Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 224 (2007) 1021–1048
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A new adaptive local mesh refinement algorithm and its application on fourth order thin film flow problem

Pengtao Sun [b,*], Robert D. Russell [a], Jinchao Xu [b]

[a] Department of Mathematics, Simon Fraser University, USA
[b] Department of Mathematics, Pennsylvania State University, USA

Received 23 February 2006; received in revised form 26 September 2006; accepted 4 November 2006
Available online 20 December 2006

Abstract

A new adaptive local mesh refinement method is presented for thin film flow problems containing moving contact lines. Based on adaptation on an optimal interpolation error estimate in the L^p norm ($1 < p \leqslant \infty$) [L. Chen, P. Sun, J. Xu, Multilevel homotopic adaptive finite element methods for convection dominated problems, in: Domain Decomposition Methods in Science and Engineering, Lecture Notes in Computational Science and Engineering 40 (2004) 459–468], we obtain the optimal anisotropic adaptive meshes in terms of the Hessian matrix of the numerical solution. Such an anisotropic mesh is optimal for anisotropic solutions like the solution of thin film equations on moving contact lines. Thin film flow is described by an important type of nonlinear degenerate fourth order parabolic PDE. In this paper, we address the algorithms and implementation of the new adaptive finite element method for solving such fourth order thin film equations. By means of the resulting algorithm, we are able to capture and resolve the moving contact lines very precisely and efficiently without using any regularization method, even for the extreme degenerate cases, but with fewer grid points and degrees of freedom in contrast to methods on a fixed mesh. As well, we compare the method theoretically and computationally to the positivity-preserving finite difference scheme on a fixed uniform mesh which has proven useful for solving the thin film problem.

Keywords: Adaptive mesh; Local refinement; Mixed finite element method; Thin film flow; Moving contact lines

1. Introduction

Adaptive procedures for the numerical solution of partial differential equations (PDEs), actively investigated since the late 1970s, are now standard tools in science and engineering – e.g. see [71] for references on adaptivity for elliptic PDEs. Adaptive finite element methods (FEMs) are a particularly meaningful approach for handling multiscale phenomena and making realistic computations feasible, especially on

* Corresponding author. Address: Department of Mathematics, Pennsylvania State University, University Park, State College, PA 16802, USA.
E-mail addresses: sun@math.psu.edu (P. Sun), rdr@math.sfu.ca (R.D. Russell), xu@math.psu.edu (J. Xu).

0021-9991/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2006.11.005

irregular domains and in higher spatial dimensions with complex boundary conditions, where *a posteriori error estimators* are available as an essential ingredient of adaptivity. Such estimators are computable quantities depending on the computed solution(s) and data which provide information about the quality of approximation and may thus be used to make judicious mesh modifications. The ultimate purpose is to construct a sequence of meshes which will eventually equidistribute the approximation errors and, as a consequence, the computational effort. To this end, the a posteriori error estimators are split into element indicators which are then employed to make local mesh modifications by refinement (and sometimes coarsening). This naturally leads to loops of the form

$$\text{Solve} \rightarrow \text{Estimate} \rightarrow \text{Refine}. \tag{1.1}$$

Starting from a coarse mesh, such an iteration has been widely successful in applications. Nevertheless, except for the rather complete description of the one-dimensional situation by Babuška and Rheinboldt [3], convergence of (1.1) in the multidimensional case is still largely an open issue. The fundamental paper [38] of Dörfler for the Poisson equation shows a linear error reduction rate for the energy norm towards a preassigned tolerance in finite steps. Recently Nochetto et al. [60] have constructed an adaptive FEM algorithm for elliptic PDEs and proved its linear convergence rate for the energy norm. Any prescribed error tolerance is thus achieved in a finite number of steps. It is in this context that there are some of the best current convergence results for adaptive finite element methods. All of the above a posteriori error estimators fall into a class of residual-type methods because they are all based on residual error on each element.

Besides the above developments for a posterior error estimators' convergence analysis, there is an alternative approach to produce a similar effect, viz., constructing (nearly) optimal meshes for suitable order piecewise finite element interpolation of a given function through a priori interpolation error estimators. Specifically, let $\Omega \in \mathscr{R}^n$ be a bounded domain, T a simplicial finite element mesh of $\overline{\Omega}$ with a fixed number N of elements, and u_I a piecewise finite element interpolation of a given function u defined on $\overline{\Omega}$. An optimal mesh could be obtained by minimizing the error $|||u - u_I|||$ in some sense, where the norm $|||\cdot|||$ is a classical *Sobolev* space norm.

This approach can be traced back to de Boor [34,33] where the problem of the best approximation by free knots splines was studied in one spatial dimension. In this work, the equidistribution principle was introduced specifically for computing equidistributed meshes. Actually the concept of equidistribution was first used by Burchard [21], and then by a number of researchers for studying grid adaptation. Pioneering work for adaptive finite element methods was done in [3] where a finite element mesh was shown nearly optimal in the sense of minimizing the H^1 norm error if the local errors are approximately equal for all elements. Thus, to get an optimal mesh, elements where the error is large are marked for refinement, while elements with a small error are left unchanged or coarsened.

Most of the adaptive finite element methods in the literature [4] are concerned with meshes that are shape-regular (which in two dimensions means that no element has a very small angle). This type of shape-regular finite elements is appropriate for physical problems that are fairly isotropic. But for many anisotropic problems (e.g. with sharp boundary layers or internal layers), the shape of elements can be further optimized, and an equidistribution of a scalar error density is not sufficient to ensure that a mesh is optimally efficient [32]. Nadler [62] studied the optimal triangulation for the discontinuous piecewise linear approximation for a quadratic function in the sense of minimizing error in the L^2 norm. For this optimal mesh, each triangle is equilateral under the Hessian metric, and the error is equidistributed on each triangle. The L^∞ case was studied by D'Azevedo and Simpson [30,31]. In [31] they developed a local linear interpolation error formula for quadratic functions. Based on this formula, D'Azevedo [30] obtained the same condition as Nadler's. Thereafter anisotropic mesh adaptation which aims to generate equilateral triangles under the metric induced by Hessian matrix was developed in [20,48,37] and successfully applied to computational fluid dynamic problems in two spatial dimensions [48,70].

Recently there have been some a priori interpolation error estimates for anisotropic finite elements [2,56,42]. Apel [2] obtained some estimates under a condition on the coordinate orientation and on the maximal allowable mesh angle. Formaggia and Perotto [42] exploited the spectral properties of the affine map from the reference triangle to the general triangle to get anisotropic estimates for the L^2 and H^1 interpolation error on linear finite elements in two dimensions. Kunert [56] introduced the matching function to measure the

alignment of an anisotropic function and an anisotropic mesh and presented error estimates using the matching function. Yet the overall optimal convergence rate in terms of the number of degree of freedom is not easy to get from these approaches.

Motivated by the fundamental papers of Huang [49,50], a general result which develops this approach of a priori interpolation error estimators further is given by Chen et al. [28]. In this paper, a new local edge-based error estimate and global interpolation error bound in the L^p $(1 \leqslant p \leqslant \infty)$ norm for any spatial dimension are given. Specifically,

$$\|u - u_I\|_{L^p(\Omega)} \leqslant CN^{-2/n} \| \sqrt[n]{\det(H)} \|_{L^{\frac{pn}{2p+n}}(\Omega)}, \tag{1.2}$$

where N is the number of elements in the triangulation, the constant C is independent of u and N, and H is a majorizing Hessian matrix of u. This estimate is optimal in the sense that it is a lower bound if u is strictly convex or concave.

This estimate can be viewed as a modification and generalization of the special case $p = \infty$ in D'Azevedo [30] and the case $p = 2$ in Huang and Sun [50]. By requiring the mesh to be quasi-uniform under the new metric $[\det(H)]^{-1/(2p+n)}H$, the simplices become locally isotropic and their volumes are globally equidistributed under the new metric. On the other hand, the same simplices may simultaneously transform back to be locally anisotropic under the standard Euclidean metric if the majorizing Hessian matrix is anisotropic, or more specifically, if its eigenvalues differ markedly from each other.

The estimate (1.2) is the theoretical foundation of our method, as the new adaptive local mesh refinement algorithm aims at minimizing (or at least reducing) this interpolation error by iteratively modifying the grids. Then the corresponding generated adaptive mesh is able to anisotropically concentrate on the sharp interface or boundary layer so that the solution singularities are fully resolved. The thin film fluid flow problem is exactly such a case where the solution has anisotropic sharp internal layers.

While some numerical results are given in [27,28], no algorithm discussing how to adequately implement the adaptive mesh method based on the new interpolation error estimator (1.2) is included. In this paper, we describe a fairly detailed implementation for producing anisotropic adaptive locally refined meshes and the techniques of local mesh improvement which aim at minimizing the interpolation error. We adopt the thin film fluid flow problem, being typical of many realistic problems, to test our new adaptive finite element method.

Thin film flow problems, which lead to fourth order nonlinear degenerate parabolic equations [45,19,12,53], represent just one type of interesting fourth order problems which have been of considerable recent interest and for which adaptive numerical methods need to be developed and analyzed. One reason we study them for our application is that not only do their solutions include natural moving contact lines – one kind of internal layers which need to be fully resolved in order to get accurate numerical solutions – but also because the solution is relatively simply distributed, being almost constant except around the moving contact lines. We can then use a local refined mesh only near contact lines and coarse meshes elsewhere. In contrast, fixed uniform meshes can be unacceptable in higher spatial dimensions because of the relatively large spatial scale as compared to the slim contact line regions and the long time scale (starting from the initial data to rupture and afterwards). Our adaptive local anisotropic mesh refinement algorithm based on minimizing the interpolation error (1.2) is able to efficiently overcome these difficulties and accurately solve the problem.

The paper is organized as follows. In Section 2 we present an interpolation error estimate and show that it is also a lower bound for convex or concave functions. In Section 3 we describe in detail the numerical implementation of the adaptive mesh method based upon the above interpolation error estimate, and we develop local mesh improvement techniques such as refinement, coarsening and smoothing strategies which aim to minimize the interpolation error. In Section 4 we introduce the thin film fluid flow problem, give its background and solution properties, and review previous efforts for solving it numerically. An implementation of our new adaptive local refinement method is given in Section 5. Some concluding remarks are made in Section 6.

2. Theoretical foundation

In this section, we give an interpolation error estimate derived by Chen et al. [28] which provides the theoretical foundation for our algorithm based upon minimizing (or at least reducing) this interpolation error by iteratively modifying the grids.

The estimate. Let Ω be a bounded domain in R^n. Given a function $u \in \mathscr{C}^2(\overline{\Omega})$, a symmetric positive definite matrix $H \in R^{n\times n}$ is majorizing the Hessian of u if

$$|\xi^t(\nabla^2 u)(\mathbf{x})\xi| \leqslant c_0 \xi^t H(\mathbf{x})\xi, \quad \xi \in R^n, \ \mathbf{x} \in \Omega, \tag{2.1}$$

for some positive constant c_0.

One example of H can be constructed as follows: Given the Hessian $\nabla^2 u = Q^t \mathrm{diag}(\sigma_i) Q$, define

$$H = Q^t \mathrm{diag}(|\sigma_i|) Q + \delta I, \quad \delta \geqslant 0. \tag{2.2}$$

It is easy to see that H is a majorizing Hessian matrix of u satisfying (2.1) with $c_0 = 1$ for any $\delta > 0$. When $\nabla^2 u$ is singular, the positive parameter δ is critical to control the variation of H. A careful analysis in Huang [49] shows that δ can be used to control the ratio of mesh points in the singular region and smooth region of the function.

There are two conditions for a triangulation $\mathscr{T}_N$, where N is the number of simplexes, to be a nearly optimal mesh in the sense of minimizing the interpolation error in the L^p norm. The first requires the mesh to capture the high oscillation of the Hessian metric, viz., H does not change very much on each element.

(**A1**) There exist two positive constants α_0 and α_1 such that

$$\alpha_0 \xi^t H_\tau \xi \leqslant \xi^t H(\mathbf{x}) \xi \leqslant \alpha_1 \xi^t H_\tau \xi, \quad \xi \in R^n,$$

where H_τ is the average of H over τ, i.e.,

$$H_\tau = \frac{1}{|\tau|} \int_\tau H(x)\mathrm{d}x.$$

The majorizing Hessian H can be used to define a new metric

$$H_p = (\det H)^{-\frac{1}{2p+n}} H, \quad p \geqslant 1, \tag{2.3}$$

which is called a scaled majorizing Hessian matrix as well. This H_p defines a Riemannian metric on Ω, with corresponding edge length $\tilde{d}_{\tau,ij}$ and element volume $|\tilde{\tau}|$. Specifically, given a simplex τ with vertices $\{a_k\}_{k=1}^{n+1}$, denoting the edge vector $l_{ij} = a_i - a_j$, the edge length under this new metric is defined by

$$\tilde{d}_{\tau,ij} = \sqrt{l_{ij}^t H_{\tau,p} l_{ij}}, \tag{2.4}$$

and the new element volume is denoted as

$$|\tilde{\tau}| = \sqrt{\det H_{\tau,p}}|\tau|, \tag{2.5}$$

where

$$H_{\tau,p} = [\det(H_\tau)]^{-\frac{1}{2p+n}} H_\tau, \quad \forall \tau \in \mathscr{T}_N. \tag{2.6}$$

The second condition asks that $\mathscr{T}_N$ be quasi-uniform under the new metric induced by H_p.

(**A2**) There exist two positive constants β_0 and β_1 such that

$$\frac{\sum_i \tilde{d}_{\tau,i}^2}{|\tilde{\tau}|^{2/n}} \leqslant \beta_0, \quad \forall \tau \in \mathscr{T}_N \quad \text{and} \quad \frac{\max_{\tau\in\mathscr{T}}|\tilde{\tau}|}{\min_{\tau\in\mathscr{T}}|\tilde{\tau}|} \leqslant \beta_1, \tag{2.7}$$

where $|\tilde{\tau}|$ is the volume of τ and $\tilde{d}_{\tau,i}$ is the length of the ith edge of τ under the new metric H_p, respectively.

