

A Constraint-based Formulation of Stable Neo-Hookean Materials

Miles Macklin
NVIDIA
Auckland, New Zealand

Matthias Müller
NVIDIA
Zurich, Switzerland

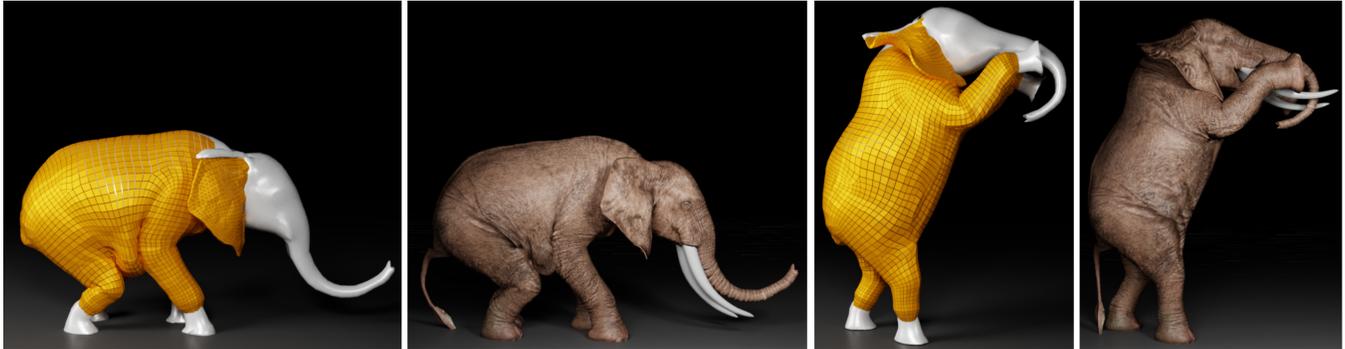


Figure 1: Large scale simulation. Our method takes 8ms per frame on a modern GPU to simulate the tissue layer of an elephant composed of 80k tetrahedra including self collision handling and skinning. It can be implemented in a few lines of code and is robust for a Poisson ratio of 0.5 as well as under large compression and stretch.

ABSTRACT

In computer graphics, soft body simulation is often used to animate soft tissue on characters or rubber like objects. Both are highly incompressible, however commonly used models such as co-rotational FEM, show significant volume loss, even under moderate strain. The Neo-Hookean model has recently become popular in graphics. It has superior volume conservation, recovers from inverted states, and does not require a polar decomposition. However, solvers for Neo-Hookean finite-element problems are typically based on Newton methods, which require energy Hessians, their Eigen-decomposition, and sophisticated linear solvers. In addition, minimizing the energy directly in this way does not accommodate modeling incompressible materials since it would require infinitely stiff forces. In this paper we present a constraint-based model of the Neo-Hookean energy. By decomposing the energy into deviatoric (distortional), and hydrostatic (volume preserving) constraints, we can apply iterative constrained-optimization methods that require only first-order gradients. We compare our constraint-based formulation to state-of-the-art force-based solvers and show that our method is often an order of magnitude more efficient for stiff volume preserving materials.

CCS CONCEPTS

• Computing methodologies → Physical simulation.

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

MIG'21, November 10–12, 2021, Lausanne, Switzerland

© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00
<https://doi.org/10.1145/1122445.1122456>

KEYWORDS

finite element method, physically-based animation, elasticity, real-time physics

ACM Reference Format:

Miles Macklin and Matthias Müller. 2021. A Constraint-based Formulation of Stable Neo-Hookean Materials. In *MIG'21: Motion, Interaction and Games*. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/1122445.1122456>

1 INTRODUCTION

An important use case for soft body simulation in computer graphics as well as medical applications is the simulation of soft tissue such as muscle, fat and skin layers on characters. Other examples are rubber-like materials to simulate tires or soft grippers. Glzman and Azhari measured the Poisson ratio – a measure for volume conservation – of porcine fat tissue, turkey breast tissue and bovine liver tissue and found that they are all within the range of 0.49999 ± 0.00001 , where a fully volume conserving material has a Poisson ratio of 0.5 [2010]. Their findings show that these materials have two important characteristics: they are incompressible to a very high degree and show a close to linear stress-strain curve.

The co-rotational model [Müller et al. 2002] is common in graphics, and is based on a linear stress-strain relation. However they are not volume preserving, and quickly lose volume under strain as shown in Figure (2). In addition, it requires a polar decomposition of the deformation gradient that brings its own challenges. A substantial body of work exists on how to perform the decomposition robustly and how to handle volume inversions [Kugelstadt et al. 2018; McAdams et al. 2011; Müller et al. 2016].

More recently, the Neo-Hookean model has become popular in graphics [Smith et al. 2018]. It has two separate energy terms, a hydrostatic energy and a deviatoric energy. While the hydrostatic term resists compression and expansion, the deviatoric term resists the distortion of the object. The hydrostatic term is sensitive to

volume inversions and resolves them automatically. Both terms are rotation invariant and therefore, no polar decomposition is required. However, incompressibility poses a problem for force-based methods, since for a material to be perfectly incompressible, its bulk modulus, which quantifies its resistance to compression, must be infinitely large. In essence, incompressibility is a hard constraint that requires infinite stiffness when modeled with a penalty term.

