. We see similar results to those shown in [Li et al. 2020], but we highlight one improved result in Figure 26 where we see reduced spurious tangential movement compared to [Li et al. 2020] and [Li et al. 2023].

Second, we reproduce the dolphin funnel (Figure 27), trash compactor (Figure 28), and mat twist (Figure 29) examples. Each of these examples features large deformations and complex contacts. Just as in [Li et al. 2020], we robustly handle these scenarios and prevent

intersections and inversions at every step. We note that in [Li et al. 2020] the fixed corotational model, which allows element inversion, was used instead of NeoHookean, so the number of iterations and timing reported in [Li et al. 2020] is significantly lower. To make fair comparisons, we also run [Li et al. 2020] with NeoHookean and report the statistics in Table 3. We refer to [Smith et al. 2018] for the artifacts in the fixed corotational model and comparisons with NeoHookean. For the dolphin funnel, we report the breakdown timing of our method: The Hessian assembly takes 34%, linear solve 28%, narrow phase CCD 19%, broad phase CCD 4%, and line search (excluding CCD) 14%.

In Figure 31, as a stress test for our local minimum and exterior direction constraints, we generate an adaptive mesh for an n-legged monkey saddle, for which the normal oscillates around the center. For IPC, we use the maximum  $\hat{d}$  that does not activate contact forces at the rest configuration, which is  $5 \times 10^{-5}$  m. Since our method allows  $\epsilon_{\text{trg}}$  larger than the edge length without activating contact, we make use of this advantage and set  $\epsilon_{\text{trg}} = 2 \times 10^{-4}$  m. Both methods produce similar results, our method takes 4255 iterations and 3.7 hours (1/3 less than IPC), while IPC takes 6710 iterations and 5.7 hours.

Finally, to validate our friction model, we simulate the bunny sliding on a slope with various friction coefficients, and with our method, Convergent IPC [Li et al. 2023], and IPC [Li et al. 2020] (Figure 30). All three methods produce similar results.

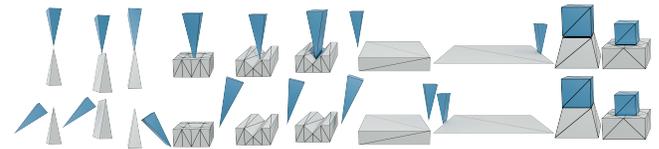


Fig. 25. **Erleben tests.** We reproduce the test-cases of Erleben [2018]. Top: initial conditions involving challenging exact point-point, point-edge, and edge-edge collisions. Bottom: as in [Li et al. 2020], our approach robustly passes all the tests.

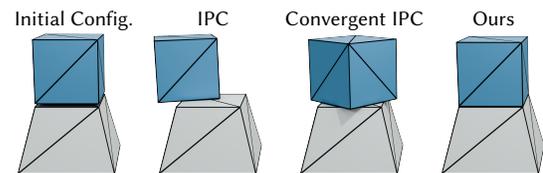


Fig. 26. **Erleben test: cliff edge.** We reproduce the cliff edge test-case of Erleben [2018] with  $\epsilon_{\text{trg}} = 0.01$  m. IPC [Li et al. 2020] and Convergent IPC [Li et al. 2023] pass the test but introduce spurious horizontal forces, causing the top block to rotate or slide. Our method significantly reduces extra sliding due to the restriction in contact forces.

## 6.3 Inverse Design

We perform a shape optimization example to show one advantage of using a large  $\hat{d}$  in the simulations. In Figure 32, we optimize the shape of a plier so that it can grasp the torus under gravity

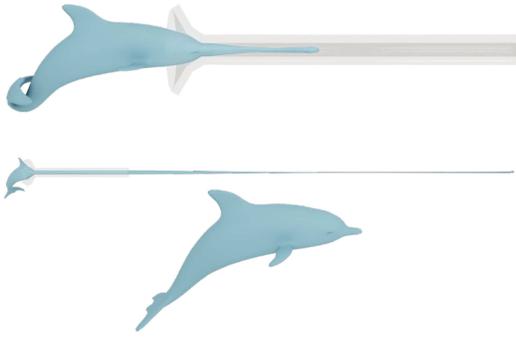


Fig. 27. **Dolphin in a funnel.** We reproduce the funnel test from [Li et al. 2020] using our method. Top: an elastic dolphin is pulled through a small tube. Middle: this causes extreme deformations. Bottom: the dolphin squeezes through without artifact and recovers its original shape.

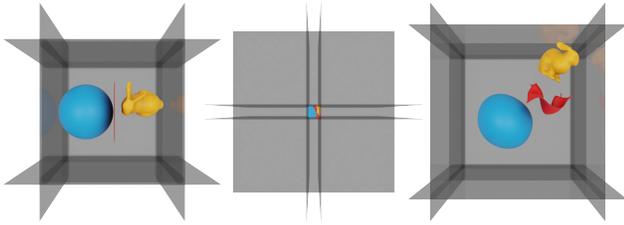


Fig. 28. **Trash-Compactor.** We reproduce the trash-compactor example from [Li et al. 2020] using our method. Left: three objects are placed in a compactor. Middle: the objects are compressed. Right: the compactor releases and the shapes return to their original shape without intersections.

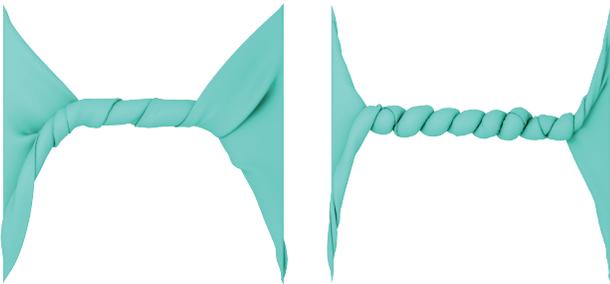


Fig. 29. **Mat-Twist.** We reproduce the mat-twist example from [Li et al. 2020], demonstrating, overall, our method is similarly robust to the original Incremental Potential Contact (IPC). Left: our simulation at 10 s after 2 rounds of twisting at both ends. Right: at 40 s after 8 rounds of twisting.

when its handles are pulled outwards. We define the objective to be the contact force magnitude in the simulation and maximize the objective with L-BFGS. To reduce the dimension of the shape design space, we use the shape representation in [Gjoka et al. 2024] and sample 15 control points on the plier surface. We compute shape derivatives of the objective following [Huang et al. 2024b].

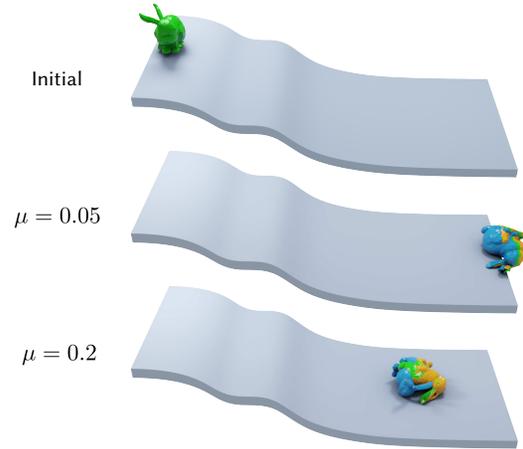


Fig. 30. **Friction: Sliding bunny.** A bunny is thrown from the top of the slope. We simulate with friction coefficients 0.05 and 0.2, and with our method  $\alpha = 0.5$  (blue), Convergent IPC [Li et al. 2023] (yellow), and IPC [Li et al. 2020] (green). The results at 2 s are shown.

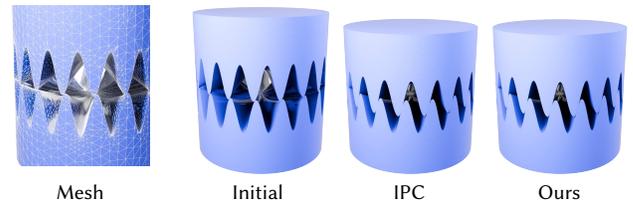


Fig. 31. **Monkey saddle.** We force the top surface of the object on top to rotate in the counter-clockwise direction and fix the bottom surface of the object at the bottom. Our method is able to handle this scene with complex normal directions.

For efficiency, we simulate only half of the plier and set symmetric boundary conditions.

As a baseline, we first run the shape optimization with [Li et al. 2020] (Figure 32 A). Due to the restriction that  $\hat{d}$  should be smaller than the minimum edge length of the mesh (to avoid spurious contact forces in Section 6.1), there is no contact force on the initial shape since the distance between the torus and the plier is larger than  $\hat{d}$ . Therefore, the shape derivatives of the objective are zero and the shape optimization cannot proceed. With our method (Figure 32 B), however, the  $\hat{d}$  can be larger than the edge length without creating spurious forces between adjacent mesh vertices, so we can pick a large enough  $\hat{d}$  so that there is nonzero contact force between the plier and the torus, and the shape optimization can proceed. The optimized plier manages to create enough contact forces to grasp the torus (Figure 32 D).

