Journal of Computational Physics 349 (2017) 328–350

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

# Monge–Ampére simulation of fourth order PDEs in two dimensions with application to elastic–electrostatic contact problems

Kelsey L. DiPietro, Alan E. Lindsay *

*Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN 46556, USA*

ARTICLE INFO

*Article history:*
Received 10 May 2017
Received in revised form 12 August 2017
Accepted 14 August 2017
Available online xxxx

*Keywords:*
Moving mesh methods
Adaptivity
Blow up
Interface dynamics
High order PDEs

ABSTRACT

We present an efficient moving mesh method for the simulation of fourth order nonlinear partial differential equations (PDEs) in two dimensions using the Parabolic Monge–Ampére (PMA) equation. PMA methods have been successfully applied to the simulation of second order problems, but not on systems with higher order equations which arise in many topical applications. Our main application is the resolution of fine scale behavior in PDEs describing elastic–electrostatic interactions. The PDE system considered has multiple parameter dependent singular solution modalities, including finite time singularities and sharp interface dynamics. We describe how to construct a dynamic mesh algorithm for such problems which incorporates known self similar or boundary layer scalings of the underlying equation to locate and dynamically resolve fine scale solution features in these singular regimes. We find a key step in using the PMA equation for mesh generation in fourth order problems is the adoption of a high order representation of the transformation from the computational to physical mesh. We demonstrate the efficacy of the new method on a variety of examples and establish several new results and conjectures on the nature of self-similar singularity formation in higher order PDEs.



## 1. Introduction

We present a parabolic Monge–Ampére (PMA) r-adaptive moving mesh method for numerical solutions of fourth order partial differential equations (PDEs) in two dimensions. There has been significant recent success on applying PMA methods to classic second order problems such as Burger's equation, semi-linear blow up equations [13] and numerical weather prediction [17]. PMA methods have not previously been considered for higher order systems, though such problems feature in many topical applications such as rock folding [26], ion bombardment lithography [59], thin film dynamics [56,45,68] and pattern formation [35,34,23]. The prototypical system considered here is the parabolic semi-linear equation

$$\frac{\partial u}{\partial t} = -\Delta^2 u + f(u),$$

for a convex nonlinearity $f(u)$, however, the methods developed here can be applied to a wide variety of fourth order problems.

* Corresponding author.
*E-mail address:* a.lindsay@nd.edu (A.E. Lindsay).

http://dx.doi.org/10.1016/j.jcp.2017.08.032
0021-9991/

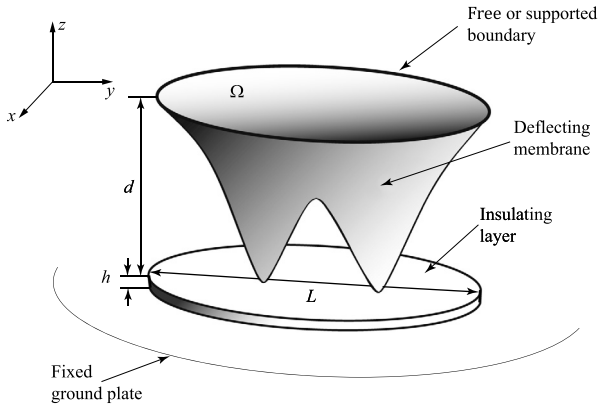



**Fig. 1.** Schematic diagram of a MEMS device a deflecting membrane in contact with an insulating layer of thickness $h$ and aspect ratio $d/L$. Figure reproduced from [63].

The principal application in the present work is to study the formation of singularities and the dynamics of sharp interfaces arising from elastic contact problems in micro-electromechanical systems (MEMS). MEMS are a family of miniaturized technologies which combine restorative elastic and attractive electrostatic forces to perform a variety of tasks on tiny scales [66]. Effective control and simulation of these dynamical processes is crucial for the design and operation of modern nanotechnology which has wide ranging applications such as microscopic drug delivery [2], resonators [73] and many more (cf. review [43]). Many components of MEMS consist of capacitor like structures (cf. Fig. 1) in which a deformable elastic plate is held fixed above a rigid and immobile ground plate. When a voltage $V$ is applied between the two surfaces, the upper surface will deflect towards the ground plate, and may come into physical contact if the voltage is large enough. This critical event is known as *touchdown* or *snap through* and its efficient numerical simulation is the topic of this work. Touchdown is a rapid event in which energy is focused at specific spatial locations over very short timescales – an adaptive strategy is essential for accurate and efficient numerical simulation of this phenomenon.

Under the assumption of a small aspect ratio ($d/L \ll 1$ in Fig. 1), the following parabolic nonlinear PDE system has recently been proposed [63] as a model for the dimensionless deflection $z = u(\mathbf{x}, t)$ of the deformable elastic surface at $\mathbf{x} = (x, y)$,

$$\frac{\partial u}{\partial t} = -(-\Delta)^p u - \frac{\lambda}{(1+u)^2} + \frac{\lambda \varepsilon^{m-2}}{(1+u)^m}, \quad (\mathbf{x}, t) \in \Omega \times (0, T) \tag{1a}$$

where the deflecting surface occupies $\Omega \subset \mathbb{R}^2$. The parameter $\lambda \propto V^2$ is a non-dimensional parameter quantifying the relative importance of electrostatic and elastic forces in the system. The term $\lambda \varepsilon^{m-2}(1+u)^{-m}$ for $m > 2$ represents surface effects which are important when the plates are in physical contact. For example, $m = 3$ represents Van der Waal forces while $m = 4$ accounts for the Casimir effect [44,6]. The dimensionless parameter $0 \leq \varepsilon \ll 1$ depends on a combination of geometrical and dielectric properties of the MEMS device [63]. The material properties of the elastic surface are reflected in the parameter $p$ signifying the order: for $p = 1$ the surface is modeled as a simple membrane with stretching energy only. The case $p = 2$ reflects a beam model of the surface which is ubiquitous in engineering models of MEMS and accurately reflects real devices. The boundary and initial conditions are specified as

$$p = 1: \begin{array}{ll} u = 0, & \mathbf{x} \in \partial\Omega; \\ u = u_0(\mathbf{x}), & \mathbf{x} \in \Omega. \end{array} \qquad p = 2: \begin{array}{ll} u = \Delta u = 0, & \mathbf{x} \in \partial\Omega; \\ u = u_0(\mathbf{x}), & \mathbf{x} \in \Omega. \end{array} \tag{1b}$$

Typical 1D solutions of (1) for $p = 2$ are shown in Fig. 2. The singular solutions of (1) are qualitatively different in the cases $\varepsilon = 0$, $\varepsilon > 0$ and we treat them separately.

