Boise State University
ScholarWorks

Mathematics Faculty Publications and Presentations   Department of Mathematics

8-1-2008

An Efficient and Robust Method for Simulating Two-Phase Gel Dynamics

Grady Wright
Boise State University

Robert D. Guy
Boise State University

Aaron L. Fogelson
Boise State University

AN EFFICIENT AND ROBUST METHOD FOR SIMULATING TWO-PHASE GEL DYNAMICS[*]

GRADY B. WRIGHT[†], ROBERT D. GUY[‡], AND AARON L. FOGELSON[§]

Abstract. We develop a computational method for simulating models of gel dynamics where the gel is described by two phases: a networked polymer and a fluid solvent. The models consist of transport equations for the two phases, two coupled momentum equations, and a volume-averaged incompressibility constraint, which we discretize with finite differences/volumes. The momentum and incompressibility equations present the greatest numerical challenges since (i) they involve partial derivatives with variable coefficients that can vary quite significantly throughout the domain (when the phases separate), and (ii) their approximate solution requires the "inversion" of a large linear system of equations. For solving this system, we propose a box-type multigrid method to be used as a preconditioner for the generalized minimum residual (GMRES) method. Through numerical experiments of a model problem, which exhibits phase separation, we show that the computational cost of the method scales nearly linearly with the number of unknowns and performs consistently well over a wide range of parameters. For solving the transport equation, we use a conservative finite-volume method for which we derive stability bounds.

1. Introduction. Polymer gels are used in a variety of industrial applications and appear naturally in many biological systems. Gels are made of two materials: a polymer network and a solvent. By weight and volume gels are mostly solvent, but the rheology of the gel can range from a very viscous fluid to an elastic solid. In addition to viscoelastic stresses, gels exhibit chemical stresses which result in swelling and deswelling behavior.

A commonly used model for the mechanics of gels is the two-phase flow (or two-fluid) model [3, 9, 12, 11, 13, 18, 33]. Each phase, network and solvent, is treated as a continuum and moves according to its own velocity field. Each region in space is composed of a mixture of the two phases that is described by the volume fractions of the phases. The equations of motion for the gel consist of two coupled momentum equations and two continuity equations. These models of gels are based on mixture theory [15, 14], which has been used for many applications and is becoming increasingly popular in models of biological materials such as tissue [21], tumors [7, 23], cytoplasm [3, 12, 11, 18], and biofilms [2, 8, 9]. While much work has gone into advancing these models, the same is not true for the development of efficient numerical methods for simulating them. Since repeated simulation of models is crucial for validating their usefulness, the need for fast and robust computational methods is essential.

[†]Department of Mathematics, Boise State University, Boise, ID 83725-1555 (wright@math.boisestate.edu).

[‡]Department of Mathematics, University of California, Davis, CA 95616 (guy@math.ucdavis.edu).

[§]Departments of Mathematics and Bioengineering, University of Utah, Salt Lake City, UT 84112-0090 (fogelson@math.utah.edu).

In this study, an efficient and robust computational methodology for simulating gel dynamics is proposed for a model problem that shares several characteristics of many other models in the literature [3, 9, 12, 11, 18]. Both the network and solvent are modeled as viscous fluids. The viscosity of the network is much larger than that of the solvent. The gel feels a chemical pressure that makes the system bistable, that is, it drives the network volume fraction to one of two preferred states. Under appropriate conditions the chemical pressure induces phase separation of the mixture and is a good test for the robustness of any procedure. Numerical tests of the proposed method are carried out for the model in one and two dimensions, with the primary investigations of efficiency and robustness done for the one-dimensional (1-D) case.

In many models of gel dynamics, including our model problem, viscous terms are assumed to dominate so that inertial terms are negligible. This assumption leads to an elliptic system for the coupled momentum and incompressibility equations that resembles Stokes equations. However, there are many key differences which increase the complexity of the problem. First, since there are two phases, there are two sets of momentum equations. Second, there are off-diagonal terms coupling the two velocity fields (and the components of the two fields) together. Third, the Laplacians involve time-dependent variable coefficients related to the volume fractions of the two phases. Finally, incompressibility is replaced with a "volume averaged" incompressibility over the two fluids.

We use second-order finite differences on a marker-and-cell (MAC) grid [17] to discretize this system. As with Stokes (and linearized Navier–Stokes) equations, this gives rise to a large, sparse linear system of saddle point type. To treat this system we propose an extension of a multigrid method initially proposed for Stokes and Navier–Stokes equations by Vanka [36], and which has seen considerable development in the past several years (see, for example, [10, 24, 32, 34, 35, 39]). The method is characterized by the smoother used and is referred to in literature as box [35, pp. 320–322], coupled [39], or Vanka [24] relaxation. The extension of this method to the gel system is straightforward, since the basic idea of the smoother is to compute updates to the solution by collectively solving for the velocity and pressure in the discrete momentum equations locally, computational cell (or box) by cell. For the gel system, this means solving a $(2^{d+1}+1)$-by-$(2^{d+1}+1)$ saddle point system for each cell, where d is the number of spatial dimensions. Depending on how the cells are processed (Jacobi- or Gauss–Seidel-like), the method can also be viewed as an additive or multiplicative Schwarz domain decomposition method, where the subdomains consist of a single computational cell [24]. We use the smoother with standard prolongation and restriction operators and do a direct discretization of the equations on the coarse grids, which makes the implementation relatively simple. We compare both the V- and F-cycle techniques for cycling through the grids.

When the network and solvent are well mixed, the multigrid box relaxation method performs quite well as a solver for the model problem considered here. However, as phase separation occurs, the solver performance degrades quite dramatically, and it, in fact, fails in some cases. One particularly effective way of improving the robustness of a nonoptimal (or even nonconverging) multigrid method is to combine it with a Krylov subspace method (see, for example, [35, sect. 7.8] or [27]). In this case, multigrid is viewed as a preconditioner for the Krylov method. We use exactly this idea for the gel system, combining the box multigrid method with the generalized minimum residual (GMRES) method [28]. This combination appears to result in an efficient and robust method for solving the momentum and incompressibility equations. It scales nearly linearly with the number of unknowns and shows little

dependence on the mixture of the two phases and on the parameters.

In addition to the momentum and incompressibility equations, models for gel dynamics include equations for the transport of the network and solvent. For the transport equations we use a conservative finite-volume discretization in which the advection term is treated explicitly with first-order upwinding, and the diffusion term is handled implicitly with backward Euler. Since the coupled system of transport and momentum equations is nonlinear, additional constraints on the time step beyond the Courant-Friedrichs-Lewy (CFL) condition are needed. Using linear stability analysis, we derive stability constraints that also appear to hold in the full nonlinear problem.

The rest of the paper is organized as follows. In section 2, the model problem for describing and testing the new computational method is introduced. Section 3 describes the discretization (in space and time) for the 1-D model. This is followed by a derivation of the stability restriction on the time step in section 4. In section 5, the multigrid box relaxation scheme is introduced followed by a description of how it is used as a preconditioner for GMRES. Numerical results for the 1-D model problem are presented in section 6. We compare the performance of the multigrid technique as a solver and as a preconditioner and investigate the spectra of the operators to understand the convergence of both methods. In section 7, extensions of the method to the two-dimensional (2-D) model problem are discussed and numerical results presented. The paper concludes with some remarks on parallelization and other Krylov techniques for solving the momentum equations.

2. Model problem. The model problem we consider is for a gel composed of two immiscible materials, a polymer network and a fluid solvent. We assume that the total amount of gel remains constant and that the transport of the network and solvent is governed by the equations:

$$\frac{\partial \theta^{\mathrm{n}}}{\partial t} + \nabla \cdot (\theta^{\mathrm{n}} \mathbf{u}^{\mathrm{n}}) = 0, \tag{2.1}$$

$$\frac{\partial \theta^{\mathrm{s}}}{\partial t} + \nabla \cdot (\theta^{\mathrm{s}} \mathbf{u}^{\mathrm{s}}) = 0, \tag{2.2}$$

where θ^{n} and $\theta^{\mathrm{s}} = 1 - \theta^{\mathrm{n}}$ are the respective volume fractions of the network and solvent, $0 < \theta^{\mathrm{n}} < 1$, and $\mathbf{u}^{\mathrm{n}}$ and $\mathbf{u}^{\mathrm{s}}$ are the respective transport velocities. Adding these two equations, and using $\theta^{\mathrm{n}} + \theta^{\mathrm{s}} = 1$ gives the incompressibility-type constraint

$$\nabla \cdot (\theta^{\mathrm{n}} \mathbf{u}^{\mathrm{n}} + \theta^{\mathrm{s}} \mathbf{u}^{\mathrm{s}}) = 0. \tag{2.3}$$

The transport velocities are determined by conservation of momentum. We assume that the network acts as a constant density viscous material, and the solvent acts as a Newtonian fluid of much less viscosity. Viscous terms are assumed to dominate so that inertial terms are negligible (similar to zero-Reynolds number flow), i.e., the system responds instantaneously to applied forces.

The network and solvent are each subject to a number of intraphase stresses. We assume that the viscous stress tensors $\underline{\underline{\sigma}}^{\mathrm{n}}$ and $\underline{\underline{\sigma}}^{\mathrm{s}}$ for the network and solvent are proportional to the respective gradient of the network and solvent velocities, i.e.,

$$\underline{\underline{\sigma}}^{\mathrm{n}} = \mu_{\mathrm{n}} \left(\nabla \mathbf{u}^{\mathrm{n}} + \nabla \mathbf{u}^{\mathrm{n}\,T} \right) + \lambda_n \delta_{ij} \nabla \cdot \mathbf{u}^{\mathrm{n}}, \tag{2.4}$$

$$\underline{\underline{\sigma}}^{\mathrm{s}} = \mu_{\mathrm{s}} \left(\nabla \mathbf{u}^{\mathrm{s}} + \nabla \mathbf{u}^{\mathrm{s}\,T} \right) + \lambda_s \delta_{ij} \nabla \cdot \mathbf{u}^{\mathrm{s}}, \tag{2.5}$$

where $\mu_{\mathrm{n,s}}$ are shear viscosities and $\lambda_{\mathrm{n,s}} + 2\mu_{\mathrm{n,s}}/d$ are the bulk viscosities of the network and solvent (d is the dimension). The network and solvent are also subject
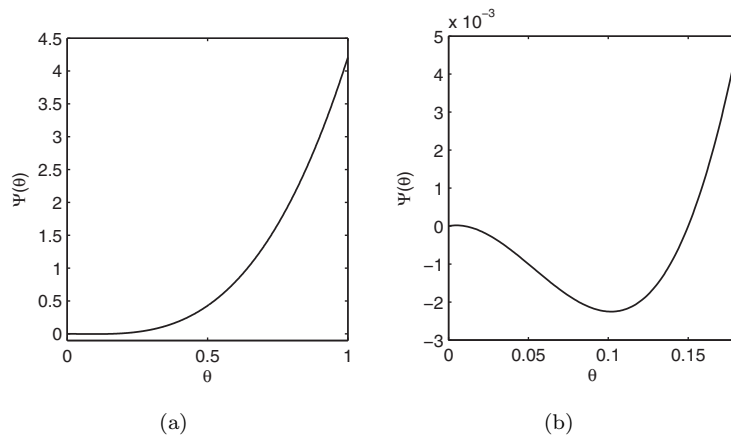
FIG. 1. (a) *Osmotic pressure function (2.8) from the model problem with parameters* $\gamma = 5$, $\theta_0^n = 0.01$, *and* $\theta_*^n = 0.15$ *used in the numerical experiments.* (b) *A detailed plot of this function near the origin.*

to a frictional drag, since the motion of the solvent influences the network. We model this by $\beta\theta^n\theta^s$, where $\beta > 0$ is the drag coefficient. The third force on each phase is due to hydrostatic pressure. Since the polymer is assumed to be chemically active within the gel and the solvent is assumed to be chemically neutral, the final force is generated by a chemical pressure $\Psi(\theta^n)$ and acts only on the network. The form of $\Psi(\theta^n)$ used in this paper is described below.

Balancing the above forces on the network and solvent yields the following equations:

$$(2.6) \qquad \nabla \cdot (\theta^n \underline{\sigma}^n) - \beta\theta^n\theta^s(\mathbf{u}^n - \mathbf{u}^s) - \theta^n \nabla p = \nabla\Psi(\theta^n),$$

$$(2.7) \qquad \nabla \cdot (\theta^s \underline{\sigma}^s) - \beta\theta^n\theta^s(\mathbf{u}^s - \mathbf{u}^n) - \theta^s \nabla p = 0,$$

where p is the hydrostatic pressure. These two equations, combined with (2.1) and (2.3) and subject to suitable boundary conditions, govern the gel dynamics and are the same as used in [3, 12, 11, 18]. For our test problem we assume no-slip boundary conditions for the network and solvent velocities, and no-flux boundary conditions for θ^n and θ^s.

The chemical pressure includes osmotic pressure, but it may also include active, contractile stresses such as in the actomyosin gels of cytoplasm. In this paper we are not concerned with the origins of the chemical stress, and refer to it simply as the osmotic pressure. We assume the osmotic pressure is of the form

$$(2.8) \qquad \Psi(\theta^n) = \gamma\theta^n(\theta^n - \theta_0^n)(\theta^n - \theta_*^n),$$

where $\gamma > 0$, $0 < \theta_0^n < \theta_*^n < 1$; see Figure 1 for a plot of $\Psi(\theta^n)$ with parameters used in the numerical experiments that follow. This functional form is chosen, since it can produce phase separation, or channeling. In the regions of space where $\Psi'(\theta^n) > 0$ the mixture is stable, but where $\Psi'(\theta^n) < 0$ the mixture tends to phase separate; see Cogan and Keener [9] for an analysis. This form of Ψ does not allow the phases to separate completely. Phase separation is generally observed in gels [33] and is vital for locomotion and for transporting nutrients in some amoeboid cells [1, 20].

Finally, we assume there is a small amount of diffusion between the network and solvent in the gel. This is mathematically and computationally useful, since it means the transition between the network and solvent will be smooth even in areas where

the two phases have separated. This additional assumption changes (2.1) and (2.2) to

$$(2.9) \qquad \frac{\partial \theta^n}{\partial t} + \nabla \cdot (\theta^n \mathbf{u}^n) = \kappa \nabla^2 \theta^n,$$

$$(2.10) \qquad \frac{\partial \theta^s}{\partial t} + \nabla \cdot (\theta^s \mathbf{u}^s) = \kappa \nabla^2 \theta^s,$$

where $\kappa \geq 0$ is the diffusion constant. We use relatively small diffusion coefficients so that this modification does not change the qualitative features of the model but ensures the solution is continuous.

