Kirchhoff-Love Shells with Arbitrary Hyperelastic Materials

JIAHAO WEN, University of Southern California, USA
JERNEJ BARBIČ, University of Southern California, USA

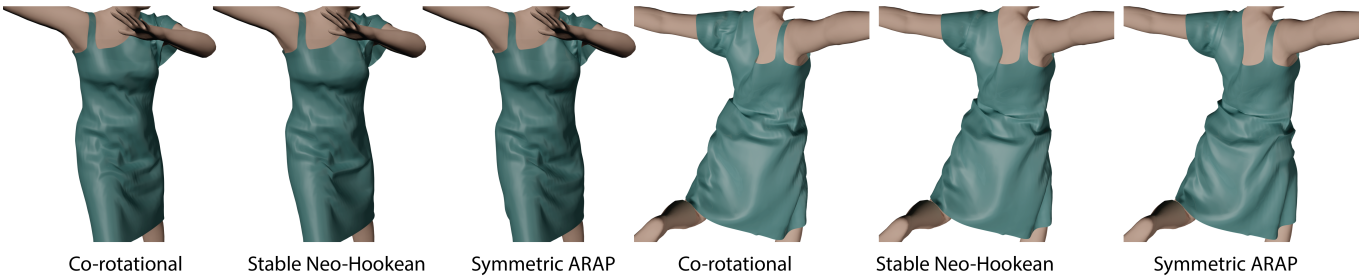

Fig. 1. **Our technique can simulate Kirchhoff-Love thin shells with arbitrary hyperelastic materials:** We show a thin-shell cloth draped against a human body in two poses, for three nonlinear materials. The Symmetric ARAP and Co-rotational materials include odd powers of principal stretches in their definitions, and cannot be simulated using prior work. Observe the different folds, and silhouette changes, under the three different materials.

Kirchhoff-Love shells are commonly used in many branches of engineering, including in computer graphics, but have so far been simulated only under limited nonlinear material options. We derive the Kirchhoff-Love thin-shell mechanical energy for an arbitrary 3D volumetric hyperelastic material, including isotropic materials, anisotropic materials, and materials whereby the energy includes both even and odd powers of the principal stretches. We do this by starting with any 3D hyperelastic material, and then analytically computing the corresponding thin-shell energy limit. This explicitly identifies and separates in-plane stretching and bending terms, and avoids numerical quadrature. Thus, in-plane stretching and bending are shown to originate from one and the same process (volumetric elasticity of thin objects), as opposed to from two separate processes as done traditionally in cloth simulation. Because we can simulate materials that include both even and odd powers of stretches, we can accommodate standard mesh distortion energies previously employed for 3D solid simulations, such as Symmetric ARAP and Co-rotational materials. We relate the terms of our energy to those of prior work on Kirchhoff-Love thin-shells in computer graphics that assumed small in-plane stretches, and demonstrate the visual difference due to the presence of our exact stretching and bending terms. Furthermore, our formulation allows us to categorize all distinct hyperelastic Kirchhoff-Love thin-shell energies. Specifically, we prove that for Kirchhoff-Love thin-shells, the space of all hyperelastic materials collapses to two-dimensional hyperelastic materials. This observation enables us to create an interface for the design of thin-shell Kirchhoff-Love mechanical energies, which in turn enables us to create thin-shell materials that exhibit arbitrary stiffness profiles under large deformations.

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

Additional Key Words and Phrases: thin shells, FEM, Kirchhoff-Love, nonlinear materials, hyperelastic, differential geometry

Authors' addresses: Jiahao Wen, University of Southern California, Los Angeles, USA, jiahaow@usc.edu; Jernej Barbič, University of Southern California, Los Angeles, USA, jnb@usc.edu.

0730-0301/2023/12-ART187
https://doi.org/10.1145/3618405

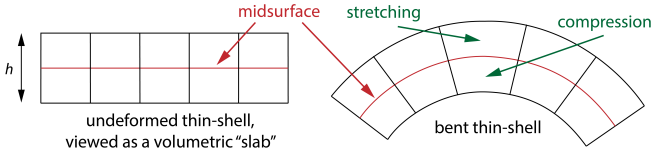

Fig. 2. **Bending forces originate from 3D elasticity:** Thin-shell bending forces occur because the volumetric 3D material stretches on one side of the mid-surface, and compresses on the other. Thin-shell stretching and bending forces are both a natural consequence of the same phenomenon (volumetric 3D elasticity), and not the result of some separate processes.

ACM Reference Format:
Jiahao Wen and Jernej Barbič. 2023. Kirchhoff-Love Shells with Arbitrary Hyperelastic Materials. *ACM Trans. Graph.* 42, 6, Article 187 (December 2023), 15 pages. https://doi.org/10.1145/3618405

1 INTRODUCTION

Kirchhoff-Love ("KL") thin shells are thin 3D objects simulated under the assumption that normal vectors remain normal to the midsurface (no out-of-plane shearing), and remain unit length. Although not universally true, this assumption is commonly made both in engineering and computer graphics, and is a reasonable assumption for many real-world thin shells. Such shells can be readily combined with the Finite Element Method or other discretization techniques. They exhibit good independence on mesh quality and edge orientation (Figure 3), and cleanly produce the shell stretching and bending energies all from one mechanical model without any special treatment of bending vs stretching. They also hold the promise of supporting general nonlinear constitutive models. Although KL shells have been simulated under complex nonlinear materials models, a key unaddressed question that we address is the following. Starting from an arbitrary 3D solid volumetric material energy density $\psi = \psi(F)$, where $F \in \mathbb{R}^{3\times3}$ is a 3D deformation gradient in an elastic solid, what is the corresponding KL thin-shell mechanical energy, obtained in the $h \to 0$ limit when a volumetric "slab" is made arbitrarily thin? When subjecting arbitrary 3D

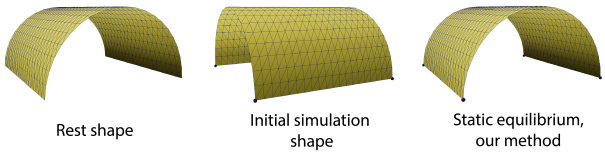

Fig. 3. **Our method has good independence on mesh edge orientation.** The rest shape of the thin shell is a half-cylinder. We then flatten it to a plane and bend it into a half-cylinder in a different (perpendicular) direction ("initial simulation shape"). Note that the rest shape has thus been transformed nonlinearly; the initial simulation shape is *not* a 90° rotation of the rest shape! We then run a quasi-static simulation, keeping the four shown corner vertices fixed. As is physically correct, our method restores back to the rest shape. Thickness is $h = 1cm$ and the size of a side of the cylinder is 200× larger ($2m$). Same camera angle for all pictures. This experiment was inspired by a figure and discussion in [Chen et al. 2018] (related to, but not identical, to the "Scordelis-Lo roof" benchmark [Macneal and Harder 1985]). Chen et al. expressed this example in a theoretical, continuous setting; we hereby run it practically on an actual triangle mesh.

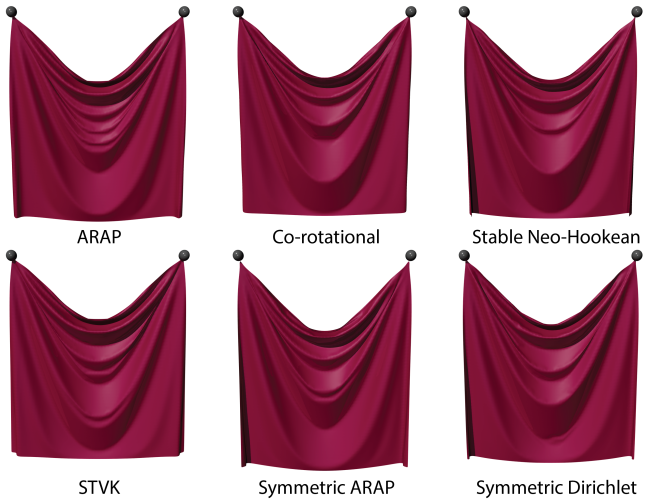

Fig. 4. **Two-point draping under several general isotropic hyperelastic materials.** ARAP, Co-rotational and Symmetric ARAP need odd powers of principal stretches and therefore require our method for simulation. We can observe that the ARAP material produces higher frequency wrinkles than St. Venant Kirchhoff [Kim 2020]. Note that we demonstrate anisotropic hyperelastic materials in Figure 16.

volumetric energy densities ψ to the thin-shell limit process, just what is the space of all distinct energies ψ, i.e., equivalence classes of ψ that produce the same thin-shell energy? Previous work has partially answered these questions, and in particular for energy densities of the form $\psi = \psi(E)$, where $E = (F^TF - I)/2$ ("E-based materials"). Because of the F^TF term, these energies only contain even powers of the singular values ("principal stretches") of F. This, however, excludes many important hyperelastic energies, and in particular those that include odd powers of the principal stretches, such as popular mesh distortion energies (Symmetric ARAP and Co-rotational materials in Figure 1). We address this case, namely we support any hyperelastic volumetric energy $\psi = \psi(F)$ ("F-based materials"). Starting from $\psi = \psi(F)$ and a thickness h, we derive an expansion of the thin-shell energy as a polynomial in h that is exact up to $O(h^5)$. For 3D volumetric solid simulation, "F-based" materials are extremely common [Kim and Eberle 2020; Sifakis and Barbič 2012; Smith et al. 2019; Teran et al. 2005] and used in many real-world applications. They support numerous simulation stabilization improvements, such as inversion handling, automatic energy extensions to inversion regime, and projection of their Hessians to SPD. Because our method is obtained as a thin-shell limit of volumetric simulation, we benefit from these improvements automatically, e.g., we also project our Hessians to SPD and use volumetric energies that guard against near-invertibility (Figure 14 in Section 6).

Simulation techniques for thin-shells to date have been largely separated from the above volumetric simulation developments. However, a thin-shell is at the end of the day simply a very thin 3D volumetric object. Many cloth simulation papers treat bending as somehow separate from stretching and shearing, but there is no special "magic" to bending: bending resistance is simply a consequence of the thin, but still volumetric thin-shell stretching or compressing in regions above and below the midsurface (Figure 2), i.e., the bending energy exists because of a 3D volumetric effect. It is therefore natural to ask if there is a way to derive a thin-shell energy as the limit of the volumetric energy, do so for an arbitrary 3D hyperelastic material, and present all numerical and simulation derivations and other components needed to successfully run FEM simulations of such thin shells. We contribute such a method.

Our work can be seen as the counterpart of volumetric "F-based" material modeling, applied to thin-shells. We achieve our results via an asymptotic expansion of a matrix inverse, and a careful application of Taylor series expansion and numerical formulas for differentiation. Although these numerical components are well-known in applied mathematics and computer graphics in general, our novelty lies in applying these tools in such a manner to ensure that the terms remain exact up to $O(h^5)$, and in a manner that removes the derivatives of ψ from the energy formulation (i.e., our energy depends purely on ψ, not $d\psi/dF$ or higher derivatives). As a critical difference to prior work [Clyde et al. 2017], this in turn enables us to avoid numerical quadrature along the thickness, and to cleanly identify all thin-shell terms up to $O(h^5)$, for an arbitrary $\psi = \psi(F)$. We combine our method with FEM, and demonstrate that we can simulate arbitrary hyperelastic materials, alongside with visual effects of the different thin-shell materiality. We visually demonstrate our results on computer simulations of thin-shells with various nonlinear material properties, and via shape deformation of thin-shells under popular distortion energies. Furthermore, we produce the first visualizations of thin-shells resulting from several standard 3D nonlinear elastic materials, such as "Co-rotational linear cloth", "Symmetric ARAP cloth" and "Anisotropic ARAP cloth".

We also ask and resolve the following question: Just what is the space of all distinct KL thin-shell energies, i.e., what are the equivalence classes of general hyperelastic energy functions ψ that result in the same thin-shell energy? We prove that for KL thin-shells, the space of all meaningful arbitrary 3D hyperelastic energies essentially collapses to 2D functions $\psi_{2D}(F_{2D})$, where F_{2D} is a 2×2

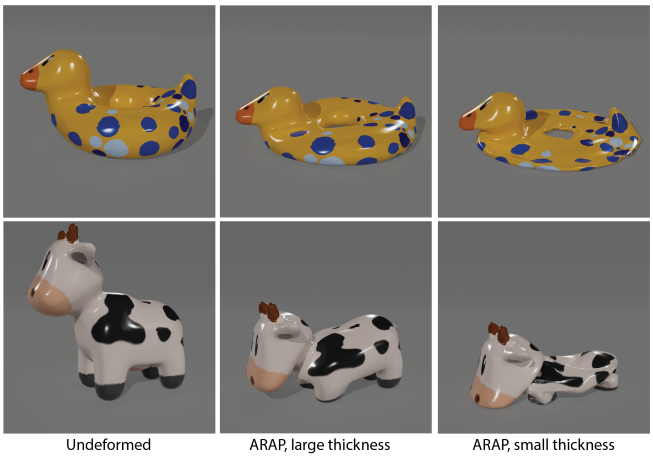

Fig. 5. **Thin-shell simulations under different thicknesses.** Because our methods explicitly identifies terms linear and cubic in the thin shell thickness h, we can adjust the thin-shell bendability by changing h.

deformation gradient. We use this observation to propose an artist-friendly approach for tweaking thin-shell material properties. Our approach permits the artist to precisely control both the small-strain material behavior and the rate at which the thin-shell nonlinearly stiffens under large deformations.