In offline computer graphics these stiff energies are typically minimized through sophisticated methods such as the projected Newton method [Baraff and Witkin 1998; Kim and Eberle 2020]. Projected Newton in this context refers the linearization of the system's optimality conditions, and projection of the energy Hessian onto the positive-definite cone, and then solve the resulting linear system with methods such as preconditioned conjugate-gradient (PCG). As the stiffness of the material increases, the conditioning of the system becomes worse, and many linear solver iterations are required to reach convergence.

To avoid these problems, we formulate the Neo-Hookean energy as a pair of compliant constraint functions [Tournier et al. 2015]. This approach has the benefit that, as stiffness increases, the system conditioning is unaffected. In the limit of infinite stiffness, the constraint has zero-compliance, and incompressibility is enforced through a Lagrange-multiplier.

Our constraint-based model may be solved with any commercially available software for constrained optimization [Johnson 2014], however, we present an implementation inside the extended position-based dynamics (XPBD) framework [Macklin et al. 2016] which uses local Gauss-Seidel iterations to enforce constraints. This approach requires only first-order gradients, is robust to large displacements, and is suitable for real-time applications.

To summarize, we propose a formulation of the stable Neo-Hookean constitutive model as a pair of compliant constraint functions. By adopting this model we can apply constrained-optimization solvers to handle arbitrarily stiff materials. We perform a convergence comparison between Newton-based solvers and the XPBD constrained-dynamics solver. Finally, we include straightforward implementation details and source-code for our solver.

2 RELATED WORK

There is a large body of work on the simulation of deformable objects in computer graphics going back to the work of Terzopoulos et al. [1987]. They used the squared norm of Green's deformation tensor as the elastic energy and finite differences to solve the equations. Their deformation energy corresponds to a Hookean model with a Poisson ratio of zero meaning there is no way to control volume conservation.

Neo-Hookean Models. Smith et al. [2018] drew our attention to Neo-Hookean material models. These models have the desirable property of not requiring a polar decomposition. However, in their original form, Neo-Hookean models are not *rest stable*, meaning they generate forces in their undeformed configuration. Smith et al. overcome this problem through a re-parameterization of the Lamé parameters. Many variations of Neo-Hookean exist, and similar models have been proposed that combine a St. Venant-Kirchhoff model with a penalty-based incompressibility term [Kozlov et al.

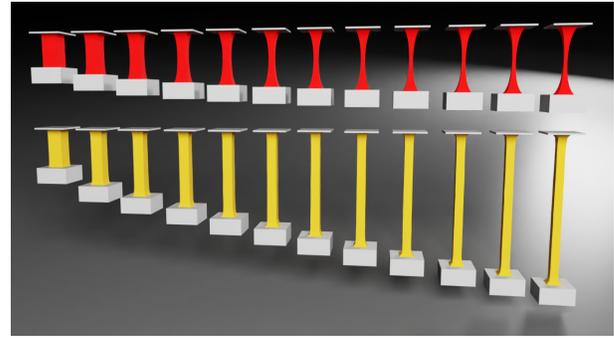


Figure 2: Volume Preservation. Varying the density of the attached boxes shows the nonlinear stress-strain relation and volume loss of Hookean model using Green strain (top). Neo-Hookean models exhibit a close to linear relationship and a high degree of volume conservation (bottom).

[2017; Picinbono et al. 2003]. For a comprehensive and enjoyable summary of Neo-Hookean models, we refer the reader to the recently published SIGGRAPH course notes of Kim and Eberle about dynamic deformables [2020].

Hard volume conservation. An important property of our model is support for incompressible materials. We achieve this by treating the hydrostatic term as a hard holonomic constraint. Outside of graphics this approach is commonly referred to as *Mixed FEM* [Zienkiewicz et al. 2005], and may be thought of as imposing a divergence free condition on an embedded velocity field. This approach is relatively uncommon in graphics, with a few exceptions. Servin et al. [2006] proposed a compliant form of elasticity that uses $\mathbb{R} \in 6 \times 6$ compliance matrices for a St. Venant Kirchhoff constitutive model. This approach adds six Lagrange multiplier variables to the system for each tetrahedral element. In contrast, our approach requires only two additional variables, and supports nonlinear Neo-Hookean materials and inverted configurations. Francu et al. [2019] also proposed a mixed FEM model, however, they base their method on the co-rotational model which requires an additional polar decomposition step. Tournier et al. [2015] proposed a compliant constrained dynamics framework with support for continuum materials, and a geometric stiffness term to stabilize a linearly implicit time-integration. Our proposed constitutive model is compatible with this framework, however we use robust nonlinear iterative methods to achieve stable simulation without explicitly constructing energy Hessians. Irving et al. [Irving et al. 2007] proposed a volume conserving finite element method that uses velocity projections to ensure divergence free conditions with a rotated linear model. Hong et al. extended a mass-spring model with volume conservation scheme [Hong et al. 2006], in this work we focus on tetrahedral elements to define our model. Francu et al. [Francu et al. 2021] present a locking free approach to stiff volume preserving materials, they enforce volume conservation through a similar constraint-based formulation, but treat the deviatoric energy in a traditional backward Euler solver. In contrast, we formulate both energies as compliant constraints, allowing us to apply fast iterative descent methods. We note that treating both energies as *hard*

constraints would lead to element locking, essentially making the material a rigid. Since we treat energies as compliant constraints we avoid the locking problem [2018].