The first inequality in (2.7) means that each τ is shape-regular under the metric $H_{\tau,p}$ in that all edges of τ are of comparable size. This is related to the so-called isotropy criterion. The second inequality in (2.7) means that all elements τ are of comparable size (under the new metric), which is a global condition since it means that the mesh will concentrate at regions where $\det H_p(\mathbf{x})$ is large. This is related to the so-called equidistribution criterion. Thus, the condition (2.7) is consistent with viewing isotropy and equidistribution criteria as the two keys for developing our adaptive mesh strategy, a basic approach taken in Huang [49].

Theorem 1. *Let $u \in \mathscr{C}^2(\overline{\Omega})$, $\mathscr{T}_N$ satisfy assumptions (A1) and (A2) and u_I be the linear finite element interpolation of u based on the triangulation $\mathscr{T}_N$. Then the following error estimate holds*:

$$\|u - u_I\|_{L^p(\Omega)} \leqslant CN^{-2/n} \|\sqrt[n]{\det(H)}\|_{L^{\frac{pn}{2p+n}}(\Omega)} \tag{2.8}$$

for some constant $C = C(n, p, c_0, \alpha_0, \alpha_1, \beta_0, \beta_1)$. *This error estimate is optimal in the sense that for a strictly convex* (*or concave*) *function, the above inequality holds in the reverse direction.*

As mentioned before, Theorem 1 will be the basis of the grid adaptation algorithm. Roughly speaking, for a given function u, we will adapt our grids in such a way that the assumptions (A1) and (A2) will be more and more closely satisfied.

Optimality of the result: a lower bound. For a family of triangulations $\mathscr{T}_N$ of Ω we define

$$h_N = \max_{\tau \in \mathscr{T}_N} \operatorname{diam}(\tau),$$

and

$$\rho_N = \frac{\max \tilde{d}_{\tau,i}}{\min \tilde{d}_{\tau,i}}.$$

All edges $\{\tilde{d}_{\tau,i}\}$ are *asymptotically equal* if $\lim_{N\to\infty}\rho_N = 1$.

The error estimate (2.8) is in some sense sharp for strictly convex (or concave) functions as shown in the following theorem from [28].

Theorem 2. *Assume that* $u \in C^2(\overline{\Omega})$ *is a strictly convex* (*or concave*) *function and* $\{\mathscr{T}_N\}$ *is a family of triangulations of* Ω *satisfying* $\lim_{N\to\infty} h_N = 0$. *Then*

$$\liminf_{N\to\infty} N^{\frac{2}{n}} \|u - u_I\|_{L^p(\Omega),N} \geqslant C \|\sqrt[n]{det(H)}\|_{L^{\frac{pn}{2p+n}}(\Omega)} \tag{2.9}$$

for some constant $C = C(p, n)$. *Furthermore, the equality holds if all edges in* $\{\mathscr{T}_N\}$ *are asymptotically equal under the new metric, viz.*, $\lim_{N\to\infty}\rho_N = 1$.

With this theorem, it becomes possible now to do the adaptive local mesh coarsening by basing it on the above lower bound estimate (2.9).

3. General implementation issues

In this section, we elaborate on our new adaptive local mesh refinement method. Recall that our goal is to construct a quasi-uniform mesh under the new metric (2.3). Our strategy is to do the local refinement on a coarse grid by equidistributing the interpolation error in the whole triangulation, viz., we make the edge lengths of each triangle element under the new metric equal to each other. In grid generation, one often begins with a uniform or regular (in the Euclidean metric) grid which usually is not regular under the new metric. As a result, local refinement is needed to get a more regular (under the new metric) grid, which means, isotropy and equidistribution. Using the local edge-based error estimator in the previous section, a natural way to do the local refinement is to split edges with longer length to shorter ones under the new metric (2.4).

In general, a residual-type error estimator gives us an a posteriori error estimate based on the numerical solution u_h [38,60]. Our local edge-based error estimate is actually an interpolation error estimate (1.2) in which the Hessian matrix is of the real solution u instead of u_h. Strictly speaking (1.2) is not an a posteriori error estimate. But on the other hand, the interpolation error estimate builds the convergence rate on the number of elements N instead of the mesh size h, which is associated with adaptive local mesh refinement. In the asymptotic sense we could use u_h instead of u to compute the Hessian matrix and the new edge length under the new metric. Because we know the value of u_h exactly on each node, the Hessian matrix of u_h and the new edge length under the new metric (2.4) are all computable. The longer the edge length, the finer the local mesh refinement done in that element via bisection techniques. Asymptotically u_h will approximate u, and eventually an accurate adaptive locally refined mesh is obtained.

In order to calculate the new edge length (2.4), we by definition need to get the Hessian matrix first. Since taking piecewise second derivatives of piecewise linear functions will give no meaningful approximation to the Hessian matrix, special post processing techniques are needed to obtain a reasonable Hessian matrix approximation for linear finite elements. (Obviously, for higher order piecewise polynomials, a Hessian matrix approximation is easy to obtain.) In the next section, we discuss how the Hessian matrix of the numerical solution can be obtained for a linear finite element approximation.

3.1. Post processing: recovery of Hessian

A C^0 finite element solution u_h has a discontinuous gradient ∇u_h. In an attempt to better obtain a approximation than ∇u_h, we shall consider an effective method which constructs a continuous gradient $G_h u_h \in S_h \times S_h$, called the recovered gradient. Since any function in S_h is completely defined by its nodal values, it suffices to define $G_h u_h$ at the mesh nodes. This definition depends on the node location in $\overline{\Omega}$.

Several robust recovery methods to get the recovered gradient have been suggested in the literature. In the following we discuss them in some detail:

(I) *Superconvergence Patch Recovery method*. One of the most popular techniques is the Superconvergence Patch Recovery (SPR) technique proposed by Zienkiewicz and Zhu [79,80] which is based on least squares fitting locally. ZZ-SPR is used to recover a gradient from the gradient of the finite element solution. Results from their applications demonstrate that it is robust and efficient.
The definition of the SPR-recovered gradient at a mesh node z is as follows:
- If $z \in \Omega$, we use a patch ω_z consisting of the triangles attached to z. To recover the x-derivative at z, we find a polynomial $p_x \in P_1(\omega_z)$ that best fits, in the least squares sense, $\partial_x u_h$ at the triangle's centroids in ω_z. The recovered x-derivative at z is defined to be $p_x(z)$. Similarly, we can define the recovered y-derivative at z.
- If $z \in \partial\Omega$, let $z_1, z_2, \ldots, z_{N_{z,s}}$ denote the mesh nodes in Ω that are directly connected to z. Let ω_i be the patch associated with z_i for $i = 1, 2, \ldots, N_{z,s}$ and let $p_{x,i} \in P_1(\omega_i)$ be the polynomial that best fits $\partial_x u_h$ sampled at the triangles' centroids in ω_i. Again, the patch ω_i consists of the triangles attached to z_i. The recovered x-derivative at z is defined to be $\frac{1}{N_{z,s}} \sum_{i=1}^{N_{z,s}} p_{x,i}(z)$. Similarly, we can define the recovered y-derivative at z.
- If $z \in \partial\Omega$ with no attached internal nodes, the recovered gradient at z is defined to be $\nabla u_h(z)$.

(II) *Polynomial Preserving Recovery method*. The Polynomial Preserving Recovery (PPR) is a new gradient recovery technique [76] that can be used to recover a superconvergent gradient under some mild conditions imposed on the mesh, as was shown in [77]. This motivates the use of the PPR-recovered gradient in building an asymptotically exact a posteriori error estimator (the PPR estimator). Tests indicate it is as good as or better than the estimator based on the SPR-recovered gradient (ZZ-SPR estimator).
The construction of the PPR-recovered gradient at mesh nodes proceeds in two stages. In the first stage mesh nodes in Ω are considered while mesh nodes on $\partial\Omega$ are considered in the second stage.
- Stage 1 As in SPR, we use a patch ω_z consisting of the triangles attached to z. To recover the gradient at z, we find a polynomial $p \in P_2(\omega_z)$ that best fits u_h sampled at the mesh nodes in ω_z, in the least squares sense, and $G_h u_h(z)$ is defined to be $\nabla p(z)$. To get p, ω_z must contain at least 6 mesh nodes. If this is not the case and if ω_z does not share any edges with $\partial\Omega$, ω_z is extended by adding every triangle sharing an edge with ω_z. If ω_z has less than 6 mesh nodes with some of its edges on $\partial\Omega$, recovering the gradient at z is delayed to Stage 2.
- Stage 2 Basically this stage uses the gradient recovered in the first stage and linear extrapolation to complete the gradient recovery at the rest of the mesh nodes. Let $N_{h,0}$ be the set of mesh nodes left in Stage 1 without recovery, which includes those on $\partial\Omega$. The gradient recovery is completed in a finite number of iterations where every iteration proceeds as follows: The iteration starts by defining $\mathscr{T}_{h,0} \in \mathscr{T}_h$ where a mesh triangle $\tau \in \mathscr{T}_{h,0}$ if and only if $G_h u_h$ is defined at each of its vertices. For $z \in N_{h,0}$, let ω_z be the patch consisting of the triangles attached to z. We have two cases.

1. The patch ω_z has common edges with triangles $\tau_1, \tau_2, \ldots, \tau_{N_{z,p}}$ in $\mathscr{T}_{h,0}$. Let ω_{τ_i} denote the union of the triangles in $\mathscr{T}_{h,0}$ that have common edges with τ_i along with τ_i. Note that ω_{τ_i} has at least four nodes for which $G_h u_h$ is well-defined. Using least squares, we can find the linear polynomial $q_{x,i} \in P_1(\omega_{\tau_i})$ that best fits the x-components of $G_h u_h$ at the mesh nodes in ω_{τ_i} . The recovered x-derivative at z is defined to be $\frac{1}{N_{z,p}} \sum_{i=1}^{N_{z,p}} q_{x,i}(z)$. The recovered y-derivative at z is defined in a similar way.
2. The patch ω_z has no common edges with triangles in $\mathscr{T}_{h,0}$. In this case z is left for another iteration where it is added to $N_{h,1}$, a set taken to be empty at the beginning of the iteration.
   After going over all the nodes in $N_{h,0}$, if $N_{h,1}$ is empty, we are done; otherwise, we set $N_{h,0} = N_{h,1}$ and start another iteration.

(III) *Global L^2 projection*. The theoretical reason why the SPR and PPR methods work is related to the superconvergence phenomenon for second order elliptic boundary value problems discretized on a finite element grid having certain local symmetry (see Wahlbin [72], Chen and Huang [25] and Babuška and Strouboulis [4]). These classic superconvergence results can be used to justify the effectiveness of the ZZ-SPR and PPR recovery methods, e.g. see [75,59] for nearly structured grids.

A significant improvement of this type of analysis was given recently by Bank and Xu [7,8]. In [7] superconvergence estimates are given for piecewise linear finite element approximation on quasi-uniform triangular meshes, where most pairs of triangles sharing a common edge form approximate parallelograms. In [8] they use the smoothing iteration of the multigrid method to develop a post processing gradient recovery scheme. This scheme has proven to be very efficient for recovering the Hessian matrix, showing that $Q_h \nabla u_h$ is a superconvergent approximation to ∇u. Here, Q_h is the global L^2 projection. This result leads to a theoretical justification of the ZZ-SPR method for such types of grids – see Xu and Zhang [74].

Based on the well known standard L^2 projection operator Q_h: $L^2 \to S_h$, the definition of the global L^2 projection-recovered gradient is as follows: suppose that the numerical solution $u_h \in S_h$ has been obtained, so $\nabla u_h \subset L^2$. We find $Q_h \nabla u_h \in S_h$ such that

$$(\nabla u_h - Q_h \nabla u_h, v_h) = 0, \quad \forall v_h \in S_h, \tag{3.1}$$

where $(\cdot,\cdot)$ is the inner product, and $S_h \subset P_1$ is the piecewise linear finite element space. Then we can discretize (3.1) by summing up the local inner products in each element $\tau \in \mathscr{T}_h$ for the right hand side only, i.e.,

$$\int_\Omega (Q_h \nabla u_h) v_h \, \mathrm{d}x = \sum_\tau \int_\tau \nabla u_h v_h \, \mathrm{d}x, \quad \forall v_h \in S_h. \tag{3.2}$$

Since ∇u_h is constant in each element τ, it is discontinuous while crossing the edge of each element, but it is still well defined in each local element. We can then get the appropriate right-hand side of (3.2) directly. By means of standard finite element approximation, the linear algebraic system with respect to the unknown $Q_h \nabla u_h$ is

$$M \nabla U = F, \tag{3.3}$$

where $M = ((\phi_i, \phi_j))_{N \times N}$ is the mass matrix, $\phi_i \in S_h$ $(i = 1, 2, \ldots, N)$ are piecewise linear nodal basis functions, $\nabla U = (\nabla_x U, \nabla_y U)^{\mathrm{T}}$ is the vector of unknown recovered gradients in which $\nabla_x U$ and $\nabla_y U$ are defined at each mesh node, and $F = (F_x, F_y)^{\mathrm{T}}, F_x = \sum_\tau \int_\tau \nabla_x u_h \phi_j \, \mathrm{d}x, F_y = \sum_\tau \int_\tau \nabla_y u_h \phi_j \, \mathrm{d}x, \ j = 1, 2, \ldots, N$.

By carefully choosing the weight function for the matrix M in the lumping mass method, we are able to get a diagonal mass matrix. The numerical solution $Q_h \nabla u_h$ is then directly found without solving a linear algebraic system. We describe below a generic method for choosing the weight coefficient for each node in a triangle while using the lumping mass method.

We just consider the scheme on a reference triangle element, namely an isosceles right triangle with unit length for the two equal sides, so its area is $\frac{1}{2}$. The basic idea is to equidistribute the area on each node in terms of the associated relationship among the nodes in the reference triangle element. First, one determines how many small equilateral triangles can be generated between other nodes and the node of interest. For a given order finite element on a triangle, assume that in addition to the three vertices there are m points on each edge, and each point on the edge has k_e small surrounding equilateral triangles which are gained by connecting this

middle point with other associated points. Moreover, there are n interior points inside the triangle, and each one has k_c small surrounding equilateral triangles gotten from its patch. For instance, $m = n = 0$, $k_e = k_c = 0$ in the linear case, $m = 1$, $n = 0$, $k_e = 2$, $k_c = 0$ in the quadratic case, and $m = 3$, $n = 1$, $k_e = 3$, $k_c = 6$ in the cubic case. Let w be the weight coefficient of a vertex, which is also the basic weight for a single point. Then the formula for the weight coefficient of each point is

$$3w + mk_e w + nk_c w = \frac{1}{2},$$

so the weight for vertices is

$$w = \frac{1}{2(3 + mk_e + nk_c)}.$$

Consequently, the weight is $k_e w$ for each point on the edge and $k_c w$ for each interior point in the triangle.