2 RELATED WORK

Mechanics of thin shells. Thin-shell structures are abundant in both computer graphics and mechanical engineering, e.g., clothing, ribbons, architectural elements such as curtains, biological membranes, carbon-fiber-reinforced polymers, sails, parachutes, etc. Based on differential geometry, there are two popular thin-shell kinematic models: the shear-rigid Kirchhoff-Love (KL) model, and the shear-deformable Cosserat model [Weischedel 2012]. For KL shells, the normals to the undeformed surface rotate into normals of the deformed surface without any change in length [Koiter 1970]. Cosserat shells introduce additional degrees of freedom such that the deformed normals can deviate from the true surface normals, thereby permitting transversal shearing. Because the shear-rigid KL model is "easier" and can handle many cases in computer graphics, we pursue the KL shell model in this paper. A thin shell is fundamentally a thin volumetric object, defined by its mid-surface and a small (relative to object size and to curvature radius) thickness h. Following the KL assumption, it is intuitive to reduce the volumetric shell behavior to a two-dimensional manifold embedded in 3D space, and tools from differential and Riemannian geometry can be applied to this setting [Simo and Fox 1989]. In engineering, [Cirak et al. 2000] and [Cirak and Ortiz 2001] proposed to apply subdivision finite elements to the FEM analysis of KL thin shells. In computer graphics, [Thomaszewski et al. 2006] first introduced these ideas for dynamic cloth simulation, but used linear strain-stress relationships. Clyde et al. [Clyde et al. 2017] extended this family of methods to nonlinear E-based orthotropic constitutive models, but did not develop a polynomial thin-shell energy expansion in h like we do, and had to use quadrature to calculate the integration in the thickness direction. Previous methods have been limited either to a specific material such as the St. Venant Kirchhoff material [Chen et al. 2018] or E-based materials [Clyde et al. 2017].

Physically based simulation of thin shells. Simulation of thin shells, and especially cloth, has always been a very popular topic in the computer graphics community, and includes several seminal methods [Baraff and Witkin 1998; Bridson et al. 2002; Grinspun et al. 2003]. To model the complex bending behavior of thin shells, Grinspun et al. [Grinspun et al. 2003] introduced a bending energy formulation based on the difference of squared mean curvatures. While simple to implement, it is highly dependent on the mesh discretization and converges slowly to the ground truth [Garg et al. 2007]. In order to capture the undeformed curvature completely, one requires full information on the extrinsic deformation of the thin shell [Grinspun et al. 2006], and therefore using only the difference of mean curvature is not sufficient. A triangle-averaged shape operator was proposed by [Gingold et al. 2004] as an alternative for the bending energy, which was later used in the well-known cloth simulation library ArcSim [Narain et al. 2013, 2012]. However, as the experiments in [Chen et al. 2018] show, the convergence to the analytical ground truth is also slow for this type of energy. Good implementation-ready formulas for hinge-based energies, and their gradients and Hessians for bending are provided in [Tamstorf and Grinspun 2013]. While FEM is popular for thin-shell simulation and has been employed in several publications [Kim 2020; Li et al. 2018; Volino et al. 2009; Wang et al. 2011], these methods did not pursue general nonlinear hyperelastic materials like we do. Thomaszewski et al. [Thomaszewski et al. 2006] adopted the KL assumption, and proposed that the in-plane-stretching and bending energies should be treated in a unified and consistent way through numerical integration in the thickness direction. The same approach was also used in [Clyde et al. 2017], who used KL thin shells for parameter estimation of woven cloth. Because they both used numerical quadrature to evaluate the thin-shell elastic energy integral in the thickness direction, their formulation fuses stretching and bending together, and is not explicitly separated and expanded in terms of h like in our work. Weischedel [Weischedel 2012] derived the continuous and discrete elastic energies for both KL and Cosserat shells. However, their energy is not independent of the undeformed thin shell parameterization, and thus has strict requirements on the shell's undeformed geometry. Chen et al. [Chen et al. 2018] made the method independent of the undeformed thin shell parameterization, and used it to simulate thin shell plasticity; their work is limited by the small in-plane strain assumption, and was only applied to the St. Venant Kirchhoff material. Our method is independent of the undeformed thin shell parameterization, supports arbitrarily-shaped undeformed thin-shells (including all mechanical terms originating from such undeformed curvature, exact up to $O(h^5)$), supports arbitrary hyperelastic materials and arbitrarily large deformations (no small in-plane stretch assumption). Furthermore, in Section 4.3, we identify *which* terms of our formulation correspond to which terms in [Chen et al. 2018], and as such we demonstrate exactly how our method generalizes [Chen et al. 2018].

Recently, based on their 2018 shell model, Chen et al. [Chen et al. 2021] proposed a method to use tension field theory to add fine wrinkle detail to coarse meshes. Guo et al. [Guo et al. 2018] proposed simulating frictional contact of thin shells using MPM. For cloth, there is also a large family of work on yarn-level cloth simulation: full garments were simulated at the yarn level [Kaldor et al. 2008], woven cloth was modeled using a reduced model of sliding yarns in persistent contact [Cirio et al. 2014], and this was extended to knitted cloth [Cirio et al. 2016]. Robust multi-layer fabrics was also investigated [Sánchez-Banderas et al. 2020]. While highly impressive, yarn-level cloth simulation has many more degrees of freedom than our method and is specific to cloth. We do not aim to compete with this family of methods and focus on thin-shell fundamentals.

Distortion energies and material optimization. Optimization of distortion energies has very broad application in computer graphics, especially in physics-based simulation and geometry processing. Smith et al. [Smith et al. 2018] proposed the stable Neo-Hookean energy, which can be easily plugged into our simulation model. In [Smith et al. 2019], the authors proposed S-invariants (also called "lower invariants"), and show how to use them to simulate volumetric materials. We use the same invariants, but use them for thin shells. Additionally, for our material design, we use the volumetric stretch-based materials [Xu et al. 2015], again adapted to thin-shells. Kim et al. [Kim et al. 2019] derived a new inversion-aware anisotropic strain invariant and formulated a robust, inversion-safe anisotropic model called "Anisotropic ARAP". We can easily plug their energy into our simulation model and produce anisotropic thin shells (Figure 16). Based on [Baraff and Witkin 1998], Kim [Kim 2020] proposed an anisotropic FEM energy matching the Baraff-Witkin cloth model, and provided a complete eigenanalysis for it. Their in-plane-stretch component is matched with FEM, whereas bending uses a hinge-based model [Grinspun et al. 2003]. By using the Valanis-Landel hypothesis [Valanis and Landel 1967], Xu et al. [Xu et al. 2015] proposed to model principal-stretch-based elastic energies, and focused on how to make the design of such materials intuitive for artists. Thanks to our classification theorem of all KL thin-shell energies (Section 4.1), we were able to develop a similar interface for material design of KL thin shells (Section 4.2). Another research topic complementary to material design is physical parameter estimation. Wang et al. [Wang et al. 2011] proposed to estimate planar and bending stiffness by two separate real-world deformation tests. An automated device was developed to acquire fabric stiffness parameters, and capture internal friction hysteresis effects [Miguel et al. 2012, 2013]. Clyde et al. [Clyde et al. 2017] proposed to use the KL shell model to investigate material parameters for woven fabrics. Their method is limited to E-based materials, which are subset of our method, and, unlike us, relies on numerical quadrature. Recently, Feng et al. [Feng et al. 2022] developed a real-world draping tester and used a neural network to infer a quadratic mean-curvature-based [Bergou et al. 2006] bending parameters of a real fabric.

3 BACKGROUND: F-BASED AND E-BASED MATERIALS

In mechanics and computer graphics, there are two popular approaches to define the 3D solid elastic energy density function ψ.

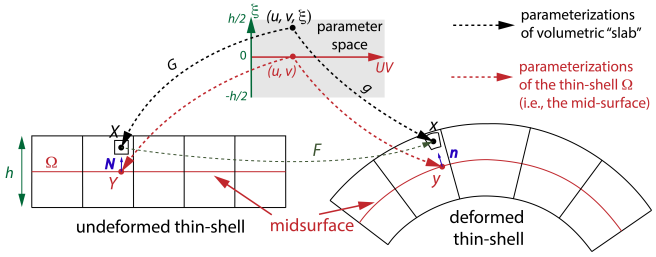

Fig. 6. **Thin-shell and volumetric "slab" parameterizations, parameterization gradients G and g, and the deformation gradient F.**

One approach is to start with the deformation gradient $F \in \mathbb{R}^{3\times3}$, form the Green-Lagrangian tensor $E = 1/2(F^TF - I)$, and then define ψ as a function of E ("E-based materials"). The other approach is to define ψ directly as a function of F ("F-based materials"). Within F-based materials, there are two substrategies: either define ψ in terms of the singular values $\lambda_1, \lambda_2, \lambda_3$ (the "principal stretches") of F [Xu et al. 2015], or in terms of the "lower invariants" $i_1 = \lambda_1 + \lambda_2 + \lambda_3$, $i_2 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2$, $i_3 = \lambda_1\lambda_2\lambda_3$ [Smith et al. 2019]. In F-based materials, the Taylor expansion of ψ in terms of λ_i can contain all powers of λ_i. In contrast, E-based materials are only able to contain even powers of λ_i, due to the F^TF term. While E-based materials could in theory express all materials, this is practically known to be difficult [Kim and Eberle 2020]. Namely, one cannot easily produce odd powers, as doing so requires computing the matrix square root of $2E+I$; but this in turn introduces singularities when eigenvalues are close, especially when ψ needs to be differentiated for elastic forces and tangent stiffness matrices. As stated in [Kim and Eberle 2020], "*ARAP cannot be written in terms of the Cauchy-Green invariants*" ($\psi = \sum_i(\lambda_i - 1)^2$ has odd terms $2\lambda_i$), and "*You cannot use a sum of squared values to express a sum of unsquared values.*" Because of these reasons, F-based materials are popular in VFX practice. In our work, we simulate F-based materials. Our examples include several F-based materials that are not E-based materials, namely ARAP, Symmetric ARAP and Co-rotational materials. The formulas for elastic energies for the F-based materials used in our work can be found in references [Smith et al. 2019] and [Kim and Eberle 2020]. In particular, for ARAP, Symmetric ARAP, Symmetric Dirichlet, Co-rotational, St. Venant Kirchoff and Stable Neo-Hookean materials, we use the lower-invariant-based formulation of these materials given in the above references. The one exception is Section 4.2 where we model ψ using a principal stretch-based approach, namely Valanis-Landel f and g functions [Xu et al. 2015]. Our thin-shell formulation is, however, agnostic with respect to the choice of $\psi = \psi(F)$.

4 THIN-SHELL ELASTIC ENERGY

We start with an arbitrary hyperelastic 3D elastic energy density function $\psi : \mathbb{R}^{3\times3} \to \mathbb{R}$ for a 3D elastic solid, i.e., $\psi(F)$ computes the energy density stored in an infinitesimal volume when that volume underwent a linear transformation (a deformation gradient) $F \in \mathbb{R}^{3\times3}$. Note that we do not necessarily assume that ψ is isotropic, and that ψ has nothing to do with thin-shells; it is any hyperelastic *volumetric* elastic energy density. A thin-shell of thickness $h > 0$ can be seen as a 3D elastic solid object, and therefore ψ can be applied to

it and integrated over the object to compute the total elastic energy. We derive an expression for the elastic energy of the thin-shell as a function of h, in the limit when h is made very small. We assume the familiar Kirchhoff-Love ("KL") thin-shell assumption: normals to the mid-surface remain normal after the deformation, and they do not change length. This means that the normal direction to the midsurface does not undergo any shear or stretching. As commonly done for KL shells, we assume that the radius of curvature and length dimensions of the shell are much larger than h. We can parameterize the undeformed thin-shell volume as

$$X(u,v,\xi) = Y(u,v) + \xi N(u,v), \tag{1}$$

where Y is the undeformed mid-surface, N is the undeformed normal, $(u,v) \in UV$ are the parameters of the mid-surface, and $\xi \in [-h/2, h/2]$ specifies the point location in the normal direction (see Figure 6). For the deformed thin shell, we similarly have

$$x(u,v,\xi) = y(u,v) + \xi n(u,v), \tag{2}$$

where y is the deformed mid-surface and n is the deformed normal. The parameterization gradients are 3×3 matrices

$$G(u,v,\xi) = \frac{\partial X}{\partial (u,v,\xi)} = \begin{bmatrix} Y_u + \xi N_u & Y_v + \xi N_v & N \end{bmatrix}, \tag{3}$$

$$g(u,v,\xi) = \frac{\partial x}{\partial (u,v,\xi)} = \begin{bmatrix} y_u + \xi n_u & y_v + \xi n_v & n \end{bmatrix}, \tag{4}$$

where the subscript represents the derivative with respect to the corresponding variable. The solid 3D deformation gradient is

$$F(u,v,\xi) = g(u,v,\xi)\, G^{-1}(u,v,\xi) = \tag{5}$$

$$= (t + \xi q)(T + \xi Q)^{-1} \in \mathbb{R}^{3\times 3}, \qquad \text{where} \tag{6}$$

$$t = [y_u, y_v, n] \in \mathbb{R}^{3\times 3}, \quad q = [n_u, n_v, 0] \in \mathbb{R}^{3\times 3}, \tag{7}$$

$$T = [Y_u, Y_v, N] \in \mathbb{R}^{3\times 3}, \quad Q = [N_u, N_v, 0] \in \mathbb{R}^{3\times 3}. \tag{8}$$

