Contents lists available at ScienceDirect

European Journal of Mechanics B/Fluids

journal homepage: www.elsevier.com/locate/ejmflu

Numerical investigations on self-similar solutions of the nonlinear diffusion equation

Yibao Li, Junseok Kim*

Department of Mathematics, Korea University, Seoul, 136-713, Republic of Korea

HIGHLIGHTS

- We describe a numerical method for calculating self-similar solutions of this film.
- We perform several numerical tests to demonstrate that the numerical simulations are in qualitative agreement with self-similar solutions.
- Various numerical experiments are performed to show that the proposed algorithm can generate a self-similar solution.

ARTICLE INFO

Article history:
Received 1 October 2012
Received in revised form
3 April 2013
Accepted 30 May 2013
Available online 10 June 2013

Keywords:
Thin film
Nonlinear multigrid method
Self-similar solution

ABSTRACT

In this paper, we present the numerical investigations of self-similar solutions for the nonlinear diffusion equation $h_t = -(h^3 h_{xxx})_x$, which arises in the context of surface-tension-driven flow of a thin viscous liquid film. Here, $h = h(x, t)$ is the liquid film height. A self-similar solution is $h(x, t) = h(\alpha(t)(x - x_0) + x_0, t_0) = f(\alpha(t)(x - x_0))$ and $\alpha(t) = [1 - 4A(t - t_0)]^{-1/4}$, where A and x_0 are constants and t_0 is a reference time. To discretize the governing equation, we use the Crank–Nicolson finite difference method, which is second-order accurate in time and space. The resulting discrete system of equations is solved by a nonlinear multigrid method. We also present efficient and accurate numerical algorithms for calculating the constants, A, x_0, and t_0. To find a self-similar solution for the equation, we numerically solve the partial differential equation with a simple step-function-like initial condition until the solution reaches the reference time t_0. Then, we take $h(x, t_0)$ as the self-similar solution $f(x)$. Various numerical experiments are performed to show that $f(x)$ is indeed a self-similar solution.

© 2013 Elsevier Masson SAS. All rights reserved.

1. Introduction

The objective of this paper is to numerically investigate self-similar solutions for the nonlinear diffusion equation

$$h_t = -(h^3 h_{xxx})_x, \tag{1}$$

which arises in the context of thin liquid film flow. Here $h = h(x, t)$ denotes the liquid film height, x, a spatial coordinate, and t, time (see Fig. 1). h_∞ is the constant upstream height and b is the precursor film thickness. Eq. (1) can be considered as a zero gravity and no surface tension gradient limit of the following equation:

$$h_t + (h^2 - h^3)_x = -(h^3 h_{xxx})_x \tag{2}$$

which governs a thin layer of liquid on an inclined substrate driven by thermally created surface tension gradients and influenced by gravity. This equation has been extensively studied experimentally, analytically, and numerically [1–24]. A liquid film driven by a thermal gradient with a counteracting gravitational force has been studied experimentally [15–17].

Jump initial data, from a moderately thick film to a thin precursor layer, is shown to give rise to a double wave structure that includes an undercompressive wave [15]. The wave structure of solutions observed in numerical simulations with Eq. (2) is related to the hyperbolic theory of the underlying scalar conservation law, $h_t + (h^2 - h^3)_x = 0$ [18]. See [19,20] and the references therein for related mathematical problems concerning the dynamics of thin films.

Alternating direction implicit schemes are constructed for the solution of the fourth-order thin film equation for surface-tension-driven fluid flows [21]. Adaptive mesh refinement for thin film equations is developed in [22]. A detailed implementation of an adaptive finite element method was presented in [23]. In [24], the authors numerically investigated the effect of the convection term treatment using the Godunov scheme, the WENO scheme, and an upwind-type scheme of a driven thin film equation.

Bernoff and Witelski [25] studied the compactly-supported self-similar solutions of $h_t = -(h^n h_{xxx})_x$ for $0 < n < 3$. Further, using linear stability analysis, they showed that the source-type solutions are stable. For further details about the self-similar

* Corresponding author. Tel.: +82 2 3290 3077; fax: +82 2 929 8562.
E-mail address: cfdkim@korea.ac.kr (J. Kim).
URL: http://math.korea.ac.kr/~cfdkim (J. Kim).

0997-7546/$ – see front matter © 2013 Elsevier Masson SAS. All rights reserved.
http://dx.doi.org/10.1016/j.euromechflu.2013.05.003

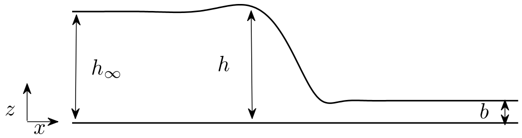

Fig. 1. Schematic diagram of thin film problem.

solutions of fourth-order nonlinear diffusion equations, we refer the reader to [25–34] and the references therein.

To find a self-similar solution of Eq. (1), we numerically solve the partial differential equation with a simple step-function like initial condition until the solution reaches a reference time. Then, we take the self-similar solution as the one at the reference time. Note that, this numerical self-similar solution is an approximation to the analytic one.

The remainder of this paper is organized as follows. In Section 2, we briefly review the governing equation and introduce a self-similar solution for the nonlinear diffusion equation. In Section 3, we present the Crank–Nicolson finite difference discretization of the governing equation and its nonlinear full-approximation storage (FAS) multigrid solver. Efficient and accurate numerical algorithms for calculating the constants, A, x_0, and t_0 are also described. In Section 4, we present various numerical results. Finally, we state our conclusions in Section 5.

