SIAM J. SCI. COMPUT.
Vol. 38, No. 1, pp. B50–B76

CONSTRAINED OPTIMIZATION FOR LIQUID CRYSTAL EQUILIBRIA*

J. H. ADLER†, D. B. EMERSON†, S. P. MACLACHLAN‡, AND T. A. MANTEUFFEL§

Abstract. This paper compares the performance of penalty and Lagrange multiplier approaches for the necessary unit-length constraint in the computation of liquid crystal equilibrium configurations. Building on previous work in [*SIAM J. Sci. Comput.*, 37 (2015), pp. S157–S176; *SIAM J. Numer. Anal.*, 53 (2015), pp. 2226–2254], the penalty method is derived and well-posedness of the linearizations within the nonlinear iteration is discussed. In addition, the paper considers the effects of tailored trust-region methods in the context of finite-element discretizations and nested iteration for both formulations. Such methods are aimed at increasing the efficiency and robustness of each algorithm's nonlinear iterations. Three representative elastic equilibrium problems are considered to examine each method's performance. The first two configurations have analytical expressions for their exact solutions and, therefore, convergence to the true solution is considered. The third problem considers complicated boundary conditions, relevant in ongoing research, simulating surface nano-patterning. Finally, a novel multigrid scheme is introduced and tested for electrically and flexoelectrically coupled models to establish scalability for highly complicated applications. The Lagrange multiplier method is found to outperform the penalty method in a number of measures, the developed trust regions are shown to improve robustness, and nested iteration proves highly effective at reducing computational costs.

Key words. nematic liquid crystals, finite-element methods, Newton linearization, energy optimization, penalty methods, trust regions, multigrid methods

AMS subject classifications. 76A15, 65N30, 49M15, 65N22, 90C30, 65K10, 65N55

DOI. 10.1137/141001846

1. Introduction.

Liquid crystals are substances that possess mesophases with characteristics spanning those of isotropic liquids and solid crystals. That is, liquid crystals are fluid yet exhibit long-range structured ordering. This paper considers nematic liquid crystals which consist of rod-like molecules whose average pointwise orientation is represented by a vector, $\mathbf{n}(x, y, z) = (n_1, n_2, n_3)^T$. This orientation vector is known as the director and is assumed to be headless for nematics. Therefore, $\mathbf{n}$ and $-\mathbf{n}$ are indistinguishable at any point in the domain, Ω, due to molecular symmetry. An important constraint on the director vector field is that $\mathbf{n}$ remain of unit length pointwise throughout Ω. Thorough overviews of liquid crystal physics are found in [21, 55, 59].

The deformable ordering of liquid crystal structures, coupled with the materials' birefringent and dielectric properties, has led to many important applications and discoveries, most famously in display technologies. Additional modern applica-

*Submitted to the journal's Computational Methods in Science and Engineering section December 30, 2014; accepted for publication (in revised form) November 2, 2015; published electronically January 6, 2016.

http://www.siam.org/journals/sisc/38-1/100184.html

†Department of Mathematics, Tufts University, Medford, MA 02155 (james.adler@tufts.edu, david.emerson@tufts.edu).

‡Department of Mathematics and Statistics, Memorial University of Newfoundland, St. John's, Newfoundland and Labrador A1C 5S7, Canada (smaclachlan@mun.ca). This author work was partially supported by an NSERC Discovery grant.

§Department of Applied Mathematics, University of Colorado–Boulder, Boulder, CO 80309 (tmanteuf@colorado.edu). This author work was sponsored by the Department of Energy under grants DE-FC02-03ER25574 and DE-NA0002376 and Lawrence Livermore National Laboratory under contract number B568677.

tions include nanoparticle organization, photorefractive cells [36], and liquid crystal elastomers designed to produce effective actuator devices such as light-driven motors [60] and artificial muscles [57]. Numerical simulations of liquid crystal equilibrium configurations are used to optimize device designs, analyze experiments, and suggest the presence of new physical phenomena [1, 6]. Many current technologies and experiments, including bistable devices [17, 45], require simulations with anisotropic physical constants on two-dimensional (2D) and 3D domains.

To this end, a theoretically supported energy-minimization finite-element approach using Newton linearization and a Lagrange multiplier for the pointwise constraint was developed in [1, 2]. The approach effectively enforces the unit-length constraint while converging to energy-minimizing configurations. However, alternative approaches to efficiently impose unit-length conformance exist. Penalty methods are widely applied to liquid crystal equilibrium problems [6, 30, 34] and are utilized extensively to simplify the Leslie–Ericksen equations [24, 40] in nematohydrodynamics simulations [42, 43, 44]. In addition, penalty methods are used for unit-length constraints in certain ferromagnetic problems [35].

In this paper, we focus on the Frank–Oseen elastic free-energy model, comparing the performance of a penalty approach for the necessary unit-length constraint to that of the Lagrange-multiplier implementation developed in [1, 2]. The analysis of a penalty formulation in the context of the energy-minimization algorithm and subsequent comparison to the Lagrange multiplier scheme undertaken here are important in understanding the potential trade-offs between accuracy and computational cost of the methods. Additionally, taking advantage of the energy-minimization framework, we derive several cheap and nonintrusive tailored trust-region methods. These trust-region approaches include 1D- and 2D-subspace minimization techniques [14, 15, 49]. A modified penalty method that normalizes the director after each step, similar to the approach in [28], is also introduced.

The resulting algorithms are tested on three benchmark elastic problems. Analytical expressions exist for the exact solutions to two of the configurations and one simulates nano-patterned boundary conditions which, as seen in [2], present a challenging configuration with regard to unit-length conformance. In each of the experiments conducted, the Lagrange multiplier method outperforms the penalty approaches in a number of measures. Moreover, the proposed trust-region methods are shown to improve convergence robustness and nested iteration (NI) proves exceptionally effective at reducing computational costs. Finally, a novel Braess–Sarazin-type multigrid scheme [13] is introduced for electrically and flexoelectrically coupled problems and demonstrates scalability for highly complicated models with coupled physics.

The energy model and minimization approaches are elaborated in section 2. In section 3, well-posedness for the intermediate Newton linearizations that arise in the penalty formulation is discussed. The trust-region methods are derived and investigated in section 4. Section 5 outlines the computational implementation and discusses the numerical results. The multigrid method is presented in section 6. Finally, section 7 gives some concluding remarks, and future work is discussed.

2. Energy model and minimization. While a number of elastic free-energy models exist [20, 26, 55], we consider the Frank–Oseen free energy [55]. The equilibrium, elastic free energy is represented by a functional depending on deformations of the nondimensional director field, $\mathbf{n}$. Liquid crystal samples favor stable configurations attaining minimal free energy. As in [1, 2], let $\bar{K}_i$, $i = 1, 2, 3$, be the Frank constants [26] with $\bar{K}_i \geq 0$ by Ericksen's inequalities [23], and define the dimensionless

tensor

$$\mathbf{Z} = \kappa \mathbf{n} \otimes \mathbf{n} + (\mathbf{I} - \mathbf{n} \otimes \mathbf{n}) = \mathbf{I} - (1 - \kappa)\mathbf{n} \otimes \mathbf{n},$$

where $\kappa = \bar{K}_2/\bar{K}_3$. In general, we consider the case that $\bar{K}_2, \bar{K}_3 \neq 0$. For both scalar and vector quantities, denote the classical $L^2(\Omega)$ inner product and norm as $\langle \cdot, \cdot \rangle_0$ and $\| \cdot \|_0$, respectively, and the standard Euclidean inner product and norm as $(\cdot, \cdot)$ and $| \cdot |$. Throughout this paper, we assume the presence of Dirichlet boundary conditions or mixed Dirichlet and periodic boundary conditions on a rectangular domain and, therefore, utilize the null Lagrangian simplification discussed in [1, 55]. Thus, the Frank–Oseen elastic free energy for a domain, $\bar{\Omega}$, is written

$$\int_{\bar{\Omega}} \bar{w}_F \, d\bar{V} = \frac{1}{2}\bar{K}_1 \|\nabla_{\bar{\mathbf{x}}} \cdot \mathbf{n}\|_0^2 + \frac{1}{2}\bar{K}_3 \langle \mathbf{Z}\nabla_{\bar{\mathbf{x}}} \times \mathbf{n}, \nabla_{\bar{\mathbf{x}}} \times \mathbf{n} \rangle_0, \tag{2.1}$$

where $\nabla_{\bar{\mathbf{x}}}$ represents the standard differential operator for $\bar{\Omega}$. Foundational theoretical work establishing the existence of minimizing director fields for this energy model is considered in [31]. Note that if $\kappa = 1$, $\mathbf{Z}$ is reduced to the identity and the energy becomes a Div-Curl system. The Frank constants may be determined experimentally for different liquid crystal types, are often anisotropic (i.e., $\bar{K}_1 \neq \bar{K}_2 \neq \bar{K}_3$), and may depend on temperature [21].

The one-constant approximation such that $\bar{K}_1 = \bar{K}_2 = \bar{K}_3$ is a widely applied simplification of (2.1) discussed in [16, 28, 43, 44, 51, 55]. This simplification significantly reduces the complexity of the elastic free-energy functional for both theoretical analysis and computational simulation. While this approximation is accurate in many scenarios, it neglects anisotropic physical characteristics, which play major roles in liquid crystal phenomena [7, 38]. Therefore, both the penalty and Lagrange multiplier methods considered below do not rely on as strong an assumption, and the computational results address configurations where anisotropy is important.

2.1. Penalty and Lagrange multiplier energy minimization. The admissible equilibrium state for a liquid crystal sample is one that minimizes the system free energy in (2.1), subject to the local constraint $(\mathbf{n}, \mathbf{n}) = 1$. Here, we discuss two energy-minimization approaches that impose this constraint via a penalty method or with Lagrange multipliers. First, let σ be a fixed length scale and K represent a characteristic Frank constant. To nondimensionalize the elastic free energy in (2.1), we apply the spatial change of variables, $\bar{\mathbf{x}} = \sigma\mathbf{x}$, divide the free energy by σK, and define nondimensional Frank constants, $K_i = \frac{\bar{K}_i}{K}$, $i = 1, 2, 3$. Note that the change of variables also scales derivatives. Thus, to compute free-energy minimizing configurations, we define the nondimensionalized functional

$$\mathcal{F}(\mathbf{n}) = K_1 \|\nabla \cdot \mathbf{n}\|_0^2 + K_3 \langle \mathbf{Z}\nabla \times \mathbf{n}, \nabla \times \mathbf{n} \rangle_0 \tag{2.2}$$

on a dimensionless domain Ω with dimensionless differential operator ∇. Throughout this paper, we will make use of the spaces

$$H(\mathrm{div}, \Omega) = \{\mathbf{v} \in L^2(\Omega)^3 : \nabla \cdot \mathbf{v} \in L^2(\Omega)\},$$
$$H(\mathrm{curl}, \Omega) = \{\mathbf{v} \in L^2(\Omega)^3 : \nabla \times \mathbf{v} \in L^2(\Omega)^3\},$$

as well as

$$\mathcal{H}^{DC}(\Omega) = \{\mathbf{v} \in H(\mathrm{div}, \Omega) \cap H(\mathrm{curl}, \Omega) : B(\mathbf{v}) = \mathbf{g}\},$$

with norm $\|\mathbf{v}\|_{DC}^2 = \|\mathbf{v}\|_0^2 + \|\nabla\cdot\mathbf{v}\|_0^2 + \|\nabla\times\mathbf{v}\|_0^2$ and appropriate boundary conditions $B(\mathbf{v}) = \mathbf{g}$. Here, we assume that $\mathbf{g}$ satisfies appropriate compatibility conditions for the operator B. For example, if B represents full Dirichlet boundary conditions and Ω has a Lipschitz continuous boundary, it is assumed that $\mathbf{g} \in H^{\frac{1}{2}}(\partial\Omega)^3$ [29]. Furthermore, let $\mathcal{H}_0^{DC}(\Omega) = \{\mathbf{v} \in H(\mathrm{div},\Omega) \cap H(\mathrm{curl},\Omega) : B(\mathbf{v}) = \mathbf{0}\}$. Note that if Ω is a Lipshitz domain and B imposes full Dirichlet boundary conditions on all components of $\mathbf{v}$, then $\mathcal{H}_0^{DC}(\Omega) = H_0^1(\Omega)^3$ [29, Lemma 2.5].

