Simulation and Optimization of Magnetoelastic Thin Shells

XUWEN CHEN, CFCS, Peking University, China
XINGYU NI, CFCS, Peking University, China
BO ZHU, Dartmouth College, United States of America
BIN WANG*, Beijing Institute for General Artificial Intelligence, China
BAOQUAN CHEN*, School of AI, Peking University, China

Fig. 1. Our computational framework, supporting both forward simulation and inverse design of magnetoelastic thin shell structure, can realize various applications, ranging from forward locomotion generation, quasi-static shape design, to motion planning. From left to right, the images present: a swimming octopus, a crawling reptile robot, a jumping starfish, a deforming Kirigami tower, and an embossed letter.

Magnetoelastic thin shells exhibit great potential in realizing versatile functionalities through a broad range of combination of material stiffness, remnant magnetization intensity, and external magnetic stimuli. In this paper, we propose a novel computational method for forward simulation and inverse design of magnetoelastic thin shells. Our system consists of two key components of forward simulation and backward optimization. On the simulation side, we have developed a new continuum mechanics model based on the Kirchhoff–Love thin-shell model to characterize the behaviors of a megnetolelastic thin shell under external magnetic stimuli. Based on this model, we proposed an implicit numerical simulator facilitated by the magnetic energy Hessian to treat the elastic and magnetic stresses within a unified framework, which is versatile to incorporation with other thin shell models. On the optimization side, we have devised a new differentiable simulation framework equipped with an efficient adjoint formula to accommodate various PDE-constraint, inverse design problems of magnetoelastic thin-shell structures, in both static and dynamic settings. It also encompasses applications of magnetoelastic soft robots, functional Origami, artworks, and meta-material designs. We demonstrate the efficacy of our framework by designing and simulating a broad array of magnetoelastic thin-shell objects that manifest complicated interactions between magnetic fields, materials, and control policies.

*corresponding authors

Authors' addresses: Xuwen Chen, pku_xwchen@pku.edu.cn, CFCS, Peking University, China; Xingyu Ni, nixy@pku.edu.cn, CFCS, Peking University, China; Bo Zhu, bo.zhu@dartmouth.edu, Dartmouth College, United States of America; Bin Wang, binwangbuaa@gmail.com, Beijing Institute for General Artificial Intelligence, China; Baoquan Chen, baoquan@pku.edu.cn, School of AI, Peking University, China.

0730-0301/2022/7-ART61 $15.00
https://doi.org/10.1145/3528223.3530142

CCS Concepts: • **Computing methodologies** → **Physical simulation**; • **Applied computing** → *Physics*.

Additional Key Words and Phrases: magnetic simulation, thin shell, magnetoelastic coupling

ACM Reference Format:
Xuwen Chen, Xingyu Ni, Bo Zhu, Bin Wang, and Baoquan Chen. 2022. Simulation and Optimization of Magnetoelastic Thin Shells. *ACM Trans. Graph.* 41, 4, Article 61 (July 2022), 18 pages. https://doi.org/10.1145/3528223.3530142

1 INTRODUCTION

Magnetic substance simulation has received much attention in the computer graphics and computational physics field over the past years. These simulations range from magnetic rigid bodies [Kim and Han 2020; Kim et al. 2018; Thomaszewski et al. 2008], elastic solids [Wang et al. 2020a; Yan et al. 2021; Zhao et al. 2019], ferrofluid [Huang et al. 2019; Huang and Michels 2020; Ishikawa et al. 2013; Ni et al. 2020], and to viscoelastic materials [Sun et al. 2021]. The most visually appealing process underpinning these magnetic-related phenomena rests in the complex interactions between a (time-varying) magnetic field and non-linear solid/fluid material properties. Despite the inspirational breakthroughs that have taken place in the field of magnetic object simulations, the modeling of magnetic thin shells remains as an unexplored problem due to the many difficulties that come along with simulating the dynamic elasto-magnetic coupling process of a thin object. In particular, there exists no effective continuum mechanics model to characterize the magnetic-induced elastic behaviors on a thin shell, not to mention a robust numerical scheme to discretize and differentiate the mechanics model on complex thin geometries.

Thin-shell objects, such as wrinkled cloth [Bridson et al. 2003; Chen et al. 2021; Guo et al. 2018; Sperl et al. 2020], fractured sheets [Busaryev et al. 2013; Manteaux et al. 2015; Pfaff et al. 2014; Zhu et al. 2015], crumpled paper [Chen et al. 2018; Dudte et al. 2016;

Narain et al. 2013], and human skin [Rémillard and Kry 2013; van Rees et al. 2017], are observed in our daily life because of their visually appealing motion and geometry. These thin structures manifest highly desirable mechanical properties: high compactness, light weight, and extreme flexibility [Liu et al. 2021; Novelino et al. 2020; Yang et al. 2021; Zirbel et al. 2013]. Countless scholar articles have explored the simulation and optimization of thin-shell objects, focusing on their dynamics and control. These literature touched on fields that include but are not limited to: computer animation, computational fabrication, material science, and soft robotics (e.g., see [Bruton et al. 2016; Chen et al. 2018; Dang et al. 2022; Guo et al. 2018; Ly et al. 2018]). Among these lines of efforts, the design and fabrication of magnetic thin shells, which aim to deliver specific elastic behaviors under magnetic control, have emerged as future crucial tasks, for its wide applications in physical and engineering sciences (e.g., actuators [Kashima et al. 2012], medical robots [Hu et al. 2018], and drug delivery [Zhao et al. 2011]). With the help of advanced manufacturing, engineers can now embed a customized magnetization profile into a soft polymeric sheet and naturally control the shape morphing and body locomotion of a magnetic thin object, using a time-varying magnetic field. However, despite its great potential in various applications, most of the previous work on magnetic thin-shell modeling/design (as well as magnetic soft bodies) were based on engineer's intuition and abundant trial-and-error experiments. An efficient material simulation algorithm of these magneto-mechanical systems, in conjunction with its differentiable optimizer and design framework, become an emergent necessity to automate the design and comprehension of various emerging applications related to magnetic-shell interactions.

We identified two challenges while developing simulation and optimization algorithms to model magnetoelastic thin shells. On the one hand, a continuum mechanics model to characterize the magnetic strain-stress relationship on a thin shell is needed in the current literature. In spite of the long history of study on mechanical thin shells and abundant models available (e.g. the Kirchhoff–Love shell [Cirak et al. 2000], Cosserat shell [Green and Naghdi 1968], elastica [Martin et al. 2010]), the extension of these models to accommodate magnetic interaction, as well as their discretization and numerical PDE solvers on a discretized thin-shell geometry, is not a trivial task for numerical simulation. On the other hand, building a fully differentiable simulation framework and the associated adjoint solvers and optimization frameworks for magnetoelastic thin shells are in need to furnish supports for the various design and optimization applications [Eshaghi et al. 2021; Kim et al. 2019; Zhao et al. 2019]. In particular, many critical factors must be taken into consideration for a thin-shell object design problem, such as the inevitable multi-physics environment interaction and the numerical instability due to nonlinear deformation.

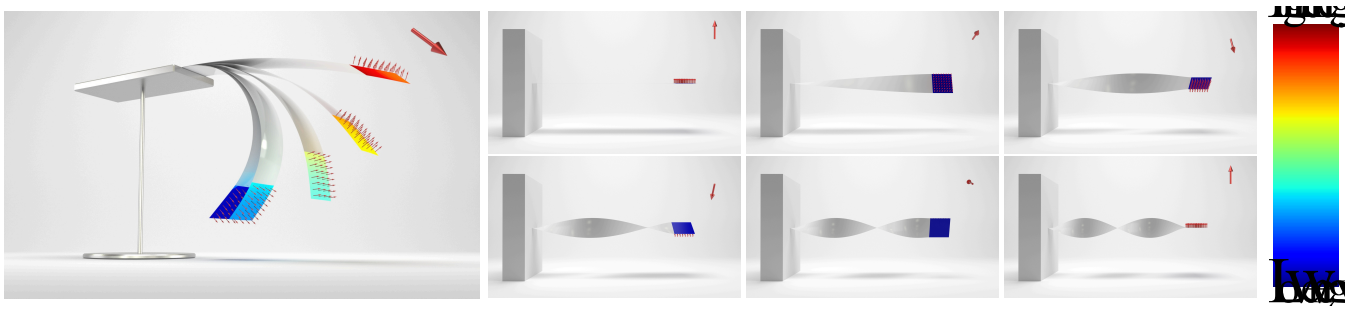
We propose a unified computational framework to solve the simulation and optimization problem for magnetic thin shells. Our model is distinguished from others for its flexibility and versatility. It can seamlessly integrate with any mainstream differentiable framework, high performance elasticity solver, and numerical optimization library, which proves that we are making substantial progress toward the goal to complete current differentiable physical ecosystem with the ability on magnetic phenomena simulation and optimization. We summarize our technical contributions as:

- A continuum mechanics model for magnetoelastic thin-shell modeling that can be integrated into the existing thin-sell simulators in a seamless fashion;
- A differentiable simulation framework enhanced to support efficient gradient calculations with respect to multi-faceted design parameters, including the material properties, the residual magnetic flux density, and control policies;
- An fully automated design system to generate high-resolution and high-performance magnetoelastic dynamic system designs, exemplified by various animation, material, and robotic designs such as crawling and jumping micro-robots, swimming fish, soft-bodied hand, and mini-Kirigami graspers.

2 RELATED WORK

Thin-shell simulation. The study on continuum models for elastic thin objects exhibiting bending resistance started from Terzopoulos et al. [1987]. Bridson et al. [2003] derived a general discrete thin-shell model based on a triangle mesh, whose elastic forces are derived from an intuitive analysis. On the other hand, Grinspun et al. [2003] introduced an alternative energy model based on differences of squared mean curvature. Both of them express the interaction as the sum of membrane part and bending part. More physically accurate analysis was introduced later by [Gingold et al. 2004], whose model of deformed shells is derived using differential geometry under the Kirchhoff–Love assumption, with bending energy discretized in terms of dihedral angles. Bergou et al. [2006] furthermore presented a class of isometric bending models that accelerates time-integration of triangular-meshed cloth and shells. Remeshing techniques were also invented under this framework in order to enable drastic deformations [Narain et al. 2012], especially when plasticity exists [Narain et al. 2013; Pfaff et al. 2014]. In recent years, further augmentations on the capability of sensing intrinsic geometry changes in response to environmental stimuli[Chen et al. 2018; van Rees et al. 2017], and describing compression/extension and shearing normal to the mid-surface [Guo et al. 2018] broaden the application scope of thin-shell models. At the same time, there was a multitude of previous work in computational design that focused on the simulation and optimization for novel thin structures and mechanics, including surface-based inflatables [Panetta et al. 2021], X-shells [Panetta et al. 2019], FlexMaps [Malomo et al. 2018], Kirchhoff–Plateau surfaces [Pérez et al. 2017]. Some other work focused on the fabrication feasibility such as developability, where thin shells are 3D printed as planar and targeted at shaping curved surfaces [Ghaffari et al. 2015; Malomo et al. 2018; Pérez et al. 2017].

Magnetic simulation. Simulation of magnetic effects based on physical computing has been promoted by a large amount of literature in the community of computer graphics. Thomaszewski et al. [2008] firstly shifted attention to magnetic rigid-body, and simplified the scenario to linear magnetization and ignored the reaction of magnetized substance to the magnetic field. Kim et al. [2020; 2018] presented novel methods of dynamic magnetization simulation. Through delicately designed magnet models, they managed to apply the *Landau–Lifshitz–Gilbert equation* in micro-magnetism



(a) (b)

Fig. 2. We test (a) bending and (b) twisting behaviors of a thin plate actuated by magnetic field (red arrow). The color on the plate represents the magnitude of local magnetic energy, which increases from blue to red.

into macro-scale simulation, which is also stable due to the nonlinear magnetization, and facilitates mutual induction and remanence as well. Other work concentrates on magnetic simulation of non-rigid bodies, including ferrofluids [Huang et al. 2019; Huang and Michels 2020; Ishikawa et al. 2013; Ni et al. 2020] and viscoelastic bodies [Sun et al. 2021]. As far as we know, simulation of magnetic deformable solids has not been specifically studied yet. In the engineering community, the model of isotropic magneto-sensitive Cauchy-elastic solids was developed by Dorfmann and Ogden [2003]. Based on this model, Zhao et al. [2019] presented the first continuum-level hard magnetoelastic material model, where the Helmholtz free energy density function comprises elastic (neo-Hookean) and magnetoelastic parts. Subsequently, a simulation framework using the finite element method, developed by the same authors, was shown to be in quantitative agreement with their experimental results. This formulation has inspired many other works on hard-magnetic beams and elastica [Wang et al. 2020a; Yan et al. 2021] modelling and simulation, or incorporating other effects such as viscosity [Garcia-Gonzalez 2019]. In concurrent work, Pezzulla et al. [2021] derive the same reduced model as proposed here, differing only by augmenting the model by also accounting for variations along thickness. They also validate the reduced model against with the 3D finite element implementation based on the full energy model proposed by Zhao et al. [2019]. However, differentiable simulation and optimization frameworks remain unexplored in Pezzulla et al. [2021].

Computational design and trajectory optimization. In computer animation and physics-based simulation, it is a common challenge to design the precise material layouts, topological structures, and external stimuli, to yield desired dynamic or quasi-static behaviors. The space-time optimization methods, being firstly introduced into graphic community by Witkin and Kass [1988], have been utilized extensively to support the design of animations [Barbič et al. 2012; Li et al. 2014; McNamara et al. 2004; Schulz et al. 2014], materials [Hahn et al. 2019; Wang et al. 2015] and structures [Wang et al. 2020b; Zhu et al. 2017]. The objective function for a typical space-time optimization problem measures a temporally-accumulated distance between the desired and the predicted motion sequence and the amount of external force required to generate the motion. In our paper, with the help of magnetic force, we can optimize trajectories without any artificial external force. Although the objective function is usually quadratic and hence straightforward to solve, the constraint function(s) can be nonlinear, which pose additional challenges to the numerical optimizer. One common approach for solving this problem is sensitivity analysis [Christopher Frey and Patil 2002], which can effectively find a feasible descent direction of the material parameters with respect to minimizing the objective while satisfying the PDE constraints. It is often combined with the adjoint method [McNamara et al. 2004] to efficiently evaluate the descent direction. This is especially important for a long temporal sequence. To efficiently calculate derivatives of various physical systems, differentiable physics simulations have emerged as active research direction crossing graphics and machine learning communities over past years. A broad array of differentiable simulators have been proposed, including rigid-body dynamics [Freeman et al. 2021; Werling et al. 2021], soft-body dynamics [Du et al. 2021; Geilinger et al. 2020; Hu et al. 2019], cloth [Liang et al. 2019; Qiao et al. 2020] , and fluid dynamics [Du et al. 2020; Holl et al. 2020; Ma et al. 2021; Schenck and Fox 2018]. Our differentiable thin-shell simulator falls into this category by extending the physical model from the volumetric to thin-shell representation.

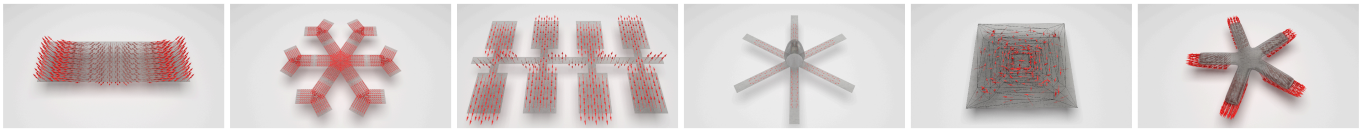
3 PHYSICAL MODELS

A fully magnetized hard-magnetic thin shell [Kim et al. 2019; Zhao et al. 2019] can generate magnetic *Cauchy stress* through the application of an actuating magnetic field to the intrinsic remnant magnetization in the material. The magnetic Cauchy stress further drives the deformation of the shell, which leads to the elastic Cauchy stress. The areal potential energy density can be calculated as the sum of the elastic part and the magnetic part:

$$\tilde{E} = \tilde{E}_{\text{elasticity}} + \tilde{E}_{\text{magnetism}}. \tag{1}$$

We will explain the calculation of the elastic term in Section 3.1 and the magnetic term in Section 3.2 respectively.