*Case $\varepsilon = 0$:* In this scenario touchdown occurs when $\lambda$ is sufficiently large and corresponds to a finite time quenching singularity of (1) for which $\min_{\mathbf{x}\in\Omega} u(\mathbf{x}, t) \to -1$ as $t \to t_c^-$ [62]. The case $p = 1$ has been studied extensively including results on existence and stability of equilibria, bounds on singularity time and blow-up dynamics [66,31]. In the fourth order case $p = 2$, equation (1) develops multiple singularities with the location and multiplicity depending sensitively on the voltage parameter $\lambda$ and the shape and topology of $\Omega$. In 1D [62], a single singularity can form at the center of the interval or two can form around the center, if $\lambda$ is large enough (cf. Fig. 2a). In 2D, the multiplicity of singularities can be higher [61,36]. This paper establishes a numerical strategy that automatically detects the number and location(s) of singularities, and dynamically resolves these features while incorporating the scaling structure of the underlying PDE.

*Case $\varepsilon > 0$:* In this scenario the presence of the regularizing term $\lambda \varepsilon^{m-2}(1+u)^{-m}$ yields a well-posed system [63] which describes the evolution of (1) when the elastic surfaces are in contact. After initial contact, the solution forms sharp propagating layers which spread throughout the domain until they are eventually pinned at the boundary [60]. Equilibria of (1) for $\varepsilon > 0$ have been studied in [53,77,59].

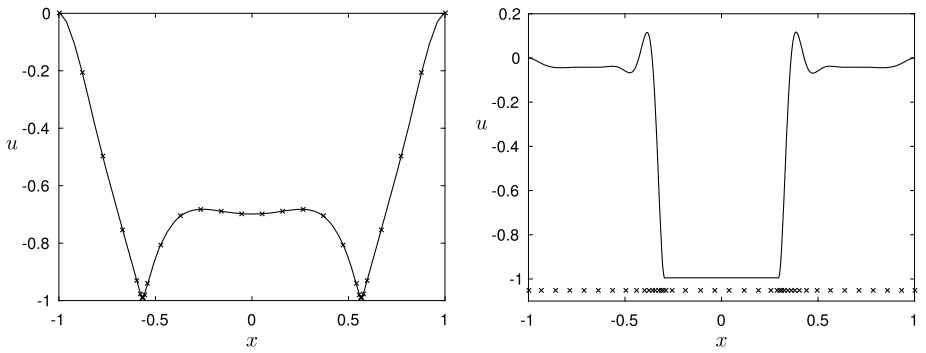



**Fig. 2.** Solutions of (1) for $p = 2$ calculated with the 1D r-adaptive method MOVCOL4 [68]. When $\varepsilon = 0$, touchdown is a finite time singularity which occurs for $\lambda$ sufficiently large at two distinct points. For $\varepsilon > 0$, a dynamic sharp interface develops after initial contact.

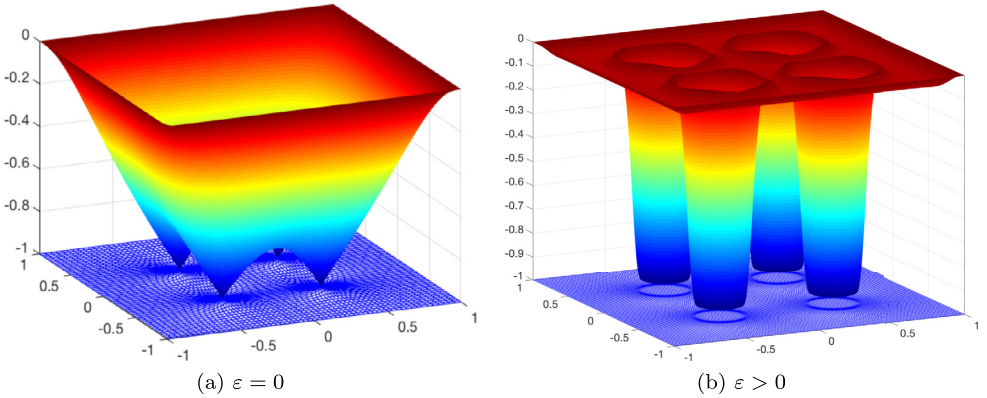



**Fig. 3.** Examples of the parabolic Monge–Ampére method developed in this paper applied to equation (1) in the singular ($\varepsilon = 0$) and regularized case ($\varepsilon > 0$).

The propagating interfaces (cf. Fig. 2b) have a *triple-deck* structure comprised of multiscale sublayers for $\varepsilon \ll 1$ [60]. The numerical method developed here resolves these propagating multiscale interfaces, and their growth, interaction and merging.

In the present work, we develop an efficient r-adaptive moving mesh strategy for the numerical solution of (1) in two dimensions for $p = 1, 2$ (cf. Fig. 3). In 1D, the r-adaptive method MOVCOL4 [68] has been successfully applied to this effect (cf. [62,60]). To generate a dynamic 2D mesh that dynamically adapts to the solution, we employ a Parabolic Monge–Ampére (PMA) approach (cf. [70,12]) which uses equidistribution to allocate mesh points in high interest regions. A central component in this approach is the determination of a transformation, $F$, between a fixed computational mesh $\Omega_C$ and a non-uniform physical mesh $\Omega_P$ (cf. Fig. 4). The underlying PDE is discretized on $\Omega_P$ while all computations are performed on the fixed mesh $\Omega_C$ via the mapping $F$. Computationally, this is significantly simpler than traditional h-adaptive approaches as the number of mesh points and its connectivity structure are fixed throughout.

The extension to fourth order PDEs presents a significant challenge largely because additional derivatives of the coordinate transformation $F$ are required to evaluate $\Delta^2 u$ on a non-uniform mesh (cf. subsection 2.3). This increases regularity requirements on $F$ which we address with a fourth order finite difference scheme. To our knowledge, this is the first application of PMA methods to MEMS problems and the first use of the PMA method for fourth order PDEs in several dimensions (cf. [68] for 1D). The methods of this paper can be applied to many topical problems featuring higher order derivatives [30, 67,64,51,42,40,27].

The rest of the paper is outlined as follows. In section 2 we summarize the theory of mesh generation using the Parabolic Monge–Ampére equation (cf. [15]). We introduce the PMA in two dimensions and discuss elements of the solution procedure, such as discretization of spatial operators and selection of the monitor function. In section 3, we present numerical results obtained from the method to solve (1) for $p = 2$ for the $\varepsilon = 0$ case and $p = 1, 2$ for the $\varepsilon > 0$ case. In each case, we validate the numerical solution with known exact and asymptotic solutions where possible. The method is able to resolve multiple sharp features and complex interfacial dynamics which would be extremely challenging and computationally expensive on a fixed mesh. In the case $p = 2$ and $\varepsilon = 0$, our results give very strong numerical evidence for self-similar blow up, and reveal a new and unexpected phenomenon in which blow up is accelerated by the fourth order term. Finally, in section 4 we conclude by summarizing the results and discussing avenues for future investigation.