2.2. Lagrange multiplier formulation. For this approach, the pointwise unit-length constraint is imposed by a continuous Lagrange multiplier. Following [1, 2], the Lagrangian is defined as

$$\mathcal{L}(\mathbf{n},\lambda) = \mathcal{F}(\mathbf{n}) + \int_\Omega \lambda(\mathbf{x})((\mathbf{n},\mathbf{n}) - 1)\, dV, \tag{2.3}$$

where we assume that $\lambda \in L^2(\Omega)$. Note that while this technical assumption is widely suitable in physical applications and used in the theoretical developments of this paper, nematic configurations exist, for instance, those containing defects, where the square integrability assumption on λ is not satisfied. The Lagrangian above has been nondimensionalized in the same fashion as the free-energy functional. In dimensional form, the Lagrange multiplier, $\bar{\lambda}$, is a pressure term. The dimensionless Lagrange multiplier in (2.3) is defined to be $\lambda = \frac{\sigma^2\bar{\lambda}}{K}$. First-order optimality conditions, given by

$$\begin{aligned} \mathcal{L}_{\mathbf{n}}[\mathbf{v}] &= \frac{\partial}{\partial\mathbf{n}}\mathcal{L}(\mathbf{n},\lambda)[\mathbf{v}] = 0 \qquad && \forall \mathbf{v} \in \mathcal{H}_0^{DC}(\Omega), \\ \mathcal{L}_{\lambda}[\gamma] &= \frac{\partial}{\partial\lambda}\mathcal{L}(\mathbf{n},\lambda)[\gamma] = 0 \qquad && \forall \gamma \in L^2(\Omega), \end{aligned}$$

are derived and linearized to yield the Newton update equations

$$\begin{bmatrix} \mathcal{L}_{\mathbf{nn}} & \mathcal{L}_{\mathbf{n}\lambda} \\ \mathcal{L}_{\lambda\mathbf{n}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \delta\mathbf{n} \\ \delta\lambda \end{bmatrix} = - \begin{bmatrix} \mathcal{L}_{\mathbf{n}} \\ \mathcal{L}_{\lambda} \end{bmatrix}, \tag{2.4}$$

where each of the system components is evaluated at the current approximations $\mathbf{n}_k$ and λ_k, while $\delta\mathbf{n} = \mathbf{n}_{k+1} - \mathbf{n}_k$ and $\delta\lambda = \lambda_{k+1} - \lambda_k$ are the desired updates to these approximations. The matrix-vector multiplication indicates the direction that the derivatives in the Hessian are taken. That is,

$$\begin{aligned} \mathcal{L}_{\mathbf{nn}}[\mathbf{v}]\cdot\delta\mathbf{n} &= \frac{\partial}{\partial\mathbf{n}}\left(\mathcal{L}_{\mathbf{n}}(\mathbf{n}_k,\lambda_k)[\mathbf{v}]\right)[\delta\mathbf{n}], \quad \mathcal{L}_{\mathbf{n}\lambda}[\mathbf{v}]\cdot\delta\lambda = \frac{\partial}{\partial\lambda}\left(\mathcal{L}_{\mathbf{n}}(\mathbf{n}_k,\lambda_k)[\mathbf{v}]\right)[\delta\lambda], \\ \mathcal{L}_{\lambda\mathbf{n}}[\gamma]\cdot\delta\mathbf{n} &= \frac{\partial}{\partial\mathbf{n}}\left(\mathcal{L}_{\lambda}(\mathbf{n}_k,\lambda_k)[\gamma]\right)[\delta\mathbf{n}], \end{aligned}$$

where the partials denote Gâteaux derivatives in the respective variables. The above system represents a linearized variational system for which we seek solutions $\delta\mathbf{n}$ and $\delta\lambda$. The complete system is found in [2].

2.3. Penalty method formulation. In order to define the penalty approach, the free-energy functional in (2.2) is augmented with a weighted, positive term,

$$\mathcal{P}(\mathbf{n}) = K_1\langle\nabla\cdot\mathbf{n},\nabla\cdot\mathbf{n}\rangle_0 + K_3\langle\mathbf{Z}\nabla\times\mathbf{n},\nabla\times\mathbf{n}\rangle_0 + \zeta\langle\mathbf{n}\cdot\mathbf{n}-1,\mathbf{n}\cdot\mathbf{n}-1\rangle_0, \tag{2.5}$$

where $\zeta > 0$ represents a constant weight, penalizing deviations of the solution from the unit-length constraint. Thus, in the limit of large ζ values, unconstrained minimization of (2.5) is equivalent to the constrained minimization of (2.2). In dimensional form, similar to the Lagrange multiplier, $\bar{\zeta}$ represents a pressure. Using $\zeta = \frac{\sigma^2\bar{\zeta}}{K}$, the functional in (2.5) has been nondimensionalized in the same way as the Lagrangian. In order to minimize $\mathcal{P}(\mathbf{n})$, we compute the Gâteaux derivative of $\mathcal{P}(\mathbf{n})$ with respect to $\mathbf{n}$ in the direction $\mathbf{v} \in \mathcal{H}_0^{DC}(\Omega)$. Hence, the first-order optimality condition is

$$\mathcal{P}_{\mathbf{n}}(\mathbf{n})[\mathbf{v}] = \frac{\partial}{\partial \mathbf{n}}\mathcal{P}(\mathbf{n})[\mathbf{v}] = 0 \qquad\qquad \forall \mathbf{v} \in \mathcal{H}_0^{DC}(\Omega).$$

Computation of this derivative yields the variational problem

$$\begin{aligned}\mathcal{P}_{\mathbf{n}}(\mathbf{n})[\mathbf{v}] = {} & 2K_1\langle \nabla\cdot\mathbf{n}, \nabla\cdot\mathbf{v}\rangle_0 + 2K_3\langle \mathbf{Z}\nabla\times\mathbf{n}, \nabla\times\mathbf{v}\rangle_0 \\ & + 2(K_2 - K_3)\langle \mathbf{n}\cdot\nabla\times\mathbf{n}, \mathbf{v}\cdot\nabla\times\mathbf{n}\rangle_0 + 4\zeta\langle \mathbf{v}\cdot\mathbf{n}, \mathbf{n}\cdot\mathbf{n} - 1\rangle_0 = 0\end{aligned}$$

for all $\mathbf{v} \in \mathcal{H}_0^{DC}(\Omega)$.

As with the Lagrangian formulation, the variational problem above contains nonlinearities. Therefore, Newton iterations are again applied, requiring computation of the second-order Gâteaux derivative with respect to $\mathbf{n}$. The Newton linearizations are written

$$\frac{\partial}{\partial \mathbf{n}}\left(\mathcal{P}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}]\right)[\delta\mathbf{n}] = -\mathcal{P}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}] \qquad\qquad \forall \mathbf{v} \in \mathcal{H}_0^{DC}(\Omega), \tag{2.6}$$

where

$$\begin{aligned}\frac{\partial}{\partial \mathbf{n}}\left(\mathcal{P}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}]\right)[\delta\mathbf{n}] = {} & 2K_1\langle \nabla\cdot\delta\mathbf{n}, \nabla\cdot\mathbf{v}\rangle_0 + 2K_3\langle \mathbf{Z}(\mathbf{n}_k)\nabla\times\delta\mathbf{n}, \nabla\times\mathbf{v}\rangle_0 \\ & + 2(K_2 - K_3)\Big(\langle \delta\mathbf{n}\cdot\nabla\times\mathbf{v}, \mathbf{n}_k\cdot\nabla\times\mathbf{n}_k\rangle_0 \\ & + \langle \mathbf{n}_k\cdot\nabla\times\mathbf{v}, \delta\mathbf{n}\cdot\nabla\times\mathbf{n}_k\rangle_0 + \langle \mathbf{n}_k\cdot\nabla\times\mathbf{n}_k, \mathbf{v}\cdot\nabla\times\delta\mathbf{n}\rangle_0 \\ & + \langle \mathbf{n}_k\cdot\nabla\times\delta\mathbf{n}, \mathbf{v}\cdot\nabla\times\mathbf{n}_k\rangle_0 + \langle \delta\mathbf{n}\cdot\nabla\times\mathbf{n}_k, \mathbf{v}\cdot\nabla\times\mathbf{n}_k\rangle_0\Big) \\ & + 4\zeta\big(\langle \mathbf{n}_k\cdot\mathbf{n}_k - 1, \mathbf{v}\cdot\delta\mathbf{n}\rangle_0 + 2\langle \delta\mathbf{n}\cdot\mathbf{n}_k, \mathbf{v}\cdot\mathbf{n}_k\rangle_0\big).\end{aligned} \tag{2.7}$$

Completing (2.6) with the above second-order derivative computation yields a linearized variational system to be solved for $\delta\mathbf{n}$.

3. Well-posedness of the penalty Newton linearizations. Existence and uniqueness of solutions to the discrete form of the linearization systems described in (2.4) are established in [2] under reasonable assumptions. These results are readily adapted to the discrete form of the penalty method linearizations in (2.6) [5].

Let $a(\delta\mathbf{n}, \mathbf{v})$ denote the bilinear form defined in (2.7) for fixed $\mathbf{n}_k$ and let $F(\mathbf{v})$ be the linear functional on the right-hand side of the linearization in (2.6). Using finite elements to approximate the desired update, $\delta\mathbf{n}$, and considering a discrete space $V_h \subset \mathcal{H}_0^{DC}(\Omega)$ yields the discrete linearized system to find $\delta\mathbf{n}_h \in V_h$ such that

$$a(\delta\mathbf{n}_h, \mathbf{v}_h) = F(\mathbf{v}_h) \qquad \forall \mathbf{v}_h \in V_h. \tag{3.1}$$

Throughout the rest of this section, the developed theory applies exclusively to discrete spaces. Therefore, except when necessary for clarity, we drop the subscript h along with the notation, $\delta\mathbf{n}$. For instance, we write $a(\mathbf{u}, \mathbf{v})$ to indicate the bilinear

form in (3.1) operating on the discrete space $V_h \times V_h$. Furthermore, we refer to the following set of assumptions.

ASSUMPTION 3.1. *Consider an open bounded domain, Ω, with a Lipschitz-continuous boundary. Further, assume that there exist constants $0 < \alpha \leq 1 \leq \beta$, such that $\alpha \leq |\mathbf{n}_k|^2 \leq \beta$ and $\mathbf{Z}(\mathbf{n}_k(\mathbf{x}))$ remains uniformly symmetric positive definite with lower and upper bounds on its Rayleigh quotient, η and Λ, respectively, as in* [2, Lemma 2.1]. *Finally, assume that Dirichlet boundary conditions are applied.*

While the assumption above and the theory below explicitly concern full Dirichlet boundary conditions, the theory is equally valid for mixed Dirichlet and periodic boundary conditions on a rectangular domain.

In order to establish well-posedness of (3.1), we show that the functional, $F(\mathbf{v})$, is continuous and that the bilinear form, $a(\mathbf{u}, \mathbf{v})$, is continuous and coercive. Here, the lemmas are stated without proof, as they are similar to those in [2]. For full details, see [5].

LEMMA 3.2. *Under Assumption* 3.1, *F is a bounded linear functional on V_h.*

LEMMA 3.3. *Under Assumption* 3.1, *$a(\mathbf{u}, \mathbf{v})$ is continuous.*

Following the theory established in [2], two coercivity lemmas for $a(\mathbf{u}, \mathbf{v})$ are given. The first addresses the case when $\kappa = 1$. The second considers coercivity when κ lies in a neighborhood of unity, $\kappa \in (1 - \epsilon_2, 1 + \epsilon_1)$. Let $\alpha_0 > 0$ be the coercivity constant from [2, Lemma 3.7].

LEMMA 3.4. *Under Assumption* 3.1, *if $\kappa = 1$ and $2\zeta(1 - \alpha) < \alpha_0$, there exists a $\beta_0 > 0$ such that $\beta_0 \|\mathbf{v}\|_{DC}^2 \leq a(\mathbf{v}, \mathbf{v})$ for all $\mathbf{v} \in V_h$.*

Therefore, $a(\mathbf{u}, \mathbf{v})$ is coercive for $\kappa = 1$ if ζ is not so large in comparison to the pointwise lower bound on the director length as to overwhelm α_0.

The assumption that $\kappa = 1$ can be loosened to include additional anisotropy and retain coercivity of $a(\mathbf{u}, \mathbf{v})$. Let $C > 0$ such that $\|\mathbf{v}\|_0^2 \leq C\big(\|\nabla \cdot \mathbf{v}\|_0^2 + \|\nabla \times \mathbf{v}\|_0^2\big)$. The following extends the results of [2, Lemma 3.8] to the penalty method.

LEMMA 3.5 (small data). *Under Assumption* 3.1, *if*

$$\beta_1 = \frac{\min(K_1, K_3)}{C + 1} - 2\zeta(1 - \alpha) > 0,$$

there exists $\epsilon_1, \epsilon_2 > 0$, dependent on $\beta = \max |\mathbf{n}_k|^2$, such that for $\kappa \in (1 - \epsilon_2, 1 + \epsilon_1)$, $a(\mathbf{u}, \mathbf{v})$ is coercive.

