

Anisotropic Elasticity for Inversion-Safety and Element Rehabilitation

THEODORE KIM, Pixar Animation Studios
FERNANDO DE GOES, Pixar Animation Studios
HAYLEY IBEN, Pixar Animation Studios

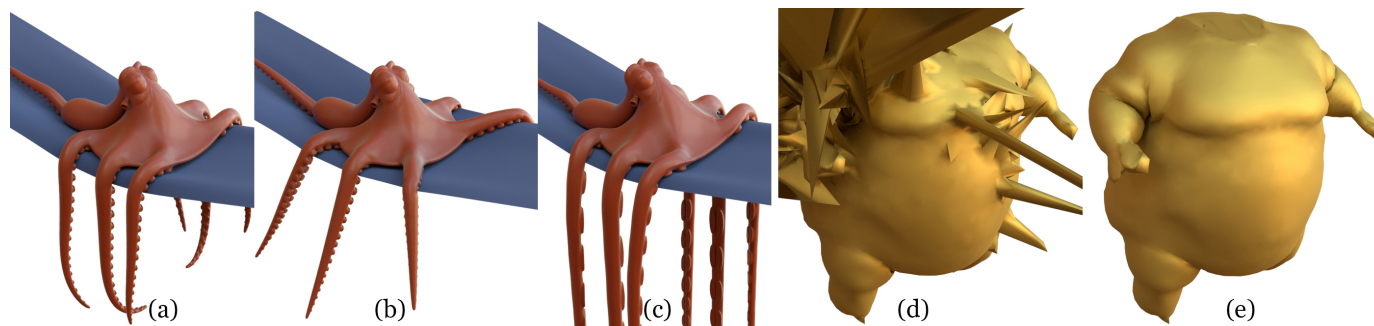


Fig. 1. Our analysis of anisotropic hyperelastic energies yields a novel, robust, and inversion-safe anisotropic energy. Our energy allow fibers along the tentacles (a) to be robustly stiffened by 100 \times (b) and softened by 100 \times (c). Our analysis also produces an anisotropic rehabilitation approach to divergent, badly-conditioned simulations (d) that, when applied, allows them to converge (e).

We present an analysis of anisotropic hyperelasticity, specifically transverse isotropy, that obtains closed-form expressions for the eigendecompositions of many common energies. We then use these to build fast and concise Newton implementations. We leverage our analysis in two separate applications. First, we show that existing anisotropic energies are not inversion-safe, and contain spurious stable states under large deformation. We then propose a new anisotropic strain invariant that enables the formulation of a novel, robust, and inversion-safe energy. The new energy fits completely within our analysis, so closed-form expressions are obtained for its eigensystem as well. Secondly, we use our analysis to *rehabilitate* badly-conditioned finite elements. Using this method, we can robustly simulate large deformations even when a mesh contains degenerate, zero-volume elements. We accomplish this by swapping the badly-behaved isotropic direction with a well-behaved anisotropic term. We validate our approach on a variety of examples.

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

Additional Key Words and Phrases: Anisotropy, element degeneracy, elasticity, physics-based simulation

ACM Reference Format:

Theodore Kim, Fernando De Goes, and Hayley Iben. 2019. Anisotropic Elasticity for Inversion-Safety and Element Rehabilitation. *ACM Trans. Graph.* 38, 4, Article 69 (July 2019), 15 pages. <https://doi.org/10.1145/3306346.3323014>

1 INTRODUCTION

Anisotropic energies are indispensable when simulating realistic phenomena such as muscles [Lee et al. 2018], plants [Wang et al.

Authors' addresses: Theodore Kim, Pixar Animation Studios; Fernando De Goes, Pixar Animation Studios; Hayley Iben, Pixar Animation Studios.

Permission to make digital or hard copies of part or all 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 third-party components of this work must be honored. For all other uses, contact the owner/author(s).

© 2019 Copyright held by the owner/author(s).
0730-0301/2019/7-ART69

<https://doi.org/10.1145/3306346.3323014>

2017], and cloth [Baraff and Witkin 1998], which exhibit directional effects that cannot be captured with isotropic energies alone. Isotropic energies have been extensively analyzed in computer graphics, with efficient methods proposed for both Newton-based [Smith et al. 2018; Stomakhin et al. 2012; Teran et al. 2005] and gradient-based [Bouaziz et al. 2014; Wang and Yang 2016] solvers. Analyses of anisotropic energies have received relatively less attention, with the important exceptions of linear [Cai 2016; Li and Barbič 2015] and spline-based [Xu et al. 2015] orthotropic materials. This knowledge gap is especially visible when incorporating anisotropic energies into Newton-type solvers, because only approximate or brute-force methods are available to project the Hessian back to semi-positive-definiteness.

We address this problem by analyzing the specific anisotropy case of transverse isotropy, i.e. a material that has been strengthened or weakened along one axis. Our analysis shows that many common anisotropic energies from graphics and biomechanics have closed-form eigendecompositions. In fact, we find that there is a family of anisotropic energies that all share the exact same eigenvectors, and the only variation appears in the eigenvalues. These results then enable the construction of fast and concise Newton implementations. We apply this analysis to two applications.

First, we observe that existing transversely isotropic energies are not inversion-safe [Irving et al. 2004], and contain spurious stable states that are especially prevalent under large deformation and high anisotropic stiffness. Under inversion, the anisotropic forces can overwhelm the isotropic forces and drive the simulation towards non-physical states. In order to introduce inversion-safety into these energies, we propose a new anisotropic strain invariant. By carefully avoiding a variety of singularities, we show that a fast, simple, robust, and inversion-safe anisotropic energy can be formulated. The energy is quadratic, so it can be used to introduce both anisotropic stiffening and *softening* to existing isotropic models (Fig. 1(a)-(c)). The energy

is compatible with our analysis, and closed-form expressions are presented for its eigensystem.

For our second application, we use our analysis to show that anisotropic energies can be used to *rehabilitate* badly-conditioned elements (Fig. 1(d)-(e)). Even if a mesh contains zero-volume elements at its rest-pose with infinitely poor conditioning, we are able to robustly simulate large deformations. We accomplish this by replacing the numerically troublesome direction of the (arbitrary) isotropic energy with a well-behaved anisotropic energy. In contrast to accuracy-centered p -refinement [Schneider et al. 2018] or XFEM [Koschier et al. 2017] techniques, our robustness-centered approach tends to decrease the condition number of the final Hessian.

Our main contributions are as follows:

- An analysis of anisotropic energies that shows that their eigensystems can often be written in closed form.
- A new, inversion-aware anisotropic strain invariant.
- A robust, inversion-safe anisotropic model that we call the *Anisotropic ARAP* energy, whose eigenstructure we are able to completely characterize.
- A *rehabilitation* approach that allows badly-conditioned elements to be simulated robustly and efficiently.

2 RELATED WORK

We will be investigating the anisotropy case of transverse isotropy, where a single direction, or “fiber”, is strengthened or weakened, but the other directions remain symmetric. Other alternatives are orthotropic materials [Li and Barbič 2015], strain energies containing explicit deformation constraints [Weiss et al. 1996], or element shape function modifications [Chen et al. 2018]. We focus on the transverse case because it can be factorized from the surrounding isotropic material and is agnostic to the choice of basis functions.

Anisotropic materials have been studied extensively in biomechanics because of their relationship to biological tissues such as skin and muscle [Holzapfel 2005]. Consequently, there are many anisotropic material models available in that literature, such as the Holzapfel-Gasser-Ogden model [Holzapfel et al. 2000], the model of Ciarletta et al. [2011], or Horgan and Saccomandi [2005]. A recent survey of the wide range of available energies is available in Chagnon et al. [2015]. All of these energies are not inversion-safe, which is a limitation that our work will address.

Computer graphics has used these energies to simulate virtual surgery [Picinbono et al. 2000] and virtual humans. For example, the work of Teran et al. [2003, 2005] uses the biomechanical model from Blemker et al. [2005]. Projective Dynamics-based [Bouaziz et al. 2014] approaches have been proposed [Liu et al. 2017] that include applications in musculoskeletal control [Lee et al. 2018]. Irving et al. [2004] proposed inversion-aware anisotropic forces for explicit time integrators, but did not provide a Hessian projection method for the implicit case, or a strain energy for line searches [Gast et al. 2015]. Subsequently, Teran et al. [2005] provided a Hessian projection method, but only for isotropic materials.

Strand-based force-length models are also common in control-based applications [Lee and Terzopoulos 2006; Lee et al. 2014; Sachdeva et al. 2015], but the strands are generally not embedded inside a

volumetric substrate, so the notion of inversion is undefined. Thus, these models are orthogonal to the issues we examine.

We will also investigate the issue of badly-conditioned meshes. Degenerate elements arise in plastic flows [Bargteil et al. 2007; Wojtan and Turk 2008], and the traditional solution is remeshing. The survey of Manteaux et al. [2017] summarizes many of the relevant issues. Simpler approaches are available, such as deleting the bad elements [Forest et al. 2002], but this creates hollowing problems.

Many meshing schemes have been proposed [Hu et al. 2018; Labelle and Shewchuk 2007; Molino et al. 2003], but the problem is still not completely solved, and scenarios remain where even high-quality initial meshes can fail. We examine the particular problem of *rest-state retargeting*, where an initially high-quality mesh is momentarily warped to match an input animation. This is a very common method in production [Gonzalez-Ochoa et al. 2002; Irving et al. 2008; Milne et al. 2016], and used to closely constrain secondary ballistic motion to an existing animation. The warp can ruin the mesh quality, but a full remesh at every frame is too costly. Relaxation methods [Li et al. 2017; Wong et al. 2018] can be used to improve the warped mesh’s quality without modifying its topology, but these approaches do not guarantee a minimum quality, so degenerate elements still appear.

We propose a *rehabilitation* approach that preserves as much of the isotropic material model as possible while avoiding precipitous increases in the condition number. This differs from recent p -refinement approaches [Schneider et al. 2018] which improve accuracy at the cost of higher condition numbers and longer iterative running times, which make these approaches more suited for direct solvers. Degenerate elements are also encountered during fracture simulation. Remeshing alternatives such as virtual node [Molino et al. 2004] and XFEM [Belytschko et al. 2013; Koschier et al. 2017] methods have been developed, but these algorithms assume that an initially well-conditioned element has fractured into badly-conditioned components. We instead examine the case where high-quality elements are ruined via warping. The most relevant work is by Iben [2007], which used a Moore-Penrose pseudo-inverse in lieu of remeshing. That work assumed that the stress field was specified procedurally, but we will use it as a stepping stone to handling general degeneracies.

3 BACKGROUND AND PRELIMINARIES

We denote scalars with unbolded lowercase, vectors with bold lowercase, and matrices with bold uppercase, respectively a , \mathbf{a} and \mathbf{A} . Scalar entries in a vector or matrix are addressed using subscripts, e.g. a_i or A_{ij} . Higher-order tensors use blackboard bold, \mathbb{A} .

We will be manipulating tensors and matrices in the manner of Smith et al. [2019] (§3 in that paper). In the interest of self-containment, we also provide a summary of the necessary tensor notation in our supplemental material.

3.1 Deformation and Tensor Notation

We follow the mechanics approach [Belytschko et al. 2013; Bonet and Wood 2008] of defining a deformation map $\phi(\bar{\mathbf{x}}) = \mathbf{x}$, which deforms a vertex at its initial rest pose, $\bar{\mathbf{x}}$, to its deformed position \mathbf{x} . This is usually defined using the affine map $\phi(\bar{\mathbf{x}}) = \mathbf{F}\bar{\mathbf{x}} + \mathbf{t}$, where

$\mathbf{t} \in \mathbb{R}^3$ encodes any translation, and $\mathbf{F} \in \mathbb{R}^{3 \times 3}$ encodes the map's scaling and rotation (combining them yields shear). The matrix \mathbf{F} is known as the *deformation gradient* because $\frac{\partial \phi(\bar{\mathbf{x}})}{\partial \bar{\mathbf{x}}} = \mathbf{F}$, and is the primary variable that we will be analyzing. We reserve the symbol \mathbf{f} to denote the flattened version of \mathbf{F} , i.e. $\mathbf{f} = \text{vec}(\mathbf{F}) \in \mathbb{R}^9$, where the $\text{vec}(\cdot)$ operator is described in §1.3 of the supplement, as well as Golub and Van Loan [2012]. When forces need to be specified, we instead write \mathbf{g} . The displacement of a vertex is denoted \mathbf{u} , so an alternative expression for the deformed position is $\mathbf{x} = \bar{\mathbf{x}} + \mathbf{u}$.

Deformations are converted to forces using hyperelastic strain energies, which we denote $\Psi(\mathbf{F})$. Designing robust expressions for $\Psi(\mathbf{F})$ is one of our overarching goals. As we are targeting Newton-type solvers, force gradients are also needed. Forces are obtained by computing the first Piola-Kirchhoff stress (PK1), $\mathbf{P}(\mathbf{F}) = \frac{\partial \Psi(\mathbf{F})}{\partial \mathbf{F}}$ and applying the change-of-basis tensor $\frac{\partial \mathbf{F}}{\partial \mathbf{u}}$ via double contraction:

$$\mathbf{g} = \frac{\partial \mathbf{F}^T}{\partial \mathbf{u}} : \mathbf{P}(\mathbf{F}). \quad (1)$$

For tetrahedral elements, $\mathbf{g} \in \mathbb{R}^{12}$ while $\mathbf{P}(\mathbf{F}) \in \mathbb{R}^{3 \times 3}$ and $\frac{\partial \mathbf{F}}{\partial \mathbf{u}} \in \mathbb{R}^{3 \times 3 \times 12}$. Double contraction between 3rd- and 2nd-order tensors is summarized in §1.1 of our supplement.

For Newton-type solvers, the Hessian of the strain energy is also needed, and its eigenvalues must be projected to positive-semi-definiteness. The Hessian can be written in either 4th-order tensor form, $\frac{\partial^2 \Psi(\mathbf{F})}{\partial \mathbf{F}^2} = \mathbb{H} \in \mathbb{R}^{3 \times 3 \times 3 \times 3}$, or in flattened matrix form, $\frac{\partial^2 \Psi(\mathbf{F})}{\partial \mathbf{f}^2} = \mathbf{H} \in \mathbb{R}^{9 \times 9}$. More details are in §1.2 and §1.3 of our supplement. We will usually write Hessians in the flattened form, because it often allows terms to be succinctly written using outer products.