#### 6.4 Comparison to Repulsive Surfaces

Tangent-point energy (TPE) of Yu et al. [2021], discussed in Section 2, has global support, which makes it expensive in simulations, even with acceleration. To adapt it to our purposes, we multiply the

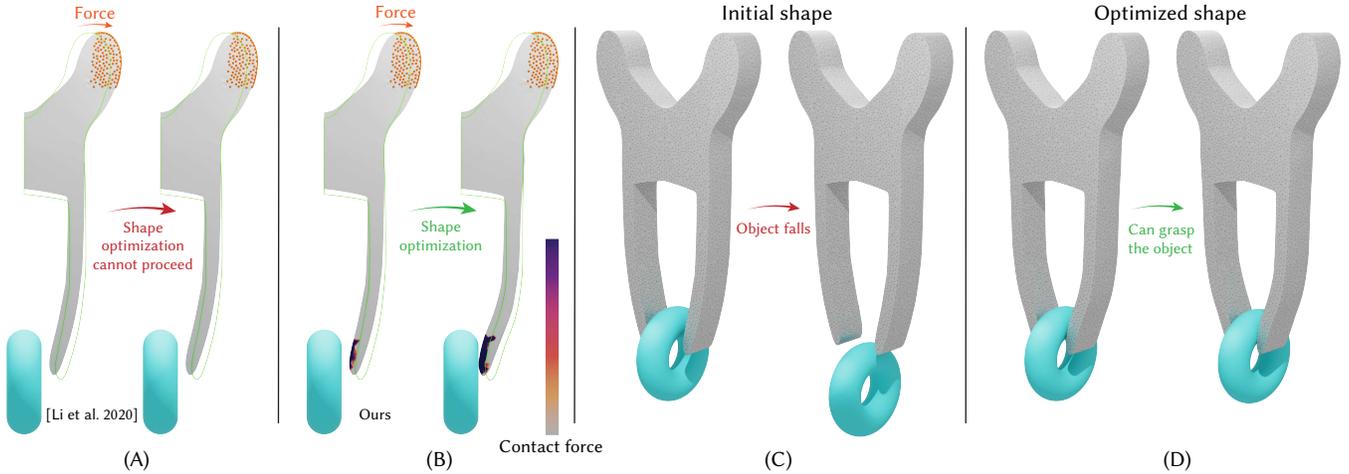


Fig. 32. Shape optimization of a plier for grasping a torus. The handles (orange nodes) are pulled outwards (specified as the Dirichlet boundary conditions) in the simulations. Rest shapes and deformed shapes are visualized as green contours and grey surfaces respectively. (A) Shape optimization with [Li et al. 2020] cannot proceed, since there is no contact force between the torus and plier due to the restriction that  $\hat{d}$  should be small to avoid spurious forces. (B) The contact force is maximized in the shape optimization with our method. (C) The initial shape fails to grasp the torus under gravity. (D) The optimized shape manages to grasp the torus under gravity.

integrand with a cubic spline, to localize it:

$$\iint_{M^2} \frac{\|P_f(x)(f(x) - f(y))\|^p}{\|f(x) - f(y)\|^{2p}} h_\epsilon(\|f(x) - f(y)\|) dx_f dy_f, \quad (15)$$

and compare with our method in Figure 33.

We observe that the TPE-based potential Equation (15) has a non-zero gradient everywhere on the surface if  $\epsilon_{\text{trg}}$  is larger than the edge length. On a single sphere (Figure 33), the gradient of Equation (15) is small but nonzero everywhere, while ours is exactly zero. E.g., in the twist-mat example, both formulations have large forces at places in contact. However, the gradient of our potential vanishes at places that do not need the contact barrier, while the gradient of Equation (15) does not vanish anywhere.

Furthermore, the TPE potentials are designed to optimize surfaces for smoothness: nonsmooth surfaces have infinite TPE [Yu et al. 2021]. In our case, our goal is to be able to preserve sharp features of the original shape.

We perform a convergence test for both formulations on a cube corner. Our potential is exactly zero for all 3 resolutions. For Equation (15), forces concentrate around (both convex and concave) sharp corners and increase under mesh refinement. As expected (and desired for smooth surface optimization) Equation (15) converges to infinity under mesh refinement for sharp features, while our potential converges to a finite number when there is a (convex or concave) corner (Figure 36) and no contact.

## 6.5 Parameter Dependence and Convergence

To study the effects of our parameters we vary  $\alpha$  and  $\epsilon_{\text{trg}}$ .

Lower values of  $\alpha$  (Figure 34) make the solution more accurate, as interaction sets are further from points, and spurious forces do not appear even for extreme deformations, but the potential is less smooth and the solver requires more iterations. This effect is more

pronounced in scenes with cusps; for simpler scenes, the difference in iteration count is less noticeable (Figure 35).

To study how much  $\alpha$  helps to remove unnecessary contact pairs, we compute the number of contact pairs for varying  $\alpha$  and  $\epsilon_{\text{trg}}$  on the fixed scene in Figure 29, at the frame obtained after running the simulation for 40 s of simulation time, and compare against IPC [Li et al. 2020] and Convergent IPC [Li et al. 2023] (Table 2). Convergent IPC has more contact pairs than IPC due to the extra contact pairs with negative weights introduced. Although our method has more types of contact pairs than IPC, even if the local minimum constraint is inactive ( $\alpha = 1$ ), it has fewer pairs than IPC due to the exterior direction constraint. The number of contact pairs drops significantly as  $\alpha$  decreases, one can use reasonably large  $\alpha$  for fast convergence while reducing the artifacts.

Table 2. **Number of contact pairs.** We compute the number of contact pairs in the fixed scene in Figure 29 at 40 s with varying  $\epsilon_{\text{trg}}$ , using IPC [Li et al. 2020], Convergent IPC [Li et al. 2023], and our method with different choices of  $\alpha$ .

$\epsilon_{\text{trg}}$	IPC	Convergent	1	0.8	0.5	0.1
0.001	226k	250k	215k	176k	128k	53k
0.002	372k	405k	322k	225k	155k	56k
0.005	1690k	1772k	830k	433k	273k	78k

We plot in Figure 35 the effect of  $\epsilon_{\text{trg}}$  on the number of iterations. As the support of the contact potential increases the problem becomes softer and the number of iterations decreases accordingly.

We also perform a convergence study for three different scenarios shown in Figure 36. We see convergence under mesh refinement in all of these scenes. Importantly, we use a fixed  $\epsilon_{\text{trg}}$  (unlike Convergent IPC [Li et al. 2023] which requires co-refinement of  $\hat{d}$ ).

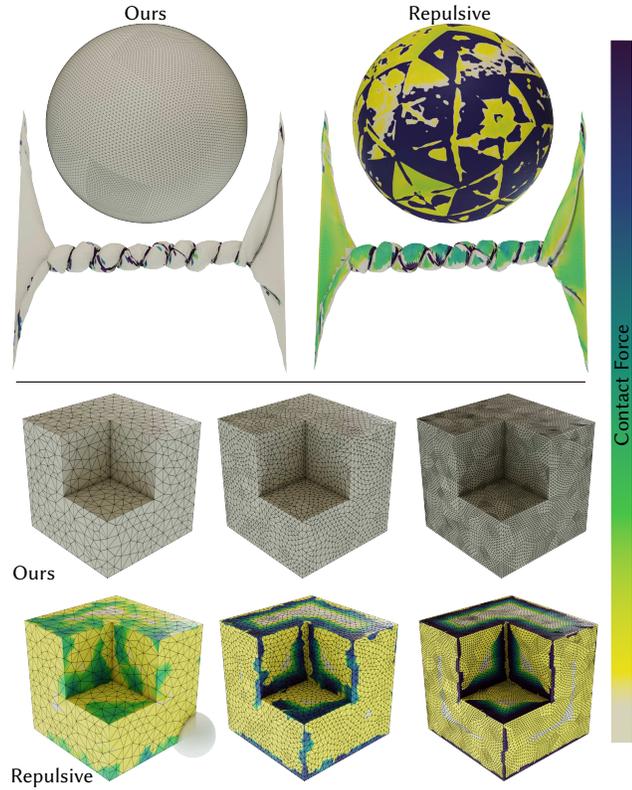


Fig. 33. Potential gradient distributions with our formulation and the repulsive formulation for three examples: (A) Single sphere with radius 1 m and  $\epsilon_{\text{trg}} = 0.1$  m, with magnitude range  $[10^{-10}, 10]$ . (B) Twist-mat with  $\epsilon_{\text{trg}} = 0.01$  m, with magnitude range  $[10^3, 10^{12}]$ . (C) Cube corner with  $\epsilon_{\text{trg}}$  visualized as a white sphere on the bottom left, with magnitude range  $[10^{-1}, 10^5]$ .

