Commun. Korean Math. Soc. **20** (2005), No. 1, pp. 179-193

A HYBRID METHOD FOR HIGHER-ORDER NONLINEAR DIFFUSION EQUATIONS

JUNSEOK KIM AND JEANMAN SUR

ABSTRACT. We present results of fully nonlinear time-dependent simulations of a thin liquid film flowing up an inclined plane. Equations of the type $h_t+f_y(h) = -\epsilon^3\nabla\cdot(M(h)\nabla\Delta h)$ arise in the context of thin liquid films driven by a thermal gradient with a counteracting gravitational force, where $h = h(\mathbf{x}, t)$ is the fluid film height. A hybrid scheme is constructed for the solution of two-dimensional higher-order nonlinear diffusion equations. Problems in the fluid dynamics of thin films are solved to demonstrate the accuracy and effectiveness of the hybrid scheme.

1. Introduction

Thin coating flows are of great technical and scientific interest [5]. Practical examples include manufacturing processes such as the production of videotapes, photographic films, and microchips [4].

Flows created by gradients in surface tension, whether induced by temperature or concentration variations, are commonly called thermocapillary or Marangoni-driven flows. For very thin capillary driven films, an experiment shows that the Marangoni stress causes a capillary ridge to form and the film to finger (see Fig. 1(a)).

Equations of the type $h_t + f_y(h) = -\epsilon^3\nabla\cdot(M(h)\nabla\Delta h)$ arise in the context of thin liquid films driven by a thermal gradient with a counteracting gravitational force, where $h = h(\mathbf{x}, t)$ is the fluid film height. We treat $f_y(h)$ with essentially non-oscillatory (ENO) scheme, which is

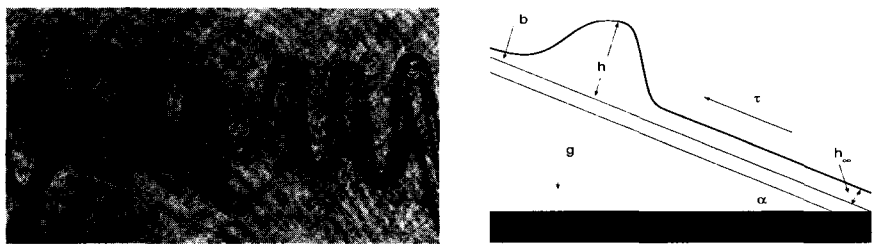
Received April 18, 2004.

2000 Mathematics Subject Classification: 76A20, 65M55.

Key words and phrases: nonlinear diffusion equations, thin film, nonlinear multigrid method.

This work was supported by National Science Foundation. The authors acknowledge the support of National Science Foundation grants No. DMS-0074049 and No. DMS-0244498 and ONR grant No. N000140110290.

good for shock capturing [6]. We combine this approach with a nonlinear multigrid method for solving nonlinear system. Multigrid methods are generally accepted as among *the fastest numerical methods* for solving this type of partial differential equations [8]. We use a nonlinear multigrid method to solve resulting equations accurately and efficiently.



(a) (b)

FIGURE 1. (a) Thin film flow climbs up an inclined plate, which is driven by Marangoni stresses against the gravity. (b) A schematic diagram of the physical problem.

This paper is organized as follows. In Section 2, we review the governing equations. In Section 3, we derive numerical solution with a nonlinear multigrid method. In Section 4, we present numerical results. Conclusions are made in Section 5.

2. Governing equations

We consider the dynamics of a thin layer of liquid of thickness h on an inclined surface driven by thermally created surface tension gradients and influenced by gravity (see Fig. 1(b)). The spatial variables x and y denote the direction normal to the flow and the direction of the flow, respectively. Let α, ρ, g, η, γ, and $\tau = d\gamma/dy$ denote the angle from horizontal of inclination of the plane, the density, the gravitational constant, the dynamic viscosity, the surface tension, and the surface tension gradient of the liquid.

We model the dynamics of the draining film using the lubrication approximation with a "depth averaged" velocity [2],

$$\vec{V} = \left(\frac{\tau h}{2\eta} - \frac{\rho g h^2 \sin\alpha}{3\eta}\right)\vec{e}_y + \frac{\gamma h^2 \nabla\Delta h}{3\eta}. \tag{1}$$

The coefficient of $\vec{e}_y = (0,1)$ in Eq. (1) represents convection due to the surface tension gradient and the component of gravity tangent to the

surface. The component of gravity normal to the surface has a negligible effect. We couple Eq. (1) with mass conservation, $h_t + \nabla \cdot (h\vec{V}) = 0$. We rescale to dimensionless units as in [2]: $h = H\hat{h}$, $(x, y) = l(\hat{x}, \hat{y})$, and $t = T\hat{t}$, where $H = \frac{3\tau}{2\rho g \sin\alpha}$, $l = (\frac{2\gamma}{3\tau H^2})^{1/3} = (\frac{3\gamma\tau}{2\rho^2 g^2 \sin^2\alpha})^{1/3}$, and $T = \frac{2\eta}{\tau^2}(\frac{4}{9}\tau\gamma\rho g \sin\alpha)^{1/3}$. Dropping the $\hat{}$ gives the dimensionless equation

$$h_t + (h^2 - h^3)_y = -\nabla \cdot (h^3 \nabla \Delta h). \tag{2}$$

In the model, we take film thickness as a given constant height h_∞ and zero third derivative $h_{yyy} = 0$ at $y = 0$. We choose the simplest boundary condition consistent with complete wetting, that of a precursor model in which $h \to b > 0$ as $y \to \infty$ [1, 7].