The above lemmas allow for the formulation of the following summary theorem.

THEOREM 3.6. *Under Assumption* 3.1, *if the conditions of Lemmas* 3.4 *or* 3.5 *are satisfied, the discrete variational problem in* (3.1) *is well-posed.*

Proof. Lemmas 3.2 and 3.3 imply that $F(\mathbf{v})$ and $a(\mathbf{u}, \mathbf{v})$ are continuous, respectively. Lemmas 3.4 or 3.5 imply that $a(\mathbf{u}, \mathbf{v})$ is coercive. Therefore, by the Lax–Milgram theorem [12], (3.1) is a well-posed discrete variational problem. ∎

Therefore, the discretization of the linearizations arising in the penalty method are always well-posed under the assumption of small anisotropy in the system coefficients and sufficient conformance to the unit-length constraint. Note that the closer α is to unity, the larger ζ may be while still maintaining coercivity in the theoretical analysis.

In the penalty approach, ζ must also be chosen appropriately to achieve accurate representation of the unit-length constraint. As discussed above, in dimensional form, the penalty parameter represents a constant pressure term energetically penalizing deviations from the unit-length constraint. If ζ is too small, constraint conformance becomes poor and the functional minimum does not accurately represent the constrained minimum. Alternatively, if ζ is too large, the solvability of the intermediate

variational systems degrades. In the theoretical analysis, the neighborhood admitting coercivity around $\kappa = 1$ shrinks. Furthermore, as ζ grows, the associated linear systems become increasingly ill-conditioned and the interaction between the penalty and elastic free-energy terms becomes progressively more imbalanced. Related effects contributing to ill-conditioning are often observed in Ginzburg–Landau-type penalty approaches and certain applications of Landau–de Gennes models [8, 9, 48]. On the other hand, the proof of well-posedness does not require establishing an inf-sup condition that necessitates subtle choices of finite-element spaces, as used for the Lagrange multiplier approach in [2, 33].

In the numerical implementation to follow, no special effort was made to ensure solvability, and few ill effects are seen at large values of ζ, especially in the presence of trust regions. Nonetheless, if the need arose, a bootstrap algorithm for the penalty weight could be implemented. The size of ζ can also affect convergence of the nonlinear iterations. In previous work concerning static liquid crystal configurations in the context of a least-squares finite-element approach, the unit-length constraint was essentially enforced via a penalty method [6]. In this framework, large weights were needed to ensure constraint conformance, which subsequently influenced nonlinear convergence. Such behavior is also expected in the context of energy minimization when large penalty weights are necessary.

4. Robust Newton step methods. The Newton method applied to the Lagrange multiplier formulation discussed in [1, 2] employs naïve damped Newton stepping. That is, for a computed Newton update direction, $\delta\mathbf{n}$, a constant damping factor, $0 < \omega \leq 1$, is applied such that the new iterate is given as $\mathbf{n}_{k+1} = \mathbf{n}_k + \omega\delta\mathbf{n}$. Such an approach aims to improve convergence robustness when dealing with an inaccurate initial guess on coarse grids. However, this procedure may miss opportunities to take larger steps in "good" descent directions that effectively reduce the free energy.

Trust-region techniques are specifically designed to improve the robustness and efficiency of iterative procedures such as Newton's method. Updates are confined to a neighborhood, known as a trust region, where the accuracy of the linearized first-order optimality conditions is "trusted." These neighborhoods are expanded or contracted based on a measure of the model fidelity for a computed update. Significant research has produced both theoretical support and practical applications of such techniques [49]. This section discusses adaptations of constrained and unconstrained trust-region methods for the Lagrangian and penalty approaches discussed in section 2. These methods are unintrusive and seamlessly integrate into the energy-minimization framework above. For a general overview of trust-region methods, see [49].

4.1. Trust-region approaches for the penalty formulation. Using the penalty functional in (2.5), the desired energy minimization is unconstrained. For this subsection, we denote the discretized forms of $\frac{\partial}{\partial \mathbf{n}}\left(\mathcal{P}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}]\right)[\delta\mathbf{n}]$ and $\mathcal{P}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}]$ as U_k and $\mathbf{f}_k$, respectively. The quadratic model of the penalty functional, for a given $\mathbf{n}_k$, is written

$$M_k(\delta\mathbf{n}) = \mathcal{P}(\mathbf{n}_k) + \mathbf{f}_k^T\delta\mathbf{n} + \frac{1}{2}\delta\mathbf{n}^T U_k \delta\mathbf{n}. \tag{4.1}$$

As a consequence of the well-posedness theory in section 3, the matrix U_k is positive definite for each iteration. Therefore, we follow the methodology in [15, 53], computing steps by solving a trust-region minimization problem.

We seek an efficiently computable correction, $\delta\mathbf{n}$, that approximately minimizes the model in (4.1). In the following, we introduce two approaches to computing step length and direction for this problem. The performance of these techniques is vetted in the numerical experiments below.

Damped Newton stepping is equivalent to taking a small step in the descent direction, $-U_k^{-1}\mathbf{f}_k$. This is an effective means of finding energy-minimizing solutions for both the penalty and Lagrangian methods. Therefore, in the first approach, a simple step selection technique is used in which the step is chosen satisfying the constrained minimization problem

$$\delta\mathbf{n}(\Delta_k) = \operatorname{argmin}\left\{\mathcal{P}(\mathbf{n}_k) + \mathbf{f}_k^T\delta\mathbf{n} + \frac{1}{2}\delta\mathbf{n}^T U_k\delta\mathbf{n} : \right. \\ \left. |\delta\mathbf{n}| \le \Delta_k,\ \delta\mathbf{n} = \mu U_k^{-1}\mathbf{f}_k\right\}, \tag{4.2}$$

where Δ_k indicates the trust-region radius for iterate $\mathbf{n}_k$. Candidate solutions of (4.2) are easily computed to be $-U_k^{-1}\mathbf{f}_k$, the full Newton step, which may or may not be inside the trust region, and $\pm\frac{\Delta_k}{|U_k^{-1}\mathbf{f}_k|}U_k^{-1}\mathbf{f}_k$, representing steps to the trust region boundary.

An important aspect of trust-region methods is the adjustment of the trust-region radius and application of a computed step. This typically involves a measure of a computed step's merit. For a computed step, $\delta\mathbf{n}$, we calculate the ratio,

$$\rho_k = \frac{\mathcal{P}(\mathbf{n}_k) - \mathcal{P}(\mathbf{n}_k + \delta\mathbf{n})}{M_k(\mathbf{0}) - M_k(\delta\mathbf{n})},$$

of the actual to the predicted reduction in $\mathcal{P}$ due to the computed step. The closer ρ_k is to 1, the more accurately the quadratic model behavior matches that of the true functional.

If the ratio, ρ_k, is deemed acceptable, the step is applied and the trust region expands, remains static, or shrinks depending on the specific value of ρ_k. If ρ_k is too small, the step is rejected, the trust-region radius is shrunk, and the process is repeated. To quantify, let $0 < \eta_3 < \eta_1 < \eta_2$ be positive constants, along with $0 < C_1 < 1 < C_3$. Further, let $\bar{\Delta}$ be a maximum limit on the trust-region size. Using these parameters, the specific decision trees for accepting a step and subsequently adjusting the trust region are given in Procedures 1 and 2, respectively.

Procedure 1. Solution update.

if $\rho_k > \eta_3$ **then**
  Accept step: $\mathbf{n}_{k+1} = \mathbf{n}_k + \delta\mathbf{n}$.
else
  Reject step: $\mathbf{n}_{k+1} = \mathbf{n}_k$.
end

Procedure 2. TR size adjustment.

if $\rho_k < \eta_1$ **then**
  Shrink region: $\Delta_{k+1} = C_1\Delta_k$.
else if $\rho_k > \eta_2$ *and* $|\delta\mathbf{n}| = \Delta_k$ **then**
  Expand region: $\Delta_{k+1} = \min(C_3\Delta_k, \bar{\Delta})$.
else
  Keep region constant: $\Delta_{k+1} = \Delta_k$.
end

TABLE 1
Trust-region parameters for the penalty and Lagrangian formulations.

Penalty	$\eta_1 = 0.25$	$\eta_2 = 0.75$	$\eta_3 = 0.125$	$C_1 = 0.5$
	$C_3 = 1.3$	$\Delta_{\text{inc}} = 0.3$	$\bar{\Delta} = 100$	$\Delta_{\text{init}} = 0.3$

Lagrangian	$\eta_1 = 0.5$	$\eta_2 = 0.25$	$w_{\text{inc}} = 0.1$	$w_{\text{dec}} = 0.1$
	$w_{\text{lev}} = 0.1$	$w_{\text{min}} = 0.1$	$w_{\text{init}} = 0.2$	–

For our algorithm, if the components of the ratio, ρ_k, are very small and the computed step lies on the interior of the trust region, representing a full step toward satisfying the first-order optimality conditions, we choose to apply the step regardless of ρ_k and the trust region remains static. In this way, the trust-region minimization approach is used until we trust in the application of full Newton steps to obtain the first-order optimality conditions. A set of typical values for the trust-region parameters above are listed in Table 1 and used in the numerical methods below.

A number of well-founded techniques improving trust-region step selection exist, including dogleg and 2D subspace methods [15, 25, 49]. Because the 2D subspace method subsumes both the simple step selection approach above and dogleg methods, it is chosen as the alternative step selection computation here. Steps are computed by solving

$$\delta\mathbf{n}(\Delta_k) = \operatorname{argmin}\left\{\mathcal{P}(\mathbf{n}_k) + \mathbf{f}_k^T\delta\mathbf{n} + \frac{1}{2}\delta\mathbf{n}^T U_k\delta\mathbf{n} : \right.$$
$$\left. |\delta\mathbf{n}| \le \Delta_k,\ \delta\mathbf{n} = \mu_1\mathbf{f}_k + \mu_2 U_k^{-1}\mathbf{f}_k\right\}. \tag{4.3}$$

Again, the candidate solutions for (4.3) are efficiently computable, amounting to solving for the zeroes of a fourth-order polynomial.

4.1.1. A renormalization penalty method. In addition to the standard penalty method discussed above, a modification is also considered in the numerical experiments below. Once the approximation to the solution has been updated with a computed and accepted step, the new approximation is renormalized at the finite-element nodes. That is, the updated approximation is projected onto the unit sphere at each finite-element node. In this way, the lower bound in Assumption 3.1, α, remains closer to unity, allowing for larger penalty weights. This procedure is similar to that presented in [28] for a discrete Lagrange multiplier formulation. There, the approach is derived within a nullspace method framework. Here, renormalization is applied to the penalty method, with and without trust regions and NI.

This renormalization aims at improving unit-length conformance for solutions computed by the penalty method. The expectation is that this will lead to enhanced constraint conformance at lower penalty weights. However, unless the renormalization scaling is relatively uniform across nodes, the Newton direction may be significantly altered. Throughout this paper, this modification will be referred to as the "renormalization" penalty method.

4.2. A trust-region method for the Lagrange multiplier formulation. Applications of trust-region techniques to optimization problems with nonlinear constraints have also been developed. However, certain challenges arise in the theory and practical use of such methods [46]. Here, we consider existing trust-region approaches

in the context of finite-element methods. For this subsection, let W_k be the matrix associated with a finite-element discretization of the second-order derivative of (2.2) (i.e., the functional without the Lagrange multiplier term), given by $\frac{\partial}{\partial \mathbf{n}} \left(\mathcal{F}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}]\right)[\delta \mathbf{n}]$. For the trust-region approach, write the constraint

$$c(\mathbf{n}) = \langle \mathbf{n} \cdot \mathbf{n} - 1, \mathbf{n} \cdot \mathbf{n} - 1 \rangle_0 = 0. \tag{4.4}$$

The Gâteaux derivative of (4.4) is

$$\frac{\partial}{\partial \mathbf{n}} c(\mathbf{n})[\mathbf{v}] = 4\langle \mathbf{n} \cdot \mathbf{n} - 1, \mathbf{n} \cdot \mathbf{v} \rangle_0. \tag{4.5}$$

Finally, let $\mathbf{c}_k$ be the column vector representing the finite-element discretized form of (4.5) at iterate $\mathbf{n}_k$.

One of the significant advantages of finite-element discretizations is the inherent sparsity of the resulting matrices. Trust-region algorithms in the Byrd–Omojokun family [14, 50, 56] require computation of the generally nonsparse matrix N_k, whose columns form an orthonormal basis for the orthogonal complement of $\mathbf{c}_k$, as well as the formation and inversion of the matrix $N_k^T W_k N_k$. In general, the matrix $N_k^T W_k N_k$ is not sparse and quite large, as W_k has dimension $m \times m$ and N_k is $m \times (m-1)$, where m is the number of discretization degrees of freedom for $\mathbf{n}$. Storage and computation with these dense matrices proves to be prohibitive, even on relatively small grids. Therefore, any advantages garnered by the use of these trust regions is outweighed by loss of the finite-element sparsity. Similarly, trust-region methods based on the fundamental work in [58] suffer from sparsity fill-in issues for large matrices in the context of finite-element methods.