In the rest of this article, we symbolize vectors and second-order tensors (matrices) using bold letters (such as $\boldsymbol{H}$ and $\boldsymbol{F}$), and scalars using italic letters (such as H and μ_0). We consider quantities in both the material (i.e., undeformed) space and the world (i.e., deformed) space, using the convention that symbols accented with tildes (e.g., $\tilde{\boldsymbol{x}}$ vs. $\boldsymbol{x}$) refer to the former.



(a) (b) (c) (d) (e) (f)

Fig. 3. Magnetic remanence distribution of some of our examples: (a) the bar for track-fitting, (b) the hexagram, (c) the reptile robot, (d) the octopus, (e) the kirigami, and (f) the starfish.

(a) (b) (c) (d)

Fig. 4. A deforming hexagram. From left to right: (a) starting from the relaxation position, the magnetized thin-shell hexagram deforms in given magnetic field, producing motions of contraction (b) curling down and (c) curling up and (d) arm-shaking. The magnetization profile is optimized using our quasi-static solver.

3.1 Elasticity of Thin Shells

We build our elastic thin-shell model under the Kirchhoff–Love assumption, which characterizes a thin shell as a codimension-one surface and a uniform thinkness h that is much smaller than its minimal radius of curvature.

We define the deformation of the thin shell as a mapping from its material space to the world space: $\boldsymbol{\varphi}(\cdot, t) : \Omega_0 \to \Omega_t$ for $\Omega_0, \Omega_t \subset \mathbb{R}^3$, where $\tilde{\boldsymbol{x}} \in \Omega_0$ refers to material points and $\boldsymbol{x} \in \Omega_t$ represents the point at time t in the world space. The second-order tensor $\boldsymbol{F} = \partial \boldsymbol{x}/\partial \tilde{\boldsymbol{x}}$, named *deformation gradient*, is used to analyse deformation, whose determinant is denoted by J, which characterizes infinitesimal volume change. The deformation gradient determinant is further used to calculate the mass density ρ and the energy density W in the world space:

$$\begin{cases} \rho = J^{-1}\tilde{\rho}, & (2) \\ W = J^{-1}\tilde{W}. & (3) \end{cases}$$

To quantitatively analyse strains, the *Green strain tensor* $\boldsymbol{G}$ is defined as $\boldsymbol{G} = (\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F} - \boldsymbol{I})/2$, in which $\boldsymbol{I}$ is the second-order unit tensor. For thin shells, this tensor is further decomposed as

$$\boldsymbol{G}(w) = \boldsymbol{G}_{\text{membrane}} + w\boldsymbol{G}_{\text{curvature}}, \quad w \in [-h/2, +h/2], \tag{4}$$

with the membrane and curvature strains calculated only on the middle surface. Specifically, $\boldsymbol{G}_{\text{membrane}}$ equals to $(\boldsymbol{S}^{\mathrm{T}}\boldsymbol{S} - \boldsymbol{I})/2$ with $\boldsymbol{S}$ as the two-dimensional in-plane deformation gradient, while the curvature part takes the *shape operator* into account [Gingold et al. 2004].

Based on the two stains, we take the *St. Venant–Kirchhoff model* of nonlinear materials to calculate the bulk energy density as

$$\tilde{W}_{\text{elasticity}} = \frac{Y}{2(1-\nu^2)}\left((1-\nu)\operatorname{Tr}\boldsymbol{G}^2 + \nu\operatorname{Tr}^2\boldsymbol{G}\right), \tag{5}$$

where the coefficients Y and ν are the *Young modulus* and the *Poisson ratio* of the material. By taking some approximations and integrating $\tilde{W}_{\text{elasticity}}$ from $-h/2$ to $+h/2$, we obtain the areal elastic energy density $\tilde{E}_{\text{elasticity}}$ as follows:

$$\begin{cases} \tilde{E}_{\text{elasticity}} = \tilde{E}_{\text{membrane}} + \tilde{E}_{\text{bending}}, & (6) \\ \tilde{E}_{\text{membrane}} = \dfrac{Yh}{2(1-\nu^2)}\left((1-\nu)\operatorname{Tr}\boldsymbol{G}_{\text{mem}}^2 + \nu\operatorname{Tr}^2\boldsymbol{G}_{\text{mem}}\right), & (7) \\ \tilde{E}_{\text{bending}} = \dfrac{Yh^3}{24(1-\nu^2)}\left((1-\nu)\operatorname{Tr}\boldsymbol{G}_{\text{cur}}^2 + \nu\operatorname{Tr}^2\boldsymbol{G}_{\text{cur}}\right), & (8) \end{cases}$$

in which the subscripts 'mem' and 'cur' are abbreviations of 'membrane' and 'curvature' respectively.

3.2 Magnetism of Thin Shells

We derive the formula of magnetic thin-shell energy from the general model of volumetric magnetoelastic bodies.

The magnetic theory in matter studies interactions between two fields: the *magnetic flux intensity* $\boldsymbol{B}$ and the *magnetic field intensity* $\boldsymbol{H}$. In the absence of any free current or displacement current, the evolution of the magnetic field is governed by quasi-static Maxwell's equations:

$$\begin{cases} \nabla \cdot \boldsymbol{B} = 0, & (9) \\ \nabla \times \boldsymbol{H} = 0. & (10) \end{cases}$$

Meanwhile, we define $\tilde{\boldsymbol{B}}$ and $\tilde{\boldsymbol{H}}$ in Maxwell's equations in the material space as

$$\begin{cases} \tilde{\boldsymbol{B}} = J\boldsymbol{F}^{-1}\boldsymbol{B}, & (11) \\ \tilde{\boldsymbol{H}} = \boldsymbol{F}^{\mathrm{T}}\boldsymbol{H}. & (12) \end{cases}$$

We provide the mathematical derivation in Appendix A.1. We also refer the readers to Dorfmann and Ogden [2014] for a detailed discussion.

The magnetization of ferromagnetic substance can be depicted by the hysteresis loop, as shown in Figure 6. Once the material is fully magnetized, there exists a characteristic remanence magnetization

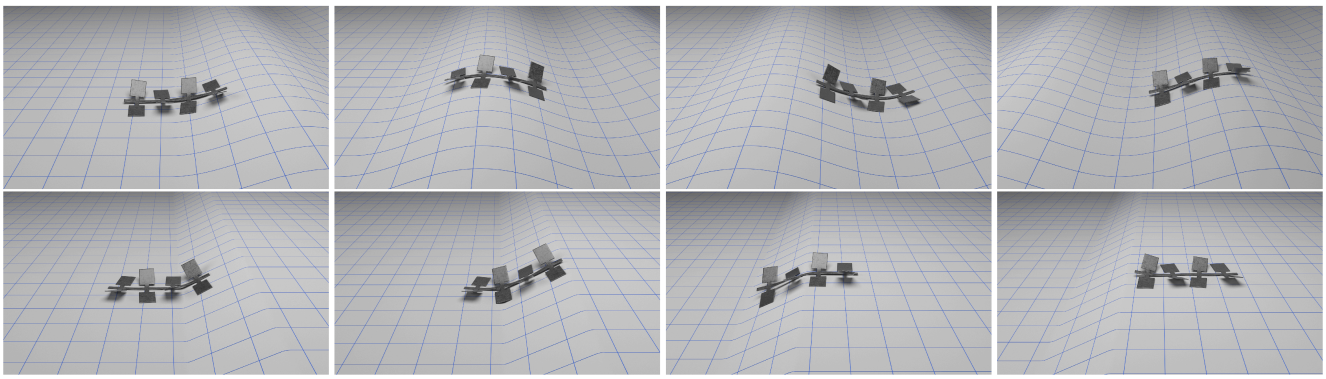


Fig. 5. A crawling reptile thin-shell robot. The magnetized thin-shell structure climbs up a bump (the top row) and over a slope (the bottom row), under a pure rotating magnetic field.

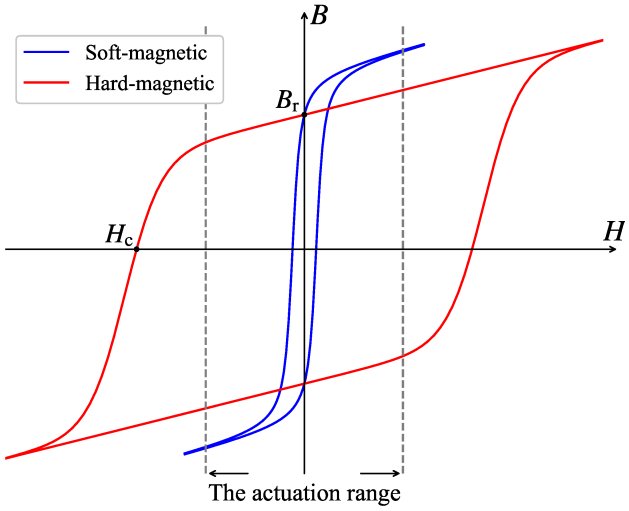


Fig. 6. Magnetic hysteresis loops and B-H curves of soft-magnetic and hard-magnetic materials, with $B_r = \mu_0 M_r$ as the *residual magnetic flux intensity* and H_c as the *coercive magnetic field intensity*.

$\boldsymbol{B}_r$ when the external magnetic field is removed. Demagnetization happens when a reversed external magnetic field with a magnitude greater than the coercive magnetic field intensity $\boldsymbol{H}_c$ is applied. According to the magnitude of $\boldsymbol{H}_c$, ferromagnetic substances can be further divided into two categories — soft magnetic material and hard magnetic material. Hard-magnetic materials maintain a stable residual magnetic flux density covering a wider range of actuation magnetic intensity. And the hysteresis loop can be further simplified with a linear relation, which leads to the ideal hard-magnetic assumption.

Our magnetic model is motivated by Zhao et al. [2019] in whose work the *ideal hard-magnetic model* was used to depict magnetization when the field strength is far below the coercivity of the immersed hard-magnetic material. As shown in Figure 6, the increase of $\boldsymbol{B}$ is proportional to the increase of $\boldsymbol{H}$ within the actuation field range, which can be formulated by

$$\boldsymbol{B} = \mu_0(\boldsymbol{H} + \boldsymbol{M}_r), \tag{13}$$

with μ_0 as the constant *vacuum permeability*. Here, the *remanent magnetization intensity* $\boldsymbol{M}_r$ is assumed to be independent of $\boldsymbol{H}$, behaving as an intrinsic, embedded quantity of hard-magnetic materials, which means $\tilde{\boldsymbol{M}}_r$, defined as

$$\tilde{\boldsymbol{M}}_r = JF^{-1}\boldsymbol{M}_r, \tag{14}$$

remains constant (Appendix A.1).

Considering the work required to realign $\boldsymbol{M}_r$ along $\boldsymbol{B}$ in the world space, we acquire the magnetic energy density of the immersed hard-magnetic material as follows:

$$W_{\text{magnetism}} = -\boldsymbol{M}_r \cdot \boldsymbol{B} = -J^{-1}\boldsymbol{F}\tilde{\boldsymbol{M}}_r \cdot \boldsymbol{B}, \tag{15}$$

which can be pulled back into the material space as

$$\tilde{W}_{\text{magnetism}} = JW_{\text{magnetism}} = -\boldsymbol{F}\tilde{\boldsymbol{M}}_r \cdot \boldsymbol{B}. \tag{16}$$

The previous two equations indicate that selecting proper external magnetic filed is able to trigger potential energy transition between elasticity and magnetism. Similar ideas have been applied to support volumetric elasticity-magnetism energy transition in Zhao et al. [2019] through its Equations (3.7) and (3.8).

In practice, the magnetic flux intensity is often divided into two components as $\boldsymbol{B} = \boldsymbol{B}_{\text{applied}} + \boldsymbol{B}_{\text{induced}}$, where $\boldsymbol{B}_{\text{applied}}$ is the external field applied and $\boldsymbol{B}_{\text{induced}}$ is the internal field induced by concerned magnetic materials. We assume that the latter is far smaller than the former, so $\boldsymbol{B}$ in Equations (15) and (16) can be approximately replaced by $\boldsymbol{B}_{\text{applied}}$.

Taking the thin-shell model, we further assume that $\boldsymbol{M}_r$ does not vary along the normal direction to the middle surface, which makes it possible to flatten the magnetic energy density as

$$\begin{aligned}\tilde{E}_{\text{magnetism}} &= \int_{-\frac{h}{2}}^{+\frac{h}{2}} \tilde{W}_{\text{magnetism}}\, \mathrm{d}z \\ &= -\left(\int_{-\frac{h}{2}}^{+\frac{h}{2}} \boldsymbol{F}(z)\, \mathrm{d}z\right) \tilde{\boldsymbol{M}}_r \cdot \boldsymbol{B}_{\text{applied}} \\ &\approx -h\boldsymbol{F}|_{z=0}\, \tilde{\boldsymbol{M}}_r \cdot \boldsymbol{B}_{\text{applied}}.\end{aligned} \tag{17}$$

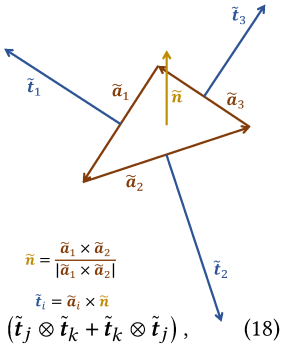
The applied magnetic field is chosen to be continuous within the concerned domain, in order that $\boldsymbol{B}_{\text{applied}}$ can be regarded as an invariant along the normal direction to the middle surface in the thin shell.

4 FORWARD SIMULATION

4.1 Discretization

Our discretization of elasticity follows [Gingold et al. 2004]. Their discretized deformation gradient is further utilized here for the discretization of magnetism. We discretize the thin shell using a triangle mesh, assuming constant $\boldsymbol{G}_{\text{mem}}$, $\boldsymbol{G}_{\text{cur}}$, and $\tilde{\boldsymbol{M}}_{\text{r}}$ for each triangle. The number of vertices, faces, and edges of the discretized thin shell object are denoted by n_{v}, n_{f} and n_{e} respectively for conveniency.

As shown in the inset figure, we use $\tilde{\boldsymbol{a}}_i$ with $i = 1, 2, 3$ to denote the three edge vectors of a triangle in the material space. We use $\tilde{\boldsymbol{a}}_i$ to denote the same edge vectors in the world space. We define vector $\tilde{\boldsymbol{t}}_i$ as 90 degree clockwise rotated $\tilde{\boldsymbol{a}}_i$. The membrane strain of the face can be calculated as



$$\boldsymbol{G}_{\text{mem}} = \frac{1}{16\tilde{A}^2} \sum_{i,j,k} \left(\tilde{\boldsymbol{a}}_i^2 - \boldsymbol{a}_i^2\right) \left(\tilde{\boldsymbol{t}}_j \otimes \tilde{\boldsymbol{t}}_k + \tilde{\boldsymbol{t}}_k \otimes \tilde{\boldsymbol{t}}_j\right), \tag{18}$$

where (i, j, k) is a circular permutation of $(1, 2, 3)$, and $\tilde{A}$ stands for the area of the undeformed triangle. Similarly, by averaging the shape operator onto faces, we acquire the curvature strain of the face as

$$\boldsymbol{G}_{\text{cur}} = \frac{1}{2\tilde{A}} \sum_{i=1}^{3} \frac{1}{|\tilde{\boldsymbol{a}}_i|} \left(\kappa(\tilde{\theta}_i) - \kappa(\theta_i)\right) \left(\tilde{\boldsymbol{t}}_i \otimes \tilde{\boldsymbol{t}}_i\right), \tag{19}$$

with $\kappa(\theta) = \theta$ in our model. Besides, given that $\tilde{\boldsymbol{n}} = \tilde{\boldsymbol{a}}_1 \times \tilde{\boldsymbol{a}}_2 / |\tilde{\boldsymbol{a}}_1 \times \tilde{\boldsymbol{a}}_2|$ and $\boldsymbol{n} = \boldsymbol{a}_1 \times \boldsymbol{a}_2 / |\boldsymbol{a}_1 \times \boldsymbol{a}_2|$, we obtain the deformation gradient at $z = 0$ as

$$\boldsymbol{F}_0 = \boldsymbol{F}|_{z=0} = \begin{pmatrix} \boldsymbol{a}_1 & \boldsymbol{a}_2 & \boldsymbol{n} \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{a}}_1 & \tilde{\boldsymbol{a}}_2 & \tilde{\boldsymbol{n}} \end{pmatrix}^{-1}. \tag{20}$$

Explanations of these formulae can be found in the work of Gingold et al. [2004].