2.1. One dimension. We describe and do the primary numerical investigations of our proposed computational methodology for the model problem in one dimension. Letting the spatial domain be $0 \leq x \leq 1$, the transport equation is

$$(2.11) \qquad \theta^n_t + (\theta^n u^n)_x = \kappa \theta^n_{xx},$$

while the momentum equations and incompressibility constraint are

$$(2.12) \qquad \alpha_n (\theta^n u^n_x)_x - \beta \theta^n \theta^s (u^n - v^n) - \theta^n p_x = (\Psi(\theta^n))_x,$$

$$(2.13) \qquad \alpha_s (\theta^s u^s_x)_x + \beta \theta^n \theta^s (u^n - v^n) - \theta^s p_x = 0,$$

$$(2.14) \qquad -(\theta^n u^n + \theta^s u^s)_x = 0,$$

where $\alpha_{n,s} = 2\mu_{n,s} + \lambda_{n,s}$. We can alternatively express the system in matrix-vector form as

$$(2.15) \qquad \underbrace{\begin{bmatrix} \alpha_n \partial_x (\theta^n \partial_x) - \beta \theta^n \theta^s & \beta \theta^n \theta^s & -\theta^n \partial_x \\ \beta \theta^n \theta^s & \alpha_s \partial_x (\theta^s \partial_x) - \beta \theta^n \theta^s & -\theta^s \partial_x \\ -\partial_x \theta^n & -\partial_x \theta^s & 0 \end{bmatrix}}_{\mathcal{A}} \begin{bmatrix} u^n \\ v^n \\ p \end{bmatrix} = \begin{bmatrix} \partial_x (\Psi(\theta^n)) \\ 0 \\ 0 \end{bmatrix}.$$

The negative sign in the volume-averaged incompressibility constraint is included to impose a useful restriction on the eigenvalues of the discrete approximation to the operator $\mathcal{A}$ as discussed in section 3. No-slip boundary conditions are used for (2.15), while no-flux conditions are used for (2.11). Finally, some initial network volume fraction θ^n is provided.

Note that the above system could be reduced quite significantly by analytically integrating (2.14) to obtain a relationship for u^n in terms of u^s. Indeed, this is exactly what is done in [3, 9]. However, since we are ultimately interested in developing computational techniques for solving the model in two (and three) dimensions for which this analytical reduction would not be possible (see section 7.1), we focus on the full system (2.15).

2.2. An example. Before going into detail on the computational techniques for solving (2.11)–(2.14), we first give an example illustrating the interesting pattern formations that can occur with this model. This also allows us to illuminate some potential computational issues that arise. Note that the simulation results that follow are generated using our proposed method.

We start with an initial distribution of network θ^n that is perturbed about the unstable region of the osmotic pressure term Ψ (cf. Figure 1):

$$(2.16) \qquad \theta^n(x) = 0.05(1 + 0.01 \cos(2\pi cx)),$$

where $c = 3$. We set the network viscosity $\alpha_{\mathrm{n}} = 0.1$, the solvent viscosity $\alpha_{\mathrm{s}} = 0.01$, the frictional coupling parameter $\beta = 1$, and the diffusion coefficient $\kappa = 10^{-7}$. Figure 2 displays snapshots of the network volume fraction and velocity u^{n} at various times corresponding to different stages of phase separation. Clearly as time increases, the network and solvent phases separate, and three channels with sharp edges form.

The system of equations for computing the network/solvent velocity and the pressure contains variable-coefficient Laplacians (see (2.12) and (2.13)), as well as a variable-coefficient divergence equation (see (2.14)). The coefficients in these cases are the volume fractions θ^{n} and θ^{s}, and as we can see from Figure 2, these can vary strongly in some regions. Thus, any algorithm for solving this system must be able to handle variable coefficients effectively.

3. Discretization of the 1-D model. The basic strategy we propose for simulating the 1-D gel model (2.11)–(2.14) is as follows:

1. For a given θ^{n} at time t, solve a discrete analog of (2.15) for u^{n}, u^{s}, and p at time t.
2. Solve for θ^{n} (and thus θ^{s}) at time $t + \Delta t$ using a discrete analog of (2.11) with the value of u^{n} at time t.
3. Repeat step 1, with the θ^{n} at $t + \Delta t$.

3.1. Spatial discretization. We use a MAC grid [17] for the unknowns. Values of the network and solvent velocities u^{n} and u^{s} reside at the cell-edges $\{x_{i+\frac{1}{2}}\}$, while values for the network and solvent volume fractions θ^{n}, θ^{s}, and the pressure p reside at cell-centers $\{x_i\}$; see Figure 3. Here $x_{i+\frac{1}{2}} = ih$, $i = 0, \ldots, N$, and $x_i = (i - \frac{1}{2})h$, $i = 1, \ldots, N$, where $Nh = 1$. All equations in (2.15) are discretized using second-order, centered finite differences. The MAC grid is used for the following reasons:

- it allows for a natural approximation to the variable-coefficient Laplacians appearing in (2.12) and (2.13);
- it requires no artificial boundary condition on the pressure;
- it is favorable for volume-conserving time integration.

However, since θ^{n} and θ^{s} appear in the frictional coupling term, the gradient of the pressure, and the incompressibility-type constraint in (2.12)–(2.14), we need the values of θ^{n} and θ^{s} at cell-edges. To obtain these we use simple two-point arithmetic averaging.

The respective approximations for (2.12) and (2.13) at $x = x_{i+\frac{1}{2}}$, $i = 1, 2, \ldots N - 1$, are given by

$$
(3.1) \quad
\begin{aligned}
\frac{\alpha_{\mathrm{n}}}{h^2} [\theta^{\mathrm{n}}_{i+1}(u^{\mathrm{n}}_{i+\frac{3}{2}} - u^{\mathrm{n}}_{i+\frac{1}{2}}) - \theta^{\mathrm{n}}_i(u^{\mathrm{n}}_{i+\frac{1}{2}} - u^{\mathrm{n}}_{i-\frac{1}{2}})] - \beta \overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2}} \overline{\theta}^{\mathrm{s}}_{i+\frac{1}{2}} [u^{\mathrm{n}}_{i+\frac{1}{2}} - u^{\mathrm{s}}_{i+\frac{1}{2}}] \\
- \frac{\overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2}}}{h}[p_{i+1} - p_i] = \frac{\Psi(\theta^{\mathrm{n}}_{i+1}) - \Psi(\theta^{\mathrm{n}}_i)}{h},
\end{aligned}
$$

$$
(3.2) \quad
\begin{aligned}
\frac{\alpha_{\mathrm{s}}}{h^2} [\theta^{\mathrm{s}}_{i+1}(u^{\mathrm{s}}_{i+\frac{3}{2}} - u^{\mathrm{s}}_{i+\frac{1}{2}}) - \theta^{\mathrm{s}}_i(u^{\mathrm{s}}_{i+\frac{1}{2}} - u^{\mathrm{s}}_{i-\frac{1}{2}})] + \beta \overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2}} \overline{\theta}^{\mathrm{s}}_{i+\frac{1}{2}} [u^{\mathrm{n}}_{i+\frac{1}{2}} - u^{\mathrm{s}}_{i+\frac{1}{2}}] \\
- \frac{\overline{\theta}^{\mathrm{s}}_{i+\frac{1}{2}}}{h}[p_{i+1} - p_i] = 0,
\end{aligned}
$$

where

$$
\overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2}} = \frac{\theta^{\mathrm{n}}_i + \theta^{\mathrm{n}}_{i+1}}{2} \quad \text{and} \quad \overline{\theta}^{\mathrm{s}}_{i+\frac{1}{2}} = 1 - \overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2}}.
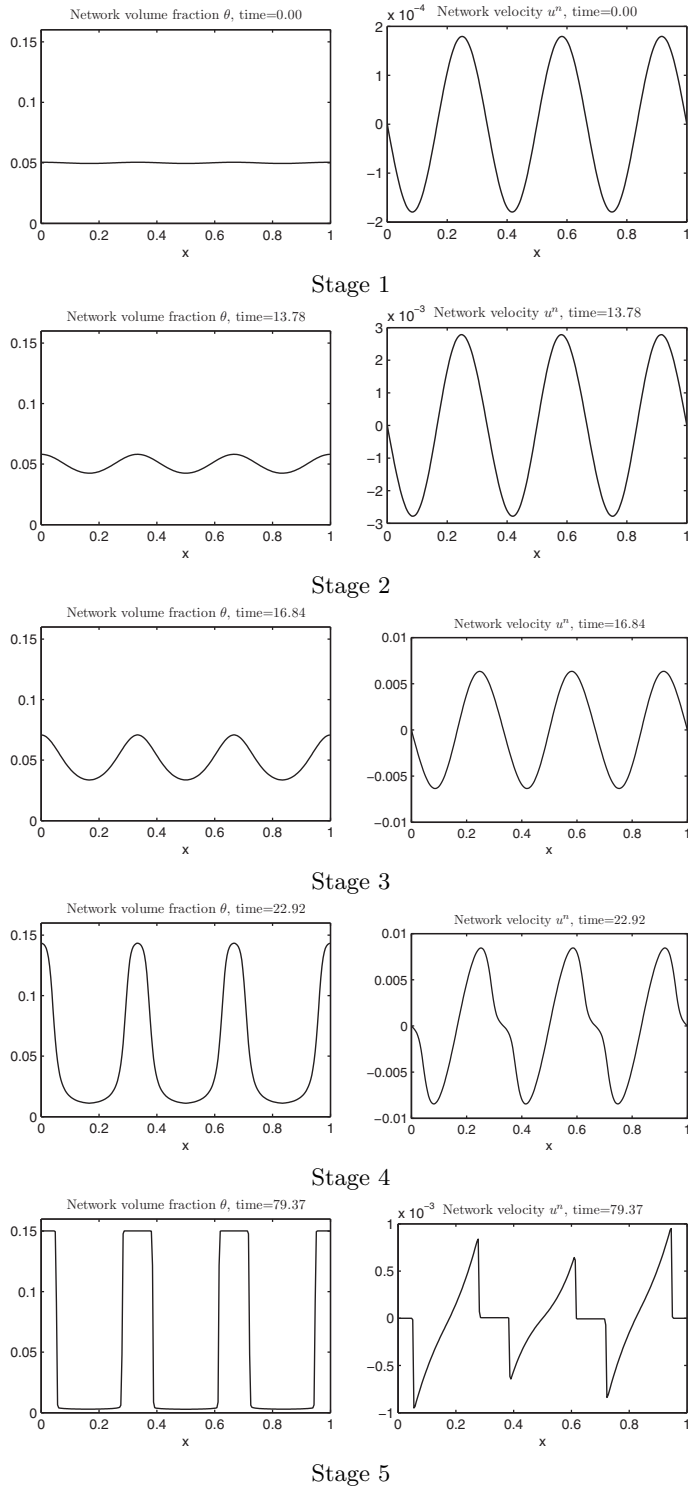$$

Stage 1

Stage 2

Stage 3

Stage 4

Stage 5

FIG. 2. *Network volume fraction (left column) and network velocity (right column) from a numerical simulation of the 1-D model problem (2.11) and (2.15) at various stages of gel phase separation. Note the fixed scale on the θ^n plots and the variable scale on the u^n plots.*
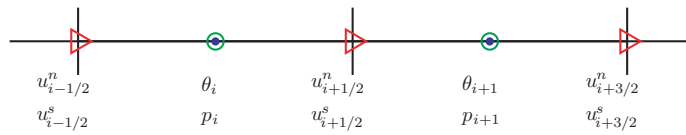
FIG. 3. *The location of the unknowns in the MAC grid for the 1-D gel model. θ here represents both the network and solvent volume fractions.*
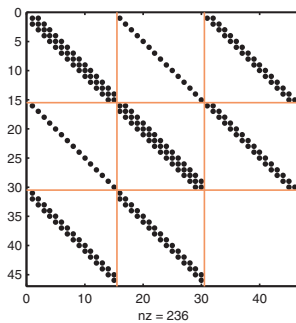

FIG. 4. *The structure of the $\mathcal{A}^h$ matrix in (3.4) for the case of $N = 2^4$ grid cells; solid dots mark nonzero entries in the matrix, while solid horizontal and vertical lines highlight the structure shown in (3.4).*

Note that the no-slip boundary conditions give $u_{1/2}^{\mathrm{n}} = u_{N+1/2}^{\mathrm{n}} = u_{1/2}^{\mathrm{s}} = u_{N+1/2}^{\mathrm{s}} = 0$. Similarly, (2.14) is approximated at x_i, $i = 1, \dots, N$, by

$$(3.3) \qquad -\frac{\overline{\theta}_{i+\frac{1}{2}}^{\mathrm{n}}}{h}u_{i+\frac{1}{2}}^{\mathrm{n}} + \frac{\overline{\theta}_{i-\frac{1}{2}}^{\mathrm{n}}}{h}u_{i-\frac{1}{2}}^{\mathrm{n}} - \frac{\overline{\theta}_{i+\frac{1}{2}}^{\mathrm{s}}}{h}u_{i+\frac{1}{2}}^{\mathrm{s}} + \frac{\overline{\theta}_{i-\frac{1}{2}}^{\mathrm{s}}}{h}u_{i-\frac{1}{2}}^{\mathrm{s}} = 0.$$

We can represent the $(3N - 2)$-by-$(3N - 2)$ system of equations (3.1)–(3.3) in matrix-vector form as

$$(3.4) \qquad \underbrace{\begin{bmatrix} \mathcal{L}_{\mathrm{n}}^h - \mathcal{C}^h & \mathcal{C}^h & -\mathcal{G}_{\mathrm{n}}^h \\ \mathcal{C}^h & \mathcal{L}_{\mathrm{s}}^h - \mathcal{C}^h & -\mathcal{G}_{\mathrm{s}}^h \\ \mathcal{G}_{\mathrm{n}}^{hT} & \mathcal{G}_{\mathrm{s}}^{hT} & 0 \end{bmatrix}}_{\mathcal{A}^h} \begin{bmatrix} u^{\mathrm{n}} \\ u^{\mathrm{s}} \\ p \end{bmatrix} = \begin{bmatrix} \partial_x^h(\Psi(\theta^{\mathrm{n}})) \\ 0 \\ 0 \end{bmatrix},$$

which forms our discrete approximation of (2.15); see Figure 4 for an illustration of the structure of $\mathcal{A}^h$. Since $\mathcal{G}_{\mathrm{n}}^h$ and $\mathcal{G}_{\mathrm{s}}^h$ annihilate constant vectors, $\mathcal{A}^h$ has at least one zero eigenvalue. More can be said about the spectrum of this saddle point system by rewriting $\mathcal{A}^h$ as

$$(3.5) \qquad \mathcal{A}^h = \begin{bmatrix} A & B \\ -B^T & 0 \end{bmatrix}.$$

Since we assume $0 < \theta^{\mathrm{n}} < 1$, A is symmetric and negative definite, and B has rank $N - 1$. It follows, for example, from [4, Thm. 3.6] that the eigenvalues of $\mathcal{A}^h$ have nonpositive real part ($\mathcal{A}^h$ is negative semistable), i.e., $\mathrm{Re}(\lambda) \leq 0$ for all $\lambda \in \sigma(\mathcal{A}^h)$. This can be advantageous for (preconditioned) Krylov subspace methods [4], which are the basis for the method we develop to solve (3.4).

3.2. Temporal discretization. To advance the system in time according to (2.11), we use explicit first-order upwinding for the flux $(\theta^n u^n)_x$ and treat the diffusion implicitly with backward Euler. The solvent volume fraction is updated according to the relationship $\theta^s = 1 - \theta^n$. We use adaptive time stepping for efficiency and stability as discussed next.

4. Stable time stepping. We postpone the discussion of a method for solving (3.4) for u^n, v^n, and p until the next section and focus here on analyzing the stability of the upwind time stepping scheme discussed in the previous section.