To preserve sparsity properties, while still maintaining some advantages of a trust-region approach, we implement a simple trust-region method specifically fitted to the Lagrange multiplier formulation of the minimization problem. For the Lagrange multiplier approach in section 2.2, we compute a Newton update direction, $\delta\chi = [\delta \mathbf{n}\ \delta\lambda]^T$. This update is meant to bring $\mathbf{n}_k$ and λ_k closer to satisfying the first-order optimality conditions. Let $\mathcal{L}_0(\mathbf{n}_k, \lambda_k)$ represent the finite-element discretized form of the right-hand side of (2.4) for $\mathbf{n}_k$ and λ_k. Define the proportions w_k and $w_{\lim}$ such that $0 < w_{\lim} \leq w_k \leq 1$, where $w_{\lim}$ is a lower bound for w_k. For a given step, $w_k \delta\chi$, the expected change in $|\mathcal{L}_0(\mathbf{n}_k, \lambda_k)|$ is equal to $w_k |\mathcal{L}_0(\mathbf{n}_k, \lambda_k)|$. Therefore, we define the ratio

$$\rho_k = \frac{|\mathcal{L}_0(\mathbf{n}_k, \lambda_k)| - |\mathcal{L}_0(\mathbf{n}_k + w_k \delta\mathbf{n}, \lambda_k + w_k \delta\lambda)|}{w_k |\mathcal{L}_0(\mathbf{n}_k, \lambda_k)|}.$$

This ratio compares the change in $\mathcal{L}_0(\mathbf{n}, \lambda)$ predicted by the linearized model to the true change in $\mathcal{L}_0(\mathbf{n}, \lambda)$ for a computed step.

Let $0 < \eta_2 < \eta_1$ and $w_{\text{inc}}, w_{\text{dec}} \in (0, 1]$. Since w_k is a scaling factor, rather than a radius length, step selection and trust-region adjustment differ slightly from the procedures discussed above and are given in Procedures 3 and 4, respectively.

Procedure 3. SOLUTION UPDATE.

if $\rho_k > \eta_2$ *or* $w_k = w_{lim}$ **then**
  Accept step:
  $[\mathbf{n}_{k+1}\ \lambda_{k+1}]^T = [\mathbf{n}_k\ \lambda_k]^T + w_k \delta\chi$.
else
  Reject step:
  $[\mathbf{n}_{k+1}\ \lambda_{k+1}]^T = [\mathbf{n}_k\ \lambda_k]^T$.
end

Procedure 4. TR SIZE ADJUSTMENT.

if $\rho_k < \eta_2$ **then**
  Shrink region:
  $w_{k+1} = \max(w_{\text{lim}}, w_k - w_{\text{dec}})$.
else if $\eta_2 < \rho_k < \eta_1$ **then**
  Keep region constant:
  $w_{k+1} = w_k$.
else
  Expand region:
  $w_{k+1} = \min(w_k + w_{\text{inc}}, 1)$.
end

5. Numerical results. In this section, we compare the performance of the methods outlined above for three benchmark equilibrium problems. The general algorithm utilized by each method has three stages; see Algorithm 5. The outer stage implements NI [54], where, at each level, the approximation to the solution is iteratively updated. These updates are computed via one of the methods described above. In general, the iteration stopping criterion, on a given level, is based on a set tolerance for the approximation's conformance to the first-order optimality conditions in the standard Euclidean l_2-norm. For the renormalization penalty method, the Newton iteration tolerance is based on the reduction of the ratio of the energy from the previous step to the current step rather than conformance to the first-order optimality conditions. In the numerical experiments carried out below, both tolerances were held at 10^{-4}. The approximate solution is then transferred to a finer grid. In the current implementation, these finer grids represent uniform refinements of the initial coarse grid. However, adaptive refinement could also be performed.

The components of the variational problems in (2.4) and (2.6) are discretized with finite elements on each grid. Both formulations use $Q_2 \times Q_2 \times Q_2$ elements for $\mathbf{n}$, while the Lagrange multiplier approach uses P_0 elements for λ, as in [1, 2]. In this section, the arising matrices are inverted using the UMFPACK LU decomposition [18, 19]. In section 6, we introduce an optimally scaling multigrid method with improved time to solution. The algorithm's discretizations and grid management are performed with the widely used deal.II finite-element and scientific computing library [10].

For the simulations considered throughout this section, the length scale is taken to be one micron such that $\sigma = 10^{-6}$ m. Furthermore, here and in subsequent sections, the characteristic Frank constant is taken to be $K = 6.2 \times 10^{-12}$ N, the value of $\bar{K}_1$ for 5CB, a common liquid crystal, for convenience in adjusting relative parameter sizes. This rescaling, for instance, yields parameters $K_1 = 1$, $K_2 = 0.62903$, and $K_3 = 1.32258$ for 5CB. Finally, note that throughout this section the penalty weight, ζ, is dimensionless. Each of the problems below is posed on a unit-square domain in

Algorithm 5. General minimization algorithm with NI.

```
0. Initialize solution approximation on coarse grid.
while Refinement limit not reached do
    while Nonlinear iteration tolerance not satisfied do
        1. Assemble discrete components of System (2.4) or (2.6) on current
        grid, H.
        2. Compute correction to current approximation.
        3. Update current approximation.
    end
    4. Uniformly refine the grid to size h.
    5. Interpolate solution u_H → u_h.
end
```

the xy-plane, such that $\Omega = \{(x, y) \mid 0 \leq x, y \leq 1\}$. It is assumed that this domain represents a uniform slab. That is, the vector $\mathbf{n}$ may have nonzero z-component but $\frac{\partial \mathbf{n}}{\partial z} = \mathbf{0}$. Thus, the nondimensional free energies reported in this section are free energies per dimensionless unit-length in the z-direction. Dirichlet boundary conditions are applied at the y-edges and periodic boundary conditions are assumed at the boundaries $x = 0$ and $x = 1$. The experiments to follow consider an 8×8 coarse mesh ascending in six uniform refinements to a 512×512 mesh.

For the numerical experiments, each of the trust-region methods discussed above is applied. For the penalty trust-region methods, the initial trust-region radius is set to Δ_{init}. At each refinement level, the trust-region radius is reset to Δ_{init} plus an incremental increase, Δ_{inc}, with a maximum of $\bar{\Delta}$. The Lagrangian trust-region approach sets the initial value of w_k to w_{init}. After each refinement, w_k is reset to w_{init} plus w_{lev}, up to a maximum of 1. These increments are due to the increasing accuracy of the iterates at each grid level. The constants are outlined in Table 1.

The nontrust-region, damped Newton stepping approach is also performed for both formulations as a comparison benchmark with an initial $\omega = 0.2$, increasing by 0.2 at each refinement to a maximum of 1. While this approach could readily be improved, for example by including inexact line searching via the Wolfe conditions [49], here we compare the trust-region methods outlined above with this naïve stepping approach, as was used in [1, 2]. The performance of each of these methods is discussed below. In the results to follow, all reported free energies are computed using only the elastic free-energy quantities without any augmentations, such as the penalty terms.

5.1. Twist equilibrium configuration. The first set of experiments consider a generalized twist equilibrium configuration. A related twist configuration was used in [2], without error analysis, for the Lagrange multiplier formulation. Here, the problem is used as a baseline for comparison and each method's error convergence is considered. For this experiment, and the tilt-twist experiment in the next subsection, let the general form of the solution be

$$\mathbf{n} = \big(\cos(\theta(y))\cos(\phi(y)), \cos(\theta(y))\sin(\phi(y)), \sin(\theta(y))\big). \tag{5.1}$$

Note that the known analytical solutions have a 1D structure, but the numerical experiments below are full 2D simulations. For the twist configuration, let $\theta_0 = \frac{\pi}{8}$. At the boundaries $\theta(0) = -\theta_0$, $\theta(1) = \theta_0$, and $\phi(0) = \phi(1) = 0$. The Frank constants for this problem are $K_1 = 1.0$, $K_2 = 1.2$, and $K_3 = 1.0$. The analytical equilibrium

TABLE 2

Statistics for the twist equilibrium solution with the different formulations and penalty weights. Included is the system free energy, the computed L^2-error on the finest grid, and the minimum and maximum deviations from unit director length at the quadrature nodes. Approximations of the cost in WUs for the corresponding method with no trust regions and simple trust regions are included. Dashes indicate divergence.

Type	Free energy	L^2-error	Min. Dev.	Max Dev.	Cost	TR cost
Lagrangian	0.370110	2.076e-11	−1.43e-14	7.00e-15	1.350	1.340
Pen. $\zeta = 10^1$	0.358832	1.589e-02	−3.96e-02	−3.59e-05	1.371	1.354
Pen. $\zeta = 10^2$	0.368481	1.993e-03	−4.32e-03	−1.16e-05	1.376	1.355
Pen. $\zeta = 10^3$	0.369931	2.107e-04	−4.32e-04	−3.68e-06	1.440	1.418
Pen. $\zeta = 10^4$	0.370092	2.143e-05	−4.32e-05	−1.14e-06	1.448	1.420
Pen. $\zeta = 10^5$	0.370108	2.154e-06	−4.32e-06	−3.32e-07	1.447	1.426
Pen. $\zeta = 10^6$	0.370110	2.157e-07	−4.32e-07	−7.27e-08	–	1.436
Pen. $\zeta = 10^7$	0.370110	2.158e-08	−5.05e-08	−9.98e-09	–	1.465
Pen. $\zeta = 10^8$	0.370110	2.158e-09	−5.18e-09	−1.06e-09	–	1.516
Pen. $\zeta = 10^9$	0.370110	2.168e-10	−5.19e-10	−1.06e-10	–	1.639

solution for these boundary conditions and Frank constants is derived, under a rotated coordinate system, in [55]. The solution is given by

$$\mathbf{n} = (\cos(\theta_0(2y-1)), 0, \sin(\theta_0(2y-1)))$$

with true free energy $2K_2\theta_0^2$. This corresponds to an expected free energy of 0.37011. The existence of an analytical expression for the solution to this problem allows for the computation of an L^2-error for each computed approximation. This error is computed as $\|\mathbf{u} - \mathbf{u}_h\|_0$ using seventh-order Gaussian quadrature on each element, where $\mathbf{u}$ and $\mathbf{u}_h$ are the true solution and discrete approximation, respectively.

Table 2 compares the performance of the Lagrange multiplier method to the penalty method without renormalization. The runs were performed with NI and the approximate work, measured in terms of assembling and solving a single linearization step on a 512×512 grid and referred to as a work unit (WU), is given for the corresponding method with no trust regions and the simple trust-region approaches, respectively. The work approximation is computed by summing the number of nonzeroes in each matrix across all grids and dividing by the number of nonzeroes in the (fixed) sparsity pattern at the finest level. This estimates the total computational cost, assuming the presence of multigrid solvers that scale optimally with grid size as in section 6, in terms of assembling and solving a single linearization step on the finest grid. The nontrust-region, damped Newton stepping discussed above diverged for penalty parameters of $\zeta = 10^6$ and greater. However, smaller damping parameters may yield convergence. Both penalty-method trust-region approaches converged without modification.

The table demonstrates the superior performance of the Lagrange multiplier method for this problem across all statistics with lower error, cost, and tighter conformance to the constraint. The penalty method does not match the free energy of the Lagrangian formulation until reaching a penalty weight of 10^6 and, without trust regions, encounters divergence issues for these large penalty weights. While trust regions do not significantly reduce overall computations costs, Table 2 suggests that they significantly improve robustness. Finally, it should be noted that the average condition number of the linearization matrices for the penalty formulation scales with

TABLE 3

A comparison of renormalization penalty methods, with and without trust-region approaches, for the twist solution. For each algorithm, the computed L^2-error on the finest grid and an approximation of the cost in WUs is included.

	No trust region		Simple trust region		2D trust region	
Type	L^2-error	Cost	L^2-error	Cost	L^2-error	Cost
Pen. $\zeta = 10^1$	1.457e-02	1.338	1.457e-02	1.334	1.457e-02	1.334
Pen. $\zeta = 10^2$	8.932e-05	1.338	8.931e-05	1.334	8.931e-05	1.334
Pen. $\zeta = 10^3$	3.358e-06	1.339	3.357e-06	1.334	3.357e-06	1.335
Pen. $\zeta = 10^4$	1.523e-07	1.340	1.116e-07	1.336	1.116e-07	1.336
Pen. $\zeta = 10^5$	6.260e-08	8.113	3.595e-09	1.364	3.592e-09	1.340
Pen. $\zeta = 10^6$	6.356e-06	81.120	1.688e-02	73.052	1.098e-07	2.731

TABLE 4

Statistics for the twist equilibrium solution with different penalty weights. Here, the penalty method with renormalization and 2D-subspace minimization is considered. Included is the system free energy, the computed L^2-error on the finest grid, the minimum and maximum deviations from unit director length at the quadrature nodes, and an approximation of the cost in WUs for the corresponding method.