We can think of this traveling wave as a viscous regularization of a shock wave, if we rescale the space and the time variables by $(x', y') = \epsilon(x, y)$ and $t' = \epsilon t$, then Eq. (2) becomes after dropping the $'$ notation

$$h_t + (h^2 - h^3)_y = -\epsilon^3 \nabla \cdot (h^3 \nabla \Delta h). \tag{3}$$

This equation is a fourth order nonlinear singular perturbation of the conservative law

$$h_t + (h^2 - h^3)_y = 0,$$

which has a nonconvex flux function $f(h) = h^2 - h^3$ [3].

3. Numerical analysis

Straightforward explicit time-marching requires that time steps be very small in order to maintain numerical stability; especially Δt must be no larger than order Δx^4, where Δx is the space step or mesh size. Fourth-order parabolic equations are very stiff; the stability constraint on the time-step for explicit methods, $\Delta t = O(\Delta x^4)$, is prohibitive, hence implicit methods are necessary. We split the Eq. (3) into a system of the following form

$$h_t + f_y(h) = \nabla \cdot (M(h) \nabla \mu), \tag{4}$$

$$\mu = -\epsilon^3 \Delta h, \tag{5}$$

where $f(h) = h^2 - h^3$ and $M(h) = h^3$.

Now we present fully discrete schemes for the Eqs. (4) and (5) in two dimensional space, i.e., $\Omega = (a, b) \times (c, d)$. Let N_x and N_y be positive even integers, $\Delta x = (b - a)/N_x$ be the uniform mesh size, and $\Omega_{\Delta x} = \{(x_i, y_j) : x_i = (i - 0.5)\Delta x, y_j = (j - 0.5)\Delta x,\ 1 \le i \le N_x,\ 1 \le j \le N_y\}$ be the set of cell-centers.

Since the film thickness h and μ satisfy Neumann boundary conditions, it is natural to define them at cell centers. Let h_{ij} and μ_{ij} be approximations of $h(x_i, y_j)$ and $\mu(x_i, y_j)$, respectively. We first define the discrete differentiation operators:

$$D_x h_{i+\frac{1}{2},j} = \frac{h_{i+1,j} - h_{ij}}{\Delta x}, \; D_y h_{i,j+\frac{1}{2}} = \frac{h_{i,j+1} - h_{ij}}{\Delta x}.$$

For a grid function h defined at cell centers, $D_x h$ and $D_y h$ are defined at cell-edges, and we use the following notation

$$\nabla_d h_{ij} = (D_x h_{i+\frac{1}{2},j}, D_y h_{i,j+\frac{1}{2}})$$

to represent the discrete gradient of h at cell-edges. Correspondingly, the divergence at cell-centers, using values from cell-edges, is

$$\nabla_d \cdot \mathbf{g}_{ij} = (g^1_{i+\frac{1}{2},j} - g^1_{i-\frac{1}{2},j} + g^2_{i,j+\frac{1}{2}} - g^2_{i,j-\frac{1}{2}})/\Delta x,$$

for a grid function $\mathbf{g} = (g^1, g^2)$ defined on cell-edges. We then define the discrete Laplacian by

$$\Delta_d h_{ij} = \nabla_d \cdot \nabla_d h_{ij}$$

and the discrete l_2 inner product by

$$(c, d)_{\Delta x} = \Delta x^2 \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} c_{ij} d_{ij}. \tag{6}$$

We also define discrete norms associated with Eq. (6) as $||h||^2 = (h, h)_{\Delta x}$.

3.1. Discretization and properties of the proposed scheme

We present a semi-implicit time (Crank-Nicholson) and centered difference space discretization of Eqs. (4) and (5).

$$\frac{h^{n+1}_{ij} - h^n_{ij}}{\Delta t} = \nabla_d \cdot [M(h)^{n+\frac{1}{2}}_{ij} \nabla_d \mu^{n+\frac{1}{2}}_{ij}] - f_y(h^{n+\frac{1}{2}}_{ij}), \tag{7}$$

$$\mu^{n+\frac{1}{2}}_{ij} = -\frac{\epsilon^3}{2} \Delta_d (h^{n+1}_{ij} + h^n_{ij}), \tag{8}$$

where $\nabla_d \cdot [M(h)^{n+\frac{1}{2}}_{ij} \nabla_d \mu^{n+\frac{1}{2}}_{ij}]$ is described at Eq. (9) in detail and the superscripts denote discrete time steps. $f_y(h^{n+\frac{1}{2}}_{ij})$ is treated by using a second-order ENO scheme [6], i.e.,

$$f_y(h^{n+\frac{1}{2}}_{ij}) = f'(h^{n+\frac{1}{2}}_{ij}) \left(\frac{\bar{h}^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} - \bar{h}^{n+\frac{1}{2}}_{i,j-\frac{1}{2}}}{\Delta x} \right),$$

where the half time values $h_{ij}^{n+\frac{1}{2}}$ are computed using an extrapolation from previous values, $h_{ij}^{n+\frac{1}{2}} = (3h_{ij}^{n} - h_{ij}^{n-1})/2$. The edge values $\bar{h}_{i,j+\frac{1}{2}}^{n+\frac{1}{2}}$ are computed as follows:

$$k = \begin{cases} j & f'(h_{i,j+\frac{1}{2}}^{n+\frac{1}{2}}) \geq 0 \\ j+1 & \text{otherwise} \end{cases}$$

$$a = \frac{h_{ik}^{n+\frac{1}{2}} - h_{i,k-1}^{n+\frac{1}{2}}}{\Delta x}, \quad b = \frac{h_{i,k+1}^{n+\frac{1}{2}} - h_{ik}^{n+\frac{1}{2}}}{\Delta x}, \quad d = \begin{cases} a & \text{if } |a| \leq |b| \\ b & \text{otherwise} \end{cases}$$

$$\bar{h}_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} = h_{ik}^{n+\frac{1}{2}} + \frac{\Delta x}{2} d(1 - 2(k-j)).$$

The quantities $\bar{h}_{i,j-\frac{1}{2}}$ are computed in a similar manner.

3.2. Numerical solution - A nonlinear multigrid method

In this section, we develop a nonlinear Full Approximation Storage (FAS) multigrid method to solve the nonlinear discrete system (7) and (8) at the implicit time level. A pointwise Gauss-Seidel relaxation scheme is used as the smoother in the multigrid method. See the reference text [8] for additional details and background. The algorithm of the nonlinear multigrid method for solving the discrete system is :
First, let us rewrite Eqs. (7) and (8) as follows.

$$N(h^{n+1}, \mu^{n+\frac{1}{2}}) = (\phi^n, \psi^n),$$

where

$$N(h^{n+1}, \mu^{n+\frac{1}{2}}) = \left(\frac{h_{ij}^{n+1}}{\Delta t} - \nabla_d \cdot (M(h)_{ij}^{n+\frac{1}{2}} \nabla_d \mu_{ij}^{n+\frac{1}{2}}), \mu_{ij}^{n+\frac{1}{2}} + \frac{\epsilon^3}{2} \Delta_d h_{ij}^{n+1} \right)$$

and the source term is

$$(\phi^n, \ \psi^n) = \left(\frac{h_{ij}^{n}}{\Delta t} - f_y(h_{ij}^{n+\frac{1}{2}}), \ -\frac{\epsilon^3}{2} \Delta_d h_{ij}^{n} \right).$$

In the following description of one FAS cycle, we assume a sequence of grids Ω_k (Ω_{k-1} is coarser than Ω_k by factor 2). Given the number ν of pre- and post- smoothing relaxation sweeps, an iteration step for the nonlinear multigrid method using the V-cycle is formally written as follows [8]:

FAS multigrid cycle

$$\{h_k^{m+1}, \mu_k^{m+\frac{1}{2}}\} = FAScycle(k, h_k^n, h_k^m, \mu_k^{m-\frac{1}{2}}, N_k, \phi_k^n, \psi_k^n, \nu).$$

That is, $\{h_k^m, \mu_k^{m-\frac{1}{2}}\}$ and $\{h_k^{m+1}, \mu_k^{m+\frac{1}{2}}\}$ are the approximations of h^{n+1} (x_i, y_j) and $\mu^{n+\frac{1}{2}}(x_i, y_j)$ before and after an FAS cycle. Now, define the FAS cycle.

1) **Presmoothing**

Compute $\{\bar{h}_k^m,\ \bar{\mu}_k^{m-\frac{1}{2}}\}$ by applying ν smoothing steps to $\{h_k^m,\ \mu_k^{m-\frac{1}{2}}\}$

$$\{\bar{h}_k^m,\ \bar{\mu}_k^{m-\frac{1}{2}}\} = SMOOTH^{\nu}(h_k^n, h_k^m, \mu_k^{m-\frac{1}{2}}, N_k, \phi_k^n, \psi_k^n),$$

which means performing ν smoothing steps with the initial approximations h_k^n, $h_k^m, \mu_k^{m-\frac{1}{2}}$, source terms ϕ_k^n, ψ_k^n, and $SMOOTH$ relaxation operator to get the approximations $\bar{h}_k^m$, $\bar{\mu}_k^{m-\frac{1}{2}}$. One $SMOOTH$ relaxation operator step consists of solving the system (11) and (12) given below by 2×2 matrix inversion for each i and j.

Here, we derive the smoothing operator in two dimensions. Rewriting Eqs. (7) and (8), we get

$$\begin{aligned} (9) \qquad & \frac{h_{ij}^{n+1}}{\Delta t} + \frac{M_{i+\frac{1}{2},j}^{n+\frac{1}{2}} + M_{i-\frac{1}{2},j}^{n+\frac{1}{2}} + M_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} + M_{i,j-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x^2} \mu_{ij}^{n+\frac{1}{2}} = \phi_{ij}^n \\ & + \frac{M_{i+\frac{1}{2},j}^{n+\frac{1}{2}} \mu_{i+1,j}^{n+\frac{1}{2}} + M_{i-\frac{1}{2},j}^{n+\frac{1}{2}} \mu_{i-1,j}^{n+\frac{1}{2}} + M_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} \mu_{i,j+1}^{n+\frac{1}{2}} + M_{i,j-\frac{1}{2}}^{n+\frac{1}{2}} \mu_{i,j-1}^{n+\frac{1}{2}}}{\Delta x^2} \end{aligned}$$

and

$$\begin{aligned} (10) \qquad -\frac{2\epsilon^2}{\Delta x^2} h_{ij}^{n+1} + \mu_{ij}^{n+\frac{1}{2}} &= \psi_{ij}^n - \frac{\epsilon^3}{2\Delta x^2}(h_{i+1,j}^{n+1} + h_{i-1,j}^{n+1}) \\ & \quad - \frac{\epsilon^3}{2\Delta x^2}(h_{i,j+1}^{n+1} + h_{i,j-1}^{n+1}). \end{aligned}$$