We note that when our discretization is applied to the piecewise linear surface, treating it as a piecewise smooth surface, the integrals for edge potentials and summations for vertex potentials are introduced, with the relative scale of potentials determined by the constant  $L$ . As the surface is refined, the scale of the constant also needs to be adjusted, for the refined mesh potential to approximate the smooth surface potential. We leave a rigorous study of convergence of the discrete potential to the potential of the limit smooth surface as future work.

## 6.6 Infinite Potential

Consider a corner hitting a plane, and suppose they are both refined (Figure 38 Right). In this case, the integral of the Convergent IPC [Li et al. 2023] potential over the plane will be finite, even if the apex of the corner is directly on the plane (violation of Requirement 2). Although each discretization will be infinite, the implication is that the total potential will decrease as we refine so that the actual distance will decrease to zero.

Figure 37 shows the configuration: for a gap of  $r$  and a distance from the closest point on the plane  $\Delta x$ , points along the square will

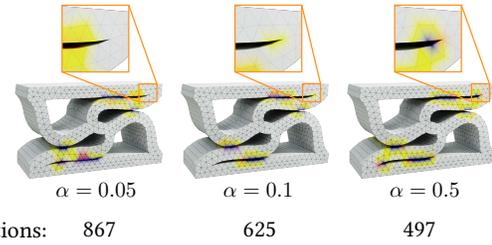


Fig. 34. **Parameter-study:  $\alpha$ .** Simulation of Figure 24 with different  $\alpha$  including the total number of solver iterations. As  $\alpha$  increases, the nonlinear problem becomes softer, hence fewer iterations. However, large  $\alpha$  introduces artifacts and spurious stresses in some cases.

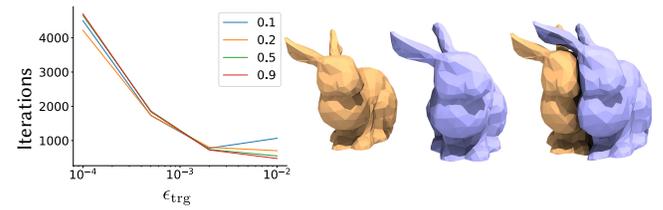


Fig. 35. **Parameter-study:  $\epsilon_{\text{trg}}$ .** Simulation of two bunnies colliding with varying  $\alpha$  and  $\epsilon_{\text{trg}}$ . We show the plot (left) of the number of iterations over  $\epsilon_{\text{trg}}$ , with a different  $\alpha$  for each curve, as well as the initial frame (middle) and the colliding frame (right) of the simulation. For simple scenes, for a large range of  $\alpha$ , the number of iterations is similar; as  $\epsilon_{\text{trg}}$  increases, it takes fewer iterations to converge.

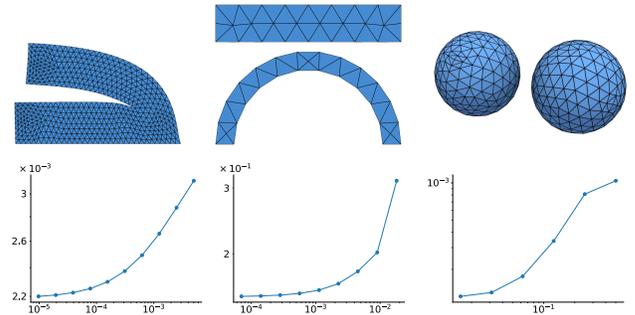


Fig. 36. Convergence of potential under mesh refinement in fixed scenes with fixed  $\epsilon_{\text{trg}}$ . We pick 2 configurations in 2D and 1 in 3D, start with the coarse mesh, and compute the potential under mesh refinement. The plots of contact barrier potential over edge length are shown.

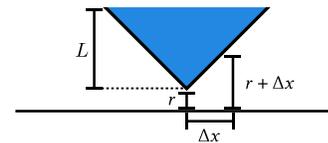


Fig. 37. Setup for corner hitting a plane discussed in Section 6.6.

be at a distance of  $\Delta x + r$ . Assume the potential<sup>2</sup> for points on the

<sup>2</sup>The extra quadratic term does not affect the conclusions.

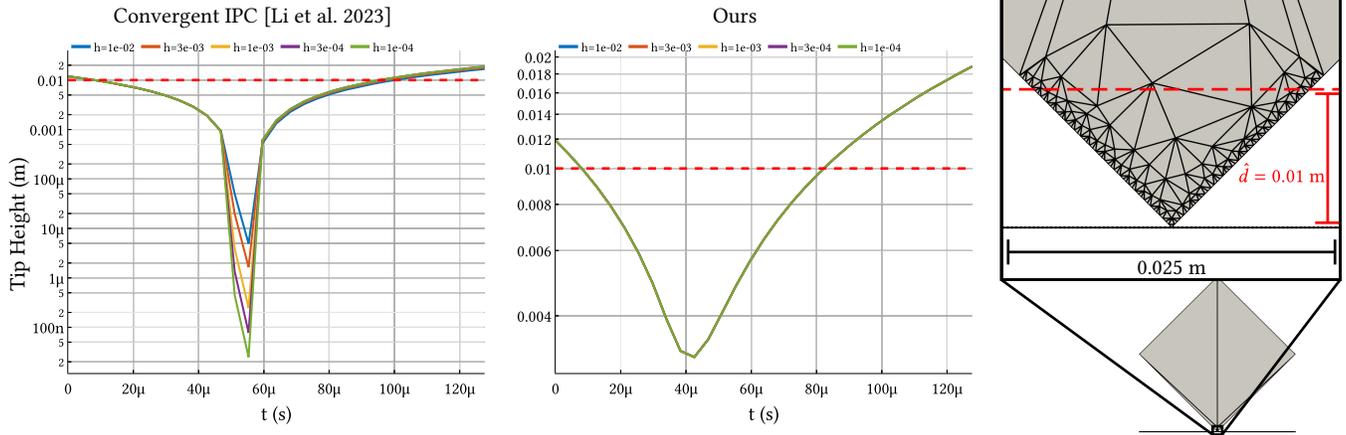


Fig. 38. **Finite Potential for Zero Distance.** A square impacts a plane at the corner of the square with a velocity of 234 m/s. We refine both meshes (only at the corner of the square for efficiency). Here we plot the height of the squares tip over time. Left: for Convergent IPC [Li et al. 2023], we see that the minimum distance shrinks with every refinement of the mesh. This is a consequence of the continuous model having a finite potential. Middle: due to our choice of barrier function, our method exhibits the same trajectory (all plots overlap) under refinement.

corner is just  $\log\left(\frac{\Delta x + r}{\hat{d}}\right)$ , then integrating over  $x$  we get

$$\int_0^L \log\left(\frac{x+r}{\hat{d}}\right) dx = (r+L) \log\left(\frac{r+L}{\hat{d}}\right) - \log\left(\frac{r}{\hat{d}}\right) r - L,$$

and for  $r \rightarrow 0$  (i.e. approaching zero distance), it has a finite limit of  $\log\left(\frac{L}{\hat{d}}\right) L - L$ .

The force (i.e., the potential derivative) at  $r = 0$  is  $\sim \log(r)$  though (i.e. is infinite), so for a finite force the static equilibrium problem will always have a valid solution. This is not the case however for a dynamic problem: if the kinetic energy exceeds the finite potential, then the barrier cannot prevent contact. We show this case in Figure 38. One can see the Convergent IPC model results in ever decreasing distances as the mesh is refined (without refining  $\hat{d}$ ). Our method in comparison exhibits the same trajectory for all levels of refinement. This is because we use a barrier function whose integral over the surface is not finite as the distance goes to zero and because the potential converges under mesh refinement.

Given that the minimum distance does not depend on the mesh resolution, one can pick  $\epsilon_{\text{trg}}$ , independent of the edge lengths, based on the desired accuracy of the contact handling.

## 7 CONCLUSION

Barrier potentials have enabled a qualitative improvement in the robustness of accurate simulation of deformable objects with complex contact. In this paper, we revisit barrier-based contact formulations and relate them to a family of potentials defined for a broad class of surfaces that satisfy a set of natural requirements. We demonstrate that applying these principles leads to a new formulation that alleviates some of the shortcomings of existing methods for barrier-based contact simulation.

## 7.1 Future work

Our derivation assumes closed surfaces without boundaries at several steps, which needs modifications for codimensional objects; a possible direction for future work is handling surfaces with boundaries, and codimensional surfaces. Besides, as Du et al. [2024] pointed out, IPC [Li et al. 2020] produces spurious tangential contact forces on a flat plane with non-uniform discretization, thus, it cannot simulate well the free-sliding of a cube on a plane without friction. Our method does not resolve this artifact. A potential solution is to use a high-order quadrature to better integrate the continuous formulation (in addition to including the closest-point quadrature), so that the potential is less affected by the non-uniform discretization. Another direction is deriving a higher-order discretization of our continuum formulation to increase accuracy.