All of the above gradient recovery methods can be extended to anisotropic grids with proper modifications, but a theoretical justification of such extensions is still lacking. Nevertheless, numerical experiments have given satisfactory results.

The gradient recovery algorithm adopted in this paper is based on a global L^2 projection approach. It has been proven in practice that the L^2 projection method is much easier to implement, efficient, and has fairly good recovery for the gradient too, although theoretically the Zienkiewicz–Zhu scheme and Polynomial Preserving Recovery method may have higher accuracy and better approximation on the boundary.

By means of the above gradient recovery techniques, once we get the recovered gradient $Q_h\nabla u_h$, we can keep doing the same post processing for $Q_h\nabla u_h$ to get its recovered Hessian matrix $Q_h\nabla(Q_h\nabla u_h)$. This is one approximation for a linear finite element, which is the case considered here. For a quadratic finite element, it is better to obtain the Hessian matrix by doing the post processing with $Q_h\nabla^2 u_h$.

For the majorizing Hessian matrix H_p, we use our edge-based local refinement method following (2.2) and (2.3), where we do the eigen decomposition for the $n \times n$ matrix $Q_h\nabla(Q_h\nabla u_h)$, taking absolute values for its eigenvalues, and add δI as in (2.2) to ensure the recovered Hessian matrix H_p is eventually symmetric positive definite.

3.2. Mesh adaptation

In this section, we discuss techniques which aim at improving the mesh quality. Here we define the mesh quality for a triangulation $\mathscr{T}$ using the interpolation error

$$Q(\mathscr{T}, u, p) = \|u - u_{I,\mathscr{T}}\|_{L^p(\Omega)}, \quad 1 \leqslant p \leqslant \infty.$$

Local improvements. There are mainly three types of mesh improvements: (1) refinement or coarsening, viz., split or merge edges [5,64,55], (2) edge swapping: replace sets of elements by other such sets, while preserving the position of the points [57] and (3) mesh local smoothing, viz., move the vertices of the mesh while keeping the connectivity [41,6,65,44]. We derive these techniques by minimizing the interpolation error in the L^p norm, which can be achieved by equidistributing edge lengths under the new metric.

Thus, we compute edge lengths under the new metric H_p via (2.4) and mark edges whose length is greater than $r_1 d$, where $r_1 \geqslant 1$ is a parameter and d is a fixed edge length. Usually we adopt the global average edge length as the fixed d. A given parameter r_1 is used to control the density of local adaptive meshes. It can be chosen as sequentially decreasing, starting from a relatively large number and gradually decreasing to, say, 1. Thus a better locally refined mesh can be gained by more tightly concentrating around the singular regions.

Such edge-based local mesh refinement has at least two advantages:

1. The anisotropic mesh can be automatically generated in terms of the new edge length defined by majorizing the Hessian matrix. By means of the longest edge principle, bisection always happens on the longest edge length under the new metric, where the Hessian matrix is large due to the solution's singularity. For instance, for a sharp interface, the locally refined mesh will be automatically stretched along the interface. We connect marked edges element-wise according to the situation, as shown in Fig. 1.
   Bisecting the marked element in this way, we get an anisotropic locally refined mesh mainly along the singular direction, i.e., the direction in which the Hessian matrix of the solution is sufficiently large.

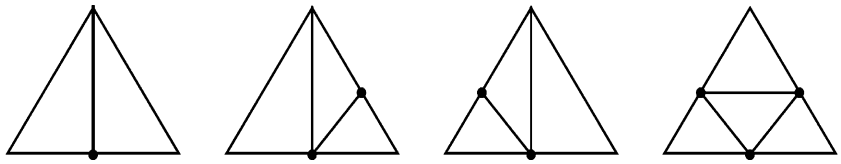

Fig. 1. Edge-based refinement.

2. In contrast to the residual-type a posteriori error estimates, there is no pollution with this refinement strategy. Based on the longest edge principle, we usually need to bisect more elements around the marked element in order to get a conforming mesh for a residual-type estimator. But for the marked element given by our new local edge-based estimator, we only need to bisect the adjacent element which shares the same edge, because it always has the same longest edge based on our theory of the edge-based local mesh refinement. There is no more bisection for other adjacent elements, which means that the generated locally refined meshes are all necessarily much more concentrated in singular regions than elsewhere. This is unlike the residual-type estimator, where more unnecessary locally refined meshes come into being due to the need for conforming the meshes instead of actually resolving singularities, which is the main goal of the mesh adaptivity.

The coarsening operates like an inverse procedure to refinement. It marks a triangle whose length is less than $r_2 d$ for some parameter $r_2 \leqslant 1$. We then shrink this edge to a point and connect the vertices of the patch of the edge (see Fig. 2).

Now we consider the edge swapping for four points $\{\mathbf{v}_i\}_{i=1}^4$ which form two adjacent triangles and a convex quadrilateral as shown in Fig. 3. Let $\mathscr{T}_1 = \Delta_{123} \cup \Delta_{134}$ and $\mathscr{T}_2 = \Delta_{124} \cup \Delta_{234}$, where Δ_{ijk} stands for the triangle made up of $\mathbf{v}_i$, $\mathbf{v}_j$ and $\mathbf{v}_k$. We choose triangulation $\mathscr{T}_1$ if and only if $Q(\mathscr{T}_1, u, p) \leqslant Q(\mathscr{T}_2, u, p)$, for some $1 \leqslant p \leqslant \infty$. In the implementation, we calculate the area of two quadrilaterals $\mathscr{T}_1$ and $\mathscr{T}_2$ in terms of the new edge length under the new metric and choose the one with smaller area as the edge swapped triangulation. Thus, it is a suitable generalization of the edge swapping used in the isotropic case to the anisotropic case, with a change in the metric, namely the method of defining the edge length.

Local mesh smoothing adjusts the location of a vertex in its patch Ω_i, which consists of all simplexes containing vertex $\mathbf{x}_i$, without changing the connectivity. Moving a vertex to the new location improves the mesh quality. Several sweeps through the whole mesh can be performed to improve the overall mesh quality. By minimizing the interpolation error in Ω_i, we move $\mathbf{x}_i$ to $\mathbf{x}^*$ such that

$$\nabla u(\mathbf{x}^*) = -\frac{1}{|\Omega_i|} \sum_{\tau_j \in \Omega_i} \left(\nabla_{x_i} |\tau_j| \sum_{\mathbf{x}_k \in \tau_j, \mathbf{x}_k \neq \mathbf{x}_i} u(\mathbf{x}_k) \right). \tag{3.4}$$

where $u \in C^1(\Omega)$ is a convex function.

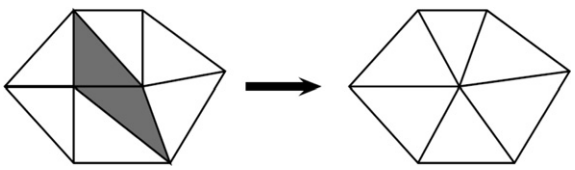

Fig. 2. Coarsening.

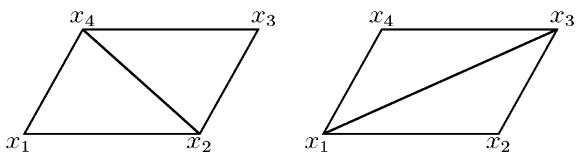

Fig. 3. Edge swapping.

The significance of (3.4) is that we can recover the derivative exactly from the nodal values of the function if the triangulation is optimized. With the gradient information, we can approximate u by higher degree polynomials or construct an a posteriori error indicator. In practice, this is applied to the numerical solution, so we use $Q_h\nabla u_h$ and u_h in (3.4) to perform the calculation.

If the triangulation is not optimized, (3.4) can be used to solve for the critical point, and the critical point can be used as the new location for the mesh smoother. When $u(x) = u_H(x) := x^{\mathrm{T}}Hx$ is a nondegenerate quadratic function, i.e. H is a $n \times n$ nonsingular matrix, we can solve for the critical point exactly and get a mesh smoother based on optimal Delaunay triangulations (ODTs) [26]. Especially when the goal of the mesh adaptation is to get a uniform and shape regular mesh, we choose $u(x) = \|x\|^2$ and get the smoother for uniform density

$$\mathbf{x}^* = -\frac{1}{2|\Omega_i|}\sum_{\tau_j\in\Omega_i}\left(\nabla_{x_i}|\tau_j|\sum_{\mathbf{x}_k\in\tau_j,\mathbf{x}_k\neq\mathbf{x}_i}\|\mathbf{x}_k\|^2\right) \tag{3.5}$$

(for more details, see [26]).

Using local smoothing for the above local improvement techniques, we can get a more anisotropic locally refined mesh.

4. Thin film fluid flows

Many physical phenomena exhibit strong anisotropic behaviors, e.g. boundary layer flows in porous media, currents and concentrations in fuel cells, characteristics of semiconductors, and stresses and strains in thin plates, shells and anisotropic materials. When an anisotropic phenomenon occurs and an accurate approximation is required, it is natural to use stretched anisotropic meshes according to the variation of the solution. This reduces the number of elements needed to partition the domain, better captures the solution behavior, and thus leads to much more efficient algorithms. A thin film fluid flow problem is one type of anisotropic physical problem where local solution difficulties make use of mesh adaptivity very attractive vis a vis obtaining a numerical solution accurately and efficiently.

In recent years several mathematical models in fluid dynamics, materials science and plasticity have led to fourth order nonlinear degenerate parabolic equations. We mention, for example, the lubrication approximation for thin viscous films [45,19,12], the Cahn–Hilliard equation with a degenerate mobility ([22,39,40]) and models that describe dislocation densities in plasticity ([46]). There has been considerable effort to analytically understand degenerate parabolic equations of higher order. There have been a number of numerical experiments for the thin viscous film equation in one space dimension (e.g. see [19,14,1]). These are extended to higher dimensions in [53,35], but only for a fixed uniform grid with a finite difference method. Even for the 1D case, the adaptive mesh method used in [14] is a so-called self similar adaptive mesh scheme and only handles a known singularity. There is no a posteriori error estimator, and this approach does not easily extend to the higher dimensional case with complex solution structure.

Accurately locating the contact lines for a thin film fluid is important in order to avoid premature rupture [18]. The most popular way to do this has been to use the h-version of adaptivity, where more grid points concentrated around the contact line results in more accurate computation. A uniformly distributed mesh throughout the domain is unsuitable because too many grid points in the smooth regions away from the contact line produce unnecessary computational cost. Also, when schemes of this type are used to compute finite time singularities describing things such as film rupture [19] or neck rupture in a Hele–Shaw cell [1], lack of further local mesh refinement prevents resolving small scale structures near the point of singularity.

4.1. Description of model

Driven films exhibit a variety of complicated dynamics ranging from rivulets and sawtooth patterns in gravity driven flows [52,66] to patterns in spin coating [43] and surfactant driven films [69]. A theoretical framework for these problems is provided by a lubrication approximation of the Navier–Stokes equations

[61,45]. This yields a single partial differential equation for the film thickness as a function in time of position on the solid substrate.

For directionally driven films, the driving force enters into the lubrication approximation as the flux F in a scalar hyperbolic conservation law $u_t + (F(u))_x = 0$. Here $u > 0$ is the film thickness and x is the direction of the driving force. For gravity driven flow on an incline, the tangential component of gravity yields a flux proportional to u^3 [51]. For surface-tension gradient driven flows F is proportional to u^2 [24]. In each of these cases, the flux is convex. Consequently, driven fronts in the film correspond to compressive shock solutions, whose simplest form is

$$u(x,t) = \begin{cases} u_-, & \text{if } x < st, \\ u_+, & \text{if } x > st, \end{cases} \tag{4.1}$$

in which the shock speed $s = (F(u_-) - F(u_+))/(u_- - u_+)$ satisfies the entropy condition

$$F'(u_+) < s < F'(u_-), \tag{4.2}$$

equivalently, characteristics enter the shock on each side. Small variations in height near a compressive shock are propagated towards the shock from both sides.

In practice, the discontinuous fronts (4.1) are smoothed by diffusive effects, primarily through surface tension. In the lubrication approximation, surface tension appears as a fourth order nonlinear regularization of the conservation law, but there is also second order nonlinear diffusion induced by the component of gravity normal to the incline, leading to the equation

$$u_t + (F(u))_x = -\gamma \nabla \cdot (u^3 \nabla \Delta u) + \beta \nabla \cdot (u^3 \nabla u). \tag{4.3}$$

In this equation, $\gamma > 0$ and $\beta \geqslant 0$ are constants. The shock waves (4.1) correspond to smooth travelling wave solutions of (4.3); for small $\beta \geqslant 0$, they typically have oscillatory overshoots and undershoots on either side of the shock. Additionally, the nonlinearity in the fourth order diffusion causes a single very pronounced overshoot or "bump" on the leading edge of the shock (see Fig. 1(a) of [15]); this structure is often referred to as a capillary ridge in experiments. Capillary ridges produced by surface tension are well known to be linearly unstable to long-wave perturbations in the transverse direction of the flow, producing the well-known fingering instability [23,68]. The effect of larger β is to suppress the bump (see Fig. 1(b) of [15]). The disappearance of the bump (for β sufficiently large) is accompanied by a transverse stabilization of the wave [18].

In this paper, we specifically focus on the travelling wave solutions $u(x,t)$ of a thin film flow equation of the form

$$\begin{cases} u_t + (F(u))_x &= -\nabla \cdot (D(u)\nabla \Delta u) + \nabla \cdot (G(u)\nabla u), & \text{in } \Omega_T := \Omega \times (0,T], \\ u(x,t) &= u_B(x,t), & \text{on } \partial\Omega_1 \times (0,T], \\ \frac{\partial u}{\partial v} &= 0, & \text{on } \partial\Omega/\Omega_1 \times (0,T], \\ u(x,0) &= u_0(x), & \forall x \in \Omega, \end{cases} \tag{4.4}$$

where

$$F(u) = \alpha u^n, \quad G(u) = \beta u^n, \quad D(u) = \gamma u^n, \quad \alpha \geqslant 0, \quad \beta \geqslant 0, \quad \gamma > 0. \tag{4.5}$$

Here, we apply mixed boundary conditions on $\partial\Omega$. The Dirichlet boundary condition $u_B(x,t)$ is usually given on the upstream and downstream boundary. In the rest of the paper n denotes the power of u in (4.5).