Substituting Equations (18), (19), and (20) into Equations (7), (8) and (17) respectively and conducting integration, we obtain each face's potential energy ε as

$$\begin{cases} \varepsilon = \varepsilon_{\text{membrane}} + \varepsilon_{\text{bending}} + \varepsilon_{\text{magnetism}} + \varepsilon_{\text{gravity}}, & (21) \\ \varepsilon_{\text{membrane}} = \dfrac{Y\tilde{A}h}{2(1-\nu^2)} \left((1-\nu)\,\text{Tr}\,\boldsymbol{G}_{\text{mem}}^2 + \nu\,\text{Tr}^2\,\boldsymbol{G}_{\text{mem}}\right), & (22) \\ \varepsilon_{\text{bending}} = \dfrac{Y\tilde{A}h^3}{24(1-\nu^2)} \left((1-\nu)\,\text{Tr}\,\boldsymbol{G}_{\text{cur}}^2 + \nu\,\text{Tr}^2\,\boldsymbol{G}_{\text{cur}}\right), & (23) \\ \varepsilon_{\text{magnetism}} \approx -\tilde{A}h\boldsymbol{F}_0\tilde{\boldsymbol{M}}_{\text{r}} \cdot \boldsymbol{B}_{\text{applied}}, & (24) \\ \varepsilon_{\text{gravity}} = \tilde{\rho}\tilde{A}h, & (25) \end{cases}$$

where the height h is measured at the barycenter of the triangle, and the quantity of $\boldsymbol{B}_{\text{applied}}$ is also sampled there.

4.2 Time Integration

We adopt the implicit Euler method to discretize forward simulation in time, which requires converting numerical integral equations into a minimization problem of $\boldsymbol{x}$ and an update step of $\boldsymbol{v}$:

$$\boldsymbol{x}^{n+1} = \arg\min_{\boldsymbol{x}} E'(\boldsymbol{x}), \tag{26}$$

$$\boldsymbol{v}^{n+1} = \frac{1}{\Delta t}\left(\boldsymbol{x}^{n+1} - \boldsymbol{x}^n\right), \tag{27}$$

in which the vertex positions $\boldsymbol{x}_i$ and velocities $\boldsymbol{v}_i$ are packed into vectors $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n_{\text{v}}})$ and $\boldsymbol{v} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{n_{\text{v}}})$. A typical representation of E' for dynamic simulation looks like:

$$E' = \frac{1}{2\Delta t^2} \sum_{i=1}^{n_{\text{v}}} m_i \left|\boldsymbol{x}_i - \boldsymbol{x}_i^n - \Delta t\, \boldsymbol{v}_i^n\right|^2 + \sum_{j=1}^{n_{\text{f}}} \varepsilon_j(\boldsymbol{x}), \tag{28}$$

where i iterates over vertex indices; j iterates over face indices. The nodal mass m_i is calculated by the weighted average of its neighboring faces. For quasi-static simulation, only the second term is left.

A Newton-like method with diagonal correction is adopted to solve the above equations iteratively, which is shown in Algorithm 1. We refer readers to Appendix A.2 for details of Jacobian and Hessian terms needed in the implicit scheme.

ALGORITHM 1: Implicit-Euler Time Integration

Input: the packed vectors $\boldsymbol{x}^n$ and $\boldsymbol{v}^n$ at the current time, the energy function $E(\boldsymbol{x})$ and the time step Δt.
Output: the packed vectors $\boldsymbol{x}^{n+1}$ and $\boldsymbol{v}^{n+1}$, after the time step.

```
x ← x^n + Δt v^n;                                   // initial guess
for i ← 1 to the maximal number of iterations do
    J ← ∂E/∂x, H ← ∂²E/∂x²;
    λ ← 10^{-6};
    while True do
        H ← H + λI;        // add regularizer to the Hessian
        Solve Hp = J with some preconditioner;
        if success then Break;
        λ ← 2λ;
    end
    x ← line search x to minimize E' in the direction of p;
    if J · p < η then Break;                        // η = 10^{-12}
end
v^{n+1} = (x^{n+1} − x^n)/Δt;
```

5 OPTIMIZATION

We define our optimization problem in a saturated-magnetized hard-magnetic thin shell structure immersed in external magnetic field. The optimization parameters include magnetic field configuration $\boldsymbol{B}_{\text{applied}}$, elastic stiffness Y, and remanent magnetization intensity $\boldsymbol{M}_{\text{r}}$.

We formulate two categories of optimization problems: (1) Given $\boldsymbol{B}_{\text{applied}}$, we want to design $\boldsymbol{M}_{\text{r}}$ or the combination of $\boldsymbol{M}_{\text{r}}$ and Y to transform the reference shape of the thin shell structure to a specified target shape. (2) Given $\boldsymbol{M}_{\text{r}}$ and Y in the domain, we want to design $\boldsymbol{B}_{\text{applied}}$ to enable the thin shell to deliver the desired dynamic motion or achieve designated positions.

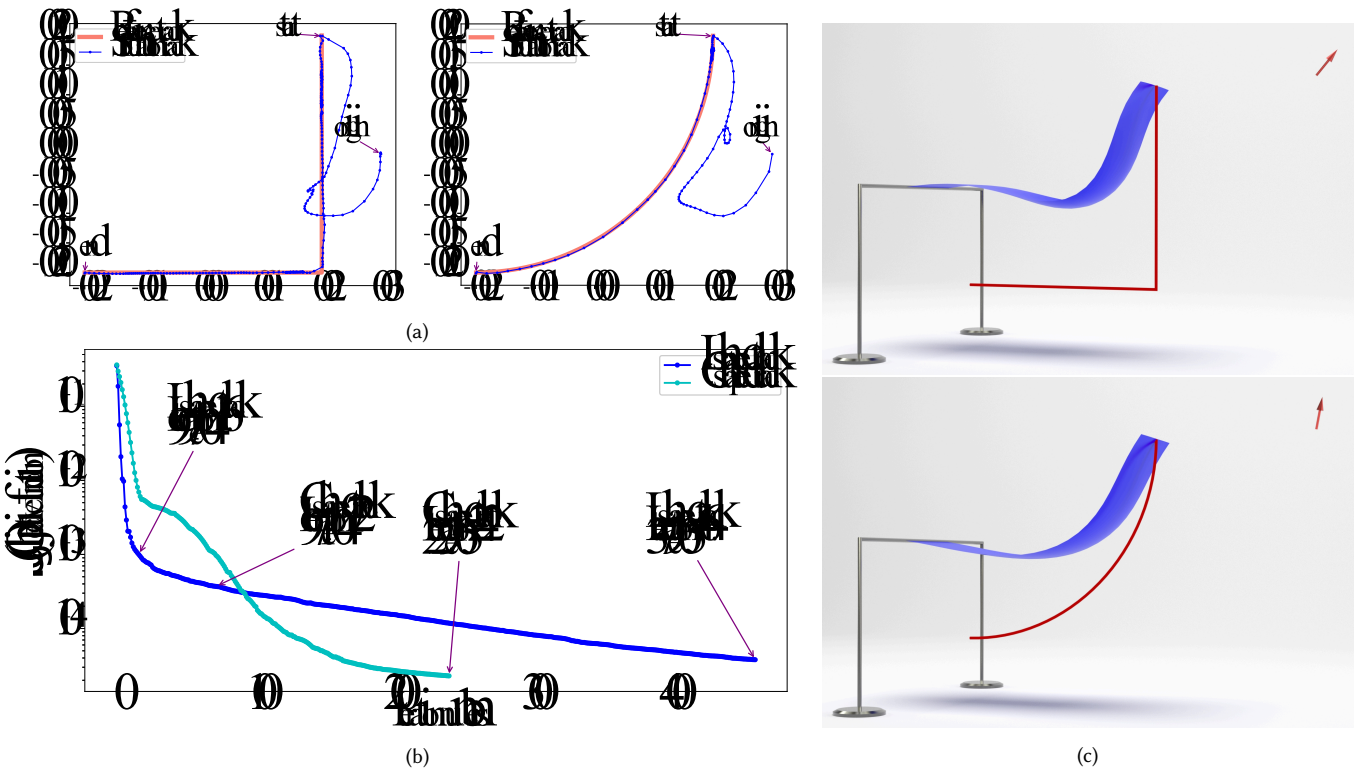


Fig. 7. Track fitting. The bar with one end fixed and the other end moves, driven by the magnetic field, precisely tracks the reference trajectory. As shown in (a), the red line shows the reference trajectory while the blue line shows the trajectory of the moving end, with dots indicating positions in progressive frames. The blue points are initially away from the red curve as it takes time for them to move to the starting point; these frames are outside the consideration of the objective function, hence will not affect the final result. (b) shows the convergence curve of the objective function. (c) shows the frame in which the bar arrives at the starting point of the reference track.

5.1 Quasi-Static Optimization

Parameters. The target shape is expressed in a $3n_{\mathrm{v}}$-dimensional stacked vector of vertex positions, denoted $\boldsymbol{x}^*$, and the parameters to optimize are stacked into a vector $\boldsymbol{k}$. Since we aim to design the remanent magnetization distribution of a thin-shell object, we stack the three-dimensional magnetization vector $\tilde{\boldsymbol{M}}_{\mathrm{r}}$ on each face of the thin shell in the material space to form a long vector $\boldsymbol{k}_{\mathrm{mag}} \in \mathbb{R}^{3n_{\mathrm{f}}}$. In order to get a better folding effect, we also optimize the membrane coefficients $\boldsymbol{k}_{\mathrm{mem}} \in \mathbb{R}^{3n_{\mathrm{f}}}$ and the curvature coefficients $\boldsymbol{k}_{\mathrm{cur}} \in \mathbb{R}^{3n_e}$ at the same time. The former is actually the Young modulus of each face, and the latter is used to scale the dihedral angle θ into the curvature for each edge. In a word, $\boldsymbol{k}$ is constructed by concatenating $\boldsymbol{k}_{\mathrm{mag}}$, $\boldsymbol{k}_{\mathrm{mem}}$ and $\boldsymbol{k}_{\mathrm{cur}}$.

Formulations. To configure the target shape, an optimization problem is formulated as follows:

$$\arg\min_{\boldsymbol{k}} \quad F(\boldsymbol{x}(\boldsymbol{k})) = \left|\boldsymbol{x}(\boldsymbol{k}) - \boldsymbol{x}^*\right|^2, \tag{29}$$

$$\text{subject to} \quad \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{k}) = \boldsymbol{0}, \tag{30}$$

with the objective function $F(\boldsymbol{x})$ as the sum of square distances, and the constraints

$$\boldsymbol{g} = \frac{\partial \sum_{j=1}^{n_{\mathrm{f}}} \varepsilon_j(\boldsymbol{k})}{\partial \boldsymbol{x}} = \boldsymbol{0} \tag{31}$$

indicating the force balance principle, derived from Equation (28). It is noted that $\boldsymbol{x}$ is determined by $\boldsymbol{k}$ implicitly.

Gradients. The derivative of the objective function F with respect to $\boldsymbol{k}$ is calculated by the method of Lagrange multipliers:

$$L = F(\boldsymbol{x}) + \boldsymbol{\lambda}^{\mathrm{T}} \boldsymbol{g}, \tag{32}$$

where $\boldsymbol{\lambda}$ is a $3n_{\mathrm{v}}$-dimensional multiplier vector. If we set $\boldsymbol{\lambda}$ so that the equation

$$\frac{\partial F}{\partial \boldsymbol{x}} + \boldsymbol{\lambda}^{\mathrm{T}} \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}} = \boldsymbol{0}, \tag{33}$$

holds, then the gradient of F is obtained by

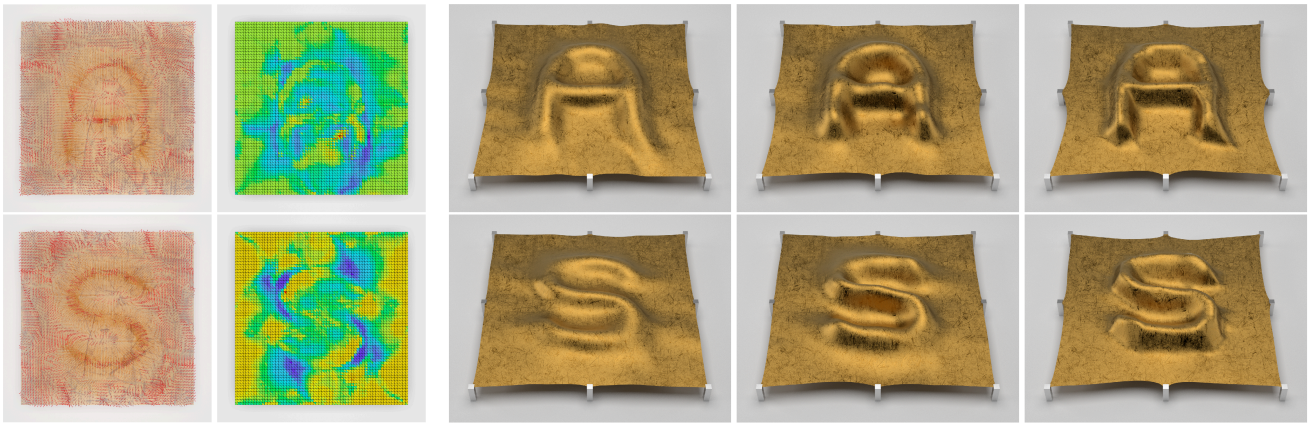
$$\frac{\mathrm{d}F}{\mathrm{d}\boldsymbol{k}} = \frac{\mathrm{d}L}{\mathrm{d}\boldsymbol{k}} = \boldsymbol{\lambda}^{\mathrm{T}} \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{k}} = -\frac{\partial F}{\partial \boldsymbol{x}} \left(\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right)^{-1} \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{k}}.$$

Optimization. We utilize the method of moving asymptotes (MMA) [Svanberg 2002] to solve the optimization problem in this section. The steps to optimize the parameter vector are summarized in Algorithm 2.

5.2 Trajectory Optimization

Parameters. Without loss of generality, we assume that there are $s+1$ frames in total, uniformly sampled from $t^0 = 0$ to $t^s = T$, with $\Delta t = T/s$ as the time step. Any quantity taking t as an argument is

Fig. 8. SIGGRAPH banner. The eight letters are individually optimized as in the right of Figure 9b, and then their optimized material properties are assembled to form the banner.