Position-based Methods. In their original XPBD paper, Macklin et al. [2016] propose a method to simulate continuous co-rotational materials with one constraint per-strain component. Besides the need to compute a polar decomposition, performing 6-constraint projections per-tetrahedron is significantly more expensive when compared to the two explicit per-element constraints we have to handle. Bender et al. [2014] formulated Hookean elasticity discretized with FEM as position based constraints with the singularity at $\mu = 0.5$. Also, they have to handle inverted elements explicitly for which they use the method of Irving et al. [2004]. Since they use PBD instead of XPBD, their resulting stiffness is iteration count and time step dependent. Projective Dynamics [Bouaziz et al. 2014] is an alternative to position based dynamics and has been extended to support hyperelastic materials [Liu et al. 2017]. However, part of the solver is global and involves a linearization and approximation of energy Hessians which introduces complexity. In addition, as it is typically force based, it requires special handling for hard constraints.

3 METHOD

Before we present our constraint-based formulation we recap the energy-based definition of the Neo-Hookean model.

3.1 Energy Based Model

There are many Neo-Hookean models. We use the simplest one with the elastic energy density

$$\begin{aligned}\Psi_{\text{Neo}} &= \frac{\lambda}{2} (\det(\mathbf{F}) - 1)^2 + \frac{\mu}{2} (\text{tr}(\mathbf{F}^T \mathbf{F}) - 3) \\ &= \Psi_{\text{H}} + \Psi_{\text{D}},\end{aligned}\quad (1)$$

where the 3×3 matrix \mathbf{F} is the deformation gradient and λ and μ the Lamé parameters. It has two separate energy terms, a hydrostatic energy Ψ_{H} resisting compression and expansion and the deviatoric energy Ψ_{D} resisting distortion. The two terms have intuitive interpretations. The columns of

$$\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3] \quad (3)$$

are the axes of the deformed coordinate system at the transformed location. The determinant $\det(\mathbf{F})$ yields the volume of the parallelepiped spanned by the deformed axes. This volume is 1 iff the transformation is locally volume preserving which is enforced by the hydrostatic energy. The trace of the deformation tensor can be expressed in terms of the deformed axes as

$$\text{tr}(\mathbf{F}^T \mathbf{F}) = |\mathbf{f}_1|^2 + |\mathbf{f}_2|^2 + |\mathbf{f}_3|^2 \quad (4)$$

and is therefore equal to the sum of their squared lengths. If the body is not deformed, all axes have unit length so their sum is 3 which makes the Ψ_{D} zero.

Both $\det(\mathbf{F})$ and $\text{tr}(\mathbf{F}^T \mathbf{F})$ are invariants of the deformation gradient, which means they do not change when the body is rotated or translated. This is the reason why our method does not require performing a polar decomposition. We will now carry these concepts over to the constrained-dynamics framework.

3.2 Constraint Formulation

In constrained-dynamics frameworks [Macklin et al. 2016; Servin et al. 2006; Tournier et al. 2015], a constraint function C is turned into an energy via.

$$\Psi_{\text{C}} = \frac{1}{2} \alpha^{-1} C(\mathbf{x})^2 \quad (5)$$

resulting in the force

$$\mathbf{f}_{\text{C}} = -\alpha^{-1} C(\mathbf{x}) \frac{\partial C(\mathbf{x})}{\partial \mathbf{x}}^T, \quad (6)$$

where α is the compliance (inverse stiffness) of the constraint. For hard-constraints $\alpha = 0$, and the energy in (5) is not well-defined, but the constraint is enforced via a Lagrange multiplier λ .

3.2.1 Hydrostatic Constraint. Given the definition of the energy associated with a compliant constraint in (5), we can write our hydrostatic energy Ψ_{H} in terms of the following constraint function:

$$C_{\text{H}}(\mathbf{F}) = \det(\mathbf{F}) - 1. \quad (7)$$

When the associated compliance parameter α is zero the constraint is imposed as a hard equality, and the material is considered incompressible from a modeling perspective. We will evaluate methods for enforcing this incompressibility condition in Section 4. We note also that \mathbf{F} is implicitly a function of \mathbf{x} , the particle positions. Please see the supplementary material for full constraint gradient derivations in terms of particle degrees of freedom.