2. Governing equation

We consider the dynamics of a thin layer of liquid of thickness $h = h(x, t)$ on a substrate, driven by surface tension. The configuration is shown schematically in Fig. 1. The spatial variables x and z denote the direction of flow and the film height, respectively. We model the dynamics of the thin film using lubrication approximation with a "depth averaged" velocity:

$$\bar{u}(x,t) = \frac{1}{h(x,t)} \int_0^{h(x,t)} u(x,z,t)dz = \frac{\gamma h^2(x,t) h_{xxx}(x,t)}{3\eta}, \quad (3)$$

where γ denotes the surface tension coefficient and η, the viscosity of the fluid [35]. Coupling Eq. (3) with the conservation of mass [15], we obtain

$$h_t(x,t) + [h(x,t)\bar{u}(x,t)]_x = 0. \quad (4)$$

To non-dimensionalize Eq. (4), we employ the non-dimensional variables denoted by hats, $\hat{h} = h/H$, $\hat{x} = x/L$, and $\hat{t} = t/T$; thus we obtain

$$\frac{H}{T}\hat{h}_{\hat{t}} + \left(\frac{H^4 \gamma \hat{h}^3 \hat{h}_{\hat{x}\hat{x}\hat{x}}}{3L^4 \eta}\right)_{\hat{x}} = 0, \quad (5)$$

where H, L, and T are the characteristic height, length, and time scales, respectively. Now choose the time scale $T = 3L^4\eta/(H^3\gamma)$ so that we have $\hat{h}_{\hat{t}} + \left(\hat{h}^3\hat{h}_{\hat{x}\hat{x}\hat{x}}\right)_{\hat{x}} = 0$. Drop the '$\hat{\ }$' to obtain the dimensionless thin film equation:

$$h_t = -(h^3 h_{xxx})_x \quad (6)$$

with the boundary conditions, $\lim_{x\to-\infty} h(x,t) = h_\infty$ and $\lim_{x\to\infty} h(x,t) = b$.

In this paper, we present a self-similar solution of Eq. (6) of the form

$$h(x,t) = f(\phi) \quad \text{and} \quad \phi = \alpha(t)(x - x_0), \quad x \in \mathbb{R},\ t \ge t_0, \quad (7)$$

where x_0 and t_0 are the reference points that satisfy $h_t(x_0, t) = 0$ and $\alpha(t_0) = 1$, respectively. Substitution of the similarity ansatz (7) into Eq. (6) yields the ordinary differential equations:

$$\frac{\alpha'(t)}{\alpha^5(t)} = -\frac{[f^3(\phi)f'''(\phi)]'}{\phi f'(\phi)} = A, \quad (8)$$

where the prime symbol denotes differentiation with respect to the argument variable of each function and A is a constant. From Eq. (8), $\alpha(t)$ is given as

$$\alpha(t) = [1 - 4A(t - t_0)]^{-1/4}. \quad (9)$$

Here we have used the initial condition $\alpha(t_0) = 1$. The similarity solution f should satisfy the equation

$$[f^3(\phi)f'''(\phi)]' = -A\phi f'(\phi) \quad (10)$$

subject to the boundary conditions: $\lim_{\phi\to-\infty} f(\phi) = h_\infty$ and $\lim_{\phi\to\infty} f(\phi) = b$.

In this study, we use dual approaches to calculate self-similar solutions. One is to calculate f directly from Eq. (10) by using the bvp5c program [36–39], which is an ordinary differential equation solver. A sample MATLAB code is given in Appendix A. The other approach is to solve the evolution equation (6) with an initial condition such as a step-function, and we take f as an intermediate solution at a certain time $t = t_0$.

3. Numerical method

We split the fourth-order equation (6) into a system of second order equations

$$h_t(x,t) = [M(h(x,t))\mu_x(x,t)]_x, \quad (11)$$

$$\mu(x,t) = -h_{xx}(x,t), \quad x \in \Omega = (0, L),\ t > 0, \quad (12)$$

where $M(h) = h^3$. Boundary conditions are given by

$$h(0,t) = h_\infty, \qquad h(L,t) = b, \quad (13)$$

$$\mu_x(0,t) = \mu_x(L,t) = 0, \quad (14)$$

where h_∞ is the constant upstream height and b is the precursor film thickness. The first two boundary conditions are Dirichlet boundary conditions and the last two boundary conditions are homogeneous Neumann boundary conditions, which are no-flux boundary conditions.

3.1. Discretization and numerical solver

Now, we present fully discrete schemes for Eqs. (11) and (12) in one dimensional space $\Omega = (0, L)$. Let N be a positive even integer, $\Delta x = L/N$, the uniform mesh size, and $x_i = (i - 0.5)\Delta x$, $1 \le i \le N$, the cell-center node point. Let h_i^n and μ_i^n be approximations of $h(x_i, n\Delta t)$ and $\mu(x_i, n\Delta t)$, respectively. Then, a Crank–Nicolson finite difference discretization of Eqs. (11) and (12) is given by

$$\begin{aligned}\frac{h_i^{n+1} - h_i^n}{\Delta t} &= \frac{M\left(h_{i+\frac{1}{2}}^{n+1}\right)(\mu_{i+1}^{n+1} - \mu_i^{n+1}) - M\left(h_{i-\frac{1}{2}}^{n+1}\right)(\mu_i^{n+1} - \mu_{i-1}^{n+1})}{2(\Delta x)^2} \\ &+ \frac{M\left(h_{i+\frac{1}{2}}^{n}\right)(\mu_{i+1}^{n} - \mu_i^{n}) - M\left(h_{i-\frac{1}{2}}^{n}\right)(\mu_i^{n} - \mu_{i-1}^{n})}{2(\Delta x)^2}, \end{aligned} \quad (15)$$

$$\mu_i^{n+1} = -\frac{h_{i-1}^{n+1} - 2h_i^{n+1} + h_{i+1}^{n+1}}{(\Delta x)^2}, \quad (16)$$

where $h_{i+1/2} = (h_i + h_{i+1})/2$. The boundary conditions are defined as

$$h_0 = 2h_\infty - h_1, \qquad h_{N+1} = 2b - h_N, \quad (17)$$

$$\mu_0 = \mu_1, \qquad \mu_{N+1} = \mu_N. \quad (18)$$

In this paper, we use a multigrid method [40] to solve the nonlinear discrete system (15) and (16) at the implicit time level. A detailed description of the numerical solution is provided in Appendix B. Note that by summing Eq. (15), we have the conservation property

$$\sum_{i=1}^{N} \frac{h_i^{n+1} - h_i^n}{\Delta t}$$
$$= \frac{M\left(h^{n+1}_{N+\frac{1}{2}}\right)(\mu^{n+1}_{N+1} - \mu^{n+1}_{N}) - M\left(h^{n+1}_{\frac{1}{2}}\right)(\mu^{n+1}_{1} - \mu^{n+1}_{0})}{2(\Delta x)^2}$$
$$+ \frac{M\left(h^{n}_{N+\frac{1}{2}}\right)(\mu^{n}_{N+1} - \mu^{n}_{N}) - M\left(h^{n}_{\frac{1}{2}}\right)(\mu^{n}_{1} - \mu^{n}_{0})}{2(\Delta x)^2} = 0,$$

where we applied the zero Neumann boundary condition equation (18).