For a given u^n and $\kappa = 0$, (2.11) looks like a linear, variable-coefficient advection equation. The CFL condition for upwinding this equation would thus be

$$\Delta t \leq \frac{h}{\max\limits_{1 \leq i \leq N-1} |u^n_{i+1/2}|}.$$

From numerical experiments, we found that this restriction on Δt appears to be sufficient for stability for almost all stages of phase separation. However, as the gel begins to reach a steady state and the magnitude of the network velocity decreases the CFL condition allows for too large of a time step to maintain stability. The reason the CFL condition is not sufficient for stability during the whole simulation is that (2.11) really depends nonlinearly on u^n. To better understand how Δt should scale for stability we do a linear stability analysis on the entire coupled system.

4.1. Linear stability analysis. A spatially uniform network volume fraction and zero network and solvent velocities are a solution to (3.1)–(3.3). We perturb about this uniform state and assume

$$\theta^n(x,t) = \theta^{n,0} + \epsilon \theta^{n,1}(x,t) + O(\epsilon^2),$$
$$u^n(x,t) = 0 + \epsilon u^{n,1}(x,t) + O(\epsilon^2),$$
$$u^s(x,t) = 0 + \epsilon u^{s,1}(x,t) + O(\epsilon^2),$$
$$p(x,t) = p^0 + \epsilon p^1(x,t) + O(\epsilon^2),$$

where $0 < \theta^{n,0} < 1$ and $0 \leq x \leq 1$; also recall that $\theta^s = 1 - \theta^n$. Since p is unique up to a constant, we set $p^0 = 0$.

Plugging the above expansions into (2.12)–(2.14) and looking at the $O(\epsilon)$ terms gives the linearized equations

$$(4.1) \qquad \alpha_n \theta^{n,0} \partial_{xx}(u^{n,1}) - \beta \theta^{n,0} \theta^{s,0}(u^{n,1} - u^{s,1}) - \theta^{n,0} \partial_x(p^1) = \Psi'(\theta^{n,0}) \partial_x(\theta^{n,1}),$$

$$(4.2) \qquad \alpha_s \theta^{s,0} \partial_{xx}(u^{s,1}) + \beta \theta^{n,0} \theta^{s,0}(u^{n,1} - u^{s,1}) - \theta^{s,0} \partial_x(p^1) = 0,$$

$$(4.3) \qquad\qquad\qquad -\theta^{n,0} \partial_x(u^{n,1}) - \theta^{s,0} \partial_x(u^{s,1}) = 0.$$

We discretize these equations using finite differences on a MAC grid as discussed in the previous section, and let ∂_x^h and ∂_{xx}^h denote the second-order accurate, centered approximation to ∂_x and ∂_{xx}.

We assume an $O(\epsilon)$ perturbation to the θ^n initial condition of form $\theta^{n,1} = \cos(\omega \pi x)$, where ω is an integer, and we assume that $\Psi'(\theta^{n,0}) > 0$, so that the gel is stable. To solve the discrete form of the equations (4.1)–(4.3) for this $\theta^{n,1}$, we first exploit (4.3) to get the following analytical expression relating $u^{s,1}$ to $u^{n,1}$:

$$(4.4) \qquad\qquad\qquad u^{s,1} = -\frac{\theta^{n,0}}{\theta^{s,0}} u^{n,1}.$$

This allows us to use the discrete forms of the two equations (4.1) and (4.2) to determine $u^{n,1}$ and p^1. Second, we plug $\theta^{n,1}$ in the right-hand side of the discrete approximation to (4.1) to obtain

$$\Psi'(\theta^{n,0})\partial_x^h\left(\theta^{n,1}\right)_{x=x_{i+1/2}} = -\frac{2\Psi'(\theta^{n,0})}{h}\sin\left(\frac{\omega\pi h}{2}\right)[\sin(\omega\pi x)]_{x=x_{i+1/2}}.$$

Third, we note that the discrete operator ∂_{xx}^h satisfies

$$\left[\partial_{xx}^h(\sin(\omega\pi x))\right]_{x=x_{i+1/2}} = -\frac{4}{h^2}\sin^2\left(\frac{\omega\pi h}{2}\right)[\sin(\omega\pi x)]_{x=x_{i+1/2}}.$$

These three results suggest solutions of the form $u^{n,1} = B_1\sin(\omega\pi x)$ and $p^1 = B_2\cos(\omega\pi x)$ for some constants B_1 and B_2. To determine these constants, and thus $u^{n,1}$ and p^1, we plug the suggested solutions for $u^{n,1}$ and p^1 into the discrete forms of (4.1) and (4.2) and solve for B_1 and B_2. We only present the final result for $u^{n,1}$, since this is all we need to know for the stability estimate:

$$(4.5) \qquad u^{n,1} = \frac{2\sin\left(\frac{\omega\pi h}{2}\right)\Psi'(\theta^{n,0})}{h\zeta}\sin(\omega\pi x),$$

where

$$\zeta = \frac{\theta^{n,0}}{\theta^{s,0}}\left[\beta + \frac{4}{h^2}\sin^2\left(\frac{\omega\pi h}{2}\right)(\alpha_n\theta^{s,0} + \alpha_s\theta^{n,0})\right].$$

Now that we have computed $u^{n,1}$, we can determine how θ^n is updated in time. Since there are no $O(1)$ terms in the expansions for u^n, u^s, and p, the $O(\epsilon)$ term in the expansion for θ^n is updated according to the linearized equation

$$\partial_t(\theta^{n,1}) + \theta^{n,0}\partial_x(u^{n,1}) = \kappa\partial_{xx}(\theta^{n,1}).$$

Discretizing this equation according to the method discussed in section 4 yields

$$(4.6)$$
$$\frac{\theta^{n,1}(x_i, t+\Delta t_\ell) - \theta^{n,1}(x_i, t)}{\Delta t_\ell} + \theta^{n,0}\left[\partial_x^h(u^{n,1})\right]_{x=x_i} = \kappa\left[\partial_{xx}^h(\theta^{n,1}(x_i, t+\Delta t_\ell))\right]_{x=x_i}.$$

Using the $O(\epsilon)$ perturbation $\theta^{n,1} = \cos(\omega\pi x)$ as the initial condition for θ^n and the corresponding $O(\epsilon)$ solution $u^{n,1}$ from (4.5) in (4.6), we obtain the tridiagonal system

$$-\frac{\kappa\Delta t_\ell}{h^2}\theta^{n,1}(x_{i-1}, t+\Delta t_\ell) + \left(1 + \frac{2\kappa\Delta t_\ell}{h^2}\right)\theta^{n,1}(x_i, t+\Delta t_\ell) - \frac{\kappa\Delta t_\ell}{h^2}\theta^{n,1}(x_{i+1}, t+\Delta t_\ell) =$$
$$\left[1 - \Delta t_\ell\frac{4\sin^2\left(\frac{\omega\pi h}{2}\right)}{h^2}\frac{\Psi'(\theta^{n,0})\theta^{n,0}}{\zeta}\right]\cos(\omega\pi x_i),$$

for which the solution is

$$\theta^{n,1}(x_i, t+\Delta t_\ell) = \frac{1}{1 + \frac{4\kappa\Delta t_\ell}{h^2}}\left[1 - \Delta t_\ell\frac{\Psi'(\theta^{n,0})\theta^{s,0}}{\alpha_n\theta^{s,0} + \alpha_s\theta^{n,0} + \frac{h^2\beta}{4\sin^2\left(\frac{\omega\pi h}{2}\right)}}\right]\cos(\omega\pi x_i)$$
$$= A_\omega\cos(\omega\pi x_i).$$

The linearized scheme is stable if $|A_\omega| < 1$. To bound Δt_ℓ so that this restriction on A_ω is satisfied, we assume that $\omega h \approx 1$ (i.e., the frequency of the perturbation is high) so that $\sin^2\left(\frac{\omega \pi h}{2}\right) \approx 1$. Also, since we are interested in very small diffusion coefficients, we set $\kappa = 0$. These assumptions lead to the time step bound

$$0 < \Delta t_\ell < \frac{2\left[\alpha_n \theta^{s,0} + \alpha_s \theta^{n,0} + \frac{h^2 \beta}{4}\right]}{\Psi'(\theta^{n,0})\theta^{s,0}},$$

which is sufficient for the stability of the linearized scheme.

4.2. Extension to the nonlinear equations. For the full nonlinear system of equations (2.11)–(2.14), we assume that the perturbation is about some initial state at each discretization point and that the above linear analysis applies locally. This gives an adaptive time stepping restriction as follows:

$$(4.7) \qquad 0 < \Delta t_\ell < \min_{1 \le i \le N}\left[\frac{2\left[\alpha_n \theta_i^s + \alpha_s \theta_i^n + \frac{h^2 \beta}{4}\right]}{|\Psi'(\theta_i^n)|\theta_i^s}\right],$$

where θ_i^n and θ_i^s indicate the respective approximate values of θ^n and θ^s at the ith cell center and current time (i.e., $\theta_i^n \approx \theta^n(x_i, t_j)$, $1 \le i \le N$, $j \ge 0$).

The absolute value sign is used around $\Psi'(\theta_i^n)$ in the above condition, since we are now allowing the phases to separate (i.e., $\Psi'(\theta_i^n) < 0$). In regions where $\Psi'(\theta^n)$ is negative, the gel is unstable, and certain modes will grow there. Thus, the stability of our numerical scheme is not a concern in these regions. What we have to be concerned about is that our scheme does not artificially excite any modes in regions where $\Psi'(\theta^n) \ge 0$. By using the magnitude of $\Psi'(\theta^n)$ in (4.7) we are thus making the bound more restrictive than is necessary.

For the nonlinear case, the restriction (4.7) may still not be sufficient, since it may allow a time step large enough to violate the CFL condition. To prevent this problem, we choose the time step according to the rule

$$(4.8) \qquad \Delta t = \min\left\{\eta_\ell \min_{1 \le i \le N}\left[\frac{2\left[\alpha_n \theta_i^s + \alpha_s \theta_i^n + \frac{h^2 \beta}{4}\right]}{|\Psi'(\theta_i^n)|\theta_i^s}\right], \eta_c \min_{1 \le i \le N-1}\left[\frac{h}{|u_{i+1/2}^n|}\right]\right\},$$

where $u_{i+1/2}^n$ is the approximate network velocity at the right edge of the ith cell center and $0 < \eta_\ell, \eta_c < 1$. While the linear stability analysis allows for $\eta_\ell = \eta_c = 1$, for the full nonlinear case, we find in practice that choosing these values slightly less than 1 results in a robust adaptive scheme. For example, in all experiments that follow, as well as the one from Figure 2, we use $\eta_\ell = \eta_c = 0.98$.

5. Solving the coupled momentum and incompressibility equations. As mentioned in the introduction, the coupled momentum and incompressibility equations (2.15) are similar to Stokes equations. However, the coupling of the two fluids (given by the $\beta \theta^n \theta^s$ term), the time-dependent strongly varying variable-coefficient Laplacian terms, and the volume-averaged incompressibility equation make numerically solving the equations more complex. In this section, we introduce a particularly effective method for solving the discrete form of this system (3.4) that is based, in part, on a multigrid technique first proposed by Vanka [36] for Stokes and Navier–Stokes. We first discuss the extension of this technique (which we call box relaxation [35, p. 320]) to the gel system. We follow this with a discussion of how to make it more efficient and robust by using it as a preconditioner for the GMRES method.
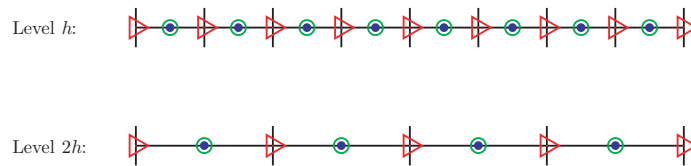
FIG. 5. *An example of coarsening an $N = 2^3$ staggered grid to an $M = 2^2$ grid. Network and solvent velocities u^n and u^s reside at cell-edges, while pressure p and volume fractions θ^n, θ^s reside at the cell-centers.*

5.1. Multigrid box relaxation. Any geometric multigrid method is characterized by the following five components: the grid coarsening, the transfer operators (restriction and prolongation) for the unknowns, the coarse grid discretization, the smoother, and the grid cycling. For our proposed method, the smoother component is the only one that is not standard, so we discuss this in detail and only briefly discuss the other components. For a more detailed description of multigrid in general, we refer the reader to the many excellent books on the subject (e.g., [6, 35, 40]).

Grid coarsening and transfer operators. For a given value of $N = 2^k$, a sequence of coarser grids Ω_N, $\Omega_{N-1}, \ldots, \Omega_M$ ($N > M$) is defined where each grid is a factor of two coarser than the previous; see Figure 5 for an illustration. Since we are using a staggered discretization, the transfer operators for the defects of the two velocity components u^n and u^s will be different from that for the defect for the pressure p. For the velocity components we use three-point, full-weighting for restriction, while for the pressure we use two-point arithmetic averaging [35, p. 69]. For prolongation, we use linear interpolation for the two velocities and the pressure [35, p. 70]. Because θ^n and θ^s are variable coefficients, it is also necessary to transfer these components from the fine to the coarse grid, but not the reverse since these are not unknowns of the system. For this transfer we use two-point arithmetic averaging.

Coarse grid discretization. We use the simple DCGA (discretization coarse grid approximation) strategy, where the equations are directly discretized on the coarser grids.

Smoother. Since there is a zero diagonal block in the $(3,3)$ entry of the matrix in (3.4), standard decoupled relaxation schemes (e.g., Jacobi or Gauss–Siedel) cannot be directly applied. Instead, following the work of Vanka [36] we use a collective (box) relaxation scheme. The basic idea of box relaxation is to compute the network velocity u^n, solvent velocity u^s, and pressure p (or, more accurately, corrections to these values) by collectively solving the discrete gel equations (3.4) locally computational cell (or box) by cell (see Figure 3). Thus for each box, a 5-by-5 system of equations must be solved.