For small values of ξ, the Neumann series gives us

$$(T + \xi Q)^{-1} = T^{-1} - \xi T^{-1} Q T^{-1} + \xi^2 (T^{-1}Q)^2 T^{-1} + O(\xi^3). \tag{9}$$

Therefore, the deformation gradient anywhere inside the "slab" can be written as

$$F(u,v,\xi) = (t + \xi q)\big(T^{-1} - \xi T^{-1} Q T^{-1} + \xi^2 (T^{-1}Q)^2 T^{-1} + O(\xi^3)\big) = \tag{10}$$

$$= F_0 + \xi F_1 + \xi^2 F_2 + O(\xi^3), \qquad \text{where} \tag{11}$$

$$F_0 = t T^{-1} \in \mathbb{R}^{3\times 3}, \tag{12}$$

$$F_1 = q T^{-1} - t T^{-1} Q T^{-1} = t \begin{bmatrix} \bar{\ell} - \ell & 0 \\ 0 & 0 \end{bmatrix} T^{-1} \in \mathbb{R}^{3\times 3}, \tag{13}$$

$$F_2 = t (T^{-1}Q)^2 T^{-1} - q T^{-1} Q T^{-1} = t \begin{bmatrix} (\bar{\ell} - \ell)\bar{\ell} & 0 \\ 0 & 0 \end{bmatrix} T^{-1} \in \mathbb{R}^{3\times 3}. \tag{14}$$

Observe the elegant form above that we discovered for F_1 and F_2 that includes $\bar{\ell} \in \mathbb{R}^{2\times 2}$ and $\ell \in \mathbb{R}^{2\times 2}$, namely the shape operators [Petersen 2016] of the undeformed and deformed surface, respectively. These operators are defined as $\ell = a^{-1} b$, where $a \in \mathbb{R}^{2\times 2}$ and $b \in \mathbb{R}^{2\times 2}$ are the first and second fundamental form, respectively [Petersen 2016]. The undeformed shape operator is $\bar{\ell} = \bar{a}^{-1}\bar{b}$ for the undeformed fundamental forms $\bar{a}$ and $\bar{b}$. Remember that we have, for any location inside the volumetric slab [Petersen 2016]

$$\begin{bmatrix} a & 0 \\ 0 & 1 \end{bmatrix} = g^T g, \qquad \text{and} \qquad \begin{bmatrix} b & 0 \\ 0 & 0 \end{bmatrix} = -g^T q, \tag{15}$$

and analogously for $\bar{a}$ and $\bar{b}$. Note that for the mid-surface, we have $g = t$ and $G = T$. Intuitively, the term F_0 measures in-plane stretching, and the terms F_1 and F_2 measure bending, by measuring the change in the shape operator. As usual in volumetric FEM simulation, the first and second derivative of ψ with respect to F are the first-Piola stress tensor $P = d\psi/dF$ and the "tangent stiffness tensor" dP/dF, respectively [Teran et al. 2005]. Taylor expansion in ξ now gives us (see Appendix A)

$$\psi(F) = \psi(F_0) + \xi\Big(P(F_0) : F_1\Big) +$$

$$+\xi^2\Big(P(F_0) : F_2 + \frac{1}{2}\big(\big(\frac{dP}{dF}_{|F=F_0} : F_1\big) : F_1\big)\Big) + O(\xi^3). \tag{16}$$

We can now compute the elastic energy of the entire "slab",

$$E = \int\limits_{\Omega\times[-h/2,h/2]} \psi(F) dV = \int\limits_{-h/2}^{h/2} d\xi \int\limits_{UV} \psi(F)|G| du dv, \tag{17}$$

where h is the thin shell thickness. In Appendix B, we show that

$$|G| = (1 - 2\xi H + \xi^2 K)\sqrt{\det(\bar{a})}, \tag{18}$$

where H and K are the mean and Gauss curvatures of the undeformed thin shell. Because $\sqrt{\det(\bar{a})}\, du dv = dS$, the integration can be rewritten as

$$E = h \int\limits_{\Omega} \psi(F_0) dS + \frac{h^3}{12} \int\limits_{\Omega} \Big(P(F_0) : \big(F_2 - 2HF_1\big) +$$

$$+ \frac{1}{2}\big(\big(\frac{dP}{dF}_{|F=F_0} : F_1\big) : F_1\big) + K\psi(F_0)\Big) dS + O(h^5). \tag{19}$$

The above formula expresses the elastic energy of the entire "slab" of thickness h. Observe that the term with h, plus the $\frac{h^3}{12} K\psi(F_0)$ term give the in-plane stretching and shearing energy, whereas the remaining h^3 terms give the bending energy. As expected, the smaller the h, the weaker the bending energy, i.e., thinner thin-shells are more bendable. As a side note, observe that the ratio of bending vs in-plane stretching is *not* exactly h^2, due to the $\frac{h^3}{12} K\psi(F_0)$ term; but because all in-plane stretching and bending terms are explicitly identified, our formulation permits one to arbitrarily scale the bending vs in-plane stretching behavior, including in a non-physical manner if so desired. Equation 19 is in principle sufficient for thin-shell simulation, as it makes it possible to evaluate the elastic energy if one knows the undeformed midsurface, the deformed midsurface, the 3D solid elastic energy function $\psi(F)$, and its derivative tensors P and dP/dF. However, in practice for most materials, the formula is difficult to apply, as the calculation of internal forces and tangent stiffness matrices necessitates differentiating E twice with respect to deformed surface positions, and this would in turn require the third and fourth derivatives of ψ with respect to F. For most materials, it is simply not practically possible to compute such high-order derivatives, and this fact alone would preclude their nonlinear simulation. However, we observe that we can avoid the derivatives as

follows, by exploiting the fact that the energy is only determined up to $O(h^5)$ anyway. We eliminate dP/dF by observing that

$$h\psi(F_0) + \frac{h^3}{24}\Big(\big(\frac{dP}{dF}_{|F=F_0} : F_1\big) : F_1\Big) = \tag{20}$$

$$= \frac{h}{2}\Big(\psi\big(F_0 + \frac{\sqrt{3}}{6}hF_1\big) + \psi\big(F_0 - \frac{\sqrt{3}}{6}hF_1\big)\Big) + O(h^5). \tag{21}$$

To eliminate P, we observe that

$$\frac{h^3}{12}P(F_0) : \big(F_2 - 2HF_1\big) = \tag{22}$$

$$= h\Big(\psi\big(F_0 + \frac{h^2}{24}(F_2 - 2HF_1)\big) - \psi\big(F_0 - \frac{h^2}{24}(F_2 - 2HF_1)\big)\Big) + O(h^7). \tag{23}$$

Therefore, the thin-shell elastic energy becomes (still accurate up to $O(h^5)$)

$$E = h\int_\Omega \Big(\frac{h^2K}{12}\psi(F_0) + \frac{1}{2}\big(\psi(F_0 + \frac{\sqrt{3}}{6}hF_1) + \psi(F_0 - \frac{\sqrt{3}}{6}hF_1)\big) +$$

$$+\psi(F_0 + \frac{h^2}{24}(F_2 - 2HF_1)) - \psi(F_0 - \frac{h^2}{24}(F_2 - 2HF_1))\Big)dS + O(h^5). \tag{24}$$

Observe that the above formula only requires knowing F_0, F_1, F_2; and the mean and Gauss curvatures in the undeformed configuration; and evaluating ψ on linear functions of F_0, F_1, F_2. This means that the internal force and tangent stiffness matrix will only require ψ, P and dP/dF; these quantities are standard and readily computable in 3D solid mechanics for a variety of material models. Furthermore, because Equation 24 reveals that ψ is only ever evaluated on arguments that are linear functions of F_0, F_1, F_2 (and the energy formula is still exact up to $O(h^5)$), this enables us to prove a categorization of all distinct Kirchoff-Love hyperelastic thin-shell energies (Section 4.1).

4.1 Categorization of Kirchoff-Love Thin-Shell Energies

In this subsection, we ask the following important question: For Kirchhoff-Love thin-shells, which hyperelastic *volumetric* energy densities ψ result in identical thin-shell energies (up to $O(h^5)$)? Just what exactly is the space of all distinct Kirchhoff-Love thin-shell energies? We answer these questions by proving the following result.

Theorem 1: Let $V \in \mathbb{R}^{3\times3}$ be any rotation matrix whereby the first two columns give two orthogonal tangent vectors to the undeformed thin shell at a given point, and the third column is the normal vector. Then two arbitrary volumetric hyperelastic energies ψ_1 and ψ_2 result in identical thin-shell energies (up to $O(h^5)$) if and only if they agree on each matrix F of the form

$$F = \begin{bmatrix} * & * & 0 \\ * & * & 0 \\ 0 & 0 & 1 \end{bmatrix} V^T, \tag{25}$$

whereby the $*$ entries are arbitrary. Proof is in Appendix C. Observe that pre-multiplying the above F with any 3×3 rotation matrix R doesn't change the above statement, because all hyperelastic energies are invariant to post-rotations, $\psi(RF) = \psi(F)$. This result means that the space of all KL thin-shell energies actually collapses to arbitrary hyperelastic energies of 2×2 matrices! And, only the in-plane stretching behavior matters for the thin-shell energy. This may seem counter-intuitive as where does the bending come from then? The bending comes from the in-plane stretching of the layers of material above and below the midsurface (see Figure 2), and this is manifested by feeding terms of the form $F_0 + \alpha F_1 + \beta F_2$ (for some proper $\alpha, \beta \in \mathbb{R}$) into ψ. If the material is isotropic, we don't need V, and the space collapses further. We are now in the familiar isotropic material situation [Teran et al. 2005], except in 2D. In other words, the space of all isotropic KL thin-shells is identical to the space of all isotropic functions on 2×2 matrices, and there is a wealth of volumetric simulation literature for this situation.

4.2 Material Design

The categorization of all KL hyperelastic materials (Section 4.1) enables us to perform nonlinear material design. We demonstrate this for isotropic materials in Figure 7; but anisotropic design would also be possible. The space of all isotropic functions of 2×2 matrices is in 1:1 correspondence with the space of all symmetric functions of two principal stretches λ_1, λ_2 [Teran et al. 2005]. This space is infinite dimensional, but there is a convenient and expressive subfamily (Valanis-Landel materials [Xu et al. 2015]), here applied in 2D:

$$\psi(\lambda_1, \lambda_2) = f(\lambda_1) + f(\lambda_2) + g(\lambda_1\lambda_2). \tag{26}$$

The 1D functions f and g give the energy due to "stretching" and "surface area change", respectively. Whereas previous work [Xu et al. 2015] operated in 3D and had to model three functions f, g, h using splines, we propose the following simplified 3-parameter family:

$$f(x) = \frac{1}{12}p(x-1)^4 + \frac{1}{2}kx^2 - kx + \frac{k}{2}, \tag{27}$$

$$g(x) = \frac{1}{2}C(x-1)^2, \quad \text{where } p, k, C \text{ are scalar parameters.} \tag{28}$$

The above formulas were chosen because they are the simplest formulas that satisfy the following requirements. First, we impose $f'(1) = g'(1) = 0$, i.e., material is in the rest state when $\lambda_1 = \lambda_2 = 1$. For cosmetic reasons, we also impose $f(1) = g(1) = 0$, although a constant energy shift doesn't affect the material. Second, we must have $f''(x) > 0$ and $g''(x) > 0$ for all $x > 0$, to ensure the stability of the material. Third, one wants the ability to control the material stiffness for small deformations, and this is done by adjusting $k = f''(1) > 0$ (for "stretching") and $C = g''(1) > 0$ (for "surface area change penalization"). Fourth, one wants to control how rapidly the material stiffens when $x \gg 1$ and $x \ll 1$, and this is controled by $p \geq 0$ (Figure 8). By keeping k constant and changing p, one can easily create KL nonlinear materials that behave equally for small deformations, but stiffen for large deformations in a controlable way (Figure 7). Contrast this ability to, say, a Neo-Hookean material which has 2 parameters, controlling stiffness and volume preservation at rest. Unlike our material, the Neo-Hookean large-strain behavior is baked-in and cannot be modified.