The PDE in (4.4) is derived in [45] as a model for the surface tension dominated motion of thin viscous films and spreading droplets. For such problems, the power n depends on the boundary condition on the liquid solid interface: no-slip gives $n = 3$ while various Navier slip conditions can yield $n < 3$. The same equation in one space dimension with $D(u) = u$ is also shown to model a thin neck in the Hele–Shaw cell [29]. Other applications include Cahn–Hilliard models with degenerate mobility [39], population dynamics [58], and problems in plasticity [46]. In all examples, in order to have a physical solution, u must be nonnegative.

While second order degenerate diffusion equations satisfy a maximum principle guaranteeing that solutions are bounded from above and below by their initial data, the fourth order analogue (4.4) does not possess such

a property even for $D(u) = 1$. In particular, positive initial data can easily lead to solutions which change sign. This occurs for the Cauchy problem with $L^2(R^d)$ initial data.

What is unusual about (4.4) is that for sufficiently large values of n, the equation does preserve positivity of the solution. This was proven in one space dimension for $n \geqslant 4$ [13] and later extended to $n \geqslant 3.5$ [19]. For the two-dimensional (2D) problem, numerical computations of thin film flows suggest the same may be true. However, for smaller values of $n > 0$, numerical simulations [19,17,1] show that solutions can develop singularities of the form $u \to 0$, which physically describe the rupture of the liquid film. If the singularity forms in finite time, then the solution past this time can be defined through a regularization method [13,16,11] as a limit of strictly positive smooth solutions of regularized problems.

The nonlinear structure of the PDE presents a challenge in the design of efficient and accurate numerical methods. Even when the analytical solution is strictly positive, the numerical solution for a generic scheme may become negative, especially when the grid is under-resolved. Since the PDE becomes degenerate as $u \to 0$ this may lead to numerical instabilities. When a positive approximation of the solution is desired, it may be necessary to do computationally local mesh refinement near the minimum of the solution in order to avoid such premature or "false" singularities [19]. Physically important examples of situations for which such singularities may arise include flow down an inclined plane [18], where resolution is required at the apparent contact line [14].

The remainder of this paper focusses on efficient numerical solution of the thin film flow equation (4.4).

4.2. Basics, challenges, and techniques

One faces a number of potential difficulties when computing solutions to fourth order PDEs. The dynamics depend highly on the smoothness of initial data. Fourth order equations require prescribing more boundary conditions than for second order equations, and these can be higher order and consequently often difficult to implement. Time stepping can also be a crucial matter for fourth order equations, where stability requirements for explicit time step methods need be on the order of h^4, where h is the grid size, instead of h^2 as for second order problems. This can make the explicit schemes prohibitively expensive. On the other hand, implicit schemes require the inversion of a linear system of equations that is typically very large for fourth order equations. The literature for fourth order PDEs in arbitrary smooth domains is still in its infancy, although a number of numerical tools, including ADI, spectral, and finite element methods have proven successful for fourth order diffusion equations occurring in areas such as thin film fluid flow and materials science, [9,47,73,78]. We consider an adaptive finite element method (AFEM) and show how AFEM performs for solving the thin film fluid flow problem described in (4.4).

Almost all of the work on numerical simulation for the thin film equation (4.4) has used finite difference schemes, although the more flexible finite element method is used in [78,9]. In [78] a positivity-preserving scheme (PPS), a kind of modified finite difference scheme, is presented. By introducing a discrete inner product instead of a continuous one, it is also shown that PPS can be considered as a special case of the general finite element approximation. In this paper, we show that using a suitable numerical quadrature, a finite element approximation is equivalent to the positivity-preserving scheme. In [9], a nonnegativity-preserving finite element method is proposed. Nonnegativity of the solution is imposed as a constraint, so one has to solve a variational problem involving a Lagrange multiplier at every time step to advance the nonnegative solution. The advantage of using this method lies in the fact that computation of nonnegative solutions requires no regularization of the PDE or initial condition, which seems particularly useful for tracking the moving contact lines. However, this method allows for solutions with positive initial data to lose positivity, which makes it less capable of capturing singularity formation comparable to that for the positivity-preserving scheme described in [78].

It is well known that in contrast to the finite difference method, the finite element method can be suited to convenient numerical solution of PDEs on arbitrary domains using complicated mesh structures. Since mesh adaptivity creates nonuniform meshes, locally concentrating grid points where solution singularities occur, the finite element method is well suited to our purpose. In the following sections, we explore how the adaptive finite element method performs on the thin film equation (4.4) using local mesh refinement to resolve small scale structures near the singularity produced by a moving contact line. More explicitly, we adopt the framework of the finite element method proposed in [78] and extend the method to use adaptive meshes.

In [36] the simpler thin film equation

$$\frac{\partial u}{\partial t} = -\nabla \cdot (u^n \nabla \Delta u), \tag{4.6}$$

which includes only capillary effects, is considered for various values of n. This is the special case of (4.4) with $\alpha = 0$, $\beta = 0$ and $\gamma = 1$.

Using an entropy function, it is shown in [13,19] that in the planar case (one space dimension) positive initial conditions always yield a positive solution for sufficiently large n. A practical importance of this fact is that one can construct a nonnegative solution of the PDE for relatively small values of n by using a modified version of the diffusivity $D(u)$, lifting of the initial condition, and passing to the limit. This fact has been proven for the planar case [11,16]. This *regularization* procedure involves replacing the diffusivity D by the modified diffusivity

$$\widetilde{D}(u) = \frac{D(u)u^4}{\varepsilon D(u) + u^4} \tag{4.7}$$

and the nonnegative initial profile $u(x,0)$ with an artificially lifted one $u(x,0) + b$ (where b is small and does not represent a physical precursor). Since $\widetilde{D}(u) \to D(u)$ as $\varepsilon \to 0$, and $\widetilde{D}(u) \to u^4/\varepsilon$ as $u \to 0$ (for $n < 4$), this guarantees positivity of the regularized solution.

It is shown in [36] that for $n = 1$ the numerical solutions are convergent if the positivity-preserving scheme (PPS) plus regularization are employed, and using a standard finite difference scheme (SS) for the interpolation of the diffusivity fails. For $n = 2$, SS and PPS become coincident for small Δx. However, both schemes show that the computation must be done with very small $\Delta x (< 10^{-3})$, in order to obtain a converged solution as $b \to 0$. For $n = 3$, [36] indicates that, by means of the precursor film model, both SS and PPS seem appropriate to perform the simulation. The only (but important) difference is that PPS converges faster, although it is less accurate than SS for large Δx. The conclusion is that the use of the precursor film model is highly advisable.

In this paper, we also use the precursor film model in our AFEM simulation. This is a crucial point for the thin film flow equation in order to preserve the positivity of the solution starting from the initial time. Using anisotropic adaptive mesh techniques, we aim to eliminate or weaken the regularization condition (4.7) and still get the always positive solution, but for much smaller ε.

5. Mixed finite element method for fourth order PDE

In principle, there are a number of natural finite element methods to numerically solve fourth order PDEs. One is the standard FEM with high order piecewise polynomials which include derivatives as degrees of freedom – say, the cubic Hermite elements; another is the mixed finite element method. Considering the difficulties of constructing a Hermite element in an arbitrary triangular element or a tetrahedron element in the 3D case, we prefer the mixed finite element method. For a fourth order PDE with a sufficiently smooth solution, namely $u \in H^4(\Omega)$, by introducing auxiliary variables of second order derivatives, the mixed method is able to reduce the original fourth order PDE to an elliptic equation system. The well known integrated theory of mixed finite element methods can guarantee an unique numerical solution exists [63], and the well-posed thin film flow Eq. (4.4) falls within its theoretical framework.

In the following, we more precisely formulate the mixed finite element approximation for (4.4). In order to achieve this, we introduce a potential w and write the fourth order parabolic equation as a system of second order equations

$$\begin{cases} \frac{\partial u}{\partial t} - \nabla \cdot (u^n \nabla w) + (\alpha u^n)_x = 0, & \text{in } \Omega_T, \\ w = -\gamma \Delta u + \beta u, & \text{in } \Omega_T, \\ u(x,t) = u_B(x,t), & \text{on } \partial\Omega_1 \times (0,T], \\ \frac{\partial u}{\partial \nu} = 0, & \text{on } \partial\Omega/\Omega_1 \times (0,T], \\ \frac{\partial w}{\partial \nu} = 0, & \text{on } \partial\Omega \times (0,T], \\ u(x,0) = u_0(x), & \forall x \in \Omega, \end{cases} \tag{5.1}$$

where the second equation holds for positive u.

We consider the finite element approximation under the following assumptions on the mesh: Let Ω be a polyhedral domain and $\mathscr{T}^h$ be a triangulation of Ω into disjoint open simplices κ with $h_\kappa = \mathrm{diam}(\kappa)$ and $h := \max_{\kappa\in\mathscr{T}^h} h_\kappa$, so that $\overline{\Omega} = \cup_{\kappa\in\mathscr{T}^h}\overline{\kappa}$.

Associated with $\mathscr{T}^h$ is the finite element space

$$S^h := \{\chi \in C(\overline{\Omega}) : \chi|_\kappa \text{ is linear } \forall \kappa \in \mathscr{T}^h\} \subset H^1(\Omega).$$

Let J be the set of nodes of $\mathscr{T}^h$ and $\{x_j\}_{j\in J}$ the coordinates of these nodes. Let $\{\chi_j\}_{j\in J}$ be the standard basis functions for S^h, i.e. $\chi_j(x_i) = \delta_{ij}$ for all $i,j \in J$.

Given a positive integer N, let $\Delta t := T/N$ denote the time step and $t_j := j\Delta t$, $j = 1 \to N$. We have considered two practical finite element discretizations of (4.4) which are combined with the Backward *Euler* scheme to approximate the time derivative $\frac{\partial u}{\partial t}$. The first is *Picard* iteration.

For $j \geqslant 1$, find $\{U^j, W^j\} \in S^h \times S^h$ such that

$$\begin{cases} \left(\frac{U_j - U_{j-1}}{\Delta t}, \chi\right) + (U^n_{j-1}\nabla W_j, \nabla\chi) - (\alpha U^{n-1}_{j-1} U_j, \chi_x) &= 0, \quad \forall \chi \in S^h, \\ \gamma(\nabla U_j, \nabla\chi) + \beta(U_j, \chi) - (W_j, \chi) &= 0, \quad \forall \chi \in S^h, \end{cases} \tag{5.2}$$

where $U_0 \in S^h$ is an interpolant of u_0. The Picard scheme provides first order accuracy. The second, *Newton's* method, improves the accuracy to second order. It has the form

$$\begin{cases} \left(\frac{U_j - U_{j-1}}{\Delta t}, \chi\right) - \alpha(nU^{n-1}_{j-1}U^j, \chi_x) + (nU^{n-1}_{j-1}\nabla W_{j-1}U^j, \nabla\chi) \\ +(U^n_{j-1}\nabla W^j, \nabla\chi) = ((1-n)U^n_{j-1}, \chi_x) + (nU^n_{j-1}\nabla W_{j-1}, \nabla\chi), \quad \forall \chi \in S^h, \\ \gamma(\nabla U^j, \nabla\chi) + \beta(U^j, \chi) - (W^j, \chi) = 0, \quad \forall \chi \in S^h. \end{cases} \tag{5.3}$$

In our numerical simulation Newton's method is employed to linearize the nonlinear diffusion term $D(U)$, assuming $D(u)$ is explicitly given in terms of u.

6. The consistency of FEM and PPS

In [78] the authors indicate that it is possible to generalize the positivity-preserving scheme (PPS) to finite element methods on arbitrary element spaces (including those involving nonuniform grids). They begin by showing that the 1D PPS finite difference scheme is equivalent to a specific finite element discretization in which a nonlinear function of the solution is represented in the element basis. This representation suggests a general abstract finite element approximation of the problem in any space dimension. They then show that the 2D positivity-preserving scheme is another special case of the general finite element approximation involving one particular numerical quadrature scheme for the uniform mesh case.

Below we consider again these two examples of 1D and 2D finite element approximation and simply prove how they are actually equivalent to the general practical finite element approximations based on the schemes presented in [78].

6.1. 1D FEM

The following finite element approximation, shown to be a positivity-preserving finite difference scheme for the simplest thin film equation (4.6) is presented in [78]: find $(U, W) \in S^h \times S^h$ such that

$$\begin{cases} (U_t, \chi)^h + (D(U)\partial_x W, \partial_x\chi) &= 0, \quad \forall \chi \in S^h, \\ (\partial_x U, \partial_x\chi) &= (W, \chi)^h, \quad \forall \chi \in S^h, \end{cases} \tag{6.1}$$

where the discrete inner product $(\cdot,\cdot)^h$ is defined by

$$(\eta_1, \eta_2)^h \overset{\text{def}}{=} \int_{S^1} \pi^h(\eta_1(x)\eta_2(x))\mathrm{d}x \equiv \sum_j \eta_1(x_j)\eta_2(x_j)h, \quad j \in J. \tag{6.2}$$

Here S^1 is a periodic domain and $\pi^h : C(S^1) \to \mathscr{T}^h$ is an interpolation operator such that $(\pi^h\eta)(x_j) = \eta(x_j)\forall j$.