(a) (b)

Fig. 9. Optimizing remanence alone vs. jointly with Young's modulus. (a) shows the remanence optimization results of letter A (the top) and S (the bottom). The arrows in the left column represent the magnetic remanence of the faces, while the colors in the right column represent the magnitude of Young's modulus of the faces; the value increases from purple to yellow. (b) From left to right, each column shows the optimization results with $\boldsymbol{M}_r$, ($\boldsymbol{M}_r$, $\boldsymbol{k}_{\text{mem}}$) and ($\boldsymbol{M}_r$, $\boldsymbol{k}_{\text{mem}}$,$\boldsymbol{k}_{\text{cur}}$) as optimization variables respectively. As can be seen clearly, co-designing of elastic and magnetic materials can yield much better results with clear edges and sharp folds.

then rewritten with a superscript n that denotes the time $t^n = n\Delta t$. Parameters $\boldsymbol{k}$ that need optimizing determine $\boldsymbol{B}_{\text{applied}}$ within the time interval. For example, we use $\boldsymbol{k} \in \mathbb{R}^{3s}$ to stack a uniform external field that can change the strength and direction at each frame. In addition, there are some key frames to control the thin-shell robot, whose indices form a set U. The initial states, $\boldsymbol{x}^0$ and $\boldsymbol{v}^0$, are given ahead.

Formulations. To design the applied magnetic field, a trajectory optimization problem is formulated as follows:

$$\arg\min_{\boldsymbol{k}} \quad F(\{\boldsymbol{x}^n\}, \boldsymbol{k}) = \sum_{n\in U} \left|\boldsymbol{x}^n - \boldsymbol{x}^{n*}\right|^2 + F_{\text{p}}(\boldsymbol{k}), \tag{34}$$

$$\text{subject to} \quad \boldsymbol{g}^n(\boldsymbol{v}^n, \boldsymbol{v}^{n-1}, \boldsymbol{x}^n, \boldsymbol{k}) = \boldsymbol{0}, \tag{35}$$

$$\text{and} \quad \boldsymbol{h}^n(\boldsymbol{x}^n, \boldsymbol{x}^{n-1}, \boldsymbol{v}^n) = \boldsymbol{0}, \quad n = 1, 2, \ldots, s, \tag{36}$$

where the constraints

$$\boldsymbol{g}^n = \boldsymbol{\mathcal{M}}\left(\boldsymbol{v}^n - \boldsymbol{v}^{n-1}\right) + \Delta t \frac{\partial \sum_{j=1}^{n_{\text{f}}} \varepsilon_j(\boldsymbol{x}^n)}{\partial \boldsymbol{x}^n} = \boldsymbol{0}, \tag{37}$$

ALGORITHM 2: Quasi-Static Optimization

Input: the objective function $F(\boldsymbol{x})$ and the constraints $g(\boldsymbol{x}, \boldsymbol{k}) = \boldsymbol{0}$.
Output: optimized $\boldsymbol{k}$.
for $i \leftarrow 1$ **to** *the maximal number of iterations* **do**
  Solve the quasi-static equation for $\boldsymbol{x}$ by Algorithm 1;
  $\boldsymbol{\lambda}^{\text{T}} \leftarrow -(\partial \boldsymbol{g}/\partial \boldsymbol{x})^{-1}(\partial F/\partial \boldsymbol{x})$;
  $\mathrm{d}F/\mathrm{d}\boldsymbol{k} \leftarrow \boldsymbol{\lambda}^{\text{T}}(\partial \boldsymbol{g}/\partial \boldsymbol{k})$;
  Pass $\mathrm{d}F/\mathrm{d}\boldsymbol{k}$ to the optimizer to calculate the descent direction;
  Use the line-search method to determine the descending step size;
  Update $\boldsymbol{k}$;
  if $|\Delta \boldsymbol{k}| < \eta$ **then** Break; // $\eta = 10^{-5}$
end

with $\boldsymbol{\mathcal{M}}$ as a $3n_{\text{v}}$th-order diagonal matrix consisting of nodal mass m_i, and

$$\boldsymbol{h}^n = \left(\boldsymbol{x}^n - \boldsymbol{x}^{n-1}\right) - \Delta t\, \boldsymbol{v}^n = \boldsymbol{0} \tag{38}$$

are derived from the implicit Euler time-integration scheme (Equations (27) and (28)), The objective function $F(\{\boldsymbol{x}^n\}, \boldsymbol{k})$ consists of

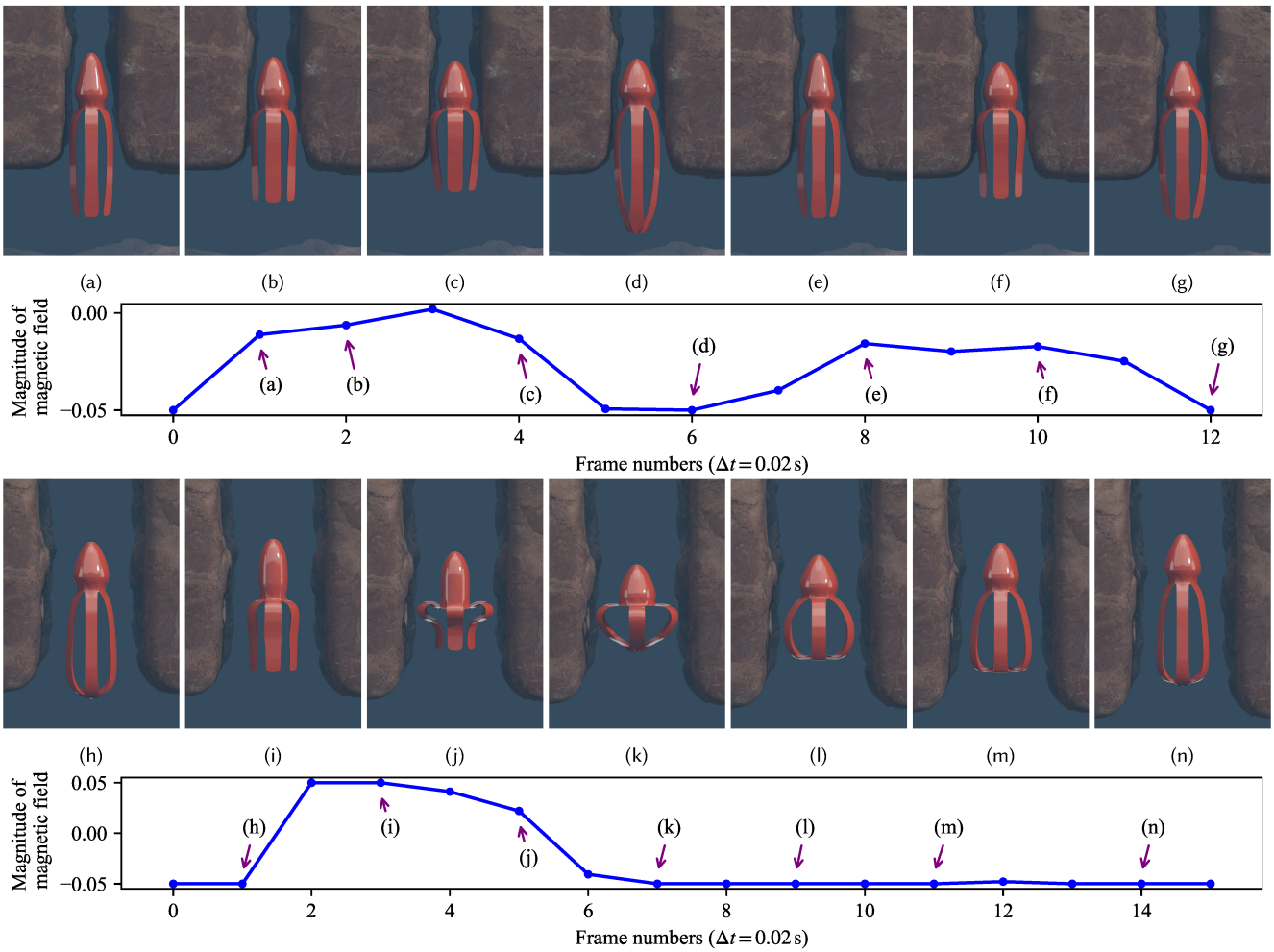


Fig. 10. Swimming octopuses. Octopuses made of thin shells swim through pipes. (a)–(g) and (h)–(n) illustrate the two typical patterns of the magnetically actuated swimming strategy that allow the octopus to go through different pipe landscapes. As shown in (a)–(g), the octopus has to adopt certain strategy to squeeze through the narrow space, while in (h)–(n), the octopus has plenty of room to stretch the tentacles, which would benefit the hydrodynamic force to create a faster swim.

two terms, in which the first term is the sum of squared distances between vertex positions and the desired ones at every key frame, and the second term, denoted $F_{\mathrm{p}}(\boldsymbol{k})$, is included to penalize excessive or discontinuous field strength if necessary.

Gradients. We utilize the adjoint method [McNamara et al. 2004] to compute the derivative of the objective function F with respect to $\boldsymbol{k}$, in which two sequences of Lagrange multipliers $\{\boldsymbol{\lambda}_v^n\}$ and $\{\boldsymbol{\lambda}_x^n\}$ are introduced:

$$L = F(\{\boldsymbol{x}^n\}, \boldsymbol{k}) + \sum_{n=1}^{s} \left(\boldsymbol{\lambda}_v^{n\mathrm{T}} \boldsymbol{g}^{\mathrm{n}} + \boldsymbol{\lambda}_x^{n\mathrm{T}} \boldsymbol{h}^{\mathrm{n}} \right), \tag{39}$$

where each $\boldsymbol{\lambda}_v^n$ or $\boldsymbol{\lambda}_x^n$ is a $3n_{\mathrm{v}}$-dimensional multiplier vector. If we set $\boldsymbol{\lambda}_v$ and $\boldsymbol{\lambda}_x$ so that the equations

$$\begin{cases} \boldsymbol{\lambda}_v^n = \boldsymbol{\lambda}_v^{n+1} + \Delta t \boldsymbol{M}^{-1} \boldsymbol{\lambda}_x^n, & \text{(40a)} \\ \boldsymbol{\lambda}_x^n = \boldsymbol{\lambda}_x^{n+1} - \Delta t \dfrac{\partial^2 \sum_{j=1}^{n_{\mathrm{f}}} \varepsilon_j(\boldsymbol{x}^n)}{(\partial \boldsymbol{x}^n)^2} \boldsymbol{\lambda}_v^n - \dfrac{\partial F}{\partial \boldsymbol{x}^n}, & \text{(40b)} \\ \boldsymbol{\lambda}_x^{s+1} = \boldsymbol{\lambda}_v^{s+1} = \boldsymbol{0}. & \text{(40c)} \end{cases}$$

hold, then the gradient of F is obtained by

$$\frac{\mathrm{d}F}{\mathrm{d}\boldsymbol{k}} = \frac{\mathrm{d}L}{\mathrm{d}\boldsymbol{k}} = \frac{\partial F_{\mathrm{p}}}{\partial \boldsymbol{k}} + \sum_{n=1}^{s} (\boldsymbol{\lambda}_v^n)^{\mathrm{T}} \frac{\partial \boldsymbol{g}^n}{\partial \boldsymbol{k}}. \tag{41}$$

Optimization. The MMA is used to solve the optimization problem here as well. The processes to solve the time-dependent optimization problem are summarized in Algorithm 3. Although the Hessian matrix in Equation (40) is time-dependent, we do not need to store

Table 1. We list parameters used in our simulation and optimization examples: the number of faces n_f (for a volumetric object, n_f denotes the number of faces on its surface mesh), the number of elements (for volumetric objects) n_e, the number of vertices n_v, Young's modulus Y, the Poisson ratio ν, the damping coefficient γ, the time step size Δt, the number of degrees of freedom m (i.e., the dimension of $\boldsymbol{k}$), the residual objective function value r, the runtime for the entire optimization t_opt, and the number of optimization iterations $iter_\mathrm{opt}$. For all the optimization examples, we set the ranges of k_mem ($k_\mathrm{cur} = k_\mathrm{mem} h^2/12$) and the external magnetic field are $[10^4, 10^8]$ (PA) and $[-0.05, 0.05]$ (T), respectively, and no bounds for the magnetic remanence.

Figure	Example†	n_f	n_e	n_v	Y [Pa]	ν	γ [N · s/m]	Δt [s]	m	r	t_opt [s]	$iter_\mathrm{opt}$
2	Bending and twisting	2048	–	1097	1×10^5	0.5	1×10^{-4}	1×10^{-2}	–	–	–	–
5	Reptile	496	–	326	1×10^5	0.5	1×10^{-2}	1×10^{-4}	–	–	–	–
4	Hexagram	2048	–	1097	1×10^5	0.5	5×10^{-3}	1×10^{-2}	6 k	6.7×10^{-2}	1.6×10^4	50
7	Track fitting (L)	768	–	417	1×10^5	0.5	1×10^{-4}	1×10^{-3}	0.5 k	3.8×10^{-5}	4.4×10^4	463
7	Track fitting (C)	768	–	417	1×10^5	0.5	1×10^{-4}	1×10^{-3}	0.3 k	2.3×10^{-5}	1.6×10^4	241
8	Letters	8192	–	4225	1×10^5‡	0.5	4	1×10^{-4}	9.0 k	2.5×10^0	3.0×10^4	90
10a–g	Octopus (pipe I)	492	–	313	1×10^4	0.5	0	1×10^{-3}	9.0 k	9.8×10^{-4}	3.6×10^4	72
10h–n	Octopus (pipe II)	492	–	313	1×10^4	0.5	0	1×10^{-3}	9.0 k	3.9×10^{-3}	2.2×10^4	41
11	Octopus (tunnel)	492	–	313	1×10^4	0.5	0	1×10^{-3}	9.0 k	8.8×10^{-3}	4.4×10^4	49
13	Kirigami*	244	–	205	1×10^5	0.5	1×10^{-3}	1×10^{-2}	0.7 k	-2.8×10^0	3.9×10^2	95
									2.1 k	4.9×10^{-4}	4.1×10^4	486
14	Hand (top)	3320	4975	1662	1×10^6	0.45	1	1×10^{-4}	1.0 k	4.1×10^{-5}	1.2×10^2	16
14	Hand (middle)	3320	4975	1662	1×10^6	0.45	1	1×10^{-4}	1.0 k	1.8×10^{-5}	1.3×10^3	30
14	Hand (bottom)	3320	4975	1662	1×10^6	0.45	1	1×10^{-4}	1.0 k	2.6×10^{-5}	5.9×10^3	45
16	Heart (top)	1326	1963	665	1×10^5	0.45	0.2	1×10^{-4}	1.4 k	3.0×10^{-3}	5.6×10^2	72
16	Heart (bottom)	1326	1963	665	1×10^5	0.45	0.2	1×10^{-4}	0.5 k	5.8×10^{-3}	9.2×10^2	42
15	Starfish (top)	1400	1870	702	1×10^6	0.45	0	5×10^{-4}	1.2 k	8.7×10^{-4}	2.5×10^4	42
15	Starfish (middle)	1400	1870	702	1×10^6	0.45	0	5×10^{-4}	1.2 k	2.0×10^{-4}	4.5×10^4	111
15	Starfish (bottom)	1400	1870	702	1×10^6	0.45	0	5×10^{-4}	1.2 k	5.3×10^{-6}	3.9×10^4	77

† All these simulations use the same values for the following parameters (except otherwise described in the text) : the contact stiffness $k_\mathrm{n} = 10^4$ N/m, the coefficient of friction $c_\mathrm{f} = 0.6$, the density of mass $\rho = 10^3$ kg/m^3 and the thickness for thin-shell objects $h = 1$ mm. All these examples use StVK model for elasticity of thin shells, except for the hands and starfish (the neo–Hookean model for elasticity of volumeric bodies).

‡ The given value is used in the case of optimizing remanence alone.

* The first line shows the data in the quasi-static optimization and the second line shows the data in the trajectory optimization. They represents different optimization tasks and share the same set of simulation parameters.

it every forward-simulation frame, due to its great memory overhead. Instead, we record $\boldsymbol{x}^n$ at each frame and calculate derivatives (Appendix A.2) on the fly.