4.3 Generalization of Previous Work

It is interesting to ask how the terms of our energy (Equation 19) relate to the state of the art method [Chen et al. 2018] on Kirchoff-Love thin-shell simulation in computer graphics. Chen's work only

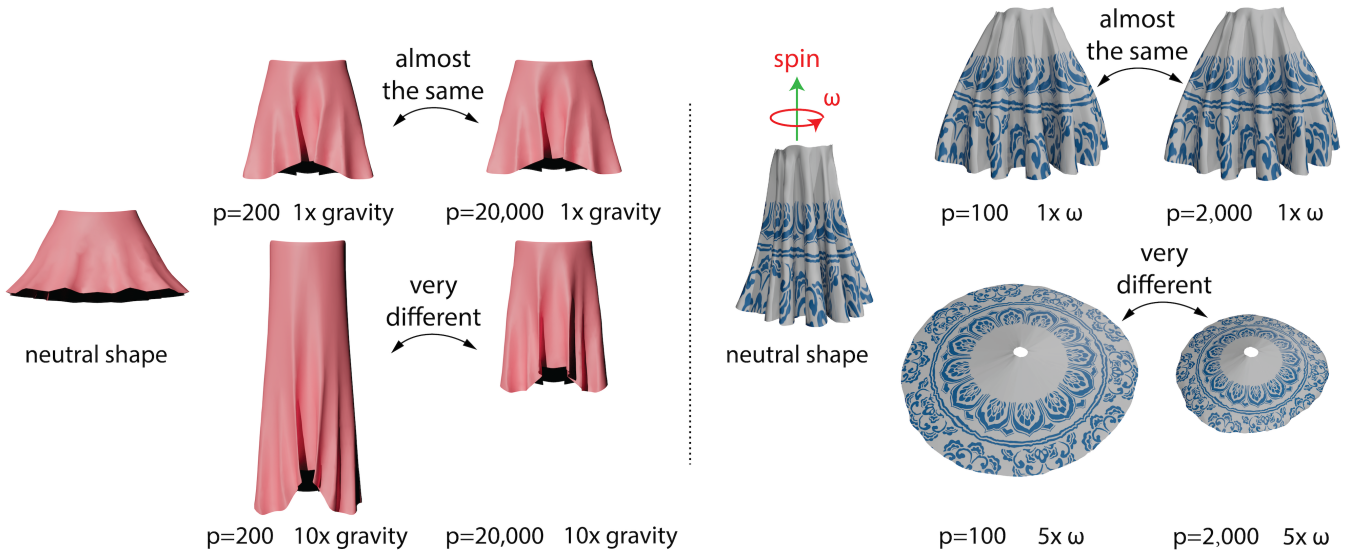

Fig. 7. **Design of general isotropic thin shell materials.** In both examples (left and right), we employ two materials: a material that does not stiffen under large deformations (left; low p; similar to the middle subfigure in Figure 8), and a material that stiffens substantially under large deformations (right; high p; similar to the right subfigure in Figure 8). When forces are small, both materials behave the same; but under large deformations, the right material stiffens and prevents the skirt from stretching. In the top and bottom examples, we drive the deformation with gravity and Coriolis forces due to spinning, respectively.

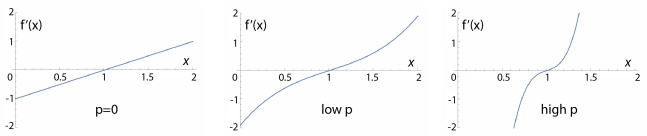

Fig. 8. **Our material curves for three different values of p.** Note that $f'(x) = \frac{1}{3}p(x-1)^3 + k(x-1)$ (the "force"). When $p = 0$, stiffness $f''(x)$ is constant. As p grows, material stiffens faster and faster.

analyzed the St. Venant-Kirchhoff material (SV), and made an assumption that we do not make: they assumed that (⋆) the thin-shell in-plane stretches are small, $||\bar{a}^{-1}(a-\bar{a})||_\infty = ||\bar{a}^{-1}a - I_2||_\infty < h$, where I_2 is the 2×2 identity matrix. Under this assumption, Chen's energy density (with respect to an infinitesimal amount of the undeformed surface) is

$$E_{\text{Chen}} = \frac{h}{4}||\bar{a}^{-1}a - I||^2_{SV} + \frac{h^3}{12}||\bar{a}^{-1}(b-\bar{b})||^2_{SV} + O(h^4). \quad (29)$$

In Appendix F, we prove that for SV, the terms of our energy (Equation 19) are

$$\psi(F_0) = \frac{1}{4}||\bar{a}^{-1}(a-\bar{a})||^2_{SV}, \quad (30)$$

$$\frac{1}{2}\left(\frac{dP}{dF} : F_1\right) : F_1 = ||\bar{a}^{-1}(\frac{\bar{b}\bar{a}^{-1}a + a\bar{a}^{-1}\bar{b}}{2} - b)||^2_{SV} +$$
$$+\frac{1}{2} < \bar{a}^{-1}(a-\bar{a}),\ \bar{a}^{-1}\left(c - \bar{b}\bar{a}^{-1}b - b\bar{a}^{-1}\bar{b} + \bar{b}\bar{a}^{-1}a\bar{a}^{-1}\bar{b}\right) >_{SV}, \quad (31)$$

$$P : \left(F_2 - 2HF_1\right) = < \bar{a}^{-1}(a-\bar{a}),\ \bar{a}^{-1}\left(\bar{b}\bar{a}^{-1} - 2H\,I_2\right)\left(\bar{b}\bar{a}^{-1}a - b\right) >_{SV}, \quad (32)$$

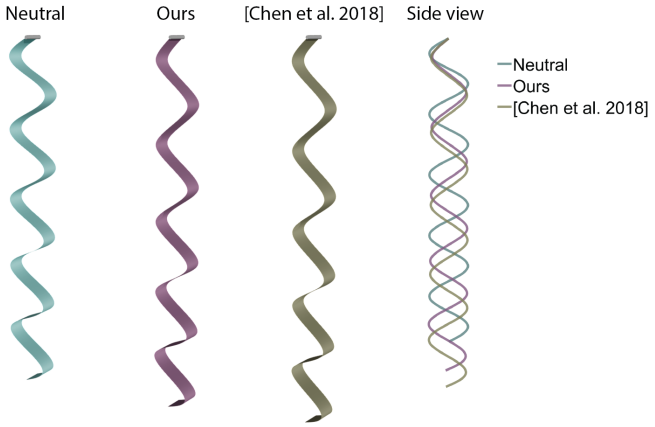

Fig. 9. **Visual effect of dropping terms in Chen's method.** We compare Chen's energy [Chen et al. 2018] to ours. Our energy is exact up to $O(h^5)$ for any deformation, whereas Chen assumed that the stretches are small. Chen's method is $O(h^4)$ under the small-stretch assumption, and the h^3 terms are not exact under large deformations.

where c is the third fundamental form. We define the $||\ ||_{SV}$ norm and the dot product in Appendix F. Therefore, our in-plane stretching term (Equation 30) matches Chen's in-plane stretching term. Under Chen's assumption (⋆), up to $O(h^4)$, the first and second term of Equation 31 become $||\bar{a}^{-1}(b-\bar{b})||^2_{SV}$ and vanish, respectively, the entire Equation 32 vanishes, and the term $h^3\psi(F_0)K$ also vanishes. Therefore, under Chen's assumption (⋆), our bending terms (Equations 31 and 32) become identical to Chen's bending term $||\bar{a}^{-1}(b-\bar{b})||^2_{SV}$. Our energy generalizes the thin-shell energy

given in [Chen et al. 2018] as follows: it is is exact up to $O(h^5)$ (no terms are dropped), it supports arbitrarily large deformations (no $\star$ assumption), and arbitrary hyperelastic materials (not only SV). Figure 9 gives a visual comparison to Chen's method.

4.4 Discretization

Let $\triangle_{ijk}$ be a triangle on the deformed thin-shell with vertices x_i, x_j, x_k, and let $\mathcal{T}$ be the canonical unit 2D triangle with vertices $(0,0), (1,0), (0,1)$. Then, locally the triangle $\triangle_{ijk}$ is embedded by the affine function

$$r_{ijk} : \mathcal{T} \to \mathbb{R}^3, \quad r_{ijk}(u,v) = x_i + u(x_j - x_i) + v(x_k - x_i). \tag{33}$$

From the definitions in Section 4, we can derive that

$$\begin{aligned} t &= \begin{bmatrix} x_j - x_i & x_k - x_i & n_{ijk} \end{bmatrix} \\ q &= 2 \begin{bmatrix} n_i - n_j & n_i - n_k & 0 \end{bmatrix} \\ a &= \begin{bmatrix} (x_j - x_i)^2 & (x_j - x_i) \cdot (x_k - x_i) \\ (x_j - x_i) \cdot (x_k - x_i) & (x_k - x_i)^2 \end{bmatrix} \\ b &= 2 \begin{bmatrix} (n_i - n_j) \cdot (x_i - x_j) & (n_i - n_j) \cdot (x_i - x_k) \\ (n_i - n_k) \cdot (x_i - x_j) & (n_i - n_k) \cdot (x_i - x_k) \end{bmatrix} \end{aligned} \tag{34}$$

where n_{ijk} is the triangle normal, and n_i is the *mid-edge* normal on the edge opposite vertex i [Chen et al. 2018]. The formulas for the undeformed quantities are the same; they just use undeformed positions and normals. The first and second fundamental forms are only used to compute the undeformed curvatures $H = \mathrm{tr}(\bar{a}^{-1}\bar{b})/2$ and $K = \det(\bar{a}^{-1}\bar{b})$, and are therefore needed only in the undeformed configuration. We note that K, H, F_0, F_1, F_2 are all constant over each triangle. Therefore, the discrete elastic energy for each triangle $\triangle_{ijk}$ can be written as

$$\begin{aligned} E_{ijk} = A_{ijk} h \Big(& \frac{h^2 K}{12} \psi(F_0) + \\ & + \frac{1}{2} \big(\psi(F_0 + \frac{\sqrt{3}}{6} h F_1) + \psi(F_0 - \frac{\sqrt{3}}{6} h F_1) \big) + \\ & + \psi(F_0 + \frac{h^2}{24}(F_2 - 2HF_1)) - \psi(F_0 - \frac{h^2}{24}(F_2 - 2HF_1)) \Big), \end{aligned} \tag{35}$$

where A_{ijk} is the triangle area in the rest state.

5 FEM SIMULATION

We can now use the discrete thin-shell elastic energy in the usual way to timestep the thin-shell,

$$M\ddot{x} + (\alpha_M M + \alpha_K K(x))\dot{x} + f_{\text{internal}}(x) = f_{\text{external}}, \tag{36}$$

where x contains the positions of the vertices of the thin-shell mesh; $f_{\text{internal}}(x)$ and $K(x)$ are the gradient and Hessian of the discrete elastic energy (Equation 35) with respect to thin shell vertex positions; α_M and α_K are Rayleigh damping coefficients; and f_{external} are per-vertex external forces such as gravity. The Hessian $K(x)$ is also needed for implicit integration; we give the gradient and Hessian in Appendix E. We use implicit backwards Euler integration, augmented with a line search for stability [Gast et al. 2015]. We give our mass matrix M in Section 5.1. We note that the above simulation setup, while "standard", is by no means the only way to use our thin-shell energy. Our energy is agnostic of the specific simulation setup, and could plug into simulations with contact, constraints, multibody dynamics simulations, simulations that couple elastic solids and thin-shells, etc.

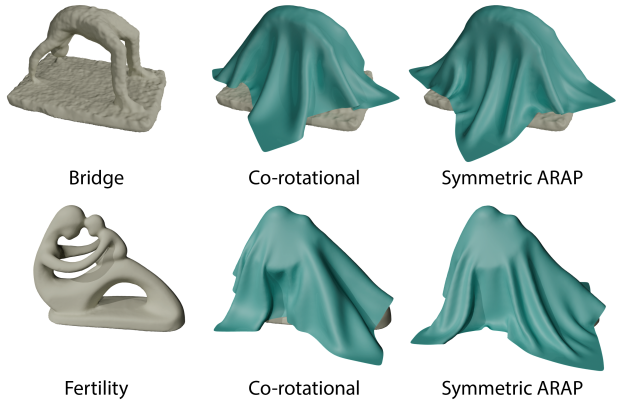

Fig. 10. **Draping thin-shells with nonlinear materials against complex objects.** Observe that different materials result in different fold formation.

5.1 Mass Matrix

The thin-shell kinetic energy is

$$T = \frac{1}{2} \int_{-h/2}^{h/2} d\xi \int_{UV} \hat{\rho} ||\dot{x} + \xi \dot{n}||^2 |G| du dv, \tag{37}$$

where $\hat{\rho}$ is the surface mass density pull-back. According to [Clyde et al. 2017], the term $\xi\dot{n}$ introduces complex dependencies on both x and $\dot{x}$, but the multiplier ξ makes this term small in most contexts. For simplicity, we discard this term, resulting in the standard form for the finite element mass matrix in Equation 38, which is also used by [Kiendl et al. 2015]. For each triangle $\triangle_{ijk}$, the mass matrix entries are (derivation is in Appendix D)

$$M_{IJ} = 2\hat{\rho} h A_{ijk} \big(1 + \frac{h^2 K}{12}\big) C_{IJ}, \qquad \text{for } I, J = 0, 1, 2, \tag{38}$$

where A_{ijk} is the surface area of the undeformed triangle, and the constant C_{IJ} is given in Appendix D. The global mass matrix can be obtained by inserting the above local triangle mass matrices into the global thin-shell triangle mesh degrees of freedom, as per the standard procedure in FEM.

6 RESULTS

In Figures 1, 4, 10 and 15, we compare various nonlinear materials by suspending or colliding the thin-shell against various geometric objects. It can be seen that different materials produce different stretching and folding patterns. We use signed distance fields and implicit penalty contact forces to resolve collisions; the specific collision method choice is orthogonal to our contribution. Our method explicitly identifies the terms linear in h (in-plane stretching) and cubic in h (bending); in Figure 5, we run thin-shell simulations whereby we observe the effect of changing the thin-shell stiffness h. Our categorization of all Kirchhoff-Love hyperelastic energies enables us to perform material design of thin-shell energies; we demonstrated this in Figure 7. We compared our method to Chen's

Table 1. **Simulation performance statistics.** All timings are in seconds. Intel i9 12900KF 16-core processor, 128 GB RAM. "Fig" gives the figure number in our paper showing this example. The "force" is the time to evaluate the internal force and tangent stiffness matrix terms, to form the linear system of equations to determine the line search direction. "solve" is the time to solve the linear system of equations. "line search" is the time to perform the line search inside the integrator loop for stability. Δt is the simulation timestep, and "total time" is the total computation time per timestep.