3.2.2 Deviatoric Constraint. The question now is how to formulate the deviatoric energy Ψ_{D} as a compliant constraint. Looking at Figure 3 we can see that the Neo-Hookean deviatoric energy can actually become negative in the compressed regime. This poses a problem since compliant constraints can generally only represent non-negative energies. To address this we propose a simple fix that provides a *force-equivalent* energy. We do this by simply shifting the Neo-Hookean energy vertically to the positive half-space. Because forces arise from the gradient of energy, i.e.: $\mathbf{f} = -\nabla \Psi$, the resulting forces from this translated energy are unchanged. The constraint function that gives rise to this shifted energy is,

$$C_{\text{D}}(\mathbf{F}) = \sqrt{\text{tr}(\mathbf{F}^T \mathbf{F})}. \quad (8)$$

Inserting this constraint definition into (5) we can see that it corresponds to the following energy:

$$\begin{aligned}\Psi_{\text{D}} &= \frac{1}{2} \alpha^{-1} C_{\text{D}}(\mathbf{F})^2 \\ &= \frac{\mu}{2} \text{tr}(\mathbf{F}^T \mathbf{F}),\end{aligned}\quad (9)$$

this relation holds when $\alpha = \frac{1}{\mu}$, and is equivalent to the original deviatoric energy shifted by a constant factor of $\frac{3\mu}{2}$.

The last question we have to answer before simulation is how the compliance parameter α of is related to μ . An important observation is that Ψ_{D} represents an energy *density*. To turn Ψ_{D} into an energy potential U , we have to integrate it over the tetrahedron. Since we use linear elements \mathbf{F} is constant within the element, we use a simple lumped mass model, so integration amounts to a multiplication with the tetrahedron's volume V_{tet} , i.e.: $U_{\text{D}} = V_{\text{tet}} \Psi_{\text{D}}$. We incorporate

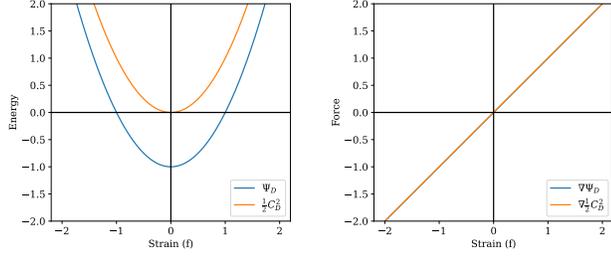


Figure 3: Left: The deviatoric energy corresponding to the original Neo-Hookean (blue) and constraint-based formulation (orange). Our constraint-based formulation requires a simple shift to the positive half-space. **Right:** The resulting forces, which are linear in strain, are equivalent for both formulations.

this volumetric scaling factor by solving the following for equation for α ,

$$\frac{1}{2}\alpha^{-1}C_D^2 = \frac{\mu}{2}V_{tet}C_D^2 \quad (11)$$

$$\alpha = \frac{1}{\mu V_{tet}}. \quad (12)$$

Similarly, when using a finite volumetric stiffness, we use the same derivation to set the volumetric compliance as:

$$\alpha = \frac{1}{\lambda V_{tet}}. \quad (13)$$

Our method does not require the computation of stress. However, both the hydrostatic as well as the deviatoric stress tensors can easily be derived from the Lagrange multipliers used to solve our constraints as we show in the supplemental material.

3.3 Rest Stability

The Neo-Hookean energy Ψ_{Neo} presented at the start of this section is not *rest-stable*. This means that there are some non-zero forces generated even in the undeformed configuration. This problem was identified and addressed by Smith et al. [Smith et al. 2018], who proposed a modification of the hydrostatic term as follows,

$$\Psi_H = \frac{\lambda}{2} (\det(\mathbf{F}) - \gamma)^2, \quad (14)$$

where $\gamma = 1 + \frac{\mu}{\lambda}$. This modification ensures that hydrostatic and deviatoric forces balance each other in the rest state. Our constraint-based formulation permits the same modification by re-defining $C_H = \det(\mathbf{F}) - \gamma$. We note that this is also well behaved in the limit as λ goes to infinity, as is the case for incompressible materials.

4 SIMULATION

For simulation we start with a tetrahedral mesh as input. Given this, we create one particle for each vertex. Each tetrahedron adds one fourth of its mass to each adjacent particle, and contributes two constraints to the system, which are solved using the method described below.

Algorithm 1: XPBD solver

```

while simulating do
  CollectCollidingPairs();
  h ← Δt/numSubsteps;
  for numSubsteps do
    for n particles do
      x_prev ← x;
      v ← v + h f_ext/m;
      x ← x + h v;
    end
    for all constraints c do
      λ = 0
    end
    for numPosIters do
      for all constraints c do
        Δλ =  $\frac{-C(\mathbf{x}) - \tilde{\alpha}\lambda}{\sum_{i=1}^n w_i |\nabla_{\mathbf{x}_i} C(\mathbf{x})|^2 + \tilde{\alpha}}$ ;
        Δx = M-1∇C(x)TΔλ
        λ ← λ + Δλ
        x ← x + Δx
      end
    end
    for n particles do
      v ← (x - x_prev)/h;
    end
  end
end

```

4.1 The XPBD Solver

The XPBD method first performs an explicit Euler step on each particle, then projects the constraints one-by-one Gauss-Seidel style over multiple iterations.

