Stabler Neo-Hookean Simulation:
Absolute Eigenvalue Filtering for Projected Newton

HONGLIN CHEN, Columbia University, USA

HSUEH-TI DEREK LIU, Roblox Research & University of British Columbia, Canada

DAVID I.W. LEVIN, University of Toronto & NVIDIA, Canada

CHANGXI ZHENG, Columbia University, USA

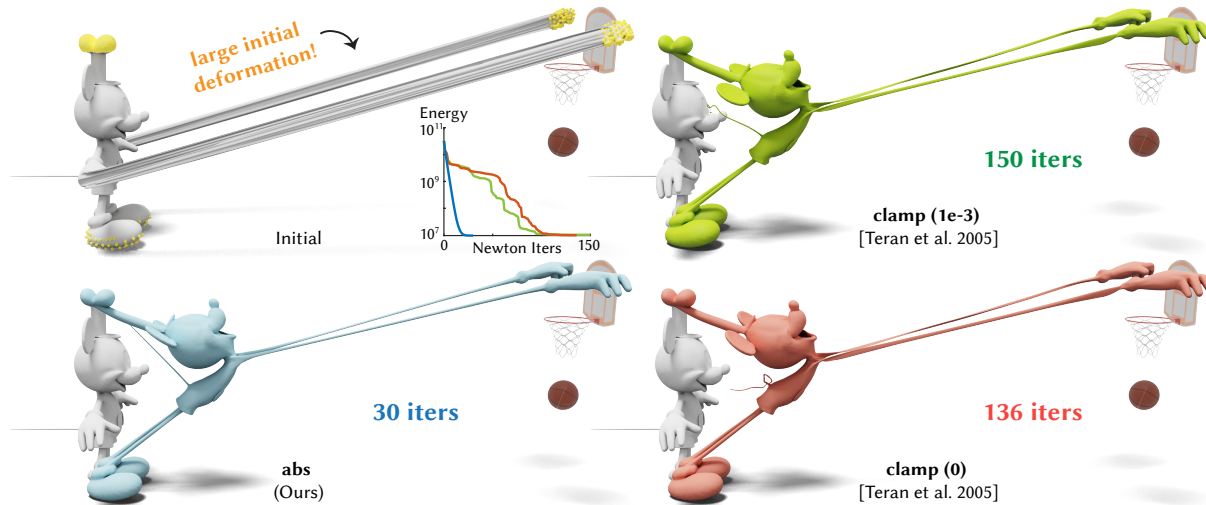ALEC JACOBSON, University of Toronto & Adobe Research, Canada

Fig. 1. Our absolute eigenvalue projection scheme stabilizes the projected Newton optimization of stable Neo-Hookean energy under *high Poisson's ratio* and *large initial volume change*, and achieves a faster convergence rate than the traditional eigenvalue clamping scheme [Teran et al. 2005]. Here we set the Poisson's ratio ν to be 0.495 and the fixed vertices are colored in yellow.

Volume-preserving hyperelastic materials are widely used to model near-incompressible materials such as rubber and soft tissues. However, the numerical simulation of volume-preserving hyperelastic materials is notoriously challenging within this regime due to the non-convexity of the energy function. In this work, we identify the pitfalls of the popular eigenvalue clamping strategy for projecting Hessian matrices to positive semi-definiteness during Newton's method. We introduce a novel eigenvalue filtering strategy for projected Newton's method to stabilize the optimization of Neo-Hookean energy and other volume-preserving variants under high Poisson's ratio (near 0.5) and large initial volume change. Our method only requires *a single line* of code change in the existing projected Newton framework, while achieving significant improvement in both stability and convergence speed. We demonstrate the effectiveness and efficiency of our eigenvalue projection scheme on a variety of challenging examples and over different deformations on a large dataset.

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

Additional Key Words and Phrases: Neo-Hookean elasticity, Projected Newton, eigenvalue filtering.

1 INTRODUCTION

Volume-preserving hyperelastic energies play a crucial role in accurately capturing the near-incompressible nature of soft tissues and rubber-like materials. Among the common hyperelastic material models, Neo-Hookean model and its variants are usually the de facto choice for modeling such materials with high Poisson's ratios $\nu \in [0.45, 0.5)$ (see Table 1). Unfortunately, the numerical optimization of Neo-Hookean material is notoriously challenging within this near-incompressible regime due to the non-convexity from the volume-preserving term, especially in the presence of large volume change. We will show that existing projection approaches to ensure positive-definite of Hessian matrices during Newton's method
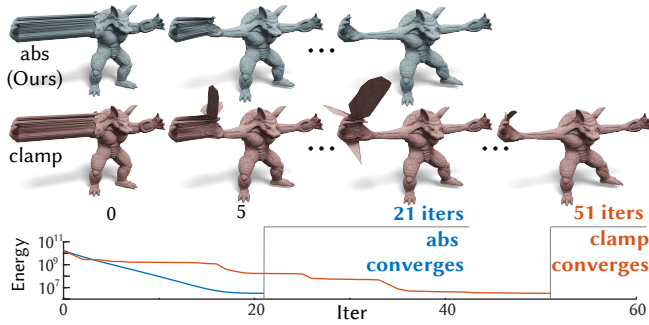
Fig. 2. Our absolute eigenvalue projection strategy is robust to high Poisson's ratio and large volume change.
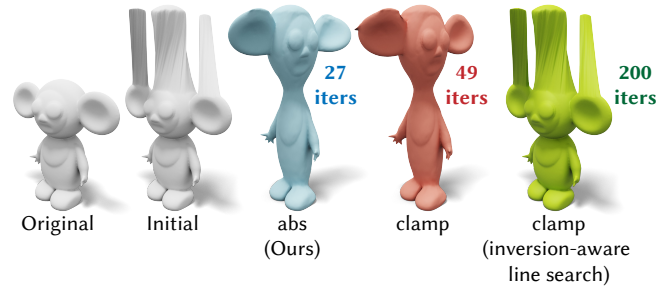

Fig. 3. Inverted elements are common in the presence of large initial deformation, and thus adding an inversion-aware line search may make the optimization stall completely (right). Here we stretch and twist the topmost vertices of the doll by 90 degrees, resulting in 144 inverted elements in the initial deformation.

effectively assume the initial shape to have small volume change, restraining the user from freely editing the initial deformation.

We seek to stabilize and accelerate the optimization of Neo-Hookean energies with high Poisson's ratios under the projected Newton framework in a *robust* and *efficient* way. For *robustness*, we target the scenarios of large initial deformation and volume change, allowing the user to freely deform the initial shape without worrying about optimization blowing up. For *efficiency*, we aim to improve the convergence speed of projected Newton method but still

Table 1. Common materials with high Poisson's ratio (PR).