	Fig	#vtx	#tri	collision	force	solve	line search	Δt	total time
cloth1	10	7321	14400	0.056	0.223	0.068	0.323	0.0008	0.6764
cloth2	4,15,16	12961	25600	0.067	0.414	0.175	0.576	0.0006	1.2340
short skirt	7	3757	7271	0.022	0.125	0.039	0.166	0.0005	0.3522
dress	1	12146	23949	0.080	0.426	0.179	0.543	0.0001	1.2287
long skirt	7	3634	7026	0.020	0.119	0.032	0.158	0.0005	0.3294
bob	5	5344	10688	0.029	0.176	0.090	0.245	0.0010	0.5397
spot	5	2930	5856	0.017	0.097	0.038	0.134	0.0010	0.2848

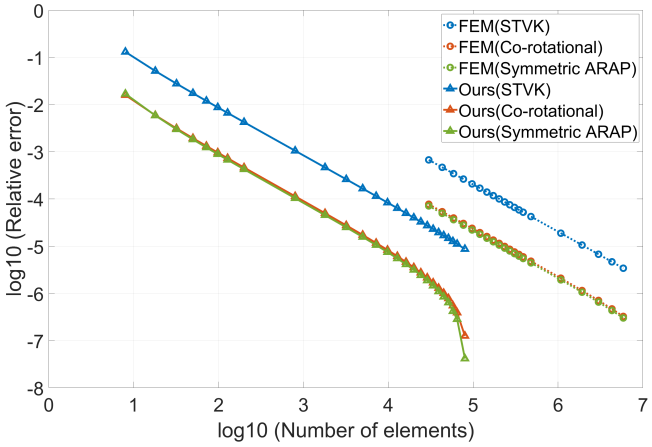

Fig. 11. **Convergence to ground truth for "F-based" materials:** Neutral shape is a cylinder, i.e., the mean curvature is non-zero. Deformed shape is a plate, i.e., we unroll the cylinder into a rectangle. In this example, the ground truth thin-shell energy can be calculated analytically ("ground truth"); we do this without any discretization. "Relative error" is the absolute value of the difference to the ground truth. Methods shown are our method and the FEM volumetric method (thin volumetric tetrahedralized slab). "Elements" are triangles for our method, and tetrahedra for the volumetric FEM method. Our method produces a much lower error for the same number of elements. StVK converges more slowly than the other materials because it is more nonlinear.

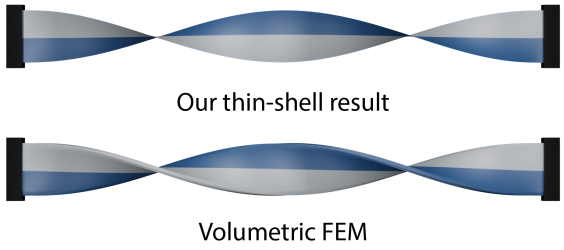

Fig. 12. **Comparison to FEM volumetric simulation (out-of-plane):** We simulated this thin ribbon using our method (1,111 vertices and 2,000 triangles). We then generated the volumetric mesh (3,333 vertices and 120,000 tetrahedra) by extruding the mid-surface, and simulated it using a FEM volumetric simulation. Our method is approximately 1.6x faster overall than the volumetric simulation. Evaluation of forces + stiffness matrices, and the system solve, are 1.3x and 2.9x times faster, respectively. Although the volumetric mesh has 6x as many elements as the thin-shell, the speedup is not 6x because evaluating thin-shell forces and stiffness matrices is more complex, due to evaluating normal derivatives. The thin shell has 3x fewer vertices than the volumetric mesh, and therefore, the system solve size is 3x smaller. This translates into a substantial speed advantage, and this advantage grows with the mesh complexity. Same material properties (Co-rotational material, i.e., non-E-based material). It can be seen that our result closely matches the midsurface of the volumetric result. Note that the thin-shell method is also substantially more accurate than the volumetric method (Figure 11).

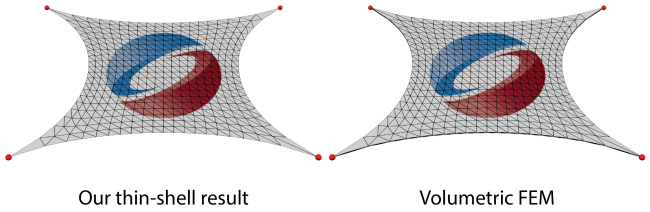

Fig. 13. **Comparison to FEM volumetric simulation (in-plane):** The rest and initial shape is a square flag. We drag the four corners to show that our model has nearly the same in-plane stretching behavior as the volumetric FEM simulation with the same material properties. ARAP material.

method in Figure 9, demonstrating the visual difference between using our exact $O(h^3)$ terms vs Chen's method where some $O(h^3)$ terms were dropped. In Figure 11, we demonstrate that the energy computed by our method, for a F-based material with odd powers, (1) converges to the analytical ground truth energy as we refine the triangle mesh, and (2) converges faster than volumetric FEM of the same h. In Figures 12 and 13, we demonstrate that our method matches the volumetric FEM simulation for a small value of h. In Figure 14, we demonstrate the benefit of our SPD projection. We note that the bending energy terms are indefinite at rest, however, due to the in-plane stretching terms, the overall system is not indefinite at rest. Under large deformations, indefiniteness (buckling) can occur. Then, without SPD projection, the solver encounters a singular matrix and output is ill-defined at best. Our solver can progress due to the SPD projection, clamping singular values to small positive values. In Figure 16, we perform anisotropic thin-shell simulations whereby we added anisotropic terms to the ARAP energy.

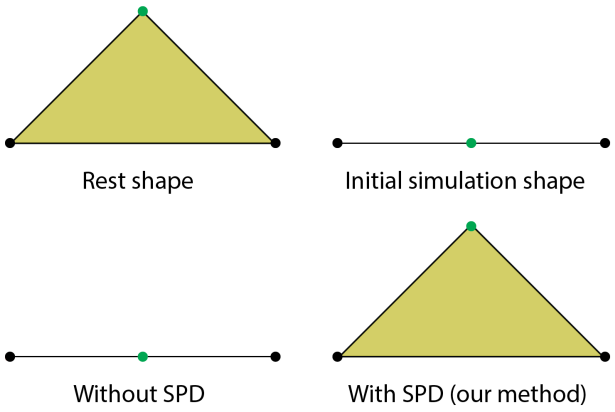

Fig. 14. **Benefit of SPD projection:** Rest shape is a triangle; fixed vertices are black and the free vertex is green. For the initial shape, we flatten the triangle to a line. Without SPD projection, simulation becomes unstable. With SPD projection, our method stably recovers the initial shape.

Table 1 gives the statistics and computational performance of our models. Observe that the "force" and "line search" times are approximately equal, with the "solve" time being substantially smaller than both of those. When using the line search, our method can stably take approximately 6× larger timesteps in our examples, which means that the total net benefit of using a line search is approximately $2-3\times$ in our examples. Our timesteps are generally approximately 5× larger than those reported in [Chen et al. 2018].

7 CONCLUSION

Starting from an arbitrary hyperelastic volumetric elastic energy density function ψ, we derived the corresponding Kirchoff-Love thin-shell energy as a function of the shell thickness h, exact up to $O(h^5)$. We are first to do so for general functions ψ and in particular our method can accommodate materials expressed as both even and odd powers of stretches, such as the Co-rotational linear FEM material, ARAP and Symmetric ARAP. Furthermore, we categorize all equivalence classes of volumetric functions ψ that result in the same thin-shell energy; we prove that the equivalence classes consists of all energy functions of 2×2 matrices. Our method makes it possible to start with any solid substance whose volumetric material behavior has been measured, and then simulate a thin-shell made from such a material.

Unlike most of prior work in computer graphics, our method does not need to treat in-plane stretching and bending separately. Instead, we observe that stretching and bending are simply consequences of deforming a thin volumetric continuum, i.e., they originate from the same process, as opposed to somehow from two separate processes. This is true in the real world, and our method follows this philosophy, and demonstrates how to apply it to arbitrary hyperelastic materials. Because of this, in-plane stretching and bending occur naturally in our formulation without any special treatment. Nonetheless, in computer graphics, one often wants to make the thin-shell more or less bendable, and we show that this can be accommodated by tweaking h.

Note that thin-shell simulation that matches a volumetric energy in the $h \to 0$ limit is inherently a highly nonlinear process and requires substantial investment of computation, as already observed by [Chen et al. 2018]. This limitation applies also to our work. However, as shown in our Figures 11 and 12, creating a 3D "slab" volumetric mesh and using a 3D solid simulator to approximate a thin-shell, results in substantially longer simulation times and less precise results.

Our method is limited to Kirchoff-Love shells; but shells that undergo out-of-plane shearing or stretching are also important in some applications; future work could investigate such shells, using the "director" concept [Weischedel 2012]. Accurate simulation of thin-shells results in highly nonlinear equations. While we address this with small timesteps, SPD projections and line-searching in our work, the tradeoffs between large timesteps and stability could be further investigated in future work.

ACKNOWLEDGMENTS

This research was sponsored in part by NSF (IIS-1911224), USC Annenberg Fellowship to Jiahao Wen, Bosch Research and Adobe Research.

REFERENCES

David Baraff and Andrew P. Witkin. 1998. Large Steps in Cloth Simulation. In *Proc. of ACM SIGGRAPH 98*. 43–54.

Miklos Bergou, Max Wardetzky, David Harmon, Denis Zorin, and Eitan Grinspun. 2006. A Quadratic Bending Model for Inextensible Surfaces. In *Proc. of the Eurographics Symp. on Geometry Processing (SGP)*. 227–230.

Javier Bonet and Richard D. Wood. 2008. *Nonlinear Continuum Mechanics for Finite Element Analysis, 2nd Ed.* Cambridge University Press.

Robert Bridson, Ronald Fedkiw, and John Anderson. 2002. Robust treatment of collisions, contact and friction for cloth animation. In *Proceedings of ACM SIGGRAPH 2002*. 594–603.

Hsiao-Yu Chen, Arnav Sastry, Wim M van Rees, and Etienne Vouga. 2018. Physical simulation of environmentally induced thin shell deformation. *ACM Trans. on Graphics (SIGGRAPH 2018)* 37, 4 (2018), 146.

Zhen Chen, Hsiao-Yu Chen, Danny M Kaufman, Mélina Skouras, and Etienne Vouga. 2021. Fine wrinkling on coarsely meshed thin shells. *ACM Transactions on Graphics (TOG)* 40, 5 (2021), 1–32.

Fehmi Cirak and Michael Ortiz. 2001. Fully C1-conforming subdivision elements for finite deformation thin-shell analysis. *Internat. J. Numer. Methods Engrg.* 51, 7 (2001), 813–833.

Fehmi Cirak, Michael Ortiz, and Peter Schröder. 2000. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *Internat. J. Numer. Methods Engrg.* 47, 12 (2000), 2039–2072.

Gabriel Cirio, Jorge Lopez-Moreno, David Miraut, and Miguel A Otaduy. 2014. Yarn-level simulation of woven cloth. *ACM Transactions on Graphics (SIGGRAPH Asia 2014)* 33, 6 (2014), 1–11.

Gabriel Cirio, Jorge Lopez-Moreno, and Miguel A Otaduy. 2016. Yarn-level cloth simulation with sliding persistent contacts. *IEEE transactions on visualization and computer graphics* 23, 2 (2016), 1152–1162.

David Clyde, Joseph Teran, and Rasmus Tamstorf. 2017. Simulation of nonlinear Kirchhoff-Love thin shells using subdivision finite elements. In *Symp. on Computer Animation (SCA)*, Vol. 17.

Xudong Feng, Wenchao Huang, Weiwei Xu, and Huamin Wang. 2022. Learning-Based Bending Stiffness Parameter Estimation by a Drape Tester. *ACM Transactions on Graphics (SIGGRAPH Asia 2022)* 41, 6 (2022), 1–16.

Akash Garg, Eitan Grinspun, Max Wardetzky, and Denis Zorin. 2007. Cubic Shells. In *Proc. of Symp. on Computer Animation*. 91–98.

Theodore F. Gast, Craig Schroeder, Alexey Stomakhin, Chenfanfu Jiang, and Joseph M. Teran. 2015. Optimization Integrator for Large Time Steps. *IEEE Trans. on Visualization and Computer Graphics* 21, 10 (2015), 1103–1115.

Yotam Gingold, Adrian Secord, Jefferson Y Han, Eitan Grinspun, and Denis Zorin. 2004. A discrete model for inelastic deformation of thin shells. In *Proceedings of Symp. on Computer Animation (SCA)*.

Eitan Grinspun, Yotam Gingold, Jason Reisman, and Denis Zorin. 2006. Computing discrete shape operators on general meshes. In *Computer Graphics Forum*, Vol. 25. Wiley Online Library, 547–556.