A constraint is projected by computing a positional correction vector $\Delta\mathbf{x}$ and applying it to the particle positions \mathbf{x} . The correction vector is chosen to point along the gradient $\nabla C(\mathbf{x})$ of the constraint function which is the direction of maximal change (using the convention that ∇ gives the function gradient as a row-vector). Therefore, the position change has the form $\Delta\mathbf{x} = \mathbf{M}^{-1}\nabla C(\mathbf{x})^T \Delta\lambda$, where the scalar Lagrange multiplier $\Delta\lambda$ tells us how far to move along the inverse mass-weighted gradient to find a position where the constraint function is zero. Note that in this section λ refers to the Lagrange multiplier rather than the Lamè parameter. The multiplier can be computed by solving

$$C(\mathbf{x} + \Delta\mathbf{x}) \approx C(\mathbf{x}) + \nabla C(\mathbf{x}) \cdot \Delta\mathbf{x} = 0 \quad (15)$$

with the explicit solution

$$\Delta\lambda = \frac{-C(\mathbf{x})}{\nabla C(\mathbf{x})\mathbf{M}^{-1}\nabla C(\mathbf{x})^T}, \quad (16)$$

where the matrix \mathbf{M} is diagonal and contains the masses of the particles. These equations use the concatenated coordinates of all adjacent particles. Expanding the denominator using individual particles, they take the form

$$\Delta\lambda = \frac{-C(\mathbf{x})}{\sum_{i=1}^n w_i |\nabla_{\mathbf{x}_i} C(\mathbf{x})|^2} \quad (17)$$

Table 1: Performance Timings for Stretch Test. We compare performance of our method to a Newton method using the open-source implementation from Smith et al. [Smith et al. 2018] with default values. For our method we have used a single-threaded CPU implementation with a fixed number of constraint iterations (64) for each problem. The convergence of force-based solvers depends heavily on the volumetric stiffness (defined by the Poisson ratio), while constraint-based formulations are less sensitive to this. Timings with a * superscript indicate that the solver failed to converge.

| Poisson (ν) | Newton | | Ours | |
|-------------------|----------|---------|----------|------|
| | Vol. Err | Time | Vol. Err | Time |
| 0.41 | 1.24 | 1m | 1.25 | 28s |
| 0.45 | 1.13 | 1m20s | 1.13 | 28s |
| 0.49 | 1.02 | 3m20s* | 1.02 | 28s |
| 0.495 | 1.03 | 7m10s* | 1.01 | 28s |
| 0.4995 | 1.001 | 11m50s* | 1.001 | 28s |

and

$$\mathbf{x}_i \leftarrow \mathbf{x}_i + w_i \nabla_{\mathbf{x}_i} C(\mathbf{x})^T \Delta \lambda, \quad (18)$$

where n is the number of adjacent particles, w_i is the inverse mass of particle i and $\nabla_{\mathbf{x}_i} C(\mathbf{x})$ is the gradient of the constraint function with respect to the position of particle i .

The projection described above can only solve hard constraints. In the original PBD approach soft constraints are handled by simply scaling the updates $\Delta \mathbf{x}$ with a scalar between zero and one which yields a stiffness that is time step size and iteration count dependent. Fortunately, this problem was fixed in XPBD [Macklin et al. 2016] with a small modification. The modified version of Equation (17) is

$$\Delta \lambda = \frac{-C(\mathbf{x}) - \tilde{\alpha} \lambda}{\sum_{i=1}^n w_i |\nabla_{\mathbf{x}_i} C(\mathbf{x})|^2 + \tilde{\alpha}}, \quad (19)$$

where $\tilde{\alpha} = \alpha/h^2$, h the substep size and α the compliance we discussed in the previous section. Each constraint stores a scalar λ which is set to zero at the beginning of each substep and updated along with the positions as

$$\mathbf{x}_i \leftarrow \mathbf{x}_i + w_i \nabla_{\mathbf{x}_i} C(\mathbf{x})^T \Delta \lambda \quad (20a)$$

$$\lambda \leftarrow \lambda + \Delta \lambda. \quad (20b)$$

The projection can handle infinite stiffness by simply setting $\alpha = 0$ in which case the PBD projection formula is recovered. The full solver algorithm is described in Algorithm 1. We use substepping as suggested by Macklin et al. [2019] to speed up convergence and resolve high frequency detail. However, collision pairs are only detected once per-frame to not slow down the solver when iterations are replaced by substeps.

5 RESULTS

We now evaluate our method for efficiency, accuracy, and robustness.

5.1 Performance

To measure the performance of our method we compare to the open source Newton solver provided by Smith et al. [Smith et al. 2018]. To make a fair comparison we create a single-threaded CPU