## ACKNOWLEDGMENTS

This work was supported in part through the NYU IT High Performance Computing resources, services, and staff expertise. This work was also partially supported by the NSF CAREER award under Grant No. 1652515, the NSF grants OAC-2411349, OAC-1835712, CHS-1908767, CHS-1901091, IIS-2313156, a Sloan Fellowship, and a gift from Adobe Research.

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Department of Energy Computational Science Graduate Fellowship under Award Number(s) DE-SC0025528.

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

**Table 3. Simulation statistics.** For each simulation we report geometry, (minimum, average, and maximum edge length  $h$ ), time step  $\Delta t$ , material (Young's modulus  $E$ , Poisson ratio  $\nu$ , and density  $\rho$ ),  $\epsilon_{\text{trg}}$ , maximum memory used, as well as average timing and number of Newton iterations. A value of 0 for iterations indicates the optimization did not run because the initial configuration was at a force equilibrium (i.e., no spurious forces at rest). Timings and iterations of [Li et al. 2020] are shown in parentheses.

Example	# nodes, # cells	$h$ (m) (min, avg, max)	$\Delta t$ (s)	$E$ (kPa), $\nu$ , $\rho$ (kg/m <sup>3</sup> )	$\epsilon_{\text{trg}}$ (m)	$\alpha$	memory (MB)	timing (s), iterations (per timestep)
Slit block (2D) (Supp. Video)	199, 325	0.04, 0.18, 0.34	0.0047	4, 0.2, 100	0.1	0.1	19	0.07, 0
Slit block (3D) (Fig 20)	1058, 3871	0.01, 0.04, 0.13	0.0047	0.1, 0.2, 10	0.0125	0.1	93	0.14, 0
Fillet block (2D) (Fig 17)	445, 672	0.01, 0.04, 0.21	0.0047	4, 0.2, 100	0.1	0.1	26	0.09, 0
Fillet block (3D) (Fig 1)	1003, 3824	0.02, 0.19, 0.42	0.0047	0.1, 0.2, 10	0.07	0.1	92	0.14, 0
Compressed block (2D) (Supp. Video)	356, 630	0.09, 0.12, 0.22	0.0047	10, 0, 100	0.04	0.1	63	0.17, 2.3
Compressed block (3D) (Fig 21)	2169, 9799	0.03, 0.10, 0.18	0.0047	10, 0, 100	0.0125	0.1	644	0.78, 2.5
Armadillo bar (Figure 19)	91777, 446158	0.00076, 0.0090, 0.084	0.02	10, 0.49, 1	0.001	0.5	9675	22.2, 3.4 (64.4, 10.4)
Two blocks (2D) (Supp. Video)	708, 1260	0.09, 0.12, 0.22	0.0047	4, 0.2, 100	0.1	0.1	24	0.08, 0
Two blocks (3D) (Fig 20)	580, 2079	0.04, 0.21, 0.55	0.0047	10, 0, 100	0.025	0.1	61	0.14, 0
Spherical balloon (3D) (Fig 22)	421, 701	0.04, 0.11, 0.18	0.0047	4000, 0, 100	0.02	0.1	722	0.67, 1
Donut balloon (3D) (Fig 23)	17122, 52601	0.013, 0.3, 0.086	0.01	1000, 0.48, 1000	0.01	0.1	2205	28.9, 17.1
Cusp compression (Fig 24)	3761, 12777	0.024, 0.047, 0.089	0.01	1000, 0.3, 100	0.015	0.05	501	5.02, 15.8
Dolphin funnel (Fig 27)	4074, 10511	0.0017, 0.020, 0.081	0.025	10, 0.4, 1000	0.001	1	1403	15.4, 58.1
Trash compactor (Fig 28)	6611, 21696	0.00033, 0.049, 0.36	0.01	10, 0.4, 1000	0.001	0.8	2833	228.9, 155 (196, 205)
Mat twist (Fig 29)	45000, 133206	0.0067, 0.0088, 0.0126	0.04	20, 0.4, 1000	0.002	0.8	17683	697.7, 61.8 (610.3, 116)
Monkey saddle (Figure 31)	67320, 198282	0.00032, 0.0080, 0.058	0.005	1e5, 0.48, 1000	0.0002	0.8	5376	66.6, 21.3 (102.6, 33.6)

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

## REFERENCES

- Christie Alappat, Achim Basermann, Alan R. Bishop, Holger Fehske, Georg Hager, Olaf Schenk, Jonas Thies, and Gerhard Wellein. 2020. A Recursive Algebraic Coloring Technique for Hardware-Efficient Symmetric Sparse Matrix-Vector Multiplication. *ACM Transactions on Parallel Computing* 7, 3, Article 19 (June 2020), 37 pages.
- Pierre Alart and Alain Curnier. 1991. A mixed formulation for frictional contact problems prone to Newton like solution methods. *Computer Methods in Applied Mechanics and Engineering* 92, 3 (Nov. 1991), 353–375.
- M.D. Alaydin, D.J. Benson, and Y. Bazilevs. 2021. An updated Lagrangian framework for Isogeometric Kirchhoff–Love thin-shell analysis. *Computer Methods in Applied Mechanics and Engineering* 384 (2021), 113977.
- Francisco Armero and Eva Petőcz. 1998. Formulation and analysis of conserving algorithms for frictionless dynamic contact/impact problems. *Computer methods in applied mechanics and engineering* 158, 3-4 (1998), 269–300.
- F.B. Belgacem, P. Hild, and P. Laborde. 1998. The mortar finite element method for contact problems. *Mathematical and Computer Modelling* 28, 4 (1998), 263–271. Recent Advances in Contact Mechanics.
- Ted Belytschko and Mark O Neal. 1991. Contact-impact by the pinball algorithm with penalty and Lagrangian methods. *Internat. J. Numer. Methods Engrg.* 31, 3 (1991), 547–572.
- David J. Benson and John O. Hallquist. 1990. A single surface contact algorithm for the post-buckling analysis of shell structures. *Computer Methods in Applied Mechanics and Engineering* 78, 2 (1990), 141–163.
- Matthias Bollhöfer, Aryan Eftekhari, Simon Scheidegger, and Olaf Schenk. 2019. Large-scale Sparse Inverse Covariance Matrix Estimation. *SIAM Journal on Scientific Computing* 41, 1 (2019), 380–401.
- Matthias Bollhöfer, Olaf Schenk, Radim Janalik, Steve Hamm, and Kiran Gullapalli. 2020. State-of-the-Art Sparse Direct Solvers. *Parallel Algorithms in Computational Science and Engineering* (2020), 3–33.
- Gregory Buck and Jeremy Orloff. 1995. A simple energy function for knots. *Topology and its Applications* 61, 3 (1995), 205–214.
- Nicholas J Carpenter, Robert L Taylor, and Michael G Katona. 1991. Lagrange constraints for transient finite element surface contact. *International journal for numerical methods in engineering* 32, 1 (1991), 103–128.
- Anka He Chen, Ziheng Liu, Yin Yang, and Cem Yuksel. 2024. Vertex Block Descent. *ACM Trans. Graph.* 43, 4, Article 116 (July 2024), 16 pages. <https://doi.org/10.1145/3658179>
- PW Christensen, Anders Klarbring, Jong-Shi Pang, and Niclas Strömberg. 1998. Formulation and comparison of algorithms for frictional contact problems. *Internat. J. Numer. Methods Engrg.* 42, 1 (1998), 145–173.
- Gilles Daviet. 2020. Simple and scalable frictional contacts for thin nodal objects. *ACM Trans. Graph.* 39, 4, Article 61 (Aug. 2020), 16 pages. <https://doi.org/10.1145/3386569.3392439>
- Peter Deuffhard, Rolf Krause, and Susanne Ertel. 2008. A contact-stabilized Newmark method for dynamical contact problems. *Internat. J. Numer. Methods Engrg.* 73, 9 (2008), 1274–1290.
- Manfredo P Do Carmo. 2016. *Differential geometry of curves and surfaces: revised and updated second edition*. Courier Dover Publications.
- Yinwei Du, Yue Li, Stelian Coros, and Bernhard Thomaszewski. 2024. Robust and Artefact-Free Deformable Contact with Smooth Surface Representations. *Computer Graphics Forum* (2024). <https://doi.org/10.1111/cgf.15187>
- Kenny Erleben. 2018. Methodology for Assessing Mesh-Based Contact Point Methods. *ACM Transactions on Graphics* 37, 3, Article 39 (July 2018), 30 pages.
- Zachary Ferguson et al. 2020. IPC Toolkit. <https://ipc-sim.github.io/ipc-toolkit/>
- Zachary Ferguson, Pranav Jain, Denis Zorin, Teseo Schneider, and Daniele Panozzo. 2023. High-Order Incremental Potential Contact for Elastodynamic Simulation on Curved Meshes. In *ACM SIGGRAPH 2023 Conference Proceedings* (Los Angeles, CA, USA) (SIGGRAPH '23). Association for Computing Machinery, New York, NY, USA, 11 pages.
- Zachary Ferguson, Minchen Li, Teseo Schneider, Francisca Gil-Ureta, Timothy Langlois, Chenfanfu Jiang, Denis Zorin, Danny M. Kaufman, and Daniele Panozzo. 2021. Intersection-Free Rigid Body Dynamics. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 40, 4, Article 183 (July 2021), 16 pages.
- Felipe Fernandez, Michael A. Puso, Jerome Solberg, and Daniel A. Tortorelli. 2020. Topology optimization of multiple deformable bodies in contact with large deformations. *Computer Methods in Applied Mechanics and Engineering* 371 (2020), 113288.
- Arvi Gjoka, Espen Knoop, Moritz Bächer, Denis Zorin, and Daniele Panozzo. 2024. Soft Pneumatic Actuator Design using Differentiable Simulation. In *ACM SIGGRAPH 2024 Conference Papers* (Denver, CO, USA) (SIGGRAPH '24). Association for Computing Machinery, New York, NY, USA, Article 106, 11 pages. <https://doi.org/10.1145/3641519.3657467>
- Gaël Guennebaud, Benoît Jacob, et al. 2010. Eigen v3. <http://eigen.tuxfamily.org>
- David Harmon, Etienne Vouga, Breannan Smith, Rasmus Tamstorf, and Eitan Grinspun. 2009. Asynchronous contact mechanics. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 28, 3 (2009), 1–12.
- Michael Hiermeier, Wolfgang A Wall, and Alexander Popp. 2018. A truly variationally consistent and symmetric mortar-based contact formulation for finite deformation solid mechanics. *Computer Methods in Applied Mechanics and Engineering* 342 (2018), 532–560.
- Kemeng Huang, Floyd M. Chitalu, Huancheng Lin, and Taku Komura. 2024a. GIPC: Fast and stable Gauss-Newton optimization of IPC barrier energy. *ACM Trans. Graph.* (jan 2024). <https://doi.org/10.1145/3643028> Just Accepted.
- Zizhou Huang, Davi Colli Tozoni, Arvi Gjoka, Zachary Ferguson, Teseo Schneider, Daniele Panozzo, and Denis Zorin. 2024b. Differentiable solver for time-dependent deformation problems with contact. *ACM Trans. Graph.* 43, 3, Article 31 (May 2024), 30 pages. <https://doi.org/10.1145/3657648>