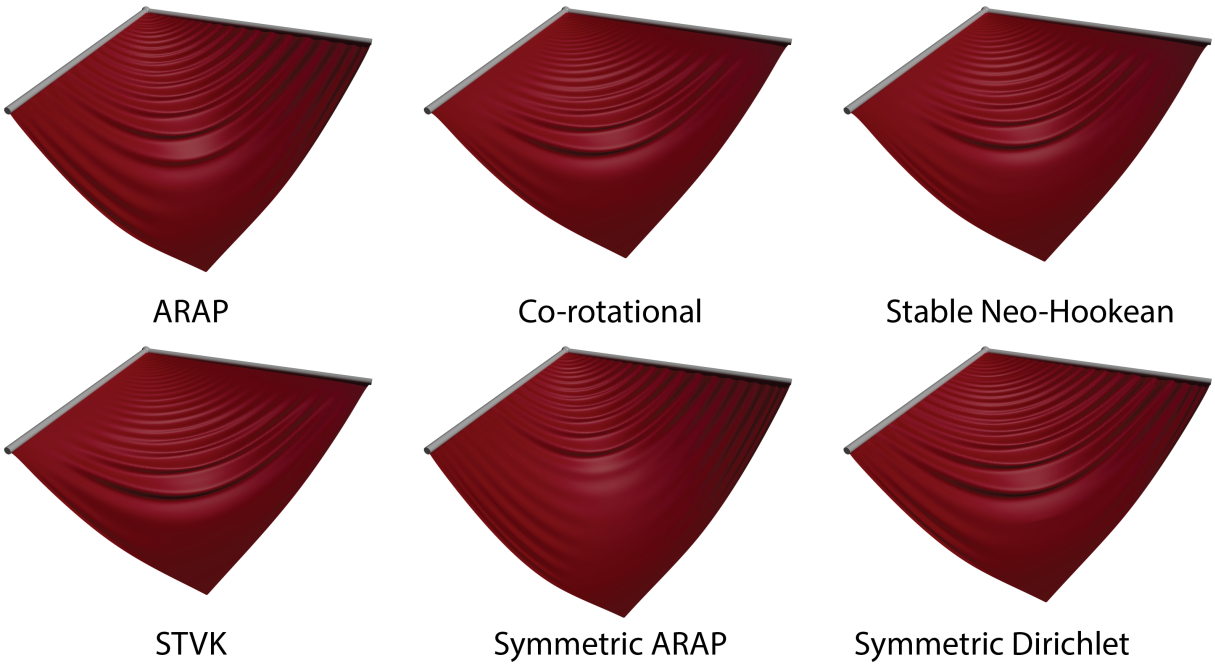

Fig. 15. **Suspending the cloth under gravity from an L-shaped frame,** for several nonlinear materials.

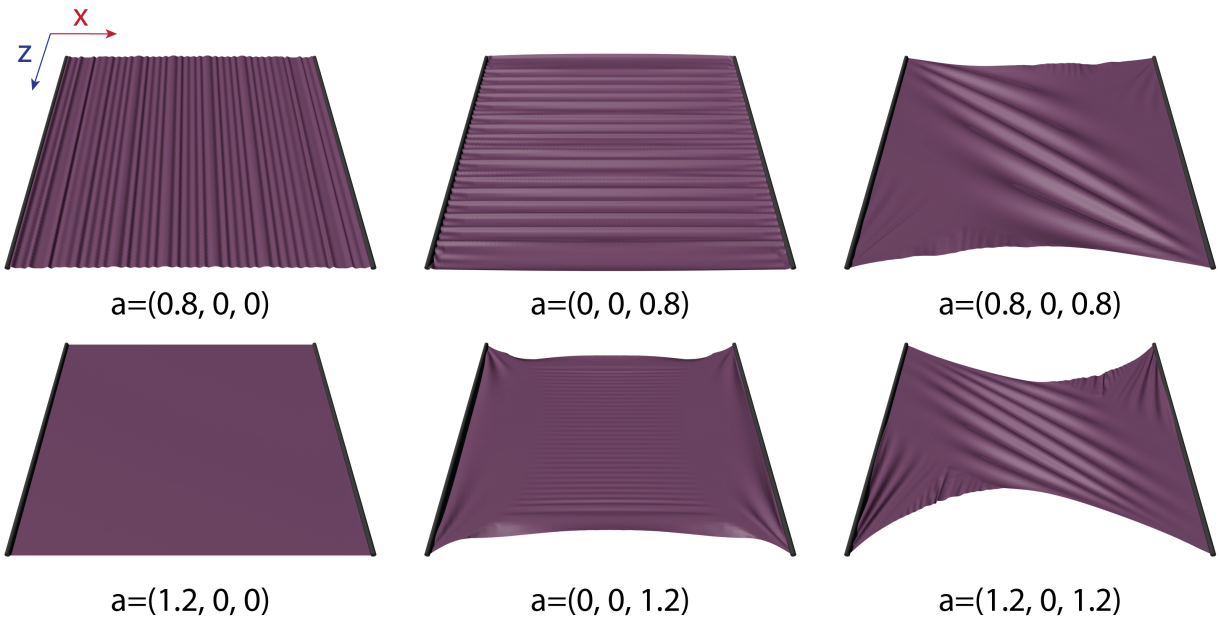

Fig. 16. **Our method supports anisotropic thin-shells formulated using odd powers of principal stretches:** Here, we start with the ARAP material, and then add an anisotropic energy term $\frac{\mu}{2}(\sqrt{I_5}\mathcal{S}(I_4))^2$, producing "Anisotropic ARAP". This is same exact anisotropic energy given in Equation 31 of [Smith et al. 2019], except that theirs was applied to 3D solids, and our paper demonstrates how to apply it to thin-shells. Note that the "Anisotropic ARAP" energy of [Smith et al. 2019] has terms that guard against near-inversion, and our method therefore inherits this useful property.

Eitan Grinspun, Anil N. Hirani, Mathieu Desbrun, and Peter Schröder. 2003. Discrete Shells. In *Proceedings of Symp. on Computer Animation (SCA).*

Qi Guo, Xuchen Han, Chuyuan Fu, Theodore Gast, Rasmus Tamstorf, and Joseph Teran. 2018. A material point method for thin shells with frictional contact. *ACM Trans. on Graphics (Proc. of SIGGRAPH 2018)* 37, 4 (2018), 1–15.

Jonathan M Kaldor, Doug L James, and Steve Marschner. 2008. Simulating knitted cloth at the yarn level. *ACM Transactions on Graphics (SIGGRAPH 2008)* 27, 4 (2008), 1–9.
Josef Kiendl, Ming-Chen Hsu, Michael CH Wu, and Alessandro Reali. 2015. Isogeometric Kirchhoff–Love shell formulations for general hyperelastic materials. *Computer Methods in Applied Mechanics and Engineering* 291 (2015), 280–303.
Theodore Kim. 2020. A Finite Element Formulation of Baraff-Witkin Cloth. In *Symp. on Computer Animation 2020.* Article 16, 9 pages.
Theodore Kim, Fernando De Goes, and Hayley Iben. 2019. Anisotropic elasticity for inversion-safety and element rehabilitation. *ACM Transactions on Graphics (SIGGRAPH 2018)* 38, 4 (2019), 1–15.
Theodore Kim and David Eberle. 2020. Dynamic Deformables: Implementation and Production Practicalities. In *ACM SIGGRAPH 2020 Courses.*
WT Koiter. 1970. On the mathematical foundation of shell theory. In *Proc. Int. Congr. of Mathematics, Nice*, Vol. 3. 123–130.
Jie Li, Gilles Daviet, Rahul Narain, Florence Bertails-Descoubes, Matthew Overby, George E. Brown, and Laurence Boissieux. 2018. An Implicit Frictional Contact Solver for Adaptive Cloth Simulation. *ACM Trans. on Graphics (SIGGRAPH 2018)* 37, 4, Article 52 (2018), 15 pages.
Richard H Macneal and Robert L Harder. 1985. A proposed standard set of problems to test finite element accuracy. *Finite elements in analysis and design* 1, 1 (1985), 3–20.
Eder Miguel, Derek Bradley, Bernhard Thomaszewski, Bernd Bickel, Wojciech Matusik, Miguel A Otaduy, and Steve Marschner. 2012. Data-driven estimation of cloth simulation models. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 519–528.
Eder Miguel, Rasmus Tamstorf, Derek Bradley, Sara C Schvartzman, Bernhard Thomaszewski, Bernd Bickel, Wojciech Matusik, Steve Marschner, and Miguel A Otaduy. 2013. Modeling and estimation of internal friction in cloth. *ACM Transactions on Graphics (SIGGRAPH Asia 2013)* 32, 6 (2013), 1–10.
Rahul Narain, Tobias Pfaff, and James F O'Brien. 2013. Folding and crumpling adaptive sheets. *ACM Transactions on Graphics (SIGGRAPH 2013)* 32, 4 (2013), 1–8.
Rahul Narain, Armin Samii, and James F O'Brien. 2012. Adaptive anisotropic remeshing for cloth simulation. *ACM Transactions on Graphics (SIGGRAPH Asia 2012)* 31, 6 (2012), 1–10.
Peter Petersen. 2016. Classical differential geometry. *Lecture notes, available from the authors webpage: http://www. math. ucla. edu/˜ petersen/DGnotes. pdf* (2016).
Rosa M Sánchez-Banderas, Alejandro Rodríguez, Héctor Barreiro, and Miguel A Otaduy. 2020. Robust eulerian-on-lagrangian rods. *ACM Transactions on Graphics (SIGGRAPH 2020)* 39, 4 (2020), 59–1.
Eftychios Sifakis and Jernej Barbič. 2012. FEM simulation of 3D deformable solids: a practitioner's guide to theory, discretization and model reduction. In *ACM SIGGRAPH 2012 Courses.* 20:1–20:50.
Juan C Simo and David D Fox. 1989. On a stress resultant geometrically exact shell model. Part I: Formulation and optimal parametrization. *Computer Methods in Applied Mechanics and Engineering* 72, 3 (1989), 267–304.
Breannan Smith, Fernando De Goes, and Theodore Kim. 2018. Stable neo-hookean flesh simulation. *ACM Transactions on Graphics (TOG)* 37, 2 (2018), 1–15.
Breannan Smith, Fernando De Goes, and Theodore Kim. 2019. Analytic Eigensystems for Isotropic Distortion Energies. *ACM Trans. Graphics (TOG)* 38, 1, Article 3 (2019), 15 pages.
Rasmus Tamstorf and Eitan Grinspun. 2013. Discrete Bending Forces and Their Jacobians. *Graph. Models* 75, 6 (2013), 362–370.
Joseph Teran, Eftychios Sifakis, Geoffrey Irving, and Ronald Fedkiw. 2005. Robust Quasistatic Finite Elements and Flesh Simulation. In *Symp. on Computer Animation (SCA).* 181–190.
Bernhard Thomaszewski, Markus Wacker, and Wolfgang Straßer. 2006. A consistent bending model for cloth simulation with corotational subdivision finite elements. In *Proceedings of Symp. on Computer Animation (SCA).*
KC Valanis and Robert F Landel. 1967. The strain-energy function of a hyperelastic material in terms of the extension ratios. *Journal of Applied Physics* 38, 7 (1967), 2997–3002.
Pascal Volino, Nadia Magnenat-Thalmann, and Francois Faure. 2009. A simple approach to nonlinear tensile stiffness for accurate cloth simulation. *ACM Trans. on Graphics (SIGGRAPH 2009)* 28, 4, Article 105 (2009), 16 pages.
Huamin Wang, James O'Brien, and Ravi Ramamoorthi. 2011. Data-Driven Elastic Models for Cloth: Modeling and Measurement. *ACM Trans. on Graphics (SIGGRAPH 2011)* 30, 4 (2011), 71:1–71:12.
Clarisse Weischedel. 2012. *A discrete geometric view on shear-deformable shell models.* Ph.D. Dissertation. Georg-August-Universität Göttingen.
Hongyi Xu, Funshing Sin, Yufeng Zhu, and Jernej Barbič. 2015. Nonlinear material design using principal stretches. *ACM Trans. on Graphics (SIGGRAPH 2015)* 34, 4 (2015), 1–11.