3.2. Calculation of a self-similar solution

In this subsection, we describe our proposed numerical algorithm for calculating a self-similar solution by solving the evolution equation (6) with an initial condition such as a step-function.

A suitable self-similar solution $f(\phi) = h(x, t_0)$ should satisfy

(i) $\max_{x\in\Omega} h(x,t) = \max_{\phi\in\Omega} f(\phi)$ and $\min_{x\in\Omega} h(x,t) = \min_{\phi\in\Omega} f(\phi)$,

(ii) $A(t) = (\alpha^4(t) - 1)/[4(t - t_0)\alpha^4(t)]$ is constant for $t > t_0$, where t_0 is the reference time when the self-similar solution begins.

Let $h^n_{\max}$ and $h^n_{\min}$ denote the maximum and minimum values of $h(x,t)$ at $t = n\Delta t$. First, we evolve the equation with a given initial condition until the relative errors $|h^{n+1}_{\max} - h^n_{\max}|/h^n_{\max}$ and $|h^{n+1}_{\min} - h^n_{\min}|/h^n_{\min}$ are smaller than a given tolerance, tol $= 1\text{e}{-6}$. Initially, we set $t_0 = n\Delta t$ and carry out the following two steps until we get the reference time t_0.

Step 1. Compute $A(t) = (\alpha(t)^4 - 1)/[4(t - t_0)\alpha(t)^4]$ at three different times $t = t_0 + \beta\Delta t,\ t_0 + 2\beta\Delta t,$ and $t_0 + 3\beta\Delta t$. Here β is some integer and we use $\beta = 10$ in this study.

Let $x_*(t)$ be the position at time t such that $h(x_*(t), t) = \max_{x\in\Omega} h(x,t)$. Since the maximum value of $h(x,t)$ should be the same in the self-similar solution, i.e., $h(x_*(t), t) = f(\phi_*) = h(x_*(t_0), t_0)$, from Eq. (7) we have

$$\alpha(t) = (x_*(t_0) - x_0)/(x_*(t) - x_0). \tag{19}$$

Let $h^{n_0}_k = \max_{1\le i\le N} h^{n_0}_i$ and we define the quadratic polynomial approximation passing through the three points, $(x_{k-1}, h^{n_0}_{k-1})$, $(x_k, h^{n_0}_k)$, and $(x_{k+1}, h^{n_0}_{k+1})$. Then define $x_*(t_0)$ as the critical point of the polynomial and $x_*(t)$ is defined similarly.

Next, we calculate x_0 which satisfies $h_t(x_0, t) = 0$ and $h_x(x_0, t) \neq 0$ for all $t \ge t_0$. Choose $t_1 = n_1\Delta t > t_0 = n_0\Delta t$; then, $h(x_0, t_1) = h(x_0, t_0)$ (see Fig. 2). We can find the unique index k such that $(h^{n_0}_{k+1} - h^{n_1}_{k+1})(h^{n_0}_k - h^{n_1}_k) \le 0$ and $b + 0.1(h_\infty - b) < h^{n_0}_k < b + 0.9(h_\infty - b)$. We define the point x_0 as the x-coordinate of the intersection point of two line segments. One line connects the points $(x_k, h^{n_0}_k)$ and $(x_{k+1}, h^{n_0}_{k+1})$, whereas the other passes through the points $(x_k, h^{n_1}_k)$ and $(x_{k+1}, h^{n_1}_{k+1})$. Then, x_0 is defined as

$$x_0 = x_k + (x_{k+1} - x_k)(h^{n_0}_k - h^{n_1}_k)/(h^{n_0}_k - h^{n_0}_{k+1} - h^{n_1}_k + h^{n_1}_{k+1}).$$

Using Eq. (19), we can calculate $A(t) = (\alpha^4(t) - 1)/[4(t - t_0)\alpha^4(t)]$.

Step 2. If $\max\left\{\left|\frac{A(t_0+2\beta\Delta t) - A(t_0+\beta\Delta t)}{A(t_0+\beta\Delta t)}\right|, \left|\frac{A(t_0+3\beta\Delta t) - A(t_0+2\beta\Delta t)}{A(t_0+2\beta\Delta t)}\right|\right\} <$ tol, we set $t_0 = n\Delta t$. Otherwise, we go back to *Step* 1 with $t_0 = t_0 + \beta\Delta t$.

After this algorithm, we get t_0, x_0, A, and $f(\phi)$.

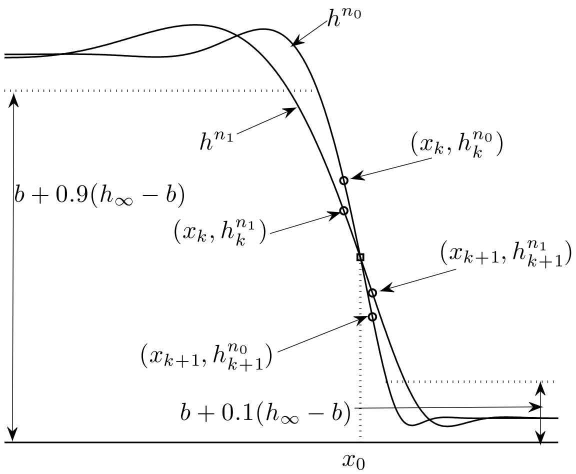

Fig. 2. Schematic illustration of finding x_0.

We note that unless the initial condition is the similarity solution and the domain is infinite, the numerical solution will always have small deviations from the similarity solution. The similarity solution will be approached as time increases (but not converged to in finite time) on the infinite domain. If the width of the domain is finite, then the boundary conditions will eventually override this approach to the similarity solution and the numerical solution converges to the steady-state solution (also only in the limit of infinite time).

4. Numerical experiments

In this section, we describe various numerical experiments, such as a convergence test, as well as other experiments to demonstrate the finite computational domain effect, long time evolution, a numerical self-similar solution, and the effect of parameters and initial profiles on self-similarity. Unless otherwise specified, we use the computational domain $\Omega = (0, 100)$ with an $N = 1024$ mesh grid and a time step $\Delta t = 1$.