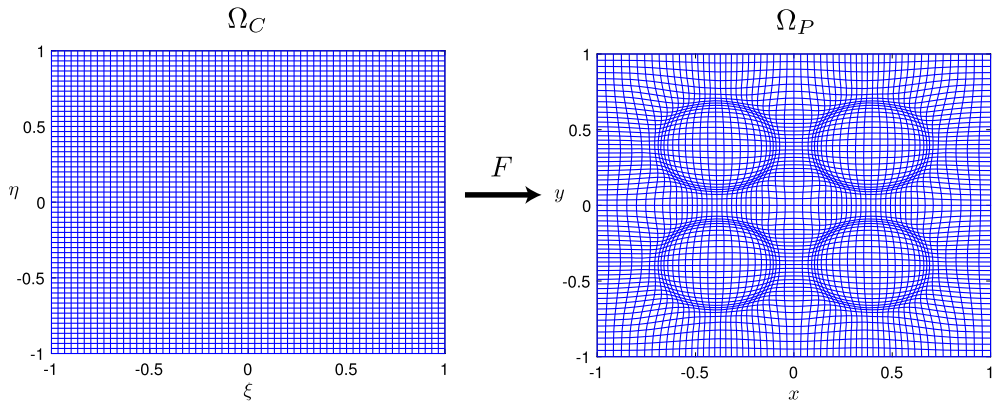



**Fig. 4.** An example of a transformation $F$ between the computational mesh $\Omega_C$ and the physical mesh $\Omega_P$. The PDE (1) is solved on the mesh $\Omega_P$ with all numerical computations performed on the uniform mesh $\Omega_C$.

## 2. Moving mesh adaptive methods and equidistribution principles

Adaptivity is essential in numerical resolution of PDEs with sharp or fine scale features. Two prevalent adaptive refinement strategies deploy additional resolution by adding mesh points (h-adaptivity) or increasing local approximation order (p-adaptivity). A third approach (r-adaptivity) uses a fixed number of mesh points coupled with a dynamic rearrangement strategy to transit mesh points so that local mesh density is increased in spatial regions where the solution is under resolved. The r-adaptive moving mesh approach generally requires fewer degrees of freedom and is simpler in its implementation details than other adaptive methods. Moreover, when certain scaling structures of the underlying PDE are already known, it is relatively convenient to incorporate these into the computational mesh.

Strategies for the dynamical generation of these meshes has been the subject of many studies [54,25,23,48,72,57]. Variational approaches [46,49] use a matrix valued monitor function that controls specific mesh quality conditions, such as alignment or orthogonality and aims to minimize a related functional. These methods generate a large computationally expensive nonlinear system to solve for the mesh.

A more recent approach, based on optimal transport theory (cf. [12,15,1,19]), utilizes a scalar monitor function to determine a coordinate transformation

$$F : \Omega_C \to \Omega_P,$$

between a fixed uniformly spaced computational mesh $\Omega_C$ with coordinates $\boldsymbol{\xi} = (\xi, \eta)$ and a non-uniform physical mesh $\Omega_P$ with coordinates $\mathbf{x} = (x, y)$ on which the PDE is solved (cf. Fig. 4). The mapping assumes the computational and physical domains are convex. In the present work, we consider transformations which leave the boundary unchanged so that $F : \partial\Omega_C \to \partial\Omega_P$. The goal of moving mesh methods is to determine a transformation $F$ which increases the local density of mesh points in spatial regions where the PDE has fine scale behavior requiring additional resolution.

A powerful tool for constructing such maps is the equidistribution principle derived by de Boor (cf. [28]). In this context, the equidistribution principle takes a monitor function $M(\mathbf{x}, t)$, which acts as a surrogate for the local error in the PDE solution, and distributes it evenly over the domain. This requires that the mesh is denser in regions where the solution $M(\mathbf{x}, t)$ is large and less dense elsewhere.

The equidistribution principle for any reference set $D \subset \Omega_c$ with corresponding image $F(D, t) \subset \Omega_P$ is constructed as

$$\frac{\int_D d\boldsymbol{\xi}}{\int_{\Omega_C} d\boldsymbol{\xi}} = \frac{\int_{F(D,t)} M(\mathbf{x}, t) d\mathbf{x}}{\int_{\Omega_P} M(\mathbf{x}, t) d\mathbf{x}}. \tag{2}$$

The reference set $D$ is arbitrary and therefore

$$M(\mathbf{x}, t)|J(\mathbf{x})| = \theta(t) := \frac{\int_{\Omega_P} M(\mathbf{x}, t) d\mathbf{x}}{\int_{\Omega_C} d\boldsymbol{\xi}}, \tag{3a}$$

where $J(\mathbf{x})$ and $|J(\mathbf{x})|$ are the Jacobian of $F$ and its determinant given by

$$J(\mathbf{x}) = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix}, \qquad |J(\mathbf{x})| = \det J(\mathbf{x}) = x_\xi\, y_\eta - x_\eta\, y_\xi . \tag{3b}$$

To avoid any mesh tangling, the map must be locally invertible which implies that $|J(\mathbf{x})| \neq 0$ for all $(\boldsymbol{\xi}, t)$.

Equation (3), supplemented with appropriate boundary conditions, is sufficient for uniquely defining a mesh in one dimension [68], but not higher dimensions. There are many options of varying complexity for constraining the transformation to obtain a unique mesh [46,49]. A simple recent method (cf. [11,16]) uses optimal transport theory to fix the transformation by seeking the $F$ which minimizes the distance

$$I = \int_{\Omega_C} |F(\boldsymbol{\xi}, t) - \boldsymbol{\xi}|^2 d\boldsymbol{\xi}. \tag{4}$$

This approach seeks a mesh which equidistributes the monitor function $M(\mathbf{x}, t)$ while staying as close to a uniform mesh as possible. These "optimally transported maps" have good regularity properties [20,71,19] and moreover allow the transformation to be represented as the gradient of a convex scalar function $P(\boldsymbol{\xi}, t)$, called the mesh potential. In this setting, the transformed mesh is given by

$$\mathbf{x} = \nabla_{\boldsymbol{\xi}} P = (P_\xi, P_\eta), \tag{5}$$

where, following from (3), $P(\boldsymbol{\xi}, t)$ satisfies the Monge–Ampére (MA) equation

$$M(\mathbf{x}, t) H(P) = \theta(t), \qquad H(P) = P_{\xi\xi} P_{\eta\eta} - P_{\xi\eta}^2, \tag{6a}$$

which describes an appropriate coordinate transformation. To fix the boundary of the physical domain, we consider a rectangular computational region $\Omega_C = [x_l, x_r] \times [y_b, y_t]$ and, recalling $\mathbf{x} = \nabla_{\boldsymbol{\xi}} P$, apply the Neumann boundary conditions

$$P_\xi = x_l, x_r \quad \text{for} \quad \xi = x_l, x_r; \qquad P_\eta = y_b, y_t \quad \text{for} \quad \eta = y_b, y_t, \tag{6b}$$

which yields the MA equation system (6). It has been shown (cf. [9, pp. 379–381], [21]), and in the context of mesh generation (cf. [29]), that for convex 2D regions, the MA system (6) admits a unique convex solution.