A TAYLOR EXPANSION OF ψ

The expansion is performed as

$$\psi(F) = \psi(F_0 + \xi F_1 + \xi^2 F_2 + O(\xi^3)) = \tag{39}$$

$$= \psi(F_0) + P(F_0) : \big(\xi F_1 + \xi^2 F_2 + O(\xi^3)\big) +$$
$$+\frac{1}{2}\Big(\frac{dP}{dF}_{|F=F_0} : \big(\xi F_1 + \xi^2 F_2 + O(\xi^3)\big)\Big) : \big(\xi F_1 + \xi^2 F_2 + O(\xi^3)\big) +$$
$$+ O((\xi F_1 + \xi^2 F_2 + O(\xi^3))^3) = \tag{40}$$

$$= \psi(F_0) + \xi\Big(P(F_0) : F_1\Big) +$$
$$+\xi^2\Big(P(F_0) : F_2 + \frac{1}{2}\big(\big(\frac{dP}{dF}_{|F=F_0} : F_1\big) : F_1\big)\Big) + O(\xi^3). \tag{41}$$

B DETERMINANT OF G

Starting from Equation 3, we have

$$|G| = \begin{vmatrix} Y_u & Y_v & N \end{vmatrix} + \xi\big(\begin{vmatrix} N_u & Y_v & N \end{vmatrix} + \begin{vmatrix} Y_u & N_v & N \end{vmatrix}\big) +$$
$$+\xi^2 \begin{vmatrix} N_u & N_v & N \end{vmatrix}. \tag{42}$$

The matrix form of the Weingarten map (the shape operator) is

$$\bar{\ell} = \bar{a}^{-1}\bar{b} = \begin{bmatrix} \bar{\ell}_{11} & \bar{\ell}_{21} \\ \bar{\ell}_{12} & \bar{\ell}_{22} \end{bmatrix}, \qquad K = \det(\bar{\ell}) \quad 2H = \mathrm{tr}(\bar{\ell}). \tag{43}$$

where K and H are the Gauss and mean curvature in the undeformed configuration, respectively. By Weingarten equations, we have

$$\begin{bmatrix} N_u & N_v \end{bmatrix} = -\begin{bmatrix} Y_u & Y_v \end{bmatrix}\bar{\ell} = -\begin{bmatrix} \bar{\ell}_{11}Y_u + \bar{\ell}_{12}Y_v & \bar{\ell}_{21}Y_u + \bar{\ell}_{22}Y_v \end{bmatrix}. \tag{44}$$

Therefore,

$$\begin{aligned}
\begin{vmatrix} N_u & Y_v & N \end{vmatrix} &= (N_u \times Y_v)\cdot N \\
&= -((\bar{\ell}_{11}Y_u + \bar{\ell}_{12}Y_v)\times Y_v)\cdot N \\
&= -\bar{\ell}_{11}(Y_u \times Y_v)\cdot N \\
&= -\bar{\ell}_{11}\sqrt{\det(\bar{a})}\,, \\
\begin{vmatrix} Y_u & N_v & N \end{vmatrix} &= (Y_u \times N_v)\cdot N \\
&= -(Y_u \times (\bar{\ell}_{21}Y_u + \bar{\ell}_{22}Y_v))\cdot N \\
&= -\bar{\ell}_{22}(Y_u \times Y_v)\cdot N \\
&= -\bar{\ell}_{22}\sqrt{\det(\bar{a})}\,, \\
\begin{vmatrix} N_u & N_v & N \end{vmatrix} &= (N_u \times N_v)\cdot N \\
&= ((\bar{\ell}_{11}Y_u + \bar{\ell}_{12}Y_v)\times(\bar{\ell}_{21}Y_u + \bar{\ell}_{22}Y_v))\cdot N \\
&= (\bar{\ell}_{11}\bar{\ell}_{22} - \bar{\ell}_{12}\bar{\ell}_{21})(Y_u \times Y_v)\cdot N = K\sqrt{\det(\bar{a})},
\end{aligned} \tag{45}$$

which gives the final result

$$|G| = (1 - 2\xi H + \xi^2 K)\sqrt{\det(\bar{a})}. \tag{46}$$

C PROOF OF THEOREM 1

Observe that matrices of format stated in Theorem 1 are those transformations that transform the tangent vectors into the xy plane in $\mathbb{R}^3$ (using an otherwise arbitrary linear transformation), but send the normal vector to $[0, 0, 1]$. Geometrically, such transformations are in-tangent-plane stretches, followed by a 3D rotation that aligns the tangent plane onto the xy plane. If ψ_1 and ψ_2 don't agree on

some F in the above format, this means that they don't agree on some in-plane stretch, and therefore, the thin-shells energies differ. Conversely, suppose ψ_1 and ψ_2 agree on any such F. Note that $F_0 = tT^{-1}$, whereby T is a matrix whose first two columns span a basis of the tangent space of the undeformed surface (but are not necessarily orthogonal), and the third column is the undeformed normal. Therefore, we have

$$T = V\begin{bmatrix} R & 0 \\ 0 & 1 \end{bmatrix}, \qquad t = U\begin{bmatrix} r & 0 \\ 0 & 1 \end{bmatrix}, \tag{47}$$

where $R, r \in \mathbb{R}^{2\times 2}$ are invertible, and U is some orthogonal basis of the linear subspace spanned by the first two columns of t. It follows that

$$F_0 = U\begin{bmatrix} r & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} R & 0 \\ 0 & 1 \end{bmatrix}^{-1} V^T. \tag{48}$$

Because ψ_1, ψ_2 are hyperelastic energies, they are invariant to rotating the deformed object, hence the presence of U does not change the energies. The rest of F_0 is now in the stated format, and therefore we have $\psi_1(F_0) = \psi_2(F_0)$. In Equations 13 and 14, we see that F_1 and F_2 are of the format

$$t\begin{bmatrix} * & * & 0 \\ * & * & 0 \\ 0 & 0 & 0 \end{bmatrix} T^{-1}. \tag{49}$$

Now, due to the presence of 0 in the lower-right corner, by the same analysis for F_1 and F_2 as above for F_0, we see ψ_1 and ψ_2 agree on any matrix of the form $F_0 + \alpha F_1 + \beta F_2$, for any $\alpha, \beta \in \mathbb{R}$. But – in our exact continuous energy (up to $O(h^5)$; Equation 24), ψ is only ever evaluated on arguments of the form $F_0 + \alpha F_1 + \beta F_2$. Therefore, the two thin-shell energies are the same. ∎

D MASS MATRIX

The mass matrix entry (I, J) (for $I = 0, 1, 2$ and $J = 0, 1, 2$) of a triangle is

$$\begin{aligned} M_{IJ} &= \int_{-h/2}^{h/2} d\xi \int_{UV} \hat{\rho} N_I N_J |G| du dv = \\ &= \int_{-h/2}^{h/2} d\xi \int_{\Omega} \hat{\rho} N_I N_J (1 - 2\xi H + \xi^2 K) dS = \\ &= \left(h\hat{\rho} + \frac{h^3 \hat{\rho} K}{12}\right) \int_{\Omega} N_I N_J dS = \\ &= 2\hat{\rho} h \left(1 + \frac{h^2 K}{12}\right) A_{ijk} \int_{UV} N_I(u,v) N_J(u,v) du dv, \end{aligned} \tag{50}$$

where $\hat{\rho}$ is the pull-back of the surface mass density to UV, and N_0, N_1, N_2 are the shape functions for barycentric coordinates:

$$N_0 = 1 - u - v, \quad N_1 = u, \quad N_2 = v. \tag{51}$$

Therefore, the integration of $N_*(u,v)N_\#(u,v)$ over the parameterization domain UV produces

$$C_{00} = \int_0^1 du \int_0^{1-u} dv \left[(1-u-v)(1-u-v)\right] = \frac{1}{12}, \tag{52}$$

$$C_{11} = C_{22} = \int_0^1 du \int_0^{1-u} dv \left[u^2\right] = \frac{1}{12}, \tag{53}$$

$$C_{12} = \int_0^1 du \int_0^{1-u} dv \left[uv\right] = \frac{1}{24}, \tag{54}$$

$$C_{01} = C_{02} = \int_0^1 du \int_0^{1-u} dv \left[(1-u-v)u\right] = \frac{1}{24}. \tag{55}$$

E GRADIENT AND HESSIAN OF E

In order to compute the gradient and Hessian of E (Equation 35) with respect to the deformed thin-shell vertex positions, one needs the following first and second derivatives. Let $F = aF_0 + bF_1 + cF_2$, for some constants a, b, c. Then, for the gradient, we need

$$\begin{aligned} \frac{\partial \psi(F)}{\partial x} &= P : \frac{\partial F}{\partial x}, \\ \frac{\partial F_0}{\partial x} &= \frac{\partial F_0}{\partial t} : \frac{\partial t}{\partial x}, \\ \frac{\partial F_1}{\partial x} &= \frac{\partial F_1}{\partial q} : \frac{\partial q}{\partial x} + \frac{\partial F_1}{\partial t} : \frac{\partial t}{\partial x}, \\ \frac{\partial F_2}{\partial x} &= \frac{\partial F_2}{\partial q} : \frac{\partial q}{\partial x} + \frac{\partial F_2}{\partial t} : \frac{\partial t}{\partial x}. \end{aligned} \tag{56}$$

The above derivatives can be computed directly from the definitions of F_0, F_1, F_2. For the Hessian, we need

$$\begin{aligned} \frac{\partial^2 \psi(F)}{\partial x^2} &= \left(\frac{\partial F}{\partial x}\right)^T : \frac{dP}{dF} : \frac{\partial F}{\partial x} + P : \frac{\partial^2 F}{\partial x^2}, \\ \frac{\partial^2 F_0}{\partial x^2} &= \frac{\partial F_0}{\partial t} : \frac{\partial^2 t}{\partial x^2}, \\ \frac{\partial^2 F_1}{\partial x^2} &= \frac{\partial F_1}{\partial q} : \frac{\partial^2 q}{\partial x^2} + \frac{\partial F_1}{\partial t} : \frac{\partial^2 t}{\partial x^2}, \\ \frac{\partial^2 F_2}{\partial x^2} &= \frac{\partial F_2}{\partial q} : \frac{\partial^2 q}{\partial x^2} + \frac{\partial F_2}{\partial t} : \frac{\partial^2 t}{\partial x^2}, \end{aligned} \tag{57}$$

where again the required second derivatives in the above equations can be derived from the definitions of F_0, F_1, F_2.

F EXPRESSING OUR ENERGY TERMS VIA SURFACE FUNDAMENTAL FORMS

In this Appendix, we express our energy terms via surface fundamental forms. In Section 4.3, these results enabled us to relate our method to prior work [Chen et al. 2018; Weischedel 2012] that used the Saint-Venant Kirchhoff (SV) material.

F.1 Preliminaries

Everywhere in the derivations below, we use the properties of trace:

$$A : B = \mathrm{tr}(AB^T), \qquad \mathrm{tr}(A) = \mathrm{tr}(A^T), \qquad \mathrm{tr}(AB) = \mathrm{tr}(BA), \quad (58)$$

$$\mathrm{tr}(A_1 \dots A_m) = \mathrm{tr}(A_2 \dots A_m A_1) \ \text{(cyclic property)}. \quad (59)$$

The definitions of the "SV" inner-product and norm are as follows:

$$< A, B >_{SV} = \frac{\lambda}{2}\mathrm{tr}(A)\mathrm{tr}(B) + \mu\,\mathrm{tr}(AB) \quad (60)$$

$$||A||^2_{SV} = < A, A >_{SV} = \frac{\lambda}{2}\mathrm{tr}^2(A) + \mu\mathrm{tr}(A^2). \quad (61)$$

Note that for a deformation gradient F, we have $\psi(F) = ||\frac{1}{2}(F^TF - I)||^2_{SV}$. Everywhere in our proofs below, we separately prove the cases $\lambda = 1, \mu = 0$ and $\lambda = 0, \mu = 1$; this is sufficient because the elastic energy ψ is a linear function of the Lamé parameters λ and μ.

F.2 Proving Equation 30

$$\psi(F_0) = \psi(gG^{-1}) \overset{\lambda=1,\mu=0}{=\!=} \frac{1}{2}\mathrm{tr}^2\Big(\frac{1}{2}(G^{-T}g^TgG^{-1} - I_3)\Big) = \quad (62)$$

$$= \frac{1}{8}\mathrm{tr}^2\big(\bar{a}^{-1}a - I_3\big) \overset{\lambda=1,\mu=0}{=\!=} \frac{1}{4}||\bar{a}^{-1}(a - \bar{a})||^2_{SV}. \quad (63)$$

$$\psi(F_0) = \psi(gG^{-1}) \overset{\lambda=0,\mu=1}{=\!=} \frac{1}{4}\mathrm{tr}\big((G^{-T}g^TgG^{-1} - I_3)^2\big) = \quad (64)$$

$$= \frac{1}{4}\mathrm{tr}\big((\bar{a}^{-1}a - I_3)^2\big) \overset{\lambda=0,\mu=1}{=\!=} \frac{1}{4}||\bar{a}^{-1}(a - \bar{a})||^2_{SV}. \quad (65)$$

F.3 Proving Equation 31

First, recall that for the Saint-Venant Kirchhoff material (SV), we have [Bonet and Wood 2008]

$$P = \frac{\lambda}{2}\Big(\mathrm{tr}(F^TF - I_3)\Big)F + \mu F\Big(F^TF - I_3\Big). \quad (66)$$

We can now derive

$$\frac{dP}{dF} : F_1 = \lim_{t\to 0}\frac{P(F_0 + tF_1) - P(F_0)}{t} \overset{\lambda=1,\mu=0}{=\!=} \quad (67)$$

$$= \frac{1}{2}\lim_{t\to 0}\frac{1}{t}\Big(\big(\mathrm{tr}(F_0^TF_0 - I_3) + t\big(\mathrm{tr}(F_1^TF_0) +$$

$$+\mathrm{tr}(F_0^TF_1)\big) + O(t^2)\big)\big(F_0 + tF_1\big) - \mathrm{tr}(F_0^TF_0 - I_3)F_0\Big) =$$

$$= \frac{1}{2}\Big(\big(\mathrm{tr}(F_1^TF_0) + \mathrm{tr}(F_0^TF_1)\big)F_0 + \mathrm{tr}(F_0^TF_0 - I)F_1\Big), \quad (68)$$

and

$$\frac{dP}{dF} : F_1 = \lim_{t\to 0}\frac{P(F_0 + tF_1) - P(F_0)}{t} \overset{\lambda=0,\mu=1}{=\!=} \quad (69)$$

$$= \lim_{t\to 0}\frac{1}{t}\Big(\big(F_0 + tF_1\big)\big((F_0 + tF_1)^T(F_0 + tF_1) - I_3\big) -$$

$$-F_0(F_0^TF_0 - I_3)\Big) = F_0F_1^TF_0 + F_0F_0^TF_1 + F_1F_0^TF_0 - F_1. \quad (70)$$