The least conventional notation we will employ is that of *eigenmatrices*. Smith et al. [2019] showed that simple analytical expressions can be found for the eigensystems of isotropic energies, but the simplicity only appears if the systems are written in terms of the eigenmatrices of a 4th-order tensor. Thus, in lieu of the usual eigenpairs $(\lambda_i, \mathbf{q}_i)$, where $\mathbf{q}_i \in \mathbb{R}^9$, we instead write $(\lambda_i, \mathbf{Q}_i)$, where $\mathbf{Q}_i \in \mathbb{R}^{3 \times 3}$. The more familiar eigenvectors are retrieved via $\mathbf{q}_i = \text{vec}(\mathbf{Q}_i)$.

3.2 Isotropic Invariants

Isotropic strain energies have historically been written in terms of the right Cauchy-Green invariants of \mathbf{F} ,

$$I_C = \text{tr}(\mathbf{C}) \quad II_C = \text{tr}(\mathbf{C}^T \mathbf{C}) \quad III_C = \det(\mathbf{C}), \quad (2)$$

where $\mathbf{C} = \mathbf{F}^T \mathbf{F}$. In computer graphics, Irving et al. [2004] observed that squaring \mathbf{F} discards sign information that signals to the strain energy when an element has inverted, and prevents the formation of proper restorative forces. A set of filters were proposed to account for this problem and Teran et al. [2005] showed how to project the resulting Hessians back to semi-positive definiteness.

Recently, Smith et al. [2019] showed that these problems can be avoided by defining a set of *lower-order, sign-preserving* invariants,

$$I_1 = \text{tr}(\mathbf{S}) \quad I_2 = \text{tr}(\mathbf{S}^T \mathbf{S}) \quad I_3 = \det(\mathbf{S}), \quad (3)$$

where \mathbf{S} is the stretch matrix from the polar decomposition $\mathbf{F} = \mathbf{R}\mathbf{S}$. All of the properties of the \mathbf{C} -based invariants are maintained, but the space of expressible energies now expands to include inversion-safe cases such as ARAP [Sorkine and Alexa 2007] and Co-rotational

elasticity [McAdams et al. 2011]. We will investigate a similar treatment for anisotropic invariants.

3.3 Anisotropic Invariants

Following from Eqn. 2, many mechanics works [Blemker et al. 2005; Weiss et al. 1996] also define anisotropic invariants,

$$IV_C = \mathbf{a}^T \mathbf{C} \mathbf{a} \quad V_C = \mathbf{a}^T \mathbf{C}^T \mathbf{C} \mathbf{a}, \quad (4)$$

where \mathbf{a} is an ‘‘anisotropy’’ or ‘‘fiber’’ direction in rest-space that specifies the orientation of the anisotropic force. In the following, we will make the common assumption that \mathbf{a} is normalized. In this formulation, fiber directions are automatically rotated into pose-space as elements deform. While V_C and other higher-order invariants can also appear, biomechanics practitioners [Weiss et al. 1996] have observed that most of their effects can be captured using solely IV_C , so we will focus on IV_C here. As the current invariants are based on \mathbf{C} , they inherit the sign-discarding, inversion-oblivious problems from the isotropic case. We will establish inversion-safe versions in §5, though it will be more involved than the $\mathbf{C} \rightarrow \mathbf{S}$ substitution from the isotropic case.

To maintain consistency with Eqn. 3, we rename and renumber the invariants so that for the remainder of the paper, $I_5 = IV_C$. The numbering has been shifted so that we can later introduce a new, lower-order invariant, I_4 . We will assume that the isotropic and anisotropic components are additively layered, i.e. $\Psi = \Psi_{\text{iso}} + \Psi_{\text{aniso}}$, which cleanly separates the qualitative behavior of each term, and allows each to be analyzed in isolation.

Finally, while fast methods are available to project isotropic Hessians back to semi-positive definiteness [Smith et al. 2019; Teran et al. 2005], equivalent methods are not available for anisotropic energies, so brute-force eigendecompositions or approximate projections are usually employed. Our analysis will instead show that fast, exact projections are possible.

4 AN EIGENANALYSIS OF I_5

4.1 The Eigensystem of I_5

We will now show that the eigensystem of any energy expressed solely in terms of I_5 can be written down in closed form. The I_5 invariant can be written in several forms,

$$I_5 = \|\mathbf{F}\mathbf{a}\|_2^2 = \mathbf{a}^T \mathbf{C} \mathbf{a} = \text{tr}(\mathbf{C}\mathbf{A}), \quad (5)$$

where $\mathbf{A} = \mathbf{a}\mathbf{a}^T$ and $\|\cdot\|_2^2$ denotes the squared Euclidean norm. The PK1 and Hessian in 3D are

$$\frac{\partial I_5}{\partial \mathbf{F}} = 2\mathbf{F}\mathbf{A} \quad (6)$$

$$\frac{\partial^2 I_5}{\partial \mathbf{f}^2} = 2 \begin{bmatrix} \mathbf{A}_{00}\mathbf{I}_{3 \times 3} & \mathbf{A}_{01}\mathbf{I}_{3 \times 3} & \mathbf{A}_{02}\mathbf{I}_{3 \times 3} \\ \mathbf{A}_{10}\mathbf{I}_{3 \times 3} & \mathbf{A}_{11}\mathbf{I}_{3 \times 3} & \mathbf{A}_{12}\mathbf{I}_{3 \times 3} \\ \mathbf{A}_{20}\mathbf{I}_{3 \times 3} & \mathbf{A}_{21}\mathbf{I}_{3 \times 3} & \mathbf{A}_{22}\mathbf{I}_{3 \times 3} \end{bmatrix} = 2\mathbf{H}_5, \quad (7)$$

where $\mathbf{I}_{3 \times 3}$ is a 3×3 identity matrix, and \mathbf{A}_{ij} is the (i, j) scalar entry of \mathbf{A} . (Appendix A shows the matrix explicitly.) Since Eqn. 7 is constant in \mathbf{a} , it is straightforward to state its eigensystem in closed form. In 3D, it contains three identical non-zero eigenvalues, $\lambda_{0,1,2} = 2\|\mathbf{a}\|_2^2$, and since fiber directions are usually normalized, this simplifies

to $\lambda_{0,1,2} = 2$. The eigenvalue is repeated, so the eigenmatrices are arbitrary up to rotation, but one convenient phrasing is:

$$\mathbf{Q}_0 = \begin{bmatrix} & \mathbf{a}^T & \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{Q}_1 = \begin{bmatrix} 0 & 0 & 0 \\ & \mathbf{a}^T & \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{Q}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ & & \mathbf{a}^T \end{bmatrix}.$$

This eigenstructure has a straightforward interpretation. I_5 introduces scaling constraints along the anisotropy direction, so the three eigenvectors encode this rank-three phenomenon. The remaining eigenvalues are all zero, so the Hessian contains a rank-six null space. We have provided supplemental Matlab/Octave code that validate these expressions.

The 2D case follows similarly. The Hessian is

$$\frac{\partial^2 I_5}{\partial \mathbf{f}^2} = 2 \begin{bmatrix} \mathbf{A}_{00} \mathbf{I}_{2 \times 2} & \mathbf{A}_{01} \mathbf{I}_{2 \times 2} \\ \mathbf{A}_{10} \mathbf{I}_{2 \times 2} & \mathbf{A}_{11} \mathbf{I}_{2 \times 2} \end{bmatrix} = 2 \mathbf{H}_5, \quad (8)$$

the eigenvalues are $\lambda_{0,1} = 2 \|\mathbf{a}\|_2^2$ and the eigenmatrices become

$$\mathbf{Q}_0 = \begin{bmatrix} & \mathbf{a}^T \\ 0 & 0 \end{bmatrix} \quad \mathbf{Q}_1 = \begin{bmatrix} 0 & 0 \\ & \mathbf{a}^T \end{bmatrix}.$$

The 2D case correspondingly contains a rank-two null space.

4.2 The Eigensystems of Arbitrary I_5 Energies

Using the eigenanalysis of I_5 , we can now write closed-form expressions for the eigenvalues and eigenvectors of *any* anisotropic energy that is written solely in terms of I_5 . The Hessian of any such arbitrary energy $\Psi(I_5) = \Psi_5$ can be written in flattened $\mathfrak{R}^{9 \times 9}$ form:

$$\frac{\partial^2 \Psi_5}{\partial \mathbf{f}^2} = \frac{\partial \Psi_5}{\partial I_5} \frac{\partial^2 I_5}{\partial \mathbf{f}^2} + \frac{\partial^2 \Psi_5}{\partial I_5^2} \begin{pmatrix} \frac{\partial I_5}{\partial \mathbf{f}} & \frac{\partial I_5}{\partial \mathbf{f}}^T \end{pmatrix}. \quad (9)$$

Inserting Eqns. 6 and 7 into this expression yields:

$$\frac{\partial^2 \Psi_5}{\partial \mathbf{f}^2} = 2 \left[\frac{\partial \Psi_5}{\partial I_5} \mathbf{H}_5 + 2 \frac{\partial^2 \Psi_5}{\partial I_5^2} \left(\mathbf{f}_a \mathbf{f}_a^T \right) \right], \quad (10)$$

where $\mathbf{f}_a = \text{vec}(\mathbf{FA})$. The vector \mathbf{f}_a is in the span of the rank-three subspace encoded by \mathbf{H}_5 , so one of the three eigenpairs becomes unique, while the other two remain arbitrary up to rotation.

The unique eigenvalue can be written in a generic form:

$$\lambda_0 = 2 \left(\frac{\partial \Psi_5}{\partial I_5} + 2 I_5 \frac{\partial^2 \Psi_5}{\partial I_5^2} \right). \quad (11)$$

The I_5 appears in the rightmost term because the corresponding eigenmatrix is:

$$\mathbf{Q}_0 = \frac{1}{\sqrt{I_5}} \mathbf{FA}, \quad (12)$$

and the $1/\sqrt{I_5}$ normalization that appears twice in the $\mathbf{f}_a \mathbf{f}_a^T$ outer product from Eqn. 10 must be accounted for.

The non-unique eigenvalues take the generic form,

$$\lambda_{1,2} = 2 \frac{\partial \Psi_5}{\partial I_5}, \quad (13)$$

and the corresponding eigenmatrices are as follows.

4.2.1 The 2D Eigenmatrix: In 2D, since scaling is only rank-two, λ_1 takes the form of Eqn. 13. Constructing the corresponding eigenmatrix is simpler in 2D, because it is uniquely defined as lying in the span of \mathbf{H}_5 , but orthogonal to \mathbf{Q}_0 . If we define the SVD of \mathbf{F} as $\mathbf{F} = \mathbf{U}\Sigma\mathbf{V}^T$, and the *twist* matrix $\mathbf{T} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ (see our supplement),

then we can write the second eigenmatrix as:

$$\mathbf{Q}_1 = \mathbf{U}\mathbf{T}\Sigma\mathbf{V}^T \mathbf{A}. \quad (14)$$

The second eigenmatrix is similar to \mathbf{Q}_0 , except that \mathbf{T} swaps and negates the coordinates in order to guarantee orthogonality to \mathbf{FA} . For simplicity, we show \mathbf{Q}_1 in non-normalized form.

4.2.2 The 3D Eigenmatrices: In 3D, the two arbitrary eigenmatrices can be constructed using the intuition from the 2D case. There are three possible twist matrices in 3D:

$$\mathbf{T}_x = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \quad \mathbf{T}_y = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad \mathbf{T}_z = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

If we arbitrarily commit to \mathbf{T}_x we can build a second eigenmatrix that closely resembles the 2D case:

$$\mathbf{Q}_1 = \mathbf{U}\mathbf{T}_x \Sigma \mathbf{V}^T \mathbf{A}. \quad (15)$$

The third eigenmatrix then becomes unique in terms of the other two twists:

$$\mathbf{Q}_2 = (\sigma_y \hat{\mathbf{a}}_y) \mathbf{U}\mathbf{T}_z \Sigma \mathbf{V}^T \mathbf{A} - (\sigma_z \hat{\mathbf{a}}_z) \mathbf{U}\mathbf{T}_y \Sigma \mathbf{V}^T \mathbf{A}, \quad (16)$$

where $\sigma_{y,z}$ are the corresponding diagonal entries from Σ and $\hat{\mathbf{a}}_{y,z}$ are entries from $\hat{\mathbf{a}} = \mathbf{V}^T \mathbf{a}$. Alternatively, we could commit to \mathbf{T}_y or \mathbf{T}_z in Eqn. 15. The eigenvectors that would have resulted are listed in Appendix B.

The eigenvectors for I_5 -based energies *always take these forms*, i.e. Eqns. 12 and 14 in 2D, and Eqns. 12, 15, 16 in 3D. The only expressions that change between energies are the eigenvalues.

5 ANISOTROPIC ENERGIES

We will now employ the results of the previous section to analyze existing energies, identify their deficiencies, and construct a new, inversion-safe energy. To begin, we identify two criteria for a good anisotropic energy.

- (1) The energy should not be unnecessarily non-linear, because in addition to enabling stiffening in the fiber direction, it should enable *softening*. If the energy is too non-linear, this will not be possible for quadratic isotropic energies.
- (2) The energy should not contain any spurious stable states under compression and reflection, which means that it should be able to detect inversion and generate appropriate inversion-reversing forces.

Softening has aesthetically intuitive interpretation, e.g. we increase a tentacle's hanging length under gravity instead of decreasing it, its physical interpretation could use further explanation. Softening could be achieved by decreasing the overall isotropic stiffness, and inserting two anisotropic fibers orthogonal to the desired softening direction. However, this control is indirect, because two directions must be specified, and both the isotropic and anisotropic stiffnesses

must be tuned. Instead we propose anisotropic softening, which allows for simpler and more direct control.

Next, we present the closed-form eigensystems for several common anisotropic energies. These expressions are themselves novel, and can be used to construct fast and concise Newton implementations. However, the energies fall short of our requirements, so we will use their drawbacks to motivate the design of a new energy.