Type	Free energy	L^2-error	Min. Dev.	Max Dev.	2D TR cost
Pen. $\zeta = 10^1$	0.370168	1.457e-02	−4.58e-11	4.58e-11	1.334
Pen. $\zeta = 10^2$	0.370111	8.931e-05	−1.68e-11	1.68e-11	1.334
Pen. $\zeta = 10^3$	0.370110	3.357e-06	−5.18e-12	5.16e-12	1.335
Pen. $\zeta = 10^4$	0.370110	1.116e-07	−1.45e-12	1.43e-12	1.336
Pen. $\zeta = 10^5$	0.370110	3.592e-09	−3.16e-13	2.98e-13	1.340
Pen. $\zeta = 10^6$	0.370110	1.098e-07	−4.04e-14	2.20e-14	2.731

the magnitude of ζ. While poorer conditioning increases the difficulty of the linear solves, the accuracy of the overall method is still improving with larger penalty weights up to $\zeta \approx 10^9$.

The results in Tables 3 and 4 show the performance of the renormalization penalty method with and without trust regions. Table 4 provides additional statistics for the 2D-subspace minimization trust-region approach shown in Table 3. For the twist equilibrium solution, the renormalization penalty method obtains better error values for smaller penalty weights than the unmodified penalty method. In Table 4, using the 2D-subspace minimization trust-region approach, we obtain an error of 3.592e-09 with a penalty weight of only $\zeta = 10^5$. Moreover, the minimum and maximum deviations of the director at the quadrature nodes is closer to that of the Lagrangian method. However, the performance improvements rely more heavily on the penalty parameter. While an error measure closer to the Lagrange multiplier formulation is achieved for $\zeta = 10^5$, performance degrades at $\zeta = 10^6$, with notable jumps in costs for all methods recorded in Table 3. The increases in error are due to the algorithm beginning to emphasize the unit-length constraint over proper director orientation. Correctly selecting the penalty weight represents a fundamental difficulty for this method.

Figure 1(a) displays the number of iterations required to reach the specified iteration tolerance within an NI scheme alongside the final solution computed by the Lagrange multiplier formulation in Figure 1(b). Counts for both the Lagrange multiplier approach and penalty formulation, with and without renormalization, for a penalty parameter $\zeta = 10^3$ are shown. In general, the trust-region methods significantly reduce iteration counts on the coarse grids. However, on the finer grids, this

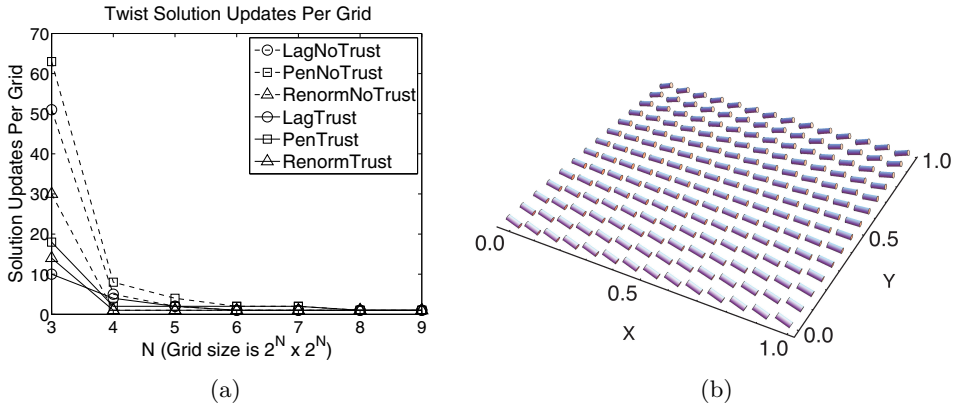


FIG. 1. (a) *Number of iterations required to reach iteration tolerance for each method with NI. The penalty weight for the penalty formulation was* $\zeta = 1000$. *Only the* $2D$*-subspace minimization trust-region approach is displayed, as the behavior of simple trust regions is similar.* (b) *The final computed solution for the Lagrangian formulation on a* 512×512 *mesh (restricted for visualization).*

reduction is not sustained due to the efficiency of NI. Because the improved iteration counts are confined to the coarsest grids, overall cost reduction is generally small. For example, the approximate cost for the Lagrange multiplier method was reduced very slightly from 1.350 WUs to 1.340 WUs, only resulting in a 1 second drop in overall time to solution.

Table 5 both summarizes the efficiency of NI and highlights the strengths of certain applications of trust-region methods. For all the constraint enforcement formulations, NI offers very clear cost improvements. Coupling NI with the Lagrange multiplier method for this problem is quite powerful, yielding the fastest run time and highest accuracy. Trust regions have a clear impact on time to solution in the absence of NI but offer modest time to solution improvements when coupled with NI.

TABLE 5

Twist statistics comparison for NI and trust region combinations. The solve cost column displays an approximation of the work in WUs for the corresponding method. The overall time to solution is also presented. Dashes indicate divergence.

Lagrangian			
Method		Solve cost	Run time
No NI	No TR	61	17,975s
NI	No TR	1.350	550s
No NI	TR	10	3,071s
NI	TR	1.340	548s
Renormalization penalty: $\zeta = 10^5$			
Method		Solve cost	Run time
No NI	No TR	38	11,838s
NI	No TR	8.113	2,272s
No NI	TR	29	9,172s
No NI	TR 2D	32	10,147s
NI	TR	1.364	585s
NI	TR 2D	1.340	584s

Unmodified penalty: $\zeta = 10^5$			
Method		Solve cost	Run time
No NI	No TR	142	41,013s
NI	No TR	1.474	593s
No NI	TR	63	18,425s
No NI	TR 2D	64	19,287s
NI	TR	1.426	569s
NI	TR 2D	1.424	574s
Unmodified penalty: $\zeta = 10^9$			
Method		Solve cost	Run time
No NI	No TR	–	–
NI	No TR	–	–
No NI	TR	1,016	294,349s
No NI	TR 2D	1,736	511,874s
NI	TR	1.639	641s
NI	TR 2D	1.958	764s

If the penalty method is used, pairing NI with trust regions increases robustness and cost consistency. For example, the use of trust regions overcomes prominent divergence issues for the unmodified penalty method with $\zeta = 10^9$ and maintains low costs for the renormalization penalty method at $\zeta = 10^5$. In addition to the improved error performance for $\zeta = 10^5$, the renormalization penalty method is generally faster than the unmodified penalty approach with the same penalty weight. The slightly slower overall run times when NI is paired with trust regions, in comparison with the unmodified penalty method, are due to the work involved in normalizing the director after each iteration. As discussed above, a shortcoming of the renormalization penalty method is sensitivity to parameter choice.

5.2. Tilt-twist equilibrium configuration. For this problem, $\mathbf{n}$ retains the form in (5.1) and the same boundary conditions are applied with $\theta_0 = \frac{\pi}{4}$ and Frank constants of $K_1 = 1.0$, $K_2 = 3.0$, and $K_3 = 1.2$. Twist solutions incorporating a nonplanar tilt deviating from parallel alignment with the xz-plane are investigated in [39, 41]. There, it is shown that nonplanar twist solutions become energetically optimal at a computable threshold depending on the relationship of the Frank constants, which is satisfied for the chosen parameters. The resulting configuration considered below is not detectable using a one-constant simplification and highlights the accuracy of the energy-minimization approach. The analytical, energy-minimizing, tilt-twist solution is defined implicitly for a rotated coordinate system in [39, 55]. The associated analytical, free energy for the chosen parameters is 3.59294.

For the tilt-twist equilibrium solution, the damped Newton stepping approach converged for all the penalty weights considered. Table 6 details the statistics for the unmodified penalty method compared with the Lagrange multiplier method. Again, the Lagrange multiplier method outperforms the penalty method in each category. The free energy of the Lagrange multiplier method is not obtained by the penalty method until ζ reaches 10^8.

It should be noted that the behavior of the error for the Lagrangian method, as well as the penalty method for weights greater than 10^7, is affected by the implicit definition of the true solution. The analytical solution for the tilt-twist equilibrium configuration is implicitly defined by a complicated set of equations, which are solved approximately at the appropriate quadrature points using Mathematica. Solving these equations involves successive root finding for an intricate set of integrals where the unknowns are limits of integration. Approximation error creates an artificial limit for the computed error at accuracies smaller than 10^{-7}.

Considering Table 7, the renormalization penalty method does not perform as well as in the previous problem. For each of the methods outlined in the table, the renormalization approach fails to reach an equivalent accuracy before performance degrades. In addition, as with the simpler twist problem, performance of the renormalization method is sensitive to an appropriate choice of penalty weight. However, the method does find the true free energy at a lower penalty weight, $\zeta = 10^4$, than the approach without renormalization. For $\zeta = 10^4$, the penalty method without renormalization has a slightly lower error measure but has not accurately matched the analytical free energy. While the unmodified method more accurately resolves the orientation of the director in comparison with the renormalization method, it slightly shrinks the director length to attain the smaller free energy seen in Table 6.

Compared to the unmodified penalty method, computational costs generally remain steadier with increasing penalty weight. It is also apparent, in both Tables 3 and 7, that trust regions improve both robustness and accuracy for the renormalization

TABLE 6

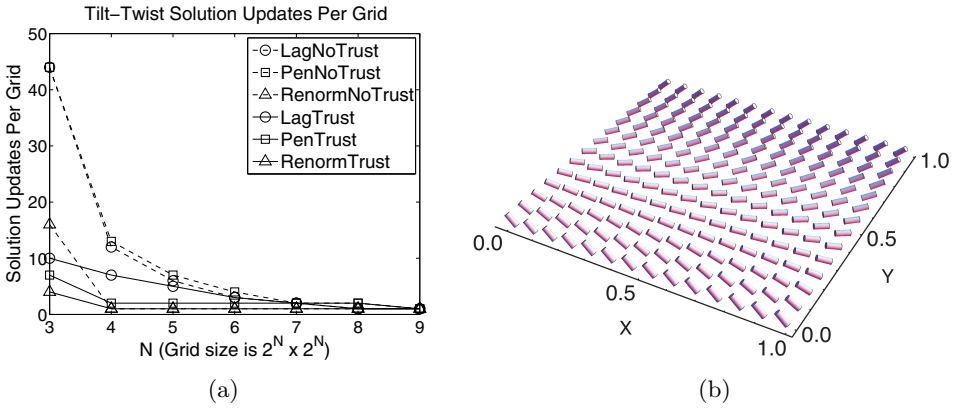
Statistics for the tilt-twist equilibrium solution with the different formulations and penalty weights. Included is the system free energy, the computed L^2-error on the finest grid, and the minimum and maximum deviations from unit director length at the quadrature nodes. Approximations of the cost in WUs for the corresponding method with no trust regions and simple trust regions are included.

Type	Free energy	L^2-error	Min. Dev.	Max Dev.	Cost	TR cost
Lagrangian	3.59294	4.717e-07	−7.89e-10	7.88e-10	1.463	1.447
Pen. $\zeta = 10^4$	3.59052	4.606e-04	−5.00e-04	−1.21e-05	2.735	2.665
Pen. $\zeta = 10^5$	3.59269	4.590e-05	−5.00e-05	−3.56e-06	2.743	2.667
Pen. $\zeta = 10^6$	3.59291	4.253e-06	−5.01e-06	−8.05e-07	2.782	2.678
Pen. $\zeta = 10^7$	3.59293	2.735e-07	−5.83e-07	−1.14e-07	2.809	2.723
Pen. $\zeta = 10^8$	3.59294	4.340e-07	−6.00e-08	−1.22e-08	2.885	2.747
Pen. $\zeta = 10^9$	3.59294	4.676e-07	−6.01e-09	−1.24e-09	3.218	2.879

TABLE 7

A comparison of renormalization penalty methods, with and without trust-region approaches, for the tilt-twist solution. For each algorithm, the computed L^2-error on the finest grid and an approximation of the cost in WUs are included.