Material	PR
Rubber	0.4999
Tongue	0.49
Soft palate	0.49
Fat	0.49
Skin	0.48
Muscle	0.47
Saturated clay	0.4-0.49

keep the per-step computation cost unchanged: filtering on the eigenvalues of per-element Hessian contributions à la Teran et al. [2005]. Surprisingly, this leads to an extremely *simple* and *elegant* solution, requiring only one line of code change in the existing projected Newton framework without any additional parameter **(TL;DR):**

```
1  Λ,U=eig(Hi)                      1  Λ,U=eig(Hi)
2  Hi_proj=U*max(Λ,0)*U'    →      2  Hi_proj=U*abs(Λ)*U'
```

While our proposed change to code is small, the technical analysis behind it is non-trivial and interesting:

- Our simple solution is supported by a thorough analysis. We show that the high non-convexity of the Neo-Hookean energy stems from high Poisson's ratio and large volume change (Sec. 6). We analyze the behavior of projected Newton method under such scenarios and identify potential pitfalls of the traditional eigenvalue-clamping projection strategy [Teran et al. 2005] (Sec. 4).
- In response, we propose a novel eigenvalue projection strategy for Newton's method to stabilize the optimization of Neo-Hookean-type energy under high Poisson's ratio and large initial volume change (Sec. 5).

Our paper is a nice complement to the "Stable Neo Hookean Flesh Simulation" paper [Smith et al. 2018] which our title alludes to, improving the numerical stability by dissecting its nonconvexity in the projected Newton optimization. We illustrate the effectiveness and efficiency of our eigenvalue projection scheme on a wide range

of challenging examples, including different deformations, geometries, mesh resolutions, elastic energies, and physical parameters. Through extensive experiments, we show that in the case of large volume change, our method outperforms the state-of-the-art eigenvalue projection strategy and other alternatives in terms of stability and convergence speed, while still, on average, being comparable for simulations with moderate volume changes. Our method is robust across different mesh resolutions and to extreme volume change and inversion. It achieves a stable and fast convergence rate in the presence of a high Poisson's ratio and large volume change.

2 RELATED WORK

While researchers have studied first-order methods for simulating volumetric objects, our approach focuses on improving the robustness and efficiency of the second-order methods, to which we limit our discussion.

Volume-preserving hyperelastic energies play a crucial role in capturing the volumetric behavior of deformable objects. Many hyperelastic energies are modeled as functions of the deformation gradient $\mathbf{F}$. As hyperelastic materials (e.g., rubber) resist volume changes, these energy functions often contain a volume term, a function of the determinant of the deformation gradient $\det(\mathbf{F})$, to penalize volume changes. In practice, Neo-Hookean energy [Ogden 1997] and other volume-preserving variants (e.g., volume-preserving ARAP [Lin et al. 2022]) have been widely used for modeling materials with high Poisson's ratios. Other alternatives such as St. Venant-Kirchhoff model [Picinbono et al. 2004] and co-rotational model [Müller et al. 2002] approximate or linearize the volume term and thus compromise its volume-preserving property and visual effects, particularly in the cases of large deformation and high Poisson's ratios (see Sec.6.2 of [Kim and Eberle 2022]).

More recently, several works have focused on improving the stability and computational efficiency of volume-preserving hyperelastic energies. To improve robustness to mesh inversion and rest-state stability, Smith et al. [2018] proposed the stable Neo-Hookean energy and further applied their analysis to stabilize other hyperelastic energies such as Fung and Arruda-Boyce elasticity [Smith et al. 2018] and Mooney-Rivlin elasticity [Kim and Eberle 2022]. To improve computational efficiency, a series of works [Lin et al. 2022; Smith et al. 2018, 2019] derive analytical eigendecompositions for
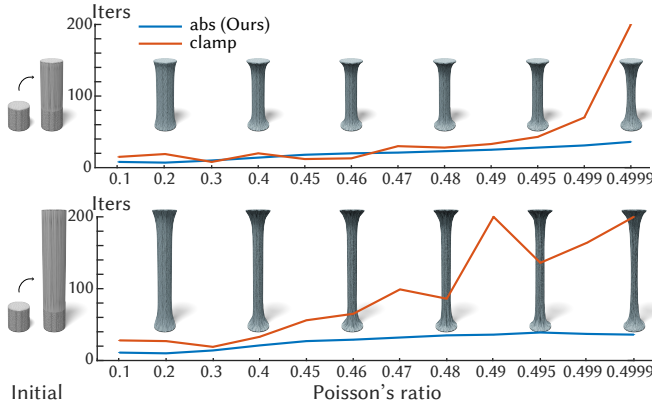
Fig. 4. The speedup we obtained over the eigenvalue clamping strategy increases as the Poisson's ratio and the volume change increase.

isotropic distortion energies to accelerate the positive semi-definite (PSD) projection of per-element Hessian contributions (following [Teran et al. 2005]).

(Element-wise) projected Newton's method. When the Hessian matrix is not positive definite, steps in Newton's method may not always find a direction leading to energy decrease [Nocedal and Wright 2006]. Several Hessian approximation strategies, a.k.a. projected Newton's methods, have been adapted. Directly computing the eigenvalues and eigenvectors of a global Hessian matrix is often too expensive. For many energy functions, their Hessian matrices can be decomposed into a summation over the sub-Hessian of each mesh element. Thereby, one can project each sub-Hessian to the PSD cone by clamping negative eigenvalues to a small positive number (or zero) [Teran et al. 2005] or adding a diagonal matrix [Fu and Liu 2016]. These approaches guarantee that the sub-Hessians are PSD and thus the resulting global Hessian is also PSD. While improving robustness, the projected Newton's method may suffer from worse convergence rate compared to the classic Newton's method [Longva et al. 2023]. What is a better projection strategy remains an open question [Nocedal and Wright 2006].

In this work, we analyze the limitations of existing per-element projected Newton strategies, supported by extensive empirical studies. In the optimization and machine learning literature, Gill et al. [1981] (Sec.4.4.2.1), Paternain et al. [2019] and Dauphin et al. [2014] suggest projecting the negative eigenvalues of the *global* Hessian to its absolute values. We demonstrate that projecting the negative eigenvalues of the *per-element* Hessian to its absolute values, similar to [Paternain et al. 2019] and [Dauphin et al. 2014], performs the best in hyperelastic simulations.

Newton-type methods. Alternatively, there are Newton-type methods that do not rely on per-element PSD projection. One strategy is to add a multiple of identity matrix to the local or global hessian [Nocedal and Wright 2006] but it tends to overdamp the convergence (see [Liu et al. 2017] Fig.8 and [Shtengel et al. 2017] Fig.2). Shtengel et al. [2017] proposed Composite Majorization, a tight convex majorizer as an analytic PSD approximation of the Hessian. While this approximation is efficient to compute, it remains unclear

how to extend it beyond 2D problems and to different types of energies. Instead of directly approximating the Hessian matrix, Lan et al. [2023] proposed a strategy to adjust the searching direction derived from the (possibly non-PSD) Hessian when performing stencil-wise Newton-CG. Chen et al. [2014] applied the asymptotic numerical method to (inverse) static equilibrium problem, and demonstrates its advantages over traditional Newton-type methods. When using incremental potential in dynamic settings, Longva et al. [2023] proposed two alternative strategies: one is to use the original Hessian matrix whenever it is positive definite and use an approximated Hessian matrix otherwise; another is to add a multiple of the mass matrix to the original Hessian until it becomes positive definite. Both of these strategies could require one or several additional Cholesky factorizations each Newton iteration anytime the original Hessian is not PSD. Moreover, this method still inherits the flaws of those fallback Hessians whenever they're invoked. In contrast, our method maintains the same per-iteration computational cost as the *element-wise* projected Newton's method [Teran et al. 2005] while achieving a better convergence rate compared to projected Newton [Teran et al. 2005] and other alternatives [Longva et al. 2023].