6 EXPERIMENTAL RESULTS

In this section, we evaluate the soundness and the efficacy of our computational method by a set of experiments, including magnetic phenomena simulation, quasi-static shape design, trajectory fitting and motion control. Parameter settings used in the simulations are summarized in Table 1. These experiments were run on 6-core 3.2GHz Intel(R) Core(TM) i7-8700 desktop with 16 GB RAM.

6.1 Validation

Bending and Twisting. As shown in Figure 2, we first validate the soundness of our proposed continuum mechanics model through the bending and twisting behaviors of a thin plate actuated by a constantly rotating magnetic field. The left end of the thin shell is fixed, and the remanent magnetization is uniformly distributed on its right end only, perpendicular to the rest shape and pointing upward. The elastic material properties are uniformly distributed over the entire plate.

Five magnetic fields with different strengths, all along the direction of $[1, -1, 0]$, are applied to the thin plate and intrigue the thin

ALGORITHM 3: Trajectory Optimization

Input: the objective function $F(\{\boldsymbol{x}^n\}, \boldsymbol{k})$, the constraints $\boldsymbol{g} = 0$ and $\boldsymbol{h} = 0$, the initial states $\boldsymbol{x}^0$ and $\boldsymbol{v}^0$, the number of total frames $s+1$ and the time step Δt.
Output: optimized $\boldsymbol{k}$.
for $i \leftarrow 1$ **to** *the maximal number of iterations* **do**
  $\boldsymbol{x} \leftarrow \boldsymbol{x}_0$, $\boldsymbol{v} \leftarrow \boldsymbol{v}_0$;
  for $n \leftarrow 1$ **to** s **do**
    Advance Δt by Algorithm 1;
    $\boldsymbol{x}^n \leftarrow \boldsymbol{x}$;
  end
  $\boldsymbol{\lambda}_x \leftarrow 0$, $\boldsymbol{\lambda}_v \leftarrow 0$;
  $\mathrm{d}F/\mathrm{d}\boldsymbol{k} \leftarrow \partial F_\mathrm{p}/\partial \boldsymbol{k}$;
  for $n \leftarrow s$ **to** 1 **do**
    Update $\boldsymbol{\lambda}_v$, $\boldsymbol{\lambda}_x$ according to Equation (40);
    $\mathrm{d}F/\mathrm{d}\boldsymbol{k} \leftarrow \mathrm{d}F/\mathrm{d}\boldsymbol{k} + \boldsymbol{\lambda}_v^\mathrm{T}(\partial \boldsymbol{g}^n/\partial \boldsymbol{k})$;
  end
  Pass $\mathrm{d}F/\mathrm{d}\boldsymbol{k}$ to the optimizer to calculate the descent direction;
  Use the line-search method to determine the descending step size;
  Update $\boldsymbol{k}$;
  if $|\Delta \boldsymbol{k}| < \eta$ **then** Break; // $\eta = 10^{-5}$
end

Fig. 11. The octopus adopts various swimming strategies to respond to the changing tunnel environments and produce suitable propulsion force to push itself through.

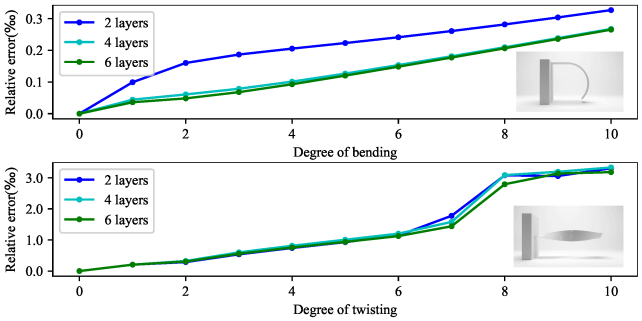


Fig. 12. Relative error of the magnetic energy derived using our shell-based and Zhao et al. [2019]'s volumetric-based derivation under different deformation configurations and resolution of volumetric discretization. The extents of bending(top) and twisting(bottom) get larger along the x-axis, with the extreme deformation demonstrated in the zoomed figures.

shell's bending deformation to different extents. We show results with increasing magnetic field magnitude from top to bottom in Figure 2a, with the color indicating the magnitude of the magnetic energy density at each material point.

As shown in Figure 2b, we further test the thin shell's twisting behavior under a rotating magnetic field orthogonal to the long axis of the plate. The magnetic field with a constant magnitude is initially pointing upward, and rotates for 720° around the horizontal axis. As Figure 2b shows, the thin plate twisted its body to allow its remanent magnetization to align with the external magnetic field. These two examples verify that the magnetic energy tends to align the remanent magnetization with the external magnetic field as depicted in Equation 15. Stronger magnetic field would introduce stronger magnetic energy to counter the elastic potential energy and consequently achieve better alignment.

We further compare our method quantitatively to the full energy model [Zhao et al. 2019] by implementing a 3D volumetric finite element method to model the magnetic thin shell. We first discretize a thin shell uniformly in the normal direction into 2/4/6 layers. We divide each layer in the tangential direction into 6144 cells, with each of which discretized into 6 tetrahedra. The nodes and triangles on the middle plane are extracted to construct the corresponding middle surface of the volumetric thin shell. As such, we obtain three volumetric meshes with $19k/32k/44k$ nodes and $74k/147k/221k$ tetrahedra respectively, whose middle surfaces consist of 12288 triangles. For each volumetric model, we assume the remanent magnetization in the model is constant. We predefined a series of bending and twisting deformations of the volumetric model. The vertex positions of the deformed middle surface are determined using its volumetric model's nodal position. For each paired volumetric and thin-shell representation, their magnetic energies are calculated using Zhao et al. [2019]'s volumetric-based and our shell-based derivation, respectively. The relative magnetic energy discrepancy between the above two schemes is defined as the accumulation of the spatial error divided by the total magnetic energy of the volumetric model. As shown in Figure 12, on different resolutions of the volumetric discretization, the energy discrepancy is consistently subtle even under large deformation.

Magnetic Reptile Robot. As shown in Figure 5, we validate our forward simulator by actuating a reptile thin-shell robot with a specifically designed remanent magnetization distribution, which is drawn from the work of Xu et al. [2019] as shown in Figure 3c. Under the actuation of a pure rotating magnetic field, the robot can realize the crawling locomotion as was claimed in the work of Xu et al. [2019]. By incorporating the frictional contact module [Geilinger et al. 2020] into our dynamic simulation, the reptile robot can maneuver over a slope or bump terrains as demonstrated in Figure 5. The magnetic force triggers the foot sway, while the frictional contact force pushes the reptile robot to move forward. It is worth noting that the symmetric and periodic pattern of the embedded remanent magnetization guarantees the steadily forward motion actuated by a single rotational magnetic field.

Magnetic Hexagram. We further validate our optimization framework with a quasi-static shape design shown in Figure 4. Motivated by the experiment setup in the work of Kim et al. [2018], we optimize the remanent magnetization of a hexagram-shaped shell to realize two target deformation modes under different magnetic inputs. One external magnetic field is a uniform vertical field and the other is its reversion. The remanent magnetization distribution as illustrated in Figure 3b is derived using our quasi-static optimizer, which shares a similar pattern as [Kim et al. 2018]. Under the actuation of a periodically reversed vertical magnetic field or a rotation field, our derived magnetic hexagram robot exhibits contraction-relaxation and arm-shaking motions, which are also consistent with the results of [Kim et al. 2018]. It is worth mentioning that this hexagram shape has been manufactured and actuated in the real physical world by Kim et al. [2018]. We recommend readers refer to the supplementary video for a side-by-side comparison between our simulation and their real-world experiment, confirming the high fidelity of our simulation.

Track Fitting. We finally validate our dynamic optimization algorithm on a trajectory fitting problem as demonstrated in Figure 7. A homogeneous elastic thin plate, with a pre-defined remanent

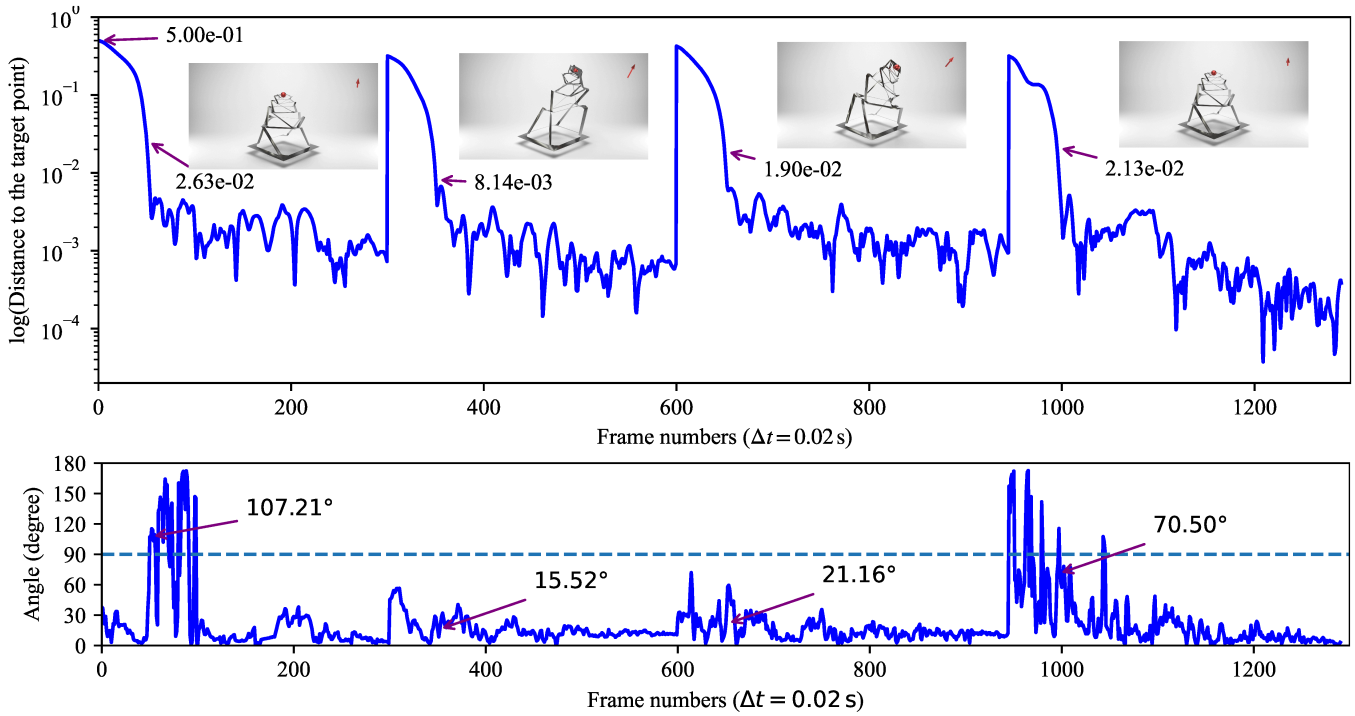


Fig. 13. A deforming Kirigami. The Kirigami moves its center point (the tower top when expanded) to 4 consecutive target locations (red balls) while its bottom is fixed on the floor. In the upper figure, the blue curve plots the distance from the Kirigami's center point to the target point in each frame; the marked distances are the recordings at exactly 1 second elapse time after a new target is initiated. As can be seen, whenever a new initiation, the large distance gets dropped very quickly; usually within 1s it reaches the new target. The blue curve in lower figure depicts the angle between the external magnetic field direction and the vector pointing from the world space's origin to the target point.

magnetization distribution (Figure 3a), is fixed on its left end (Figure 7c). The temporal evolution of the external magnetic field is optimized to control the trajectory of the middle point on the free end to follow two assigned analytical curves, which are indicated using red color in both Figure 7c and Figure 7a. One curve consists of two perpendicular segments and the other is a quarter circle. We use a three-termed objective function in this example, with two measuring how reachable to the start and end points at specific time instances, and the third one being the summation of point-wise distances to the target analytical trajectories. The tracking result and the convergence speed are shown in Figure 7a and Figure 7b, respectively, which demonstrate that precise dynamic control can be achieved using our method.

6.2 Quasi-Static Optimization

In this section, we demonstrate a magnetic-material co-design example. As shown in Figure 9, a squared thin sheet with 8 fixed boundary points is specified to enforce deforming to different letter shapes under a uniform vertical magnetic field. The target shapes are height maps of letters A and S, and the desired 3D position of each vertex is extracted from the height map through bi-linear interpolation using rest shape position. The objective function evaluates the total mismatch between the current shape and the target height map on each vertex. Compared with the results obtained by only optimizing the thin shell's magnetic properties, the results obtained by co-optimizing both remanent magnetization and Young's modulus (Figure 9a) yield significantly better designs with clear edges, sharp corners, and bugling deformation (Figure 9b). This proves that elastic and magnetic material co-design can significantly increase the controllability of object deformation. Further stitching the optimized material properties of individual letters, we create a vividly embossed banner with deformation driven by a uniform vertical magnetic field (Figure 8).

6.3 Trajectory Optimization

Motion control is more involving than the quasi-static shape design due to its larger searching space, time correlation between frames, and the sparse reward signals. In this section we mainly focus on optimizing external magnetic actuation to realize various goals. An extra regularization term penalizing the magnitude of the applied external magnetic field is employed for of all the examples.

Swimming Octopuses. As shown in Figure 10, we validate our framework by generating effective control policies under multi-physics circumstances. We use a thin-shell bullet-headed octopus model with its remanence magnetization uniformly distributed along each of its six tentacles as shown in Figure 3d. Following Min et al. [2019], the hydrodynamic force exerted on the octopus is

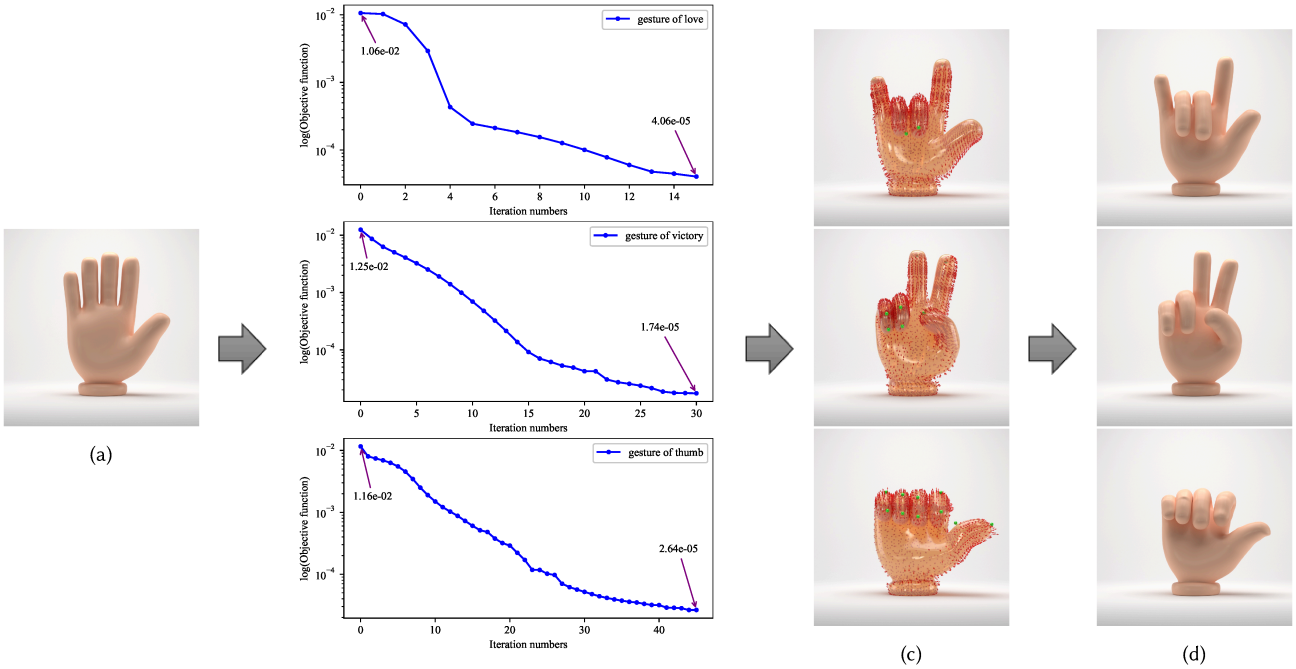


Fig. 14. Gesturing hands. (a) shows the rest shape of the hands. Three target gestures are shown in (c) with their respective anchor points marked in green dots. (b) shows the descending curve of the objective function, leading to convergence. Correspondingly, the hand objects deform and achieve force equilibrium in (d) with the optimized remanence depicted in red arrows in (c).