The monitor function $M(\mathbf{x}, t)$ must be chosen judiciously for the method to generate good meshes for solving the underlying PDE. The particular form of $M(\mathbf{x}, t)$ is application dependent and in problems like (1) where blow-up is a possibility, it must also be chosen so that the mesh inherits the strong scaling structures of the PDE.

### 2.1. Parabolic Monge–Ampére equation

The mesh equation (6) is a fully nonlinear equation that can be computationally expensive to solve, especially when coupled to a high order nonlinear PDE [29]. As the main goal is obtaining an accurate solution of the underlying PDE (1), it is not necessary to solve (6) to high precision. Instead Budd et al. [12,15] have suggested solving a time dependent relaxation of (6), where the gradient of the relaxed equations evolve toward the gradient of the solutions of (6) over a relatively short timescale similar to that of the underlying PDE. Assuming the relation $\mathbf{x} = \nabla_{\boldsymbol{\xi}} Q(\boldsymbol{\xi}, t)$ the equations take the form of the Parabolic Monge–Ampére Equation (PMA),

$$\alpha(I - \gamma \Delta_{\boldsymbol{\xi}}) Q_t = (|H(Q)| M(\nabla_{\boldsymbol{\xi}} Q))^{\frac{1}{2}}, \tag{7a}$$

with the boundary conditions

$$Q_\xi = x_l, x_r \quad \text{for} \quad \xi = x_l, x_r; \qquad Q_\eta = y_b, y_t \quad \text{for} \quad \eta = y_b, y_t. \tag{7b}$$

Assuming that the monitor function evolves on a similar time scale as the solution to the MA equation (i.e., $\varepsilon P_t$ is small) the mesh potential function $Q(\boldsymbol{\xi}, t)$ evolves towards $P(\boldsymbol{\xi}, t)$, the solution of (6) and provides a mesh with good regularity properties (see Lemmas 5.2 and 5.3 in [12]). This evolution toward the solution of the MA requires that for certain initial conditions, such as sharp fronts, the PMA must be solved independently to equidistribute the monitor function and create a good initial mesh that will evolve on a similar time scale as required. The right hand side of (7) is raised to the power $1/2$ so that it scales linearly with $Q$ to guarantee global solution existence. The operator $\mathcal{L} = \alpha(I - \gamma \Delta_{\boldsymbol{\xi}})$ on the left hand side of (7) decreases stiffness (cf. [25,74]). The parameter $\gamma > 0$ controls the extent of smoothing while $\alpha > 0$ controls the mesh adaption speed to help the mesh evolve on a similar timescale as the underlying PDE. In our simulations we choose $\alpha = 0.1$ and $\gamma = 0.1$. With a judicious choice of the monitor function $M(\mathbf{x}, t)$, the PMA allows for dynamic allocation of mesh density to resolve fine scale features of the PDE solution.

The PMA (7) is typically initialized from the uniform computational mesh

$$Q(\boldsymbol{\xi}, 0) = \frac{|\boldsymbol{\xi}|^2}{2} = \frac{1}{2}\left(\xi^2 + \eta^2\right). \tag{8}$$

When the initial data $u_0(\mathbf{x})$ of (1) has fine scale features, such as interfaces or layers, a refined initial mesh is necessary to diminish errors early in the simulation. This is achieved by solving the PMA (9a) for a fixed computational time, yielding a mesh adapted to fine scale structures of $u_0(\mathbf{x})$, before coupling it to the PDE.

### 2.2. Coupling the PDE and PMA

The moving mesh formulation requires solving a system of parabolic nonlinear equations for both the mesh (7) and the underlying PDE (1). There are two main ways to couple the equations (cf. [15]). First, the mesh equation (7) and the underlying PDE can be solved in an alternating procedure, using interpolation to cast the solution of the PDE onto the updated mesh at a discrete number of timesteps (cf. [16]). Second, the PMA and PDE can be solved simultaneously by including a Lagrangian term in the PDE, which is determined from the solution $Q$ of the PMA.

In our implementation, we have found the second approach works well. The coupling method reformulates the underlying PDE with an additional Lagrangian term $\nabla_{\mathbf{x}} u \cdot \mathbf{x}_t = \nabla_{\mathbf{x}} u \cdot \nabla_{\boldsymbol{\xi}} Q_t$ which creates a single coupled system. Here $\nabla_{\mathbf{x}}$ is the gradient operator in physical variables. This prevents lag in mesh movement and avoids an expensive interpolation step, however, the additional Lagrangian term must be carefully discretized to avoid instabilities, especially as singularities are approached (cf. [55,56]). With the Lagrangian coupling term included, the coupled system is

$$\alpha(I - \gamma \Delta_{\boldsymbol{\xi}}) Q_t = (|H(Q)| M(\mathbf{x}, t))^{\frac{1}{2}}, \tag{9a}$$

$$u_t - \nabla_{\mathbf{x}} u \cdot \nabla_{\boldsymbol{\xi}} Q_t = -(-\Delta_{\mathbf{x}})^p u - \frac{\lambda}{(1+u)^2} + \frac{\lambda \varepsilon^{m-2}}{(1+u)^m}. \tag{9b}$$

In the next section we describe the finite difference discretization of (9) on the computational domain $\Omega_C = [-1, 1]^2$ with a fixed square mesh of $N^2$ points given by

$$(\xi_i, \eta_j) = (-1 + (i-1)\,\Delta\xi, -1 + (j-1)\,\Delta\eta), \qquad \Delta\xi = \Delta\eta = \frac{2}{N-1}, \tag{10}$$

for $i = 1, \ldots, N$ and $j = 1, \ldots, N$.

### 2.3. Discretizing the equations

The fully coupled system with the additional Lagrangian term (9) is discretized by finite differences. The first equation (9a) is solved on the uniform computational mesh $\Omega_C$ with mesh derivatives $Q_{\xi\xi}$, $Q_{\eta\eta}$ and $Q_{\xi\eta}$ approximated by fourth order finite differences which invoke the Neumann boundary conditions (7b) on $Q$. From these terms, the Hessian $H(Q) = Q_{\xi\xi} Q_{\eta\eta} - Q_{\xi\eta}^2$ is calculated directly. As discussed below, high order approximations of the mesh derivatives are required for accurate discretization of $\Delta_{\mathbf{x}}^2 u$ near the boundary. So that the boundary $\partial\Omega_P$ remains fixed, a homogeneous Neumann boundary condition is invoked on $Q_t$ in the discretization of $\Delta_{\boldsymbol{\xi}}$. For completeness, we give the full finite difference stencils in Appendix A.