3 BACKGROUND

Our approach focuses on quasi-static simulation, which amounts to solving a minimization problem for a (typically nonlinear) energy f with respect to parameters $\mathbf{x}$

$$\min_{\mathbf{x}} f(\mathbf{x}) . \tag{1}$$

Perhaps the most popular way of solving this problem is Newton's method. Given a set of parameters $\mathbf{x}$ at the current iteration, Newton's method approximates the energy f locally with a quadratic function

$$\tilde{f}(\mathbf{x} + \mathbf{d}) \approx f(\mathbf{x}) + \mathbf{g}^T \mathbf{d} + \frac{1}{2} \mathbf{d}^T \mathbf{H} \mathbf{d}, \tag{2}$$

where $\mathbf{g} = \nabla f(\mathbf{x})$ and $\mathbf{H} = \nabla^2 f(\mathbf{x})$ are the gradient and the Hessian of the energy f evaluated at $\mathbf{x}$, and $\mathbf{d}$ denotes the update vector to
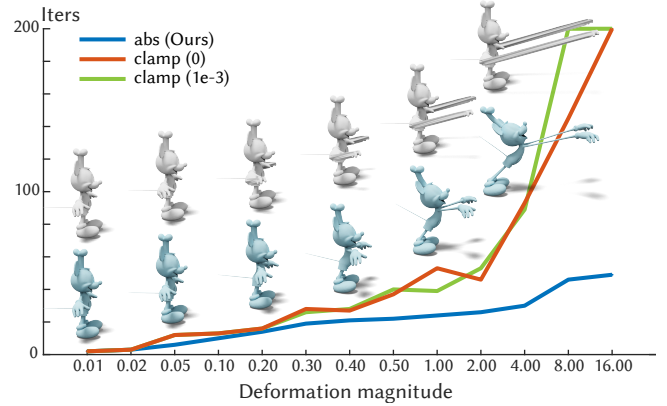

Fig. 5. The speedup we obtained over the eigenvalue clamping strategy increases as the initial volume change increases, while still being comparable for moderate volume change cases. Here we visualize the initial deformation (white) and our final results (blue) under deformation magnitudes 0.01, 0.05, 0.20, 0.40, 1.00 and 2.00 (from left to right).
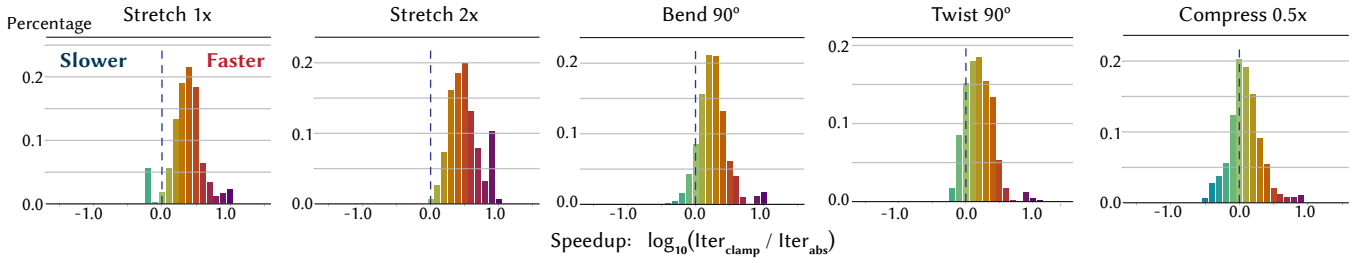
Fig. 6. Histogram of the speedup of our abs strategy over the eigenvalue clamping strategy [Teran et al. 2005] on the TetWild Thingi10k dataset [Hu et al. 2018; Zhou and Jacobson 2016] with diverse deformations, including stretching (1x and 2x), compression, twisting, and bending. Our method offers significant improvements for large stretching, twisting and bending where previous methods may fail, while still, on average, being at least comparable in other regimes (e.g., compression).

Fig. 7. We stress test our abs projection strategy under large stretching (11x) with high Poisson's ratio $\nu = 0.4999$. Here we create the initial deformation by moving the rightmost vertices of a cylinder by a factor of 10x.

the parameter $\mathbf{x}$. To obtain the optimal $\mathbf{d}$, one can set the derivative of $\partial \tilde{f}/\partial \mathbf{d}$ to zero and obtain the update with

$$\mathbf{d} = -\mathbf{H}^{-1}\mathbf{g}. \qquad (3)$$

Because $\tilde{f}$ is merely an approximation of the energy, the resulting $\mathbf{d}$ may not guarantee energy decrease at the end of the iteration. Thus, a line search is needed to find a step size α such that $\mathbf{x} + \alpha\mathbf{d}$ achieves a sufficient decrease in the energy f. Newton's method iterates between these steps until convergence.

However, the process summarized above only works in the ideal situation where the Hessian matrix $\mathbf{H}$ is *positive definite*. Geometrically, a positive definite Hessian $\mathbf{H}$ corresponds to a convex energy space, which ensures that the update direction $\mathbf{d}$ towards the *critical point* $\partial \tilde{f}/\partial \mathbf{d} = 0$ is a decent direction towards the minimum of $\tilde{f}$ (see inset). An indefinite and a negative definite Hessian $\mathbf{H}$, however, correspond to a saddle and a concave shape, respectively (see inset). In both cases, the direction $\mathbf{d}$ towards the critical point could be an energy *ascent* direction, and thus the Newton's method may fail to converge (see inset). To address this issue, a popular approach is the (per-element) *Projected Newton's method*.

3.1 Projected Newton

The idea of projected Newton's method is to approximate a non-positive definite Hessian matrix using a positive definite one. In the case of finite-element simulations, the (global) Hessian matrix $\mathbf{H}_k$
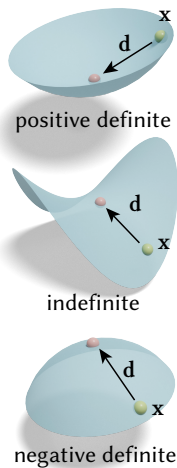
can be assembled from the per-element Hessian matrix $\mathbf{H}_i$ for each mesh (or tetrahedral) element i:

$$\mathbf{H} = \sum_i \mathbf{P}_i^\top \mathbf{H}_i \mathbf{P}_i, \qquad (4)$$

where $\mathbf{P}_i$ is a selection matrix that maps the per-element degrees of freedom to the global degrees of freedom.

The de facto way of making the global Hessian $\mathbf{H}$ positive definite is to project the per-element Hessian $\mathbf{H}_i$ to positive (semi-)definite (PSD) by performing eigen decomposition on the per-element Hessian $\mathbf{H}_i$ numerically or analytically, followed by a clamping strategy to set the negative eigenvalues λ_k to zero (or a small positive number ϵ) [Teran et al. 2005]:

$$\lambda_k^+ = \begin{cases} \epsilon & \text{if } \lambda_k \leq \epsilon, \\ \lambda_k & \text{otherwise.} \end{cases} \qquad (5)$$

Then, one can use the clamped eigenvalues λ_k^+ with the original eigenvectors of $\mathbf{H}_i$ of obtain a PSD projected per-element Hessian $\mathbf{H}_i^+$. Although the projection happens locally, this guarantees the approximated global Hessian

$$\mathbf{H}^+ = \sum_i \mathbf{P}_i^\top \mathbf{H}_i^+ \mathbf{P}_i \qquad (6)$$

assembed from $\mathbf{H}_i^+$ to be PSD [Rockafellar 1970], thus a more robust simulation compared to using $\mathbf{H}$. This per-element Hessian modification strategy is also scalable because it only requires eigen-decomposition on each (small) per-element Hessian $\mathbf{H}_i$, instead of the global matrix.