4.1. Convergence test

We start with spatial and temporal convergence tests of the numerical scheme. In order to obtain the spatial convergence rate, we perform a number of simulations with increasingly finer grids $\Delta x = 100/2^{n-1}$ for $n = 6, 7, 8, 9,$ and 10. The initial condition is

$$h(x, 0) = 0.5[h_\infty + b - (h_\infty - b)\tanh(3(x - 50))], \tag{20}$$

where $h_\infty = 0.3$ and $b = 0.1$. Numerical solutions are computed up to time $T = 100$ with the time step $\Delta t = \Delta x$. We define the error of the numerical solution on a grid as the discrete l_2-norm of the difference between that grid and the average of the next finer grid numerical solution: $e_{\Delta x/\frac{\Delta x}{2}\,i} := h_{\Delta x i} - (h_{\frac{\Delta x}{2}2i-1} + h_{\frac{\Delta x}{2}2i})/2$. The rate of convergence is defined as the ratio of successive errors: $\log_2(\|e_{\Delta x/\frac{\Delta x}{2}}\|_2/\|e_{\frac{\Delta x}{2}/\frac{\Delta x}{4}}\|_2)$. Here $\|e\|_2^2$ is a discrete l_2 norm and is defined as $\|e\|_2^2 = \sum_{i=1}^{N} e_i^2/N$. Fig. 3 shows a log–log plot of l_2-norm of errors (circle) against various mesh grids at $T = 100$ with a linear fitting (solid line). The second-order accuracy with respect to space and time is observed as expected from the discretization.

Furthermore, we consider the CPU time in seconds for the convergence test. Tests were performed on a system with a 3-GHz Intel Pentium CPU and 3-GB RAM, loaded with C++. If we refine the spatial and temporal grids by a factor of 2, the CPU time should increase by a factor of 4. As can be observed from Table 1, using the multigrid method, the CPU time increases by a factor of approximately four. This result indicates that the computational complexity of the multigrid method is indeed $O(N)$.

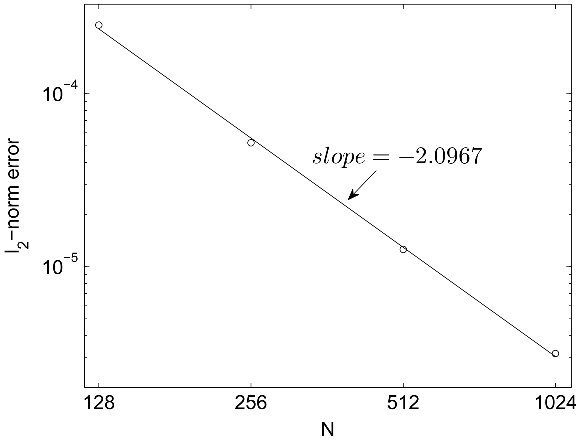

Fig. 3. Log–log plot of l_2-norm of errors (circle) against various mesh grids at $T = 100$ with a linear fitting (solid line).

Table 1
CPU time in seconds.

Grid	256		512		1024		2048
CPU time	5.787		28.360		114.563		442.828
Factor		4.90		4.04		3.87	

4.2. Evolution of thin liquid film

In this subsection, we show the evolution of the thin liquid film and its self-similar solution. The initial condition is the same as in Eq. (20). Solutions are computed up to time $T = 5000$. Fig. 4(a) shows the evolutions of the thin film height h. We can observe that there exits a point $x_0 = 50.02$, which satisfies $h_t(x_0, t) = 0$ and $h_x(x_0, t) \neq 0$ for all $t \geq t_0 = 891$ with $A = -1.81\text{e}{-4}$. In Fig. 4(b), the starred and circled lines represent the evolutions of $h^n_{\max} - h_\infty$ and $b - h^n_{\min}$, respectively. They start from zero and converge quickly to constant values. Fig. 4(c) shows snapshots of the thin film shapes at t_0 and $t = 5000$. A comparison of the numerical solutions at $t = 5000$ and from a self-similar solution at t_0 is shown in Fig. 4(d). This result shows that our numerical algorithm can generate a self-similar solution.

4.3. Effect of the finite computational domain size

The original equation is in the infinite domain; however, to get numerical solutions by using a finite difference method, the domain must be truncated and suitable boundary conditions should be applied. In order to show the effect of the computational domain size on the numerical solutions, we take a set of different domains $\Omega = (-25, 25)$, $(-50, 50)$, and $(-100, 100)$, with a fixed space step size $\Delta x = 50/512$. The initial condition is $h(x, 0) = 0.5[h_\infty + b - (h_\infty - b)\tanh(3x)]$, where $h_\infty = 0.3$ and $b = 0.1$. We run the computation up to $T = 2000$. Fig. 5 shows a comparison of the thin liquid film profiles in three different computational domains. This result suggests that the effect of the domain size on the numerical result is negligible as long as the computational domain is sufficiently large and the temporal evolution is not too long.

Next, we perform a long time evolution. The initial condition is $h(x, 0) = h_\infty$, if $x < 0$ and $h(x, 0) = b$, otherwise, where $h_\infty = 0.3$ and $b = 0.1$. Fig. 6(a) shows the evolution of the thin film in the domain $\Omega = (-25, 25)$ up to time $T = 8000000$. As can be seen, the numerical solution becomes a steady state (thick solid line) that is nearly linear profile owing to the boundary conditions. The symbol 'o' indicates the self-similar solution, $f(\phi)$, from the ODE solver. We can observe a good agreement between $f(\phi)$ and one of evolution profiles. In Fig. 6(b), we show the discrete l_2-norm of the difference between the numerical solution and the self-similar solution (circled line in Fig. 6(a)) obtained by solving Eq. (10) with