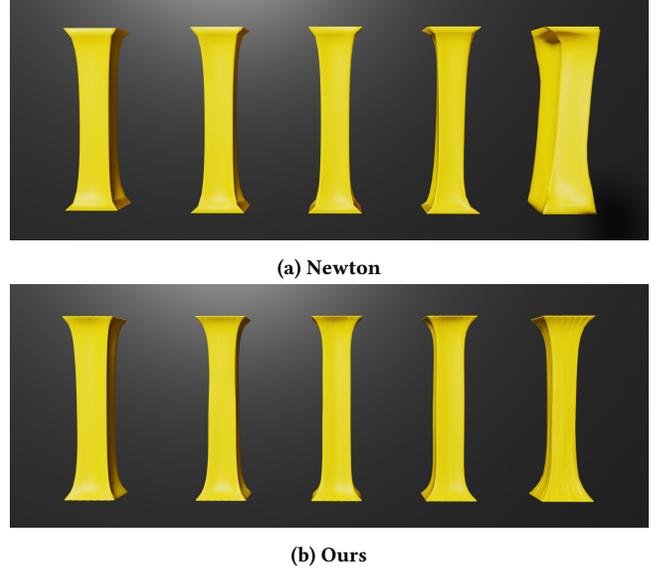


Figure 4: Stretch Test. We compare the results of our constraint-based solver to the Newton solver of Smith et al. [Smith et al. 2018] with Poisson ratio increasing from $[0.4, \dots, 0.4995]$ left to right respectively. The Newton-based method fails to converge for high volumetric stiffness resulting in significant artifacts even after 20x longer computation times. Please see Table 1 for detailed performance numbers.

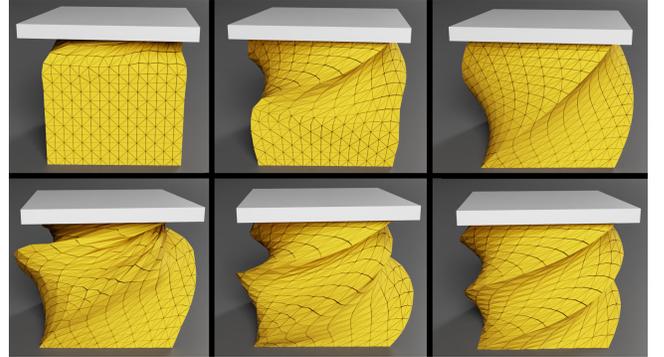


Figure 5: Cube Twist. Two consecutive 90° twists each solved statically. The second, fourth and fifth images show states within a solve.

implementation of our method and run both methods on an AMD Ryzen 5-5600X with 32GB of RAM. We run the Stretch Test shown in Figure 4 with Poisson ratios between $[0.4, \dots, 0.4995]$. We use default solver tolerances for the Newton method and a fixed 64 iterations for our method. In this test we found that Newton would fail to converge for high volumetric stiffness, even when the computational budget was more than 20x larger than our constraint-based formulation.

The twisted experiment shown in Figure 5 is a replication of the test presented in [Smith et al. 2018]. Here a cube with 15^3 hex

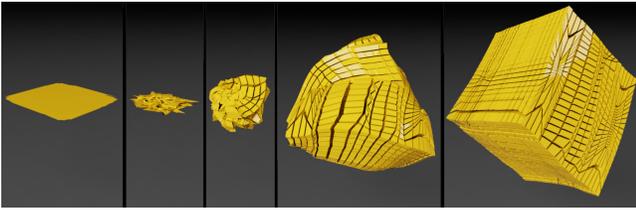


Figure 6: Crazy Cube Scramble. Unfolding the crazy cube from a configuration in which the vertex positions are randomized within a plane, the cube has a Poisson ratio of 0.5.

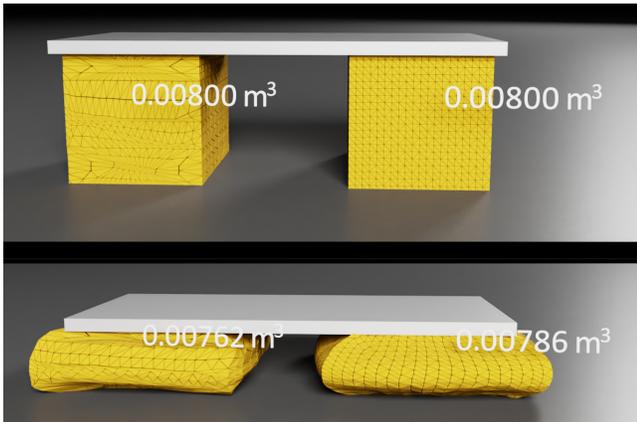


Figure 7: Crazy Cube Compress. The crazy and a regularly tessellated cube are stretched by a factor of 4.5 and compressed by a factor of 3. This experiment demonstrates the stability of our method. The volume gain in the fully stretched state is below 3 percent.

cells is twisted in two steps of 90 degrees. After each turn a static solve is performed. In each case it converges to the correct shape in 400 Gauss-Seidel iterations, while Smith et al. report 3k and 5k Conjugate Gradient iterations, making our method about ten times faster assuming CG and GS iterations have similar costs. We further extend their test to a full 360 degree twist, while this causes element collapse our method remains stable.

5.2 Robustness

To demonstrate the robustness of our approach, we created the crazy cube shown in Figure 6. It is tessellated into 40k tetrahedra. While the smallest angle at a tetrahedral face is below 0.3 degrees, the largest edge ratio is 67:1. We use a Poisson ratio of 0.5. Figure 6 shows how it unfolds from a state in which the vertex positions are randomized within a plane. Our method solves this problem in a fast and robust way.

We further test the robustness of our method by collapsing the dragon model in Figure 8 to a plane, and randomizing the vertex positions inside the plane. Again, our method recovers the original shape in a fast and stable way.