4 PITFALLS OF EIGENVALUE CLAMPING

Although the projected Newton method has improved robustness over the vanilla Newton's method, the widely used eigenvalue clamping strategy described in Sec. 3.1 still suffers from poor convergence when the simulated object undergoes large volume changes. The cause lies in the operation in

Eq. 5, which clamps the minimum eigenvalue $\lambda_{\min}$ to zero (or a small positive number). This operation effectively turns the local quadratic approximation from a saddle shape (see inset in Sec. 3.1) to a "valley" shape (see inset). In this situation, the Newton update
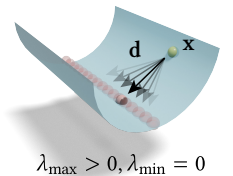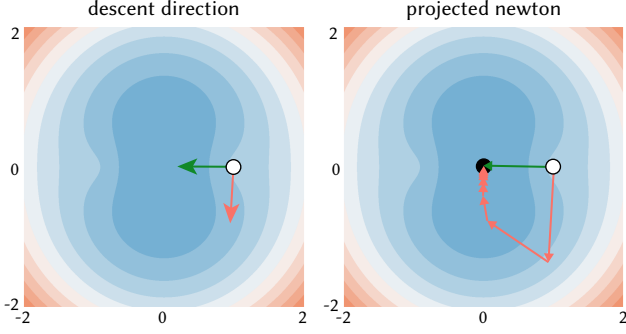
Fig. 8. Consider a function $f(x)$, we visualize the descent direction (left) and the corresponding projected Newton trajectory (right). Our abs-projection (green) takes 3 iterations to converge while ϵ-projection direction (red) [Teran et al. 2005] takes 11 iterations.
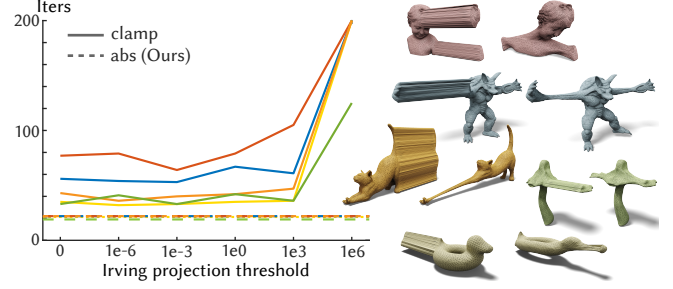

Fig. 9. Our absolute projection strategy is parameter-free and achieves consistent speedup over the eigenvalue clamping strategy in the cases of high Poisson's ratio and large volume change. On the contrary, for the eigenvalue clamping strategy, the optimal ϵ varies across different examples and picking the optimal one requires manual tuning.

direction $\mathbf{d}$ could point toward any point on the bottom of the valley—including points that are far away from the current parameter location $\mathbf{x}$ and even points leading to energy increase.

To solidify this intuition, let us consider a minimal example which exemplifies the extreme failure case of eigenvalue clamping. The function of two variables visualized in in Fig. 8 is defined as

$$f(x,y) = \left(\sqrt{(x+1)^2 + y^2} - 1\right)^2 + \left(\sqrt{(x-1)^2 + y^2} - 1\right)^2. \quad (7)$$

At the point $(x,y) = (1 - 10^{-6}, 10^{-8})$ (indicated by the white point), we have:

$$f = 2, \quad \nabla f = \begin{bmatrix} 3.99 \\ -0.02 \end{bmatrix}, \quad \mathbf{H} = \Phi \begin{bmatrix} -1.99 \times 10^6 & 0 \\ 0 & 3.99 \end{bmatrix} \Phi^\top \quad (8)$$

with eigenvectors $\Phi = \begin{bmatrix} 0.01 & -0.99 \\ 0.99 & 0.01 \end{bmatrix}$.

If we project the negative eigenvalues to a small positive value ϵ (e.g., $\delta = 10^{-3}$), the corresponding Newton direction can be computed as

$$-(\mathbf{H}^+)^{-1}\nabla f = -\left(\Phi \begin{bmatrix} \epsilon & 0 \\ 0 & 3.99 \end{bmatrix} \Phi^\top\right)^{-1} \begin{bmatrix} 3.99 \\ -0.02 \end{bmatrix} = -\begin{bmatrix} 1.19 \\ 19.99 \end{bmatrix}. \quad (9)$$

For small ϵ, the update direction is dominated by y-coordinate. For extreme cases where $\epsilon = 10^{-9}$, the update direction even blows up along the y-coordinate. Although f technically deceases along the search direction, its projection violates the spirit of Newton's method which relies on the quadratic approximation of the energy function in the local region.

The issue of eigenvalue clamping also appears in 3D. This is especially common when simulating large deformations on materials with high Poisson ratios (see Sec. 7).

In our 2D example, we can drive this bad eigenvalue arbitrarily negative by moving the evaluation toward $(x,y) = (1,0)$. A more negative eigenvalue means the function along the corresponding eigendirection is more concave, thus more poorly approximated by a convex quadratic. Unfortunately, clamping to near-zero effectively *prefers* this direction (because after inversion the coefficient explodes, see Eq. 3). We would rather like a filtering method which *avoids* this direction. It is much more reasonable to make the effect of non-convex directions directly proportional to their eigenvalue magnitude. This immediately motivates using the absolute value as a filter.

5 ABSOLUTE VALUE EIGENVALUE PROJECTION

The analysis above suggests that projecting a large negative eigenvalue to a small positive value may lead to a poor estimate of the descent direction. The optimization may get stuck in a local minimum or even diverge in this case. Inspired by [Gill et al. 1981] (Sec.4.4.2.1) and [Paternain et al. 2019], we propose to project the negative eigenvalues of the local Hessian $\mathbf{H}_i$ to its absolute value:

$$\lambda_k^+ = |\lambda_k|. \quad (10)$$

The resulting projected Newton step can also be interpreted as the result of a generalized trust-region method where the model is a first-order Taylor expansion and the trust region is defined using the Hessian metric (see [Dauphin et al. 2014]).

Implementation. For implementations already using per-element projection and scatter-gather assembly of Hessians following Teran et al. [2005], our method is a single line change (see Page 2).