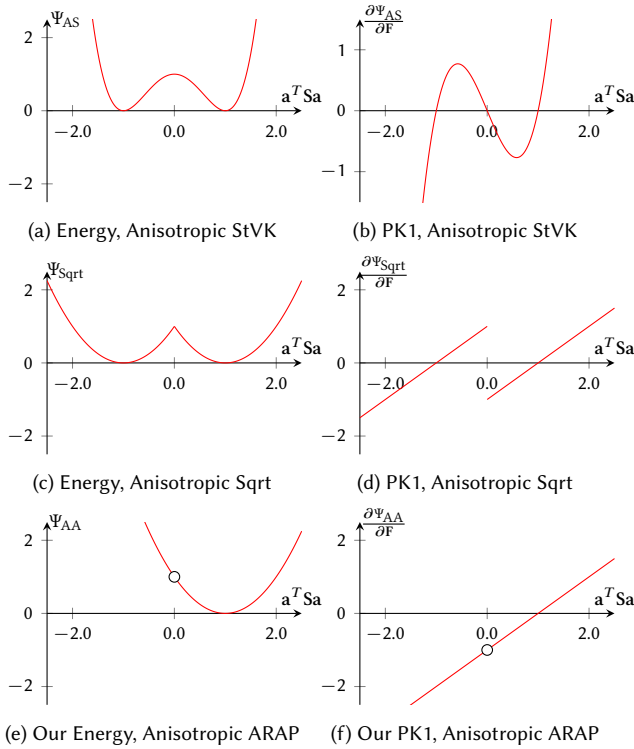


Fig. 2. The Anisotropic StVK energy grows faster than the others, making softening impossible, and also contains two spurious minima. The Anisotropic Sqrt energy contains a troublesome C_1 discontinuity, and its PK1 contains both a C_0 discontinuity and a spurious root. Our Anisotropic ARAP energy contains no spurious minima, but has a point discontinuity at $a^T Sa = 0$. It is easily removed by patching the value from $a^T Sa = 2$. The $a^T Sa$ measure receives a more formal treatment in §5.2.1

5.1 Eigensystems of Existing Energies

5.1.1 Anisotropic StVK. The following energy is common in both graphics [Liu et al. 2017] and biomechanics [Holzapfel 2005]:

$$\Psi_{AS} = \mu/2(I_5 - 1)^2 \quad (17)$$

$$\frac{\partial \Psi_{AS}}{\partial \mathbf{F}} = 2\mu(I_5 - 1)\mathbf{F}\mathbf{A} \quad (18)$$

$$\frac{\partial^2 \Psi_{AS}}{\partial \mathbf{f}^2} = 2\mu \left((I_5 - 1)\mathbf{H}_5 + 2\mathbf{f}_a \mathbf{f}_a^T \right). \quad (19)$$

Due to its close resemblance to the quartic stretching term from the isotropic Saint Venant-Kirchhoff (StVK) energy [Barbić and James 2005; Bonet and Wood 2008], $\Psi = \mu\|\mathbf{C} - \mathbf{I}\|_F^2$, where $\|\cdot\|_F^2$ is the

squared Frobenius norm, we refer to this as the *Anisotropic StVK* energy, or AS for short. The eigenvalues for this energy follow from Eqns. 11 and 13:

$$\lambda_0 = 2\mu(I_5 - 1) + 4\mu I_5 \quad (20)$$

$$\lambda_{1,2} = 2\mu(I_5 - 1). \quad (21)$$

The eigenvectors are as described in §4.2.

This energy fails to meet all of our criteria. As shown in Fig. 2a, since the model is quartic, its forces grow too fast to enable softening of quadratic isotropic energies. Instead, it produces large negative forces under softening that halt the simulation. Similar to the isotropic StVK model, it also contains a spurious stable state when flattened ($I_5 = 0$) and reflected (Fig. 2b). Again, this occurs because I_5 is based on the C invariant, which cannot detect inversion because it squares away the signs of the singular values.

5.1.2 Anisotropic Square Root. The polynomial order of the previous model can be improved to quadratic by applying a square root [Alastrué et al. 2008]:

$$\Psi_{Sqrt} = \mu/2(\sqrt{I_5} - 1)^2 \quad (22)$$

$$\frac{\partial \Psi_{Sqrt}}{\partial \mathbf{F}} = \mu \left(1 - \frac{1}{\sqrt{I_5}} \right) \mathbf{F}\mathbf{A} \quad (23)$$

$$\frac{\partial^2 \Psi_{Sqrt}}{\partial \mathbf{f}^2} = \mu \left[\left(1 - \frac{1}{\sqrt{I_5}} \right) \mathbf{H}_5 + \frac{1}{I_5^{3/2}} \mathbf{f}_a \mathbf{f}_a^T \right]. \quad (24)$$

The eigenvalues are again straightforward:

$$\lambda_0 = \mu \quad (25)$$

$$\lambda_{1,2} = \mu \left(1 - \frac{1}{\sqrt{I_5}} \right). \quad (26)$$

While this energy addresses our first criterion, it is still based on I_5 , and therefore does not satisfy the second. The spurious stable state under reflection remains (Fig. 2c), and the meta-stable state at $I_5 = 0$ is now replaced with a C_0 force discontinuity (Fig. 2d). Fig. 4 shows that this introduces a variety of troublesome stable states under inversion which become dominant as the fiber stiffness increases.

Alternative Energies. Other I_5 -based energies exist, such as the piecewise-cubic and piecewise-exponential models of Blemker et al. [2005]. As with the prior energies, we can derive closed-form expressions for their eigensystems (Appendix D). They use higher-order nonlinearities, so they re-introduce the problems from our first criterion while failing to address the second one.

5.2 An Inversion-Safe Anisotropic Energy

The core problem is that I_5 is based on C, which squares away inversion information, so any energy that relies solely on this invariant will fail to meet our criteria. Smith et al. [2019] observed that formulating lower-order isotropic invariants ($I_1 = \text{tr}(\mathbf{S})$, where $\mathbf{S} = \mathbf{R}^T \mathbf{F}$), expands the space of invariant-expressible isotropic energies into the inversion-safe regime. The S matrix does not square away the signs of the singular values, the inversion information is preserved, and rotation invariance is maintained. We now show that a similar approach can be applied to anisotropic energies.

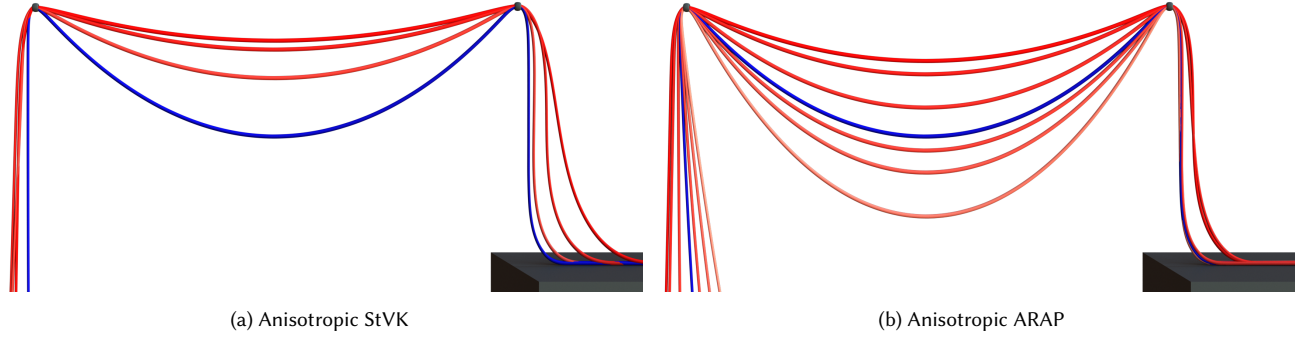


Fig. 3. A volumetric, tetrahedral rope with isotropic-only material is shown in blue. In red, progressively stiffer fibers are inserted in the rest-state longitudinal direction. *Left:* The Anisotropic StVK model (§5.1.1) can only be stiffened to decrease the slack. *Right:* Our Anisotropic ARAP (§5.2.2) model allows the slack to be both decreased (stiffened) and increased (softened).

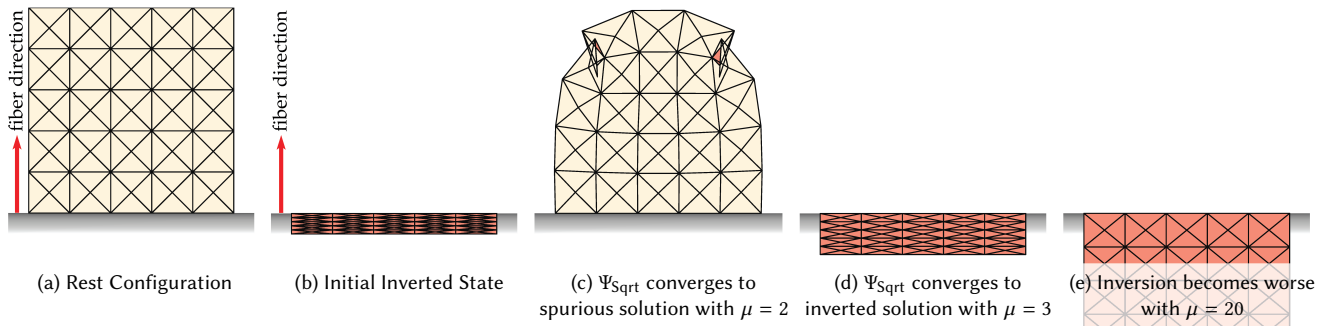


Fig. 4. The Anisotropic Square Root model (Ψ_{Sqrt} in §5.1.2) behaves badly under inversion. A 2D mesh composed of isotropic ARAP material in (a) is initially inverted into (b). When $\mu = 2$ in (c), Ψ_{Sqrt} takes 32 Newton iterations and 15,217 PCG iterations to converge to a spurious solution. Increasing μ to 3 in (d) takes 1 Newton iteration and 165 PCG iterations but finds an *inverted* solution. Increasing to $\mu = 20$ in (e) biases the solution even more towards inversion. Our Anisotropic ARAP model (Ψ_{AA}) in §5.2.2 recovers the correct configuration (a) in a single Newton iteration. For $\mu = 2$, it takes 113 PCG iterations, $\mu = 3$ takes 112, and $\mu = 20$ takes 94. Stiffening our material *accelerates* the solve in this scenario.

5.2.1 *A Lower-Order Invariant.* We use the same $\mathbf{C} \rightarrow \mathbf{S}$ substitution to formulate a new, anisotropic invariant: $I_4 = \mathbf{a}^T \mathbf{S} \mathbf{a}$. This can be written in several different forms:

$$I_4 = \mathbf{a}^T \mathbf{S} \mathbf{a} = \text{tr}(\mathbf{S} \mathbf{A}) = \text{tr}(\mathbf{R}^T \mathbf{F} \mathbf{A}). \quad (27)$$

However, unlike the isotropic case, problems immediately appear when the PK1 is computed:

$$\frac{\partial I_4}{\partial \mathbf{F}} = \frac{\partial \mathbf{R}}{\partial \mathbf{F}} : \mathbf{F} \mathbf{A} + \mathbf{R} \mathbf{A}. \quad (28)$$

A 4th-order rotation gradient term appears, $\frac{\partial \mathbf{R}}{\partial \mathbf{F}}$, which usually only occurs in Hessians, not PK1s. The problem it introduces can be seen by expanding the double-contraction term in 2D. By writing the rotation gradient in closed form [Smith et al. 2019] using an eigenmatrix,

$$\mathbf{G} = \frac{1}{\sqrt{2}} \mathbf{U} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mathbf{V}^T, \quad (29)$$

and pushing this through the double-contraction, we obtain:

$$\frac{\partial \mathbf{R}}{\partial \mathbf{F}} : \mathbf{F} \mathbf{A} = \sqrt{2} \hat{\mathbf{a}}_x \hat{\mathbf{a}}_y \begin{pmatrix} \sigma_x - \sigma_y \\ \sigma_x + \sigma_y \end{pmatrix} \mathbf{G}. \quad (30)$$

As in Eqn. 16, $\hat{\mathbf{a}} = \mathbf{V}^T \mathbf{a}$, and $\hat{\mathbf{a}}_{x,y}$ refer to the entries of $\hat{\mathbf{a}}$. The $\frac{\sigma_x - \sigma_y}{\sigma_x + \sigma_y}$ term is the source of the trouble, because as an element approaches $\sigma_x = -\sigma_y$, the forces become singular, which includes flattened or reflected elements. The situation worsens in 3D, because two more singular terms appear, $\frac{\sigma_x - \sigma_z}{\sigma_x + \sigma_z}$ and $\frac{\sigma_y - \sigma_z}{\sigma_y + \sigma_z}$.

Optimistic that these configurations were rare, we implemented a naïve I_4 -based energy (see Appendix C) in our production simulator, but a singular configuration was encountered in the *very first* simulation we ran (i.e. the scene in Fig. 3) and the simulation halted. Clearly these configurations are quite common, so the singularities need to be addressed.