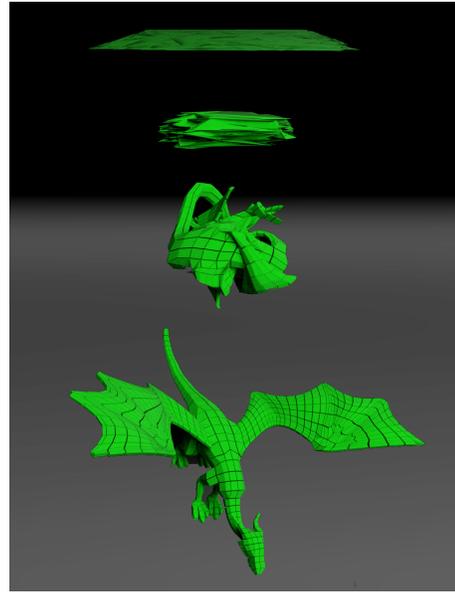


Figure 8: Dragon Unfold. The dragon unfolds from a randomized configuration in which the vertices have random positions in a plane.

5.3 Parallelism

For the elephant scene shown in Figure 1 we parallelized the method using CUDA. Again, we use a Poisson ratio of 0.5. To parallelize the Gauss-Seidel iteration, we split the constraints into independent sets. In a tetrahedral mesh, the number of elements adjacent to a particle is a lower bound on the number of sets, and can be high for non-uniform meshes. To avoid this we derive our tetrahedral mesh from a hexahedral mesh by subdividing each cell into 5 or 6 tetrahedra. If a thread handles an entire cell and the hexahedral mesh is regular, 8 sets are sufficient.

On an NVIDIA GeForce RTX 2080Ti GPU the simulation of the tissue layer composed of 80k tetrahedra including self-collision handling and skinning of the surface mesh take 8ms per frame. As the accompanying video shows, our method yields realistic secondary motion of the ears and skin during a walk cycle and is robust under large compression and stretch.

5.4 Implementation

To demonstrate the simplicity of our method we created an interactive demo in a single small HTML file containing a Javascript implementation of our method, a GUI, and the dragon model. It runs in any browser. The solver takes 100 lines of code. We provide the HTML file as supplementary material. While the dragon can be dragged with the mouse and squeezed onto a plane it remains stable.

6 CONCLUSION AND FUTURE WORK

We have presented a novel method to simulate incompressible Neo-Hookean soft bodies in a simple and robust way. The main idea is to use two positional constraints, one for volume conservation and a

shifted energy to control distortions. Unlike Newton-based methods our solver does not require energy Hessians (or their projection), making it significantly simpler to implement. In addition, using a compliance-based formulation lets us simulate volume conserving materials that are stable at the limit, when the Poisson ratio is 0.5.

As with any iterative numerical method, the solution accuracy is dependent on the computational budget. However we have demonstrated that constraint-based formulations are less sensitive to volumetric stiffness and can out perform Newton methods when stiffness is high.

Currently we can only simulate elastic materials that return to their rest state independently of the amount of stretch. However, past a certain point, tissue starts to deform plastically and eventually fails. There is a large body of work on plastic deformations and ductile fracture and we expect that these ideas can be integrated into our model in a straightforward way.