Taking a nodal numerical quadrature to discretize the continuous inner product (η_1, η_2), we get

$$\begin{aligned}(\eta_1,\eta_2) &= \sum_{\kappa}\int_{\kappa}\eta_1(x)\eta_2(x)\mathrm{d}x = \sum_{\kappa}\sum_{j=1}^{2}\eta_1(x_j)\eta_2(x_j)\frac{\Delta_\kappa}{2} = \sum_{\kappa}\sum_{j=1}^{2}\eta_1(x_j)\eta_2(x_j)\frac{h}{2}\\ &= \sum_{j\in\Omega}\eta_1(x_j)\eta_2(x_j)h + \sum_{j\in\partial\Omega}\eta_1(x_j)\eta_2(x_j)\frac{h}{2},\end{aligned} \tag{6.3}$$

where Δ_κ is the area of element κ. Since $\Omega = (S^1)^d$ is a periodic domain, no grid point really falls on the boundary, and we can eliminate the second term of (6.3). This implies that

$$(\eta_1,\eta_2) = \sum_j \eta_1(x_j)\eta_2(x_j)h = (\eta_1,\eta_2)^h, \tag{6.4}$$

i.e. the discrete inner product (6.2) is equivalent to the standard inner product (η_1, η_2) in terms of the nodal numerical quadrature (6.3). Furthermore, the corresponding mixed finite element approximation which determines $(U, W) \in S^h \times S^h$ such that

$$\begin{cases}(U_t,\chi) + (D(U)\partial_x W, \partial_x\chi) &= 0, & \forall\chi\in S^h,\\ (\partial_x U, \partial_x\chi) &= (W,\chi), & \forall\chi\in S^h,\end{cases} \tag{6.5}$$

is also equivalent to the 1D positivity-preserving finite difference scheme in [78].

6.2. Higher dimensional FEM

The general finite element method considered in [78] for the higher dimensional thin film equation (4.6) consists of finding $(U,W) \in S^h \times S^h$ such that

$$\begin{cases}(U_t,\chi)^{I_1} + (D(U)\nabla W, \nabla\chi)^{I_2} &= 0, & \forall\chi\in S^h,\\ (\nabla U, \nabla\chi)^{I_2} &= (W,\chi)^{I_1}, & \forall\chi\in S^h,\end{cases} \tag{6.6}$$

Defining the discrete inner product by $(\eta_1,\eta_2)^{I_1} = \int_{(S^1)^2}\pi^{I_1}(\eta_1(x)\eta_2(x))\mathrm{d}x$ to be an associated inner product on $C((S^1)^2)$, where $\pi^{I_1} : C((S^1)^2) \to \mathscr{T}^h$ is an interpolation operator, (6.6) is proven to also be a 2D positivity-preserving finite difference scheme. The term $(\xi_1,\xi_2)^{I_2}$ denotes a numerical quadrature rule replacing an $L^2(\Omega)$ inner product.

More specifically, in the square domain

$$(\eta_1,\eta_2)^{I_1} = \sum_{i,j}\eta_1(x_{ij})\eta_2(x_{ij})h_x h_y, \quad i\in J,\ j\in J, \tag{6.7}$$

where h_x, h_y denote the mesh size in the X and Y directions respectively, and

$$\begin{aligned}(\xi_1,\xi_2)^{I_2} &= \sum_{i,j}\int_{x_i}^{x_{i+1}}\frac{1}{2}(\xi_1^1(x,y_{j+1})\xi_2^1(x,y_{j+1}) + \xi_1^1(x,y_j)\xi_2^1(x,y_j)h_y\,\mathrm{d}x\\ &\quad + \sum_{i,j}\int_{y_i}^{y_{i+1}}\frac{1}{2}(\xi_1^2(x_{i+1},y)\xi_2^2(x_{i+1},y) + \xi_1^2(x_i,y)\xi_2^2(x_i,y)h_x\,\mathrm{d}y.\end{aligned} \tag{6.8}$$

For this product, in the first set of terms the integration is performed exactly for the first variable but the trapezoidal rule is used for the second variable. Similarly, in the second set of terms, integration is performed exactly for the second variable but the trapezoidal rule is used for the first variable. For this specific numerical integration rule (6.6) is shown to be exactly the 2D positivity-preserving scheme in [78].

We indicate below why the discrete inner product (6.7) is equivalent to the standard inner product (η_1, η_2) with the nodal numerical quadrature. By discretizing the continuous inner product (η_1, η_2) in terms of the nodal numerical quadrature for a triangular partitioning, we get

$$
\begin{aligned}
(\eta_1,\eta_2) &= \sum_{\kappa}\int_{\kappa}\eta_1(x)\eta_2(x)\mathrm{d}x = \sum_{\kappa}\sum_{j=1}^{3}\eta_1(\bar{x}_j)\eta_2(\bar{x}_j)\bar{\omega}_j = \sum_{\kappa}\sum_{j=1}^{3}\eta_1(x_j)\eta_2(x_j)\bar{\omega}_j|J_\kappa| \\
&= \sum_{\kappa}\sum_{j=1}^{3}\eta_1(x_j)\eta_2(x_j)\frac{2\Delta_\kappa}{6} = \sum_{\kappa}\sum_{j=1}^{3}\eta_1(x_j)\eta_2(x_j)\frac{h_x h_y}{6} \\
&= \sum_{j\in\Omega}\eta_1(x_j)\eta_2(x_j)h_x h_y + \sum_{j\in\partial\Omega}\eta_1(x_j)\eta_2(x_j)\frac{h_x h_y}{2},
\end{aligned}
\tag{6.9}
$$

where $\bar{x}_j$ is a vertex of a triangle in the reference coordinate system and $\bar{\omega}_j$ is the corresponding weight. For nodal numerical quadrature on triangles we usually take $\bar{\omega}_j = \frac{1}{6}$ in the reference triangle element, which corresponds to a unit isosceles triangle with a right angle. J_κ is the *Jacobi* transformation matrix between the original and reference coordinate system, so $J_\kappa = 2\Delta_\kappa$. In the last step of (6.9) we use the fact that for the case of a uniform mesh in a square domain, every grid point is the common vertex of the surrounding six triangle elements, the only exception being on the boundary, where three triangles share one grid point.

As in the 1D case, due to the periodic domain $\Omega = (S^1)^2$, we can then get rid of the second term (6.9) on $\partial\Omega$. Thus we have

$$
(\eta_1,\eta_2) = \sum_{j}\eta_1(x_j)\eta_2(x_j)h_x h_y = (\eta_1,\eta_2)^h,
\tag{6.10}
$$

which proves that the standard inner product (η_1,η_2) is equivalent to the discrete inner product $(\eta_1,\eta_2)^h$ in terms of the nodal numerical quadrature. As for the definition (6.8), because it is one kind of special numerical quadrature for the diffusion terms, the equivalence is trivial on a uniform rectangle mesh in a regular domain. These imply that the standard mixed finite element approximation which finds $(U,W)\in S^h\times S^h$ such that

$$
\begin{cases}
(U_t,\chi) + (D(U)\nabla W,\nabla\chi) &= 0, \quad \forall\chi\in S^h, \\
(\nabla U,\nabla\chi) &= (W,\chi), \quad \forall\chi\in S^h,
\end{cases}
\tag{6.11}
$$

is equivalent to the 2D positivity-preserving finite difference scheme presented in [78] when corresponding numerical quadrature formulas are employed.

But for an arbitrary unstructured mesh, the particular numerical quadrature form (6.8) will not again hold. Even though (6.10) still works for an unstructured mesh, the exact equivalence between the positivity-preserving finite difference scheme and the general finite element discretization no longer holds, and whether or not there is an asymptotic equivalence requires further investigation. Regardless, the numerical simulations with anisotropic adaptive meshes in the latter part of this paper indicate that the general finite element method works well on unstructured meshes for the thin film flow problem with moving contact line.

In the arbitrary case the PPS cannot be implemented because it is only based on a finite difference approach and because it requires that the mesh be tessellated. In contrast, the finite element method is straightforward to implement on an arbitrary unstructured mesh.

In the next sections, we adopt the precursor film model with the initial profile $u(x,0)+b$ (b relatively small), but implement a general finite element method on unstructured anisotropic locally refined meshes. It will be numerically shown that the thickness is still always positive for smaller n in (4.4) on the unstructured meshes, where a positive numerical solution is also guaranteed in [36] by using PPS and a regularization method (4.7). We shall see that this generalizes to the finite element method for (4.4) on any arbitrary mesh, but normally regularization is not necessary.

7. Time stepping

Explicit time stepping procedures present a major challenge for fourth order equations, with the general time step restriction that Δt be on the order of h^4. This restriction, far worse than the corresponding stability requirement for second order problems, is particularly severe when fine grids are needed to obtain highly resolved solutions accurately, as is the case for the thin film problem. Reduction of the thin film equation from fourth order form to a second order system (5.1) is critical when the explicit time stepping is adopted.

Alternatively, we can use implicit schemes, which require the inversion of a linear system at each time step. This can become computationally expensive for fine grids. Fortuitously, mesh adaptivity can substantially reduce the number of grid points and thus computational cost. This has been our experience for the thin film problem.

We employ higher order backward differentiation formulas (BDF) to approximate the time derivative. These schemes are well known for being stable with large time step size and have been widely used for solving stiff ODEs and differential-algebraic equations. For completeness, some BDF formulas we use are summarized for the model differential equation

$$\frac{\partial u}{\partial t} = f.$$

Denoting the time level by a subscript j, we have

- Second order BDF formula:

$$\frac{1}{2\Delta t}(3u_j - 4u_{j-1} + u_{j-2}) = f_j. \tag{7.1}$$

- Fourth order BDF formula:

$$\frac{1}{12\Delta t}(25u_j - 48u_{j-1} + 36u_{j-2} - 16u_{j-3} + 3u_{j-4}) = f_j. \tag{7.2}$$

For multi-step BDF methods we also need difference formulas for several initial steps. In order not to lose accuracy, we use several high order finite difference schemes.

- Crank–Nicolson (second order accuracy) using two time levels:

$$\frac{1}{\Delta t}(u_j - u_{j-1}) = \frac{1}{2}(f_j + f_{j-1}). \tag{7.3}$$

- Fourth order accuracy using three time levels:

$$\frac{1}{2\Delta t}(u_j - u_{j-2}) = \frac{1}{6}(f_j + 4f_{j-1} + f_{j-2}). \tag{7.4}$$

- Fourth order accuracy using four time levels:

$$\frac{1}{24\Delta t}(17u_j - 9u_{j-1} - 9u_{j-2} + u_{j-3}) = \frac{1}{4}(f_j + 3f_{j-1}). \tag{7.5}$$

Being more accurate and stabler than the lower order ones, the fourth order BDF scheme can enlarge the time step size and still preserve stability. While an adaptive time stepping procedure can also be used as in [54] for strongly nonlinear problems with solution having sharp interfaces, this kind of adaptive time stepping method is not necessarily efficient if used together with local mesh refinement. Nevertheless, the dependence of the time step size on the mesh size h and the width of the sharp interface ε is far from straightforward to interpret and requires investigation. For simplicity, results given here are for constant time step size Δt.

8. Numerical results

In this section, we consider the thin film fluid flow equation (4.4) where $\alpha = 1$, $\beta = 0$, $\gamma = 1$, and consider the thin film dynamics in a rectangle domain whose lateral dimension is comparable to the wavelength of maximum growth. We take the domain size as 100×16 in the following simulations.

The experiments and numerical simulations done in [54] demonstrate that, when a thin film flows down an inclined plane, after some time the initially straight line where the liquid, gas and solid phase meet becomes

unstable with respect to transverse perturbations, and finger-shaped patterns are generated. We consider mainly the one finger case in our numerical simulation, where the narrow domain results allow for insight into the fluid instability and the effectiveness of the local mesh refinement.

The simulations are performed using as initial condition

$$x_f(y) = x_{f0} - A_0 \cos(2\pi y/\lambda_0), \tag{8.1}$$

where x_{f0} is the unperturbed position for the front at $t = 0$, which is perturbed by a single transverse mode characterized by the wavelength $\lambda_0 = L_y$. The perturbation is characterized by a small amplitude A_0 and a phase such that this initial condition satisfies $\frac{\partial u}{\partial y} = 0$ at $y = 0$, L_y. As a representative case, we use $L_y = 16$, and the size of the computational domain varies between $L_x = 50$ and $L_X = 100$. Most of the computations are performed using a precursor film of thickness $b = 0.01$.

The numerical results for our adaptive mixed finite element approximation (5.3) and anisotropic mesh refinement technique in the case of $n = 3$, $b = 0.01$ are shown in Figs. 4–11. The precursor film model with $u(x, 0) + b$ as the initial film is employed, but the regularization method is unnecessary for this power n.

The main feature of the profile $u(x, t)$ is the presence of a bump near the contact line. This bump, resulting from the fluid accumulation behind the front, is due to the fact that the viscous stress on the plane is much greater in the contact line region than in the fluid bulk. The increased viscous stress is balanced by the component of the bump weight in the down-slope direction. The presence of the bump is a necessary condition for the instability of the fluid to small perturbations in the transverse direction [68,18,67].

In the above simulations, the smallest edge length of the anisotropic mesh around the moving contact line is about 0.1, in contrast with the uniformly distributed mesh size of 8.0 in the flat solution regions. The smallest aspect ratio of mesh elements around the moving contact line is roughly 1:7. (Incidentally, note that the initial mesh is generated by an automatic mesh generator, so that a slight asymmetry is present in the adaptive meshes computed using a posteriori error estimation, although the approximate solution itself is basically symmetric in y.) Corresponding to these mesh sizes, the total number of triangles for each step is on average around 1600. In the domain 100×16, having commensurate mesh sizes in the moving contact region would require around 320,000 elements with a uniform triangulation, which is about 200 times larger. The computational cost for such a uniform mesh would easily be prohibitive. For example, in [54], using a uniform mesh of size 0.5, five times the mesh size 0.1, the computing times on the fastest available workstations (R12000 CPU) vary between 15 and 20 h for the smaller typical simulations to a couple of weeks for large simulations in order to get relatively accurate solutions.