formulated as:

$$\begin{cases} \boldsymbol{f}_{\text{drag}} = \frac{1}{2}\rho A C_{\text{d}}(\phi)\,|\boldsymbol{v}_{\text{rel}}|^2\,\boldsymbol{d}, & (42) \\ \boldsymbol{f}_{\text{thrust}} = -\frac{1}{2}\rho A C_{\text{t}}(\phi)\,|\boldsymbol{v}_{\text{rel}}|^2\,\boldsymbol{n}, & (43) \end{cases}$$

where A denotes the area of the face, while relative velocity between surface and fluid is defined as $\boldsymbol{v}_{\text{rel}} = \boldsymbol{v}_{\text{fluid}} - (\boldsymbol{v}_0 + \boldsymbol{v}_1 + \boldsymbol{v}_2)/3$, $\boldsymbol{n}$ and $\boldsymbol{d}$ are normalized surface normal and relative velocity direction respectively. Here we set the density of water $\rho = 1000\,\text{kg/m}^3$. $\phi = \pi/2 - \cos^{-1}(\boldsymbol{n} \cdot \boldsymbol{v}_{rel})$ denotes the attack angle that measures the angle of surface facet facing the fluid flow, and induces the coefficients C_{d} and C_{t}. We set these two coefficients to be the same for convenience:

$$C(\phi) = 2\sin^2\phi. \tag{44}$$

As for the optimization, since the drag and thrust forces are differentiable, their gradients can be straightforwardly incorporated in Equation (37) onto the term which is multiplied by Δt. In this example, we aim for exploring suitable magnetic actuation strategy to allow the octopus to spend minimum efforts to pass through the narrow channel within given time constraint. The objective function contains three terms, including (1) the distance between the octopus's mass center and the specified trajectory sample points; (2) the magnitude of the magnetic field, and (3) a DMP regularization term to enforce a low-frequency control signal [Pan and Manocha 2018]. For the frame at the moment t, the DMP term of its external magnetic field $\boldsymbol{B}_t$ is formulated as:

$$F_{\text{DMP}}(\boldsymbol{B}_t, t, \boldsymbol{\omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{i=1}^{3}\left(\boldsymbol{B}_t^i - \sum_{j=1}^{J}(\alpha_j \cos(\omega_j t) + \beta_j \sin(\omega_j t))\right)^2,$$

in which the indices i, j indicate the dimension of variable, and the number of trigonometric-function pairs respectively. We chose $J = 5$ in our practical implementation. The optimization variables include both the temporally varying magnetic field and the frequencies($\boldsymbol{\omega}$) and amplitudes($\boldsymbol{\alpha}, \boldsymbol{\beta}$) of the DMP regularizer. Our results show that our optimizer can discover effective swimming strategies to adapt the octopus' thin-shell body to the environment without any prior guidance. According to the comparison results shown in Figure 10, the octopus adopts a high-frequency strategy to squeeze out of the narrow tunnel by relying on the frictional contact force with the environment. In a different strategy, a more stretchable posture is employed to gain speed from the hydrodynamic force for the wider tunnel environment. We refer the reader to our supplementary video for more details. On an irregular tunnel terrain, the octopus takes varying swimming strategies that continuously adapt itself to the local environment and produce suitable propulsion forces to go through the tunnel more efficiently.

Kirigami Tower. Kirigami structures are featured by their large compression ratio which is specifically suited for a highly compact environment. In this example (Figure 13), we apply our quasi-static optimizer to optimizing the remanence magnetization distribution which could react to a given vertical uniform magnetic field and

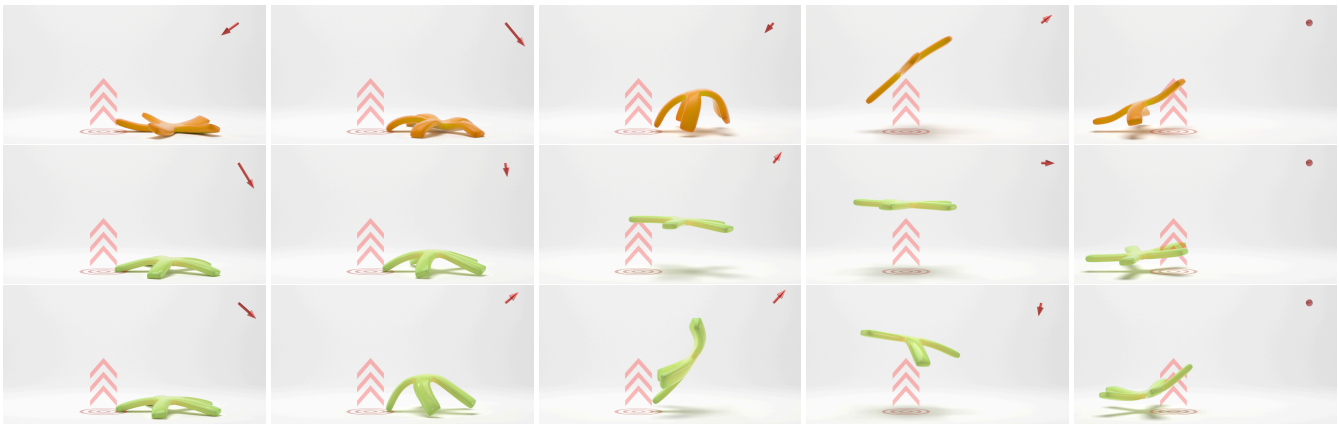


Fig. 15. Jumping starfish. The three rows show three different settings: the orange starfish in the top row is three times mass density as the green ones in the bottom two rows, and the third row conditions an objective function with less penalty of the magnitude of the external magnetic field. The starfish is set to jump over the pink arrows. With different settings, the starfish nevertheless successfully reaches the target, however through different maneuvers automatically generated through optimized magnetic fields.

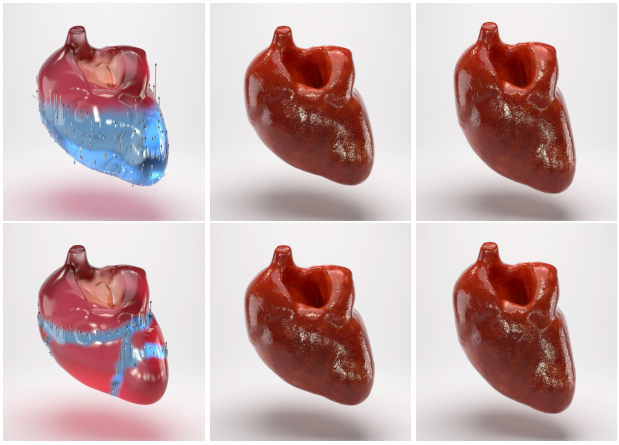


Fig. 16. Beating hearts. The left column shows the surfaces with magnetic material in blue (the lower half for the above one, and the ribbon area for the bottom one), with the silver arrows depicting the optimized magnetic remanence in these areas. The hearts get expanded (horizontally) and contracted (vertically elongated) responding to the optimized magnetic fields, shown in the middle and right columns, respectively.

drive the Kirigami structure stretching as much as possible in vertical direction. Next, by taking the derived remanence magnetization as known, we perform a trajectory optimization on the time-dependent external magnetic field for a fast-reaching task. Four target points (indicated using red ball in Figure 13) are randomly placed in the surrounding 3D space. They are placed consecutively at predefined time instances. The objective function measures the distance from Kirigami's center point to the active target point starting *onesecond* after the new target initialization, until the target is replaced. The total magnetic field strength is treated as a regularization term in our objective function. As the curve in Figure 13 shows, every 1 s after the target appears, the center of our Kirigami structure can reach the target point with extremely high accuracy. The tiny fluctuation of position error for the remaining time confirms that our optimized control policy can effectively suppress large vibration that would have been introduced by sudden maneuvers. Please refer to the supplementary video for visual evidence.

6.4 Volumetric Objects

Our magnetic-elastic model is not limited to simulations of thin-shell objects, but can also be integrated into any state-of-the-art FEM-based simulation framework to imitate volumetric soft objects with a magnetic thin layer, with the only change is for all the surface triangles, their potential energies need an extra term from magnetic side. This model provides new capability on realizing unthehered active exoskeleton control.

Gesturing Hands. As illustrated in Figure 14, we first experiment on a static shape-fitting problem. Initially, the volumetric hand model has a homogeneous elastic material distribution. Quasi-static optimizations are conducted to find three sets of remanence magnetization distribution on the hand surface, which allow the hand's rest shape to deform to three given gestures driven by a given vertical magnetic field. To alleviate the side effects caused by the distorted mesh discretization, we designate a set of anchor points on each finger and evaluate the shape similarity base on these points.This sparse position constrains help our optimizer converge to a more natural result.

Beating Hearts. As demonstrated in Figure 16, a heart exoskeleton is designed to enhance the heart's beating deformation. Similar to the previous example, we conduct a quasi-static optimization to find the optimized surface remanence magnetization distribution which can fit to the most contracted shape driven by a constant magnetic field. The only difference is that the surfaces allowed to maintain magnetic material are restricted to the areas highlighted using blue color in the left column of Figure 16. Through the comparison shown in Figure 16, with optimized remanence magnetization, the heart

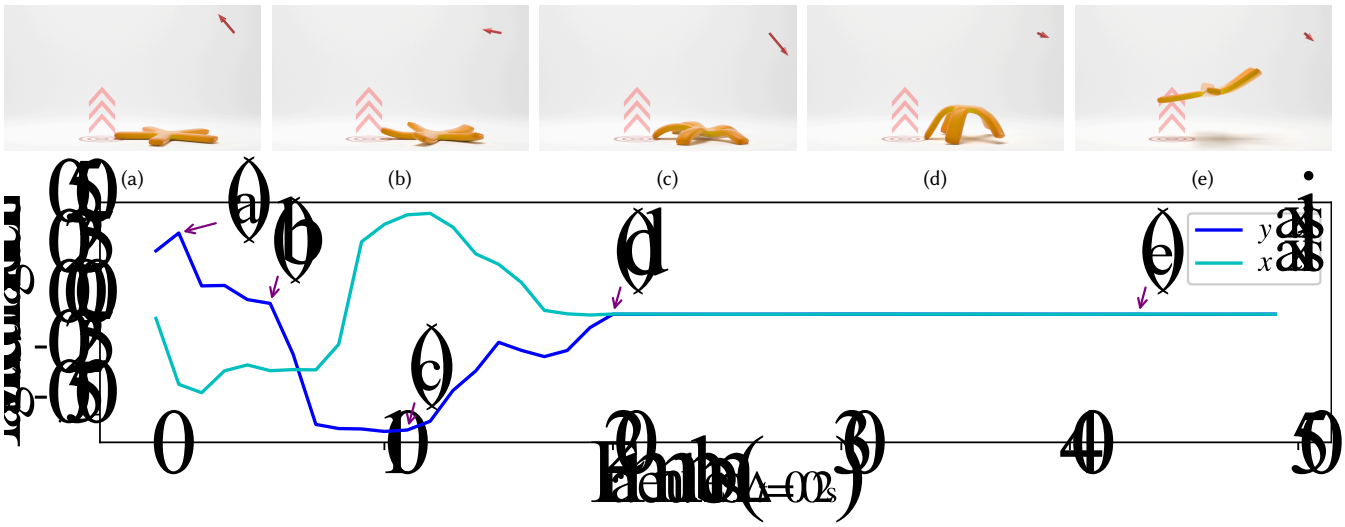


Fig. 17. The two curves show the applied external magnetic field in the first 50 frames that facilitates the orange starfish's jumping strategy, as depicted in the top row. The figures show the correlation between the external magnetic stimuli and the actions taken in the process of starfish's taking-off in details.

delivers a strong beating sequence driven by a periodically switching magnetic field.

Jumping Starfish. The starfish example in Figure 15 shows that our proposed trajectory optimization algorithm can be employed to deform soft objects through magnetic thin layers on their surfaces. The external magnetic field is optimized to make the starfish jump over an obstacle at designated time, depicted as pink arrows in Figure 15. The green and orange starfishes are identical with respect to their geometry and elastic material distribution, but the orange one is 3-times heavier than the green one. The differentiable frictional contact model mentioned in the Section 6.1 is adopted here as well. Through the results shown in Figure 15, the starfish is optimized to gain an initial take-off momentum through the frictional contact with the ground. This strategy is more obvious for the heavier starfish as shown in Figure 17; in the first stage ((a)–(b)), the starfish lifts its tentacles driven by an upward magnetic field and stores elastic potential energy in its body; then as the magnetic field turns downward, the stored elastic potential energy is quickly released, which makes the starfish suddenly flap its tentacles down to hit the ground with high speed. Such strong impact helps the starfish gain initial take-off momentum in both vertical and horizontal direction.

7 CONCLUSIONS AND DISCUSSIONS

We have proposed a novel computational framework to design magnetoelastic thin shells and demonstrated a broad array of applications. Our main contribution is finding an effective computational solution to support both forward simulation and inverse design tasks of a new category of physical objects that are on the periphery of the previous literature. For forward simulation, we developed the first continuum mechanics model based on the Kirchhoff–Love thin-shell model to characterize the behaviors of a magnetalelastic thin shell under external magnetic stimuli. The underlying constitutive model decouples the overall potential energy into its elastic and magnetic components, and the magneto-mechanical coupling arises only from the deformation-induced variation of the remanent magnetization in the applied magnetic field. Based on this model, we provide a complete numerical recipe, which includes the discretization formula and the Hessian matrix derivation. Due to its conciseness, our method can be easily integrated into the existing finite-element thin-shell framework to support novel magnetic phenomena simulations. For the inverse design problems, we build a fully differentiable simulation framework, in conjunction with its adjoint solvers, to support a plethora of design tasks, ranging from magnetoelastic soft robots, functional Origami, to artworks and metamaterial designs. For both static and dynamic PDE-constraint problems, our differentiable solver improves the optimization performance on magnetoelastic thin-shell structures.

The ideal deformation behaviour of magnetoelastic material is nonlinear, incompressible, and time/strain rate-dependent. Our proposed magnetoelastic constitutive model possesses many simplifications. The hysteresis loop is not considered in our model. A conventional finite element frame like what we employed in this paper has great difficulties on modeling incompressible deformations. Moreover, when the applied magnetic field is stronger than the coercivity of the immersed magnetic material, the proposed continuum mechanics model will no longer be applicable, because the nonlinear magnetization is not negligible anymore. Last but not the least, the influence of the magnetization on elastic materials property change should also be involved into the current framework. Currently, we use a gradient-based method MMA for parameter fitting. We plan to explore other optimization schemes to avoid any local minimal trap.

Because the relevant facility has limited accessibility, this work focuses on tackling the computational challenges and aims to provide a simulation tool to open the stage for the potentially close collaboration among graphics, design, and fabrication. Nevertheless, it is worth discussing here the feasibility of our optimization approaches

in physical-world fabrication. An abrupt residual magnetization can be realized by direct ink writing of an elastomer composite containing ferromagnetic microparticles [Kim et al. 2018]. The spatially varying bending stiffness can be realized by embedding indentations and local structures, which has been a common practice in Origami design and fabrication. From the perspective of algorithm itself, additional regularization terms which penalize the abrupt material or magnetic field change need to be involved into the objective function to ensure the design's fabricability.