- S. Hübner and B.I. Wohlmuth. 2006. *Mortar methods for contact problems*. Springer Berlin Heidelberg, Berlin, Heidelberg, 39–47.
- Amin Joodaky. 2020. *Mechanics and Design of Polymeric Metamaterial Structures for Shock Absorption Applications*. Ph.D. Dissertation. Purdue University Graduate School.
- Jonathan M. Kaldor, Doug L. James, and Steve Marschner. 2008. Simulating knitted cloth at the yarn level. In *ACM SIGGRAPH 2008 Papers* (Los Angeles, California) (SIGGRAPH '08). Association for Computing Machinery, New York, NY, USA, Article 65, 9 pages.
- David Kamensky, Fei Xu, Chung-Hao Lee, Jinhui Yan, Yuri Bazilevs, and Ming-Chen Hsu. 2018. A contact formulation based on a volumetric potential: Application to isogeometric simulations of atrioventricular valves. *Computer Methods in Applied Mechanics and Engineering* 330 (2018), 522–546.
- Couro Kane, Eduardo A Repetto, Michael Ortiz, and Jerrold E Marsden. 1999. Finite element analysis of nonsmooth contact. *Computer Methods in Applied Mechanics and Engineering* 180, 1-2 (1999), 1–26.
- Danny M. Kaufman, Shinjiro Sueda, Doug L. James, and Dinesh K. Pai. 2008. Staggered Projections for Frictional Contact in Multibody Systems. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 27, 5 (2008), 1–11.
- Theodore Kim and David Eberle. 2022. Dynamic Deformables: Implementation and Production Practicalities (Now with Code!). In *ACM SIGGRAPH 2022 Courses* (Vancouver, British Columbia, Canada). ACM, New York, NY, Article 7, 259 pages.
- G Kloosterman, Rudolf MJ van Damme, Antonius H van den Boogaard, and J Huetink. 2001. A geometrical-based contact algorithm using a barrier method. *Internat. J. Numer. Methods Engrg.* 51, 7 (2001), 865–882.
- Alexander Konyukhov and Karl Schweizerhof. 2013. *Computational Contact Mechanics: Geometrically Exact Theory for Arbitrary Shaped Bodies*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-31531-2>
- Lei Lan, Danny M. Kaufman, Minchen Li, Chenfanfu Jiang, and Yin Yang. 2022. Affine Body Dynamics: Fast, Stable and Intersection-Free Simulation of Stiff Materials. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 41, 4, Article 67 (July 2022), 14 pages.
- Lei Lan, Yin Yang, Danny Kaufman, Junfeng Yao, Minchen Li, and Chenfanfu Jiang. 2021. Medial IPC: Accelerated Incremental Potential Contact with Medial Elastics. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 40, 4, Article 158 (July 2021), 16 pages.
- Egor Larionov, Ye Fan, and Dinesh K. Pai. 2021. Frictional Contact on Smooth Elastic Solids. *ACM Trans. Graph.* 40, 2, Article 15 (apr 2021), 17 pages.
- TA Laursen and GR Love. 2002. Improved implicit integrators for transient impact problems—geometric admissibility within the conserving framework. *Internat. J. Numer. Methods Engrg.* 53, 2 (2002), 245–274.
- Minchen Li, Zachary Ferguson, Teseo Schneider, Timothy Langlois, Denis Zorin, Daniele Panozzo, Chenfanfu Jiang, and Danny M. Kaufman. 2020. Incremental Potential Contact: Intersection- and Inversion-free Large Deformation Dynamics. *ACM Transactions on Graphics* 39, 4, Article 49 (July 2020), 20 pages.
- Minchen Li, Zachary Ferguson, Teseo Schneider, Timothy Langlois, Denis Zorin, Daniele Panozzo, Chenfanfu Jiang, and Danny M. Kaufman. 2023. Convergent Incremental Potential Contact. [arXiv:2307.15908 \[math.NA\]](https://arxiv.org/abs/2307.15908)
- Minchen Li, Danny M. Kaufman, and Chenfanfu Jiang. 2021. Codimensional incremental potential contact. *ACM Trans. Graph.* 40, 4, Article 170 (jul 2021), 24 pages. <https://doi.org/10.1145/3450626.3459767>
- M. Macklin, K. Erleben, M. Müller, N. Chentanez, S. Jeschke, and T. Y. Kim. 2020. Primal/Dual Descent Methods for Dynamics. In *Proceedings of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA '20)*. Eurographics Association, Goslar, DEU, Article 9, 12 pages.
- Miles Macklin, Kenny Erleben, Matthias Müller, Nuttapong Chentanez, Stefan Jeschke, and Viktor Makoviychuk. 2019. Non-Smooth Newton Methods for Deformable Multi-Body Dynamics. *ACM Transactions on Graphics* 38, 5, Article 140 (July 2019), 20 pages.
- Christoph Meier, Wolfgang A. Wall, and Alexander Popp. 2017. A unified approach for beam-to-beam contact. *Computer Methods in Applied Mechanics and Engineering* 315 (2017), 972–1010. <https://doi.org/10.1016/j.cma.2016.11.028>
- Miguel Otaduy, Rasmus Tamstorf, Denis Steinemann, and Markus Gross. 2009. Implicit Contact Handling for Deformable Objects. *Computer Graphics Forum* 28 (April 2009), 559–568.
- G. Pietrzak and A. Curnier. 1999. Large deformation frictional contact mechanics: continuum formulation and augmented Lagrangian treatment. *Computer Methods in Applied Mechanics and Engineering* 177, 3 (1999), 351–381.
- A. Popp, B. I. Wohlmuth, M. W. Gee, and W. A. Wall. 2012. Dual Quadratic Mortar Finite Element Methods for 3D Finite Deformation Contact. *SIAM Journal on Scientific Computing* 34, 4 (2012), B421–B446.
- Michael A. Puso, T.A. Laursen, and Jerome Solberg. 2008. A segment-to-segment mortar contact method for quadratic elements and large deformations. *Computer Methods in Applied Mechanics and Engineering* 197, 6 (2008), 555–566.
- Michael A. Puso and Tod A. Laursen. 2004. A mortar segment-to-segment contact method for large deformation solid mechanics. *Computer Methods in Applied Mechanics and Engineering* 193, 6 (2004), 601–629.
- M. A. Puso, J. H. Porter, and T. Slavik. 2024. Structure preserving and energy dissipative contact approaches for implicit dynamics. *Internat. J. Numer. Methods Engrg.* 125, 10 (2024), e7454.
- Alan Marquez Razon, Yizhou Chen, Yushan Han, Steven Gagniere, Michael Tupek, and Joseph Teran. 2023. A Linear and Angular Momentum Conserving Hybrid Particle/Grid Iteration for Volumetric Elastic Contact. *Proc. ACM Comput. Graph. Interact. Tech.* 6, 3, Article 44 (Aug. 2023), 25 pages. <https://doi.org/10.1145/3606924>
- Josua Sassen, Henrik Schumacher, Martin Rumpf, and Keenan Crane. 2024a. Repulsive Shells. *ACM Trans. Graph.* 43, 4 (2024).
- Josua Sassen, Henrik Schumacher, Martin Rumpf, and Keenan Crane. 2024b. Repulsive Shells. *ACM Trans. Graph.* 43, 4 (2024).
- Roger A Sauer and Laura De Lorenzis. 2013. A computational contact formulation based on surface potentials. *Computer Methods in Applied Mechanics and Engineering* 253 (2013), 369–395.
- Teseo Schneider, Jérémie Dumas, Xifeng Gao, Denis Zorin, and Daniele Panozzo. 2019. PolyFEM. <https://polyfem.github.io/>.
- Xing Shen, Runyuan Cai, Mengxiao Bi, and Tangjie Lv. 2024. Preconditioned Nonlinear Conjugate Gradient Method for Real-time Interior-point Hyperelasticity. In *ACM SIGGRAPH 2024 Conference Papers* (Denver, CO, USA) (SIGGRAPH '24). Association for Computing Machinery, New York, NY, USA, Article 96, 11 pages. <https://doi.org/10.1145/3641519.3657490>
- J.C. Simo and T.A. Laursen. 1992. An augmented lagrangian treatment of contact problems involving friction. *Comput. & Structures* 42, 1 (1992), 97–116.
- Breannan Smith, Fernando De Goes, and Theodore Kim. 2018. Stable Neo-Hookean Flesh Simulation. *ACM Trans. Graph.* 37, 2, Article 12 (mar 2018), 15 pages. <https://doi.org/10.1145/3180491>
- Pawel Strzelecki and Heiko von der Mosel. 2013. Tangent-Point Repulsive Potentials for a Class of Non-smooth m-dimensional Sets in  $\mathbb{R}^n$ . Part I: Smoothing and Self-avoidance Effects. *Journal of Geometric Analysis* 23, 3 (2013), 1085–1139.
- Robert Taylor and Peter Wriggers. 1999. *Smooth surface discretization for large deformation frictionless contact*. Technical Report. University of California Berkeley.
- Robert L Taylor and Panayiotis Papadopoulos. 1993. On a finite element method for dynamic contact/impact problems. *International journal for numerical methods in engineering* 36, 12 (1993), 2123–2140.
- İ. Temizer, M.M. Abdalla, and Z. Gürdal. 2014. An interior point method for isogeometric contact. *Computer Methods in Applied Mechanics and Engineering* 276 (2014), 589–611.
- İ. Temizer, P. Wriggers, and T.J.R. Hughes. 2012. Three-dimensional mortar-based frictional contact treatment in isogeometric analysis with NURBS. *Computer Methods in Applied Mechanics and Engineering* 209-212 (2012), 115–128.
- Mickeal Verschoor and Andrei C Jalba. 2019. Efficient and accurate collision response for elastically deformable models. *ACM Transactions on Graphics* 38, 2, Article 17 (March 2019), 20 pages.
- Didier Vola, Elaine Pratt, Michel Jean, and Michel Raous. 1998. Consistent time discretization for a dynamical frictional contact problem and complementarity techniques. *Revue Européenne des éléments finis* 7, 1-3 (1998), 149–162.
- Etienne Vouga, David Harmon, Rasmus Tamstorf, and Eitan Grinspun. 2011. Asynchronous Variational Contact Mechanics. <https://github.com/evouga/collisiondetection>. *Computer Methods in Applied Mechanics and Engineering* 200, 25 (2011), 2181–2194. <https://github.com/evouga/collisiondetection> <https://github.com/evouga/collisiondetection>
- Jingyi Wang, Jerome Solberg, Mike A. Puso, Eric B. Chin, and Cosmin G. Petra. 2024. Design optimization in unilateral contact using pressure constraints and Bayesian optimization. [arXiv:2405.03081 \[math.NA\]](https://arxiv.org/abs/2405.03081) <https://arxiv.org/abs/2405.03081>
- Peter Wriggers. 1995. Finite Element Algorithms for Contact Problems. *Archives of Computational Methods in Engineering* 2 (Dec. 1995), 1–49.
- Peter Wriggers and Tod A Laursen. 2006. *Computational contact mechanics*. Vol. 2. Springer.
- Tianyi Xie, Minchen Li, Yin Yang, and Chenfanfu Jiang. 2023. A Contact Proxy Splitting Method for Lagrangian Solid-Fluid Coupling. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 42, 4 (2023), 1–14.
- Jonathan Youett, Oliver Sander, and Ralf Kornhuber. 2019. A globally convergent filter-trust-region method for large deformation contact problems. *SIAM Journal on Scientific Computing* 41, 1 (2019), B114–B138.
- Chris Yu, Caleb Brakensiek, Henrik Schumacher, and Keenan Crane. 2021. Repulsive Surfaces. *ACM Trans. Graph.* 40, 6 (2021).