In order to show quantitatively the improvement of our adaptive algorithm over a standard method on uniform grids, we carry out the following numerical experiments: compute the solution to the same problem with the adaptive FEM on an adaptive mesh and the standard FEM on a uniform mesh. In the first step, we try to let the meshes have the same number of triangles. Then we double the resolution for the uniform mesh only, so the number of triangles is quadruple that of the adaptive mesh. By comparing these two solutions for the

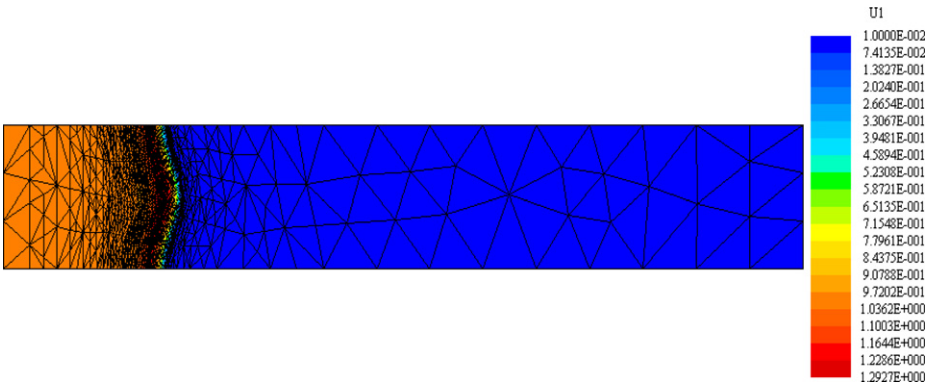

Fig. 4. Global solution at $T = 1$ for $n = 3$, $b = 0.01$.

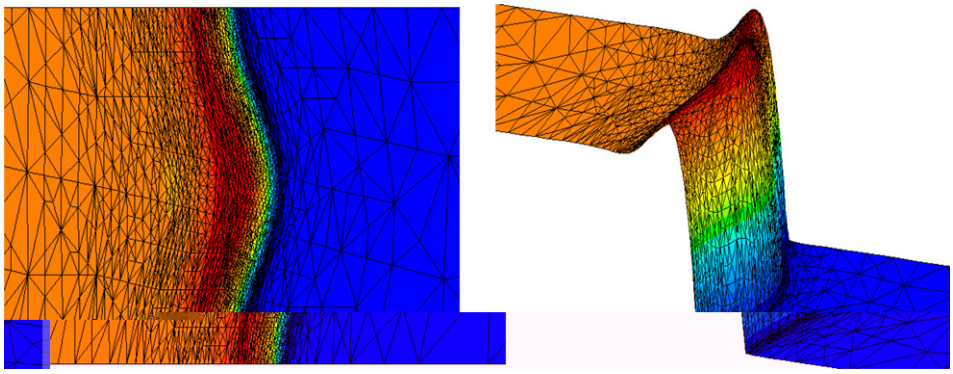

Fig. 5. Local solution at $T = 1$ for $n = 3$, $b = 0.01$.

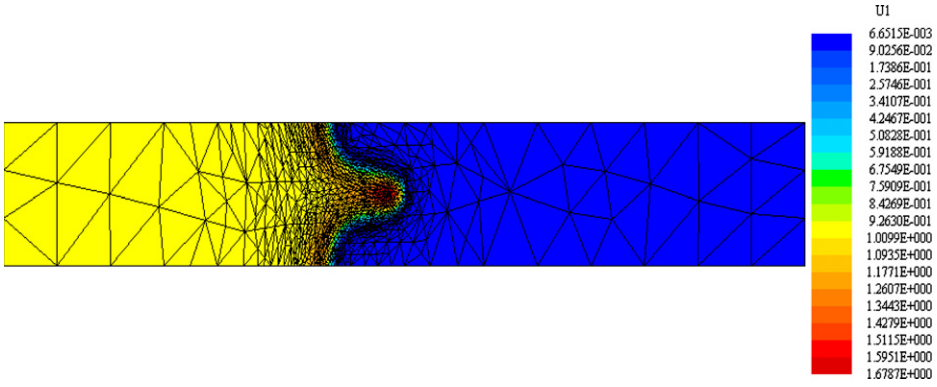

Fig. 6. Global solution at $T = 25$ for $n = 3$, $b = 0.01$.

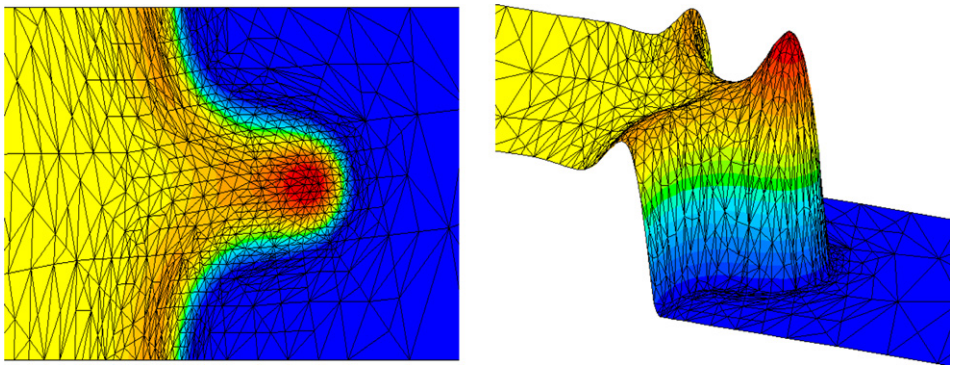

Fig. 7. Local solution at $T = 25$ for $n = 3$, $b = 0.01$.

standard FEM with the solution for the adaptive FEM at the same time point, we can better understand how much more accurate our adaptive method is in contrast to the standard FEM.

First of all, in the domain 100×16 we take the uniform mesh with 1722 triangles to match our adaptive mesh density (with 1537 triangles) at $T = 20$. Secondly, we globally refine the uniform mesh with half the mesh size to get 6752 triangles. After doing the computation with the standard FEM on both meshes, we get the

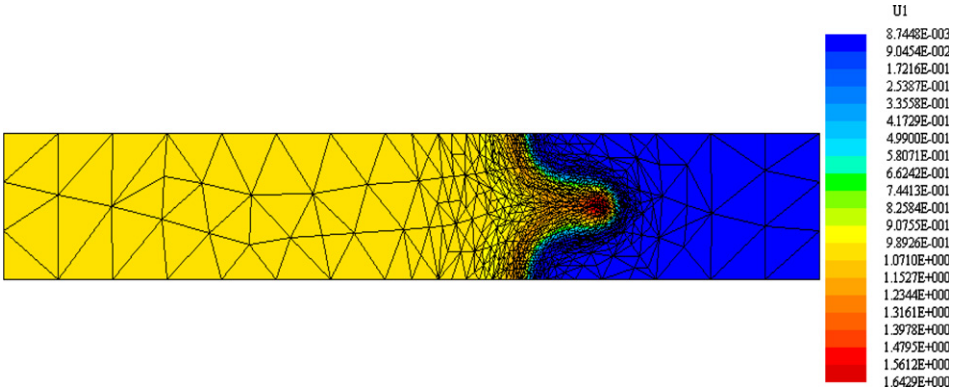

Fig. 8. Global solution at $T = 50$ for $n = 3$, $b = 0.01$.

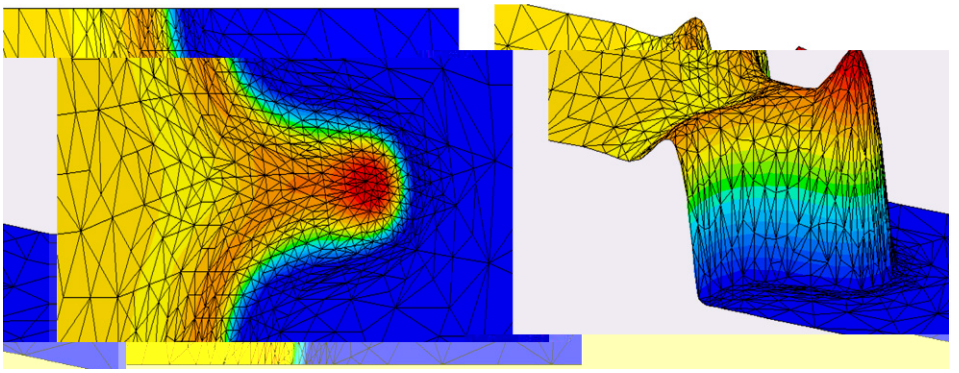

Fig. 9. Local solution at $T = 50$ for $n = 3$, $b = 0.01$.

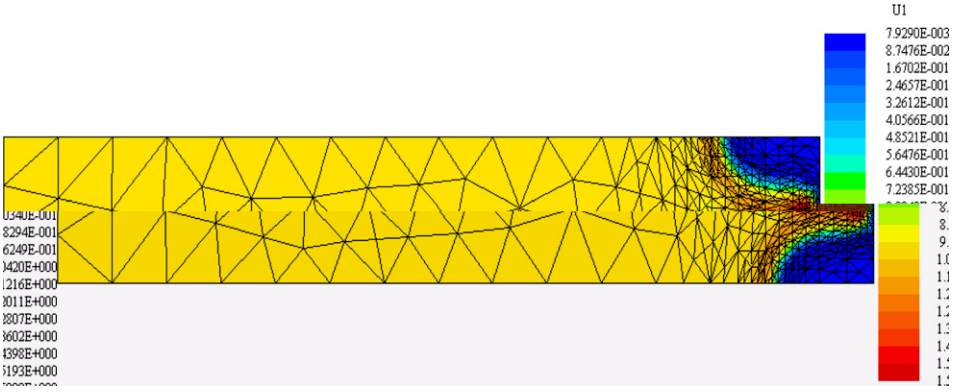

Fig. 10. Global solution at $T = 75$ for $n = 3$, $b = 0.01$.

results shown in Figs. 12–14, together with the solution on the adaptive mesh at time point $T = 20$. For this value of T, the differences in speed and shape of the contact line are apparent for these approximate solutions.

The local blow-up behaviors are shown in Fig. 15. Figs. 16–18 show the corresponding 1-dimensional profiles of the moving contact lines.

These pictures indicate that when the mesh size is decreased, the solution for the standard FEM is approximating that for the adaptive FEM on the coarsest mesh level. A measure of accuracy for the uniform mesh solutions is also exhibited in Table 1, where the following calculation of the approximation error is computed:

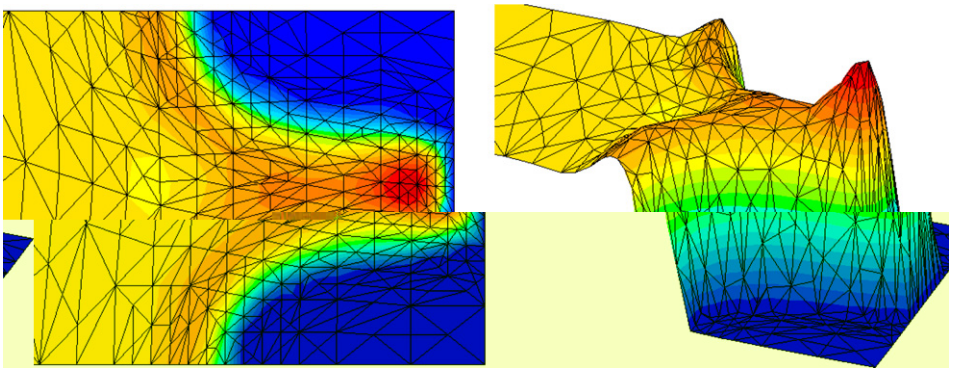

Fig. 11. Local solution at $T = 75$ for $n = 3$, $b = 0.01$.

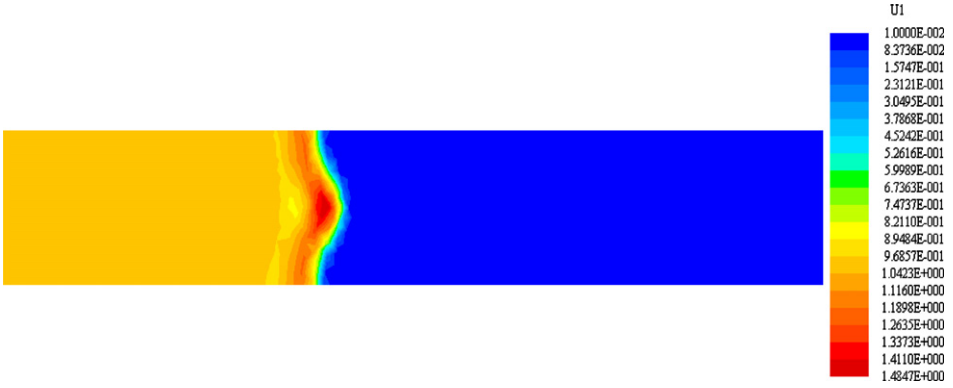

Fig. 12. Global solution at $T = 20$ for $n = 3$,$b = 0.01$ on 1722 uniform grids.

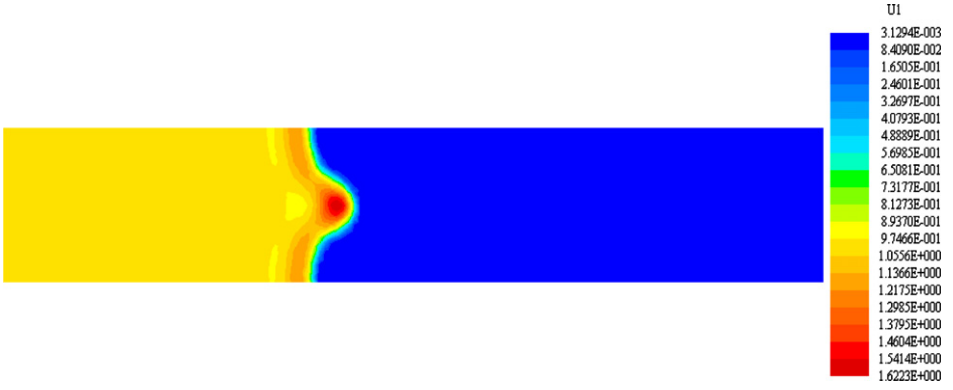

Fig. 13. Global solution at $T = 20$ for $n = 3$, $b = 0.01$ on 6752 uniform grids.

$$\text{approximation error} = \frac{\|u_{\text{uniform}} - u_{\text{adaptive}}\|_{L^2}}{\|u_{\text{adaptive}}\|_{L^2}},$$

where u_{uniform} denotes the solution for the uniform mesh with different resolutions, and u_{adaptive} denotes the solution for the adaptive mesh with the almost same number of triangles as for the coarsest uniform mesh.