Letting the superscript k denote the iteration of the relaxation scheme, the linear system for solving for the five unknowns in box i ($2 \leq i \leq N-1$) at the $k+1$ iteration is given by

$$(5.1) \quad \begin{bmatrix} L_n^h - C^h & C^h & -G_n^h \\ C^h & L_s^h - C^h & -G_s^h \\ G_n^{h\,T} & G_s^{h\,T} & 0 \end{bmatrix} \begin{bmatrix} (\mathbf{u}_i^n)^{k+1} \\ (\mathbf{u}_i^s)^{k+1} \\ p_i^{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_{\mathbf{u}_i^n} \\ \mathbf{b}_{\mathbf{u}_i^s} \\ b_{p_i} \end{bmatrix},$$

where

$$(\mathbf{u}_i^n)^{k+1} = \begin{bmatrix} (u_{i-1/2}^n)^{k+1} \\ (u_{i+1/2}^n)^{k+1} \end{bmatrix} \text{ and } (\mathbf{u}_i^s)^{k+1} = \begin{bmatrix} (u_{i-1/2}^s)^{k+1} \\ (u_{i+1/2}^s)^{k+1} \end{bmatrix}$$

are the unknown network and solvent velocities at the edges of box i, and

$$L_{\mathrm{n}}^h = \frac{\alpha_{\mathrm{n}}}{h^2} \begin{bmatrix} -(\theta_i^{\mathrm{n}} + \theta_{i-1}^{\mathrm{n}}) & \theta_i^{\mathrm{n}} \\ \theta_i^{\mathrm{n}} & -(\theta_{i+1}^{\mathrm{n}} + \theta_i^{\mathrm{n}}) \end{bmatrix}, \ L_{\mathrm{s}}^h = \frac{\alpha_{\mathrm{s}}}{h^2} \begin{bmatrix} -(\theta_i^{\mathrm{s}} + \theta_{i-1}^{\mathrm{s}}) & \theta_i^{\mathrm{s}} \\ \theta_i^{\mathrm{s}} & -(\theta_{i+1}^{\mathrm{s}} + \theta_i^{\mathrm{s}}) \end{bmatrix},$$

$$C^h = \frac{\beta}{2} \begin{bmatrix} (\theta_i^{\mathrm{n}} \theta_i^{\mathrm{s}} + \theta_{i-1}^{\mathrm{n}} \theta_{i-1}^{\mathrm{s}}) & 0 \\ 0 & (\theta_{i+1}^{\mathrm{n}} \theta_{i+1}^{\mathrm{s}} + \theta_i^{\mathrm{n}} \theta_i^{\mathrm{s}}) \end{bmatrix},$$

$$G_{\mathrm{n}}^h = \frac{1}{2h} \begin{bmatrix} \theta_i^{\mathrm{n}} + \theta_{i-1}^{\mathrm{n}} \\ -(\theta_{i+1}^{\mathrm{n}} + \theta_i^{\mathrm{n}}) \end{bmatrix}, \ G_{\mathrm{s}}^h = \frac{1}{2h} \begin{bmatrix} \theta_i^{\mathrm{s}} + \theta_{i-1}^{\mathrm{s}} \\ -(\theta_{i+1}^{\mathrm{s}} + \theta_i^{\mathrm{s}}) \end{bmatrix},$$

$$\mathbf{b}_{\mathbf{u}_i^{\mathrm{n}}} = \begin{bmatrix} \frac{\Psi(\theta_i^{\mathrm{n}}) - \Psi(\theta_{i-1}^{\mathrm{n}})}{h} - \frac{\alpha_{\mathrm{n}}}{h^2} \theta_{i-1}^{\mathrm{n}} (u_{i-3/2}^{\mathrm{n}})^{k+1} - \frac{\theta_i^{\mathrm{n}} + \theta_{i-1}^{\mathrm{n}}}{2h} p_{i-1}^{k+1} \\ \frac{\Psi(\theta_{i+1}^{\mathrm{n}}) - \Psi(\theta_i^{\mathrm{n}})}{h} - \frac{\alpha_{\mathrm{n}}}{h^2} \theta_{i+1}^{\mathrm{n}} (u_{i+3/2}^{\mathrm{n}})^k + \frac{\theta_{i+1}^{\mathrm{n}} + \theta_i^{\mathrm{n}}}{2h} p_{i+1}^k \end{bmatrix},$$

$$\mathbf{b}_{\mathbf{u}_i^{\mathrm{s}}} = \begin{bmatrix} -\frac{\alpha_{\mathrm{s}}}{h^2} \theta_{i-1}^{\mathrm{s}} (u_{i-3/2}^{\mathrm{s}})^{k+1} - \frac{\theta_i^{\mathrm{s}} + \theta_{i-1}^{\mathrm{s}}}{2h} p_{i-1}^{k+1} \\ -\frac{\alpha_{\mathrm{s}}}{h^2} \theta_{i+1}^{\mathrm{s}} (u_{i+3/2}^{\mathrm{s}})^k + \frac{\theta_{i+1}^{\mathrm{s}} + \theta_i^{\mathrm{s}}}{2h} p_{i+1}^k \end{bmatrix}, \ b_{p_i} = 0.$$

Note that, for $i = 1$ and $i = N$, only three equations need to be solved because of the no-slip boundary conditions. The above procedure updates the unknowns box-by-box in a Gauss–Seidel-type manner. For all of the numerical results, we process the blocks using red-black ordering.

In the context of Stokes and Navier–Stokes equations, it is sometimes necessary to combine the box scheme with underrelaxation to achieve better smoothing properties (cf. [35, p. 321]). We also find this to be the case for the gel system, for which the relaxation scheme takes the form

$$(5.2) \quad \begin{bmatrix} (\mathbf{u}_i^{\mathrm{n}})^{k+1} \\ (\mathbf{u}_i^{\mathrm{s}})^{k+1} \\ p_i^{k+1} \end{bmatrix} = (1 - \omega) \begin{bmatrix} (\mathbf{u}_i^{\mathrm{n}})^k \\ (\mathbf{u}_i^{\mathrm{s}})^k \\ p_i^k \end{bmatrix} + \omega \begin{bmatrix} L_{\mathrm{n}}^h - C^h & C^h & -G_{\mathrm{n}}^h \\ C^h & L_{\mathrm{s}}^h - C^h & -G_{\mathrm{s}}^h \\ G_{\mathrm{n}}^{hT} & G_{\mathrm{s}}^{hT} & 0 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{b}_{\mathbf{u}_i^{\mathrm{n}}} \\ \mathbf{b}_{\mathbf{u}_i^{\mathrm{s}}} \\ b_{p_i} \end{bmatrix},$$

where $0 < \omega < 1$.

Multigrid cycles. We use two different strategies for cycling through the grids: the V- and F-cycles; see Figure 6 for an illustration. The V-cycle is cheaper and simpler, while the F-cycle is more expensive, but also more robust [25]. Letting ν_1
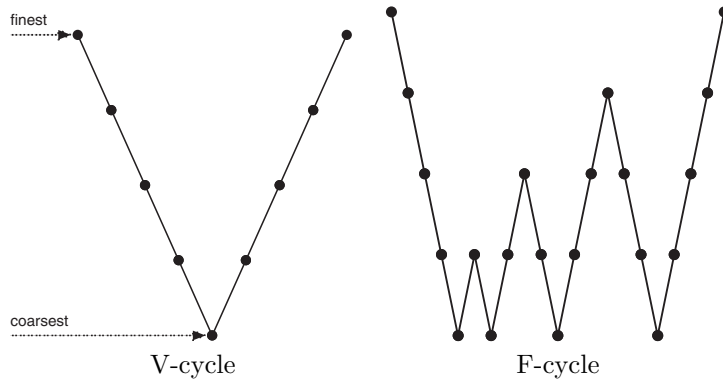

finest

coarsest

V-cycle   F-cycle

FIG. 6. *An illustration of the grid-cycling strategies used in the multigrid method in the case of five grids.*

and ν_2 be the number of pre and postsmoothing operations for each grid, we denote these respective methods as $V(\nu_1, \nu_2)$ and $F(\nu_1, \nu_2)$. In all of our numerical results, we coarsen the grid until there are two cell-centers; i.e. $N = 2$, at which point we solve the system (3.4) directly. In this case, the total computational work for one V-cycle is $W_V = 4CN - 8C$, while for the F-cycle it is $W_F = 8CN - 8C(\log_2(N) + 1)$, for some constant C independent of N. Thus, the computational complexity of both cycles is $O(N)$, with the cost of an F-cycle about twice that of a V-cycle in 1-D.

The above multigrid procedure has the benefit of being relatively straightforward to implement. However, as shown in the numerical experiments in the next section, it may converge slowly or even fail to converge when using it as a standalone solver for the gel system (3.4), especially as phase separation increases; i.e., as the variation in the variable coefficient increases. More sophisticated multigrid components like operator-dependent interpolation/restriction [35, p. 272], Galerkin coarse-grid correction [35, p. 273], distributive relaxation [5], and the streamline-diffusion method [16] could potentially be developed to improve the convergence and obtain an $O(N)$ solver. However, a much simpler idea is to use the multigrid method above as a preconditioner for a Krylov subspace method. The idea of combining multigrid with Krylov methods has been successfully used for complex systems of linear and nonlinear equations (e.g., Trottenberg, Oosterlee, and Schüller [35, sect. 7.8], [25, 26, 27, 38]). While h-independent convergence may not be achieved, robustness in the range of systems and parameters is typically gained, and the methods are easier to implement.

5.2. Multigrid box relaxation as a preconditioner for GMRES. We propose using the multigrid procedure described above as a right-preconditioner for GMRES(m) [28] for solving the system (3.4). This Krylov method is applicable to nonsymmetric matrices and nonpositive definite preconditioners, both of which are found for the gel system. The parameter m represents the maximum dimension the Krylov subspace can reach before the algorithm is restarted. All of our results are for $m = 20$. Following the description of GMRES(m) in [29], this method requires one matrix-vector product with $\mathcal{A}^h$ in (3.4) and one preconditioning step per iteration. Upon the termination of the algorithm, or when a restart occurs, one extra preconditioning step is required.

If we let $y^T = [(u^{\mathrm{n}})^T \; (u^{\mathrm{s}})^T \; p^T]$ and f be the right-hand side of (3.4), the preconditioned system of equations to solve with GMRES(m) is given by

$$(5.3) \qquad \mathcal{A}^h \mathcal{M}^h z = f,$$

where $z = (\mathcal{M}^h)^{-1} y$ and $\mathcal{M}^h$ represents an application of the multigrid box relaxation scheme. Letting $y^{(0)}$ be some initial guess to the solution of (3.4), $\widetilde{r}^{(\ell)} = f - \mathcal{A}^h \mathcal{M}^h z^{(\ell)}$, and defining

$$\mathcal{K}_\ell = \mathrm{span}\left(\widetilde{r}^{(0)}, \mathcal{A}^h \mathcal{M}^h \widetilde{r}^{(0)}, \ldots, \left(\mathcal{A}^h \mathcal{M}^h\right)^{\ell-1} \widetilde{r}^{(0)} \right),$$

$\ell \leq m$, GMRES(m) finds a $z^{(\ell)} \in z^{(0)} + \mathcal{K}_\ell$ such that $\|\widetilde{r}^{(\ell)}\|_2$ is minimal. This minimization can alternatively be expressed as

$$(5.4) \quad \|\widetilde{r}^{(\ell)}\|_2 = \min \left\{ \left\| q_\ell \left(\mathcal{A}^h \mathcal{M}^h \right) r^{(0)} \right\|_2 \middle| q_\ell \text{ is a poly. of degree } \ell \text{ and } q_\ell(0) = 1 \right\}.$$

Since $y^{(\ell)} = \mathcal{M}^h z^{(\ell)}$, minimizing $\|\widetilde{r}^{(\ell)}\|_2$ is equivalent to minimizing $\|f - \mathcal{A}^h y^{(\ell)}\|_2$ (i.e., the standard residual); see, for example, [29, p.272].

The spectrum of the preconditioned matrix is key to understanding the convergence behavior of this Krylov method (see, for example, [25]). To calculate the spectrum we need a way to construct the operator $\mathcal{M}^h$. This can be done using the procedure outlined in Trottenberg, Oosterlee, and Schüller [35, p. 280]. Briefly, the idea is to set $f = 0$ with zero boundary conditions and perform one multigrid cycle (V or F, in our case) for each unit vector as an initial approximation to the system (3.4). The resulting solution from the first standard basis vector initial guess becomes the first column of the iteration matrix, the solution for the second standard basis vector becomes the second column, and so forth.

It turns out that knowing the spectrum of $\mathcal{A}^h\mathcal{M}^h$ gives us information about the spectrum of the multigrid procedure as a standalone method. As shown in [29], one iteration of the multigrid procedure can be written in the form

(5.5) $$y^{(m+1)} = (I - (\mathcal{B}^h)^{-1}\mathcal{A}^h)y^{(m)} + (\mathcal{B}^h)^{-1}f,$$

where $\mathcal{B}^h$ is determined by the multigrid components defined above. Applying this once with a zero initial guess gives

$$y^{(1)} = (\mathcal{B}^h)^{-1}f.$$

This is exactly what we do when applying multigrid once as a preconditioner, so $\mathcal{M}^h = (\mathcal{B}^h)^{-1}$. It follows that the multigrid iteration matrix $(I - (\mathcal{B}^h)^{-1}\mathcal{A}^h)$ can also be written $(I - \mathcal{M}^h\mathcal{A}^h)$. Since $\mathcal{M}^h\mathcal{A}^h$ and $\mathcal{A}^h\mathcal{M}^h$ have the same spectra, $\sigma(I - \mathcal{M}^h\mathcal{A}^h) = 1 - \sigma(\mathcal{A}^h\mathcal{M}^h)$.

For "well-behaved" multigrid problems, the spectrum of the multigrid iteration matrix will be clustered around zero, and the spectral radius will be small making the solver very efficient [38]. For more difficult problems, it has been often found that most of the eigenvalues of the multigrid iteration matrix are clustered around zero, with a few outliers which can slow convergence or even cause the method to diverge [25]. For this situation, combining multigrid with GMRES can be very effective. A full analysis on how GMRES(m) eliminates these outliers and the resulting bounds on the residual reduction are given by Oosterlee and Washio [25, 38].

To understand the key idea of their analysis, suppose that $\mathcal{A}^h\mathcal{M}^h$ has isolated eigenvalues $\lambda_1, \ldots, \lambda_j$, $j < \ell$ far from 1 with the remaining eigenvalues contained in a small disk centered at 1. Consider the polynomial

$$\bar{q}_\ell(x) = (1-x)^{\ell-j}\prod_{i=1}^{j}\frac{\lambda_i - x}{\lambda_i},$$

which is in the class of polynomials defined in (5.4). Now applying $\bar{q}_\ell(\mathcal{A}^h\mathcal{M}^h)$ to an eigenvector of $\mathcal{A}^h\mathcal{M}^h$ corresponding to one of the eigenvalues λ_k for $k > j$, we see that the factor $(I - \mathcal{A}^h\mathcal{M}^h)^{\ell-j}$ in $\bar{q}_\ell(\mathcal{A}^h\mathcal{M}^h)$ substantially reduces the magnitude of this eigenvector. On the other hand, applying $(\lambda_i - \mathcal{A}^h\mathcal{M}^h)/\lambda_i$ to an eigenvector that corresponds to an eigenvalue λ_k, with $k \leq j$, causes a similar reduction. Since $\bar{q}_\ell$ bounds the action of the true minimizing polynomial selected by GMRES, this suggests that this is an effective method for operators with spectra of this form. As noted in [25, Rem. 2.1.3], the asymptotic reduction of the residual of the multigrid preconditioned GMRES(m) for a reasonable initial guess is at least as fast as for the multigrid solver with an initial guess for which the residual does not contain any components corresponding to the outlying eigenvalues.

The spectral analysis of $\mathcal{A}^h\mathcal{M}^h$ for our model problem is included in the next section.

6. Numerical results for the 1-D model. In the first two tests, we compare the multigrid box relaxation method as a standalone solver with GMRES(20), preconditioned by this multigrid method, for solving the momentum and incompressibility equations at various stages of gel phase separation. The upwind scheme discussed in section 4 for the transport equation (2.11) is used for all tests. We use the same setup for the model problem as given in section 2.2 and illustrated in Figure 2. We consider the system (3.4) at time t "solved" when the residual of the ℓth iterate of the respective method satisfies

$$(6.1) \qquad \frac{\|r^{(\ell)}\|_2}{\|f\|_2} \leq 10^{-6},$$

where f equals the right-hand side of (3.4). Unless otherwise specified, for all time steps but the initial, we use the previous time step's values for the network and solvent velocities and pressure as initial guesses for the iterative methods. All methods were implemented in MATLAB and run on an AMD Opteron 850 2.4 GHz processor under MATLAB version 7.1 (R14).