	No trust region		Simple trust region		2D trust region	
Type	L^2-error	Cost	L^2-error	Cost	L^2-error	Cost
Pen. $\zeta = 10^3$	4.708e-03	1.335	4.533e-03	1.332	4.493e-03	1.332
Pen. $\zeta = 10^4$	1.085e-03	1.335	8.662e-04	1.332	8.536e-04	1.333
Pen. $\zeta = 10^5$	1.650e-03	1.336	9.487e-04	1.333	7.012e-04	1.333
Pen. $\zeta = 10^6$	9.414e-01	87.362	5.375e-04	1.341	7.344e-04	1.341



(a) (b)

FIG. 2. (a) *Number of iterations required to reach iteration tolerance for each method with NI. The penalty weight for the penalty formulation was $\zeta = 1000$. Only the 2D-subspace minimization trust-region approach is displayed, as the behavior of simple trust regions is similar.* (b) *The final computed solution for the Lagrangian formulation on a 512×512 mesh (restricted for visualization).*

method. In Table 7, for a penalty weight of 10^6, without trust regions, there is a large jump in computational cost and a significant loss of accuracy which is controlled in the presence of either trust-region approach.

Figure 2(a) presents similar behavior to Figure 1(a), in that trust regions productively reduce the number of iterations on the coarsest grids but have less effect on

iteration counts on the finest levels. This is again due to the efficacy of NI. Nonetheless, for the unmodified penalty method with NI and a penalty weight of $\zeta = 10^9$, trust regions successfully reduce the computational cost from 3.218 WUs to 2.879 WUs. This results in an 8.9% decrease in overall time to solution. Improvements from trust regions for the Lagrange multiplier method are smaller, decreasing computational costs from 1.463 WUs to 1.447 WUs and reducing overall time to solution by only 0.82%. Figure 2(b) displays the solution computed by the Lagrange multiplier approach.

5.3. Nano-patterned boundary conditions. In this numerical experiment, we use Frank constants $K_1 = 1.0$, $K_2 = 0.62903$, and $K_3 = 1.32258$. Letting $r = 0.25$ and $s = 0.95$, the boundary conditions are defined as

$$n_1 = 0, \tag{5.2}$$

$$n_2 = \cos\big(r(\pi + 2\tan^{-1}(X_m) - 2\tan^{-1}(X_p))\big), \tag{5.3}$$

$$n_3 = \sin\big(r(\pi + 2\tan^{-1}(X_m) - 2\tan^{-1}(X_p))\big), \tag{5.4}$$

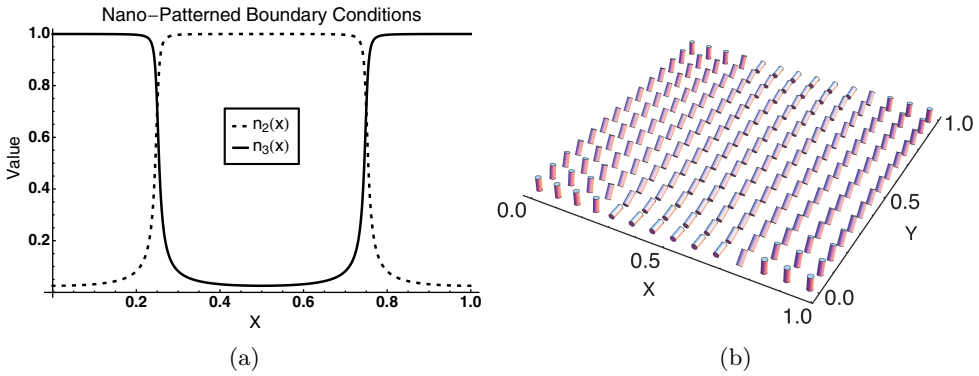
where $X_m = \frac{-s\sin(2\pi(x+r))}{-s\cos(2\pi(x+r))-1}$ and $X_p = \frac{-s\sin(2\pi(x+r))}{-s\cos(2\pi(x+r))+1}$. The result is a sharp transition from vertical nematics to planar-aligned rods followed by a rapid transition back to vertical alignment at each boundary, as shown in Figure 3. Such boundary conditions produce configuration distortions throughout the interior of the domain. Due to this complexity, no known analytical solution currently exists. The performance of the Lagrangian formulation for this problem was studied in [2], but the challenging nature of enforcing the unit-length constraint observed therein makes the problem a good candidate for comparing the constraint enforcement techniques discussed above. Moreover, as the configuration is important in physical applications [6, 7], performance across methods is pertinent for determining the most effective approach.

The more complicated nature of the nano-patterned boundary conditions is reflected in the data of Table 8. The overall approximate costs for the methods with and without trust regions are larger than previous examples and the unit-length constraint is more difficult to capture. Nevertheless, the Lagrange multiplier method provides an accurate and cost-effective approach. The penalty method without trust regions diverges for penalty weights greater than $\zeta = 10^4$. At higher penalty weights, even the trust-region approach suffers jumps in computational costs. At $\zeta = 10^9$, the system becomes overconstrained and accuracy begins to degrade. Hence, results for this weight are not included.

As with the tilt-twist equilibrium solution, the renormalization penalty method approaches the Lagrangian formulation's free energy and unit-length constraint bounds earlier than the unmodified penalty method, at $\zeta = 10^5$. It also yields a lower computational cost for most penalty weights. However, as was seen in the tilt-twist data, matching the energy earlier than the unmodified penalty approach does not directly indicate higher accuracy in resolving the correct orientation of the director. Moreover, notable divergence issues are apparent in Table 9 for the renormalization method at high penalty weights, even when applying the simple trust-region scheme.

Tables 8 and 9 reinforce the conclusion that trust regions positively influence the robustness of penalty method approaches. While the simple trust-region approach works most effectively for the nonrenormalization penalty method, the 2D-subspace minimization approach is more favorable for the renormalization penalty formulation.

Table 10 reiterates the efficacy of NI for efficient computation and trust regions for robustness. The cost savings from trust regions within an NI scheme are slightly higher for this problem due to its complexity. Table 10 also shows that the renormalization



(a) (b)

FIG. 3. (a) *Line plot representing values of the director components n_2 (dashed) and n_3 (solid) along the boundary for nano-patterned boundary conditions.* (b) *The final computed solution for the Lagrangian formulation on a 512×512 mesh (restricted for visualization).*

TABLE 8

Statistics for the nano-patterned equilibrium solution with the different formulations and penalty weights. Included is the system free energy and the minimum and maximum deviations from unit director length at the quadrature nodes. Approximations of the cost in WUs for the corresponding method with no trust regions and simple trust regions are included. Dashes indicate divergence.

Type	Free energy	Min. Dev.	Max Dev.	Cost	TR cost
Lagrangian	3.89001	−6.92e-05	5.89e-05	2.864	2.779
Pen. $\zeta = 10^3$	3.88331	−1.80e-02	7.32e-03	2.868	2.749
Pen. $\zeta = 10^4$	3.88819	−6.58e-03	5.81e-03	2.886	2.757
Pen. $\zeta = 10^5$	3.88965	−1.60e-03	2.01e-03	–	2.805
Pen. $\zeta = 10^6$	3.88996	−2.90e-04	4.55e-04	–	3.736
Pen. $\zeta = 10^7$	3.89001	−7.92e-05	1.01e-04	–	4.797
Pen. $\zeta = 10^8$	3.89001	−6.76e-05	5.83e-05	–	22.328

TABLE 9

A comparison of renormalization penalty methods, with and without trust-region approaches, for the nano-pattern solution. For each algorithm, the computed free energy on the finest grid and an approximation of the cost in WUs is included. Dashes indicate divergence.

	No trust region		Simple trust region		2D trust region	
Type	Free energy	Cost	Free energy	Cost	Free energy	Cost
Pen. $\zeta = 10^3$	3.89006	1.682	3.89006	1.666	3.89006	1.666
Pen. $\zeta = 10^4$	3.89002	1.683	3.89002	1.669	3.89002	1.669
Pen. $\zeta = 10^5$	–	–	3.89001	2.133	3.89001	2.433
Pen. $\zeta = 10^6$	–	–	–	–	3.89001	5.418

penalty method with NI and trust regions has a somewhat shorter overall run time than that of the Lagrange multiplier approach. Moreover, the renormalization approach matches the free-energy and unit-length conformance of the Lagrangian formulation. However, while the overall run time and approximate cost of the approach are slightly larger, the accuracy of the Lagrange multiplier formulation is expected to be much higher. For the Lagrange multiplier approach, the l_2-norm of the first-order optimality conditions is 7.386e-13, whereas the same measure for the renormalization penalty method is 1.603e-02.

TABLE 10

Nano-pattern statistics comparison for NI and trust region combinations. The solve cost column displays an approximation of the work in WUs for the corresponding method. The overall time to solution is also presented. Dashes indicate divergence.

Lagrangian			
Method		Solve cost	Run time
No NI	No TR	63	18,861s
NI	No TR	2.864	983s
No NI	TR	10	3,113s
NI	TR	2.779	960s
Renormalization penalty: $\zeta = 10^5$			
Method		Solve cost	Run time
No NI	No TR	35	10,918s
NI	No TR	–	–
No NI	TR	32	9,893s
No NI	TR 2D	34	10,976s
NI	TR	2.133	789s
NI	TR 2D	2.433	901s

Unmodified penalty $\zeta = 10^5$			
Method		Solve cost	Run time
No NI	No TR	169	49,654s
NI	No TR	–	–
No NI	TR	73	21,415s
No NI	TR 2D	75	22,366s
NI	TR	2.805	958s
NI	TR 2D	3.530	1,202s

In all the experiments above, the accuracy per unit cost of the Lagrange multiplier method convincingly outperforms that of either of the penalty methods. In agreement with previous experience in [6], where large weights were needed to ensure appropriate unit-length conformance, the performance of both penalty methods is intricately linked with the magnitude of the penalty parameter and efficiency degrades as weights increase. As discussed above, a balance between strict unit-length enforcement and the convergence and efficiency of the method must be sought. Problems with more drastic deformation features or larger intrinsic free energy generally require larger penalty weights to ensure accurate free-energy representation and unit-length coherence. The Lagrangian formulation offers better constraint conformance and advanced linear solvers for saddle point problems, like the multigrid solver proposed below, are available to further increase its efficiency.

Trust regions significantly improve the robustness of the penalty approach, enabling the method to attain accuracy comparable to that of the Lagrange multiplier scheme. The associated robustness improvements and the limited cost of the trust-region methods make them attractive for either constraint formulation. The simple trust-region approach works best for the unmodified penalty method with stopping tolerances based on the first-order optimality conditions, whereas the 2D-subspace minimization trust regions are most effective for the renormalization penalty method with an energy reduction based stopping tolerance. Though larger penalty weights are generally necessary, the unmodified penalty method offers more consistent error reduction and performance with respect to an increasing weight.

6. Multigrid solver. With the superiority of the Lagrange multiplier approach paired with NI and trust regions established above, we demonstrate the full efficiency of the scheme for problems with electric and flexoelectric coupling as well as more complicated nano-patterened boundary conditions. In addition, a highly efficient, coupled multigrid method for the associated linear systems is introduced. Let $\bar{\phi}$ be an electric potential, let $\bar{\epsilon}_0$ denote the permittivity of free space, and let $\bar{e}_s$ and $\bar{e}_b$ be flexoelectric constants following the sign convention of Rudquist and Lagerwall [52]. Further, let the dimensionless constants $\epsilon_\perp$ and ϵ_a denote the perpendicular dielectric permittivity and dielectric anisotropy of the liquid crystal, respectively. Finally, let

$\phi_0 > 0$ be a characteristic voltage and define a dimensionless potential $\phi = \frac{\bar{\phi}}{\phi_0}$. We then construct dimensionless parameters $\epsilon_0 = \frac{\bar{\epsilon}_0 \phi_0^2}{K}$, $e_s = \frac{\bar{e}_s \phi_0}{K}$, and $e_b = \frac{\bar{e}_b \phi_0}{K}$. Nondimensionalizing in similar fashion to the elastic functional, the dimensionless flexoelectric functional with appropriate boundary conditions is

$$\begin{aligned}\mathcal{F}(\mathbf{n}, \phi) = K_1 \|\nabla \cdot \mathbf{n}\|_0^2 + K_3 \langle \mathbf{Z} \nabla \times \mathbf{n}, \nabla \times \mathbf{n} \rangle_0 - \epsilon_0 \epsilon_\perp \langle \nabla \phi, \nabla \phi \rangle_0 \\ - \epsilon_0 \epsilon_a \langle \mathbf{n} \cdot \nabla \phi, \mathbf{n} \cdot \nabla \phi \rangle_0 + 2 e_s \langle \nabla \cdot \mathbf{n}, \mathbf{n} \cdot \nabla \phi \rangle_0 + 2 e_b \langle \mathbf{n} \times \nabla \times \mathbf{n}, \nabla \phi \rangle_0.\end{aligned}$$