For completeness, the full algorithm per-element absolute eigenvalue projection Hessian construction follows as:

```
1   initialize H_proj to empty sparse matrix
2   foreach element i:
3       # Pi*x selects element i's local variables from x
4       either:
5           Hi = constructLocalHessian(i,Pi*x)
6           Λ,U = eig(Hi)
7       or:
8           Λ,U = analyticDecomposition(i,Pi*x)
9       # Our one-line change
10      Hi_proj = U*abs(Λ)*U'
11      # accumulate into global Hessian
12      H_proj += Pi*Hi_proj*Pi'
```

Applying our projection to the minimal example above, the Newton direction now becomes

$$\mathbf{d} = -(\mathbf{H}^+)^{-1}\mathbf{g} = -\left(\Phi \begin{bmatrix} |-1.99 \times 10^6| & 0 \\ 0 & 3.99 \end{bmatrix} \Phi^\top\right)^{-1} \begin{bmatrix} 3.99 \\ -0.02 \end{bmatrix}$$

$$= -\left(\Phi \begin{bmatrix} 5 \times 10^{-7} & 0 \\ 0 & 0.25 \end{bmatrix} \Phi^\top\right) \begin{bmatrix} 3.99 \\ -0.02 \end{bmatrix} = -\begin{bmatrix} 0.99 \\ -0.01 \end{bmatrix}. \quad (11)$$

which is more aligned with the direction towards the global minimum (see Fig. 8).

Fig. 10. As Poisson's ratio $v \to {}^1/_2$, Lamé's second parameter $\lambda \to \infty$.

6 EIGENANALYSIS OF VOLUME-PRESERVING ENERGY

While the two variable example may seem contrived, we show that the existence of large negative eigenvalues is closely connected to the large Poisson's ratio and large volume change due to the volume-preserving term in Neo-Hookean energy.

When the Hessian contains large negative eigenvalues, the absolute value projection gives a better estimate of the descent direction, making the optimization more stable and faster to converge. Then the question comes, when does the Hessian contain large negative eigenvalues? It turns out that it appears more often than we thought.

The class of Neo-Hookean energy usually takes the form of

$$\Psi = \frac{\mu}{2}(I_C - 3) - \mu \log(J) + \frac{\lambda}{2}(J - 1)^2, \quad (12)$$

where μ and λ are the first and second Lame parameters, $I_C = \mathrm{tr}(\mathbf{F}^\top \mathbf{F})$ is the first right Cauchy-Green invariant and $J = \det(\mathbf{F})$ is the determinant of the deformation gradient $\mathbf{F}$.

To ensure the inversion stability and rest stability, Smith et al. [2018] proposes to use the stable Neo-Hookean energy:

$$\Psi = \frac{\mu}{2}(I_C - 3) + \frac{\lambda}{2}(J - \alpha)^2, \quad (13)$$

where $\alpha = 1 + \frac{\mu}{\lambda}$.

As shown by the eigenanalysis in [Kim and Eberle 2022; Smith et al. 2018], aside from the three eigenvalues corresponding to the scaling, the other six twist and flip eigenvalues of the local Hessian matrix for stable Neo-Hookean energy can be written as

$$\Lambda_3 = \mu + \sigma_z \left(\lambda (J - 1) - \mu\right) \quad (14)$$
$$\Lambda_4 = \mu + \sigma_x \left(\lambda (J - 1) - \mu\right) \quad (15)$$
$$\Lambda_5 = \mu + \sigma_y \left(\lambda (J - 1) - \mu\right) \quad (16)$$
$$\Lambda_6 = \mu - \sigma_z \left(\lambda (J - 1) - \mu\right) \quad (17)$$
$$\Lambda_7 = \mu - \sigma_x \left(\lambda (J - 1) - \mu\right) \quad (18)$$
$$\Lambda_8 = \mu - \sigma_y \left(\lambda (J - 1) - \mu\right). \quad (19)$$

where μ and λ are the Lame parameters, and σ_x, σ_y, and σ_z are the singular values of the deformation gradient $\mathbf{F}$.

When the Poisson's ratio v is close to 0.5, the value of Lamé's second parameter $\lambda = \frac{2v}{1-2v}\mu$ is very large (see Fig. 10). Thus the magnitude of potential negative eigenvalues $\Lambda_{6\sim8}$ largely depends on the Lamé's second parameter λ and the volume change $(J - 1)$. For instance, a Poisson's ratio of 0.495 corresponds to a Lamé's second parameter of $\lambda \approx 100\mu$. This gives the eigenvalues $\Lambda_{6\sim8}$ a large negative value if there is a relatively large volume change for a specific element (i.e., when $(J - 1)$ is large, see Fig. 11).

When the Hessian contains large negative eigenvalues, projecting it to a small positive value may lead to a poor estimate of the descent direction which strongly biases towards the negative eigenvectors (see Sec. 4). In contrast, the absolute value projection gives a better
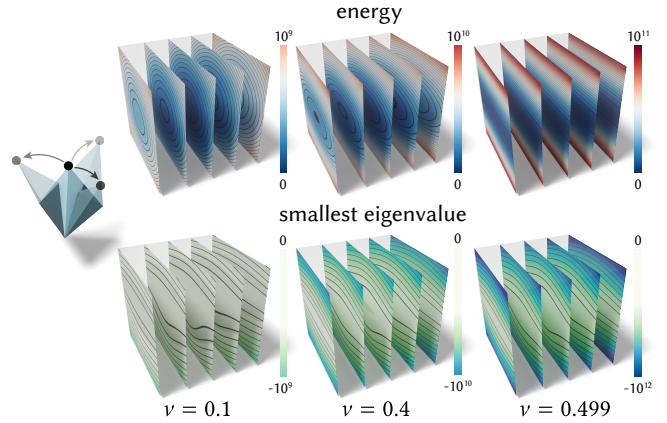

Fig. 11. Energy and smallest eigenvalues of stable Neo-Hookean energy with different Poisson's ratios v and deformation. Here the deformation is created by dragging the top vertex of a regular tetrahedron around, and the center of the plot is with zero displacement.
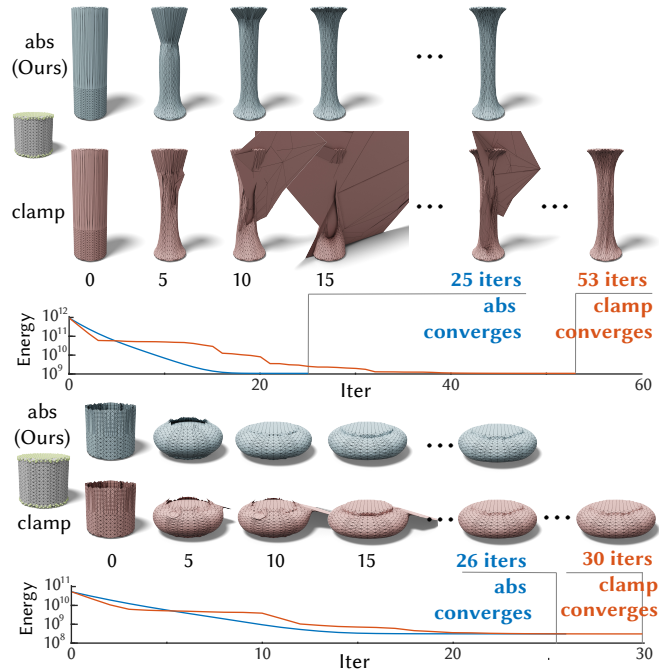

Fig. 12. Our abs projection strategy stabilizes the optimization under large volume change and high Poisson's ratio, and achieves faster convergence rates than the traditional eigenvalue clamping strategy [Teran et al. 2005]. Here we stretch a cylinder to 3.0x (top) and compress a cylinder to 0.5x (bottom) by moving the topmost vertices.

estimate of the descent direction, making the optimization more stable and faster to converge.