## A DISTANCE FUNCTION MOLLIFICATION

When differentiating the potential, since the closest points are used for evaluating the quadrature, one needs to differentiate through the closest points with respect to the vertex positions of the mesh.

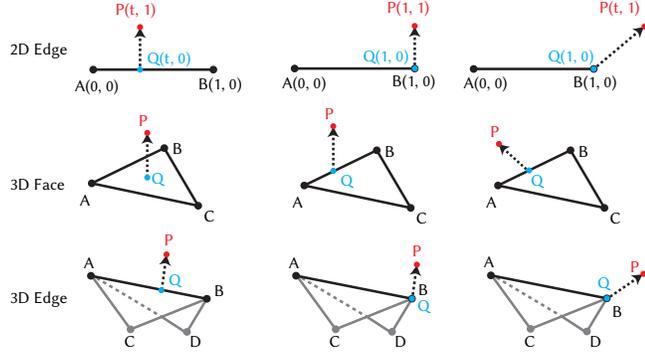


Fig. 39. Non-smoothness of the closest point position. From left to right, the closest point  $Q$  of a point  $P$  outside of an edge/face moves from the interior of the edge/face to its boundary. The coordinate of  $Q$  does not depend smoothly on  $P$  as  $P$  crosses the boundary.

However, the closest points may not smoothly depend on the vertex positions. For example, in 2D (top row in Figure 39), the closest point on the unit edge  $AB$  to a point  $P = (t, 1)$  ( $t > 0$ ) is  $Q = (\min(t, 1), 0)$ , which is only  $C^0$  continuous at  $t = 1$ . Below we introduce new types of mollification different from [Li et al. 2020] to resolve this problem. First, we define  $d(\cdot, \cdot)$  as the shortest distance between two primitives (vertex, edge, or face), e.g.  $d(P, A)$  denotes the distance between two points  $P$  and  $A$ ,  $d(P, AB)$  denotes the distance of point  $P$  to edge  $AB$ . We reuse the  $C^1$  continuous mollifier function from [Li et al. 2020]:

$$h(z) := \begin{cases} z(2-z) & 0 \leq z < 1 \\ 1 & 1 \leq z \end{cases} \quad h_c(s) := h\left(\frac{s-1}{c}\right),$$

we use  $c = 0.01$  in our examples. In the definition of  $M(x, y)$ , we denote  $P = f(x)$  and  $Q = f(y)$  as the pair of closest points.

For Edge-Vertex contact (either in 2D or 3D), suppose  $Q$  is on an edge  $AB$ , we define the mollification as

$$M(x, y) := h_c\left(\frac{d(A, P)}{d(P, AB)}\right)h_c\left(\frac{d(B, P)}{d(P, AB)}\right),$$

which vanishes as the closest point  $Q$  approaches either  $A$  or  $B$  (Figure 39). For Face-Vertex contact, let the vertices of the triangle face be  $A, B, C$ , and the vertex outside be  $P$  (Figure 39). Similarly, the mollification is

$$M(x, y) := h_c\left(\frac{d(P, AB)}{d(P, ABC)}\right)h_c\left(\frac{d(P, AC)}{d(P, ABC)}\right)h_c\left(\frac{d(P, BC)}{d(P, ABC)}\right),$$

which vanishes as  $Q$  on the triangle  $ABC$  approaches the boundary of the triangle.