REFERENCES

- David Baraff and Andrew Witkin. 1998. Large steps in cloth simulation. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. 43–54.
- Jan Bender, Dan Koschier, Patrick Charrier, and Daniel Weber. 2014. Position-Based Simulation of Continuous Materials. *Computers & Graphics* 44, 0 (2014), 1–10. <https://doi.org/10.1016/j.cag.2014.07.004>
- Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. 2014. Projective Dynamics: Fusing Constraint Projections for Fast Simulation. *ACM Trans. Graph.* 33, 4, Article 154 (July 2014), 11 pages. <https://doi.org/10.1145/2601097.2601116>
- Mihai Francu, Arni Asgeirsson, and Kenny Erleben. 2019. High Fidelity Simulation of Corotational Linear FEM for Incompressible Materials. In *Motion, Interaction and Games* (Newcastle upon Tyne, United Kingdom) (MIG '19). Association for Computing Machinery, New York, NY, USA, Article 28, 6 pages. <https://doi.org/10.1145/3359566.3360079>
- Mihai Francu, Arni Asgeirsson, Kenny Erleben, and Mads J. L. Rønnow. 2021. Locking-Proof Tetrahedra. *ACM Trans. Graph.* 40, 2, Article 12 (April 2021), 17 pages. <https://doi.org/10.1145/3444949>
- Tanya Glazman and Haim Azhari. 2010. A method for characterisation of tissue elastic properties combining ultrasonic computed tomography with elastography. *J. Ultrasound. Journal of Ultrasound in Medicine* 29 (2010), 387–398.
- Min Hong, Sunhwa Jung, Min-Hyung Choi, and Samuel W.J. Welch. 2006. Fast Volume Preservation for a Mass-Spring System. *IEEE Computer Graphics and Applications* 26, 5 (2006), 83–91. <https://doi.org/10.1109/MCG.2006.104>
- Geoffrey Irving, Craig Schroeder, and Ronald Fedkiw. 2007. Volume Conserving Finite Element Simulations of Deformable Models. *ACM Trans. Graph.* 26, 3 (July 2007), 13–es. <https://doi.org/10.1145/1276377.1276394>
- G. Irving, J. Teran, and R. Fedkiw. 2004. Invertible Finite Elements for Robust Simulation of Large Deformation. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Grenoble, France) (SCA '04). Eurographics Association, Goslar, DEU, 131–140. <https://doi.org/10.1145/1028523.1028541>
- Steven G Johnson. 2014. The NLOpt nonlinear-optimization package.
- Theodore Kim and David Eberle. 2020. Dynamic Deformables: Implementation and Production Practicalities. In *ACM SIGGRAPH 2020 Courses* (Virtual Event, USA) (SIGGRAPH 2020). Association for Computing Machinery, New York, NY, USA, Article 23, 182 pages. <https://doi.org/10.1145/3388769.3407490>
- Yeara Kozlov, Derek Bradley, Moritz Bächer, Bernhard Thomaszewski, Thabo Beeler, and Markus Gross. 2017. Enriching Facial Blendshape Rigs with Physical Simulation. *Comput. Graph. Forum* 36, 2 (May 2017), 75–84.
- Tassilo Kugelstadt, Dan Koschier, and Jan Bender. 2018. Fast Corotated FEM using Operator Splitting. *Comput. Graph. Forum* 37, 8 (2018), 149–160. <https://doi.org/10.1111/cgf.13520>
- Tiantian Liu, Sofien Bouaziz, and Ladislav Kavan. 2017. Quasi-Newton Methods for Real-Time Simulation of Hyperelastic Materials. *ACM Trans. Graph.* 36, 4, Article 116a (May 2017), 16 pages. <https://doi.org/10.1145/3072959.2990496>
- Miles Macklin, Matthias Müller, and Nuttapon Chentanez. 2016. XPBD: Position-Based Simulation of Compliant Constrained Dynamics. In *Proceedings of the 9th International Conference on Motion in Games* (Burlingame, California) (MIG '16). Association for Computing Machinery, New York, NY, USA, 49–54. <https://doi.org/10.1145/2994258.2994272>
- Miles Macklin, Kier Storey, Michelle Lu, Pierre Terdiman, Nuttapon Chentanez, Stefan Jeschke, and Matthias Müller. 2019. Small Steps in Physics Simulation. In *Proceedings of the 18th Annual ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Los Angeles, California) (SCA '19). Association for Computing Machinery, New York, NY, USA, Article 2, 7 pages. <https://doi.org/10.1145/3309486.3340247>
- Aleka McAdams, Andrew Selle, Rasmus Tamstorf, Joseph Teran, and Eftychios Sifakis. 2011. *Computing the singular value decomposition of 3x3 matrices with minimal branching and elementary floating point operations*. Technical Report. University of Wisconsin-Madison Department of Computer Sciences.
- Matthias Müller, Jan Bender, Nuttapon Chentanez, and Miles Macklin. 2016. A robust method to extract the rotational part of deformations. In *Proceedings of the 9th International Conference on Motion in Games, MIG 2016, Burlingame, California, USA, October 10-12, 2016*, Michael Neff, Roland Geraerts, and Hubert P. H. Shum (Eds.). ACM, 55–60. <https://doi.org/10.1145/2994258.2994269>
- Matthias Müller, Julie Dorsey, Leonard McMillan, Robert Jagnow, and Barbara Cutler. 2002. Stable Real-Time Deformations. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (San Antonio, Texas) (SCA '02). Association for Computing Machinery, New York, NY, USA, 49–54. <https://doi.org/10.1145/545261.545269>
- Guillaume Picinbono, Hervé Delingette, and Nicholas Ayache. 2003. Non-linear anisotropic elasticity for real-time surgery simulation. *Graphical Models* 65, 5 (2003), 305–321. [https://doi.org/10.1016/S1524-0703\(03\)00045-6](https://doi.org/10.1016/S1524-0703(03)00045-6) Special Issue on SMI 2002.
- Teseo Schneider, Yixin Hu, Jérémie Dumas, Xifeng Gao, Daniele Panozzo, and Denis Zorin. 2018. Decoupling Simulation Accuracy from Mesh Quality. 37, 6 (2018). <https://doi.org/10.1145/3272127.3275067>
- Martin Servin, Claude Lacoursiere, and Niklas Melin. 2006. Interactive simulation of elastic deformable materials. In *Proceedings of SIGRAD Conference*. 22–32.
- Breannan Smith, Fernando De Goes, and Theodore Kim. 2018. Stable Neo-Hookean Flesh Simulation. *ACM Trans. Graph.* 37, 2, Article 12 (March 2018), 15 pages.
- D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. 1987. Elastically deformable models. *Proceedings of SIGGRAPH 87* (1987), 205–214.
- Maxime Tournier, Matthieu Nesme, Benjamin Gilles, and François Faure. 2015. Stable constrained dynamics. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–10.
- Olek C Zienkiewicz, Robert L Taylor, and Jian Z Zhu. 2005. *The finite element method: its basis and fundamentals*. Elsevier.