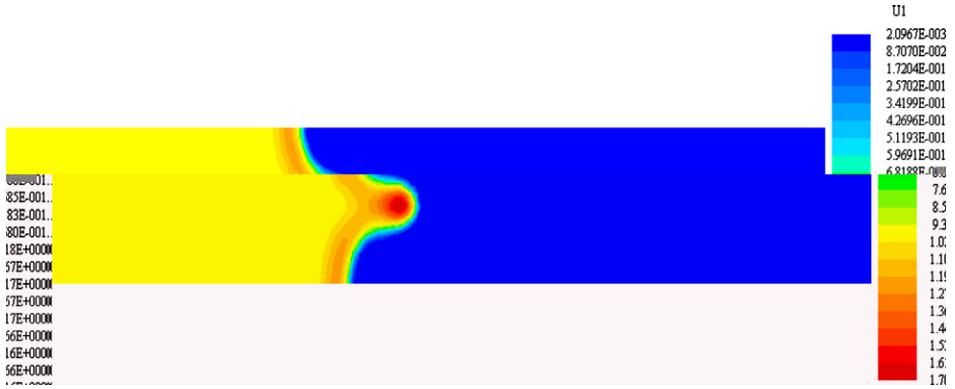

Fig. 14. Global solution at $T = 20$ for $n = 3$, $b = 0.01$ on 1537 adaptive grids.

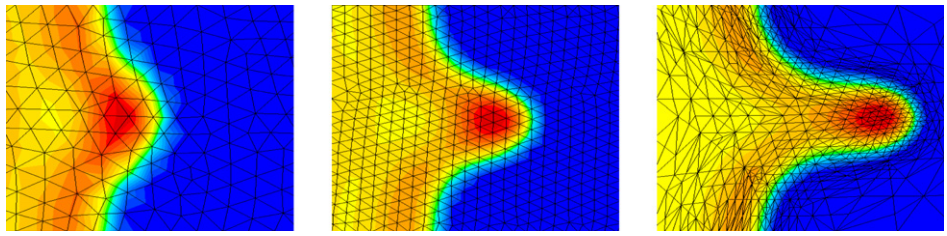

Fig. 15. The comparison of standard FEM and adaptive FEM on two uniform meshes and one adaptive mesh, respectively, at $T = 20$ for $n = 3$, $b = 0.01$.

In the calculation, we interpolate the solution for the uniform mesh onto that for the adaptive mesh. Since we are only using linear finite element to compute the solutions, linear interpolation is used for these error calculations.

Table 1 indicates that at $T = 20$, a typical time point, the approximations of the solutions for the uniform meshes become close to the solution for the adaptive mesh with the almost same number of triangles as for the coarsest uniform mesh, as the uniform meshes are refined.

While the convergence theories for the adaptive and standard numerical methods are quite different, specific accuracy comparisons between the solutions on adaptive and unadaptive meshes are complicated, but based on the decreasing approximation error in Table 1 and the approximate moving contact line images in the figures, we can conclude that our adaptive algorithm obtains an accuracy which the standard FEM would achieve only with substantially more elements. In summary, our adaptive finite element method works very well around the contact line, with the locally refined meshes appearing to closely match the contact line. Since only the meshes which distribute elements along the moving contact line are able to resolve the singularity and improve the accuracy, getting commensurate numerical results with a uniform mesh would be extremely difficult if not impossible.

We have also applied the AFEM for various values of the power n and precursor thickness b successfully without need for the regularization (4.7). This is in contrast with previous results where regularization together with PPS on fine grids are used for the case of small n and b (e.g. see [54]).

Realistic results obtained for $b = 0.001$ and $n = 3$ are shown in Figs. 19 and 20.

For the case $n = 1$, we get the results shown in Figs. 21 and 22 for the small precursor film $b = 0.001$. We see from these results that for the case of small $n(=1)$, the bump near the contact line is gradually smeared, viz., stabler fluid flow is obtained with a smaller power n. Again, no regularization (4.7) is used to resolve the contact line accurately with the help of an adaptive local refinement. This is still in contrast with the numerical experiments of [54], where it is used for $n \leqslant 2$.

Fig. 16. Moving contact line on coarse uniform mesh.

Fig. 17. On fine uniform mesh with quadruple number of elements as coarse uniform mesh.

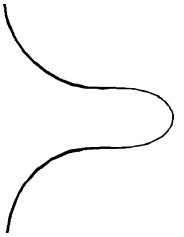

Fig. 18. On adaptive mesh with almost same number of elements as coarse uniform mesh.

Table 1
The approximation error at $T = 20$ to the solution on adaptive mesh with the almost same number of triangles as for the coarsest uniform mesh

No. of elements of uniform meshes	Approximation errors
1722	0.09237
6752	0.03181
27258	0.01415

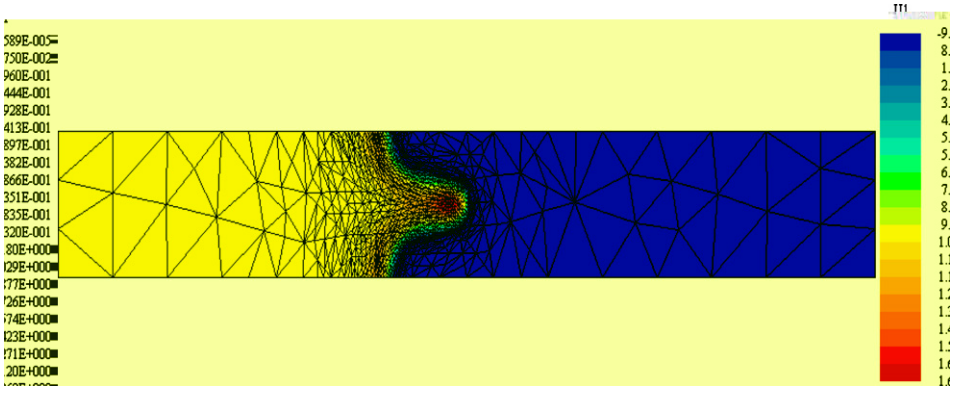

Fig. 19. Global solution at $T = 25$ for $n = 3$, $b = 0.001$.

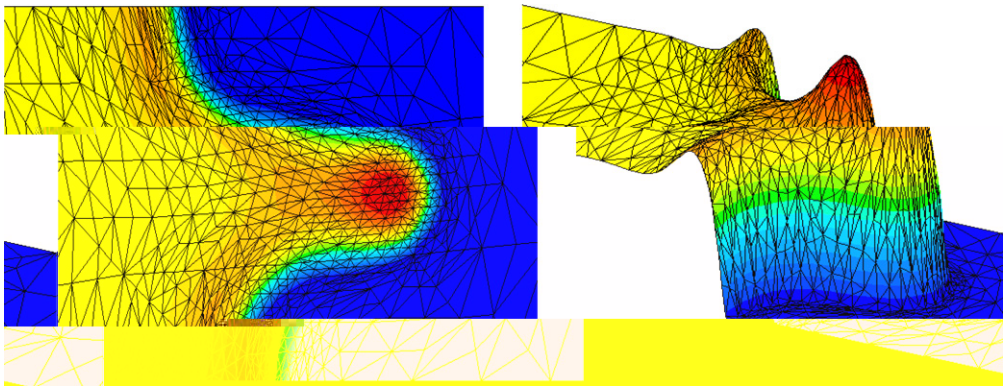

Fig. 20. Local solution at $T = 25$ for $n = 3$, $b = 0.001$.

According to our theory, the mesh density and anisotropy depend on the singularity of the solution u. For thin film flow, the moving contact line is the place where the singular sharp interface happens. The more dramatic the singularity is, the more anisotropic and denser the adaptive mesh will get, as is measured by the Hessian matrix of the solution u. For the case of a smaller power $n < 2$, since the moving contact lines are no longer sharp after some time, the corresponding adaptive meshes are not dramatically anisotropic.

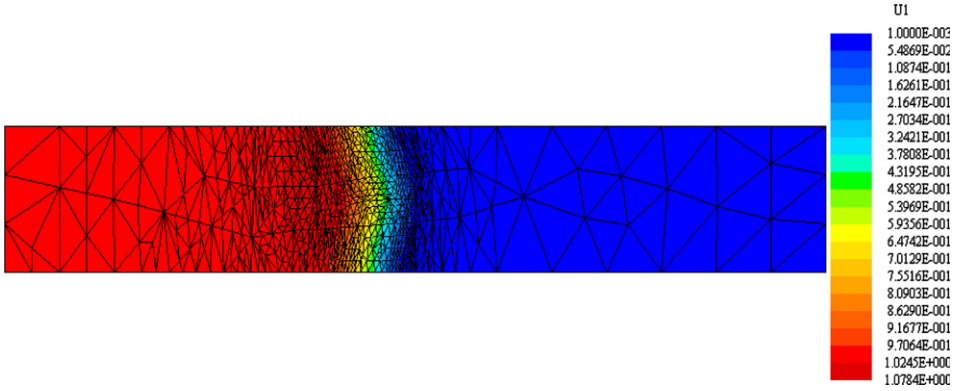

Fig. 21. Global solution at $T = 25$ for $n = 1$, $b = 0.001$.

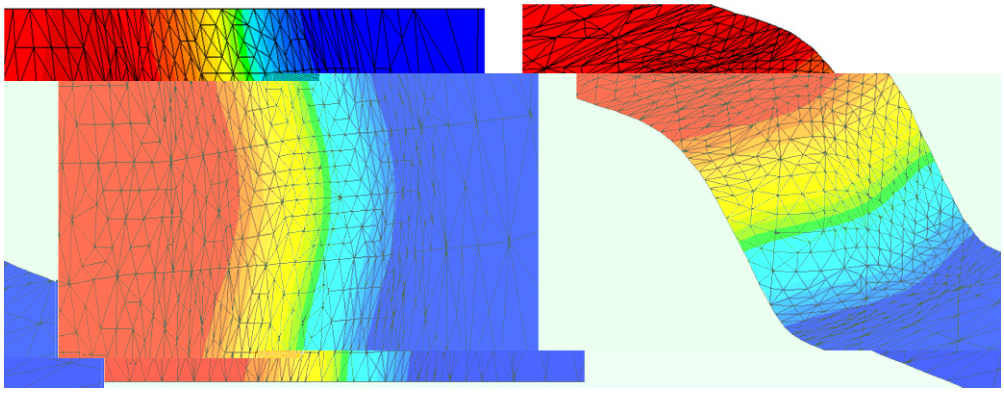

Fig. 22. Local solution at $T = 25$ for $n = 1$, $b = 0.001$.

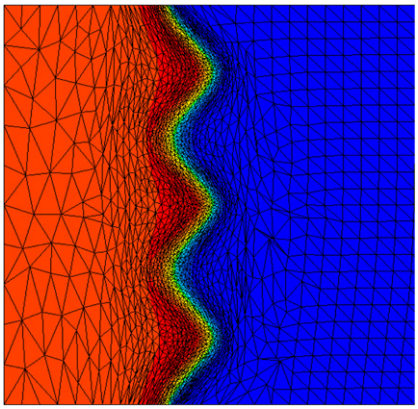

Fig. 23. Multiple fingers on the adaptive mesh.

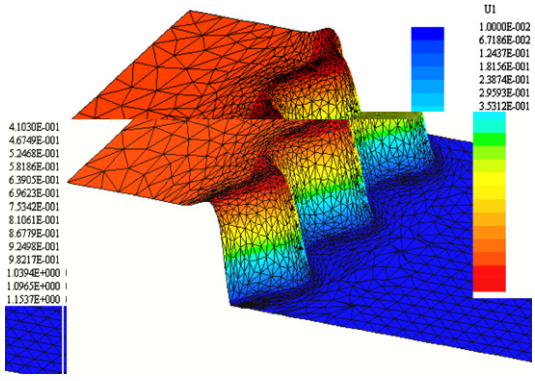

Fig. 24. Multiple fingers in 3D visualization.

As a final example, we show numerical results of a multi-finger phase computation. Figs. 23 and 24 give the computed solution for the thin film fluid flow with a three fingers phase.

9. Conclusions

In this paper, we have presented a detailed implementation of an adaptive finite element method (AFEM) and demonstrated that it can perform very well on a challenging problem like the thin film problem with a moving contact line. The results for our anisotropic AFEM are consistent with those obtained previously, most using the positivity-preserving finite difference method [19,14,1,53,35,54]. It requires much fewer grid points for comparable accuracy than with the uniform grid (which still needs roughly the same density around the contact line).

We prove a close relationship between the finite element method on a structured mesh and one kind of positivity-preserving scheme (PPS). Unlike the PPS and other finite difference methods which require structured meshes, the finite element method is straightforward to apply for unstructured grids. Based on the numerical results, good local refinement gives in turn an accurate approximation along the contact lines, which stays positive without the need for any regularization, even for an extremely low precursor film case.

Using an anisotropic locally refined mesh, along with local smoothing and edge swapping techniques to improve the mesh quality, we are able to get very long thin elements along finger-shaped contact lines by concentrating the grid points on the sharp interface. Theoretically, this kind of anisotropic mesh is optimal or quasi-optimal in the sense of reducing the interpolation error. Practically, it is able to obtain a numerical

solution at minimal cost since the grid points are only concentrating around the moving contact line where the solution is of greatest interest.

To our knowledge, this is the first application of a fully adaptive finite element method which produces both anisotropic and unstructured locally refined meshes to solve the fourth order thin film fluid flow problem in 2D. The successful numerical simulations demonstrate that our adaptive finite element method has good performance, even for the case of small powers $n < 3$ and small thickness of precursor film $b \leqslant 10^{-3}$, without the need to use the regularization method. This is in contrast to the previously used finite difference methods, even with the PPS, for small values of the parameter sets (n,b) on related problems [53,35,54]. In the recent paper [10], some interesting theoretical and computational challenges for similar problems are laid out, and a finite element method using nonuniform but isotropic meshes is used to study them numerically. A further study of these problems and comparison of this and the other popular algorithms to our adaptive approach should prove most fruitful and is a matter we intend to pursue.

Acknowledgements

P.S. and R.D.R. are partially supported by NSERC Grant A8781, Canada. J.X. is partially supported by NSF Grant #DMS-0074299 and Center for Computational Mathematics and Applications, Penn State.