The first test is to see how well the momentum and incompressibility equations solvers scale as the grid is refined from $h = 2^{-7}$ to $h = 2^{-11}$ (recall that $N = 1/h$ is the number of cell-centers). Table 6.1 displays the results for the standalone multigrid

TABLE 6.1
A comparison of the multigrid V-cycle method (MGV), multigrid F-cycle (MGF), GMRES(20) preconditioned with MGV, and GMRES(20) preconditioned with MGF for solving the system (3.4) at the various stages of phase separation shown in Figure 2. Integer values are the number of $V(1,1)$- or $F(1,1)$-cycles required to satisfy (6.1) when using values from the previous time step as the initial guess (nonbracket) and using all zeros as the initial guess (bracket). Decimal values in parenthesis are the wall-clock times (in seconds) for the methods using the former of the initial guess methods. DIV means that the method diverged. The underrelaxation was set to $\omega = 0.675$ for all results.

Stage	$h = 2^{-7}$	$h = 2^{-8}$	$h = 2^{-9}$	$h = 2^{-10}$	$h = 2^{-11}$
			MGV		
1	7 [7] (0.074)	9 [9] (0.170)	11 [11] (0.388)	14 [14] (0.945)	18 [18] (2.379)
2	7 [7] (0.074)	8 [9] (0.152)	9 [11] (0.318)	10 [14] (0.678)	12 [18] (1.590)
3	7 [7] (0.074)	7 [9] (0.134)	8 [11] (0.286)	9 [14] (0.609)	11 [18] (1.450)
4	7 [8] (0.074)	10 [13] (0.189)	25 [34] (0.880)	DIV	DIV
5	DIV	DIV	DIV	DIV	DIV
			MGF		
1	3 [3] (0.089)	3 [3] (0.158)	3 [3] (0.289)	2 [2] (0.360)	2 [2] (0.687)
2	3 [3] (0.089)	3 [3] (0.158)	3 [3] (0.289)	3 [3] (0.540)	3 [3] (1.031)
3	3 [3] (0.089)	3 [3] (0.158)	3 [3] (0.289)	3 [3] (0.540)	3 [3] (1.031)
4	3 [3] (0.089)	3 [3] (0.158)	3 [3] (0.289)	3 [3] (0.540)	2 [3] (0.687)
5	3 [5] (0.089)	5 [8] (0.262)	DIV	DIV	DIV
			GMRES-MGV		
1	7 [7] (0.075)	8 [8] (0.153)	9 [9] (0.321)	10 [10] (0.688)	11 [11] (1.477)
2	7 [7] (0.075)	7 [8] (0.135)	7 [9] (0.250)	8 [10] (0.551)	8 [11] (1.065)
3	7 [7] (0.075)	7 [8] (0.135)	7 [9] (0.250)	7 [10] (0.483)	8 [11] (1.065)
4	6 [8] (0.065)	7 [8] (0.135)	7 [9] (0.250)	8 [10] (0.551)	8 [11] (1.065)
5	6 [8] (0.065)	6 [8] (0.114)	7 [9] (0.250)	7 [9] (0.483)	8 [10] (1.065)
			GMRES-MGF		
1	4 [4] (0.119)	4 [4] (0.212)	4 [4] (0.386)	3 [3] (0.542)	3 [3] (1.037)
2	4 [4] (0.119)	4 [4] (0.212)	4 [4] (0.386)	4 [4] (0.724)	4 [4] (1.384)
3	4 [4] (0.119)	4 [4] (0.212)	4 [4] (0.386)	4 [4] (0.724)	4 [4] (1.384)
4	4 [4] (0.119)	4 [4] (0.212)	4 [4] (0.386)	4 [4] (0.724)	3 [4] (1.037)
5	4 [5] (0.119)	4 [6] (0.212)	5 [5] (0.479)	6 [6] (1.082)	6 [6] (2.065)

V(1,1)-cycle (MGV) and F(1,1)-cycle (MGF) methods and the right-preconditioned versions of these methods for GMRES(20) (GMRES-MGV and GMRES-MGF, respectively). Integer values listed in the table are the number of multigrid cycles (V(1,1) or F(1,1)) required to satisfy (6.1) using values from the previous time step as the initial guess (nonbracket) and using all zeros as the initial guess (bracket). Decimal values in parenthesis are the corresponding wall-clock times (in seconds) for the former initial guess method. Since the computational cost of one preconditioning step of GMRES(20) by a V- or F-cycle dwarfs the other costs of the method, we have chosen to list the number of these iterations for the GMRES(20) method.

Looking first at the MGV results, we see that the computational cost is increasing faster than linearly. However, what is more problematic is that the method fails to converge for Stage 5 phase separation, and even for Stage 4 when h is fine enough. The MGF scaling results are much better, showing a linear growth in the computational cost. However, we again see that, for fine enough h, the method failed to converge for Stage 5 phase separation. The results for GMRES-MGV and GMRES-MGF are much more encouraging. Both of these methods converged for all stages of phase separation and their computational cost scales very well. The cost of GMRES-MGV appears to be increasing at a rate slightly larger than linear with a decreasing slope as the phases become increasingly separated, while the cost of GMRES-MGF is increasing linearly for all stages of separation. However, in terms of wall-clock time our implementation of GMRES-MGV is more efficient than GMRES-MGF for almost all h and stages of phase separation. While the theory from section 5 predicts the F-cycle to be twice the cost of a V-cycle, we see that, in actual wall-clock time, it is about three times the cost in our implementation. Finally, we note that the F-cycle, whether used as a standalone solver or a preconditioner, is much less sensitive to the initial guess. Using the values for the pressure and two velocities from the previous time step as the initial guess instead of all zeros, drops the iterations for the V-cycle but has little effect for the F-cycle.

Figure 7 shows details on the convergence history of the solvers at the various stages of phase separation shown in Figure 2 for the case of $h = 2^{-10}$. Comparing the MGV and GMRES-MGV results for Stages 1–3, we see that the slope of the relative residual is much steeper for the latter. The MGV method diverged for Stages 4–5, so no results are presented. For the MGF and GMRES-MGF methods, the slopes are nearly identical for Stages 1–3 and differ very slightly for Stage 4. No MGF results are presented for Stage 5, since the method diverged in this case.

6.1. Number of channels. In the next set of experiments, we test how well the method scales as the number of channels that form in the gel increases. This is accomplished by changing c to the number of desired channels in the initial perturbation to the gel given by (2.16). The results for $h = 2^{-10}$ are presented in Table 6.2. The stages of phase separation for different c are similar to what is shown in Figure 2, which is for $c = 3$. The table clearly shows that the MGV standalone solver depends quite significantly on the number of channels, while the MGF solver has less of a dependence for the lower stages of phase separation and numbers of channels. However, we see that MGF fails to converge at Stage 4 for 6 or more channels. The GMRES-MGV and GMRES-MGF methods are much less sensitive to the number of channels for Stages 1–4 of phase separation and have a very tolerable increase for Stage 5. It is also interesting that the number of cycles for the GMRES-MGV and GMRES-MGF seem to approach one another for Stage 5 as the number of channels increases.
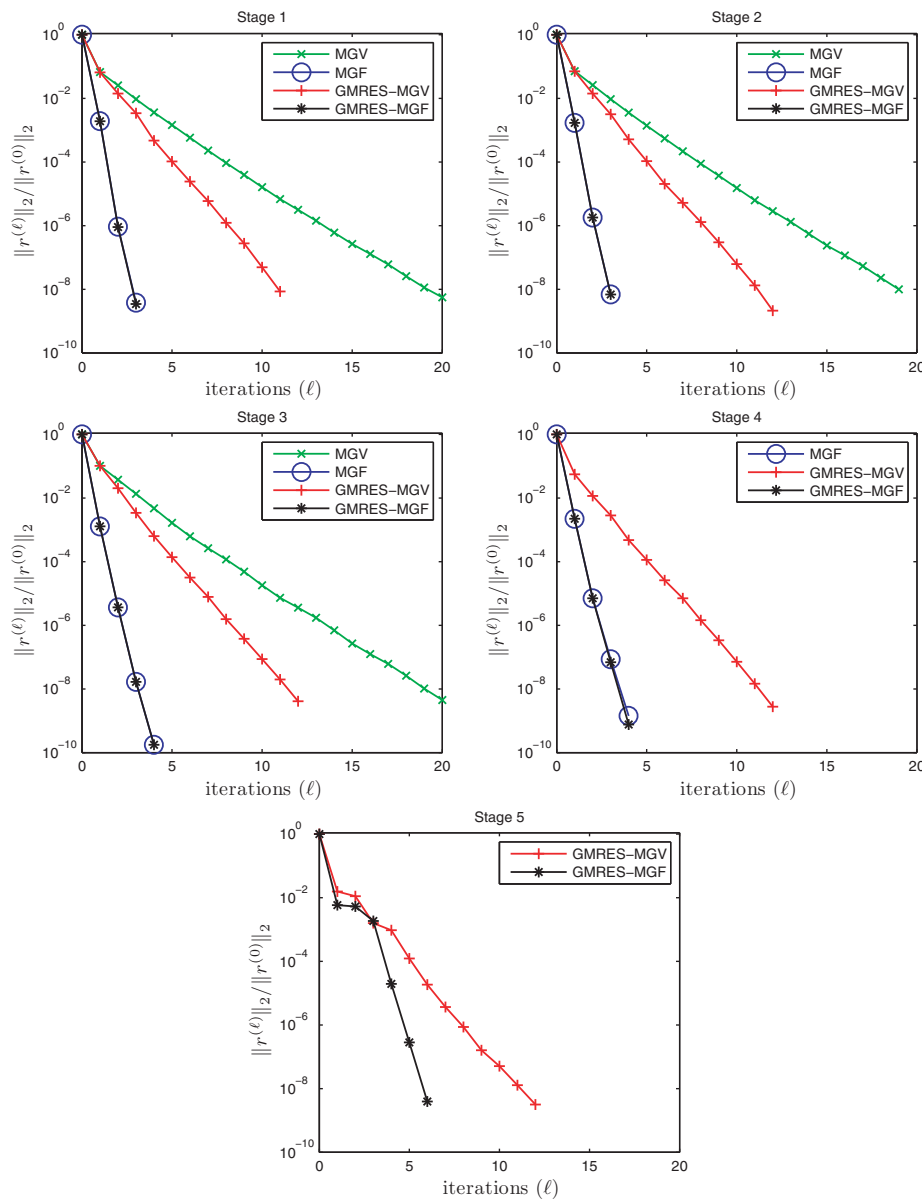
FIG. 7. *The convergence history of the MGV and MGF standalone solvers and the preconditioned GMRES-MGV and GMRES-MGF solvers for* (3.4) *at the various stages of phase separation shown in Figure* 2. *Missing results for MGV and MGF are because these methods diverged for that particular stage. The mesh spacing is given by* $h = 2^{-10}$, *and the underrelaxation was set to* $\omega = 0.675$ *for all results.*

6.2. Spectrum analysis of the preconditioned operator. To get a better understanding of the convergence of the multigrid solver and preconditioner, we return to the three channel example in Figure 2 and analyze the spectrum of the preconditioned operator $\mathcal{A}^h \mathcal{M}^h$ in (5.3). The eigenvalues of $\mathcal{A}^h \mathcal{M}^h$ for the multigrid V- and F-cycle for the various stages of phase separation are shown in Figures 8(a) and (b), respectively. As discussed in section 5.2, the spectrum of the multigrid iteration

TABLE 6.2

A comparison of the four solution methods as the number of channels c that form in the gel is increased. The stages of phase separation are similar to those in Figure 2. Integer values are the number of $V(1,1)$- or $F(1,1)$-cycles required to satisfy (6.1). DIV means the method did not converge. All results are for $h = 2^{-10}$ and an underrelaxation of $\omega = 0.675$.

Stage	$c = 1$	$c = 2$	$c = 3$	$c = 4$	$c = 5$	$c = 6$	$c = 7$
	MGV						
1	17	16	14	14	12	11	11
2	12	12	10	10	9	9	9
3	10	10	9	9	9	9	11
4	8	17	DIV	DIV	DIV	DIV	DIV
5	39	DIV	DIV	DIV	DIV	DIV	DIV
	MGF						
1	2	2	2	3	3	3	3
2	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3
4	2	2	3	3	6	DIV	DIV
5	3	DIV	DIV	DIV	DIV	DIV	DIV
	GMRES-MGV						
1	11	10	10	9	9	9	8
2	8	8	8	7	7	7	7
3	8	7	7	7	7	7	7
4	7	7	8	7	8	8	9
5	6	6	7	8	9	10	10
	GMRES-MGF						
1	3	3	3	4	4	4	4
2	4	4	4	4	4	4	4
3	4	4	4	4	4	4	4
4	3	3	4	4	4	5	5
5	4	5	6	5	8	9	9

matrix in (5.5) is simply $1 - \sigma\left(\mathcal{A}^h \mathcal{M}^h\right)$, so we can also easily obtain this spectrum from Figure 8. The majority of the eigenvalues of $\mathcal{A}^h \mathcal{M}^h$ are clustered very close to $\lambda = 1$ for Stages 1–3 with the V-cycle preconditioner with outliers that are still close to $\lambda = 1$. The corresponding spectrum of the multigrid V-cycle iteration matrix is thus similarly clustered around $\lambda = 0$. These results explain the convergence rates in Table 6.1 and Figure 7 for the V-cycle solver and preconditioner. At Stages 4 and 5, we see that the outlying eigenvalues become increasingly far from $\lambda = 1$. For the V-cycle solver, this means that the spectral radius is increasing, and for Stages 4 and 5 we see that it increases beyond 1, which explains the failure to converge shown in Table 6.1. However, for the V-cycle preconditioner, we see that these outliers have little effect on the convergence of the preconditioned GMRES(20) method, since the majority of the eigenvalues remain clustered at $\lambda = 1$. The spectrum for the F-cycle in Figure 8(b) is even more clustered near $\lambda = 1$ and changes very little for Stages 1–4. This explains the very rapid convergence of this method as both a solver and preconditioner for these stages as shown in Table 6.1 and Figure 7. However, we see that at Stage 5 two outlying eigenvalues have appeared which are a distance more than 1 from $\lambda = 1$; this explains the failure of the F-cycle solver for this stage. Again, however, this has little effect on the preconditioned GMRES(20) method as discussed in section 5.2.