For a full derivation of the Lagrange multiplier approach extended to include electric and flexoelectric effects, see [1]. Using the electric potential, ϕ, the discretized and linearized system is written

$$\mathcal{M} \begin{bmatrix} \mathbf{n} \\ \phi \\ \lambda \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 \\ B_1^T & -D & \mathbf{0} \\ B_2^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{n} \\ \phi \\ \lambda \end{bmatrix} = \begin{bmatrix} f_{\mathbf{n}} \\ f_\phi \\ f_\lambda \end{bmatrix}.$$

Define blocks of $\mathcal{M}$ as

$$\hat{A} = \begin{bmatrix} A & B_1 \\ B_1^T & -D \end{bmatrix}, \qquad \hat{B} = \begin{bmatrix} B_2 \\ \mathbf{0} \end{bmatrix}. \tag{6.1}$$

Furthermore, let $\hat{u} = [\mathbf{n}\ \phi]^T$ and $\hat{f}_{\hat{u}} = [f_{\mathbf{n}}\ f_\phi]^T$. With these block definitions, the "exact" Braess–Sarazin update, originally formulated in [13] for Stokes flows, is used and takes the form

$$\begin{bmatrix} \hat{u}_{k+1} \\ \lambda_{k+1} \end{bmatrix} = \begin{bmatrix} \hat{u}_k \\ \lambda_k \end{bmatrix} + \begin{bmatrix} \gamma_b R & \hat{B} \\ \hat{B}^T & \mathbf{0} \end{bmatrix}^{-1} \left(\begin{bmatrix} \hat{f}_{\hat{u}} \\ f_\lambda \end{bmatrix} - \begin{bmatrix} \hat{A} & \hat{B} \\ \hat{B}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \hat{u}_k \\ \lambda_k \end{bmatrix} \right), \tag{6.2}$$

where R is an appropriate preconditioner for $\hat{A}$ and γ_b is a weighting parameter. For our multigrid approach, the matrix

$$\begin{bmatrix} \gamma_b R & \hat{B} \\ \hat{B}^T & \mathbf{0} \end{bmatrix}$$

is only approximately inverted, leading to an "inexact" Braess–Sarazin method [61]. We use Q_2 elements for both $\mathbf{n}$ and ϕ, and, hence, the degrees of freedom for the components of $\mathbf{n}$ and ϕ are collocated. Considering the work in [3], we construct the preconditioner, R, by extracting 4×4 blocks of $\hat{A}$ corresponding to the nodally collocated degrees of freedom for $\mathbf{n}$ and ϕ. With careful permutation of the degrees of freedom in (6.2), R becomes a block-diagonal matrix consisting of these 4×4 collocation blocks.

Comparative studies of Braess–Sarazin-type relaxation schemes in a multigrid framework for Stokes flows are found in [37], while numerical studies of their extension to multigrid methods for magnetohydrodynamic equations are performed in [3]. Here, as part of the underlying multigrid method, we use standard finite-element interpolation operators and Galerkin coarsening. In this section, we investigate the performance of the multigrid method as a preconditioner for GMRES.

We first study the optimal value of the relaxation parameter. The solver convergence tolerance, which is based on a ratio of the current solution's residual to that of the initial guess, remains fixed at 10^{-6} for each grid level and Newton step. For these experiments, the parameter γ_b was varied from 1.00 to 2.00 in increments of

TABLE 11
Relevant dimensionless constants for parameter study simulations.

	K_1	K_2	K_3	$\epsilon_\perp$	ϵ_a	e_s	e_b
Simple flexo.	1.0	4.0	1.0	7.0	0.0	1.5	-1.5
5CB flexo.	1.0	0.62903	1.32258	7.0	11.5	2.5	-2.5
Fréedericksz	1.0	0.62903	1.32258	7.0	11.5	–	–

0.05. Displayed in Figure 4(a) are the pertinent multigrid iteration counts, averaged over Newton iterations, on a 512×512 grid with respect to varying values of γ_b for three problems with electric or flexoelectric coupling. The boundary conditions for the two flexoelectric problems consider a doubling of the nano-pattern described by (5.2)–(5.4) such that the pattern contains a second strip parallel to the xy-plane. Therefore, these problems have four sharp transitions at each boundary. While there is no applied electric field, the curvature induced by the nano-patterning generates an internal electric field due to the flexoelectric properties of the liquid crystals [1, 47]. The electric problem models a Fréedericksz transition [27], where, at the Dirichlet boundaries, $\mathbf{n} = (1, 0, 0)^T$ and a voltage is applied such that $\phi = 1$ at $y = 1$ and $\phi = 0$ at $y = 0$. With the Frank constants used in this problem, the applied voltage is above the critical threshold, inducing a true Fréedericksz transition. For these simulations, σ remains equal to 10^{-6} m, the characteristic Frank constant is again taken to be $K = 6.2 \times 10^{-12}$ N, the $\bar{K}_1$ value for 5CB, and $\phi_0 = 1$ V. These values imply that $\epsilon_0 = 1.42809$. Additional relevant, dimensionless, elastic, and electric parameters are listed in Table 11.

The parameter studies suggest that the optimum γ_b value, while problem dependent, varies only slightly, ranging from 1.10 to 1.20. Additionally, the iteration counts are relatively insensitive to increases in γ_b above the ideal value. Correspondingly, the multigrid solve timings remain relatively robust for γ_b values above the optimal magnitude. Similar behavior is observed in applications to magnetohydrodynamic simulations where the best choice for γ_b is 1.0 [3]. While choosing an appropriate value for γ_b is important, the method is resilient and optimal scaling persists with respect to parameter choices above the ideal value. Moreover, theory informing the choice of γ_b as part of standard Braess–Sarazin relaxation for fluids exists in [37], where a lower bound for γ_b is prescribed.

Figure 4(b) displays the average total setup and solve times across grids and varying solver tolerances for the Braess–Sarazin-type multigrid scheme, compared with the UMFPACK direct solver, applied to the flexoelectric problem with the 5CB constants outlined in Table 11. The optimal parameter value, $\gamma_b = 1.10$, is used. With Q_2–Q_2–P_0 elements for $\mathbf{n}$, ϕ, and λ, respectively, the matrices on the 512×512 grid are of dimension 4,464,644 $\times$ 4,464,644 with 289,969,900 nonzero entries. Note that the number of Newton steps does not vary for the multigrid tolerances used and exactly matches that of the method with the UMFPACK solver, implying that the Newton method is robust with respect to solver tolerance within a reasonable range.

On the finest grid, the multigrid solver is more than 2.8 times faster than the direct approach for a linear solve tolerance of 10^{-5} and 3.8 times faster with a tolerance of 10^{-2}. Furthermore, the Braess–Sarazin multigrid scheme scales optimally with grid size for each of the tolerance values and outpaces the direct approach by the 16×16 mesh for solver tolerances up to 10^{-8}. These timing intersections occur considerably earlier than for the multigrid method with Vanka-type relaxation discussed in [1]. Moreover, the optimal scaling remains for larger values of γ_b, with the multigrid-

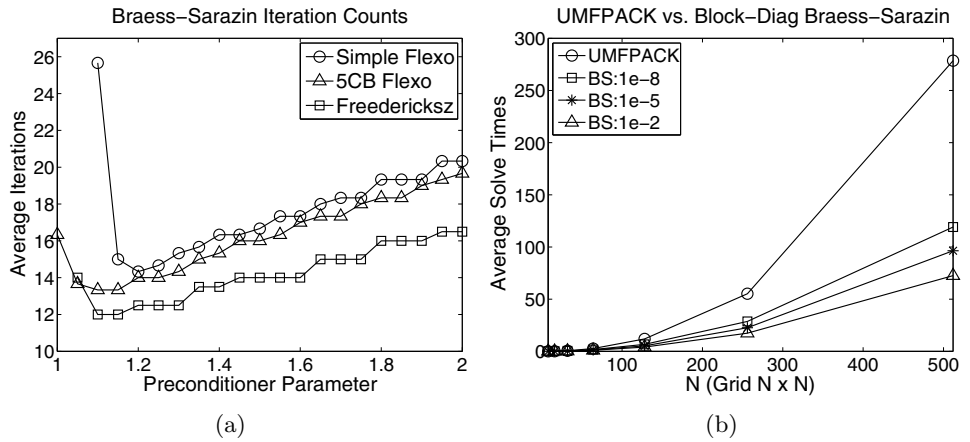


FIG. 4. (a) *The average number of multigrid iterations for varying* γ_b *on a* 512×512 *grid.* (b) *The average time to solution for the Braess–Sarazin-type multigrid scheme with varying solver tolerances compared to the UMFPACK direct solver.*

TABLE 12

A comparison of timing breakdowns for runs using the UMFPACK direct solver or the Braess–Sarazin multigrid scheme. Each solver is run with and without trust regions. The applied solver tolerance is listed in parentheses.

	UMFPACK		Braess–Sarazin (1e-5)		Braess–Sarazin (1e-2)	
Trust-region type	None	Simple	None	Simple	None	Simple
System assembly	138.5s	133.6s	136.0s	130.4s	135.2s	131.1s
Data conversion	–	–	137.0s	132.5s	137.1s	132.8s
Linear setup/solve	1091.0s	1084.2s	391.0s	378.5s	296.0s	290.2s
Memory/output	277.8s	279.6s	305.9s	301.4s	305.8s	302.9s
Total time	1507.3s	1497.4s	969.9s	942.8s	874.1s	857.0s

based approach outperforming UMFPACK on all mesh sizes from 32×32 onward for values as large as $\gamma_b = 2.0$ at all solver tolerances. With a preconditioner parameter of $\gamma_b = 2.0$ and a solver tolerance of 10^{-6}, the multigrid method remains more than twice as fast as the UMFPACK solver on the finest level.

Table 12 details a timing breakdown of the UMFPACK direct solver's performance compared to that of the Braess–Sarazin-type multigrid scheme for linear solve tolerances of 10^{-5} and 10^{-2}. With and without trust regions, the computed free energy between the solvers is identically 7.927. However, Braess–Sarazin-type multigrid reduces overall run time by approximately 37% and 43%, respectively. This speedup is most notable when considering the fact that overall run time for the multigrid experiments includes porting variables to types compatible with the Trilinos computational library [32] and computing collocation information for the Braess–Sarazin-type relaxation. The solver demonstrates good performance for the flexoelectric simulation incorporating sharp transitions considered here, as well as problems with steep boundary layers, as in [3]. Using the Lagrange multiplier formulation, NI, multigrid, and trust regions, we obtain a robust and efficient algorithm.

7. Discussion and future work. In this paper, we have compared the performance of the Lagrange multiplier method developed in [2] with a penalty formulation. Such penalty approaches are often applied in liquid crystal simulations to enforce the

challenging unit-length constraint. The experiments also included a renormalization penalty method motivated by the well-posedness theory presented in section 3. The ensuing algorithms were compared for three benchmark equilibrium problems. The experiments suggested that the Lagrange multiplier method coupled with NI is the most accurate and efficient approach for enforcing the unit-length constraint. Improvements for penalty formulations, such as augmented Lagrangian schemes, exist but require thoughtful adaptation for use in the context of finite-element methods and pointwise constraints. These schemes will be studied in future work as an intermediate approach between the two methods [49].

The numerical results, while not perfectly generalizable, edify the performance of these types of Newton methods in other settings, as this application shares structural similarities with other problems considering coupled multiphysics and constraints, such as magnetohydrodynamics. In addition, the results parallel the preference for two- and three-field formulations in the context of fully incompressible elasticity in place of penalty-type methods (commonly called nearly incompressible formulations) to improve accuracy and avoid phenomena such as volumetric locking [11, 22].

Trust-region schemes, developed in the context of finite-element discretizations, were introduced and readily improved robustness, enabling competitive accuracy for the penalty formulations. These methods naturally integrate into the energy-minimization framework with very little extra cost. In particular, the trust-region method introduced for the Lagrange multiplier formulation in section 4.2 maintains discretization sparsity while still effectively measuring step merit. Such a method is generalizable to a broad range of nonlinear optimization problems using finite-element discretizations. Additionally, without NI, the applied trust regions significantly accelerate convergence.

All the gathered experimental results imply that NI should be used when considering any of the methods above. More broadly, so long as sufficiently good approximations are representable on coarser grids, NI may be productively applied in the context of general Newton-type methods, in conjunction with trust regions, to improve approximations on finer grids while emphatically reducing computational work. In addition, we expect the precision and efficiency of the iterative schemes to be sustained for a variety of finite-element discretizations. The discretization spaces used in the numerical experiments are considered due to their relationship with the well-posedness results in [2]. Nevertheless, the energy-minimization implementation and trust-region methods are not discretization dependent. Moreover, multigrid methods with Braess–Sarazin-type relaxation have been successfully applied for other discretizations and applications [3, 4]. While the use of higher-order elements may affect the conditioning of the linear systems, thereby impacting linear solver iteration counts, improvements precipitating from NI and trust regions could also grow.