Another possible topic for further investigation is incorporating topological optimization into the current framework, which will facilitate more complex structures and functionalities of the design. At the same time, we plan to explore high-performance implementation of our thin-shell solver in order to support large-scale topological optimization applications. This is a challenging problem to solve because the dimension of the design space would be significantly enlarged. Lastly, we are also considering incorporating our fully differentiable simulator and the accompanied adjoint solvers into the reinforcement learning frameworks, which can extend our simulator to accommodate the various control policy learning tasks.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their constructive comments. This work was supported in part by NSFC Projects of International Cooperation and Exchanges (62161146002); Shenzhen Collaborative Innovation Program (CJGJZD2021048092601003). Bo Zhu acknowledges supports from NSF-1919647, 2106733, and 2144806. We credit the Houdini Education license for producing the video animations.

REFERENCES

Jernej Barbič, Funshing Sin, and Eitan Grinspun. 2012. Interactive editing of deformable simulations. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–8.

Miklos Bergou, Max Wardetzky, David Harmon, Denis Zorin, and Eitan Grinspun. 2006. Discrete Quadratic Curvature Energies. In *ACM SIGGRAPH 2006 Courses* (Boston, Massachusetts) *(SIGGRAPH '06)*. Association for Computing Machinery, New York, NY, USA, 20–29.

R. Bridson, S. Marino, and R. Fedkiw. 2003. Simulation of Clothing with Folds and Wrinkles. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (San Diego, California) *(SCA '03)*. Eurographics Association, Goslar, DEU, 28–36.

Jared T Bruton, Todd G Nelson, Trent K Zimmerman, Janette D Fernelius, Spencer P Magleby, and Larry L Howell. 2016. Packing and deploying Soft Origami to and from cylindrical volumes with application to automotive airbags. *Royal Society open science* 3, 9 (2016), 160429.

Oleksiy Busaryev, Tamal K Dey, and Huamin Wang. 2013. Adaptive fracture simulation of multi-layered thin plates. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–6.

Hsiao-Yu Chen, Arnav Sastry, Wim M. van Rees, and Etienne Vouga. 2018. Physical Simulation of Environmentally Induced Thin Shell Deformation. *ACM Trans. Graph.* 37, 4, Article 146 (jul 2018), 13 pages.

Zhen Chen, Hsiao-Yu Chen, Danny M. Kaufman, Mélina Skouras, and Etienne Vouga. 2021. Fine Wrinkling on Coarsely Meshed Thin Shells. *ACM Trans. Graph.* 40, 5, Article 190 (aug 2021), 32 pages.

H Christopher Frey and Sumeet R Patil. 2002. Identification and review of sensitivity analysis methods. *Risk analysis* 22, 3 (2002), 553–578.

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 (2000), 2039–2072.

Xiangxin Dang, Fan Feng, Paul Plucinsky, Richard D James, Huiling Duan, and Jianxiang Wang. 2022. Inverse design of deployable origami structures that approximate a general surface. *International Journal of Solids and Structures* 234 (2022), 111224.

A Dorfmann and RW2008258 Ogden. 2003. Magnetoelastic modelling of elastomers. *European Journal of Mechanics-A/Solids* 22, 4 (2003), 497–507.

Luis Dorfmann and Ray W. Ogden. 2014. *Nonlinear Theory of Electroelastic and Magnetoelastic Interactions.* Springer, Heidelberg, Germany.

Tao Du, Kui Wu, Pingchuan Ma, Sebastien Wah, Andrew Spielberg, Daniela Rus, and Wojciech Matusik. 2021. DiffPD: Differentiable Projective Dynamics. *ACM Trans. Graph.* 41, 2, Article 13 (Oct. 2021), 21 pages.

Tao Du, Kui Wu, Andrew Spielberg, Wojciech Matusik, Bo Zhu, and Eftychios Sifakis. 2020. Functional Optimization of Fluidic Devices with Differentiable Stokes Flow. *ACM Trans. Graph.* 39, 6, Article 197 (Dec. 2020), 15 pages.

Levi H Dudte, Etienne Vouga, Tomohiro Tachi, and Lakshminarayanan Mahadevan. 2016. Programming curvature using origami tessellations. *Nature materials* 15, 5 (2016), 583–588.

Mehdi Eshaghi, Mohsen Ghasemi, and Korosh Khorshidi. 2021. Design, manufacturing and applications of small-scale magnetic soft robots. *Extreme Mechanics Letters* 44 (2021), 101268.

C. Daniel Freeman, Erik Frey, Anton Raichuk, Sertan Girgin, Igor Mordatch, and Olivier Bachem. 2021. Brax – A Differentiable Physics Engine for Large Scale Rigid Body Simulation. arXiv:2106.13281 [cs.RO]

Daniel Garcia-Gonzalez. 2019. Magneto-visco-hyperelasticity for hard-magnetic soft materials: theory and numerical applications. *Smart Materials and Structures* 28, 8 (2019), 085020.

Moritz Geilinger, David Hahn, Jonas Zehnder, Moritz Niklaus Bacher, B. Thomaszewski, and Stelian Coros. 2020. ADD: Analytically Differentiable Dynamics for Multi-Body Systems with Frictional Contact. *ACM Trans. Graph.* 39 (2020), 190:1–190:15.

Ali Ghaffari, Seyed Hassan Hashemabadi, and Mansour Bazmi. 2015. CFD simulation of equilibrium shape and coalescence of ferrofluid droplets subjected to uniform magnetic field. *Colloids and Surfaces A: Physicochemical and Engineering Aspects* 481 (2015), 186–198.

Yotam Gingold, Adrian Secord, Jefferson Y Han, Eitan Grinspun, and Denis Zorin. 2004. A discrete model for inelastic deformation of thin shells. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Grenoble, France). Eurographics Association, Goslar, DEU, 1–12.

A.E. Green and P.M. Naghdi. 1968. The linear elastic cosserat surface and shell theory. *International Journal of Solids and Structures* 4, 6 (1968), 585–592.

Eitan Grinspun, Yotam Gingold, Jason Reisman, and Denis Zorin. 2006. Computing discrete shape operators on general meshes. *Computer Graphics Forum* 25, 3 (2006), 547–556.

Eitan Grinspun, Anil N. Hirani, Mathieu Desbrun, and Peter Schröder. 2003. Discrete Shells. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (San Diego, California) *(SCA '03)*. Eurographics Association, Goslar, DEU, 62–67.

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 Transactions on Graphics (TOG)* 37, 4 (2018), 1–15.

David Hahn, Pol Banzet, James Bern, and Stelian Coros. 2019. Real2Sim: visco-elastic parameter estimation from dynamic motion. *ACM Trans. Graph.* 38 (11 2019), 1–13.

Philipp Holl, Vladlen Koltun, Kiwon Um, and Nils Thuerey. 2020. phiflow: A differentiable pde solving framework for deep learning via physical simulations. In *Thirty-fourth Workshop on Neural Information Processing Systems*. The Neural Information Processing Systems Foundation, Virtual, 1–5.

Wenqi Hu, Guo Zhan Lum, Massimo Mastrangeli, and Metin Sitti. 2018. Small-scale soft-bodied robot with multimodal locomotion. *Nature* 554, 7690 (2018), 81–85.

Yuanming Hu, Jiancheng Liu, Andrew Spielberg, Joshua B. Tenenbaum, William T. Freeman, Jiajun Wu, Daniela Rus, and Wojciech Matusik. 2019. ChainQueen: A Real-Time Differentiable Physical Simulator for Soft Robotics. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, Montreal, Canada, 6265–6271.

Libo Huang, Torsten Hädrich, and Dominik L. Michels. 2019. On the Accurate Large-scale Simulation of Ferrofluids. *ACM Trans. Graph.* 38, 4, Article 93 (July 2019), 15 pages.

Libo Huang and Dominik L. Michels. 2020. Surface-Only Ferrofluids. *ACM Trans. Graph.* 39, 6, Article 174 (Nov. 2020), 17 pages.

Tomokazu Ishikawa, Yonghao Yue, Kei Iwasaki, Yoshinori Dobashi, and Tomoyuki Nishita. 2013. Visual Simulation of Magnetic Fluid Using a Procedural Approach for Spikes Shape. In *Computer Vision, Imaging and Computer Graphics. Theory and Application*. Springer Berlin Heidelberg, Berlin, Heidelberg, 112–126.

Shunta Kashima, Fumikazu Miyasaka, and Katsuhiro Hirata. 2012. Novel soft actuator using magnetorheological elastomer. *IEEE Transactions on magnetics* 48, 4 (2012), 1649–1652.

Seung-wook Kim and JungHyun Han. 2020. Simulation of Arbitrarily-shaped Magnetic Objects. *Computer Graphics Forum* 39, 7 (2020), 119–130.

Seung-Wook Kim, Sun Young Park, and Junghyun Han. 2018. Magnetization Dynamics for Magnetic Object Interactions. *ACM Trans. Graph.* 37, 4, Article 121 (July 2018), 13 pages.

Yoonho Kim, German A. Parada, Shengduo Liu, and Xuanhe Zhao. 2019. Ferromagnetic soft continuum robots. *Science Robotics* 4, 33 (2019), eaax7329.

Siwang Li, Jin Huang, Fernando de Goes, Xiaogang Jin, Hujun Bao, and Mathieu Desbrun. 2014. Space-time editing of elastic motion through material optimization and reduction. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–10.

Junbang Liang, Ming Lin, and Vladlen Koltun. 2019. Differentiable Cloth Simulation for Inverse Problems. In *Advances in Neural Information Processing Systems*, Vol. 32. Curran Associates, Inc., Vancouver, Canada.

Ke Liu, Felix Hacker, and Chiara Daraio. 2021. Robotic surfaces with reversible, spatiotemporal control for shape morphing and object manipulation. *Science Robotics* 6, 53 (2021), Art–No.

Mickaël Ly, Romain Casati, Florence Bertails-Descoubes, Mélina Skouras, and Laurence Boissieux. 2018. Inverse elastic shell design with contact and friction. *ACM Transactions on Graphics (TOG)* 37, 6 (2018), 1–16.

Pingchuan Ma, Tao Du, John Z Zhang, Kui Wu, Andrew Spielberg, Robert K Katzschmann, and Wojciech Matusik. 2021. DiffAqua: A Differentiable Computational Design Pipeline for Soft Underwater Swimmers with Shape Interpolation. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 132.

Luigi Malomo, Jesús Pérez, Emmanuel Iarussi, Nico Pietroni, Eder Miguel, Paolo Cignoni, and Bernd Bickel. 2018. FlexMaps: Computational Design of Flat Flexible Shells for Shaping 3D Objects. *ACM Trans. Graph.* 37, 6, Article 241 (dec 2018), 14 pages. https://doi.org/10.1145/3272127.3275076

Pierre-Luc Manteaux, Wei-Lun Sun, François Faure, Marie-Paule Cani, and James F. O'Brien. 2015. Interactive Detailed Cutting of Thin Sheets. In *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games* (Paris, France) *(MIG '15)*. Association for Computing Machinery, New York, NY, USA, 125–132.

Sebastian Martin, Peter Kaufmann, Mario Botsch, Eitan Grinspun, and Markus Gross. 2010. Unified Simulation of Elastic Rods, Shells, and Solids. *ACM Trans. Graph.* 29, 4, Article 39 (jul 2010), 10 pages.

Antoine McNamara, Adrien Treuille, Zoran Popović, and Jos Stam. 2004. Fluid Control Using the Adjoint Method. In *ACM SIGGRAPH 2004 Papers (SIGGRAPH '04)*. Association for Computing Machinery, New York, NY, USA, 449–456.

Sehee Min, Jungdam Won, Seunghwan Lee, Jungnam Park, and Jehee Lee. 2019. SoftCon. *ACM Trans. Graph.* 38 (2019), 1–12.

Rahul Narain, Tobias Pfaff, and James F. O'Brien. 2013. Folding and Crumpling Adaptive Sheets. *ACM Trans. Graph.* 32, 4, Article 51 (jul 2013), 8 pages.

Rahul Narain, Armin Samii, and James F. O'Brien. 2012. Adaptive Anisotropic Remeshing for Cloth Simulation. *ACM Trans. Graph.* 31, 6, Article 152 (nov 2012), 10 pages.

Xingyu Ni, Bo Zhu, Bin Wang, and Baoquan Chen. 2020. A Level-Set Method for Magnetic Substance Simulation. *ACM Trans. Graph.* 39, 4, Article 29 (jul 2020), 15 pages.

Larissa S Novelino, Qiji Ze, Shuai Wu, Glaucio H Paulino, and Ruike Zhao. 2020. Untethered control of functional origami microrobots with distributed actuation. *Proceedings of the National Academy of Sciences* 117, 39 (2020), 24096–24101.

Zherong Pan and Dinesh Manocha. 2018. Active Animations of Reduced Deformable Models with Environment Interactions. *ACM Trans. Graph.* 37, 3, Article 36 (aug 2018), 17 pages.

Julian Panetta, Florin Isvoranu, Tian Chen, Emmanuel Siéfert, Benoît Roman, and Mark Pauly. 2021. Computational Inverse Design of Surface-Based Inflatables. *ACM Trans. Graph.* 40, 4, Article 40 (jul 2021), 14 pages. https://doi.org/10.1145/3450626.3459789

J. Panetta, M. Konaković-Luković, F. Isvoranu, E. Bouleau, and M. Pauly. 2019. X-Shells: A New Class of Deployable Beam Structures. *ACM Trans. Graph.* 38, 4, Article 83 (jul 2019), 15 pages. https://doi.org/10.1145/3306346.3323040

Jesús Pérez, Miguel A. Otaduy, and Bernhard Thomaszewski. 2017. Computational Design and Automated Fabrication of Kirchhoff-Plateau Surfaces. *ACM Trans. Graph.* 36, 4, Article 62 (jul 2017), 12 pages. https://doi.org/10.1145/3072959.3073695

Matteo Pezzulla, Dong Yan, and Pedro M. Reis. 2021. A geometrically exact model for thin magneto-elastic shells.

Tobias Pfaff, Rahul Narain, Juan Miguel De Joya, and James F O'Brien. 2014. Adaptive tearing and cracking of thin sheets. *ACM Trans. Graph.* 33, 4 (2014), 1–9.

Yi-Ling Qiao, Junbang Liang, Vladlen Koltun, and Ming Lin. 2020. Scalable Differentiable Physics for Learning and Control. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, Virtual, 7847–7856.

Olivier Rémillard and Paul G Kry. 2013. Embedded thin shells for wrinkle simulation. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–8.

Connor Schenck and Dieter Fox. 2018. SPNets: Differentiable Fluid Dynamics for Deep Neural Networks. In *Proceedings of The 2nd Conference on Robot Learning (Proceedings of Machine Learning Research, Vol. 87)*. PMLR, Zrich, Switzerland, 317–335.

Christian Schulz, Christoph von Tycowicz, Hans-Peter Seidel, and Klaus Hildebrandt. 2014. Animating deformable objects using sparse spacetime constraints. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–10.

Georg Sperl, Rahul Narain, and Chris Wojtan. 2020. Homogenized yarn-level cloth. *ACM Trans. Graph.* 39, 4 (2020), 48.

Yuchen Sun, Xingyu Ni, Bo Zhu, Bin Wang, and Baoquan Chen. 2021. A Material Point Method for Nonlinearly Magnetized Materials. *ACM Trans. Graph.* 40, 6, Article 205 (dec 2021), 13 pages.

Krister Svanberg. 2002. A Class of Globally Convergent Optimization Methods Based on Conservative Convex Separable Approximations. *SIAM Journal on Optimization* 12 (2002), 555–573.

Rasmus Tamstorf and Eitan Grinspun. 2013. Discrete Bending Forces and Their Jacobians. *Graph. Models* 75, 6 (nov 2013), 362–370.