For Edge-Edge contact, the smoothness is more complicated since  $P$  is not smooth anymore. In Figure 40, we observe that the closest points  $P, Q$  are non-smooth only when the Edge-Edge distance reduces to Vertex-Edge or Vertex-Vertex distance. E.g. in Figure 40, line  $BP$  overlaps with  $QP$  when  $Q$  is only  $C^0$  continuous. Inspired by this, we define our mollification as

$$M(x, y) := h_c\left(\frac{d(B, CD)}{d(AB, CD)}\right)h_c\left(\frac{d(A, CD)}{d(AB, CD)}\right)h_c\left(\frac{d(C, AB)}{d(AB, CD)}\right)h_c\left(\frac{d(D, AB)}{d(AB, CD)}\right),$$

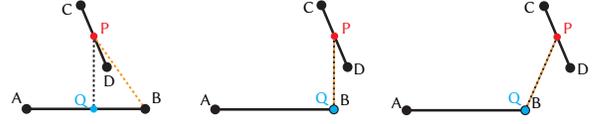


Fig. 40. Non-smoothness of the Edge-Edge contact. Point  $Q$  and  $P$  are the closest point pair between edge  $AB$  and  $CD$ , the orange line is the closest direction between vertex  $B$  and edge  $CD$ . As edge  $CD$  moves from left to right,  $Q$  is only  $C^0$  continuous when it reaches  $B$ . At the same time,  $BP$  starts to overlap with  $QP$ .

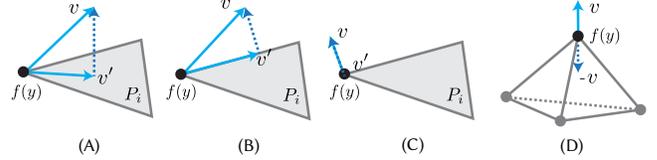


Fig. 41. (A) The closest point  $v'$  on face  $P_i$  is in the interior of  $P_i$ . (B) The closest point  $v'$  is on the boundary of  $P_i$ . (C) The closest point  $v'$  overlaps with  $f(y)$ . (D) The closest points on all faces overlap with  $f(y)$ . In this case, instead of deciding if  $v$  points inside, we flip  $v$  and decide if  $-v$  points outside.

where  $d(PQ, UV)$  is the distance between two edges  $PQ$  and  $UV$ . The mollification is  $C^1$  smooth and vanishes when  $Q$  overlaps with  $A$  or  $B$ . Note that the parallel edge-edge mollification in [Li et al. 2020], which was introduced to avoid non-smooth distance with almost parallel edges, is not needed anymore, since our mollification also vanishes when the edges are in parallel.

All the distance functions used in the mollifications above are  $C^\infty$  smooth as long as there is no intersection or degenerate element, and  $h_c$  is  $C^1$  smooth, so the mollifications are  $C^1$  smooth (same as [Li et al. 2020]).

*Remark.* The proposed mollifications make the contact potential vanish in more cases than in [Li et al. 2020], however, it still guarantees intersection-free, i.e. the Requirement 2 (*Barrier*) is still satisfied: When an Edge-Vertex contact vanishes due to mollification, the closest point on the edge is at one of the endpoints and the Vertex-Vertex contact becomes active; when a Face-Vertex (or Edge-Edge) contact vanishes, some Edge-Vertex or Vertex-Vertex contact is active.

## B DETERMINING ORIENTATION IN PIECEWISE LINEAR CASE

In this appendix, we detail our method for determining if  $v := (f(y) - f(x))_+$  at a point  $y$  of an oriented piecewise linear surface points inside. For high-order surfaces, one only needs to consider local tangent planes at  $y$ , so the problem reduces to the piecewise linear case.

As shown in Figure 15, when contact happens (i.e. as  $f(x)$  approaches  $f(y)$ ), if the cone around  $f(y)$  is convex, one may conclude that  $n_i(y) \cdot v \leq 0$  for all  $i$ . However, in general,  $n_i(y) \cdot v \leq 0$  only holds for some  $i$ , and  $\min_i n_i(y) \cdot v < 0$  may be extremely close to zero. As a consequence, even though one can simply drop contact pairs based on normals for [Li et al. 2020] in some simple scenes

(Figure 22), in general, the filtering on normal directions has to be constructed to be both numerically robust and differentiable.

We first discuss the robust and exact algorithm (not differentiable). For convenience, let  $f(y)$  be the origin of the coordinate system. let  $e_i$  ( $i = 1, \dots, M$ ) be the vector pointing from vertex  $f(y)$  to its  $i$ -th neighboring vertex,  $P_i$  be the face formed by point  $f(y)$  and vectors  $e_i$  and  $e_{i-1}$  (let  $e_0 = e_M$ ). Denote  $n_i$  as the normal of  $P_i$ , the  $\{e_i\}$  follow the ordering so that  $(e_{i-1} \times e_i) \cdot n_i > 0$ . We assume that the length of  $e_i$  is much larger than  $v$  since we only care about the direction of  $v$ . In the trivial case, where every  $P_i$  is perpendicular to  $v$ , i.e. all  $P_i$  share the same plane, we can define

$$\Phi^e(x, y) := -v \cdot n,$$

where  $n$  is the normal shared by all  $P_i$ . Below we assume that there exists  $P_i$  that is not perpendicular to  $v$ .

We first find the closest point to  $v$  on the cone surface. In Figure 41, we find the closest point to  $v$  on each face  $P_i$ , and denote the projected vector as  $v'_i$ . Suppose

$$j = \operatorname{argmin}_i \|v - v'_i\|,$$

which means  $v'_j$  is the closest point to  $v$  over the whole surface of the cone formed by  $f(y)$  and its 1-ring. As a consequence, the line segment connecting  $v'_j$  and  $v$  doesn't intersect with any other faces of the cone, i.e. the segment lies completely inside or outside of the cone.

- If  $v'_j$  is on the interior of face  $P_j$ , we can define

$$\Phi^e(x, y) := -v \cdot n_j,$$

where  $n_j$  is the normal of  $P_j$ .

- If  $v'_j$  is shared by two faces, i.e. it lies on an edge, the problem reduces to 2D (Figure 14): we project the space onto the plane perpendicular to the edge  $e_j$ , and the problem becomes checking if 2D vector  $\tilde{v}$  is inside the sector bounded by  $\tilde{e}_1, \tilde{e}_2$ . Since  $\tilde{e}_1 \times \tilde{e}_2 > 0$  (by construction of face normals),  $\tilde{v}$  points inside if and only if

$$\tilde{e}_1 \rightarrow \tilde{e}_2 \rightarrow \tilde{v}$$

is counter-clockwise. If  $\tilde{v}, \tilde{e}_1, \tilde{e}_2$  are all unit-length, then it's equivalent to

$$\Phi^e(x, y) := (\tilde{v} - \tilde{e}_1) \times (\tilde{v} - \tilde{e}_2) > 0.$$

- If  $v'_j = f(y)$ , i.e. the closest point to  $v$  is the center of the cone. In this case, the closest point to  $-v$  on the cone surface must not be  $f(y)$ , otherwise all faces  $P_i$  are perpendicular to  $v$  and it becomes the trivial case we discussed above. Thus we can perform the above algorithm on  $-v$  instead: Deciding if  $v$  points inside is equivalent to deciding if  $-v$  points outside.

*Remark.* If the closest point  $v'_i$  to  $v$  on some face  $P_i$  is not at  $f(y)$ , then  $\|v - v'_i\| < \|v - f(y)\|$ , because  $f(y)$  is on all faces and the distance (which is a strictly convex function) reaches the minimum at  $v'_i$  on face  $P_i$ .

The  $\Phi^e(x, y)$  defined above satisfies that  $\Phi^e(x, y) > 0$  if and only if  $v$  points inside the cone, and  $\Phi^e(x, y) = 0$  if and only if  $v$  lies on the surface of the cone.

We summarize the algorithm in Algorithm 1 and describe the purpose of each function below.

- **DETERMINEINSIDE3D**( $v, \{e_i\}, \{n_i\}$ ) returns a flag indicating if  $v$  points inside the cone bounded by rays  $\{e_i\}$ .  $n_i$  is the normal of face bounded by rays  $e_{i-1}, e_i$ .
- **CLOSESTPOINT**( $v, \{e_i\}$ ) finds the closest point to  $v$  on the cone surface.
- **FACECLOSESTPOINT**( $v, a, b$ ) returns the closest point to  $v$  on the angle sector spanned by  $a, b$ , and a flag indicating whether the closest point is on the boundary or in the interior of the sector.
- **PROJECT**( $a, b$ ) projects vector  $a$  onto the plane perpendicular to vector  $b$ , and normalize the projected vector.
- **EDGECLOSESTDISTANCE**( $v, e$ ) returns the distance of  $v$  to the ray  $e$ .