Case 1: $\lambda = 1, \mu = 0$

In order to compute $\frac{1}{2}\big(\frac{dP}{dF} : F_1\big) : F_1$, we first evaluate

$$\big(\mathrm{tr}(F_1^TF_0) + \mathrm{tr}(F_0^TF_1)\big)\big(F_0 : F_1\big) = \quad (71)$$

$$= \Big(\mathrm{tr}\big(\bar{a}^{-1}(\bar{b}\bar{a}^{-1}a - b)\big) + \mathrm{tr}\big(\bar{a}^{-1}(a\bar{a}^{-1}\bar{b} - b)\big)\Big)\mathrm{tr}\big(\bar{a}^{-1}(a\bar{a}^{-1}\bar{b} - b)\big) = \quad (72)$$

$$= 2\mathrm{tr}\big(\bar{a}^{-1}\big(\frac{\bar{b}\bar{a}^{-1}a + a\bar{a}^{-1}\bar{b}}{2} - b\big)\big)\mathrm{tr}\big(\bar{a}^{-1}(a\bar{a}^{-1}\bar{b} - b)\big) = \quad (73)$$

$$= 2\mathrm{tr}\big(\bar{a}^{-1}\big(\frac{\bar{b}\bar{a}^{-1}a + a\bar{a}^{-1}\bar{b}}{2} - b\big)\big)^2. \quad (74)$$

Note that $a, \bar{a}, b, \bar{b}$ are symmetric matrices. Above, we used the property that for a symmetric matrix A, we have $\mathrm{tr}(A(B+B^T)) = 2\mathrm{tr}(AB)$. Next, we evaluate

$$\mathrm{tr}(F_0^TF_0 - I_3)(F_1 : F_1) = \quad (75)$$

$$= \mathrm{tr}(G^{-T}g^TgG^{-1} - I_3)\mathrm{tr}\big(G^{-T}\begin{bmatrix}(\bar{\ell} - \ell)^T & 0\\ 0 & 0\end{bmatrix}g^Tg\begin{bmatrix}\bar{\ell} - \ell & 0\\ 0 & 0\end{bmatrix}G^{-1}\big) = \quad (76)$$

$$= \mathrm{tr}\big(\bar{a}^{-1}(a - \bar{a})\big)\mathrm{tr}\big(\bar{a}^{-1}(q^Tg^{-T} - Q^TG^{-T})a(g^{-1}q - G^{-1}Q)\big) = \quad (77)$$

$$= \mathrm{tr}\big(\bar{a}^{-1}(a - \bar{a})\big)\mathrm{tr}\big(\bar{a}^{-1}\big(c - b\bar{a}^{-1}\bar{b} - \bar{b}\bar{a}^{-1}b + \bar{b}\bar{a}^{-1}a\bar{a}^{-1}\bar{b}\big)\big). \quad (78)$$

We can now finalize

$$\frac{1}{2}\big(\frac{dP}{dF} : F_1\big) : F_1 \overset{\lambda=1,\mu=0}{=\!=} \frac{1}{4}\big(\mathrm{tr}(F_1^TF_0) + \mathrm{tr}(F_0^TF_1)\big)\big(F_0 : F_1\big) +$$

$$+\frac{1}{4}\mathrm{tr}(F_0^TF_0 - I_3)(F_1 : F_1) = ||\bar{a}^{-1}\big(\frac{\bar{b}\bar{a}^{-1}a + a\bar{a}^{-1}\bar{b}}{2} - b\big)||^2_{SV} +$$

$$+\frac{1}{2} < \bar{a}^{-1}(a - \bar{a}),\ \bar{a}^{-1}\big(c - b\bar{a}^{-1}\bar{b} - \bar{b}\bar{a}^{-1}b + \bar{b}\bar{a}^{-1}a\bar{a}^{-1}\bar{b}\big) >_{SV} . \quad (79)$$

Case 2: $\lambda = 0, \mu = 1$

We first compute

$$F_0^TF_1 = G^{-T}g^T(qG^{-1} - gG^{-1}QG^{-1}) = \quad (80)$$

$$= -G^{-T}bG^{-1} - G^{-T}g^TgG^{-1}QG^{-1} = \quad (81)$$

$$= -G^{-T}\Big(\begin{bmatrix}b & 0\\ 0 & 0\end{bmatrix} - \begin{bmatrix}a & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\bar{a}^{-1}\bar{b} & 0\\ 0 & 0\end{bmatrix}\Big)G^{-1} = \quad (82)$$

$$= -G^{-T}\Big(\begin{bmatrix}b - a\bar{a}^{-1}\bar{b} & 0\\ 0 & 0\end{bmatrix}G^{-1} \quad (83)$$

Similarly,

$$F_1^TF_0 = -G^{-T}\Big(\begin{bmatrix}b - \bar{b}\bar{a}^{-1}a & 0\\ 0 & 0\end{bmatrix}G^{-1}. \quad (84)$$

Now, we can compute

$$\mathrm{tr}(F_0^TF_1F_0^TF_1) + \mathrm{tr}(F_1^TF_0F_0^TF_1) = \quad (85)$$

$$= \mathrm{tr}\big(\bar{a}^{-1}(a\bar{a}^{-1}\bar{b} - b)\bar{a}^{-1}(a\bar{a}^{-1}\bar{b} - b)\big) +$$

$$+\mathrm{tr}\big(\bar{a}^{-1}(\bar{b}\bar{a}^{-1}a - b)\bar{a}^{-1}(a\bar{a}^{-1}\bar{b} - b)\big) = \quad (86)$$

$$= 2\mathrm{tr}\Big(\bar{a}^{-1}\big(\frac{a\bar{a}^{-1}\bar{b} + \bar{b}\bar{a}^{-1}a}{2} - b\big)\bar{a}^{-1}\big(a\bar{a}^{-1}\bar{b} - b\big)\Big) = \quad (87)$$

$$= 2\mathrm{tr}\Big(\bar{a}^{-1}\big(\frac{a\bar{a}^{-1}\bar{b} + \bar{b}\bar{a}^{-1}a}{2} - b\big)\bar{a}^{-1}\big(\frac{a\bar{a}^{-1}\bar{b} + \bar{b}\bar{a}^{-1}a}{2} - b\big)\Big), \quad (88)$$

where the last equality holds because of

$$\mathrm{tr}\Big(A\frac{B+B^T}{2}A\frac{B+B^T}{2}\Big) = \tag{89}$$
$$= \frac{1}{4}\Big(\mathrm{tr}\big(A(B+B^T)AB\big)) + \mathrm{tr}\big(A(B+B^T)AB^T\big)\Big) = \tag{90}$$
$$= \frac{1}{4}\Big(\mathrm{tr}\big(A(B+B^T)AB\big)) + \mathrm{tr}\big(BA(B+B^T)A\big)\Big) = \tag{91}$$
$$= \frac{1}{4}\Big(\mathrm{tr}\big(A(B+B^T)AB\big)) + \mathrm{tr}\big(A(B+B^T)AB\big)\Big) = \tag{92}$$
$$= \mathrm{tr}\big(A\frac{B+B^T}{2}AB\big), \quad \text{for symmetric } A. \tag{93}$$

Next, we evaluate

$$\mathrm{tr}(F_0^TF_0F_1^TF_1) - \mathrm{tr}(F_1^TF_1) = \mathrm{tr}\big((F_0^TF_0 - I_3)F_1^TF_1\big) = \tag{94}$$
$$= \mathrm{tr}\Big((G^{-T}g^TgG^{-1} - I_3)G^{-T}\begin{bmatrix}(\bar\ell-\ell)^T & 0\\ 0 & 0\end{bmatrix}g^Tg\begin{bmatrix}\bar\ell-\ell & 0\\ 0 & 0\end{bmatrix}G^{-1}\Big) = \tag{95}$$
$$= \mathrm{tr}\Big(G^{-1}\big(G^{-T}aG^{-1} - GG^{-1}\big)G^{-T}\begin{bmatrix}(\bar\ell-\ell)^T & 0\\ 0 & 0\end{bmatrix}a\begin{bmatrix}\bar\ell-\ell & 0\\ 0 & 0\end{bmatrix}\Big) = \tag{96}$$
$$= \mathrm{tr}\Big(\big(G^{-1}G^{-T}a - G^{-1}G\big)G^{-1}G^{-T}\begin{bmatrix}(\bar\ell-\ell)^T & 0\\ 0 & 0\end{bmatrix}a\begin{bmatrix}\bar\ell-\ell & 0\\ 0 & 0\end{bmatrix}\Big) = \tag{97}$$
$$= \mathrm{tr}\Big(\big(\bar a^{-1}a - I_3\big)\bar a^{-1}\begin{bmatrix}(\bar\ell-\ell)^T & 0\\ 0 & 0\end{bmatrix}a\begin{bmatrix}\bar\ell-\ell & 0\\ 0 & 0\end{bmatrix}\Big) = \tag{98}$$
$$= \mathrm{tr}\Big(\Big(\bar a^{-1}(a-\bar a)\Big)\Big(\bar a^{-1}\big(c - b\bar a^{-1}\bar b - \bar b\bar a^{-1}b + \bar b\bar a^{-1}a\bar a^{-1}\bar b\big)\Big)\Big). \tag{99}$$

We can now finalize

$$\frac{1}{2}\big(\frac{dP}{dF} : F_1\big) : F_1 \overset{\lambda=0,\,\mu=1}{=\!=} \frac{1}{2}\Big(\mathrm{tr}(F_0^TF_1F_0^TF_1) + \mathrm{tr}(F_1^TF_0F_0^TF_1) + \\ + \mathrm{tr}(F_0^TF_0F_1^TF_1) - \mathrm{tr}(F_1^TF_1)\Big) = \tag{100}$$
$$= ||\bar a^{-1}\big(\frac{a\bar a^{-1}\bar b + \bar b\bar a^{-1}a}{2} - b\big)||^2_{SV} + \\ + \frac{1}{2} < \bar a^{-1}(a-\bar a),\ \bar a^{-1}\big(c - b\bar a^{-1}\bar b - \bar b\bar a^{-1}b + \bar b\bar a^{-1}a\bar a^{-1}\bar b\big) >_{SV} . \tag{101}$$

F.4 Proving Equation 32

Case 1: $\lambda = 1, \mu = 0$

$$P : (F_2 - 2HF_1) \overset{\lambda=1,\,\mu=0}{=\!=} \frac{1}{2}\big(\mathrm{tr}(F_0^TF_0 - I_3)\big)F_0 : \big(F_2 - 2HF_1\big) = \tag{102}$$
$$= \frac{1}{2}\Big(\mathrm{tr}(G^{-T}g^TgG^{-1} - I_3)\Big)gG^{-1} : \Big(g\begin{bmatrix}(\bar\ell-\ell)(\bar\ell-2HI_2) & 0\\ 0 & 0\end{bmatrix}G^{-1}\Big) = \tag{103}$$
$$= \frac{1}{2}\Big(\mathrm{tr}(\bar a^{-1}(a-\bar a))\Big)\mathrm{tr}\Big(gG^{-1}G^{-T}\begin{bmatrix}(\bar\ell-2HI_2)^T(\bar\ell-\ell)^T & 0\\ 0 & 0\end{bmatrix}g^T\Big) = \tag{104}$$
$$= \frac{1}{2}\Big(\mathrm{tr}(\bar a^{-1}(a-\bar a))\Big)\mathrm{tr}\Big(\bar a^{-1}(\bar b\bar a^{-1} - 2HI_2)(\bar b\bar a^{-1} - ba^{-1})a\Big) = \tag{105}$$
$$=< \bar a^{-1}(a-\bar a),\ \bar a^{-1}(\bar b\bar a^{-1} - 2HI_2)(\bar b\bar a^{-1}a - b) >_{SV} . \tag{106}$$

Case 2: $\lambda = 0, \mu = 1$

$$P : (F_2 - 2HF_1) \overset{\lambda=0,\,\mu=1}{=\!=} F_0(F_0^TF_0 - I_3) : (F_2 - 2HF_1) = \tag{107}$$
$$= gG^{-1}(G^{-T}g^TgG^{-1} - I_3) : \Big(g\begin{bmatrix}(\bar\ell-\ell)(\bar\ell-2HI_2) & 0\\ 0 & 0\end{bmatrix}G^{-1}\Big) = \tag{108}$$
$$= \mathrm{tr}\Big(gG^{-1}(G^{-T}g^TgG^{-1} - I_3)G^{-T}\begin{bmatrix}(\bar\ell-2HI_2)^T(\bar\ell-\ell)^T & 0\\ 0 & 0\end{bmatrix}g^T\Big) = \tag{109}$$
$$= \mathrm{tr}\Big(G^{-1}(G^{-T}g^TgG^{-1} - I_3)G^{-T}\begin{bmatrix}(\bar\ell-2HI_2)^T(\bar\ell-\ell)^T & 0\\ 0 & 0\end{bmatrix}g^Tg\Big) = \tag{110}$$
$$= \mathrm{tr}\Big(G^{-1}(G^{-T}g^Tg - G)G^{-1}G^{-T}\begin{bmatrix}(\bar\ell-2HI_2)^T(\bar\ell-\ell)^T & 0\\ 0 & 0\end{bmatrix}g^Tg\Big) = \tag{111}$$
$$= \mathrm{tr}\Big(\bar a^{-1}(a-\bar a)\bar a^{-1}\big(\bar b\bar a^{-1} - 2HI_2\big)\big(\bar b\bar a^{-1}a - b\big)\Big) = \tag{112}$$
$$=< \bar a^{-1}(a-\bar a),\ \bar a^{-1}\big(\bar b\bar a^{-1} - 2HI_2\big)\big(\bar b\bar a^{-1}a - b\big) >_{SV} . \tag{113}$$