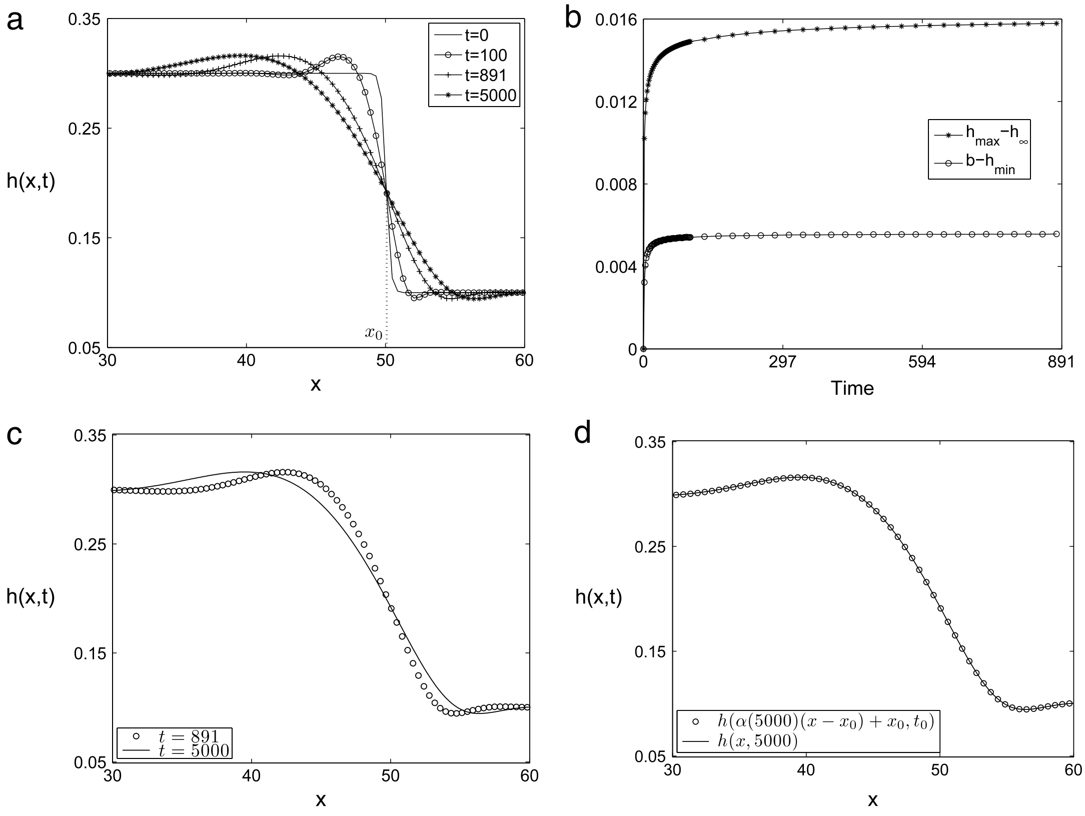

Fig. 4. (a) Evolution of the thin film height h with $h_\infty = 0.3$ and $b = 0.1$. (b) The starred and circled lines represent evolutions of $h_{\max} - h_\infty$ and $b - h_{\min}$, respectively. (c) Snapshots of the thin film shapes at $t_0 = 891$ and $t = 5000$. (d) Profiles of $h(x, 5000)$ and $h(\alpha(5000)(x - x_0) + x_0, t_0)$.

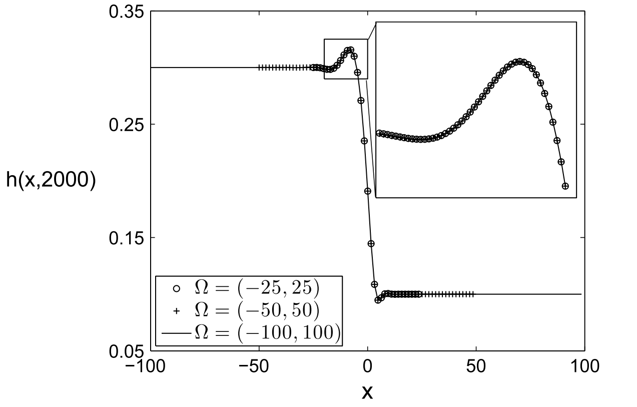

Fig. 5. Comparison of the thin liquid film profiles in three different computational domains at $T = 2000$.

Table 2
Comparison of A with different b and h_∞. Here, $t_0 = 2000$.

b	$h_\infty = 0.2$	$h_\infty = 0.3$	$h_\infty = 0.4$
0.05	−1.23e−4	−1.24e−4	−1.24e−4
0.1	−1.22e−4	−1.25e−4	−1.26e−4
0.15	−1.23e−4	−1.26e−4	−1.26e−4

Table 3
Comparison of A with different t_0. Here, $b = 0.1$ and $h_\infty = 0.3$.

t_0	2000	2400	2800	3200
A	−1.25e−4	−1.04e−4	−8.67e−5	−7.86e−5

$A = -1.81\text{e}-4$. The result shows that the error starts at a moderate initial value, decreases for a while, then increases for longer times.

4.4. *Effect of parameters, h_∞, b, and t_0*

In this subsection, we study the effect of the parameters, h_∞, b, and t_0 on self-similarity. The initial condition is the same as in Eq. (20) with different h_∞ and b values. Here, we take $t_0 = 2000$. With a set of numerical solutions of the thin film at $t = 2200, 2400, \ldots, 4000$, the averaged values of A are calculated and listed in Table 2. From the results, we observe that although different values of h_∞ and b are used, the value of A is nearly the same. However, with different t_0 values, the value of A is different (see Table 3). Here $h_\infty = 0.3$, $b = 0.1$, $N = 2048$, and $\Delta t = 0.5$ are used.

4.5. *Simulation with another initial condition*

To demonstrate the independence of the initial profiles on generating a self-similar solution, we perform a numerical test with another initial condition

$$h(x,0) = \begin{cases} 0.5[h_\infty + h_f - (h_\infty - h_f)\tanh(3(x-45))] \\ \quad \text{if } x < 48 \\ 0.5[h_\infty + b - (h_\infty - b)\tanh(3(x-50))] \\ \quad \text{otherwise.} \end{cases}$$

Here $h_\infty = 0.3$, $h_f = 0.35$, and $b = 0.1$ are used. Solutions are computed up to time $T = 20\,000$. Fig. 7(a) shows the temporal evolution of the numerical solution. Fig. 7(b) shows snapshots of the thin film shapes at $t_0 = 9542$ and $t = 20\,000$. Fig. 7(c) shows profiles of $h(x, 20\,000)$ and $h(\alpha(20\,000)(x - x_0) + x_0, t_0)$, where $\alpha(20\,000) = 1.201$ and $x_0 = 50.13$. The results indicate that the numerical self-similar solution is in good agreement. Further, the self-similar solution is independent of initial profiles.