7 RESULTS

We evaluate our method by comparing it against the traditional eigenvalue clamping strategy [Teran et al. 2005] and other alternatives on a wide range of challenging examples, including different
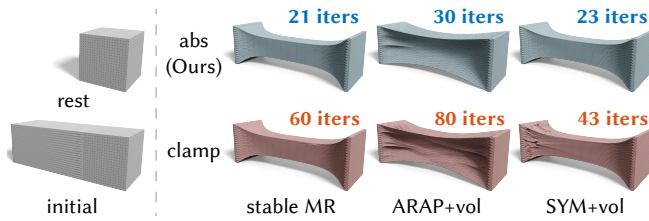
Fig. 13. Our abs projection strategy generalizes over various volume-preserving hyperelastic models. Here we add the volume term $(J - 1)^2$ to different strain energies, including Mooney-Rivlin [Smith et al. 2018], ARAP [Lin et al. 2022] and Symmetric Dirichlet [Smith and Schaefer 2015] energy.
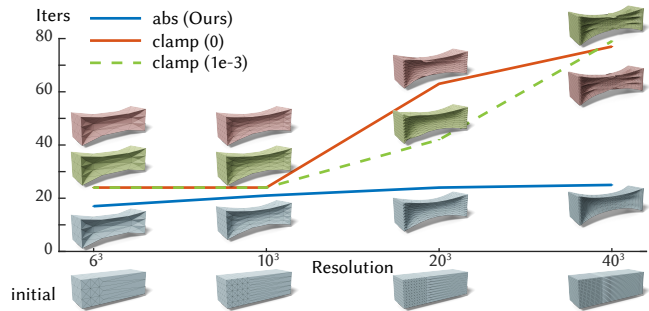

Fig. 14. The speedup of our abs projection strategy over the eigenvalue clamping strategy increases with the mesh resolution. Here we stretch the rightmost vertices of a cube (of 4 different resolutions) by a factor of 3.0x.

deformations, geometries, elastic energies and physical parameters. Furthermore, we experiment on the TetWild Thingi10k dataset [Hu et al. 2018; Zhou and Jacobson 2016] with diverse deformations to test the robustness and scalability of our method.

Unless stated otherwise, we use stable Neo-Hookean model [Smith et al. 2018] with Young's Modulus $E = 10^8$ and Poisson's ratio $v = 0.495$ for all the experiments, and use 0 as the threshold for the eigenvalue clamping strategy, as suggested by Sec.8 of [Teran et al. 2005]. We use a direct sparse Cholesky solver to solve the linear system at each Newton iteration. The convergence threshold is set to be when the newton decrement $0.5\mathbf{d}^\top\mathbf{g}$ is less than 10^{-5} (times λ for the scale of the gradient). The initial deformation is created by moving some selected vertices of the mesh. We run Newton's method for a maximum of 200 iterations with a classical backtracking line search strategy [Nocedal and Wright 2006]. As inversion is extremely common in the presence of large initial deformation (see the inset), we do not use an inversion-aware line search.

We implement our method in C++ with libigl [Jacobson et al. 2018] and use TinyAD [Schmidt et al. 2022] for the automatic differentiation. Experiments are performed using a MacBook Pro with an Apple M2 processor and 24GB of RAM.

Convergence and Stability. We compare our abs projection strategy to [Teran et al. 2005] under a variety of deformations, including stretching, compression, shearing and twisting in Fig. 17, Fig. 12,
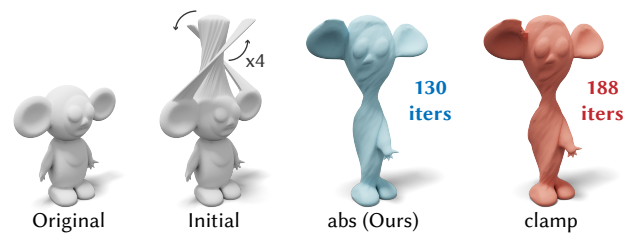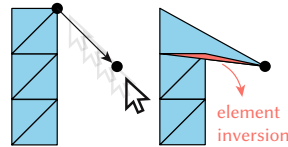

Fig. 15. Our method stabilizes and accelerates the optimization in the presence of large rotations. Here we stretch and twist the topmost vertices of the doll by 360 degrees (divided into four 90° subsolves to avoid ambiguity).
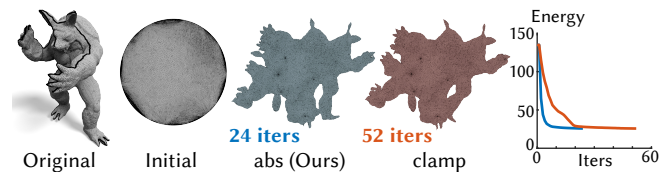

Fig. 16. Our method can also accelerate the optimization of volume-preserving parameterization, where we minimize the energy $0.5E_{\mathrm{MIPS}} + 0.5(J - 1)^2$. Here we start from the Tutte embedding with the same total area as the original surface mesh (the cut is visualized in black), but nevertheless the per-element area distortion could still be large.

Fig. 2, Fig. 3 and Fig. 15. Our abs projection strategy converges swiftly and smoothly under large local volume change and high Poisson's ratio, while [Teran et al. 2005] suffers from a much slower convergence rate due to the instability in the optimization. The energies and shapes for both methods are generally consistent within each example when converged, except in a few cases where the other method converges to a different local minimum after "blowing up" mid-optimization (e.g., Fig. 1 (clamp) and Fig. 19 (global abs)). We also compare our local abs approach to the global abs approach [Dauphin et al. 2014; Paternain et al. 2019] in Fig. 19. Compared to our method, the global abs approach is computationally intractable due to a full eigendecomposition of the global Hessian (takes more than 3 hours for one $38k \times 38k$ Hessian in Fig. 18), and leads to a damped convergence and suboptimal configurations (Fig. 19). We further stress test our method under extreme volume change (stretch to 11x) and an even higher Poisson's ratio ($v = 0.4999$) in Fig. 7. Our method still maintains a stable and fast convergence rate, while [Teran et al. 2005] still fails to converge after 200 iterations.