5.2.2 *An Inversion-Safe Anisotropic Energy.* The main advantage of I_4 is that it produces a sign-preserving stretch measure in the fiber direction. How can we keep this sign information while suppressing the PK1 singularities? We propose an energy that accomplishes this,

$$\Psi_{\text{AA}} = \mu/2 \left(\sqrt{I_5} - \mathcal{S}(I_4) \right)^2, \quad (31)$$

where $S(x)$ is the signum function:

$$S(x) = \begin{cases} -1 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases}. \quad (32)$$

We call this an Anisotropic ARAP (AA) energy because it is both quadratic and inversion-safe. The sign information from I_4 is preserved, and the singularities are filtered from the PK1 because $\frac{\partial S(x)}{\partial x} = 2\delta(x)$, where $\delta(x)$ is the Dirac delta:

$$\delta(x) = \begin{cases} 0 & \text{if } x \neq 0 \\ \infty & \text{if } x = 0 \end{cases}. \quad (33)$$

To see the singularity filtering, we write out the PK1:

$$\begin{aligned} \frac{\partial \Psi_{AA}}{\partial \mathbf{F}} &= \mu \left(\sqrt{I_5} - S(I_4) \right) \left[\frac{\partial \sqrt{I_5}}{\partial \mathbf{F}} - \frac{\partial S(I_4)}{\partial \mathbf{F}} \right] \\ &= \mu \left(\sqrt{I_5} - S(I_4) \right) \left[\frac{1}{2\sqrt{I_5}} \mathbf{F} \mathbf{A} - \delta(I_4) \frac{\partial I_4}{\partial \mathbf{F}} \right]. \end{aligned} \quad (34)$$

The troubling $\frac{\partial I_4}{\partial \mathbf{F}}$ term still appears, but it is masked by the $\delta(I_4)$, which limits its influence to zero everywhere except $I_4 = 0$. The ∞ at $x = 0$ is a removable singularity that we will address shortly. The effective PK1 everywhere except $I_4 = 0$ is then:

$$\frac{\partial \Psi_{AA}}{\partial \mathbf{F}} = \mu \left(1 - \frac{S(I_4)}{\sqrt{I_5}} \right) \mathbf{F} \mathbf{A}. \quad (35)$$

The Hessian is correspondingly:

$$\frac{\partial^2 \Psi_{AA}}{\partial \mathbf{f}} = \mu \left[\left(1 - \frac{S(I_4)}{\sqrt{I_5}} \right) \mathbf{H}_5 + \frac{S(I_4)}{I_5^{3/2}} \mathbf{f}_a \mathbf{f}_a^T \right]. \quad (36)$$

Since $S(I_4)$ is constant everywhere but $I_4 = 0$, we can apply our I_5 eigenanalysis to obtain the eigenvalues:

$$\lambda_0 = \mu \quad (37)$$

$$\lambda_{1,2} = \mu \left(1 - \frac{S(I_4)}{\sqrt{I_5}} \right). \quad (38)$$

The expressions are almost exactly the same as the Anisotropic Square Root model, with the addition of $S(I_4)$. This subtle change has a dramatic effect on inverted configurations. The Anisotropic Square Root model converged slowly to spurious minima, but our energy finds the correct state in a single Newton iteration (Fig. 4). Moreover, stiffening the fibers accelerates convergence to the correct solution, not the spurious one in Fig. 4e.

An important detail to address is the Dirac singularity at $I_4 = 0$. Unlike the $\frac{\partial I_4}{\partial \mathbf{F}}$ case, this is a well-behaved *removable* singularity, i.e. the function does not explode to infinity in the neighborhood of the discontinuity. Examining the plot of the energy (Fig. 2e), it is clear that C_0 continuity can be restored by patching the value of Ψ_{AA} at $\mathbf{a}^T \mathbf{S} \mathbf{a} = I_4 = 0$ with the value at $I_4 = 2$. Similarly, continuity can be restored to the PK1 by patching $\partial \Psi_{AA} / \partial \mathbf{F}$ at $I_4 = 0$ with the negated PK1 from $I_4 = 2$ (Fig. 2f). A Hessian patch is not needed, because two of its eigenvalues are already negative when the fiber direction is under compression ($I_5^2 < 1$). Since $\lambda_{1,2}$ are zero, the

Hessian is computed from λ_0 and \mathbf{Q}_0 . If $I_5 = 0$, the Hessian is zeroed. We demonstrate the effectiveness of our new energy in §7.

6 ANISOTROPIC ELEMENT REHABILITATION

Next, we apply our anisotropic analysis to the problem of badly-conditioned elements. Nearly-zero volume elements can insidiously inflate the global condition number of the Hessian, causing iterative solvers to waste the majority of their time resolving visually negligible elements. The effect cascades further if a low-quality result triggers additional Newton iterations. Direct solvers are insensitive to the condition number, but zero-volume elements can still introduce NaNs into the forces and Hessians that also halt these solvers. We will show that the conditioning-ruining problems can be excised from the isotropic energy, and visual fidelity can be restored by inserting a well-conditioned anisotropic energy.

6.1 How Bad Elements Ruin Conditioning

We investigate the case of force computation, where all the relevant conditioning-ruining features already appear. Expanding the deformation gradient into $\mathbf{F} = \mathbf{D}_s \mathbf{D}_m^{-1}$, we didactically define the tetrahedral case,

$$\mathbf{D}_s = \begin{bmatrix} \mathbf{x}_1 - \mathbf{x}_0 & \mathbf{x}_2 - \mathbf{x}_0 & \mathbf{x}_3 - \mathbf{x}_0 \end{bmatrix} = \begin{bmatrix} \mathbf{e}_0 & \mathbf{e}_1 & \mathbf{e}_2 \end{bmatrix} \quad (39)$$

$$\mathbf{D}_m = \begin{bmatrix} \bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_0 & \bar{\mathbf{x}}_2 - \bar{\mathbf{x}}_0 & \bar{\mathbf{x}}_3 - \bar{\mathbf{x}}_0 \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{e}}_0 & \bar{\mathbf{e}}_1 & \bar{\mathbf{e}}_2 \end{bmatrix}, \quad (40)$$

where \mathbf{x}_i and $\bar{\mathbf{x}}_i$ are respectively the nodes of the tetrahedron in pose- and rest-space. We expand the change-of-basis to $\frac{\partial \mathbf{F}}{\partial \mathbf{u}} = \mathbf{v} \left(\frac{\partial \mathbf{D}_s}{\partial \mathbf{u}} \cdot \mathbf{D}_m^{-1} \right)$, where \mathbf{v} is the rest-space element volume, and rewrite Eqn. 1 as,

$$\mathbf{g} = \mathbf{v} \left[\frac{\partial \mathbf{D}_s}{\partial \mathbf{u}} \cdot \mathbf{D}_m^{-1} \right]^T : \mathbf{P} \left(\mathbf{D}_s \mathbf{D}_m^{-1} \right). \quad (41)$$

If the element is degenerate, \mathbf{D}_m will be nearly rank-deficient, and the entries of \mathbf{D}_m^{-1} will be spuriously large. The inverse then creates two problems in Eqn. 41:

- (1) To the left of the double-contraction, in the expanded $\frac{\partial \mathbf{F}}{\partial \mathbf{u}}$, the \mathbf{D}_m^{-1} will transform a well-behaved PK1 into large, badly-behaved forces.
- (2) To the right of the double-contraction, the $\mathbf{P}(\cdot)$ will catastrophically amplify the large entries of \mathbf{D}_m^{-1} if the isotropic energy is highly non-linear.

To rehabilitate a badly-conditioned element, both of these problems must be addressed. The small \mathbf{v} in Eqn. 41 does not address the core problems, because the $\mathbf{P}(\cdot)$ grows non-linearly and overwhelms this constant.

A natural attempt is to approximate the inverse with a Moore-Penrose pseudo-inverse, i.e. $\mathbf{D}_m^{-1} \approx \mathbf{D}_m^\dagger$. This suffices for the first problem because $\frac{\partial \mathbf{F}}{\partial \mathbf{u}}$ is linear, but it fails when applied to the second problem of the non-linear $\mathbf{P}(\cdot)$. The \mathbf{D}_m^\dagger constrains one of the singular values of \mathbf{F} to zero, which corresponds to a highly compressed

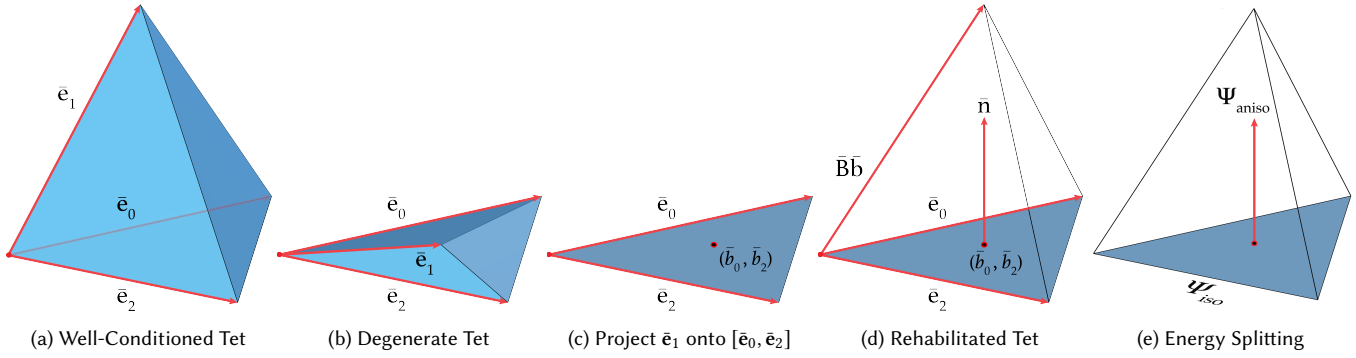


Fig. 5. (a) A well-conditioned tet spans three directions, $[\bar{e}_0, \bar{e}_1, \bar{e}_2]$. (b) A pancaked, badly-conditioned tet has a degenerate direction, \bar{e}_1 , very close to the span $[\bar{e}_0, \bar{e}_2]$. (c) We explicitly project \bar{e}_1 onto the $[\bar{e}_0, \bar{e}_2]$ plane and obtain its barycentric coordinate (\bar{b}_0, \bar{b}_2) . (d) We use this coordinate and the normal \bar{n} to build a non-degenerate direction, $\bar{B}\bar{b}$. (e) Finally, we use $\bar{B}\bar{b}$ to limit the influence of the isotropic energy, Ψ_{iso} , to the $[\bar{e}_0, \bar{e}_2]$ plane. We then inject a well-behaved anisotropic energy, Ψ_{aniso} , in the degenerate direction to stably preserve the original, pancake shape.

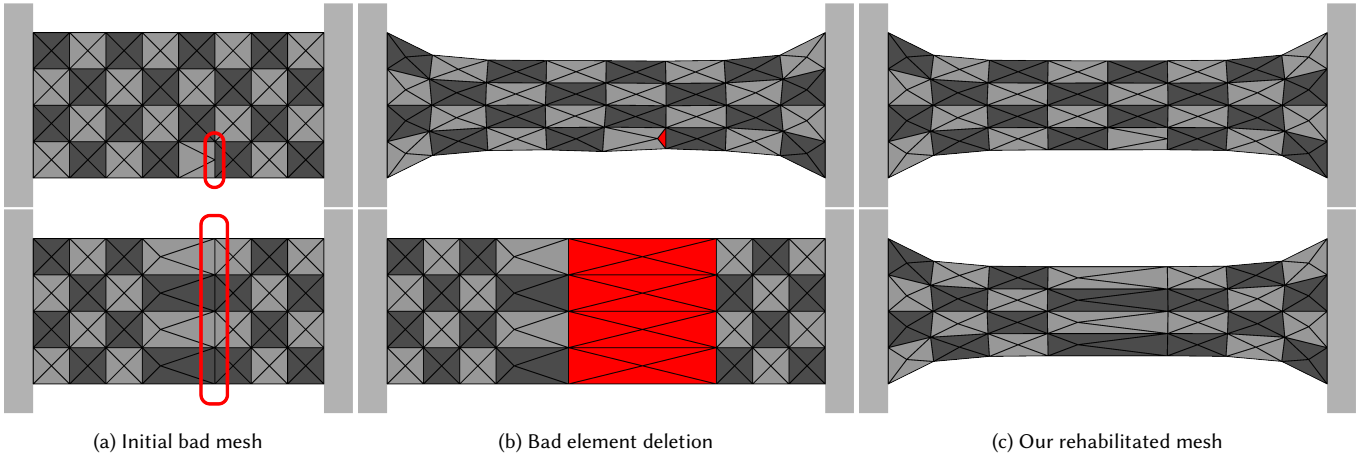


Fig. 6. (a) We ruin the conditioning of both a single element (top row) and an entire column of elements (bottom row). (b) Element deletion [Forest et al. 2002] can be used, but cracking artifacts appear (shown in red) and the clean line along the bottom of the mesh develops a bump (top row, middle image). In the extreme, deletion can cause the mesh to disconnect (bottom row, middle image). (c) Our rehabilitation approach produces a result very close to the well-conditioned solution (see video).

element. The isotropic energy then dutifully generates large restorative forces, but since the singular value is constrained, the forces persist and the simulation diverges. However, this line of inquiry is still promising, because Iben [2007] showed that D_m^\dagger can suffice if $P(\cdot)$ is defined *procedurally*, as then only the first problem in Eqn. 41 appears. We show how to extend this intuition to the general case.

6.2 Rehabilitating A Bad Element

Our element rehabilitation approach is outlined in Fig. 5. From a geometric perspective, the isotropic energy is never shown a degenerate tet with near-zero altitude (Fig. 5b). Instead, it sees a *rehabilitated* tet with finite altitude (Fig. 5d) that is rigid in one direction, i.e. one singular value is constrained to equal one. Deformation in the badly-behaved direction is instead handled by a well-conditioned anisotropic energy (Fig. 5e).

Deleting the Degeneracy. We construct an $F_{iso} = D_{s,iso} D_{m,iso}^{-1}$ for the isotropic energy that deletes the degenerate direction and guarantees that $D_{m,iso}$ is full-rank. In the following, we assume that \bar{e}_1 is the degenerate direction, but the other permutations are described in Appendix E. We compute a modified D_m ,

$$D_{m,iso} = \begin{bmatrix} \bar{e}_0 & \bar{B}\bar{b} & \bar{e}_2 \end{bmatrix}, \quad (42)$$

where

$$\bar{B} = \begin{bmatrix} \bar{e}_0 & \bar{e}_2 & \bar{n} \end{bmatrix} \text{ and } \bar{b} = \begin{bmatrix} \bar{b}_0 \\ \bar{b}_2 \\ \|\bar{e}_{max}\| \end{bmatrix}. \quad (43)$$

Here, \bar{b}_0 and \bar{b}_2 are the barycentric coordinates of \bar{e}_1 projected onto the span of \bar{e}_0 and \bar{e}_2 (Fig. 5c), $\|\bar{e}_{max}\| = \max(\|\bar{e}_0\|, \|\bar{e}_2\|)$ and

$\bar{\mathbf{n}} = \frac{\bar{\mathbf{e}}_2 \times \bar{\mathbf{e}}_0}{\|\bar{\mathbf{e}}_2 \times \bar{\mathbf{e}}_0\|}$ is the inward-facing normal of the plane spanned by $\bar{\mathbf{e}}_0$ and $\bar{\mathbf{e}}_2$ (Fig. 5d).