Once the right hand side of (9a) is formed, inversion yields that

$$Q_t = \mathcal{L}^{-1} \mathcal{F}, \qquad \mathcal{F} = (|H(Q)| M(\mathbf{x}, t))^{\frac{1}{2}}. \tag{11}$$

The inversion of $\mathcal{L} = \alpha(I - \gamma \Delta_{\boldsymbol{\xi}})$ is efficiently performed by discrete cosine transform.

The spatial derivatives in physical space are determined in computational space via the coordinate transformation

$$\begin{pmatrix} u_\xi \\ u_\eta \end{pmatrix} = \begin{pmatrix} x_\xi & y_\xi \\ x_\eta & y_\eta \end{pmatrix} \begin{pmatrix} u_x \\ u_y \end{pmatrix}. \tag{12}$$

Inverting (12) and using $(x, y) = \nabla_{\boldsymbol{\xi}} Q = (Q_\xi, Q_\eta)$ gives that

$$u_x = \bar{J}\big(Q_{\eta\eta} u_\xi - Q_{\xi\eta} u_\eta\big), \tag{13a}$$

$$u_y = \bar{J}\big(-Q_{\eta\xi} u_\xi + Q_{\xi\xi} u_\eta\big), \tag{13b}$$

where $\bar{J} = |H(Q)|^{-1}$ is the reciprocal of the Jacobian. The second derivatives are

$$u_{xx} = \bar{J}\, Q_{\eta\eta}\Big(\bar{J}(Q_{\eta\eta} u_\xi - Q_{\xi\eta} u_\eta)\Big)_\xi - \bar{J}\, Q_{\xi\eta}\Big(\bar{J}(Q_{\eta\eta} u_\xi - Q_{\xi\eta} u_\eta)\Big)_\eta, \tag{14a}$$

$$u_{yy} = \bar{J}\, Q_{\xi\eta}\Big(\bar{J}(Q_{\xi\eta} u_\xi - Q_{\xi\xi} u_\eta)\Big)_\xi + \bar{J}\, Q_{\xi\xi}\Big(\bar{J}(-Q_{\eta\xi} u_\xi + Q_{\xi\xi} u_\eta)\Big)_\eta. \tag{14b}$$

The Laplacian is formed from these expressions and concisely rearranged as

$$\Delta_{\mathbf{x}} u = u_{xx} + u_{yy} = \bar{J}\, \nabla_{\boldsymbol{\xi}} \cdot [\bar{J}\, \mathcal{A}\, \nabla_{\boldsymbol{\xi}} u], \tag{15a}$$

where $\mathcal{A}$ is the symmetric $2 \times 2$ matrix whose entries are

$$\mathcal{A}_{11} = Q_{\eta\xi}^2 + Q_{\eta\eta}^2, \quad \mathcal{A}_{21} = \mathcal{A}_{12} = -Q_{\eta\xi}(Q_{\xi\xi} + Q_{\eta\eta}), \quad \mathcal{A}_{22} = Q_{\eta\xi}^2 + Q_{\xi\xi}^2. \tag{15b}$$

Equation (15a) indicates that the non-uniform mesh results in an anisotropic diffusion problem which can be discretized by a finite difference approximation (see Appendix B). The discretization of the bi-Laplacian

$$\Delta_{\mathbf{x}}^2 u := \Delta_{\mathbf{x}}(\Delta_{\mathbf{x}} u) = u_{xxxx} + 2u_{xxyy} + u_{yyyy}, \tag{16}$$

can be achieved one of two ways. The first option is to compute expressions for the fourth order derivatives (16) directly from further differentiation of (14). This results in a large number of individual fourth order derivative terms to approximate. The second approach is to apply (15a) twice – first to obtain $v = \Delta_{\mathbf{x}} u$, and once more to find $\Delta_{\mathbf{x}} v = \Delta_{\mathbf{x}}^2 u$. The second option is appealing since the discretization of (15a) is straightforward and applying it twice to obtain $\Delta_{\mathbf{x}}^2 u$ adds minimal complexity.

A complication of this approach, arising from the implementation of boundary conditions, is that the finite difference approximations of $Q_{\xi\xi}$, $Q_{\eta\eta}$, $Q_{\xi\eta}$, and consequently $\Delta_{\mathbf{x}} u$, exhibit discontinuous truncation errors at mesh points adjacent to $\partial\Omega$. As $v = \Delta_{\mathbf{x}} u$ has discontinuities, applying the discrete Laplacian a second time to obtain $\Delta_{\mathbf{x}}^2 u$ will result in a loss of two orders of numerical accuracy at those near boundary points. Therefore, the frequently used second order discretization of the Laplacian gives rise to a non-convergent approximation of the bi-Laplacian.

To maintain convergence near $\partial\Omega$, we employ a fourth order finite difference method to discretize the operator (14) and obtain $v = \Delta_{\mathbf{x}} u$. The mesh derivatives terms $Q_{\eta\eta}$, $Q_{\xi\xi}$ and $Q_{\eta\xi}$ are also approximated by fourth order finite differences. The Laplacian $\Delta_{\mathbf{x}} v$ is then obtained using a second order finite difference approximation of (14), which completes the calculation of the bi-Laplacian of $u$. We give the appropriate fourth order stencils in Appendix B. We remark that it is not sufficient to apply a fourth order approximation at certain near boundary points and a second order method elsewhere since this will only propagate the discontinuity.

The discontinuities in the truncation error result in a numerical method which converges with a reduced order of accuracy at near boundary points. In summary, we expect $\Delta_{\mathbf{x}} u$ to converge with order four at internal points and order three at near boundary points. The bi-Laplacian $\Delta_{\mathbf{x}}^2 u$ converges with order two at internal points and order one at near boundary points. This reduction in boundary accuracy can not be bypassed by direct evaluation of $\Delta_{\mathbf{x}}^2 u$ which requires evaluation of the third derivatives of $Q_{\eta\eta}$, $Q_{\xi\xi}$ and $Q_{\eta\xi}$ and results in an equivalent reduction of order.

The final term considered is the Lagrangian term $\nabla_{\mathbf{x}} u \cdot \nabla_{\boldsymbol{\xi}} Q_t$ where $Q_t$ is calculated in (11). Approximation of this coupling term by centered differences is well known to generate instabilities (cf. [24,55,56]) which are especially pronounced in the vicinity of a forming singularity. Following [55], we implement a second order upwinding scheme extended to two dimensions which results in a stable solution. The appropriate stencils are given in Appendix C.