The results displayed so far have shown that for the V-cycle, it is almost always more efficient to use it as a preconditioner than as a solver regardless of gel separation. The same is not true of the F-cycle, since it appears to perform very efficiently as
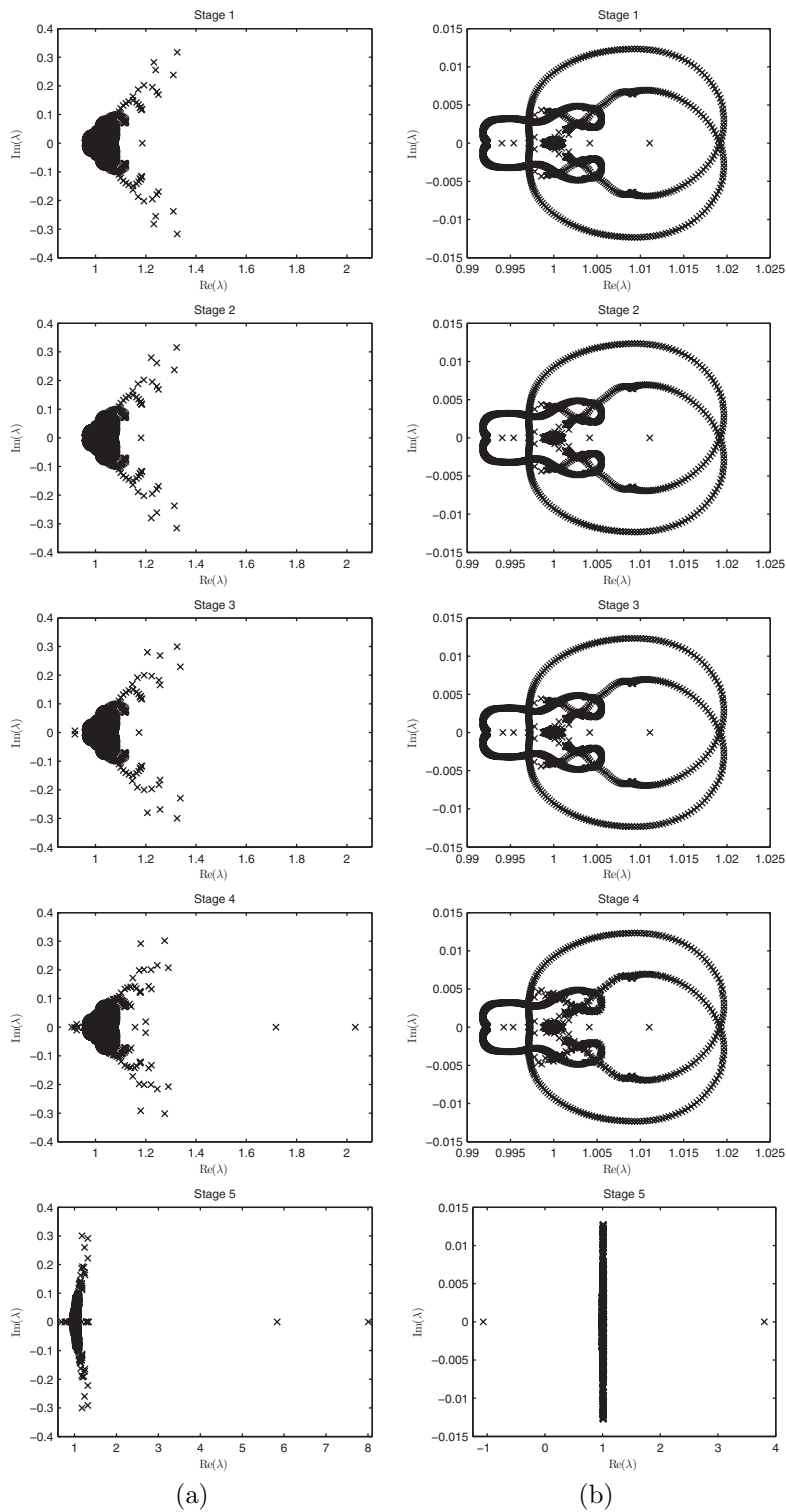
FIG. 8. *The eigenvalues of a multigrid box relaxation preconditioned matrix $\mathcal{A}^h \mathcal{M}^h$ at the five stages of phase separation shown in Figure 2 using the (a) $V(1,1)$ and (b) $F(1,1)$ cycles. Note the different scales on the plots in the (a) and (b) column as well as the different scale for Stage 5. All results are for a mesh spacing of $h = 2^{-10}$ and an underrelaxation of $\omega = 0.675$.*

TABLE 6.3

A comparison of the GMRES-MGV and GMRES-MGF methods for solving (3.4) for different model parameter values and different stages of phase separation. Integer values are the number of $V(1,1)$- or $F(1,1)$-cycles required to satisfy (6.1). The stages of separation are similar to those in Figure 2. All results are for $h = 2^{-10}$ and an underrelaxation of $\omega = 0.675$.

Stage	$\alpha_n = 10^{-1}$			$\alpha_n = 10^{-2}$			$\alpha_n = 10^{-3}$		
	$\alpha_s = 10^{-2}$	$\alpha_s = 10^{-3}$	$\alpha_s = 10^{-4}$	$\alpha_s = 10^{-3}$	$\alpha_s = 10^{-4}$	$\alpha_s = 10^{-5}$	$\alpha_s = 10^{-4}$	$\alpha_s = 10^{-5}$	$\alpha_s = 10^{-6}$
GMRES-MGV, $\beta = 1$									
1	9	8	8	9	9	9	10	9	10
2	7	7	7	9	8	8	10	9	10
3	7	7	7	8	8	8	9	8	10
4	7	7	7	8	8	8	9	9	10
5	6	5	6	6	6	8	7	8	13
GMRES-MGF, $\beta = 1$									
1	4	3	3	4	4	4	4	4	4
2	4	3	3	4	4	4	4	4	5
3	4	3	3	4	4	4	4	4	5
4	4	3	3	4	4	4	5	5	5
5	4	3	3	4	3	4	4	4	7

a standalone solver when the gel mixture is still relatively homogeneous, and the number of channels is not large. For these cases, using it as a preconditioner for GMRES(20) typically takes one more F-cycle iteration due to the overhead of the GMRES method (see section 5.2). However, since we want a robust method for simulation, the remainder of the tests will only include results for the preconditioned V- and F-cycles, which have yet to fail in our experiments. In actual practice, the entire simulation could be made efficient and robust by using a hybrid method, where the V- or F-cycle preconditioners are only used when needed.

6.3. Model parameters. In this next experiment, we test the robustness of the preconditioned solvers as the parameters of the model are changed. We vary the network and solvent viscosity, α_n and α_s, respectively, while holding the frictional coupling parameter β constant at 1. The solvers are tested over five stages of phase separation similar to what is shown in Figure 2 (for which $\beta = 1$, $\alpha_n = 0.1$, and $\alpha_s = 0.01$). The number of channels is set to three and $h = 2^{-10}$. Table 6.3 displays the number of V- and F-cycle iterations for the different parameters. The results indicate the method is quite robust over a wide range of parameters. The only moderate increase in the number of iterations occurs in the last column for $\alpha_n = 10^{-3}$ and $\alpha_s = 10^{-6}$. It is less pronounced in the F-cycle than the V-cycle.

6.4. Underrelaxation parameter. For all previous experiments, the underrelaxation parameter has been fixed at $\omega = 0.675$. So in this final experiment, we test how the number of multigrid cycles for the GMRES-MGV and GMRES-MGF depends on ω. The results are displayed in Figure 9 for the example shown in Figure 2. We see from the figure that for both the GMRES-MGV and GMRES-MGF methods, the optimal ω is nearly the same for all five stages of phase separation. Furthermore, the range of acceptable choices for ω is quite large and is similar between the V- and F-cycles.

7. Extension to two dimensions. In two and higher dimensions the complexity of the gel dynamics model from section 2 increases; however, the extension of the computational method is relatively straightforward. We present here an overview
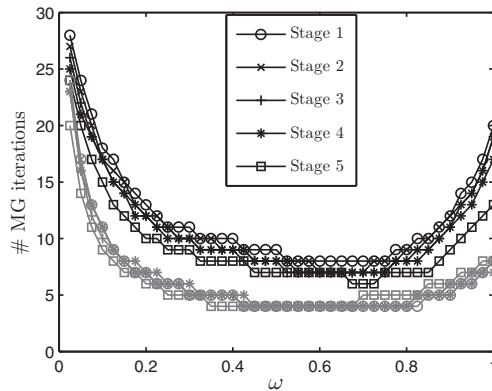
FIG. 9. *A comparison of the number of multigrid cycles needed for the GMRES-MGV (black) and GMRES-MGF (gray) to solve (3.4) as the underrelaxation parameter ω is changed. Symbols represent the different stages of phase separation shown in Figure* 2.

of the two-dimensional model problem, discuss the extension of the computational procedure, and present some numerical results.

7.1. Model problem. Let $\mathbf{u}^n = (u^n, v^n)^T$ and $\mathbf{u}^s = (u^s, v^s)^T$, where u^n, v^n and u^s, v^s are the respective network and solvent velocity components in the x and y directions, and let the spatial domain be $\Omega = \{(x, y) | 0 \le x \le a, 0 \le y \le b\}$. Then the momentum equations (2.6)–(2.7) and volume-averaged incompressibility (2.3) are given in matrix-vector form as

$$(7.1) \qquad \begin{bmatrix} \mathcal{L}_n - \mathcal{C} & \mathcal{C} & -\mathcal{G}_n \\ \mathcal{C} & \mathcal{L}_s - \mathcal{C} & -\mathcal{G}_s \\ -\mathcal{D}_n^T & -\mathcal{D}_s^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u}^n \\ \mathbf{u}^s \\ p \end{bmatrix} = \begin{bmatrix} \nabla \Psi(\theta^n) \\ 0 \\ 0 \end{bmatrix},$$

where

$$\mathcal{L}_{n,s} = \begin{bmatrix} \alpha_{n,s}\partial_x(\theta^{n,s}\partial_x) + \mu_{n,s}\partial_y(\theta^{n,s}\partial_y) & \mu_{n,s}\partial_y(\theta^{n,s}\partial_x) + \lambda_{n,s}\partial_x(\theta^{n,s}\partial_y) \\ \mu_{n,s}\partial_x(\theta^{n,s}\partial_y) + \lambda_{n,s}\partial_y(\theta^{n,s}\partial_x) & \alpha_{n,s}\partial_y(\theta^{n,s}\partial_y) + \mu_{n,s}\partial_x(\theta^{n,s}\partial_x) \end{bmatrix},$$

$$\mathcal{C} = \begin{bmatrix} \beta\theta^n\theta^s & 0 \\ 0 & \beta\theta^n\theta^s \end{bmatrix}, \quad \mathcal{G}_{n,s} = \begin{bmatrix} \theta^{n,s}\partial_x \\ \theta^{n,s}\partial_y \end{bmatrix}, \quad \mathcal{D}_{n,s} = \begin{bmatrix} \partial_x\theta^{n,s} \\ \partial_y\theta^{n,s} \end{bmatrix},$$

and $\alpha_{n,s} = (2\mu_{n,s} + \lambda_{n,s})$. Since $\theta^s = 1 - \theta^n$, we need only an equation for the transport of θ^n, which is again given by (2.9). As in 1-D, we use no-slip boundary conditions for (7.1) and no-flux for (2.9). Finally, the cubic function (2.8) is again used for modeling the osmotic pressure.

7.2. Discretization. The same computational strategy as outlined at the beginning of section 3 is used for the 2-D model. For the spatial discretization we again use a MAC grid where the positions of the unknowns are indicated in Figure 10. The mesh spacing in the x and y direction is set equal and is given by h.

All equations in (7.1) are discretized using second-order, centered finite differences, which leads to the following approximation of the first row of (7.1) at the
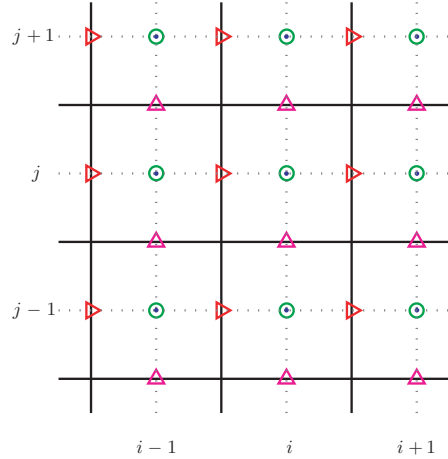
FIG. 10. *The location of the unknowns in the MAC grid for the 2-D gel model:* $\triangleright$=*network/solvent horizontal velocity,* $\triangle$=*network/solvent vertical velocity,* $\bullet$ =*pressure, and* $\bigcirc$ =*network/solvent volume fractions.*

interior point $(x_{i+\frac{1}{2},j}, y_{i+\frac{1}{2},j})$

$$
\begin{aligned}
(7.2) \quad & \frac{\alpha_{\mathrm{n}}}{h^2}\left[\theta^{\mathrm{n}}_{i+1,j}(u^{\mathrm{n}}_{i+\frac{3}{2},j} - u^{\mathrm{n}}_{i+\frac{1}{2},j}) - \theta^{\mathrm{n}}_{i,j}(u^{\mathrm{n}}_{i+\frac{1}{2},j} - u^{\mathrm{n}}_{i-\frac{1}{2},j})\right] \\
& + \frac{\mu_{\mathrm{n}}}{h^2}\left[\overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2},j+\frac{1}{2}}(u^{\mathrm{n}}_{i+\frac{1}{2},j+1} - u^{\mathrm{n}}_{i+\frac{1}{2},j}) - \overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2},j-\frac{1}{2}}(u^{\mathrm{n}}_{i+\frac{1}{2},j} - u^{\mathrm{n}}_{i+\frac{1}{2},j-1})\right] \\
& + \frac{\mu_{\mathrm{n}}}{h^2}\left[\overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2},j+\frac{1}{2}}(v^{\mathrm{n}}_{i+1,j+\frac{1}{2}} - v^{\mathrm{n}}_{i,j+\frac{1}{2}}) - \overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2},j-\frac{1}{2}}(v^{\mathrm{n}}_{i+1,j-\frac{1}{2}} - v^{\mathrm{n}}_{i,j-\frac{1}{2}})\right] \\
& + \frac{\lambda_{\mathrm{n}}}{h^2}\left[\theta^{\mathrm{n}}_{i+1,j}(v^{\mathrm{n}}_{i+1,j+\frac{1}{2}} - v^{\mathrm{n}}_{i+1,j-\frac{1}{2}}) - \theta^{\mathrm{n}}_{i,j}(v^{\mathrm{n}}_{i,j+\frac{1}{2}} - v^{\mathrm{n}}_{i,j-\frac{1}{2}})\right] \\
& - \beta\overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2},j}\overline{\theta}^{\mathrm{s}}_{i+\frac{1}{2},j}(u^{\mathrm{n}}_{i+\frac{1}{2},j} - u^{\mathrm{s}}_{i+\frac{1}{2},j}) - \overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2},j}\frac{p_{i+1,j} - p_{i,j}}{h} = \frac{\Psi(\theta^{\mathrm{n}}_{i+1,j}) - \Psi(\theta^{\mathrm{n}}_{i,j})}{h},
\end{aligned}
$$

while the approximation to the second row at the interior point $(x_{i,j+\frac{1}{2}}, y_{i,j+\frac{1}{2}})$ is given by

$$
\begin{aligned}
(7.3) \quad & \frac{\mu_{\mathrm{n}}}{h^2}\left[\overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2},j+\frac{1}{2}}(u^{\mathrm{n}}_{i+\frac{1}{2},j+1} - u^{\mathrm{n}}_{i+\frac{1}{2},j}) - \overline{\theta}^{\mathrm{n}}_{i-\frac{1}{2},j+\frac{1}{2}}(u^{\mathrm{n}}_{i-\frac{1}{2},j+1} - u^{\mathrm{n}}_{i-\frac{1}{2},j})\right] \\
& + \frac{\lambda_{\mathrm{n}}}{h^2}\left[\theta^{\mathrm{n}}_{i,j+1}(u^{\mathrm{n}}_{i+\frac{1}{2},j+1} - u^{\mathrm{n}}_{i-\frac{1}{2},j+1}) - \theta^{\mathrm{n}}_{i,j}(u^{\mathrm{n}}_{i+\frac{1}{2},j} - u^{\mathrm{n}}_{i-\frac{1}{2},j})\right] \\
& + \frac{\alpha_{\mathrm{n}}}{h^2}\left[\theta^{\mathrm{n}}_{i,j+1}(v^{\mathrm{n}}_{i,j+\frac{3}{2}} - v^{\mathrm{n}}_{i,j+\frac{1}{2}}) - \theta^{\mathrm{n}}_{i,j}(v^{\mathrm{n}}_{i,j+\frac{1}{2}} - v^{\mathrm{n}}_{i,j-\frac{1}{2}})\right] \\
& + \frac{\mu_{\mathrm{n}}}{h^2}\left[\overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2},j+\frac{1}{2}}(v^{\mathrm{n}}_{i+1,j+\frac{1}{2}} - v^{\mathrm{n}}_{i,j+\frac{1}{2}}) - \overline{\theta}^{\mathrm{n}}_{i-\frac{1}{2},j+\frac{1}{2}}(v^{\mathrm{n}}_{i,j+\frac{1}{2}} - v^{\mathrm{n}}_{i-1,j+\frac{1}{2}})\right] \\
& - \beta\overline{\theta}^{\mathrm{n}}_{i,j+\frac{1}{2}}\overline{\theta}^{\mathrm{s}}_{i,j+\frac{1}{2}}(v^{\mathrm{n}}_{i,j+\frac{1}{2}} - v^{\mathrm{s}}_{i,j+\frac{1}{2}}) - \overline{\theta}^{\mathrm{n}}_{i,j+\frac{1}{2}}\frac{p_{i,j+1} - p_{i,j}}{h} = \frac{\Psi(\theta^{\mathrm{n}}_{i,j+1}) - \Psi(\theta^{\mathrm{n}}_{i,j})}{h}.
\end{aligned}
$$