Next, we replace h_{kl}^{n+1} and $\mu_{kl}^{n+\frac{1}{2}}$ in the Eqs. (9) and (10) with $\bar{h}_{kl}^{m}$ and $\bar{\mu}_{kl}^{m-\frac{1}{2}}$ if $k \le i$ and $l \le j$, otherwise with h_{kl}^{m} and $\mu_{kl}^{m-\frac{1}{2}}$, i.e.,

$$(11)\quad \frac{\bar{h}_{ij}^{m}}{\Delta t} + \frac{M_{i+\frac{1}{2},j}^{m-\frac{1}{2}} + M_{i-\frac{1}{2},j}^{m-\frac{1}{2}} + M_{i,j+\frac{1}{2}}^{m-\frac{1}{2}} + M_{i,j-\frac{1}{2}}^{m-\frac{1}{2}}}{\Delta x^2}\bar{\mu}_{ij}^{m-\frac{1}{2}} = \phi_{ij}^{n} + \frac{M_{i+\frac{1}{2},j}^{m-\frac{1}{2}}\mu_{i+1,j}^{m-\frac{1}{2}} + M_{i-\frac{1}{2},j}^{m-\frac{1}{2}}\bar{\mu}_{i-1,j}^{m-\frac{1}{2}} + M_{i,j+\frac{1}{2}}^{m-\frac{1}{2}}\mu_{i,j+1}^{m-\frac{1}{2}} + M_{i,j-\frac{1}{2}}^{m-\frac{1}{2}}\bar{\mu}_{i,j-1}^{m-\frac{1}{2}}}{\Delta x^2},$$

where $M_{i+\frac{1}{2},j}^{m-\frac{1}{2}} = M((h_{ij}^{m} + h_{i+1,j}^{m} + h_{ij}^{n} + h_{i+1,j}^{n})/4)$ and the other terms are similarly defined.

$$(12)\quad -\frac{2\epsilon^3\bar{h}_{ij}^{m}}{\Delta x^2} + \bar{\mu}_{ij}^{m-\frac{1}{2}} = \psi_{ij}^{n} - \frac{\epsilon^3(h_{i+1,j}^{m} + \bar{h}_{i-1,j}^{m} + h_{i,j+1}^{m} + \bar{h}_{i,j-1}^{m})}{2\Delta x^2}.$$

Now, we obtain the smoothing operator (11) and (12).

2) **Compute the defect**

$$(\bar{d_1}^{m}{}_{k}, \bar{d_2}^{m}{}_{k}) = (\phi_k^n, \psi_k^n) - N_k(\bar{h}_k^m, \bar{\mu}_k^{m-\frac{1}{2}}).$$

3) **Restrict the defect and** $\{\bar{h}_k^m,\ \bar{\mu}_k^{m-\frac{1}{2}}\}$

$$(\bar{d_1}^{m}{}_{k-1}, \bar{d_2}^{m}{}_{k-1}) = I_k^{k-1}(\bar{d_1}^{m}{}_{k}, \bar{d_2}^{m}{}_{k}),\ (\bar{h}_{k-1}^m, \bar{\mu}_{k-1}^{m-\frac{1}{2}}) = I_k^{k-1}(\bar{h}_k^m, \bar{\mu}_k^{m-\frac{1}{2}}).$$

The restriction operator I_k^{k-1} maps k-level functions to $(k-1)$-level functions.

$$d_{k-1}(x_i, y_j) = I_k^{k-1}d_k(x_i, y_j) = \frac{1}{4}\Big[d_k(x_{i-\frac{1}{2}}, y_{j-\frac{1}{2}}) + d_k(x_{i-\frac{1}{2}}, y_{j+\frac{1}{2}}) + d_k(x_{i+\frac{1}{2}}, y_{j-\frac{1}{2}}) + d_k(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}})\Big].$$

That is, coarse grid values are obtained by averaging the four nearby fine grid values.

4) **Compute the right-hand side**

$$(\phi_{k-1}^n, \psi_{k-1}^n) = (\bar{d_1}^{m}{}_{k-1}, \bar{d_2}^{m}{}_{k-1}) + N_{k-1}(\bar{h}_{k-1}^m, \bar{\mu}_{k-1}^{m-\frac{1}{2}}).$$

5) **Compute an approximate solution** $\{\hat{h}_{k-1}^m,\ \hat{\mu}_{k-1}^{m-\frac{1}{2}}\}$ **of the**

coarse grid equation on Ω_{k-1}, i.e.

$$N_{k-1}(h_{k-1}^m, \mu_{k-1}^{m-\frac{1}{2}}) = (\phi_{k-1}^n, \psi_{k-1}^n). \tag{13}$$