Demetri Terzopoulos, John Platt, Alan Barr, and Kurt Fleischer. 1987. Elastically Deformable Models. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '87)*. Association for Computing Machinery, New York, NY, USA, 205–214.

Bernhard Thomaszewski, Andreas Gumann, Simon Pabst, and Wolfgang Straßer. 2008. Magnets in Motion. *ACM Trans. Graph.* 27, 5, Article 162 (Dec. 2008), 9 pages.

Wim M van Rees, Etienne Vouga, and Lakshminarayanan Mahadevan. 2017. Growth patterns for shape-shifting elastic bilayers. *Proceedings of the National Academy of Sciences* 114, 44 (2017), 11597–11602.

Bin Wang, Longhua Wu, KangKang Yin, Uri Ascher, Libin Liu, and Hui Huang. 2015. Deformation Capture and Modeling of Soft Objects. *ACM Trans. Graph.* 34, 4, Article 94 (jul 2015), 12 pages.

Liu Wang, Yoonho Kim, Chuan Fei Guo, and Xuanhe Zhao. 2020a. Hard-magnetic elastica. *Journal of the Mechanics and Physics of Solids* 142 (2020), 104045.

Weiming Wang, Dirk Munro, Charlie CL Wang, Fred van Keulen, and Jun Wu. 2020b. Space-time topology optimization for additive manufacturing. *Structural and Multidisciplinary Optimization* 61, 1 (2020), 1–18.

Keenon Werling, Dalton Omens, Jeongseok Lee, Ioannis Exarchos, and C Karen Liu. 2021. Fast and Feature-Complete Differentiable Physics Engine for Articulated Rigid Bodies with Contact Constraints. In *Proceedings of Robotics: Science and Systems*. The RSS Foundation, Virtual, 15.

Andrew Witkin and Michael Kass. 1988. Spacetime Constraints. In *Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '88)*. Association for Computing Machinery, New York, NY, USA, 159–168.

Tianqi Xu, Jiachen Zhang, Mohammad Salehizadeh, Onaizah Onaizah, and Eric Diller. 2019. Millimeter-scale flexible robots with programmable three-dimensional magnetization and motions. *Science Robotics* 4, 29 (2019), eaav4494.

Dong Yan, Matteo Pezzulla, Lilian Cruveiller, Arefeh Abbasi, and Pedro M Reis. 2021. Magneto-active elastic shells with tunable buckling strength. *Nature communications* 12, 1 (2021), 1–9.

Yi Yang, Katherine Vella, and Douglas P Holmes. 2021. Grasping with kirigami shells. *Science Robotics* 6, 54 (2021), eabd6426.

Ruike Zhao, Yoonho Kim, Shawn A. Chester, Pradeep Sharma, and Xuanhe Zhao. 2019. Mechanics of hard-magnetic soft materials. *Journal of the Mechanics and Physics of Solids* 124 (2019), 244–263.

Xuanhe Zhao, Jaeyun Kim, Christine A Cezar, Nathaniel Huebsch, Kangwon Lee, Kamal Bouhadir, and David J Mooney. 2011. Active scaffolds for on-demand drug and cell delivery. *Proceedings of the National Academy of Sciences* 108, 1 (2011), 67–72.

Bo Zhu, Mélina Skouras, Desai Chen, and Wojciech Matusik. 2017. Two-Scale Topology Optimization with Microstructures. *ACM Trans. Graph.* 36, 5, Article 164 (jul 2017), 16 pages.

Yufeng Zhu, Robert Bridson, and Chen Greif. 2015. Simulating Rigid Body Fracture with Surface Meshes. *ACM Trans. Graph.* 34, 4, Article 150 (jul 2015), 11 pages.

Shannon A. Zirbel, Robert J. Lang, Mark W. Thomson, Deborah A. Sigel, Phillip E. Walkemeyer, Brian P. Trease, Spencer P. Magleby, and Larry L. Howell. 2013. Accommodating thickness in origami-based deployable arrays. *Journal of Mechanical Design* 135, 11 (2013), 11.

A PHYSICAL ANALYSIS

A.1 Lagrangian Formulations of Magnetostatics

We integrate Equation (9) over an arbitrary volume $\mathcal{D}$ with suitably regular boundary $\partial\mathcal{D}$ and apply *Gauss's theorem* to obtain

$$\int_{\partial\mathcal{D}} \boldsymbol{B}\cdot\boldsymbol{n}\,\mathrm{d}S = \int_{\mathcal{D}} \nabla\cdot\boldsymbol{B}\,\mathrm{d}V = 0. \tag{45}$$

Considering that $\boldsymbol{n}\,\mathrm{d}S = J\boldsymbol{F}^{-\mathrm{T}}\tilde{\boldsymbol{n}}\,\mathrm{d}\tilde{S}$, we acquire

$$\int_{\partial\mathcal{D}} \boldsymbol{B}\cdot\boldsymbol{n}\,\mathrm{d}S = \int_{\partial\tilde{\mathcal{D}}} \boldsymbol{B}\cdot J\boldsymbol{F}^{-\mathrm{T}}\tilde{\boldsymbol{n}}\,\mathrm{d}\tilde{S} = \int_{\partial\tilde{\mathcal{D}}} J\boldsymbol{F}^{-1}\boldsymbol{B}\cdot\tilde{\boldsymbol{n}}\,\mathrm{d}\tilde{S}, \tag{46}$$

which means

$$\int_{\partial\tilde{\mathcal{D}}} J\boldsymbol{F}^{-1}\boldsymbol{B}\cdot\tilde{\boldsymbol{n}}\,\mathrm{d}\tilde{S} = \int_{\tilde{\mathcal{D}}} \tilde{\nabla}\cdot J\boldsymbol{F}^{-1}\boldsymbol{B}\,\mathrm{d}\tilde{V} = 0 \tag{47}$$

by Gauss's theorem. Since $\mathcal{D}$ is also an arbitrary volume, the equation

$$\tilde{\nabla}\cdot J\boldsymbol{F}^{-1}\boldsymbol{B} = 0 \tag{48}$$

holds, which suggests us to define $\tilde{\boldsymbol{B}} = J\boldsymbol{F}^{-1}\boldsymbol{B}$.

Similarly, we integrate Equation (10) over an arbitrary (but suitably regular) open surface $\mathcal{S}$ and then apply *Stoke's theorem* to obtain

$$\int_{\partial\mathcal{S}} \boldsymbol{H}\cdot \mathrm{d}\boldsymbol{x} = \int_{\mathcal{S}} (\nabla\times\boldsymbol{H})\cdot\boldsymbol{n}\,\mathrm{d}S = 0. \tag{49}$$

Considering that $\mathrm{d}\boldsymbol{x} = \boldsymbol{F}\mathrm{d}\tilde{\boldsymbol{x}}$, we acquire

$$\int_{\partial\mathcal{S}} \boldsymbol{H}\cdot \mathrm{d}\boldsymbol{x} = \int_{\partial\tilde{\mathcal{S}}} \boldsymbol{H}\cdot\boldsymbol{F}\mathrm{d}\tilde{\boldsymbol{x}} = \int_{\partial\tilde{\mathcal{S}}} \boldsymbol{F}^{\mathrm{T}}\boldsymbol{H}\cdot \mathrm{d}\tilde{\boldsymbol{x}}, \tag{50}$$

which means

$$\int_{\partial\tilde{\mathcal{S}}} \boldsymbol{F}^{\mathrm{T}}\boldsymbol{H}\cdot \mathrm{d}\tilde{\boldsymbol{x}} = \int_{\tilde{\mathcal{S}}} (\tilde{\nabla}\times\boldsymbol{F}^{\mathrm{T}}\boldsymbol{H})\cdot\tilde{\boldsymbol{n}}\,\mathrm{d}\tilde{S} = 0 \tag{51}$$

by Stoke's theorem. Since $\mathcal{S}$ is also an arbitrary open surface, the equation

$$\tilde{\nabla}\times\boldsymbol{F}^{\mathrm{T}}\boldsymbol{H} = 0 \tag{52}$$

holds, which suggests us to define $\tilde{\boldsymbol{H}} = \boldsymbol{F}^{\mathrm{T}}\boldsymbol{H}$.

As to the magnetization intensity $\boldsymbol{M}$, since the *effective magnetic charge density* $\rho_{\mathrm{M}} = -\nabla\cdot\boldsymbol{M}$ is conservative ($\tilde{\rho}_{\mathrm{M}} = J\rho_{\mathrm{M}}$), there exists a conservation equation as follows:

$$\tilde{\nabla}\cdot\tilde{\boldsymbol{M}} = J\nabla\cdot\boldsymbol{M}. \tag{53}$$

Observing the above derivation of $\tilde{\boldsymbol{B}} = J\boldsymbol{F}^{-1}\boldsymbol{B}$, it is not hard to see that

$$\tilde{\nabla}\cdot\tilde{\boldsymbol{A}} = J\nabla\cdot\boldsymbol{A} = J\nabla\cdot J^{-1}\boldsymbol{F}\tilde{\boldsymbol{A}} \tag{54}$$

holds for any vector field $\boldsymbol{A}$, due to $\mathrm{d}V = J\mathrm{d}\tilde{V}$. For this reason, it is natural to define $\tilde{\boldsymbol{M}} = J\boldsymbol{F}^{-1}\boldsymbol{M}$.

More detailed derivations and applications can be seen in the work of Dorfmann and Ogden [2014].

A.2 Derivatives in Forward Simulation

The Jacobian and Hessian terms of discrete thin-shell elastic energy have been introduced in previous work, and we refer readers to the work of Grinspun et al. [2006] or the work of Tamstorf and Grinspun [2013] for a clear derivation. Next, we will concentrate on the derivatives of discrete thin-shell hard-magnetic energy, which is novel for the community of computer graphics.

For a triangle face with its three vertices located at $\tilde{\boldsymbol{x}}_0, \tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2$ in the material space, and $\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2$ in the world space after deformation, we define

$$\tilde{\boldsymbol{X}} = \begin{pmatrix}\tilde{\boldsymbol{x}}_1 - \tilde{\boldsymbol{x}}_0 & \tilde{\boldsymbol{x}}_2 - \tilde{\boldsymbol{x}}_0\end{pmatrix}, \tag{55}$$

$$\boldsymbol{X} = \begin{pmatrix}\boldsymbol{x}_1 - \boldsymbol{x}_0 & \boldsymbol{x}_2 - \boldsymbol{x}_0\end{pmatrix}. \tag{56}$$

Then the deformation gradient can be represented as

$$\boldsymbol{F} = \boldsymbol{X}(\tilde{\boldsymbol{X}}^{\mathrm{T}}\tilde{\boldsymbol{X}})^{-1}\tilde{\boldsymbol{X}}^{\mathrm{T}} + \boldsymbol{n}\tilde{\boldsymbol{n}}^{\mathrm{T}}. \tag{57}$$

Substituting $\boldsymbol{F}$ into Equation (24), we obtain

$$\begin{aligned}\varepsilon_{\mathrm{magnetism}} &= -\tilde{A}h\left(\boldsymbol{X}\left(\tilde{\boldsymbol{X}}^{\mathrm{T}}\tilde{\boldsymbol{X}}\right)^{-1}\tilde{\boldsymbol{X}}^{\mathrm{T}} + \boldsymbol{n}\tilde{\boldsymbol{n}}^{\mathrm{T}}\right)\tilde{\boldsymbol{M}}_{\mathrm{r}}\cdot\boldsymbol{B}_{\mathrm{applied}}\\ &= -\tilde{A}h\boldsymbol{X}\left(\tilde{\boldsymbol{X}}^{\mathrm{T}}\tilde{\boldsymbol{X}}\right)^{-1}\tilde{\boldsymbol{X}}^{\mathrm{T}} - \tilde{A}h\boldsymbol{n}\tilde{\boldsymbol{n}}^{\mathrm{T}}\tilde{\boldsymbol{M}}_{\mathrm{r}}\cdot\boldsymbol{B}_{\mathrm{applied}}.\end{aligned} \tag{58}$$

Equation (58) actually separates the magnetic energy into two parts, with the first part denoting the part contributed by remanence component parallel to the face while the second part denoting the part contributed by remanence component perpendicular to the face. Denoting the two parts of magnetic energy as ε_1, ε_2 respectively, then $\varepsilon_{\mathrm{magnetism}} = \varepsilon_1 + \varepsilon_2$. Use $[\boldsymbol{e}]$ to denote the cross product matrix of the 3D vector $\boldsymbol{e}$, and $\mathrm{vec}(\boldsymbol{E})$ to denote the corresponding vector of the matrix $\boldsymbol{E}$ concatenated by column, and additionally define

$$\boldsymbol{P} = \frac{1}{2A}\left(\boldsymbol{B}_{\mathrm{applied}} - \boldsymbol{n}\boldsymbol{B}_{\mathrm{applied}}^{\mathrm{T}}\boldsymbol{n}\right), \tag{59}$$

$$\boldsymbol{Q} = \begin{pmatrix}[\boldsymbol{x}_2 - \boldsymbol{x}_0] & -[\boldsymbol{x}_1 - \boldsymbol{x}_0]\end{pmatrix}^{\mathrm{T}}. \tag{60}$$

Here $\boldsymbol{P}\in\mathbb{R}^{3\times1}$, $\boldsymbol{Q}\in\mathbb{R}^{6\times3}$. Taking the derivative of the above equation, we acquire their Jacobian terms as

$$\frac{\mathrm{d}\varepsilon_1}{\mathrm{d}\boldsymbol{X}} = -\tilde{A}h\left(\left(\tilde{\boldsymbol{X}}^{\mathrm{T}}\tilde{\boldsymbol{X}}\right)^{-1}\tilde{\boldsymbol{X}}^{\mathrm{T}}\tilde{\boldsymbol{M}}_{\mathrm{r}}\boldsymbol{B}_{\mathrm{applied}}^{\mathrm{T}}\right)^{\mathrm{T}}, \tag{61}$$

$$\frac{\mathrm{d}\varepsilon_2}{\mathrm{d}\,\mathrm{vec}(\boldsymbol{X})} = -\tilde{A}h\boldsymbol{Q}\boldsymbol{P}. \tag{62}$$

Equation (61) has nothing to do with $\boldsymbol{X}$, thus the Hessian matrix is only yielded from the second term of Equation (62):

$$\begin{aligned}\frac{\mathrm{d}^2\varepsilon_{\mathrm{magnetism}}}{\mathrm{d}\,\mathrm{vec}(\boldsymbol{X})^2} =& \frac{\mathrm{d}^2\varepsilon_2}{\mathrm{d}\,\mathrm{vec}(\boldsymbol{X})^2}\\ =& -\tilde{A}h\left(\tilde{\boldsymbol{n}}\cdot\boldsymbol{B}_{\mathrm{applied}}\right)\left(\boldsymbol{Q}\Big(\frac{3}{4A^2}\boldsymbol{n}\boldsymbol{B}_{\mathrm{applied}}^{\mathrm{T}}\boldsymbol{n}\boldsymbol{n}^{\mathrm{T}}\right.\\ &-\frac{1}{4A^2}\left(\boldsymbol{n}\boldsymbol{B}_{\mathrm{applied}}^{\mathrm{T}} + \boldsymbol{B}_{\mathrm{applied}}\boldsymbol{n}^{\mathrm{T}} - \boldsymbol{B}_{\mathrm{applied}}^{\mathrm{T}}\boldsymbol{n}\boldsymbol{I}\right)\Big)\boldsymbol{Q}^{\mathrm{T}}\\ &\left.+\begin{pmatrix}\boldsymbol{0} & [\boldsymbol{P}]\\ [\boldsymbol{P}]^{\mathrm{T}} & \boldsymbol{0}\end{pmatrix}\right),\end{aligned} \tag{63}$$

where $\boldsymbol{I}$ denotes the 3×3 identity matrix.