---

**Algorithm 1** Determine if a vector  $v$  is inside a cone bounded by a set of rays  $\{e_i\}$ .

---

```

function DETERMINEINSIDE3D( $v, \{e_i\}, \{n_i\}$ )
  [ $v'$ , type,  $j$ ]  $\leftarrow$  CLOSESTPOINT( $v, \{e_i\}$ )
  if  $v' == 0$  then                                      $\triangleright$  The closest point is  $f(y)$ 
    if  $\{n_i\}$  are close to the same vector then
      return  $-v \cdot n_0 > 0$ 
    else
      return DETERMINEINSIDE3D( $-v, \{e_i\}, \{n_i\}$ ) == False
    end if
  end if
  if type == Interior then                              $\triangleright$  The closest point is inside a face
    return  $-v \cdot n_j > 0$ 
  else                                                  $\triangleright$  The closest point is on an edge
    for  $i = 1$  to  $M$  do
       $d_i \leftarrow$  EdgeClosestDistance( $v, e_i$ )
    end for
     $j = \operatorname{argmin}_i d_i$ 
     $\tilde{e}_1 \leftarrow$  PROJECT( $e_{j-1}, e_j$ )
     $\tilde{e}_2 \leftarrow$  PROJECT( $e_{j+1}, e_j$ )
     $\tilde{v} \leftarrow$  PROJECT( $v, e_j$ )
    return  $(\tilde{v} - \tilde{e}_1) \times (\tilde{v} - \tilde{e}_2) \cdot e_j < 0$ 
  end if
end function

function CLOSESTPOINT( $v, \{e_i\}$ )
  for  $i = 1$  to  $M$  do
    [ $p_i$ , type $_i$ ]  $\leftarrow$  FACECLOSESTPOINT( $v, e_{i-1}, e_i$ )
  end for
   $j \leftarrow \operatorname{argmin}_i \|p_i - v\|$ 
  return [ $p_j$ , type $_j$ ,  $j$ ]
end function

function EDGECLOSESTDISTANCE( $v, e$ )
   $w \leftarrow \max\{(v \cdot e) / \|e\|^2, 0\}$ 
  return  $\|v - we\|$ 
end function

function PROJECT( $a, b$ )
   $a' \leftarrow a - (a \cdot b / \|b\|^2)b$ 
  return  $a' / \|a'\|$ 
end function

```

---

We note that if one uses the algorithm above to filter the contact pairs directly, the resulting contact potential is not continuous: In Figure 13, if the vector  $v$  lies on the intersection of the two sub-domains, under small perturbations, the contact pair may be dropped or kept in the filtering. Thus, the potential requires complicated mollification to guarantee  $C^1$  smoothness with this algorithm. Instead, we use a simplified smooth formulation as in Equation (17), which is not exact but conservative.

## C FORMULAS FOR SMOOTHING FUNCTIONS

$$B^3(v) = \begin{cases} \frac{2}{3} - v^2 + \frac{1}{2}|v|^3 & |v| < 1 \\ \frac{1}{6}(2 - |v|)^3 & 1 \leq |v| < 2 \\ 0 & 2 \leq |v|. \end{cases}$$

$$H(z) = \begin{cases} 0 & z < -3 \\ \frac{1}{6}(3+z)^3 & -3 \leq z < -2 \\ \frac{1}{6}(3-9z-9z^2-2z^3) & -2 \leq z < -1 \\ 1 + \frac{1}{6}z^3 & -1 \leq z < 0 \\ 1 & 0 \leq z. \end{cases}$$

## D DETAILS OF THE P.W. SMOOTH FORMULATION

### D.1 Details of the local minimum constraint

If  $y$  is on more than one patch, i.e. on the shared boundary of patches, on each patch  $\Omega_i$ , there are two tangent vectors  $t_i^1(y)$  and  $t_i^2(y)$ , along two edge curves of  $\Omega_i$  meeting at  $y$  (Figure 12). If  $t_i^j(y)^\top (f(y) - f(x))_+ \geq 0$ , then for any  $t = \sum_{j=1}^2 a_j t_i^j(y)$  ( $a_j \geq 0$ ), we have

$$t^\top (f(y) - f(x))_+ = \sum_{j=1}^2 a_j t_i^j(y)^\top (f(y) - f(x))_+ \geq 0,$$

which means Equation (8) holds for any  $t$  in the convex cone spanned by  $t_i^1(y)$  and  $t_i^2(y)$  (Figure 12 A). However, the tangent directions of  $\Omega_i$  at  $y$  may not always lie in the cone. For example, when  $y$  is on an edge of a linear triangle as  $\Omega_i$ ,  $t_i^1(y) + t_i^2(y) = 0$ , then the convex cone degenerates to a line, while the tangent directions form a half-plane (Figure 12 B); when  $y$  is at a concave corner of a high order patch  $\Omega_i$ , then the convex cone is the complement of  $\Omega_i$  (Figure 12 C).

To treat all cases uniformly, we add a halfway vector  $t_i^3$ , corresponding to the angle bisector of the angle between  $t_i^1$  and  $t_i^2$  in  $\Omega_i$  (Figure 12). Then every tangent vector  $t$  of  $\Omega_i$  at  $y$  can be represented as a convex combination of  $\{t_i^j\}_{j=1}^3$ , and consequently, Equation (8) along these three directions ensures that the point is a local minimum of the distance.

### D.2 Local minimum term

The mollification of local minimum constraints  $\Phi_{ik}^m(x, y) \geq -\alpha$ ,  $\forall i \in I$  is straightforward and similar to the smooth surface case:

$$g^m(x, y) := \prod_{i \in I} \prod_{k=1}^3 H^\alpha(\Phi_{ik}^m(x, y))$$

reaches maximum of 1 when  $\Phi_{ik}^m(x, y) \geq 0$ ,  $\forall i \in I$ , and vanishes if there exists  $i \in I$  such that  $\Phi_{ik}^m(x, y) \leq -\alpha$ .

Since  $\Phi_{ik}^m$  is a dot product between unit vectors,  $\alpha$  acts as a threshold of contact potential on the cosine value of the angle between contact pairs. One can pick  $\alpha$  based on the desired angle threshold, we use  $0.05 \leq \alpha \leq 0.8$  in the examples. Although extremely small  $\alpha$  leads to numerical issues in the simulation, Requirement 2 is satisfied as long as  $\alpha > 0$ , i.e., the simulation is always contact-free.

### D.3 Exterior direction term

For edge points in Edge-Edge and Edge-Vertex contact, based on Equation (9), we can define

$$g^e(x, y) := H^\beta(\Phi^e(x, y)),$$

where  $\beta$  is a parameter that controls the smoothness of exterior direction constraints.  $g^e(x, y)$  reaches maximum when  $\Phi^e(x, y) \geq 0$  and vanishes if  $\Phi^e(x, y) \leq -\beta$ .

For vertices in Face-Vertex, Edge-Vertex, and Vertex-Vertex contact, we first define

$$\Phi_i^e(x, y) := -n_i(y)^\top (f(y) - f(x))_+.$$

However, unlike the case of the smooth surfaces, there are multiple normal directions at one vertex (Figure 15). To eliminate the contact between opposite sides of the object (Figure 10), the following should hold for some  $i \in I$

$$\Phi_i^e(x, y) \geq 0. \quad (16)$$

To make the constraint smooth, we relax this to a simpler, weaker condition. We define

$$g^e(x, y) := H^1\left(-1 + \sum_{i \in I} H^\beta(\Phi_i^e(x, y))\right), \quad (17)$$

which reaches a maximum of 1 when there exists  $i$  such that  $\Phi_i^e(x, y) \geq 0$ , and vanishes if for all  $i$ ,  $\Phi_i^e(x, y) \leq -\beta$ .

## E EXCLUSION OF ADJACENT ELEMENTS

Adjacent elements such that one does not contain another can be vertex-adjacent face and edge, and vertex- and face-adjacent faces. Suppose an edge and a face are in contact, i.e., an edge is contained in the plane of the face, and they share a vertex and other points. Then the other edge endpoint is either inside the face, or the edge intersects the boundary of the face, crossing an edge which it does not share a vertex with. In the first case, we have vertex-face contact, in the second case, we have non-adjacent edge-edge contact. Thus, even if we do not apply the potential to the adjacent edge-face pair, when they are close to contact, it will be activated for a non-adjacent element pair. If it is a face-face contact, the argument reduces to the first, if applied to one of the face edges that is not a common edge with the other face.