

Stabilizing Impossible Collisions with Loki

Supplemental Document

Xiao Zhai
Wētā FX
New Zealand
xzhai@wetafx.co.nz

Eston Schweickart
Wētā FX
United States
eschweickart@wetafx.co.nz

Jefri Haryono
Wētā FX
New Zealand
jharyono@wetafx.co.nz

Nikolay Ilinov
Wētā FX
Australia
nilinov@wetafx.co.nz

Andrea Merlo
Wētā FX
New Zealand
merlo@wetafx.co.nz

1 Compliant Kinematics

Kinematic mesh interpenetrations commonly occur during animation sequences. Although meshes might not intersect on integer frames, interpenetrations can arise due to interpolation between keyframes and pose significant challenges in terms of resolution time and feasibility within tight production schedules.

To address this challenge in Loki [Lesser et al. 2022], we introduce a semi-automatic resolution strategy for problematic animations. Our approach empowers artists to designate intersecting-prone regions as compliant. Underlying our approach is the creation of compliant kinematic surfaces that adhere closely to their kinematic guidance; however, upon contact with other surfaces, these compliant vertices recede to form gaps. Conceptually, this method resembles constraining a piece of cloth to a guiding kinematic mesh and allowing the collision solver to dynamically generate necessary separations dynamically. However, our implementation streamlines user interaction by eliminating explicit constraint configuration and optimizing solver performance for computational efficiency.

For elastodynamic simulation, we utilize backward Euler time stepping with optimization integration. The update from timestep n to $n + 1$, excluding external impulses, is governed by

$$E(\mathbf{v}, \mathbf{x}) = \int \frac{\rho}{2} \|\mathbf{v} - \mathbf{v}^n\|^2 dV + \Psi(\mathbf{v}, \mathbf{x}),$$

$$\mathbf{x} = \mathbf{x}^n + h\mathbf{v},$$

$$\mathbf{v}^{n+1} = \arg \min_{\mathbf{v}} E(\mathbf{v}, \mathbf{x}), \quad s.t. \quad C(\mathbf{v}, \mathbf{x}) = 0,$$

where ρ represents material density, V is volume, \mathbf{v} and \mathbf{x} denote the vertex velocities and positions, and h is the timestep. Ψ encapsulates mechanical energy potentials (including conservative forces like gravity and hyperelastic materials, and user-defined constraints), and $C(\mathbf{v}, \mathbf{x})$ enforces contact projection constraints.

Instead of explicitly enforcing kinematic guidance with a force or constraint for compliant kinematic vertices, we opt to omit Ψ

and modify the momentum term directly. Our goal is to encourage these vertices towards the kinematic guidance in a quasistatic manner. We achieve this by setting $\mathbf{v}^n = \mathbf{0}$, effectively removing their momentum, and directing their velocities toward “goals” defined as $\mathbf{v} = (\mathbf{x}^g - \mathbf{x})/h$, where \mathbf{x}^g is the vertex guidance position. Putting it together, for compliant kinematic vertices, Equation 1 simplifies to

$$E_{ck}(\mathbf{v}, \mathbf{x}) = \int \frac{\rho}{2} \left\| \frac{1}{h} (\mathbf{x}^g - \mathbf{x}) \right\|^2 dV$$

$$= \int \frac{\rho}{2h^2} \|\mathbf{x}^g - \mathbf{x}\|^2 dV,$$

$$\mathbf{x} = \mathbf{x}^n + h\mathbf{v}.$$

Discretizing this into vertices yields

$$E_{ck}(\mathbf{v}, \mathbf{x}) = \frac{\rho}{2h^2} \sum_i \|\mathbf{x}_i^g - \mathbf{x}_i\|^2 V_i,$$

$$\mathbf{x}_i = \mathbf{x}_i^n + h\mathbf{v}_i,$$

where i is the vertex index, and $\frac{\rho}{h^2} V_i$ is conveniently the stiffness of an imaginary spring pulling each vertex towards its goal position \mathbf{x}_i^g .

The final system incorporating elasticity, contact, and compliant kinematics becomes

$$[\mathbf{v}, \hat{\mathbf{v}}]^{n+1} = \arg \min_{[\mathbf{v}, \hat{\mathbf{v}}]} E(\mathbf{v}, \mathbf{x}) + E_{ck}(\hat{\mathbf{v}}),$$

$$s.t. \quad C([\mathbf{v}, \hat{\mathbf{v}}], [\mathbf{x}, \hat{\mathbf{x}}]) = 0,$$

where $\hat{\mathbf{v}}$ and $\hat{\mathbf{x}}$ denote the velocities and positions on the new vertices introduced into the system by the compliant kinematic objects, and $[\cdot]$ is vector concatenation.

From the mathematical derivation, our method is functionally equivalent to treating compliant kinematic vertices as dynamic vertices and pulling them towards their kinematic guidance using explicit spring constraints. However, our approach offers several key advantages. Primarily, it significantly simplifies the user interaction with the solver. When an artist tries to alleviate a pinching region in an animation, they simply paint a non-zero stiffness map onto the mesh; all subsequent setup happens automatically behind the scenes. This stiffness map directly corresponds to the “stiffness density” $\frac{\rho}{h^2}$ in Equation 3, intuitively resembling a paintable spring stiffnesses. Notably, we incorporate the $\frac{1}{h^2}$ factor into the painted values to ensure timestep independence of the effective stiffness; using different numbers of substeps should yield consistent results.