References

[1] R. Almgren, A.L. Bertozzi, M.P. Brenner, Stable and unstable singularities in the unforced hele-shaw cell, Phys. Fluids 8 (1996) 1356–1370.
[2] T. Apel, Anisotropic finite elements: local estimates and applications, in: J.P. Pier (Ed.), Advances in Numerical Mathematics, Stuttgart: Teubner 1999, Birkhauser, 1999.
[3] I. Babuška, W. Rheinboldt, Error estimates for adaptive finite element computations, SIAM J. Numer. Anal. 15 (1978) 736–754.
[4] I. Babuška, T. Strouboulis, The finite element method and its reliability, in: Numerical Mathematics and Scientific Computation, Oxford Science Publication, 2001.
[5] R.E. Bank, A.H. Sherman, A. Weiser, Refinement algorithms and data structures for regular local mesh refinement, in: R. Stepleman et al. (Eds.), Scientific Computing, IMACS/North-Holland Publishing Company, Amsterdam, 1983, pp. 3–17.
[6] R.E. Bank, R.K. Smith, Mesh smoothing using a posteriori error estimates, SIAM J. Numer. Anal. 34 (1997) 979–997.
[7] R.E. Bank, J. Xu, Asymptotically exact a posteriori error estimators, part I: grids with superconvergence, SIAM J. Numer. Anal. 41 (2003) 2294–2312.
[8] R.E. Bank, J. Xu, Asymptotically exact a posteriori error estimators, part II: general unstructured grids, SIAM J. Numer. Anal. 41 (2003) 2313–2332.
[9] J. Barrett, J.F. Blowey, H. Garcke, Finite element approximation of a fourth order degenerate parabolic equation, Numer. Math. 80 (1998) 525–556.
[10] J. Becker, G. Grün, The thin film equation: recent advances and some new perspectives, J. Phys. Condens. Mat. 17 (2005) S291–S307.
[11] E. Beretta, M. Berstch, R.D. Passo, Nonnegative solutions of a fourth order nonlinear degenerate parabolic equation, Arch. Rational Mech. Anal. 129 (1995) 175–200.
[12] F. Bernis, Viscous flows, fourth order nonlinear degenerate parabolic equations and singular elliptic problems, in free boundary problems: theory and applications, in: Pitman Research Notes in Mathematics, vol. 323, 1995, pp. 40–56.
[13] F. Bernis, A. Friedman, Higher order nonlinear degenerate parabolic equations, J. Differ. Equat. 83 (1990) 179–206.
[14] A. Bertozzi, Loss and gain of regularity in a lubrication equation for thin viscous films, in Free boundary problems: theory and applications, in: J.I. Díaz, M.A. Herrero, A. Liñán, J.L. Vázquez (Eds.), Pitman Research Notes in Mathematics, vol. 323, Longman, Harlow, 1995.
[15] A. Bertozzi, A. Münch, M. Shearer, K. Zumbrun, Stability of compressive and undercompressive thin film travelling waves, Eur. J. Appl. Math. 12 (2001) 253–291.
[16] A. Bertozzi, M. Pugh, The lubrication approximation for thin viscous films: regularity and long time behaviour of weak solutions, Commun. Pure Appl. Math. 49 (1996) 85–123.
[17] A.L. Bertozzi, Symmetric singularity formation in lubrication-type equations for interface motion, SIAM J. Appl. Math. 56 (1996) 681–714.
[18] A.L. Bertozzi, M.P. Brenner, Linear stability and transient growth in driven contact lines, Phys. Fluids 9 (1997) 530–539.
[19] A.L. Bertozzi, M.P. Brenner, T.F. Dupont, L.P. Kadanoff, Singularities and similarities in interface flow, in: L. Sirovich (Ed.), Trends and Perspectives in Applied Mathematics, Appl. Math. Sci, vol. 100, Springer Verlag, New York, 1994.
[20] H. Borouchaki, M. Castro-Diaz, P. George, F. Hecht, B. Mohammadi, Anisotropic adaptive mesh generation in two dimensions for cfdProceedings of the 5th International Conference On Numerical Grid Generation in Computational Field Simulations, vol. 3, Mississppi State University, 1996, pp. 197–206.
[21] H.G. Burchard, Splines (with optimal knots) are better, Appl. Anal. 3 (1974) 309–319.

[22] J. Cahn, C. Elliott, A. Novick-Cohen, The Cahn–Hilliard equation with a concentration dependent mobility: motion by minus the laplacian of the mean curvature, Eur. J. Appl. Math. 7 (1996) 287–301.
[23] P. Carles, A.M. Cazabat, On the origin of the bump in the profile of surface-tension-gradient-driven spreading films, Mat. Res. Soc. Symp. Proc. 248 (1990).
[24] A.M. Cazabat, F. Heslot, S.M. Troian, P. Carles, Finger instability of this spreading films driven by temperature gradients, Nature 346 (1990) 824–826.
[25] C.M. Chen, Y. Huang, High Accuracy Theory of Finite Element Methods, Hunan, Science Press, Hunan, China, 1995 (in Chinese).
[26] L. Chen, Robust and accurate algorithms for solving anisotropic singularities, Ph.D. thesis, Pennsylvania State University, 2005.
[27] L. Chen, P. Sun, J. Xu, Multilevel homotopic adaptive finite element methods for convection dominated problems, in: Domain Decomposition Methods in Science and Engineering, Lecture Notes in Computational Science and Engineering 40 (2004) 459–468.
[28] L. Chen, P. Sun, J. Xu, Optimal anisotropic simplicial meshes for minimizing interpolation errors in L^p-norm, Math. Comp. 76 (2007) 179–204.
[29] P. Constantin, T.F. Dupont, R.E. Goldstein, L.P. Kadanoff, M.J. Shelley, S.-M. Zhou, Droplet breakup in a model of the Hele–Shaw cell, Phys. Rev. E 47 (1993) 4169–4181.
[30] E. D'Azevedo, Optimal triangular mesh generation by coordinate transforma tion, SIAM J. Sci. Stat. Comp. 12 (1991) 755–786.
[31] E. D'Azevedo, R. Simpson, On optimal interpolation triangle incidences, SIAM J. Sci. Stat. Comp. 6 (1989) 1063–1075.
[32] E.F. D'Azevedo, Optimal triangular mesh generation by coordinate transformation, SIAM J. Sci. Stat. Comp. 12 (1991) 755–786.
[33] C. de Boor, Good approximation by splines with variables knots II, in: G.A. Watson (Ed.), Proceedings of the 11th International Conference on Numerical Methods in Fluid Dynamics, of Lecture Notes in Mathematics, vol. 363, Springer-Verlag, Dundee, Scotland, 1974, pp. 12–20.
[34] C. de Boor, Good Approximation by Splines with Variable Knots. Int. Seines Numer. Math, vol. 21, Birkhauser Verlag, Basel, 1973, pp. 57–72.
[35] J.A. Diez, L. Kondic, Computing three-dimensional thin film flows including contact lines, J. Comp. Phys. 183 (2002) 274–306.
[36] J.A. Diez, L. Kondic, A. Bertozzi, Global models for moving contact lines, Phys. Rev. E 63 (2000) 011208-1–011208-13.
[37] V. Dolejsi, Anisotropic mesh adaptation for finite volume and finite element methods on triangular meshes, Comput. Visual. Sci. 1 (1998) 165–178.
[38] W. Dörfler, A convergent adaptive algorithm for poissons equation, SIAM J. Numer. Anal. 33 (1996) 1106–1124.
[39] C. Elliott, H. Garcke, On the Cahn–Hilliard equation with degenerate mobility, SIAM J. Math. Anal. 27 (1996) 401–423.
[40] C. Elliott, H. Garcke, Diffusional phase transitions in multicomponent systems with a concentration dependent mobility matrix, Physica D 109 (1997) 242–256.
[41] D.A. Field, Laplacian smoothing and delaunay triangulation, Commun. Appl. Numer. Method 4 (1988) 709–712.
[42] L. Formaggia, S. Perotto, New anisotropic a priori error estimates, Numer. Math. 89 (4) (2001) 641–667.
[43] N. Fraysse, G.M. Homsy, An experimental study of rivulet instabilities in centrifugal spin coating of viscous newtonian and non-newtonian fluids, Phys. Fluids 6 (1994) 1491–1504.
[44] L.A. Freitag, M.T. Jones, P.E. Plassmann, An efficient parallel algorithm for mesh smoothing, in: Proceedings of the 4th International Meshing Roundtables, Sandia Labs, 1995, pp. 47–58.
[45] H. Greenspan, On the motion of a small viscous droplet that wets a surface, J. Fluid Mech. 84 (1978) 125–143.
[46] G. Grün, Degenerate parabolic differential equations of fourth order and a plasticity model with nonlocal hardening, Z. Anal. Anwend. 14 (1995) 541–574.
[47] G. Grün, M. Rumpf, Simulation of singularities and instabilities arising in thin film flow, Eur. Appl. Math. 12 (2001) 293–320.
[48] W.G. Habashi, M. Fortin, J. Dompierre, M. Vallet, D. Ati-Ali-Yahia, Y. Bourgault, M.P. Robichaud, A. Tam, S. Boivin, Anisotropic mesh optimization for structured and unstructured meshes, in: Proceedings of the 28th Computational Fluid Dynamics Lecture Series, von Karman Institute, March 1997.
[49] W. Huang, Variational mesh adaptation: isotropy and equidistribution, J. Comput. Phys. 174 (2001) 903–924.
[50] W. Huang, W. Sun, Variational mesh adaptation II: error estimates and monitor functions, J. Comput. Phys. 184 (2003) 619–648.
[51] H. Huppert, Flow and instability of a viscous current down a slope, Nature 300 (1982) 427–429.
[52] J.M. Jerrett, J.R.D. Bruyn, Finger instability of a gravitationally driven contact line, Phys. Fluids A 4 (1992) 234–242.
[53] L. Kondic, Instabilities in gravity driven flow of thin fluid, SIAM Rev. 45 (2003) 95–115.
[54] L. Kondic, J. Diez, Pattern formation in the flow of thin films down an incline:constant flux configuration, Phys. Fluids 13 (2001) 3168–3184.
[55] R. Kornhuber, R. Roitzsch, On adaptive grid refinement in the presence of internal or boundary layers, IMPACT Comput. Sci. Eng. 2 (1990) 40–72.
[56] G. Kunert, A a posteriori error estimation for anisotropic tetrahedral and triangular finite element meshes, Ph.D. Thesis, 1999.
[57] C.L. Lawson, Software for C^1 surface interpolation, in: J.R. Rice (Ed.), Mathematical Software III, Academic Press, 1977, pp. 161–194.
[58] M.A. Lewis, Spatial coupling of plant and herbivore dynamics: the contribution of herbivore dispersal to transient and persistent waves of damage, Theor. Popul. Biol. 45 (1994) 277.
[59] B. Li, Z. Zhang, Analysis of a class of superconvergence patch recovery techniques for linear and bilinear finite elements, Numer. Method PDEs 15 (1999) 151–167.
[60] P. Morin, R.H. Nochetto, K.G. Siebert, Convergence of adaptive finite element methods, SIAM Rev. 44 (2002) 631–658.
[61] A. Münch, B. Wagner, Numerical and asymptotic results on the linear stability of a thin film spreading down a slope of small inclination, Eur. J. Appl. Math. 10 (1999) 297–318.

[62] E. Nadler, Piecewise linear best L_2 approximation on triangulations, Approx. Theory V (1986) 499–502.
[63] R. Raviart, J. Thomas, A mixed finite element method for 2nd order elliptic problems, in: Mathematical Aspects of the Finite Element MethodLecture Notes in Mathematics, vol. 606, Springer Verlag, New York, 1977.
[64] M.C. Rivara, Mesh refinement processes based on the generalized bisection of simplices, SIAM J. Numer. Anal. 21 (1984) 604–613.
[65] M. Shephard, M. Georges, Automatic three-dimensional mesh generation by the finite octree technique, Internat. J. Numer. Method Eng. 32 (1991) 709–749.
[66] N. Silvi, E.B. Dussan, On the rewetting of an inclined solid surface by a liquid, Phys. Fluids 28 (1985) 5–7.
[67] M. Spaid, G. Homsy, Stability of newtonian and viscoelastic dynamic contact line, Phys. Fluids 8 (1996) 460–478.
[68] S.M. Troian, E. Herbolzheimer, S.A. Safran, J.F. Joanny, Fingering instabilities of driven spreading films, Europhys. Lett. 10 (1989) 25–30.
[69] S.M. Troian, X.L. Wu, S.A. Safran, Fingering instability in thin wetting films, Phys. Rev. Lett. 62 (1989) 1496–1499.
[70] V. Dolejsi, Adaptive methods for the numerical solution of the compressible navier-stokes equations, in: K.D. Papailiou, D. Tsahalis, J. Pe'riaux, C. Hirsch, M. Pandolfi (Eds.), Computational Fluid Dynamics'98, ECCOMAS, vol. 1, John Wiley and Sons, 1998, pp. 393–397.
[71] R. Verfürth, A review of a posteriori error estimation and adaptive mesh-refinement techniques, Wiley-Teubner, Chichester, UK, 1996.
[72] L. Wahlbin, Superconvergence in Galkerkin Finite Element Methods, Springer Verlag, Berlin, 1995.
[73] T.P. Witelski, M. Bowen, ADI schemes for higher-order nonlinear diffusion equations, Appl. Numer. Math. 45 (2003) 331–351.
[74] J. Xu, Z. Zhang, Analysis of recovery type a posteriori error estimators for mildly structured grids, Math. Comp. 73 (2004) 1139–1152.
[75] Z. Zhang, Ultraconvergence of the patch recovery technique II, Math. Comp. 69 (1999) 141–158.
[76] N.A. Zhang Z, A meshless gradient recovery method, part i: superconvergence recovery. Research Report #2, Department of Mathematics, Wayne State University, 2002.
[77] N.A. Zhang Z, A posteriori error estimates based on polynomial preserving recovery. Research Report #9, Department of Mathematics, Wayne State University, 2002.
[78] L. Zhornitskaya, A.L. Bertozzi, Positivity preserving numerical schemes for lubrication-type equations, SIAM J. Numer. Anal. 37 (2000) 523–555.
[79] O. Zienkiewicz, J. Zhu, The superconvergence patch recovery and a posteriori error estimates. part 1: the recovery techniques, Internat. J. Numer. Method Eng. 33 (1992) 1331–1364.
[80] O. Zienkiewicz, J. Zhu, The superconvergence patch recovery and a posteriori error estimates. part 2: error estimates and adaptivity, Int. J. Numer. Methods Eng. 33 (1992) 1365–1382.