The construction of the new, rehabilitated tet can be seen by expanding $\bar{\mathbf{B}}\bar{\mathbf{b}} = \bar{b}_0\bar{\mathbf{e}}_0 + \bar{b}_2\bar{\mathbf{e}}_2 + \|\bar{\mathbf{e}}_{\max}\|\bar{\mathbf{n}}$. The plane formed by $\bar{\mathbf{e}}_0$ and $\bar{\mathbf{e}}_2$ is well-defined, but when $\bar{\mathbf{e}}_1$ is close to this plane, \mathbf{D}_m is near-degenerate (Fig. 5b). We replace this bad edge with a well-formed one by first projecting $\bar{\mathbf{e}}_1$ onto the $(\bar{\mathbf{e}}_0, \bar{\mathbf{e}}_2)$ plane, i.e. $\bar{b}_0\bar{\mathbf{e}}_0 + \bar{b}_2\bar{\mathbf{e}}_2$ (Fig. 5c). Then, we displace the projection a finite amount along the normal, $\|\bar{\mathbf{e}}_{\max}\|\bar{\mathbf{n}}$, and use this as our new edge (Fig. 5d).

The modified \mathbf{D}_s is similar:

$$\mathbf{D}_{s,\text{iso}} = \begin{bmatrix} \mathbf{e}_0 & \mathbf{B}\bar{\mathbf{b}} & \mathbf{e}_2 \end{bmatrix} \quad \text{and} \quad \mathbf{B} = \begin{bmatrix} \mathbf{e}_0 & \mathbf{e}_2 & \mathbf{n} \end{bmatrix}, \quad (44)$$

where $\mathbf{n} = \frac{\bar{\mathbf{e}}_2 \times \bar{\mathbf{e}}_0}{\|\bar{\mathbf{e}}_2 \times \bar{\mathbf{e}}_0\|}$. Crucially, we re-use the rest-space $\bar{\mathbf{b}}$ from $\mathbf{D}_{m,\text{iso}}$. This ensures that the rehabilitated direction will be the same length ($\|\bar{\mathbf{e}}_{\max}\|$) in both rest- and pose-space, and the singular value of $\mathbf{F}_{\text{iso}} = \mathbf{D}_{s,\text{iso}}\mathbf{D}_{m,\text{iso}}^{-1}$ in this direction will always equal one. The second problem from Eqn. 41 has now been addressed, i.e. $\mathbf{P}(\mathbf{F}_{\text{iso}})$ can never produce a (badly-behaved) force in the degenerate direction, because it never sees it. Our supplement describes how a similar approach applies when an element has two degenerate directions.

Adding Anisotropic Energy. Removing the badly-behaved isotropic forces can greatly improve the conditioning of the mesh Hessian. However, the deleted forces must be accounted for to prevent cracking artifacts (Fig. 6b, top row, middle), or in the extreme, non-physical tearing (Fig. 6b, bottom row, middle).

We observe that since the degenerate element is a pancake, the visually consistent choice is to select an energy that encourages the element to flatten. We accomplish this by inserting $\mathbf{F}_{\text{aniso}} = \mathbf{D}_s\mathbf{D}_{m,\text{iso}}^{-1}$ into the *Anisotropic Dirichlet* energy,

$$\Psi_{\text{D}}(\mathbf{F}_{\text{aniso}}) = \mu_{\text{aniso}}I_5, \quad (45)$$

in the fiber direction $\mathbf{a} = \bar{\mathbf{n}}$. Unlike the other models, Ψ_{D} is minimal when the fiber length is zero, so it is not *rest stable* [Smith et al. 2018] (i.e. $\mathbf{P}_{\text{D}}(\mathbf{I}) \neq 0$) which is the reason we use it. From §4, we also know that its eigenvalues are constant, $\lambda_{0,1,2} = 2\mu$, so it is always well-conditioned.

Computing the anisotropic force requires some care. We use $\mathbf{F}_{\text{aniso}} = \mathbf{D}_s\mathbf{D}_{m,\text{iso}}^{-1}$ to compute the PK1, because using the original \mathbf{D}_m^{-1} produces badly-behaved results. Intuitively, Ψ_{D} needs to measure the distance of the fiber length from zero, but the rest-space fiber length from \mathbf{D}_m is sufficiently close to zero that the measurement is numerically ambiguous. Using $\mathbf{D}_{m,\text{iso}}$ removes the ambiguity.

Finally, in order to ensure that the volume constant v from Eqn. 41 does not nullify this force, we set $\mu_{\text{aniso}} = \frac{\mu_{\text{iso}}\sqrt{2}}{v \cdot 12} \|\bar{\mathbf{e}}_{\max}\|^3$, where $\frac{\sqrt{2}}{12} \|\bar{\mathbf{e}}_{\max}\|^3$ is the volume of the nearest equilateral tetrahedron, and μ_{iso} is the Lamé parameter from the original isotropic material. This will also lead to visually consistent results if two adjacent rehabilitated elements have different degeneracy directions. The element with the larger non-degenerate face will produce the larger force, which will then work to reduce to most significant visual artifact.

Using this overall approach, the cracking and tearing artifacts from the rehabilitated mesh are mitigated (Fig. 6c).

Computing the $\frac{\partial \mathbf{F}}{\partial \mathbf{u}}$ tensors. We have proposed two new deformation gradients, \mathbf{F}_{iso} and $\mathbf{F}_{\text{aniso}}$, so two new change-of-basis tensors $\frac{\partial \mathbf{F}_{\text{iso}}}{\partial \mathbf{u}}$ and $\frac{\partial \mathbf{F}_{\text{aniso}}}{\partial \mathbf{u}}$ are needed. These tensors only depend on the pose-space positions, so the anisotropic case is simple: $\frac{\partial \mathbf{F}_{\text{aniso}}}{\partial \mathbf{u}} = \frac{\partial \mathbf{D}_s}{\partial \mathbf{u}} \cdot \mathbf{D}_{m,\text{iso}}^{-1}$. The $\frac{\partial \mathbf{F}_{\text{iso}}}{\partial \mathbf{u}}$ tensor requires more attention, because the original $\frac{\partial \mathbf{D}_s}{\partial \mathbf{u}}$ tensor contained only constants, but $\frac{\partial \mathbf{D}_{s,\text{iso}}}{\partial \mathbf{u}}$ now contains non-constant entries.

After straightforward but tedious manipulations, compact expressions for $\frac{\partial \mathbf{D}_{s,\text{iso}}}{\partial \mathbf{u}}$ can be obtained. The details are in the supplement, but the main finding is that the $\frac{\partial \mathbf{F}}{\partial \mathbf{u}} = \frac{\partial \mathbf{D}_s}{\partial \mathbf{u}} \cdot \mathbf{D}_m^{-1}$ expression from Iben [2007] is actually an *excellent* approximation of the exact $\frac{\partial \mathbf{F}_{\text{iso}}}{\partial \mathbf{u}}$ tensor. We ran simulations using both tensors, and found the differences to be visually negligible. All of the components of element rehabilitation are now in place, and the expressions are assembled in Appendix F.

6.3 When to Rehabilitate?

We now describe *when* to rehabilitate a badly-conditioned element. There many element degeneracy measures [Shewchuk 2002], and we use the Knupp [2003] score $\kappa(\mathbf{D}_m) = \kappa_{\text{angle}}(\mathbf{D}_m) \cdot \kappa_{\text{vol}}(\mathbf{D}_m)$, where

$$\kappa_{\text{vol}}(\mathbf{D}_m) = v \frac{12}{\sqrt{2} \|\mathbf{e}_{\max}\|^3} \quad (46)$$

$$\kappa_{\text{angle}}(\mathbf{D}_m) = 3 \left(\sqrt{2}v \right)^{2/3} \left[\frac{7}{4} \|\mathbf{D}_m\|_F^2 - \frac{1}{4} \text{tr} \left(\mathbf{J}_3 \mathbf{D}_m^T \mathbf{D}_m \right) \right]^{-1}, \quad (47)$$

the volume term has been constrained to compare against the closest equilateral tetrahedron with edge length $\|\mathbf{e}_{\max}\|^3$, and \mathbf{J}_3 denotes a 3×3 matrix filled with all ones. The \mathbf{D}_m terms on the right hand side of $\kappa_{\text{angle}}(\mathbf{D}_m)$ essentially sum the diagonal terms of \mathbf{D}_m and then subtract off the sum of the off-diagonal terms. Intuitively, $\kappa_{\text{angle}}(\mathbf{D}_m) = 1$ when the element is equilateral, and becomes zero when the element pancakes. However, the broad trends that we observed of quality measure vs. matrix conditioning was virtually identical regardless of the measure used.

We ran a 3D stretch test similar to Fig. 6 on a cube mesh containing 100K tetrahedra, with materials set uniformly to $\mu = 1$ and, when applicable, $\lambda = 100$. We then progressively ruined the conditioning of a single tet (Fig. 7, left) and computed the condition number of the global stiffness matrix using Spectra [Qiu 2018].

As a tet approached total degeneracy ($\kappa = 0$, Fig. 7), the condition number increased exponentially, and the choice of material model only translated the trend by a constant. We computed the same plots at the rest pose after scaling the mesh by a factor of 2 along the y -axis (Fig. 7), and also by ruining the conditioning of an entire z -slice of the cube. While the translations changed, the trend remained exactly the same, even in 2D. Additional scenarios are shown in the supplement. Based on these experiments, if an element's κ suggests that it will ruin the global conditioning by more than *two orders of magnitude*, then we chose to rehabilitate. Across all 3D isotropic energies we examined, this corresponds to $\kappa \approx 10^{-5}$.

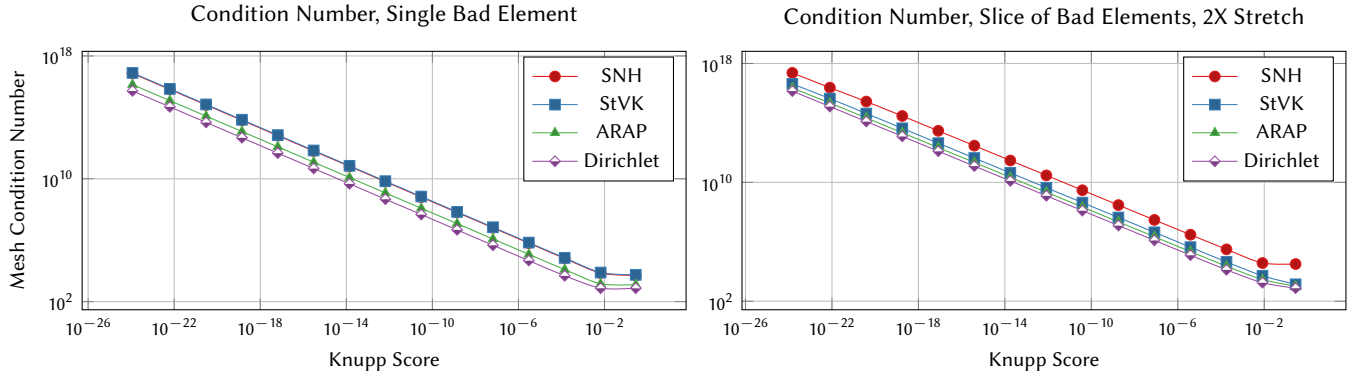


Fig. 7. Condition number of a 3D version of Fig. 6 as different strategies are used to progressively ruin the conditioning. The exponential trend is the same regardless of the material model or ruining strategy. In the left plot, the SNH and StVK points are almost identical, so only the StVK plot is visible.

7 DISCUSSION AND RESULTS

7.1 Anisotropic ARAP

2D Square. We constructed a 2D square mesh composed of 100 triangles and isotropic ARAP material with $\mu_{iso} = 1$ (Fig. 4). The fiber is in the positive y direction, and we initially inverted the square (Fig. 4b) to observe its behavior in this regime. We first applied the Anisotropic Square Root model (Eqn. 24) with $\mu = 2$, which is close to that of the isotropic model. Due to the presence of many spurious solutions, the Newton solver converged slowly to a tangled configuration (Fig. 4c). At higher μ , the anisotropic stiffness overwhelms the isotropic forces, and the solver converges to an inverted solution (Fig. 4d). As μ increases, the spurious reflected solution dominates, and the model becomes progressively *more* inverted (Fig. 4e).

In contrast, our Anisotropic ARAP model (Eqn. 31) converged to the correct solution (i.e. reproduces Fig. 4a) after a single Newton iteration. Higher values of μ accelerated the speed of the solve, as it provided a stronger force towards the rest configuration.

Volumetric Rope Slack. We constructed a volumetric rope composed of a mesh containing 16,800 tetrahedra, 18,927 degrees of freedom, and isotropic ARAP material. We then used anisotropy to increase and decrease its slack in the longitudinal direction while keeping the transverse shape intact.

The Anisotropic StVK model (Eqn. 17) is applied in Fig. 3a, where the fiber direction is defined along the longitudinal direction of the rope (i.e. positive x). The slack decreases quickly as μ is increased, but the model cannot increase the slack, because the quartic model grows too quickly along the fiber direction. Setting $\mu < 0$ causes the simulation to immediately diverge.

Our Anisotropic ARAP model in Fig. 3b successfully increases and decreases the slack. The model is less non-linear, so it also stiffens in a more predictably linear manner as μ is increased. The Anisotropic ARAP model ran slightly faster than the StVK model, e.g. 3.75 secs per frame instead of 3.99 secs with $\mu = 10$, while running on 8 cores of a 2.3 GHz Intel Xeon.