5. Conclusions

In this article, we numerically investigated the self-similar solutions for the nonlinear diffusion equation $h_t = -(h^3 h_{xxx})_x$, which arises in the context of surface-tension driven flow of thin viscous liquid film. The suggested self-similar solution is $h(x,t) = h(\alpha(t)(x - x_0) + x_0, t_0) = f(\alpha(t)(x - x_0))$ and $\alpha(t) = [1 - 4A(t - t_0)]^{-1/4}$, where A and x_0 are constants and t_0 is a reference time. To numerically solve the governing equation, we used the Crank–Nicolson finite difference method with a nonlinear multigrid solver. We provided numerical algorithms for calculating the constants, A, x_0, and t_0 in detail. To find a self-similar solution of the equation, we numerically solved the partial differential equation with a simple step-function like initial condition until the solution reached the reference time t_0. Then, we took $h(x, t_0)$ as the self-similar solution $f(x)$. Various numerical experiments were performed to demonstrate that $f(x)$ is indeed a self-similar solution. In particular, we found a self-similar solution could be obtained from arbitrary initial profiles.

Acknowledgments

The corresponding author (J. Kim) was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2011-0023794). The authors thank the help of Dr. Youngsoo Ha and Professor Tim Myers in using the ODE solver, bvp5c. The authors greatly appreciate the reviewers for their constructive and insightful comments on this article.

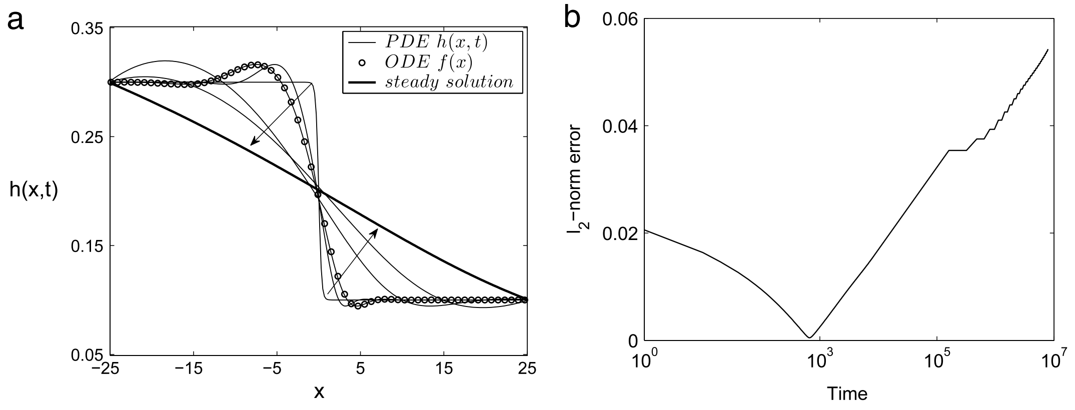

Fig. 6. (a) Evolution of thin film on a domain $\Omega = (-25, 25)$ until time $T = 8\,000\,000$, solid lines. The symbol 'o' indicates the self-similar solution, $f(x)$, from the ODE solver. (b) Discrete l_2-norm of the difference between the numerical solution and the self-similar solution (circled line in Fig. 6(a)) obtained by solving Eq. (10) with $A = -1.81\text{e}-4$.

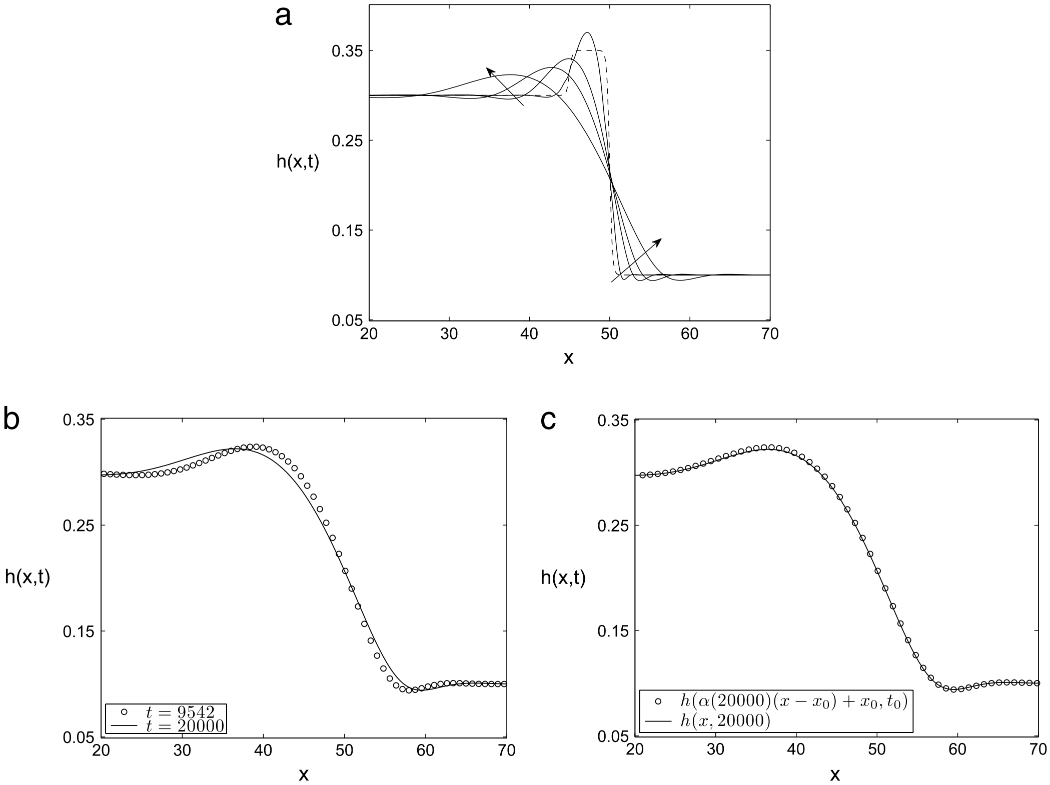

Fig. 7. Simulation with another initial condition. (a) Temporal evolution of the numerical solution. (b) Snapshots of the thin film shapes at $t_0 = 9542$ and $t = 20\,000$. (c) Profiles of $h(x, 20\,000)$ and $h(\alpha(20\,000)(x-x_0)+x_0, t_0)$.