Generalization. We further evaluate the generality of our method over different mesh resolutions, Poisson's ratios, mesh deformations and hyperelastic models. Under the same deformation, the speedup we gain over [Teran et al. 2005] increases with the mesh resolutions (Fig. 14). Our abs projection approach maintains a relatively stable convergence rate across different resolutions while the traditional eigenvalue clamping method requires drastically more iterations as the resolution increases. In Fig. 4, we show the Newton iteration counts of our method and [Teran et al. 2005] under different Poisson's ratios and volume changes. For relatively small volume change (top), the speedup of our method over [Teran et al. 2005] appears after, e.g., the Poisson's ratio is larger than 0.47. But for
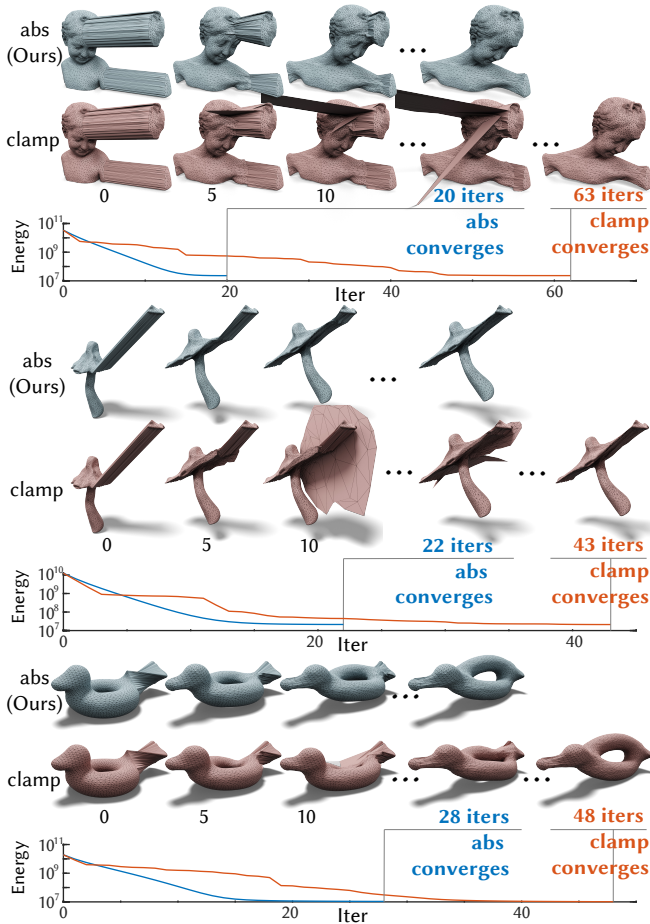
Fig. 17. Our abs projection strategy enables stability and acceleration over diverse deformations with large volume change, including stretching (top), shearing (middle) and twisting (bottom).

volume change that is large enough (bottom), our method can accelerate the convergence rate even when the Poisson's ratio is low, e.g., $\nu = 0.3$. We further demonstrate the growth of our speedup as the initial volume change increases in Fig. 5. Our abs projection strategy can also generalize to other tasks, such as surface parameterization in Fig. 16, stable Neo-Hookean energy with collisions in Fig. 21, and other volume-preserving hyperelastic models in Fig. 13, including Mooney-Rivlin [Smith et al. 2018], ARAP [Lin et al. 2022] and Symmetric Dirichlet energy (each with an additional volume term $(J - 1)^2$ to make it volume-preserving).

Comparison. We evaluate our method and [Teran et al. 2005] (with threshold 10^{-3}) on a total of 593 the closed, genus-0 high-resolution tetrahedral meshes (with more than 5,000 vertices) from the TetWild Thingi10k dataset [Hu et al. 2018; Zhou and Jacobson 2016] with diverse deformations, including stretching, stretching (2x), compression, twisting and bending. As shown by the histogram in Fig. 6, our method is at least comparable with the eigenvalue clamping and offers, on average, 2.5 times speedup for large deformations. In Fig. 9, since the eigenvalue clamping strategy contains an additional user-picked parameter ϵ to control the clamping threshold,
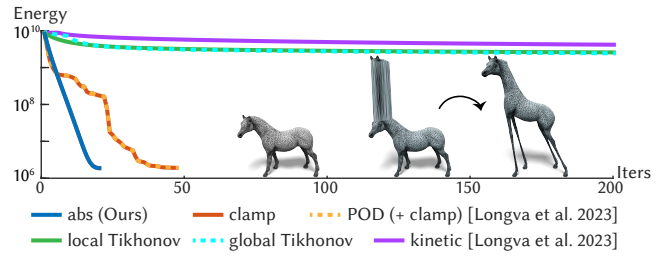
Fig. 18. We compare our abs projection strategy with eigenvalue clamping strategy [Teran et al. 2005], adding a multiple of Identity matrix to the local Hessian or the global Hessian (until it becomes PSD) [Martin et al. 2011], and the Projection-on-Demand and Kinetic strategy [Longva et al. 2023]. Note that the Projection-on-Demand strategy [Longva et al. 2023] requires an additional Cholesky factorization to check the positive-definiteness of the original Hessian at each Newton iteration.
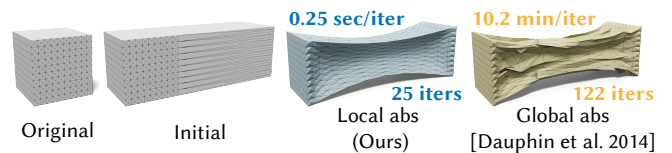

Fig. 19. Compared to our local approach, the global abs approach [Dauphin et al. 2014; Paternain et al. 2019] is computationally intractable due to a full eigendecomposition of the global Hessian (takes more than 3 hours for one $38k \times 38k$ Hessian in Fig. 18), and leads to a damped convergence and suboptimal configurations (right). Here on average, the global abs approach takes 10.2 minutes per Newton iteration for one $8.8k \times 8.8k$ Hessian, while our local abs approach takes only 0.25 seconds for one Newton iteration.
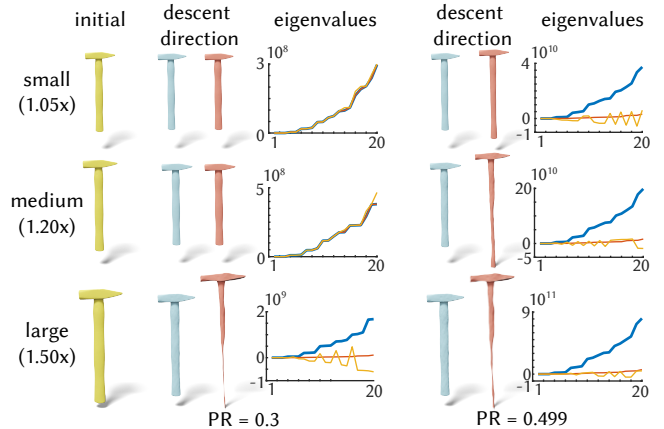

Fig. 20. We visualize the first 20 eigenvalues of the Hessian and the resulting descent direction using our abs projection strategy (red) and the eigenvalue clamping strategy (blue) under different volume changes and Poisson's ratio. Here we uniformly scale the mesh by 1.05x, 1.2x and 1.5x. The eigenvalues of the original Hessian are denoted by yellow.

we compare the convergence rate of our method and [Teran et al. 2005] with different ϵ. Our method is parameter-free and achieves consistent speedup over [Teran et al. 2005]. We further compare our abs projection scheme with other alternatives in Fig. 18, including adding a multiple of Identity matrix to the local Hessian or the
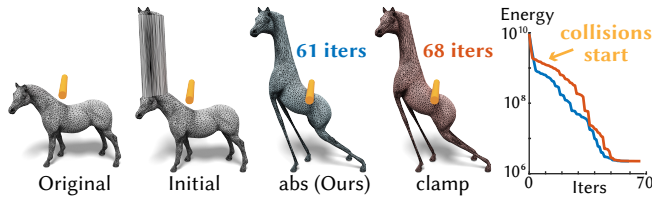
Fig. 21. We perform a collision experiment using Incremental Potential Contact (IPC) [Li et al. 2020], where we put a cylinder (orange) above the back of a horse. Our method is still able to achieve speedup with collisions, even though IPC's intersection-aware line search clamps down the step size and dominates convergence after collisions happen.

global Hessian (until it becomes PSD) [Martin et al. 2011], and the Projection-on-Demand and Kinetic strategy [Longva et al. 2023]. Adding a multiple of Identity or mass matrix to the Hessian eventually makes the optimization closer to (preconditioned) gradient descent, and thus damps the convergence too much for large deformation problems. Note that the Projection-on-Demand strategy [Longva et al. 2023] requires an additional Cholesky factorization to check the positive-definiteness of the original Hessian at each Newton iteration.