If $k = 1$, we explicitly invert a 2×2 matrix to obtain the solution. If $k > 1$, we solve (13) by performing a FAS k-grid cycle using $\{\bar{h}_{k-1}^m,\ \bar{\mu}_{k-1}^{m-\frac{1}{2}}\}$ as an initial approximation:

$$\{\hat{h}_{k-1}^m, \hat{\mu}_{k-1}^{m-\frac{1}{2}}\} = \text{FAScycle}(k-1, h_{k-1}^n, \bar{h}_{k-1}^m, \bar{\mu}_{k-1}^{m-\frac{1}{2}}, N_{k-1}, \phi_{k-1}^n, \psi_{k-1}^n, \nu).$$

6) Compute the coarse grid correction (CGC):

$$\hat{v}_{1k-1}^m = \hat{h}_{k-1}^m - \bar{h}_{k-1}^m,\ \hat{v}_{2k-1}^{m-\frac{1}{2}} = \hat{\mu}_{k-1}^{m-\frac{1}{2}} - \bar{\mu}_{k-1}^{m-\frac{1}{2}}.$$

7) Interpolate the correction

$$\hat{v}_{1k}^m = I_{k-1}^k \hat{v}_{1k-1}^m,\ \hat{v}_{2k}^{m-\frac{1}{2}} = I_{k-1}^k \hat{v}_{2k-1}^{m-\frac{1}{2}}.$$

The interpolation operator I_{k-1}^k maps $(k-1)$-level functions to k-level functions. Here, the coarse values are simply transferred to the four nearby fine grid points, $v_k(x_i, y_j) = I_{k-1}^k v_{k-1}(x_i, y_j) = v_{k-1}(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}})$ for i and j odd-numbered integers. The values at the other node points are given by

$$v_k(x_{i+1}, y_j) = v_k(x_i, y_{j+1}) = v_k(x_{i+1}, y_{j+1}) = v_{k-1}(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}),$$

where i and j are odd.

8) Compute the corrected approximation on Ω_k

$$h_k^{m,\ \text{after } CGC} = \bar{h}_k^m + \hat{v1}_k^m,\ \mu_k^{m-\frac{1}{2},\ \text{after } CGC} = \bar{\mu}_k^{m-\frac{1}{2}} + \hat{v}_{2k}^{m-\frac{1}{2}}.$$

9) Postsmoothing

$$\{h_k^{m+1}, \mu_k^{m+\frac{1}{2}}\} = SMOOTH^{\nu}(h_k^n, h_k^{m,\ \text{after } CGC}, \mu_k^{m-\frac{1}{2},\ \text{after } CGC}, N_k, \phi_k^n, \psi_k^n).$$

This completes the description of a nonlinear FAScycle.

3.3. Local Fourier analysis

To analyze the behavior of the multigrid method, we linearize the nonlinear scheme and perform a local Fourier analysis (e.g.see [8]). In particular, we analyze the smoother since the performance of the multigrid method depends strongly on the smoother.

We "freeze" the coefficient, $M(h)$, at a representative value $M_\xi = M(\xi)$, for some $0 \le \xi \le h_{max}$. Substituting $\mu_{ij}^{n+\frac{1}{2}}$ into Eq. (7), the scheme becomes

$$L_{\Delta x} h_{\Delta x}^{n+1} = s_{\Delta x}^n,$$

where

$$\begin{aligned} L_{\Delta x} h_{\Delta x}^{n+1} \quad := \quad & \frac{h_{ij}^{n+1}}{\Delta t M_\xi} + \frac{\epsilon^3}{2\Delta x^4}[h_{i-2,j}^{n+1} + h_{i+2,j}^{n+1} + h_{i,j-2}^{n+1} + h_{i,j+2}^{n+1} \\ & +2(h_{i-1,j+1}^{n+1} + h_{i-1,j-1}^{n+1} + h_{i+1,j+1}^{n+1} + h_{i+1,j-1}^{n+1}) \\ & -8(h_{i-1,j}^{n+1} + h_{i,j-1}^{n+1} + h_{i+1,j}^{n+1} + h_{i,j+1}^{n+1}) + 20h_{ij}^{n+1}] \end{aligned} \tag{14}$$

and

$$s_{\Delta x}^n = -\frac{\epsilon^3}{2}\Delta_d^2 h_{ij}^n + \frac{h_{ij}^n}{\Delta t M_\xi} - \frac{1}{M_\xi} f_y(h_{ij}^{n+\frac{1}{2}}).$$

For Gauss-Seidel iteration with a lexicographic ordering of the grid points applied to the above Eq. (14), we have the following operator decomposition.

$$\begin{aligned} L_{\Delta x}^+ h_{\Delta x}^{n+1} \quad := \quad & \frac{h_{ij}^{n+1}}{\Delta t M_\xi} + \frac{\epsilon^3}{2\Delta x^4}[h_{i-2,j}^{n+1} + h_{i,j-2}^{n+1} + 2(h_{i-1,j+1}^{n+1} + h_{i-1,j-1}^{n+1}) \\ & -8(h_{i-1,j}^{n+1} + h_{i,j-1}^{n+1}) + 20h_{ij}^{n+1}], \\ L_{\Delta x}^- h_{\Delta x}^{n+1} \quad := \quad & \frac{\epsilon^3}{2\Delta x^4}[h_{i+2,j}^{n+1} + h_{i,j+2}^{n+1} + 2(h_{i+1,j+1}^{n+1} + h_{i+1,j-1}^{n+1}) \\ & -8(h_{i+1,j}^{n+1} + h_{i,j+1}^{n+1})]. \end{aligned}$$