### 2.4. The monitor function

An important component of a successful moving mesh method is the choice of monitor function so that mesh density is increased at regions of high interest in the PDE solution, such as singularities or propagating fronts. A common monitor function is the arc-length function,

$$M(\mathbf{x}, t) = \sqrt{1 + |\nabla_{\mathbf{x}} u|^2}, \tag{17}$$

which clusters points in locations of high changes in the gradient of the solution. This monitor function has been successfully used to resolve fronts in Burger's equation [15] and problems in numerical weather prediction [16]. In examples with blow-up [11,15], the monitor function can be chosen such that the mesh inherits the correct dynamic length scale of the underlying PDE, allowing singularities to be accurately resolved and tracked in time. To see this, consider the PMA

$$\alpha(I - \gamma\Delta_{\boldsymbol{\xi}})Q_t = (|H(Q)|M\left(\nabla_{\boldsymbol{\xi}} Q\right))^{\frac{1}{2}},$$

and a small length scale $L(t)$. Near a singularity, we can assume that

$$Q \sim L(t)S(\boldsymbol{\xi}, t).$$

The PMA will be independent of $L(t)$ near the singularity so long as

$$M \sim \left(\frac{1}{L}\frac{dL}{dt}\right)^2.$$

For power law blow up profiles, $L(t) = (t_c - t)^{\alpha}$, this yields that $M$ should scale like

$$M \sim \frac{1}{(t_c - t)^2}, \tag{18}$$

so that the PMA is length scale independent near the singularity. A rigorous account of this scaling analysis is given in [11]. For the MEMS problem (1) in the case $\varepsilon = 0$, the underlying PDE reduces to

$$u_t = -(-\Delta)^p u - \frac{\lambda}{(1+u)^2}, \qquad (\mathbf{x}, t) \in \Omega \times (0, t_c). \tag{19}$$

To obtain the correct scaling laws for the forming singularity, we observe that the transformations

$$t \to at, \qquad \mathbf{x} \to a^{\frac{1}{2p}}\mathbf{x}, \qquad (1+u) \to a^{\frac{1}{3}}(1+u), \tag{20}$$

leave (19) unchanged. Therefore, if the distance to singularity $(1+u)$ halves, the time to singularity $(t_c - t)$ decreases by a factor of 8. Comparing with (18), an $M(\mathbf{x}, t)$ which gives a mesh that is independent of the length-scale near the singularity is

$$M(\mathbf{x}, t) = \frac{1}{(1+u(\mathbf{x}, t))^6}, \qquad \text{Case } \varepsilon = 0. \tag{21}$$

For the regularized MEMS model (1) with $\varepsilon > 0$, choosing a proper monitor function is more complicated. The analysis discussed in [60] shows the presence of sharp interfaces with sublayer structure that must be detected, resolved, and tracked. Because the solution resolves curved interface problems, it was natural to consider arc-length and Laplacian monitor functions for tracking the interfaces. After testing both arc-length and Laplacian monitor functions in the $p = 1, 2$ cases, we found that the arc-length function worked well for the second order case, but caused instabilities in the fourth order case. This lead to the adoption of a Laplacian based monitor function in the $p = 2$ case. After accounting for the square root dependence on the right hand side of (9a), the monitor functions chosen are

$$M(\mathbf{x}, t) = 1 + u_x^2 + u_y^2, \quad \text{Case } p = 1; \tag{22a}$$

$$M(\mathbf{x}, t) = |u_{xx} + u_{yy}|^2, \quad \text{Case } p = 2. \tag{22b}$$

Before discretizing and solving (9), some additional steps are taken to assure good quality of the mesh. When singularities emerge in (1), equidistribution can cause many mesh points to rush toward the maxima of the monitor function, potentially leaving the solution under resolved in the rest of the domain. This issue can be mitigated by integral averaging, or Mackenzie regularization [7], to give

$$M(\mathbf{x}, t) \to M(\mathbf{x}, t) + C \int_{\Omega_p} M(\mathbf{x}', t)\, d\mathbf{x}', \tag{23}$$

where $C$ tunes the proportion of mesh points which concentrate on high interest areas as determined by $M(\mathbf{x}, t)$. For $C = 1$, half the points are concentrated around the maxima of $M(\mathbf{x}, t)$ and half over the rest of the domain. In the case $\varepsilon > 0$, smaller values of $C$ are chosen because the solution is largely uniform in regions behind the sharp interfaces, and so less resolution is required there.

It is also important to smooth the monitor function, since rapidly varying components in the monitor function could be hard to resolve and thus propagate errors (cf. [12,16,13]). We apply the fourth-order filter

$$\begin{aligned} M_{i,j} \to M_{i,j} &+ \frac{2}{16}(M_{i+1,j} + M_{i-1,j} + M_{i,j-1} + M_{i,j+1}) \\ &+ \frac{1}{16}(M_{i+1,j+1} + M_{i-1,j-1} + M_{i+1,j-1} + M_{i-1,j+1}), \end{aligned} \tag{24}$$

on $M(\mathbf{x}, t)$ four times during each iteration. This smoothing operation distributes variations over nearby grid points, creating a more regular mesh.

### 2.5. Adaptive time stepping

In the singularity forming case $\varepsilon = 0$, it is essential to dynamically reduce the timestep as the singularity is approached. To approach the finite time singularities, which form in (19) for $\lambda$ sufficiently large [62], a computational time step $\tau$ based on the Sundman transformation (cf. [47,18]),

$$\frac{dt}{d\tau} = g(u), \tag{25}$$

is applied. The function $g(u)$ measures the closeness of the solution to singularity so that for a fixed $d\tau$, the time step $dt$ gradually decreases as a singularity is approached. For the problem (19), singularities correspond to $u \to -1$ in finite time so that $g(u) \to 0$ as $u \to -1$. The scalings (20), motivate the particular choice

$$g(u) = \min_{\mathbf{x}\in\Omega}(1+u)^3 \tag{26}$$

for the computational time step. This can be easily integrated into our system of equations for the mesh and the PDE, creating a system of $2N^2 + 1$ equations to be solved (27) (where $N$ is the number of mesh points in each spatial direction). In the case $\varepsilon > 0$, there are no finite time singularities so a computational time step is not required ($g = 1$).