Octopus Hammock. The octopus example in Fig. 1 and 8 is composed of a mesh containing 927,368 tetrahedra, 592,692 degrees of

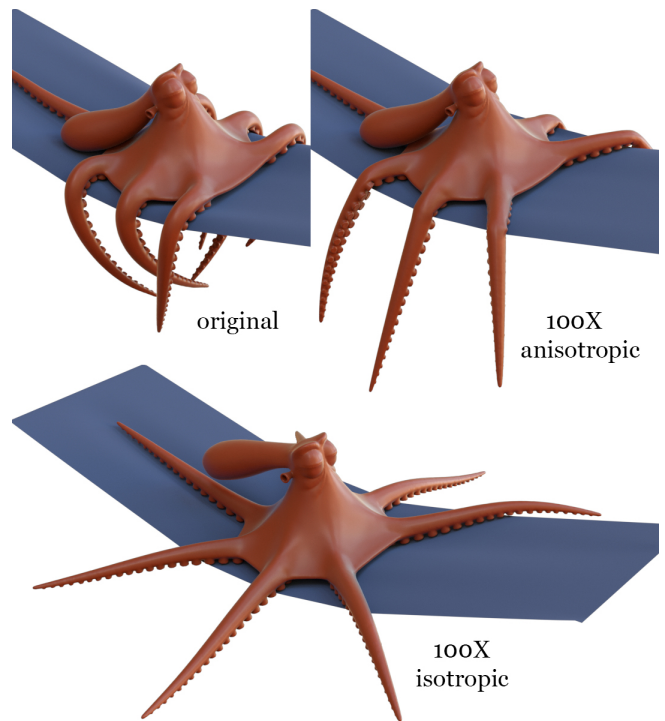


Fig. 8. Starting from the original simulation (top left), we stiffened fibers by 100× along the tentacles (top right). The tentacles become more rigid in the fiber direction, but still bend in the other directions. Increasing the isotropic stiffness by the same amount does not achieve the same look (bottom).

freedom, and isotropic ARAP material. We ran it in our Baraff and Witkin [1998]-style production simulator, and dropped the mesh onto a cloth hammock (Fig. 1). We then defined a fiber along each tentacle, and both increased and decreased the stiffness by two orders of magnitude. The tentacle shape in the longitudinal direction becomes more rigid or more slack, while the bending behavior in the other directions remains the same. In contrast, increasing the isotropic stiffness of the model (Fig. 8) stiffens the entire tentacle, and does not achieve an equivalent look.

In the softening case, the mesh becomes extremely stretchy, but our model still allows the dynamics to be handled robustly. As shown in the supplemental video, softening by two orders of magnitude (Fig. 1) essentially turns the tentacles into goo under extension. Since no remeshing is employed, the elements stretch out far beyond anything that would be used in a production environment. Widening the hammock (see video) shows that the tentacles lose volume under compression, because resistance has been decreased in the fiber direction. In all these extreme cases, our model provides a consistent and robust response. The simulation was run on 12 cores of a 2.3 GHz Intel Xeon. The original simulation took 48.61s per frame, and increasing the isotropic stiffness increased the running time, with 10× stiffness taking 67.92s and 100× taking 212.25s. Increasing the stiffness in the fiber direction also increased the running time with 10× taking 57.21s and 100× taking 64.76s. Decreasing the fiber stiffness resulted in slight speedups, with 0.1× stiffness taking 35.96s and 0.01× taking 32.32s.

We ran the same simulations using the brute-force 9×9 eigen-decomposition of the anisotropic Hessian at each quadrature point instead of our closed-form expressions. Our fast projection method consistently made Hessian computation 3.2× faster, and accelerated the overall simulation time by a maximum of 1.52× for the softest example and minimum of 1.29× for the stiffest example. The acceleration decreased in the stiffer examples because the number of necessary PCG iterations increased.

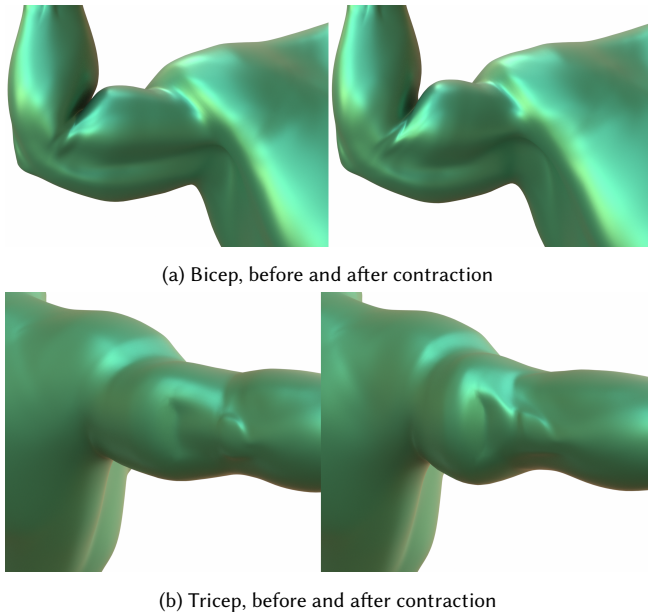


Fig. 9. Our Anisotropic ARAP model can be used for muscle contraction. The background Stable Neo-Hookean material is much more non-linear, but our quadratic Anisotropic ARAP model still achieves the characteristic shapes of contracted muscles. A pectoralis example is also in the video.

Muscle Contraction. Fiber models can be adapted for muscle contraction. We add a contraction length l , and an activation level a to

our Anisotropic ARAP model:

$$\Psi_{\text{muscle}} = a \left(\sqrt{I_5} - l \cdot \mathcal{S}(I_4) \right)^2. \quad (48)$$

We layered this model atop an isotropic Stable Neo-Hookean [Smith et al. 2018] material with high incompressibility ($\nu = 0.499$), and contracted a bicep, tricep (Fig. 9) and pectoralis (see video). While the muscle model is quadratic and the isotropic model is hexic, it achieves the same qualitative shapes as higher-order anisotropic models [Blemker et al. 2005]. The lower-order non-linearity does not appear to compromise the expressivity of the model.

The deformation was computed on a 25,822 element hexahedral mesh, with 8 quadrature points per element. The fiber directions were computed by modeling each muscle as a tetrahedral mesh, manually specifying origin and insertion points, and solving for a vector field using standard gradient-based techniques [Saito et al. 2015; Tong et al. 2003]. The directions were then transferred to the nearest quadrature points on the hexahedral mesh. The bicep contraction averaged 1.02 secs per frame, the tricep 5.82 secs, and the pectoral 9.74 secs. The running times increased with the number of quadrature points undergoing contraction (respectively, 2917, 3535 and 4124). Most of the time (respectively, 39%, 43% and 41%) was spent in Preconditioned Conjugate Gradients (PCG). We used the OpenMP version of PCG from ViennaCL [Rupp et al. 2016].

7.2 Mesh Rehabilitation

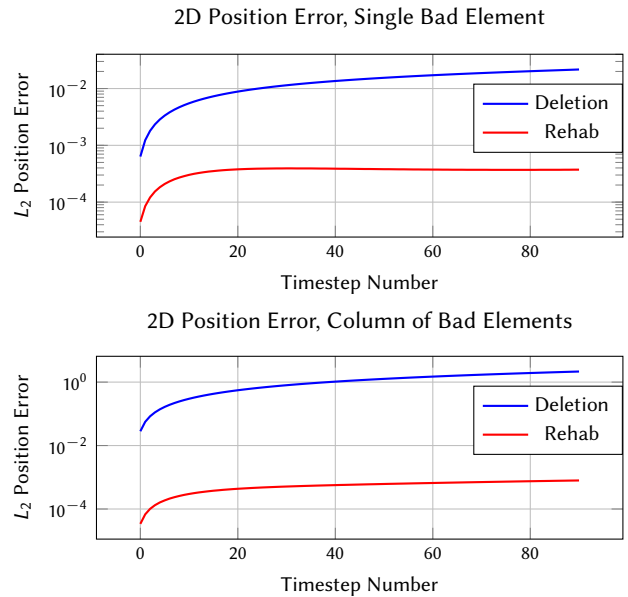


Fig. 10. Position error of a unit-length mesh similar to Fig. 6. Circled vertices from Fig. 6a are not included in the measurement. The error of the rehabbed mesh never exceeds $3.93e^{-4}$ (top) and $7.94e^{-4}$ (middle). On top, the rehabbed error slightly decreases over time.

2D Stretch. In Fig. 6a, we ruined the conditioning of a 2D mesh containing 200 triangles and 116 vertices (the mesh in Fig. 6 is

slightly sparser for legibility) composed of Stable Neo-Hookean material [Smith et al. 2018] by crushing a single element (top) and a column of elements (bottom). Deleting bad elements (Fig. 6b) introduces non-physical cracking and tearing, which we show in red. Our rehabilitated result matches the original well-conditioned solution up to four digits of accuracy (Fig. 10). The rehabilitated simulation took the exact same number of Newton iterations (2 per timestep) as the original well-conditioned mesh, and only increased the number of total PCG iterations by 10%.

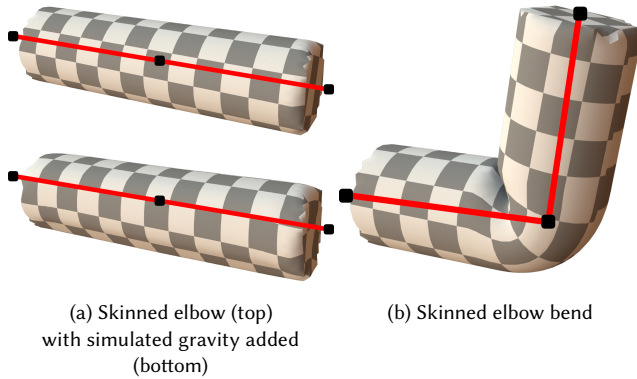


Fig. 11. A skinned, well-conditioned mesh has gravity added (a). As the elbow bends, the rest state is retargeted to the skinned geometry (b).

Rest-State Retargeting. We next examine two scenarios where the conditioning of a mesh has been ruined through the common production technique of *rest-state retargeting* [Gonzalez-Ochoa et al. 2002; Irving et al. 2008; Li et al. 2017; Milne et al. 2016; Wong et al. 2018]. Given a surface-based input animation, the goal is to add volumetric effects such as sagging under gravity, or jiggling under ballistic motion, but the effects should adhere as closely as possible to the original surface-based animation. A common approach is to use the animation’s rig to warp the volumetric mesh, and then use this warped version as the new rest-state. (The process can informally be viewed as animation-driven pseudo-plastic flow.) As the rig can contain arbitrarily deformations, the conditioning of the warped mesh can be arbitrarily ruined. However, each individual badly-conditioned tet only persists a handful of frames, so remeshing at each timestep is too drastic a solution.

One potential solution is to relax the warped mesh and improve its conditioning using a distortion-minimizing energy such as ARAP [Wong et al. 2018] or inversion-aware StVK [Li et al. 2017]. We found that this approach allows the simulation to proceed further than before, but the solve still fails under sufficient deformation. Relaxation techniques can improve the conditioning of individual elements, but badly-conditioned elements will eventually appear because these algorithms provide no global guarantee that *all* elements will be well-conditioned. Our rehabilitation approach instead circumvents the conditioning problem and allows the simulations to complete.

Retargeting 1: Elbow Bend. As a simple test (Fig. 11), we apply rest-state retargeting to a cylinder that has been rigged used linear blend skinning (LBS) [Jacobson et al. 2014] to bend like an elbow.

The mesh contains 38,032 tetrahedra and 7,977 vertices, is composed of Stable Neo-Hookean material [Smith et al. 2018], and was generated using Isosurface Stuffing [Labelle and Shewchuk 2007] (note the characteristic smoothed corners), so it is known to be well-conditioned. The tetrahedra that lie along the cylinder’s center line are hard-constrained to transform along with the bone, and a gravity load is applied to the entire mesh (Fig. 11a). We then run a quasistatic solve to resolve the deformation. Collisions are deactivated so that the forces only arise from mesh conditioning. The usual LBS joint-collapse artifact has been deliberately preserved because it is part of the input animation. The goal is to add gravity sagging onto this existing deformation.

As the elbow bend increases, the conditioning of the mesh gradually worsens until the Newton solve diverges. When using pure LBS-based warping, the simulation diverges at frame 132 (Fig. 12a). ARAP-based relaxation [Wong et al. 2018] allows the simulation to proceed further (Fig. 12b) to frame 140, and the inversion-aware StVK-based relaxation [Li et al. 2017] gets to frame 150 (Fig. 12c). Using our rehabilitation approach on the original LBS-based warp, the simulation completes the entire 160 frames (Fig. 12d). The element deletion approach [Forest et al. 2002] failed in this case, because several adjacent degenerate elements appeared, and deleting all their forces caused vertices to become totally unconstrained.

Retargeting 2: Big Baby. We have a real-world production example of a big baby animation that needs both collision resolution and gravity sagging, and is composed of a mesh containing 35,617 tetrahedra, 7,948 vertices, and Stable Neo-Hookean material.

The original simulation uses ARAP-based relaxation [Wong et al. 2018], but still diverges early in the sequence (Fig. 1), because the rig severely compresses elements around the right armpit. Our rehabilitation method allows the simulation to complete (Fig. 13). The original simulation averaged 1.60s seconds per frame before diverging, while ours averaged 1.34s over the entire sequence, so the cost of applying element rehabilitation is negligible.

8 FUTURE WORK

We have presented an inversion-safe transversely isotropic energy. While our energy is simple and robust, we have only begun to explore the energies that can be designed using our new I_4 invariant. Such design must be done carefully, and singularity-avoiding alternatives to our $S(I_4)$ formulation may be waiting to be uncovered.

Some biomechanics works [Nolan et al. 2014] define anisotropic invariants to account for shearing effects when multiple fiber direction are defined at a quadrature point. Formulating a fast Hessian projection scheme for these cases would require extending our existing analysis. However, the visual impact of such cross-fiber terms remains unproven.

We have presented an element rehabilitation method for badly-conditioned meshes. While this method was presented for tetrahedral meshes, it only relies on the columns of \mathbf{D}_m , which is defined at the quadrature point of any arbitrary spatial discretization. While it should be able to handle quadrature degeneracy for any arbitrary polyhedron, more work is needed to validate this hypothesis.