Appendix A

A MATLAB m-file, which calculates the self-similar solution by solving Eq. (10).

```
function selfODE       % This solves f''''= -(Axf'+3f'f'''f^2)/f^3
global heq b A
b=0.1; heq=0.3; A=-1.81e-4; x=linspace(-25,25,3000);
sol=bvp5c(@twoode,@twobc,bvpinit(x,@mat4init)); y=deval(sol,x); plot(x,y(1,:),'k-');
function yinit = mat4init(x)       % initial guess
global heq b
m=0.01; yinit = [0.5*(heq+b)-0.5*(heq-b)*tanh(m*x);  -0.5*(heq-b)*m*sech(m*x)^2
                (heq-b)*m*m*tanh(m*x)*sech(m*x)^2
                (heq-b)*m*m*m*sech(m*x)^2-2*(heq-b)*m*m*m*tanh(m*x)*sech(m*x)^2];
function dydx = twoode(x,y)       % ODEs
global A
rhs=-(A*x*y(2)+3*y(1)^2*y(2)*y(4))/y(1)^3; dydx = [y(2);y(3);y(4);rhs];
function res = twobc(ya,yb)       % Boundary condition
global heq b
res = [ya(1)-heq; yb(1)-b; ya(2); yb(2)];
```

Appendix B

To solve the discrete system (15) and (16), we use a nonlinear full approximation storage (FAS) multigrid method. Let us rewrite Eqs. (15) and (16) as

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

where the left- and right-hand side equations contain $n+1$ and n time level terms, respectively. Let us assume that a sequence of grids Ω_k (Ω_{k-1} is coarser than Ω_k by a factor of 2).

FAS multigrid cycle

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

where ν is the number of smoothing relaxation sweeps. h_k^m and h_k^{m+1} are the approximations of h_k^{n+1} before and after an FAS cycle.

(1) *Presmoothing*

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

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

which implies performing ν smoothing steps with the initial approximations h_k^m, μ_k^m, source terms ϕ_k^n, ψ_k^n, and SMOOTH relaxation operator to get the approximations $\bar{h}_k^m, \bar{\mu}_k^m$. One SMOOTH relaxation operator step consists of solving the system (21) and (22) given below by 2×2 matrix inversion for each i.

$$\frac{\bar{h}_i^m}{\Delta t} + \frac{M^m_{i+\frac{1}{2}} + \bar{M}^m_{i-\frac{1}{2}}}{2(\Delta x)^2}\bar{\mu}_i^m = \phi_i^n + \frac{M^m_{i+\frac{1}{2}}\mu^m_{i+1} + \bar{M}^m_{i-\frac{1}{2}}\bar{\mu}^m_{i-1}}{2(\Delta x)^2}, \tag{21}$$

$$-\frac{2\bar{h}_i^m}{(\Delta x)^2} + \bar{\mu}_i^m = \psi_i^n - \frac{h^m_{i+1} + \bar{h}^m_{i-1}}{(\Delta x)^2}, \tag{22}$$

where $M^m_{i+\frac{1}{2}} = M(h^m_{i+\frac{1}{2}})$ and the other terms are similarly defined.

(2) *Compute the defect*: $(\bar{d}_{1k}^{\,m}, \bar{d}_{2k}^{\,m}) = (\phi_k^n, \psi_k^n) - N_k(\bar{h}_k^m, \bar{\mu}_k^m)$.

(3) *Restrict the defect and* $\{\bar{h}_k^m, \bar{\mu}_k^m\}$

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

(4) *Compute the right-hand side*

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

(5) *Compute an approximate solution* $\{\hat{h}_{k-1}^m,\ \hat{\mu}_{k-1}^m\}$ *of the coarse grid equation on* Ω_{k-1}, *i.e.*,

$$N_{k-1}(h_{k-1}^m, \mu_{k-1}^m) = (\phi_{k-1}^n, \psi_{k-1}^n). \tag{23}$$

If $k = 1$, we explicitly invert a 2×2 matrix to obtain the solution. If $k > 1$, we solve (23) by performing an FAS k-grid cycle using $\{\bar{h}^m_{k-1}, \ \bar{\mu}^m_{k-1}\}$ as an initial approximation:

$$\{\hat{h}^m_{k-1}, \hat{\mu}^m_{k-1}\} = \text{FAScycle}(k-1, \bar{h}^m_{k-1}, \bar{\mu}^m_{k-1}, N_{k-1}, \phi^n_{k-1}, \psi^n_{k-1}, \nu).$$

(6) *Compute the coarse grid correction (CGC)*:

$$\hat{v}^m_{1k-1} = \hat{h}^m_{k-1} - \bar{h}^m_{k-1}, \qquad \hat{v}^m_{2k-1} = \hat{\mu}^m_{k-1} - \bar{\mu}^m_{k-1}.$$

(7) *Interpolate the correction*: $\hat{v}^m_{1k} = I^k_{k-1}\hat{v}^m_{1k-1}, \ \hat{v}^m_{2k} = I^k_{k-1}\hat{v}^m_{2k-1}$.

(8) *Compute the corrected approximation on* Ω_k

$$\bar{h}^{m+1}_k = \bar{h}^m_k + \hat{v}^m_{1k}, \qquad \bar{\mu}^{m+1}_k = \bar{\mu}^m_k + \hat{v}^m_{2k}.$$

(9) *Postsmoothing*

$$\{h^{m+1}_k, \mu^{m+1}_k\} = \text{SMOOTH}^{\nu}(\bar{h}^{m+1}_k, \bar{\mu}^{m+1}_k, N_k, \phi^n_k, \psi^n_k).$$

This completes the description of a nonlinear FAS cycle. See the reference text [40] for additional details and background.

References

[1] M. Amaouche, A. Djema, H.A. Abderrahmane, Film flow for power-law fluids: modeling and linear stability, Eur. J. Mech. B Fluids 34 (2012) 70–84.

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

[3] J. Becker, G. Grüun, The thin-film equation: recent advances and some new perspectives, J. Phys.: Condens. Matter 17 (2005) 291–307.