### 2.6. Fully spatially discrete system

After all terms have been discretized, we obtain the coupled system of ODEs,

$$\frac{d}{d\tau}\begin{pmatrix} Q \\ u \\ t \end{pmatrix} = g(u)\begin{pmatrix} \mathcal{L}^{-1}\mathcal{F} \\ -(-\Delta_{\mathbf{x}})^p u - \dfrac{\lambda}{(1+u)^2} + \dfrac{\lambda\varepsilon^{m-2}}{(1+u)^m} + \nabla_{\mathbf{x}} u \cdot \mathbf{x}_t \\ 1 \end{pmatrix}, \tag{27}$$

which is solved using the MATLAB routine ode23 with relative and absolute tolerances set at $1 \times 10^{-6}$. Here $\mathcal{F}$ is the result of evaluating the right hand side of the PMA (9a) and the operator of $\mathcal{L} = \alpha(1 - \gamma\Delta_{\boldsymbol{\xi}})$ is inverted by discrete cosine transform. The moving mesh parameters are set at $\alpha = \gamma = 0.1$. The function $g(u)$ controls the adaptive computational time step used to approach the singularity. For the singularity forming case $\varepsilon = 0$, we use $g(u) = \min_{\mathbf{x}\in\Omega}(1+u)^3$ and for $\varepsilon > 0$, $g(u) = 1$.

## 3. Numerical results

### 3.1. Test of discretization on the non-uniform mesh

In this section, we use a test example to verify the accuracy of the discretizations of the Laplacian and bi-Laplacian operators on a non-uniform mesh. The test mesh is given by the potential $Q$ as

$$Q(\xi,\eta) = \frac{\xi^2}{2} + r\exp\left[r(\xi^2-1)^2(\eta^2-1)^2\right] + \frac{\eta^2}{2}, \qquad (\xi,\eta) \in (-1,1)^2. \tag{28}$$

Here $r \geq 0$ is a parameter for tuning the non-uniformity of the mesh such that more mesh points are concentrated near the origin for $r > 0$. A uniform mesh corresponds to $r = 0$. The discretization is verified against constructed solutions of the test problem

$$\Delta^2 u = f(\mathbf{x}), \quad \mathbf{x} \in \Omega; \tag{29a}$$

$$u = g(\mathbf{x}), \quad \mathbf{x} \in \partial\Omega; \tag{29b}$$

$$\Delta u = h(\mathbf{x}), \quad \mathbf{x} \in \partial\Omega, \tag{29c}$$

for which we choose a $u(\mathbf{x})$ and calculate the corresponding functions $f, g, h$ exactly. In this example, we use function (30) corresponding to a spike at the origin (cf. Fig. 5a) and verify that the spatial operator $\Delta^p u$ for $p = 1, 2$ are being discretized correctly.

$$u(x,y) = \sqrt{20}\cos\left(\frac{\pi x}{2}\right)\cos\left(\frac{\pi y}{2}\right)e^{-20(x^2+y^2)}. \tag{30}$$

The discrete approximations $\Delta u_h$ and $\Delta^2 u_h$ are formulated by fourth and second order finite differences respectively, as discussed in Sec. 2.3. The absolute error is computed as

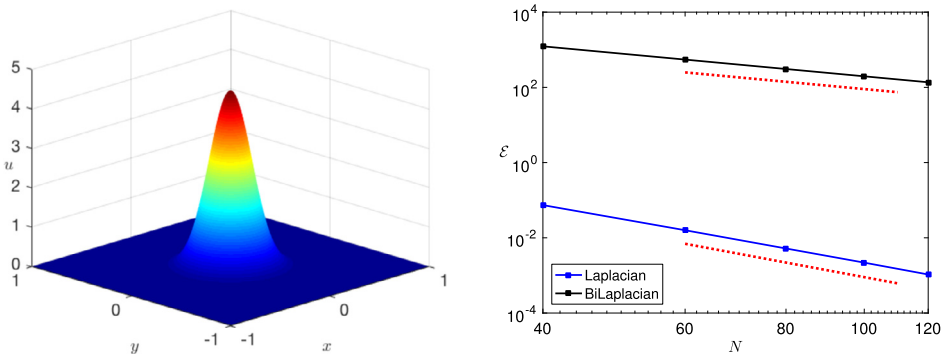
$$\mathcal{E} = \|\Delta^p u - \Delta^p u_h\|_{L^\infty(\Omega)}. \tag{31}$$

In Fig. 5b we see that the numerical error decreasing at the expected rate as the number of mesh points increases. In Table 1, the absolute errors in the numerical approximation are displayed for the non-uniform ((28) with $r = 0.24$) and uniform mesh. The test solution is localized at the origin and the error is roughly halved on the non-uniform mesh for an equivalent number of grid points on a uniform mesh.

We remark that this example shows the method correctly discretizes the spatial operators for a smooth solution profile on a fixed mesh and that the approximation error can be reduced by increasing grid density in regions where the solution is localized. In the nonlinear time dependent problems of the following section, the r-adaptive method reaps additional advantages when resolving forming singularities and the dynamics of sharp layers. While reduced errors are expected on the adaptive mesh, a crucial additional advantage is that the mesh inherits the correct dynamic length-scale of the underling solution, and so accurately captures the self-similar structure of the underlying equation.

### 3.2. Adaptivity to touchdown

For the finite time singularity case of equation (1) with $\varepsilon = 0$, we discretize the computational domain $\Omega = (-1,1)^2$ using $N$ grid points and integrate until $\min_{\mathbf{x}\in\Omega}(1+u) = 1 \times 10^{-3}$. In the particular case $\varepsilon = 0$, we work with a rescaled form for the equation (19) by taking a new time variable $t \to \lambda t$ and defining $\lambda = \beta^{-2}$. This rescaling give the equation




(a) Test solution (30). (b) Absolute global error against $N$.

**Fig. 5.** The test case on the non-uniform mesh (28) with $r = 0.24$. Panel (a): the solution corresponding to a spike at the origin (30). Panel (b): The absolute error $\mathcal{E}$ against mesh points $N$ on a logarithmic scale. Straight lines of slope $-4$ and $-2$ (red dotted) are present for comparison.

**Table 1**
The absolute global error $\mathcal{E} = \|\Delta^p u - \Delta^p u_h\|_{L^\infty(\Omega)}$ for the test case (29) on a non-uniform grid (28) with $r = 0.24$ and a uniform grid ($r = 0$).