Therefore, this relaxation method can be written *locally* as

$$L_{\Delta x}^+ \tilde{z}_{\Delta x} + L_{\Delta x}^- z_{\Delta x} = s_{\Delta x}^n, \tag{15}$$

where $z_{\Delta x}$ corresponds to the old approximation of $h_{\Delta x}$ (approximation before the relaxation step) and $\tilde{z}_{\Delta x}$ to the new approximation (after the step). Subtracting Eq. (15) from the discrete equation $L_{\Delta x} h_{\Delta x} = s_{\Delta x}$ and letting $\tilde{v}_{\Delta x} = h_{\Delta x} - \tilde{z}_{\Delta x}$ and $v_{\Delta x} = h_{\Delta x} - z_{\Delta x}$, we obtain the equation

$$L_{\Delta x}^+ \tilde{v}_{\Delta x} + L_{\Delta x}^- v_{\Delta x} = 0,$$

or, equivalently,

$$\tilde{v}_{\Delta x} = S_{\Delta x} v_{\Delta x},$$

where $S_{\Delta x} = -(L_{\Delta x}^+)^{-1} L_{\Delta x}^-$ is the resulting smoothing operator. Applying $L_{\Delta x}^+$ and $L_{\Delta x}^-$ to the formal eigenfunctions $e^{i\theta_1 x/\Delta x} e^{i\theta_2 y/\Delta x}$, we obtain

$$L_{\Delta x}^+ e^{i\theta_1 x/\Delta x} e^{i\theta_2 y/\Delta x} = \hat{L}_{\Delta x}^+ e^{i\theta_1 x/\Delta x} e^{i\theta_2 y/\Delta x},$$

$$L^-_{\Delta x} e^{i\theta_1 x/\Delta x} e^{i\theta_2 y/\Delta x} = \hat{L}^-_{\Delta x} e^{i\theta_1 x/\Delta x} e^{i\theta_2 y/\Delta x},$$

where $\hat{L}^+_{\Delta x}$ and $\hat{L}^-_{\Delta x}$ are the formal eigenvalues of the operators $L^+_{\Delta x}$ and $L^-_{\Delta x}$, respectively:

$$\begin{aligned}
\hat{L}^+_{\Delta x}(\theta_1,\theta_2) &= \frac{1}{\Delta t M_\xi} + \frac{\epsilon^3}{2\Delta x^4}[e^{-2i\theta_1} + e^{-2i\theta_2} \\
&\quad +2(e^{-i(\theta_1-\theta_2)} + e^{-i(\theta_1+\theta_2)}) - 8(e^{-i\theta_1} + e^{-i\theta_2}) + 20], \\
\hat{L}^-_{\Delta x}(\theta_1,\theta_2) &= \frac{\epsilon^3}{2\Delta x^4}[e^{2i\theta_1} + e^{2i\theta_2} + 2(e^{i(\theta_1+\theta_2)} + e^{i(\theta_1-\theta_2)}) \\
&\quad -8(e^{i\theta_1} + e^{i\theta_2})].
\end{aligned}$$

The amplification factor of the relaxation scheme is

$$\hat{S}_{\Delta x}(\theta_1,\theta_2) := -\frac{\hat{L}^-_{\Delta x}(\theta_1,\theta_2)}{\hat{L}^+_{\Delta x}(\theta_1,\theta_2)}.$$

Define the smoothing factor:

$$\mu_{loc}(S_{\Delta x}) := \sup\{|\hat{S}_{\Delta x}(\theta_1,\theta_2)| : \ \frac{\pi}{2} \le |\theta_1|, \ |\theta_2| \le \pi\}.$$

Table 1 shows $\mu_{loc}(S_{\Delta x})$ factors with $M_\xi = 1.0$ and different mesh sizes. This result suggests that the multigrid method with time step $\Delta t \sim \Delta x$ converges uniformly with respect to increasing resolutions. Correspondingly, this would impose a first order time step constraint on our discrete scheme to solve the equation.

TABLE 1. Convergence factors for different mesh sizes. $\Delta x = 1/N_x$, $\Delta t = 0.1\Delta x$, and $\epsilon = 1$.

Case	16 x 16	32 x 32	64 x 64	128 x 128
μ_{loc}	0.669433	0.669521	0.669532	0.669534
Case	256 x 256	512 x 512	1024 x 1024	2048 x 2048
μ_{loc}	0.669534	0.669534	0.669534	0.669534

4. Numerical experiments

In this section, we validate our scheme by verifying the second-order convergence. We then perform numerical simulations of thin film propagation.