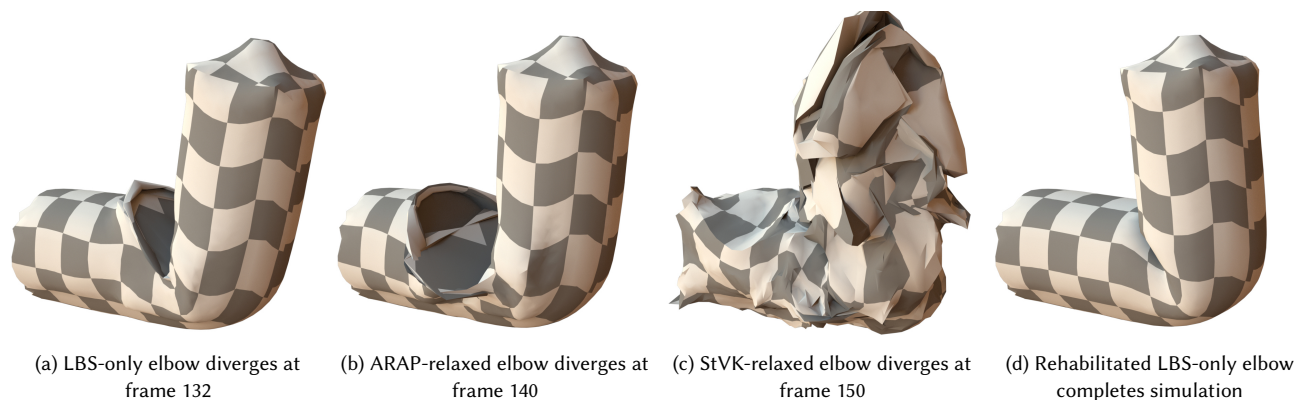


Fig. 12. The skinned elbow from Fig. 11, undergoing rest-state retargetted simulation. (a) Using LBS-only, the element conditioning is sufficiently ruined by frame 132 that the simulation diverges. (b) ARAP-based relaxation improves the conditioning, and allows the simulation to get to frame 140 before diverging. (c) StVK-based relaxation enables another 10 frames before diverging. (d) Only by using our rehabilitation scheme allows the simulation to complete.

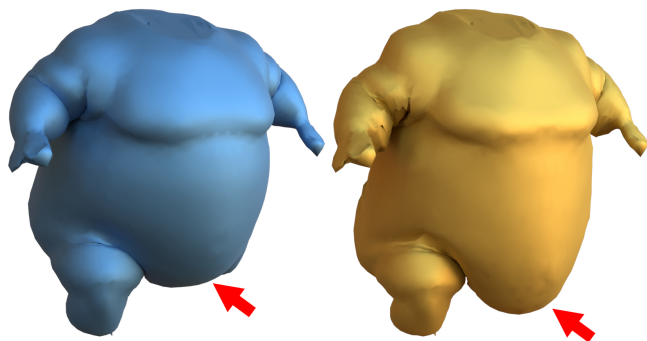


Fig. 13. On the left, in blue, is the original animation of the big baby. On the right in yellow, the rest state has been retargetted so that the physics adhere to the animation. The red arrow shows sagging under gravity that adds physicality to the animation. The effects of gravity have been exaggerated in this example for legibility; the actual production result is much subtler (see video). The original simulation diverged (Fig. 1), but our rehabilitated elements allow the sequence to complete.

Finally, we have presented a closed-form method for projecting the Hessian of any I_5 -based energy back to semi-positive definiteness, while a tighter projection would be for the Hessian of the entire energy, $\Psi = \Psi_{\text{iso}} + \Psi_{\text{aniso}}$. The eigenspaces of the two classes of energies do not appear to align, so it is not clear if a closed form approach is even possible. However, further work is needed to definitively answer this question.

ACKNOWLEDGMENTS

The authors would like to thank David Eberle for help with the Octopus example, Audrey Wong for both initially simulating and later resurrecting the Big Baby example, Bill Wise for laying out the fiber directions in the muscle contraction example, and Witawat Rungjiratananon for helping to integrate the fiber model into Presto. We would also like to thank Ryan Kautzman for early discussions regarding element degeneracy.

REFERENCES

- V. Alastrué, E. Peña, M. Martínez, and M. Doblaré. 2008. Experimental study and constitutive modelling of the passive mechanical properties of the ovine infrarenal vena cava tissue. *J. of Biomechanics* 41, 14 (2008), 3038–3045.
- M. Alexa, D. Cohen-Or, and D. Levin. 2000. As-rigid-as-possible Shape Interpolation. In *Proceedings of SIGGRAPH*. 157–164.
- D. Baraff and A. Witkin. 1998. Large Steps in Cloth Simulation. In *Proceedings of SIGGRAPH*. 43–54.
- J. Barbič and Doug L. James. 2005. Real-Time Subspace Integration for St. Venant-Kirchhoff Deformable Models. *ACM Trans. Graph.* 24, 3 (2005), 982–990.
- A. Bargteil, C. Wojtan, J. Hodgins, and G. Turk. 2007. A Finite Element Method for Animating Large Viscoplastic Flow. *ACM Trans. Graph.* 26, 3 (2007).
- T. Belytschko, W. Liu, B. Moran, and K. Elkhodary. 2013. *Nonlinear finite elements for continua and structures*. John Wiley & Sons.
- S. Blemker, P. Pinsky, and S. Delp. 2005. A 3D model of muscle reveals the causes of nonuniform strains in the biceps brachii. *J. of Biomechanics* 38, 4 (2005), 657–665.
- J. Bonet and R. D. Wood. 2008. *Nonlinear continuum mechanics for finite element analysis*. Cambridge University Press.
- S. Bouaziz, S. Martin, T. Liu, L. Kavan, and M. Pauly. 2014. Projective Dynamics: Fusing Constraint Projections for Fast Simulation. *ACM Trans. Graph.* 33, 4 (2014).
- J. Cai. 2016. *Simulating Deformable Models with Anisotropic Materials*. Ph.D. Dissertation. Nanyang Technological University.
- G. Chagnon, M. Rebouah, and D. Favier. 2015. Hyperelastic energy densities for soft biological tissues: a review. *J. of Elasticity* 120, 2 (2015), 129–160.
- J. Chen, H. Bao, T. Wang, M. Desbrun, and J. Huang. 2018. Numerical Coarsening Using Discontinuous Shape Functions. *ACM Trans. Graph.* 37, 4, Article 120 (July 2018), 12 pages.
- P. Ciarletta, I. Izzo, S. Micera, and F. Tendick. 2011. Stiffening by fiber reinforcement in soft materials: a hyperelastic theory at large strains and its application. *Journal of the Mechanical Behavior of Biomedical Materials* 4, 7 (2011), 1359–1368.
- C. Forest, H. Delingette, and N. Ayache. 2002. Removing tetrahedra from a manifold mesh. In *Proceedings of Computer Animation*. 225–229.
- T. Gast, C. Schroeder, A. Stomakhin, C. Jiang, and J. Teran. 2015. Optimization Integrator for Large Time Steps. *IEEE Transactions on Visualization and Computer Graphics* 21, 10 (2015), 1103–1115.
- G. H. Golub and C. F. Van Loan. 2012. *Matrix computations*. Vol. 3. JHU Press.
- C. Gonzalez-Ochoa, D. Eberle, and R. Dressel. 2002. Dynamic Simulation of Wing Motion on "Reign of Fire". In *ACM SIGGRAPH Sketches*. 174–174.
- G. Holzapfel. 2005. Similarities between soft biological tissues and rubberlike materials. In *Constitutive Models for Rubber IV*, Vol. 4. 607.
- G. Holzapfel, T. Gasser, and R. Ogden. 2000. A new constitutive framework for arterial wall mechanics and a comparative study of material models. *Journal of Elasticity and the Physical Science of Solids* 61, 1-3 (2000), 1–48.
- C. Horgan and G. Saccomandi. 2005. A new constitutive theory for fiber-reinforced incompressible nonlinearly elastic solids. *Journal of the Mechanics and Physics of Solids* 53, 9 (2005), 1985–2015.
- Y. Hu, Q. Zhou, X. Gao, A. Jacobson, D. Zorin, and D. Panozzo. 2018. Tetrahedral Meshing in the Wild. *ACM Trans. Graph.* 37, 4, Article 60 (July 2018), 14 pages.
- H. Iben. 2007. *Generating Surface Crack Patterns*. Ph.D. Dissertation. University of California, Berkeley.

G. Irving, R. Kautzman, G. Cameron, and J. Chong. 2008. Simulating the Develved: Finite Elements on WALL-E. In *ACM SIGGRAPH Talks*. Article 54, 1 pages.

G. Irving, J. Teran, and R. Fedkiw. 2004. Invertible Finite Elements for Robust Simulation of Large Deformation. In *SIGGRAPH/Eurog. Symp. on Comp. Anim.* 131–140.

A. Jacobson, Z. Deng, L. Kavan, and J. P. Lewis. 2014. Skinning: Real-time Shape Deformation. In *ACM SIGGRAPH Courses*. Article 24, 24:1–24:1 pages.

P. M. Knupp. 2003. Algebraic mesh quality metrics for unstructured initial meshes. *Finite Elements in Analysis and Design* 39, 3 (2003), 217–241.

D. Koschier, J. Bender, and N. Thuerey. 2017. Robust eXtended Finite Elements for Complex Cutting of Deformables. *ACM Trans. Graph.* 36, 4, Article 55 (July 2017).

F. Labelle and J. Shewchuk. 2007. Isosurface Stuffing: Fast Tetrahedral Meshes with Good Dihedral Angles. *ACM Trans. Graph.* 26, 3, Article 57 (July 2007).

S. Lee and D. Terzopoulos. 2006. Heads Up!: Biomechanical Modeling and Neuro-muscular Control of the Neck. *ACM Trans. Graph.* 25, 3 (July 2006), 1188–1198.

S. Lee, R. Yu, J. Park, M. Aanjaneya, E. Sifakis, and J. Lee. 2018. Dexterous Manipulation and Control with Volumetric Muscles. *ACM Trans. Graph.* 37, 4, Article 57 (July 2018), 13 pages.

Y. Lee, M. Park, T. Kwon, and J. Lee. 2014. Locomotion Control for Many-muscle Humanoids. *ACM Trans. Graph.* 33, 6, Article 218 (Nov. 2014), 11 pages.

Y. Li and J. Barbič. 2015. Stable anisotropic materials. *IEEE Transactions on Visualization and Computer Graphics* 21, 10 (2015), 1129–1137.

Y. Li, H. Xu, and J. Barbič. 2017. Enriching Triangle Mesh Animations with Physically Based Simulation. *IEEE Transactions on Visualization and Computer Graphics* 23, 10 (2017), 2301–2313.

T. Liu, S. Bouaziz, and L. Kavan. 2017. Quasi-Newton Methods for Real-Time Simulation of Hyperelastic Materials. *ACM Trans. Graph.* 36, 3 (2017).

P.-L. Manteaux, C. Wojtan, R. Narain, S. Redon, F. Faure, and M.-P. Cani. 2017. Adaptive Physically Based Models in Computer Graphics. *Comput. Graph. Forum* 36, 6 (Sept. 2017), 312–337.

A. McAdams, Y. Zhu, A. Selle, M. Empey, R. Tamstorf, J. Teran, and E. Sifakis. 2011. Efficient Elasticity for Character Skinning with Contact and Collisions. *ACM Trans. Graph.* 30, 4 (2011).

A. Milne, M. McLaughlin, R. Tamstorf, A. Stomakhin, N. Burkard, M. Counsell, J. Canal, D. Komorowski, and E. Goldberg. 2016. Flesh, Flab, and Fascia Simulation on Zootopia. In *ACM SIGGRAPH Talks*. Article 34, 2 pages.

N. Molino, Z. Bao, and R. Fedkiw. 2004. A Virtual Node Algorithm for Changing Mesh Topology During Simulation. *ACM Trans. Graph.* 23, 3 (Aug. 2004), 385–392.

N. Molino, R. Bridson, J. Teran, and R. Fedkiw. 2003. A Crystalline, Red Green Strategy for Meshing Highly Deformable Objects with Tetrahedra. In *International Meshing Roundtable*. 103–114.

D. Nolan, A. Gower, M. Destrade, R. Ogden, and J. McGarry. 2014. A robust anisotropic hyperelastic formulation for the modelling of soft tissue. *Journal of the Mechanical Behavior of Biomedical Materials* 39 (2014), 48–60.

G. Picinbono, J. Lombardo, H. Delingette, and N. Ayache. 2000. Anisotropic elasticity and force extrapolation to improve realism of surgery simulation. In *IEEE International Conference on Robotics and Automation*. IEEE, 596–602.

Y. Qiu. 2018. Spectra, a C++ Library For Large Scale Eigenvalue Problems. <https://spectralib.org>.

K. Rupp, P. Tillet, F. Rudolf, J. Weinbub, A. Morhammer, T. Grasser, A. Jüngel, and S. Selberherr. 2016. ViennaCL, Linear Algebra Library for Multi- and Many-Core Architectures. *SIAM Journal on Scientific Computing* 38, 5 (2016), 412–439.

P. Sachdeva, S. Sueda, S. Bradley, M. Fain, and D. K. Pai. 2015. Biomechanical Simulation and Control of Hands and Tendinous Systems. *ACM Trans. Graph.* 34, 4, Article 42 (July 2015).

S. Saito, Z. Zhou, and L. Kavan. 2015. Computational Bodybuilding: Anatomically-based Modeling of Human Bodies. *ACM Trans. Graph.* 34, 4, Article 41 (July 2015).

T. Schneider, Y. Hu, J. Dumas, X. Gao, D. Panozzo, and D. Zorin. 2018. Decoupling Simulation Accuracy from Mesh Quality. *ACM Trans. Graph.* (2018).

J. Shewchuk. 2002. What is a good linear finite element? interpolation, conditioning, anisotropy, and quality measures. *University of California at Berkeley* 73 (2002).

B. Smith, F. de Goes, and T. Kim. 2018. Stable Neo-Hookean Flesh Simulation. *ACM Trans. Graph.* 37, 2 (2018).

B. Smith, F. de Goes, and T. Kim. 2019. Analytic Eigensystems For Isotropic Distortion Energies. *ACM Trans. Graph.* (2019).

O. Sorkine and M. Alexa. 2007. As-rigid-as-possible surface modeling. In *Eurog. Symposium on Geometry processing*, Vol. 4.