In the final section, a Braess–Sarazin-type multigrid method, based on work in [3], was introduced for the Lagrange multiplier formulation and successfully applied to highly difficult liquid crystal problems with electric and flexoelectric coupling. The parameter studies indicated robustness with respect to larger than optimal values of the weighting parameter, γ_b, and optimal scaling was observed. Furthermore, the method quickly and accurately computed the expected equilibrium configurations.

The current implementation utilizes uniform grid refinement to build the NI hierarchy of grids. Future work will include study of adaptive refinement techniques. Because the energy minimization formulation does not yield an obvious a posteriori error estimator, new techniques will be explored to flag cells for refinement.

With an accurate and efficient approach for liquid crystal equilibrium configurations established, research into the application of energy-minimization finite-element approaches to liquid crystal flow problems will also be undertaken.

Acknowledgments. The authors would like to thank Professor Timothy Atherton for his useful suggestions and contributions. We also thank Thomas Benson for allowing us to use and adapt his code, as well as for his helpful contributions. Finally, we would like to thank the referees for their diligent reading and valuable feedback.

REFERENCES

[1] J. H. Adler, T. J. Atherton, T. R. Benson, D. B. Emerson, and S. P. MacLachlan, *Energy minimization for liquid crystal equilibrium with electric and flexoelectric effects*, SIAM J. Sci. Comput., 37 (2015), pp. S157–S176.

[2] J. H. Adler, T. J. Atherton, D. B. Emerson, and S. P. MacLachlan, *An energy-minimization finite-element approach for the Frank–Oseen model of nematic liquid crystals*, SIAM J. Numer. Anal., 53 (2015), pp. 2226–2254.

[3] J. H. Adler, T. R. Benson, E. C. Cyr, S. P. MacLachlan, and R. S. Tuminaro, *Monolithic multigrid methods for 2D resistive magnetohydrodynamics*, SIAM J. Sci. Comput., to appear.

[4] J. H. Adler, T. R. Benson, and S. P. MacLachlan, *Preconditioning a mass-conserving discontinuous Galerkin discretization of the Stokes equations*, Numer. Linear Algebra Appl., to appear.

[5] J. H. Adler, D. B. Emerson, S. P. MacLachlan, and T. A. Manteuffel, *Constrained Optimization for Liquid Crystal Equilibria: Extended Results*, Tech. report, Tufts University, 2014.

[6] T. J. Atherton and J. H. Adler, *Competition of elasticity and flexoelectricity for bistable alignment of nematic liquid crystals on patterned surfaces*, Phys. Rev. E, 86 (2012).

[7] T. J. Atherton and J. R. Sambles, *Orientational transition in a nematic liquid crystal at a patterned surface*, Phys. Rev. E, 74 (2006).

[8] S. Badia, F. Guillén-Gónzalez, and J. V. Gutiérrez-Santacreu, *Finite element approximation of nematic liquid crystal flows using a saddle-point structure*, J. Comput. Phys., 230 (2011), pp. 1686–1706.

[9] S. Badia, F. Guillén-Gónzalez, and J. V. Gutiérrez-Santacreu, *An overview on numerical analyses of nematic liquid crystal flows*, Arch. Comput. Methods Eng., 18 (2011), pp. 285–313.

[10] W. Bangerth, R. Hartmann, and G. Kanschat, *deal.*II*—a general purpose object oriented finite element library*, ACM Trans. Math. Software, 33 (2007), pp. 24/1–24/27.

[11] T. Belytschko, W. K. Liu, B. Moran, and K. Elkhodary, *Nonlinear Finite Elements for Continua and Structures*, 2nd ed., Wiley, New York, 2014.

[12] D. Braess, *Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics*, Cambridge University Press, Cambridge, UK, 1997.

[13] D. Braess and R. Sarazin, *An efficient smoother for the Stokes problem*, Appl. Numer. Math., 23 (1997), pp. 3–19.

[14] R. H. Byrd, R. B. Schnabel, and G. A. Shultz, *A trust region algorithm for nonlinearly constrained optimization*, SIAM J. Numer. Anal., 24 (1987), pp. 1152–1170.

[15] R. H. Byrd, R. B. Schnabel, and G. A. Shultz, *Approximate solution of the trust region problem by minimization over two-dimensional subspaces*, Math. Program., 40 (1988), pp. 247–263.

[16] R. Cohen, R. Hardt, D. Kinderlehrer, S. Lin, and M. Luskin, *Minimum energy configurations for liquid crystals: Computational results*, in Theory and Applications of Liquid Crystals, IMA Vol. Math. Appl. 5, Springer-Verlag, Berlin, 1987, pp. 99–121.

[17] A. J. Davidson and N. J. Mottram, *Flexoelectric switching in a bistable nematic device*, Phys. Rev. E, 65 (2002).

[18] T. A. Davis, *Algorithm* 832*: UMFPACK, an unsymmetric-pattern multifrontal method*, ACM Trans. Math. Software, 30 (2004), pp. 196–199.

[19] T. A. Davis and I. S. Duff, *An unsymmetric-pattern mulitfrontal method for sparse LU factorization*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 140–158.

[20] T. A. Davis and E. C. Gartland, Jr., *Finite element analysis of the Landau-de Gennes minimization problem for liquid crystals*, SIAM J. Numer. Anal., 35 (1998), pp. 336–362.

[21] P. G. de Gennes and J. Prost, *The Physics of Liquid Crystals*, 2nd ed., Clarendon Press, Oxford, UK, 1993.
[22] I. Doghri, *Mechanics of Deformable Solids: Linear, Nonlinear, Analytical and Computational Aspects*, Springer-Verlag, Berlin, 2000.
[23] J. L. Ericksen, *Inequalities in liquid crystal theory*, Phys. Fluids, 9 (1966), pp. 1205–1207.
[24] J. Ericksen, *Continuum theory of nematic liquid crystals*, Res. Mechanica, 21 (1987), pp. 381–392.
[25] R. Fletcher, *Practical Methods of Optimization*, Vol. 1, Wiley, New York, 1980.
[26] F. C. Frank, *On the theory of liquid crystals*, Discuss. Faraday Soc., 25 (1958), pp. 19–28.
[27] V. Fréedericksz and V. Zolina, *Forces causing the orientation of an anisotropic liquid*, Trans. Faraday Soc., 29 (1933), pp. 919–930.
[28] E. C. Gartland, Jr., and A. Ramage, *A renormalized Newton method for liquid crystal director models with pointwise unit-vector constraints*, SIAM J. Numer. Anal., 53 (2015), pp. 251–278.
[29] V. Girault and P. Raviart, *Finite Element Methods for Navier-Stokes Equations*, Springer-Verlag, Berlin, 1986.
[30] R. Glowinski, P. Lin, and X. B. Pan, *An operator-splitting method for a liquid crystal model*, Comput. Phys. Comm., 152 (2003), pp. 242–252.
[31] R. Hardt, D. Kinderlehrer, and F. -H. Lin, *Existence and partial regularity of static liquid crystal configurations*, Commun. Math. Phys., 105 (1986), pp. 547–570.
[32] M. A. Heroux, R. A. Bartlett, V. E. Howle, R. J. Hoekstra, J. J. Hu, T. G. Kolda, R. B. Lehoucq, K. R. Long, R. P. Pawlowski, E. T. Phipps, A. G. Salinger, H. K. Thornquist, R. S. Tuminaro, J. M. Willenbring, A. Williams, and K. S. Stanley, *An overview of the Trilinos Project*, ACM Trans. Math. Software, 31 (2005), pp. 397–423.
[33] Q. Hu, X. -C. Tai, and R. Winther, *A saddle point approach to the computation of harmonic maps*, SIAM J. Numer. Anal., 47 (2009), pp. 1500–1523.
[34] Q. Hu and L. Yuan, *A Newton-penalty method for a simplified liquid crystal model*, Adv. Comput. Math., 40 (2014), pp. 201–244.
[35] M. Kružík and A. Prohl, *Recent developments in the modeling, analysis, and numerics of ferromagnetism*, SIAM Rev., 48 (2006), pp. 439–483.
[36] J. P. F. Lagerwall and G. Scalia, *A new era for liquid crystal research: Applications of liquid crystals in soft matter, nano-, bio- and microtechnology*, Current Appl. Phys., 12 (2012), pp. 1387–1412.
[37] M. Larin and A. Reusken, *A comparative study of efficient iterative solvers for generalized Stokes equations*, Numer. Linear Algebra Appl., 15 (2008), pp. 13–34.
[38] B. W. Lee and N. A. Clark, *Alignment of liquid crystals with patterned isotropic surfaces*, Science, 291 (2001), pp. 2576–2580.
[39] F. M. Leslie, *Distorted twisted orientation patterns in nematic liquid crystals*, Pramana, Suppl. 1 (1975), pp. 41–55.
[40] F. Leslie, *Theory of flow phenomenon in liquid crystals*, in The Theory of Liquid Crystals, Vol. 4, Academic Press, New York, 1979, pp. 1–81.
[41] F. M. Leslie, *Some topics in equilibrium theory of liquid crystals*, in Theory and Applications of Liquid Crystals, J. L. Ericksen and D. Kinderlehrer, eds., Springer-Verlag, New York, 1987, pp. 211–234.
[42] P. Lin, C. Liu, and H. Zhang, *An energy law preserving C0 finite element scheme for simulating the kinematic effects in liquid crystal flow dynamics*, J. Comput. Phys., 227 (2007), pp. 1411–1427.
[43] C. Liu and N. J. Walkington, *Mixed methods for the approximation of liquid crystal flows*, ESAIM Math. Model. Numer. Anal., 36 (2002), pp. 205–222.
[44] C. Liu, H. Zhang, and S. Zhang, *Numerical simulations of hydrodynamics of nematic liquid crystals: Effects of kinematic transports*, Commun. Comput. Phys., 9 (2011), pp. 974–993.
[45] A. Majumdar, C. J. P. Newton, J. M. Robbins, and M. Zyskin, *Topology and bistability in liquid crystal devices*, Phys. Rev. E, 75 (2007).
[46] N. Maratos, *Exact Penalty Function Algorithms for Finite Dimensional and Control Optimization*, Ph.D. thesis, University of London, 1978.
[47] R. B. Meyer, *Piezoelectric effects in liquid crystal*, Phys. Rev. Lett., 22 (1969).
[48] S. Mkaddem and E. C. Gartland, Jr., *Fine structure of defects in radial nematic droplets*, Phys. Rev. E, 62 (2000), pp. 6694–6705.
[49] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed., Springer-Verlag, New York, 2006.
[50] E. O. Omojokun, *Trust Region Algorithms for Optimization with Nonlinear Equality and Inequality Constraints*, Ph.D. thesis, University of Colorado, Boulder, 1989.

[51] A. Ramage and E. C. Gartland, Jr., *A preconditioned nullspace method for liquid crystal director modeling*, SIAM J. Sci. Comput., 35 (2013), pp. B226–B247.
[52] P. Rudquist and S. T. Lagerwall, *On the flexoelectric effect in nematics*, Liq. Cryst., 23 (1997), pp. 503–510.
[53] G. A. Shultz, R. B. Schnabel, and R. H. Byrd, *A family of trust-region-based algorithms for unconstrained minimization with strong global convergence properties*, SIAM J. Numer. Anal., 22 (1985), pp. 47–67.
[54] G. Starke, *Gauss-Newton multilevel methods for least-squares finite element computations of variably saturated subsurface flow*, Computing, 64 (2000), pp. 323–338.
[55] I. W. Stewart, *The Static and Dynamic Continuum Theory of Liquid Crystals: A Mathematical Introduction*, Taylor and Francis, London, 2004.
[56] K. Su and D. Pu, *A nonmonotone filter trust region method for nonlinear constrained optimization*, J. Comput. Appl. Math., 223 (2009), pp. 230–239.
[57] D. Thomsen, P. Keller, J. Naciri, R. Pink, H. Jeon, D. Shenoy, and B. Ratna, *Liquid crystal elastomers with mechanical properties of a muscle*, Macromolecules, 34 (2001), pp. 5868–5875.
[58] A. Vardi, *A trust region algorithm for equality constrained minimization: Convergence properties and implementation*, SIAM J. Numer. Anal., 22 (1985), pp. 575–591.
[59] E. G. Virga, *Variational Theories for Liquid Crystals*, Chapman and Hall, London, 1994.
[60] M. Yamada, M. Kondo, J. Mamiya, Y. Yu, M. Kinoshita, C. Barrett, and T. Ikeda, *Photomobile polymer materials: Towards light-driven plastic motors*, Angew. Chem. Int., 47 (2008), pp. 4986–4988.
[61] W. Zulehner, *A class of smoothers for saddle point problems*, Computing, 65 (2000), pp. 227–246.