4.1. convergence test

To obtain an estimate of the rate of convergence, we perform a number of simulations for a sample initial problem on a set of increasingly finer grids. The initial condition is

$$\begin{aligned} h(x,y) &= 0.5(h_f - b)(\tanh(-y + 20 + \cos(2\pi x/40))) + b \\ &\quad +0.5(h_f - h_\infty)(\tanh(y-3)+1) + h_\infty - h_f, \end{aligned} \tag{16}$$

on a domain, $[0,40]\times[0,40]$, $h_\infty = 0.75$, $h_f = 0.75$, and $b = 0.005$. The numerical solutions are computed on the uniform grids, $\Delta x = 40/2^n$ for $n = 5,6,7,8,$ and 9. For each case, the convergence is measured at time $t = 0.1$. The uniform time steps, $\Delta t = 0.4\Delta x$, and $\epsilon = 1.0$ are used to establish the convergence rates. We define the error to be the discrete l_2-norm of the difference between that grid and the average of the next finer grid cells covering it:

$$e_{\Delta x/\frac{\Delta x}{2}\,ij} \stackrel{def}{=} h_{\Delta x ij} - \frac{h_{\frac{\Delta x}{2}2i,2j} + h_{\frac{\Delta x}{2}2i-1,2j} + h_{\frac{\Delta x}{2}2i,2j-1} + h_{\frac{\Delta x}{2}2i-1,2j-1}}{4}.$$

The rate of convergence is defined as the ratio of successive errors:

$$\log_2(\|e_{\Delta x/\frac{\Delta x}{2}}\|/\|e_{\frac{\Delta x}{2}/\frac{\Delta x}{4}}\|).$$

The errors and rates of convergence are given in table 2. The results suggest that the scheme is indeed second order accurate.

TABLE 2. l_2-norm of the errors and convergence rates.

Case	32-64	rate	64-128	rate	128-256	rate	256-512
l_2	4.9663e-3	2.05	1.1983e-3	2.02	2.595e-4	2.00	7.377e-5

4.2. Marangoni-stress-driven flow - fingering instability

We are concerned here with the slow flow of thin liquid layers on a solid or substrate. Slight perturbations of the base state, applied along the front, initiate the fingering instability. The mathematical model (Eq. (3)) will be used for numerical simulation of fingering flows. There is no well-defined perturbation of the contact line at $t = 0$; it is perturbed by noise. We model this noise by perturbing the contact line by small random numbers.

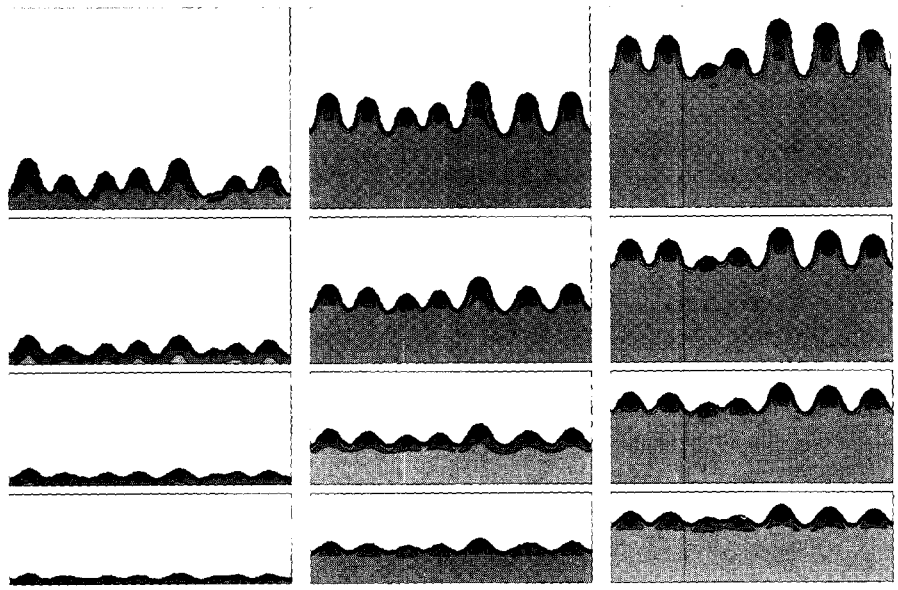


FIGURE 2. The effect of h_∞ on the finger shape for Marangoni-driven flow. (a) $h_\infty = 0.175$, (b) $h_\infty = 0.2$, and (c) $h_\infty = 0.225$. The evolution of the fluid front is from bottom to top. The times are at $t = 1250$, 1500, 1875, and 2500.

4.2.1. The effect of h_∞. The effect of h_∞ on the finger shape for Marangoni-driven flow is studied in this section. The initial condition is

$$h(x,y) = 0.5[h_\infty + b - (h_\infty - b)\tanh(3(y-7) + \mathrm{rand}(x,y))]$$

where $b = 0.002$ and $\mathrm{rand}(x,y)$ is a random number in $[-1,1]$. These perturbations model deviations from the straight front in the experiments. The computational domain using a spatial mesh of 128×256 is $\Omega = [0,200] \times [0,400]$. The uniform time step, $\Delta t = 0.5$ and $\epsilon^3 = 100$ are used. Fig. 2 shows the evolution of the fluid front with (a) $h_\infty = 0.175$, (b) $h_\infty = 0.2$, and (c) $h_\infty = 0.225$, respectively. The times are at $t = 1250$, 1500, 1875, and 2500 from bottom to top. As shown in [3], with $\epsilon = 0$ limit, the Eq. (3) has a single Lax shock solution

$$h(x,y,t) = \begin{cases} h_\infty, & y < st, \\ b, & y > st. \end{cases}$$

where the shock speed s is given by

$$s = \frac{f(h_\infty) - f(b)}{h_\infty - b} = -h_\infty^2 + (1-b)h_\infty + b - b^2. \tag{17}$$

Different front evolution speeds with three h_∞ values in Fig. 2 can be explained by Eq. (17). The shock speed is an increasing function of h_∞ until $h_\infty = \frac{1-b}{2}$.