A. Stomakhin, R. Howes, C. Schroeder, and J. M. Teran. 2012. Energetically Consistent Invertible Elasticity. In *ACM SIGGRAPH/Eurog. Symp. Comp. Anim.* 25–32.

J. Teran, S. Blemker, V. Hing, and R. Fedkiw. 2003. Finite volume methods for the simulation of skeletal muscle. In *SIGGRAPH/Eurog. Symp. on Comp. Anim.* 68–74.

J. Teran, E. Sifakis, G. Irving, and R. Fedkiw. 2005. Robust Quasistatic Finite Elements and Flesh Simulation. In *ACM SIGGRAPH/Eurog. Symp. on Comp. Anim.* 181–190.

Y. Tong, S. Lombeyda, A. N. Hirani, and M. Desbrun. 2003. Discrete Multiscale Vector Field Decomposition. *ACM Trans. Graph.* 22, 3 (July 2003), 445–452.

B. Wang, Y. Zhao, and J. Barbič. 2017. Botanical Materials Based on Biomechanics. *ACM Trans. Graph.* 36, 4, Article 135 (July 2017), 13 pages.

H. Wang and Y. Yang. 2016. Descent Methods for Elastic Body Simulation on the GPU. *ACM Trans. Graph.* 35, 6 (2016).

J. Weiss, B. Maker, and S. Govindjee. 1996. Finite element implementation of incompressible, transversely isotropic hyperelasticity. *Computer Methods in Applied Mechanics and Engineering* 135, 1-2 (1996), 107–128.

C. Wojtan and G. Turk. 2008. Fast viscoelastic behavior with thin features. *ACM Trans. Graph.* 27, 3 (2008), 47.

A. Wong, D. Eberle, and T. Kim. 2018. Clean Cloth Inputs: Removing Character Self-intersections with Volume Simulation. In *ACM SIGGRAPH Talks*. Article 42.

H. Xu, F. Sin, Y. Zhu, and J. Barbič. 2015. Nonlinear Material Design Using Principal Stretches. *ACM Trans. Graph.* 34, 4 (2015).

A THE I_5 HESSIAN

In 3D, the explicit expression for the Hessian of I_5 is:

$$\frac{\partial^2 I_5}{\partial \mathbf{f}^2} = 2 \begin{bmatrix} a_0 & 0 & 0 & a_1 & 0 & 0 & a_2 & 0 & 0 \\ 0 & a_0 & 0 & 0 & a_1 & 0 & 0 & a_2 & 0 \\ 0 & 0 & a_0 & 0 & 0 & a_1 & 0 & 0 & a_2 \\ a_1 & 0 & 0 & a_3 & 0 & 0 & a_4 & 0 & 0 \\ 0 & a_1 & 0 & 0 & a_3 & 0 & 0 & a_4 & 0 \\ 0 & 0 & a_1 & 0 & 0 & a_3 & 0 & 0 & a_4 \\ a_2 & 0 & 0 & a_4 & 0 & 0 & a_5 & 0 & 0 \\ 0 & a_2 & 0 & 0 & a_4 & 0 & 0 & a_5 & 0 \\ 0 & 0 & a_2 & 0 & 0 & a_4 & 0 & 0 & a_5 \end{bmatrix} = 2\mathbf{H}_5, \quad (49)$$

where

$$\mathbf{A} = \mathbf{a}\mathbf{a}^T = \begin{bmatrix} a_0 & a_1 & a_2 \\ a_1 & a_3 & a_4 \\ a_2 & a_4 & a_5 \end{bmatrix}. \quad (50)$$

In 2D, these become:

$$2 \begin{bmatrix} a_0 & 0 & a_1 & 0 \\ 0 & a_0 & 0 & a_1 \\ a_1 & 0 & a_2 & 0 \\ 0 & a_1 & 0 & a_2 \end{bmatrix} = 2\mathbf{H}_5 \quad \text{and} \quad \mathbf{A} = \begin{bmatrix} a_0 & a_1 \\ a_1 & a_2 \end{bmatrix}. \quad (51)$$

B ALTERNATE 3D EIGENMATRICES

The two final eigenmatrices in §4.2.2 are not unique. In lieu of selecting \mathbf{T}_x to populate the second eigenmatrix, selecting \mathbf{T}_y yields the following eigenmatrices:

$$\mathbf{Q}_1 = \mathbf{U}\mathbf{T}_y\Sigma\mathbf{V}\mathbf{A} \quad (52)$$

$$\mathbf{Q}_2 = (\sigma_z \hat{\mathbf{a}}_z)\mathbf{U}\mathbf{T}_x\Sigma\hat{\mathbf{A}}\mathbf{V}^T - (\sigma_x \hat{\mathbf{a}}_x)\mathbf{U}\mathbf{T}_z\Sigma\hat{\mathbf{A}}\mathbf{V}^T. \quad (53)$$

Selecting \mathbf{T}_z yields:

$$\mathbf{Q}_1 = \mathbf{U}\mathbf{T}_z\Sigma\mathbf{V}^T\mathbf{A} \quad (54)$$

$$\mathbf{Q}_2 = (\sigma_y \hat{\mathbf{a}}_y)\mathbf{U}\mathbf{T}_x\Sigma\mathbf{V}^T\mathbf{A} - (\sigma_x \hat{\mathbf{a}}_x)\mathbf{U}\mathbf{T}_y\Sigma\mathbf{V}^T\mathbf{A}. \quad (55)$$

As with the previous eigenmatrices, these are non-normalized.

C A NAÏVE ANISOTROPIC ENERGY

A naïve anisotropic version of the ARAP energy, $\Psi_{\text{ARAP}} = \mu/2\|\mathbf{F} - \mathbf{R}\|_F^2$ [Alexa et al. 2000], is:

$$\Psi_{\text{bad}} = \mu/2\|(\mathbf{F} - \mathbf{R})\mathbf{a}\|_F^2. \quad (56)$$

We use I_4 and I_5 to rewrite this as:

$$\begin{aligned}\Psi_{\text{bad}} &= \mu/2 \|(\mathbf{F} - \mathbf{R})\mathbf{a}\|_2^2 \\ &= \mu/2 \|\mathbf{F}\mathbf{a}\|_2^2 - \|\mathbf{R}\mathbf{a}\|_2^2 - 2\mathbf{a}^T \mathbf{F}^T \mathbf{R}\mathbf{a} \\ &= \mu/2 \left(\mathbf{a}^T \mathbf{F}^T \mathbf{F}\mathbf{a} - \mathbf{a}^T \mathbf{R}^T \mathbf{R}\mathbf{a} - 2\mathbf{a}^T \mathbf{F}^T \mathbf{R}\mathbf{a} \right) \\ &= \mu/2 (I_5 - 2I_4 - 1)\end{aligned}\quad (57)$$

The PK1 is then:

$$\frac{\partial \Psi_{\text{bad}}}{\partial \mathbf{F}} = \mu \left((\mathbf{F} - \mathbf{R})\mathbf{A} - \frac{\partial \mathbf{R}}{\partial \mathbf{F}} : \mathbf{F}\mathbf{A} \right). \quad (58)$$

The first term, $(\mathbf{F} - \mathbf{R})\mathbf{A}$, is appealingly similar to the PK1 of the isotropic ARAP energy, $(\mathbf{F} - \mathbf{R})$. However, the second term contains the troubling rotation gradient term, and causes the forces to become singular for the reasons described in §5.2.1.

D ANISOTROPIC ENERGY OF BLEMKER ET AL. [2005]

Blemker et al. [2005] presents their muscle activation fiber model as a force-length relation, which can be written in terms of $\sqrt{I_5}$,

$$f_{\text{active}} = \begin{cases} 9 \left(\sqrt{I_5}/l - 2/5 \right)^2 & \sqrt{I_5} \leq 3/5l \\ 9 \left(\sqrt{I_5}/l - 8/5 \right)^2 & \sqrt{I_5} \geq 7/5l \\ -4 \left(-\sqrt{I_5}/l + 1 \right)^2 + 1 & 3/5l < \sqrt{I_5} < 7/5l, \end{cases} \quad (59)$$

where l is the optimal fiber length. These can be integrated to obtain a strain energy:

$$\Psi_{\text{active}} = \begin{cases} 9 \left(-4I_5^{3/2}/2/15l + I_5^2/2l^2 + I_5/2/5^2 \right) & \sqrt{I_5} \leq 3/5l \\ 9 \left(-4I_5^{3/2}/8/15l + I_5^2/2l^2 + I_5/8/5^2 \right) & \sqrt{I_5} \geq 7/5l \\ 16I_5^{3/2}/3l - 2I_5^2/l^2 - 3I_5 & 3/5l < \sqrt{I_5} < 7/5l \end{cases} \quad (60)$$

The first eigenvalue is:

$$\lambda_{0,\text{active}} = \begin{cases} -144\sqrt{I_5}/5l + 54I_5/l^2 + 72/25 & \sqrt{I_5} \leq 3/5l \\ -576\sqrt{I_5}/5l + 54I_5/l^2 + 1152/25 & \sqrt{I_5} \geq 7/5l \\ 32\sqrt{I_5}/l - 24I_5/l^2 - 6 & 3/5l < \sqrt{I_5} < 7/5l \end{cases}. \quad (61)$$

The other two are:

$$\lambda_{\{1,2\},\text{active}} = \begin{cases} -72\sqrt{I_5}/5l + 18I_5/l^2 + 72/25 & \sqrt{I_5} \leq 3/5l \\ -288\sqrt{I_5}/5l + 18I_5/l^2 + 1152/25 & \sqrt{I_5} \geq 7/5l \\ 16\sqrt{I_5}/l - 8I_5/l^2 - 6 & 3/5l < \sqrt{I_5} < 7/5l \end{cases}. \quad (62)$$

The non-linearity of the model is equivalent to Anisotropic StVK, so none of the issues of C-based invariants have been addressed. Blemker et al. [2005] also presents a piecewise-exponential passive fiber model, which only exacerbates the issue of non-linearity.

E DEGENERACY PERMUTATIONS

We have assumed that face 0 on the tetrahedron has maximal area (see unfolded tet in inset, normal of face 2 points out of page).

In order to support degeneracies for other faces, we reorder the tet indices so that the degeneracy coincides with face 0. When computing the final forces and Hessians, we then augment $\frac{\partial \mathbf{f}}{\partial \mathbf{u}} = \text{vec} \left(\frac{\partial \mathbf{F}}{\partial \mathbf{u}} \right)$ with a permutation matrix \mathbf{E} :

$$\mathbf{f} = -v \cdot \mathbf{E} \frac{\partial \mathbf{f}^T}{\partial \mathbf{u}} \text{vec}(\mathbf{P}(\mathbf{F})) \quad (63)$$

$$\frac{\partial \mathbf{f}}{\partial \mathbf{u}} = -v \cdot \mathbf{E} \frac{\partial \mathbf{f}^T}{\partial \mathbf{u}} \text{vec} \left(\frac{\partial \mathbf{P}(\mathbf{F})}{\partial \mathbf{f}} \right) \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \mathbf{E}^T. \quad (64)$$

For each case where face i has the maximal area in the tet, we define a permutation matrix \mathbf{E}_i :

$$\mathbf{E}_1 = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{I} \\ 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 \end{bmatrix}, \mathbf{E}_2 = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & \mathbf{I} \\ 0 & \mathbf{I} & 0 & 0 \end{bmatrix}, \mathbf{E}_3 = \begin{bmatrix} 0 & 0 & \mathbf{I} & 0 \\ \mathbf{I} & 0 & 0 & 0 \\ 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & \mathbf{I} \end{bmatrix}. \quad (65)$$

Above, \mathbf{I} is a 3×3 identity. If face 0 has maximal area, \mathbf{E}_0 is identity.

F REHABILITATION EXPRESSIONS

If a degenerate element is found, then the existing isotropic energy (Ψ_{iso}), forces, and force gradients are replaced with:

$$\Psi = \Psi_{\text{iso}}(\mathbf{F}_{\text{iso}}) + \Psi_{\text{D}}(\mathbf{F}_{\text{aniso}}) \quad (66)$$

$$\mathbf{g} = v \left[\frac{\partial \mathbf{F}_{\text{iso}}^T}{\partial \mathbf{u}} : \mathbf{P}_{\text{iso}}(\mathbf{F}_{\text{iso}}) + \frac{\partial \mathbf{F}_{\text{aniso}}^T}{\partial \mathbf{u}} : \mathbf{P}_{\text{D}}(\mathbf{F}_{\text{aniso}}) \right] \quad (67)$$

$$\frac{\partial \mathbf{g}}{\partial \mathbf{u}} = v \left[\frac{\partial \mathbf{f}_{\text{iso}}^T}{\partial \mathbf{u}} \frac{\partial^2 \Psi_{\text{iso}}}{\partial \mathbf{f}^2} \frac{\partial \mathbf{f}_{\text{iso}}}{\partial \mathbf{u}} + \frac{\partial \mathbf{f}_{\text{aniso}}^T}{\partial \mathbf{u}} \frac{\partial^2 \Psi_{\text{D}}}{\partial \mathbf{f}^2} \frac{\partial \mathbf{f}_{\text{aniso}}}{\partial \mathbf{u}} \right] \quad (68)$$

where

$$\frac{\partial \mathbf{F}_{\text{iso}}}{\partial \mathbf{u}} = \frac{\partial \mathbf{D}_{s,\text{iso}}}{\partial \mathbf{u}} \cdot \mathbf{D}_{m,\text{iso}}^{-1} \quad \text{and} \quad \frac{\partial \mathbf{F}_{\text{aniso}}}{\partial \mathbf{u}} = \frac{\partial \mathbf{D}_s}{\partial \mathbf{u}} \cdot \mathbf{D}_{m,\text{iso}}^{-1}, \quad (69)$$

$$\text{and } \frac{\partial \mathbf{f}_{\text{iso}}}{\partial \mathbf{u}} = \text{vec} \left(\frac{\partial \mathbf{F}_{\text{iso}}}{\partial \mathbf{u}} \right), \quad \frac{\partial \mathbf{f}_{\text{aniso}}}{\partial \mathbf{u}} = \text{vec} \left(\frac{\partial \mathbf{F}_{\text{aniso}}}{\partial \mathbf{u}} \right).$$