8 CONCLUSION & FUTURE WORK

We propose a Hessian modification strategy to improve the robustness and efficiency of hyperelastic simulations.

Our method is a simple one-line change to the existing projected Newton's method, making it extremely reproducible and widely applicable in many simulation frameworks. We primarily evaluate our method on Neo-Hookean simulations with large deformations. For small deformation, especially with compression (see the inset), our method can sometimes slightly damp the

convergence compared to [Teran et al. 2005]. Extending our analysis to a wider range of finite element simulations (e.g., collisions (Fig. 21)) could better identify the applicability of our approach, and potentially deriving an even better adaptive PSD projection method. Exploring other Hessian modification strategies, such as adding higher-order regularization terms to the elastic energy [Kim and Eberle 2022] or a combination with [Longva et al. 2023], could also be an interesting future direction. Considering using another volume-preserving term that does not introduce large negative eigenvalues in the presence of large deformations could be another promising future direction.

ACKNOWLEDGMENTS

REFERENCES

Xiang Chen, Changxi Zheng, Weiwei Xu, and Kun Zhou. 2014. An Asymptotic Numerical Method for Inverse Elastic Shape Design. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2014)* 33, 4 (Aug. 2014).

Yann N. Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. 2014. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2* (Montreal, Canada) *(NIPS'14)*. MIT Press, Cambridge, MA, USA, 2933–2941.

Xiao-Ming Fu and Yang Liu. 2016. Computing inversion-free mappings by simplex assembly. *ACM Trans. Graph.* 35, 6 (2016), 216:1–216:12.

P.E. Gill, W. Murray, and M.H. Wright. 1981. *Practical Optimization*. Academic Press.

Yixin Hu, Qingnan Zhou, Xifeng Gao, Alec Jacobson, Denis Zorin, and Daniele Panozzo. 2018. Tetrahedral Meshing in the Wild. *ACM Trans. Graph.* 37, 4, Article 60 (July 2018), 14 pages. https://doi.org/10.1145/3197517.3201353

Alec Jacobson, Daniele Panozzo, et al. 2018. libigl: A simple C++ geometry processing library. https://libigl.github.io/.

Theodore Kim and David Eberle. 2022. Dynamic Deformables: Implementation and Production Practicalities (Now with Code!). In *ACM SIGGRAPH 2022 Courses* (Vancouver, British Columbia, Canada) *(SIGGRAPH '22)*. Association for Computing Machinery, New York, NY, USA, Article 7, 259 pages. https://doi.org/10.1145/3532720.3535628

Lei Lan, Minchen Li, Chenfanfu Jiang, Huamin Wang, and Yin Yang. 2023. Second-Order Stencil Descent for Interior-Point Hyperelasticity. *ACM Trans. Graph.* 42, 4, Article 108 (jul 2023), 16 pages. https://doi.org/10.1145/3592104

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 Trans. Graph. (SIGGRAPH)* 39, 4, Article 49 (2020).

Huancheng Lin, Floyd M. Chitalu, and Taku Komura. 2022. Isotropic ARAP Energy Using Cauchy-Green Invariants. *ACM Trans. Graph.* 41, 6, Article 275 (nov 2022), 14 pages. https://doi.org/10.1145/3550454.3555507

Tiantian Liu, Sofien Bouaziz, and Ladislav Kavan. 2017. Quasi-Newton Methods for Real-Time Simulation of Hyperelastic Materials. *ACM Transactions on Graphics (TOG)* 36, 3 (2017), 23.

Andreas Longva, Fabian Löschner, José Antonio Fernández-Fernández, Egor Larionov, Uri M. Ascher, and Jan Bender. 2023. Pitfalls of Projection: A study of Newton-type solvers for incremental potentials. arXiv:2311.14526

Sebastian Martin, Bernhard Thomaszewski, Eitan Grinspun, and Markus Gross. 2011. Example-based elastic materials. In *ACM SIGGRAPH 2011 papers* (Vancouver, British Columbia, Canada) *(SIGGRAPH '11)*. Association for Computing Machinery, New York, NY, USA, Article 72, 8 pages. https://doi.org/10.1145/1964921.1964967

Matthias Müller, Julie Dorsey, Leonard McMillan, Robert Jagnow, and Barbara Cutler. 2002. Stable Real-Time Deformations. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (San Antonio, Texas) *(SCA '02)*. Association for Computing Machinery, New York, NY, USA, 49–54. https://doi.org/10.1145/545261.545269

Jorge Nocedal and Stephen J. Wright. 2006. Numerical Optimization. (2006).

Raymond W Ogden. 1997. *Non-linear elastic deformations*. Courier Corporation.

Santiago Paternain, Aryan Mokhtari, and Alejandro Ribeiro. 2019. A Newton-Based Method for Nonconvex Optimization with Fast Evasion of Saddle Points. *SIAM Journal on Optimization* 29, 1 (2019), 343–368. https://doi.org/10.1137/17M1150116

Guillaume Picinbono, Hervé Delingette, and Nicholas Ayache. 2004. Real-Time Large Displacement Elasticity for Surgery Simulation: Non-linear Tensor-Mass Model. *MICCAI* 1935, CH41–CH41. https://doi.org/10.1007/978-3-540-40899-4_66

Ralph Tyrell Rockafellar. 1970. *Convex Analysis*. Princeton University Press, Princeton. https://doi.org/doi:10.1515/9781400873173

Patrick Schmidt, Janis Born, David Bommes, Marcel Campen, and Leif Kobbelt. 2022. TinyAD: Automatic Differentiation in Geometry Processing Made Simple. *Computer Graphics Forum* 41, 5 (2022).

Anna Shtengel, Roi Poranne, Olga Sorkine-Hornung, Shahar Z. Kovalsky, and Yaron Lipman. 2017. Geometric Optimization via Composite Majorization. *ACM Trans. Graph.* 36, 4, Article 38 (jul 2017), 11 pages. https://doi.org/10.1145/3072959.3073618

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

Breannan Smith, Fernando De Goes, and Theodore Kim. 2019. Analytic Eigensystems for Isotropic Distortion Energies. *ACM Trans. Graph.* 38, 1, Article 3 (feb 2019), 15 pages. https://doi.org/10.1145/3241041

Jason Smith and Scott Schaefer. 2015. Bijective parameterization with free boundaries. *ACM Trans. Graph.* 34, 4, Article 70 (jul 2015), 9 pages.

Joseph Teran, Eftychios Sifakis, Geoffrey Irving, and Ronald Fedkiw. 2005. Robust Quasistatic Finite Elements and Flesh Simulation. In *ACM/Eurographics Symposium on Computer Animation (SCA)*, K. Anjyo and P. Faloutsos (Eds.). 181–190.

Qingnan Zhou and Alec Jacobson. 2016. Thingi10K: A Dataset of 10,000 3D-Printing Models. *arXiv preprint arXiv:1605.04797* (2016).