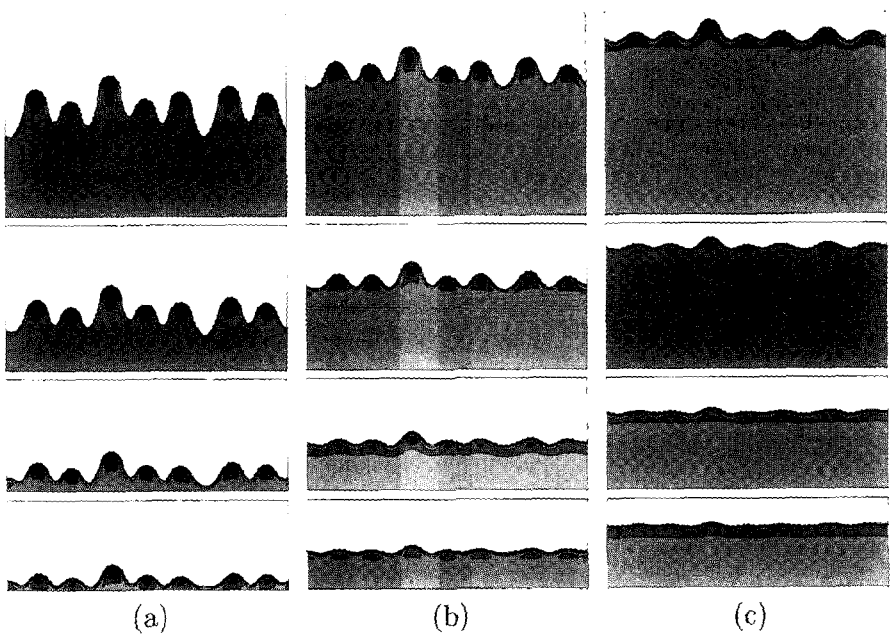


FIGURE 3. The effect of b on the finger shape for Marangoni-driven flow. (a) $b = 0.001$, (b) $b = 0.007$, and (c) $b = 0.013$. The evolution of the fluid front is from bottom to top. The times are at $t = 1500$, 1800, 2250, and 2700.

4.2.2. The effect of b. We now investigate the effect of b on the finger shape for Marangoni-driven flow. The initial condition is

$$h(x,y) = 0.5[h_\infty + b - (h_\infty - b)\tanh(3(y-7) + \mathrm{rand}(x,y))]$$

where $h_\infty = 0.2$. The computational domain using a spatial mesh of 128×256 is $\Omega = [0, 200] \times [0, 400]$. The uniform time step, $\Delta t = 1.0$ and $\epsilon^3 = 100$ are used. Fig. 3 shows the evolution of the fluid front with (a) $b = 0.001$, (b) $b = 0.007$, and (c) $b = 0.013$, respectively. The times are

at $t = 1500$, 1800, 2250, and 2700 from bottom to top. Again different front evolution speeds with three b values in Fig. 3 can be explained by Eq. (17). The shock speed is an increasing function of b until $b = \frac{1-h_\infty}{2}$.

5. Conclusions

In this study we have developed a hybrid numerical method for a mathematical model for flow in thin fluid layers including the influence of capillarity. The numerical scheme combines ENO method for treating convection term and a nonlinear multigrid method for the nonlinear high order term to solve the governing equations efficiently and accurately. The resulting scheme is robust and second order accurate in space and time. We also have demonstrated the effects of h_∞ and b on the front evolution of the thin films and found qualitative agreements with the Lax shock speed.

ACKNOWLEDGMENTS We thank John Lowengrub and Bob Behringer for intellectual and financial support.

References

[1] A. L. Bertozzi and M. P. Brenner, *Linear stability and transient growth in driven contact lines*, Phys. Fluids **9** (1997), 530–539.

[2] A. L. Bertozzi, A. Münch, X. Fanton, and A.M. Cazabat, *Contact line stability and "undercompressive shocks" in driven thin film flow*, Phys. Rev. Lett. **81** (1998), 5169–5172.

[3] A. L. Bertozzi, A. Münch, and M. Shearer, *Undercompressive shocks in thin film flows*, Phys. D **134** (1999), 431–464.

[4] C. H. Ho and Y. C. Tai, *Micro-electro-mechanical-systems (MEMS) and fluid flows*, Annu. Rev. Fluid Mech. **30** (1998), 579–612.

[5] T. G. Myers, *Thin films with high surface tension*, SIAM Rev. **40** (1998), 441–462.

[6] C. W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock capturing schems II*, J. Comput. Phys. **83** (1989), 32–78.

[7] S. M. Troian, E. Herbolzheimer, S. A. Safran, and J. F. Joanny, *Fingering instability of driven spreading films*, Europhys. Lett. **10** (1989), 25–30.

[8] U. Trottenberg, C. Oosterlee and A. Schüller, *Multigrid*, Academic press, 2001.

Junseok Kim
Department of Mathematics
University of California
103 Multipurpose Science and Technology Bldg.
Irvine, CA 92697-3875, USA
E-mail: jskim@math.uci.edu

Jeanman Sur
Department of Physics
Center for Nonlinear and Complex Systems
Duke University
Durham, North Carolina 27708, USA
E-mail: jmsur@phy.duke.edu