[4] J. Becker, G. Grüun, M. Lenz, M. Rumpf, Numerical methods for fourth order nonlinear diffusion problems, Appl. Math. 47 (2002) 517–543.

[5] V.A. Galaktionov, J.R. King, Source-type solutions of the fourth-order unstable thin film equation, European J. Appl. Math. 18 (2007) 273–321.

[6] G. Grün, M. Lenz, M. Rumpf, A finite volume scheme for surfactant driven thin film flow, in: Finite Volumes for Complex Applications III, Hermes Penton Sciences, 2002, pp. 567–574.

[7] G. Grün, M. Rumpf, Nonnegativity preserving convergent schemes for the thin film equation, Numer. Math. 87 (2000) 113–152.

[8] G. Grün, M. Rumpf, Simulation of singularities and instabilities arising in thin film flow, European J. Appl. Math. 12 (2001) 293–320.

[9] H. Jiang, W.M. Ni, On steady states of van der Waals force driven thin film equations, European J. Appl. Math. 18 (2007) 153–180.

[10] D.E. Kataoka, S.M. Troian, A theoretical study of instabilities at the advancing front of thermally driven coating films, J. Colloid Interface Sci. 192 (1997) 350–362.

[11] J. Liang, R. Huang, J.H. Prevost, Z. Suo, Evolving crack patterns in thin films with the extended finite element method, Internat. J. Numer. Methods Engrg. 67 (2003) 1119–1136.

[12] J. Sung, H.G. Choi, J.Y. Yoo, Finite element simulation of thin liquid film flow and heat transfer including a hydraulic jump, Internat. J. Numer. Methods Engrg. 46 (1999) 83–101.

[13] S. Veremieiev, H.M. Thompson, Y.C. Lee, P.H. Gaskell, Inertial thin film flow on planar surfaces featuring topography, Comput. Fluids 39 (2010) 431–450.

[14] Y. Li, H.G. Lee, W. Hwang, D. Yoon, S. Shin, Y. Ha, J. Kim, Numerical studies of the fingering phenomena for the thin film equation, Internat. J. Numer. Methods Fluids 67 (2011) 1358–1372.

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

[16] X. Fanton, A.M. Cazabat, Q. Quéré, Thickness and shape of films driven by a Marangoni flow, Langmuir 12 (1996) 5875–5880.

[17] J. Sur, A.L. Bertozzi, R.P. Behringer, Reverse undercompressive shock structures in driven thin film flow, Phys. Rev. Lett. 90 (2003) 126105.

[18] R. Levy, M. Shearer, Kinetics and nucleation for driven thin film flow, Physica D 209 (2005) 145–163.

[19] A.L. Bertozzi, The mathematics of moving contact lines in thin liquid films, Notices Amer. Math. Soc. 45 (1998) 697–698.

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

[21] T.P. Witelski, M. Bowen, ADI schemes for higher-order nonlinear diffusion equations, Appl. Numer. Math. 45 (2003) 331–351.

[22] J. Kim, Adaptive mesh refinement for thin-film equations, J. Korean Phys. Soc. 49 (2006) 1903–1907.

[23] P. Sun, R.D. Russel, J. Xu, A new adaptive local mesh refinement algorithm and its application on fourth order thin film flow problem, J. Comput. Phys. 224 (2007) 1021–1048.

[24] Y. Ha, Y.-J. Kim, T.G. Myers, On the numerical solution of a driven thin film equation, J. Comput. Phys. 227 (2008) 7246–7263.

[25] A.J. Bernoff, T.P. Witelski, Linear stability of source-type similarity solutions of the thin film equation, Appl. Math. Lett. 15 (2002) 599–606.

[26] T.P. Witelski, A.J. Bernoff, Stability of self-similar solutions for van der Waals driven thin film rupture, Phys. Fluids 11 (1999) 2443–2445.

[27] J.A. Carrillo, G. Toscani, Long-time asymptotics for strong solutions of the thin film equation, Comm. Math. Phys. 225 (2002) 551–571.

[28] F.D. Richard, E.J. Hinch, R.L. John, Self-similar capillary pinchoff of an inviscid fluid, Phys. Rev. Lett. 80 (1998) 704–707.

[29] E. Magyari, B. Keller, Exact solutions for self-similar boundary-layer flows induced by permeable stretching walls, Eur. J. Mech. B Fluids 19 (2000) 109–122.

[30] J.S. Mathunjwa, A.J. Hogg, Self-similar gravity currents in porous media: linear stability of the Barenblatt–Pattle solution revisited, Eur. J. Mech. B Fluids 25 (2006) 360–378.

[31] E. Momoniat, C. Harley, E. Adlem, Numerical investigation of the generalized lubrication equation, Appl. Math. Comput. 217 (2010) 2631–2638.

[32] H. Steiner, T. Hirschler, A self-similar solution of a shock propagation in a dusty gas, Eur. J. Mech. B Fluids 21 (2002) 371–380.

[33] M. Xu, S.L. Zhou, Stability and regularity of weak solutions for a generalized thin film equation, J. Math. Anal. Appl. 337 (2008) 49–60.

[34] W.W. Zhang, J.R. Lister, Similarity solutions for van der Waals rupture of a thin film on a solid substrate, Phys. Fluids 11 (1999) 2454–2462.

[35] R. Levy, Partial differential equations of thin liquid films: analysis and numerical simulation, Ph.D. Thesis, North Carolina State University, Raleigh, NC, 2005.

[36] J.D. Evans, V.A. Galaktionov, J.R. King, Unstable sixth-order thin film equation: II. Global similarity patterns, Nonlinearity 20 (2007) 1843–1881.

[37] J.D. Evans, V.A. Galaktionov, J.R. King, Blow-up similarity solutions of the fourth order unstable thin film equation, European J. Appl. Math. 34 (2007) 195–231.

[38] J. Kierzenka, L.F. Shampine, A BVP solver based on residual control and the Maltab PSE, ACM Trans. Math. Software 27 (2001) 299–316.

[39] J. Kierzenka, L.F. Shampine, A BVP solver that controls residual and error, JNAIAM J. Numer. Anal. Ind. Appl. Math. 3 (2008) 27–41.

[40] U. Trottenberg, C. Oosterlee, A. Schüller, Multigrid, Academic Press, London, 2001.