Bars over θ^{n} and θ^{s} represent the arithmetic averages of the values of these variables at nearest neighbor cells with two-point averages when there is a mix of integer and half-integer indices, and four-point averages when there are two half-integer indices. The discretizations of the third and fourth row of (7.1) are the same, but with variables for the network replaced accordingly by the variables for the solvent. Finally, the last
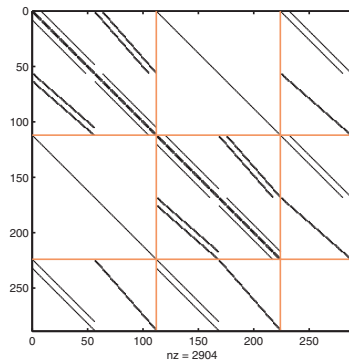
Fig. 11. *The structure of the $\mathcal{A}^h$ matrix in (7.5) for the case of a square domain with $N = M = 2^3$ grid cells; solid dots mark nonzero entries in the matrix, while solid horizontal and vertical lines highlight the structure shown in (7.5).*

row of (7.1) is approximated at $(x_{i,j}, y_{i,j})$ by

$$
(7.4) \quad \frac{-\overline{\theta}^{\mathrm{n}}_{i+\frac{1}{2},j} u^{\mathrm{n}}_{i+\frac{1}{2},j} + \overline{\theta}^{\mathrm{n}}_{i-\frac{1}{2},j} u^{\mathrm{n}}_{i-\frac{1}{2},j}}{h} + \frac{-\overline{\theta}^{\mathrm{n}}_{i,j+\frac{1}{2}} v^{\mathrm{n}}_{i,j+\frac{1}{2}} + \overline{\theta}^{\mathrm{n}}_{i,j-\frac{1}{2}} v^{\mathrm{n}}_{i,j-\frac{1}{2}}}{h}
$$
$$
+ \frac{-\overline{\theta}^{\mathrm{s}}_{i+\frac{1}{2},j} u^{\mathrm{s}}_{i+\frac{1}{2},j} + \overline{\theta}^{\mathrm{s}}_{i-\frac{1}{2},j} u^{\mathrm{s}}_{i-\frac{1}{2},j}}{h} + \frac{-\overline{\theta}^{\mathrm{s}}_{i,j+\frac{1}{2}} v^{\mathrm{s}}_{i,j+\frac{1}{2}} + \overline{\theta}^{\mathrm{s}}_{i,j-\frac{1}{2}} v^{\mathrm{s}}_{i,j-\frac{1}{2}}}{h} = 0.
$$

Where necessary, we use second-order extrapolation to account for the no-slip boundary conditions.

For a grid with N cell-centers in the x direction and M cell-centers in the y direction, the above approximations can be collected in a $(5NM - 2(N+M))$-by-$(5NM - 2(N+M))$ linear system, which we denote by

$$
(7.5) \quad \underbrace{\begin{bmatrix} \mathcal{L}^h_{\mathrm{n}} - \mathcal{C}^h & \mathcal{C}^h & -\mathcal{G}^h_{\mathrm{n}} \\ \mathcal{C}^h & \mathcal{L}^h_{\mathrm{s}} - \mathcal{C}^h & -\mathcal{G}^h_{\mathrm{s}} \\ \mathcal{G}^{h\,T}_{\mathrm{n}} & \mathcal{G}^{h\,T}_{\mathrm{s}} & 0 \end{bmatrix}}_{\mathcal{A}^h} \begin{bmatrix} \mathbf{u}^{\mathrm{n}} \\ \mathbf{u}^{\mathrm{s}} \\ p \end{bmatrix} = \begin{bmatrix} \nabla^h \Psi(\theta^{\mathrm{n}}) \\ 0 \\ 0 \end{bmatrix}.
$$

This forms the discrete approximation to (7.1). The properties that hold for the 1-D system (3.4) mentioned at the end of section 3.1 also hold for the 2-D system. The structure of (7.5) is illustrated in Figure 11 for the case of a square domain.

For the temporal discretization of (2.9), we again use explicit, first-order upwinding for the flux $\nabla \cdot (\theta^{\mathrm{n}} \mathbf{u}^{\mathrm{n}})$ (with LeVeque's transverse propagation correction [22]) and treat the diffusion implicitly with backward Euler. We use the same adaptive time stepping strategy as discussed in section 4.

7.3. Solving the coupled momentum and continuity equations. To solve the discrete system (7.5), we extend the multigrid box relaxation scheme introduced in section 5 to the 2-D system and use it as a right-preconditioner for GMRES(m). We briefly discuss the extension of the multigrid components for the 2-D system.

Transfer operators. As mentioned above, the mesh spacing in both the x and y directions is given by h, with $1/h$ a power of 2. We define a sequence of coarser grids, where each grid is a factor of two coarser than the previous in both the x

and y direction. For restriction and prolongation operators between the grids, we use standard full-weighting and bilinear interpolation defined on a MAC grid [35, p. 69–70].

Coarse grid discretization. The simple DCGA strategy for discretizing the gel system of equations on the coarser grids is again employed.

Smoother. We extend the box relaxation method described in section 5.1. For the 2-D system, this involves solving the discrete equations (7.5) locally in each computational cell (or box). For each interior box, this requires solving a 9-by-9 linear system (4 equations for the network velocity, 4 equations for the solvent velocity, and 1 equation for the pressure) for corrections to the unknowns $\mathbf{u}^n$, $\mathbf{u}^s$, and p. Boxes in the corners of the domain require solving 5-by-5 linear systems, while boxes on the edges require solving 7-by-7 systems. The exact form of all these systems can be worked out from (7.2)–(7.4). We update corrections to the unknowns in a Gauss–Seidel-type manner and combine this with underrelaxation similar to (5.2). The boxes are processed using red-black ordering.

Multigrid cycles We again compare the V- and F-cycle techniques illustrated in Figure 6 for cycling through the coarse grids. In all of our numerical results, we coarsen the grid until there are two cell-centers on the smaller side of the rectangular domain. At this point we solve the system (3.4) directly using MATLAB's sparse solver library. For a square domain in 2-D with $N = 2^k$ cell-centers in either direction (N^2 total points), the computational complexity of one V-cycle is $W_V = \frac{8}{3}C(N^2 - 4)$, while for the F-cycle it is $W_F = \frac{32}{9}C(N^2 - (3\log_2(N) + 1))$, for some C independent of N. For general rectangular domains with equal mesh spacing in both directions and standard coarsening, the constants in front of these work units will change; however, the ratio will remain the same with the F-cycle being 4/3 the cost of a V-cycle. Compare this to 1-D, where the F-cycle is twice the cost of a V-cycle.

We use one iteration of the above multigrid procedure with an initial guess of zero as a right-preconditioner for GMRES(20). Since the GMRES method is black-box, it requires no special modifications for the 2-D system.

7.4. Numerical results. To test the method in 2-D, we let the domain be the unit square and start with an initial distribution of network θ^n that is perturbed about the unstable region of the osmotic pressure Ψ (cf. Figure 1):

$$\theta^n = 0.08 + 2.5 \cdot 10^{-4}(\cos(6\pi x) + \cos(4\pi y)).$$

The model parameters are set to $\mu_n = 0.1$, $\lambda_n = 0.3$, $\mu_s = 0.025$, $\lambda_s = 0$, $\beta = 1$, and $\kappa = 10^{-7}$. The first four of these values make $\alpha_n = 0.5$ and $\alpha_s = 0.05$. As in 1-D, we define five stages of phase separation for the gel as illustrated in Figure 12, and we test the performance of the standalone MGV and MGF methods and preconditioned GMRES-MGV and GMRES-MGF methods at these stages.

Table 7.1 displays the scaling results for these solvers as the grid is refined from $h = 2^{-7}$ to $h = 2^{-9}$. Integer values listed in the table are the number of multigrid cycles (V(1,1) or F(1,1)) required to satisfy (6.1) using values from the previous time step as the initial guess (nonbracket) and using all zeros as the initial guess (bracket). Decimal values in parenthesis are the corresponding wall-clock times (in seconds) for the former method of generating initial guesses. The results are similar to those in 1-D. The MGV solver does not scale well and cannot solve the system for all stages. The MGF method scales linearly for all but Stage 5, for which the method actually diverges for $h = 2^{-9}$. The GMRES-MGV method performs very well for all stages of phase separation and shows a near linear scaling as h is refined. Finally, as with the
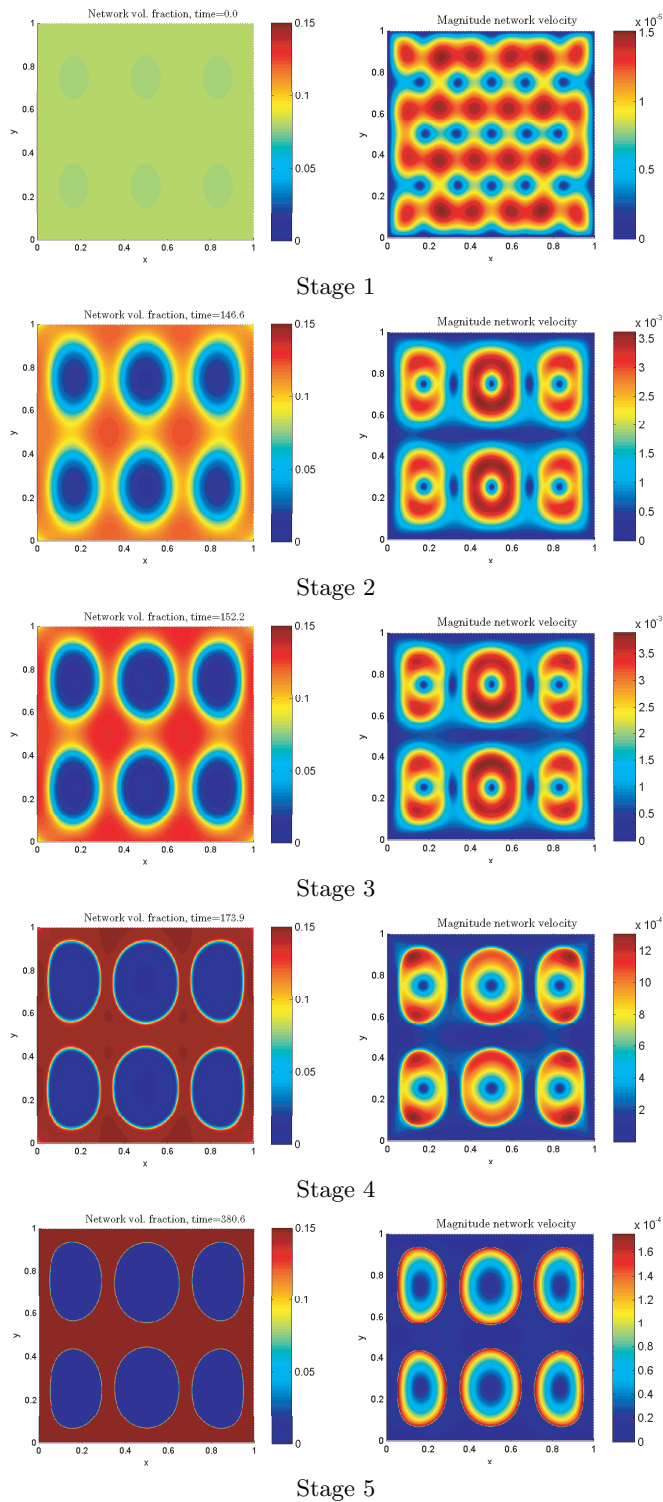
FIG. 12. *The network volume fraction (left column) and the magnitude of the network velocity (right column) from a numerical simulation of the 2-D model problem at various stages of gel phase separation. Note the fixed scale on the volume fraction plots and the variable scale on the velocity plots.*

TABLE 7.1

A comparison of the four techniques for solving the 2-D system (7.5) at the various stages of phase separation shown in Figure 12. Integer values are the number of V(1,1)- or F(1,1)-cycles required to satisfy (6.1) when using values from the previous time step as the initial guess (nonbracket) and using all zeros as the initial guess (bracket). Decimal values in parenthesis are the wall-clock times (in seconds) for the methods using the former of the initial guess methods. DIV means that the method diverged. The underrelaxation was set to $\omega = 0.675$ for all results.

Stage	$h = 2^{-7}$	$h = 2^{-8}$	$h = 2^{-9}$
	MGV		
1	11 [11] (18.5)	17 [17] (123)	27 [27] (798)
2	11 [11] (18.5)	14 [17] (101)	21 [26] (614)
3	10 [12] (16.9)	14 [16] (101)	22 [26] (648)
4	11 [13] (18.5)	27 [33] (197)	DIV
5	DIV	DIV	DIV
	MGF		
1	5 [5] (15.2)	5 [5] (65.1)	5 [5] (266)
2	5 [5] (15.2)	4 [5] (52.2)	3 [5] (161)
3	5 [5] (15.2)	4 [5] (52.2)	3 [5] (161)
4	5 [5] (15.2)	4 [5] (52.2)	5 [5] (266)
5	5 [7] (15.2)	4 [6] (52.2)	DIV
	GMRES-MGV		
1	10 [10] (17.4)	11 [11] (87.5)	12 [12] (389)
2	10 [10] (17.4)	10 [11] (79.5)	10 [12] (336)
3	10 [10] (17.4)	10 [11] (79.5)	10 [12] (336)
4	10 [11] (17.4)	10 [12] (79.5)	11 [14] (370)
5	9 [12] (15.9)	9 [15] (71.4)	10 [17] (336)
	GMRES-MGF		
1	6 [6] (18.9)	6 [6] (79.3)	6 [6] (359)
2	6 [6] (18.9)	5 [6] (66.4)	4 [6] (223)
3	5 [6] (15.7)	5 [6] (66.4)	4 [6] (223)
4	5 [6] (15.7)	5 [6] (66.4)	5 [6] (288)
5	5 [7] (15.7)	5 [7] (66.4)	10 [13] (542)

MGF method, the GMRES-MGF method scales linearly for all but Stage 5, where we see a jump in the number of iterations at $h = 2^{-9}$. However, unlike the MGF method, the GMRES-MGF method does not diverge but converges in a reasonable number of iterations. Basing the initial guesses on the previous time steps typically reduces the number of iterations for all the methods compared with using all zeros. This decrease is more pronounced in the MGV and GMRES-MGV methods. Comparing the wall-clock times, we see that the MGF method as a standalone solver or preconditioner is almost always more efficient than the MGV methods; this was not the case in 1-D (cf. Table 7.1).