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, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM 1557-7368/2025/2-ART
<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

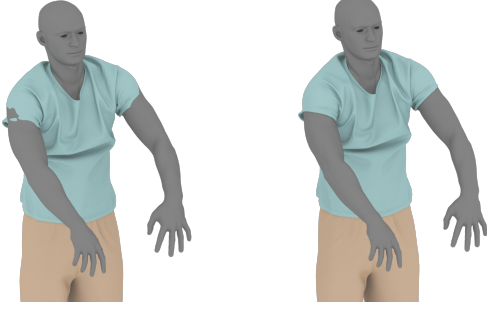


Figure 1: Comparison of strain limiting as a post-collision step (left) and constitutive strain limiting solved together with collision (right). ©Wētā FX.

Furthermore, the modified formula ensures that these newly introduced compliant vertices serve solely for contact resolution and lack momentum, coupling, or other dynamic effects. Finally, within the system matrix layout, these compliant vertices cluster as diagonal sub-blocks disconnected from the rest of the matrix. This allows us to employ a fast preconditioner specifically designed for these diagonal blocks during the solve process, minimizing computational overhead despite introducing additional degrees of freedom.

2 Constitutive Strain Limiting

Strain limiting, often modeled as an edge shrinking step occurring post-collision in FEM simulations, can invalidate collision resolution and lead to penetrations. This issue arises because the collision pass is unaware of the edge shrinking process, which itself does not account for collision constraints – a common issue in projective solvers like Position-Based Dynamics. To address this problem, one could iterate between these passes until convergence or unify the constraints into a single solvable system.

Inspired by the Constitutive Strain Limiting from Codimensional Incremental Potential Contact [Li et al. 2021], Loki adopts the latter approach, combining collisions and strain limits within a unified representation.

Examining more closely on the collision constraint $C(\mathbf{v}, \mathbf{x})$ from Equation 1, contacts typically involves evaluating the gap function ϕ and its time derivative $\dot{\phi}$. The non-penetrating condition can be formulated as a non-negative gap at the end of the timestep with first-order Taylor approximation, $\phi \approx \phi^n + h\dot{\phi} \geq 0$. Given that velocity is linearized within each timestep, we can define an intermediate vector \mathbf{u} at the contact points as

$$\begin{aligned} \mathbf{u} &= \frac{\phi^n}{h} \mathbf{n} + \dot{\phi}, \\ &= \frac{\phi^n}{h} \mathbf{n} + \mathbf{J}\mathbf{v}, \end{aligned} \quad (5)$$

where \mathbf{n} is the collision normal, and \mathbf{J} is the mapping from vertex velocities to contact velocities, often referred to as constraint Jacobian. The final collision constraint can then be formulated as

$$\mathbf{u}_N \geq 0, \quad (6)$$

for simple friction-less contacts; or

$$\begin{cases} 0 \leq \mathbf{u}_N \perp \mathbf{r}_N \geq 0 \\ \mathbf{r} \in K_\mu \\ \mathbf{r}_T = -\mu \mathbf{r}_N \frac{\mathbf{u}_T}{\|\mathbf{u}_T\|}, \text{ if } \mathbf{u}_T \neq \mathbf{0}, \end{cases} \quad (7)$$

for contacts with Coulomb friction, where N and T denote the vector components parallel and tangential to the collision normal \mathbf{n} , μ is the friction coefficient, K_μ denotes the friction cone of aperture μ , \mathbf{r} is the contact force at collision points, and \perp represents the complementarity condition. See [Daviet 2020] for more details about the frictional contacts.

A similar formulation is applicable for assessing elongation of mesh edges. To determine the equivalence of vector \mathbf{u} along an edge, we use

$$\mathbf{u}_{\text{stretch}} = \mathbf{J}_e \mathbf{v} + \frac{l_e^n}{h} \mathbf{e}, \quad (8)$$

where \mathbf{J}_e is the differential stencil of the edge endpoints, l_e^n represents the edge length at the beginning of the timestep, and \mathbf{e} denotes the edge direction. To enforce a maximum edge elongation ε_e , the constraint becomes

$$h\mathbf{u}_{\text{stretch}} \cdot \mathbf{e} = h\mathbf{J}_e \mathbf{v} \cdot \mathbf{e} + l_e^n \leq (1 + \varepsilon_e) l_e^0, \quad (9)$$

where l_e^0 is the rest length of the edge. This constraint can seamlessly be integrated into a projective solver alongside contacts for coherent solutions, or adapted to fit complementarity problem solvers as

$$h\mathbf{u}_{\text{stretch}} \cdot \mathbf{e} \leq (1 + \varepsilon_e) l_e^0 \perp \mathbf{r}_e \geq 0. \quad (10)$$

References

- Gilles Daviet. 2020. Simple and scalable frictional contacts for thin nodal objects. *ACM Trans. Graph.* 39, 4, Article 61 (Aug. 2020), 16 pages. doi:10.1145/3386569.3392439
- Steve Lesser, Alexey Stomakhin, Gilles Daviet, Joel Wretborn, John Edholm, Noh-Hoon Lee, Eston Schweickart, Xiao Zhai, Sean Flynn, and Andrew Moffat. 2022. Loki: A Unified Multiphysics Simulation Framework for Production. *ACM Trans. Graph.* 41, 4, Article 50 (jul 2022), 20 pages. doi:10.1145/3528223.3530058
- Minchen Li, Danny M. Kaufman, and Chenfanfu Jiang. 2021. Codimensional incremental potential contact. *ACM Trans. Graph.* 40, 4, Article 170 (July 2021), 24 pages. doi:10.1145/3450626.3459767