To better understand the convergence behavior of the four methods for the finer $h = 2^{-9}$ mesh, we plot in Figure 13 their convergence history at the five stages of phase separation. The results for Stages 1–3 are similar for the four methods, with the GMRES-MGF method having the steepest slope and the MGV method the flattest. The individual slopes do not change significantly over these stages. At Stage 4, all but the MGV methods converged, and we see that the slopes of the remaining three methods are similar to those in Stages 1–3. The big difference comes with Stage 5. The MGF method diverges, while the decrease in the residual for the GMRES-MGV and GMRES-MGF methods is not very consistent until about ten iterations, at which the slopes of the lines again appear to resemble those from the previous stages. As discussed at the end of section 5.2 and observed in the 1-D experiments, this is most
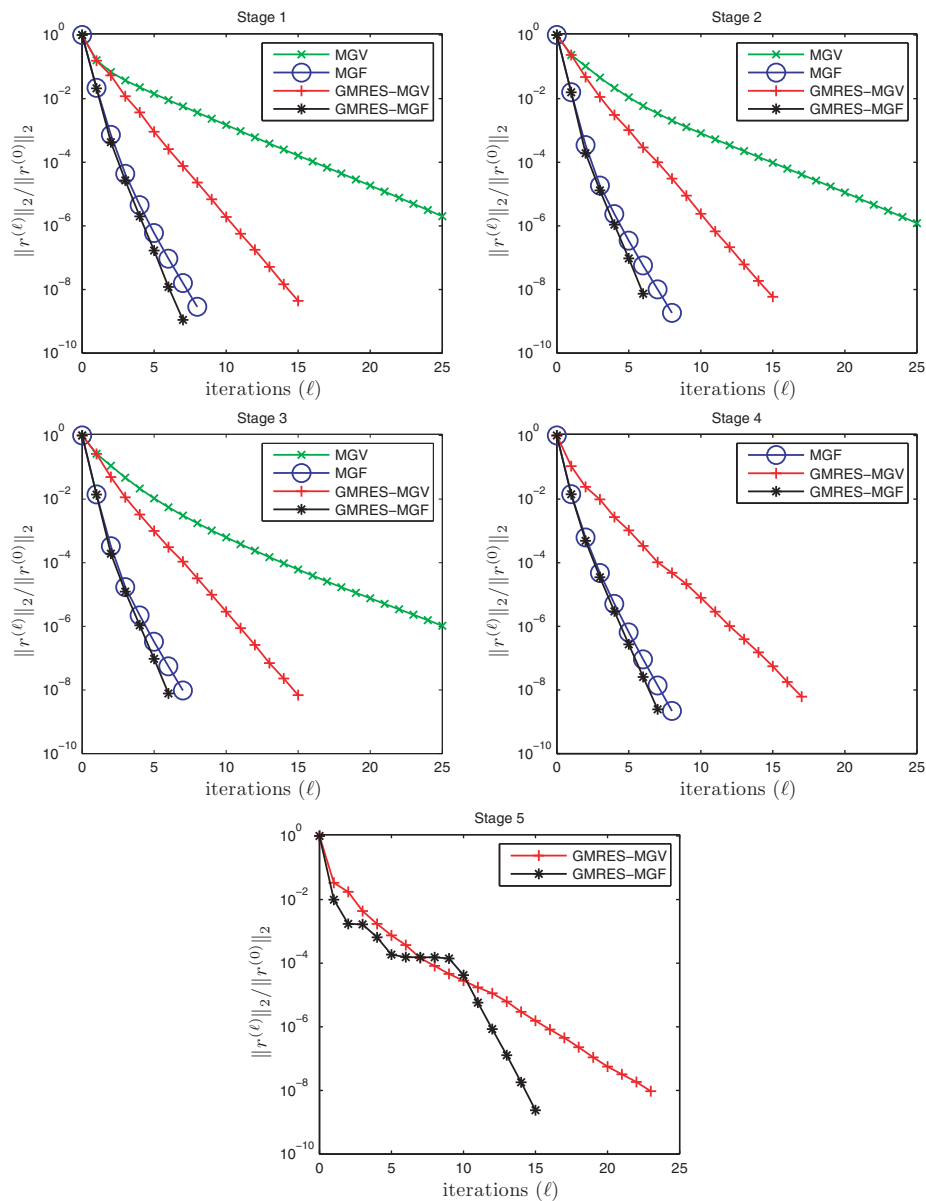
FIG. 13. *The convergence history of the MGV and MGF standalone solvers and the precondi-tioned GMRES-MGV and GMRES-MGF solvers for* (7.5) *at the various stages of phase separation shown in Figure* 12. *Missing results for MGV and MGF are because these methods diverged for that particular stage. The mesh-spacing is given by* $h = 2^{-9}$, *and the underrelaxation was set to* $\omega = 0.675$ *for all results.*

likely due to the GMRES method eliminating the problem eigenvalues of the MGF or MGV iteration matrices. Once these have been eliminated, the rapid convergence of these methods is again observed.

8. Concluding remarks. We have presented a computational methodology for simulating models of two-phase gel dynamics. The main computational challenge of these models is in solving the momentum and incompressibility equations which in-

volve variable-coefficient differential terms and terms coupling the two fluids. When discretized, these equations lead to a large, sparse linear system of saddle point type. Our method of solving this system by using multigrid with a box-type relaxation procedure as a preconditioner for GMRES(m) appears to be very effective. Numerical results from model problems in one and two dimensions indicate the method is both robust and efficient, with a near linear scaling in the computational cost. Furthermore, the method is straightforward to implement, since it uses standard transfer operators and direct coarse-grid discretization. For 2-D, using an F-cycle in the multigrid preconditioner appears to be more efficient than the classical V-cycle.

In our numerical experiments, storage was not an issue, since the size of the Krylov subspaces was not required to grow excessively large in order to solve the systems. If this happens to be the case and the number of preconditioned GMRES iterations is much larger than the restart value, one might consider using the BiCGSTAB method of van der Vorst [37]. This Krylov method only requires storing six intermediate vectors per iteration. However, BiCGSTAB would require two applications of the box-type multigrid preconditioner per iteration, whereas GMRES(m) only requires one.

The obvious way to improve the efficiency of the momentum and incompressibility equations solver is through parallelization. This will be especially important for 3-D applications. Parallel implementations of GMRES(m) are relatively straightforward and readily available [29, Ch. 11]. For Stokes and Navier–Stokes, an efficient parallel implementation of multigrid box relaxation is presented in [10]. The extension of this parallel scheme to the gel system will be pursued in a future study.

For geometrically complicated domains, one could consider combining our method with the embedded boundary method of Johansen and Colella [19] (see also [30, 31]). This too will be pursued in a future study.

The overall accuracy and efficiency of our computational methodology may be further improved through the use of high-resolution or implicit time-stepping schemes for numerically solving (2.1) [22]. Additionally, in parameter regimes where it may be necessary to simultaneously solve the momentum and incompressibility equations together with (2.1), our method would also be applicable. In this case, one could use a Newton-type iteration on the full nonlinear system in which the 1-D system (2.15) or 2-D system (7.1) would need to be solved at each iteration.

While the model we have considered in this paper treats the two fluids of the gel as Newtonian, it is common to use a non-Newtonian fluid model for the network phase where viscoelastic stresses on the gel are included [9, 13, 33]. The computational methodology we have introduced in this paper serves as a starting point for numerically addressing these more complicated models.

REFERENCES

[1] R. D. ALLEN AND R. R. COWDEN, *Syneresis in ameboid movement: Its localization by interference microscopy and its significance*, J. Cell Biol., 12 (1962), pp. 185–189.

[2] E. ALPKVIST AND I. KLAPPER, *A multidimensional multispecies continuum model for heterogenous biofilm development*, Bull. Math. Biol., 69 (2007), pp. 765–789.

[3] W. ALT AND M. DEMBO, *Cytoplasm dynamics and cell motion: Two-phase flow models*, Math. Biosci., 156 (1999), pp. 207–228.

[4] M. BENZI, G. H. GOLUB, AND J. LIESEN, *Numerical solution of saddle point problems*, Acta Numer., 14 (2005), pp. 1–137.

[5] A. BRANDT AND N. DINAR, *Multigrid solutions to flow problems*, in Numerical Methods for Partial Differential Equations, S. Parter, ed., Academic Press, New York, 1979, pp. 53–147.

[6] W. L. BRIGGS, V. E. HENSON, AND S. F. MCCORMICK, *A Multigrid Tutorial*, 2nd ed., SIAM, Philadelphia, 2000.

[7] H. BYRNE AND L. PREZIOSI, *Modelling solid tumor growth using the theory of mixtures*, Math. Med. Biol., 20 (2003), pp. 341–366.

[8] N. G. COGAN AND J. P. KEENER, *The role of the biofilm matrix in structural development*, Math. Med. Biol., 21 (2004), pp. 147–166.

[9] N. G. COGAN AND J. P. KEENER, *Channel formation in gels*, SIAM J. Appl. Math., 65 (2005), pp. 1839–1854.

[10] A. T. DEGANI AND G. C. FOX, *Parallel multigrid computation of the unsteady incompressible Navier–Stokes equations*, J. Comput. Phys., 128 (1996), pp. 223–236.

[11] M. DEMBO AND F. HARLOW, *Cell motion, contractile networks, and the physics of interpenetrating reactive flow*, Biophys. J., 50 (1986), pp. 109–121.

[12] M. DEMBO, *Mechanics and control of the cytoskeleton in Amoeba proteus*, Biophys. J., 55 (1989), pp. 1053–1080.

[13] M. DOI AND A. ONUKI, *Dynamic coupling between stress and composition in polymer solutions and blends*, J. Phys. II France, 2 (1992), pp. 1631–1656.

[14] D. DREW AND S. PASSMAN, *Theory of Multicomponent Fluids*, Springer, New York, 1999.

[15] D. DREW, *Mathematical modeling of two-phase flow*, Annu. Rev. Fluid Mech., 15 (1983), pp. 261–291.

[16] H. C. ELMAN, D. J. SILVESTER, AND A. J. WATHEN, *Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluid Dynamics*, Oxford University Press, Oxford, UK, 2005.

[17] F. H. HARLOW AND J. E. WELCH, *Numerical calculation of time-dependent viscous incompressible flow of fluid with free surfaces*, Phys. Fluids, 8 (1965), pp. 2182–2189.

[18] X. HE AND M. DEMBO, *On the mechanics of the first cleavage division of the sea urchin egg*, Exp. Cell. Res., 233 (1997), pp. 252–273.

[19] H. JOHANSEN AND P. COLELLA, *A Cartesian grid embedded boundary method for Poisson's equation on irregular domains*, J. Comput. Phys., 147 (1998), pp. 60–85.

[20] D. KESSLER, *Plasmodial structure and motility*, in Cell Biology of Physarum and Didymium, H. C. Aldrich and J. W. Daniel, eds., Academic Press, New York, 1982, pp. 145–196.

[21] G. LEMON, J. KING, H. BYRNE, O. JENSEN, AND K. SHAKESHEFF, *Mathematical modelling of engineered tissue growth using a multiphase porous flow mixture theory*, J. Math. Biol., 52 (2006), pp. 571–594.

[22] R. J. LEVEQUE, *High-resolution conservative algorithms for advection in incompressible flow*, SIAM J. Numer. Anal., 33 (1996), pp. 627–665.

[23] S. LUBKIN AND T. JACKSON, *Multiphase mechanics of capsule formation in tumors*, J. Biomech. Eng., 124 (2002), pp. 237–243.

[24] S. MANSERVISI, *Numerical analysis of Vanka-type solvers for steady Stokes and Navier-Stokes flows*, SIAM J. Sci. Comput., 44 (2006), pp. 2025–2056.

[25] C. W. OOSTERLEE AND T. WASHIO, *An evaluation of parallel multigrid as a solver and a preconditioner for singularly perturbed problems*, SIAM J. Sci. Comput., 19 (1998), pp. 87–110.

[26] C. W. OOSTERLEE AND T. WASHIO, *Krylov subspace acceleration of nonlinear multigrid with applications to recirculating flows*, SIAM J. Sci. Comput., 21 (2000), pp. 1670–1690.

[27] C. W. OOSTERLEE, R. WIENANDS, T. WASHIO, AND F. J. GASPAR, *The acceleration of multigrid convergence by recombination techniques*, in Multigrid methods, VI (Gent, 1999), Lecture Notes Comput. Sci. Eng. 14, Springer, Berlin, 2000, pp. 34–43.

[28] Y. SAAD AND M. H. SCHULTZ, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Comput., 7 (1986), pp. 856–869.

[29] Y. SAAD, *Iterative Methods for Sparse Linear Systems*, SIAM, Philadelphia, 2003.

[30] R. SAMULYAK, J. DU, J. GLIMM, AND Z. XU, *A numerical algorithm for mhd of free surface flows at low magnetic reynolds numbers*, J. Comput. Phys., 226 (2007), pp. 1532–1549.

[31] P. SCHWARTZ, M. BARAD, P. COLELLA, AND T. LIGOCKI, *A cartesian grid embedded boundary method for the heat equation and Poisson's equation in three dimensions*, J. Comput. Phys., 211 (2006), pp. 531–550.

[32] S. SIVALOGANATHAN, *The use of local mode analysis in the design and comparison of multigrid methods*, Comput. Phys. Comm., 65 (1991), pp. 246–252.

[33] H. TANAKA, *Viscoelastic phase separation*, J. Phys.: Condens. Matter, 12 (2000), pp. R207–R264.

[34] M. C. Thompson and J. H. Ferziger, *An adaptive multigrid technique for the incompressible Navier-Stokes equations*, J. Comput. Phys., 82 (1989), pp. 94–121.

[35] U. Trottenberg, C. W. Oosterlee, and A. Schüller, *Multigrid*, Academic Press, London, 2000.

[36] S. P. Vanka, *Block-implicit multigrid solution of Navier-Stokes equations in primitive variables*, J. Comput. Phys., 65 (1986), pp. 138–158.

[37] H. A. van der Vorst, *BI-CGSTAB: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems*, SIAM J. Sci. Comput., 13 (1992), pp. 631–644.

[38] T. Washio and C. W. Oosterlee, *Krylov subspace acceleration for nonlinear multigrid schemes*, Electron. Trans. Numer. Anal., 6 (1997), pp. 271–290.

[39] P. Wesseling and C. W. Oosterlee, *Geometric multigrid with applications to computational fluid dynamics*, J. Comput. Appl. Math., 128 (2001), pp. 311–334.

[40] P. Wesseling, *An Introduction to Multigrid Methods*, Wiley, Chichester, UK, 1992.