| Grid size | Laplacian abs. error ($p = 1$) | | Bi-Laplacian abs. error ($p = 2$) | |
|---|---|---|---|---|
| Mesh | Uniform | Non-uniform | Uniform | Non-uniform |
| $40 \times 40$ | $7.50 \times 10^{-2}$ | $7.41 \times 10^{-2}$ | $2.48 \times 10^{3}$ | $1.23 \times 10^{3}$ |
| $60 \times 60$ | $2.53 \times 10^{-2}$ | $1.59 \times 10^{-2}$ | $1.12 \times 10^{3}$ | $5.47 \times 10^{2}$ |
| $80 \times 80$ | $9.31 \times 10^{-3}$ | $5.20 \times 10^{-3}$ | $6.32 \times 10^{2}$ | $3.08 \times 10^{2}$ |
| $100 \times 100$ | $4.07 \times 10^{-3}$ | $2.19 \times 10^{-3}$ | $4.06 \times 10^{2}$ | $1.97 \times 10^{2}$ |
| $120 \times 120$ | $2.03 \times 10^{-3}$ | $1.06 \times 10^{-3}$ | $2.82 \times 10^{2}$ | $1.37 \times 10^{2}$ |

$$\frac{\partial u}{\partial t} = -\beta^2(-\Delta)^p u - \frac{1}{(1+u)^2}, \quad (\mathbf{x}, t) \in \Omega \times (0, t_c(\beta)), \tag{32a}$$

$$u(\mathbf{x}, 0) = 0, \quad \mathbf{x} \in \Omega; \qquad u = \Delta u = 0, \quad \mathbf{x} \in \partial\Omega. \tag{32b}$$

We solve the PMA system (27) with the monitor function and computational timestep

$$M(\mathbf{x}, t) = \frac{1}{(1+u(\mathbf{x}, t))^6} + \int_\Omega \frac{1}{(1+u(\mathbf{x}', t))^6}\, d\mathbf{x}', \tag{33a}$$

$$g(u) = \min_{\mathbf{x} \in \Omega_P} (1 + u(\mathbf{x}, t))^3, \tag{33b}$$

with the `MATLAB` routine `ode23`.

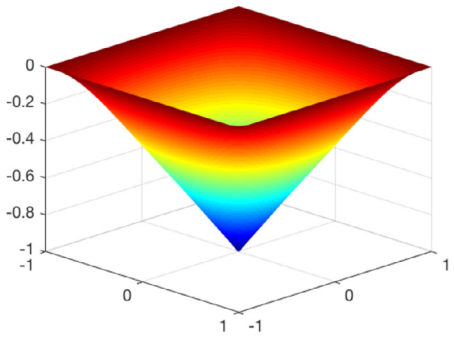
*3.2.1. Numerical simulations*

In the results shown in Fig. 6, we observe that the PDE solution for $p = 2$ is well resolved by the PMA method in this singular regime. In particular Figs. 6e–6f show that the mesh exhibits no signs of tangling or crossing nodes and stays regular even while very close to the singularity point. As the parameter $\beta$ decreases through $\beta \approx 0.09$ in the case $p = 2$, the multiplicity of singularities increases from one to four. The case $p = 1$ does not exhibit this multiplicity in blow-up location and only one singularity will form at the origin [58]. In Fig. 7, we display the dependence of the number, location(s) and time of singularities on the parameter $\beta$. The method is able to automatically detect and resolve the multiple singularities that may develop. In examining the dependence of the singularity time $t_c(\beta)$ as $\beta \to 0$, we observe a new phenomenon in that the bi-Laplacian term may accelerate the formation of the singularity. In the case of no elastic resistance force $\beta = 0$, equation (32) reduces to the ODE $\bar{u}_t = -(1+\bar{u})^{-2}$ with initial condition $\bar{u}(0) = 0$. This is solved to give

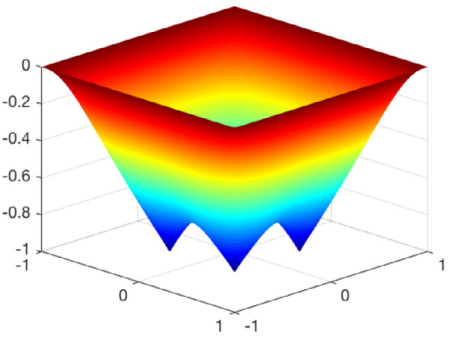
$$\bar{u} = -1 + (1 - 3t)^{\frac{1}{3}}, \quad t \in (0, t_c(0)), \qquad t_c(0) = \frac{1}{3}. \tag{34}$$

In the case $p = 1$, we see in Fig. 7b that $t_c(\beta) > t_c(0)$ with $\lim_{\beta \to 0} t_c(\beta) = t_c(0)$. This result follows from the maximum principle which gives that $\bar{u}(t) < u(\mathbf{x}, t)$ since $\bar{u}(t)$ is a subsolution of $u(\mathbf{x}, t)$ in the case $p = 1$. In the fourth order case $p = 2$, there is no maximum principle and we observe that $t_c(\beta) < t_c(0)$ for $\beta$ small enough ($\beta \lesssim 0.28$). Moreover, we conjecture from these numerical results that

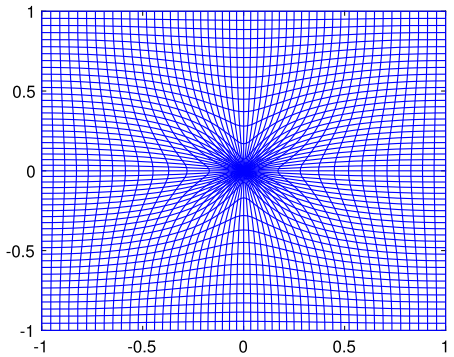
$$\lim_{\beta \to 0} t_c(\beta) \neq t_c(0), \qquad (\text{Case } p = 2). \tag{35}$$



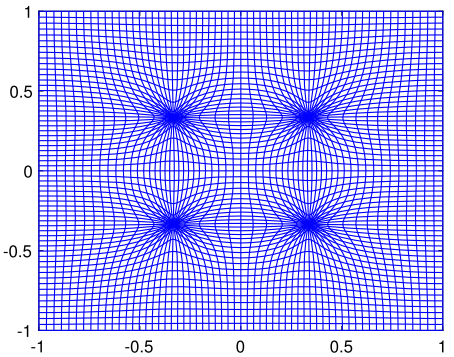

(a) Solution and mesh of (32) and $\beta = 0.15$



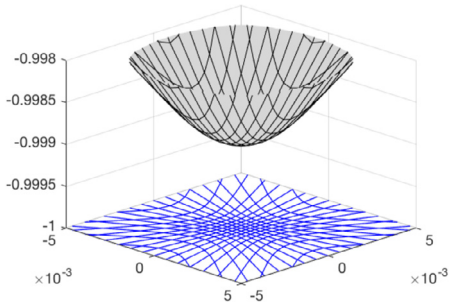

(b) Solution and mesh of (32) and $\beta = 0.07$



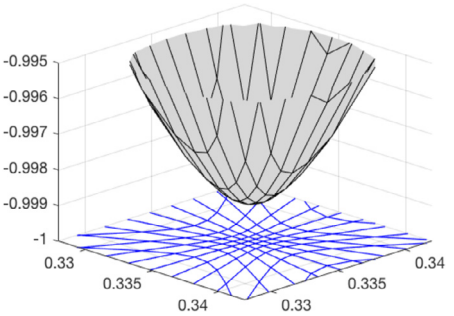

(c) Final mesh for $\beta = 0.15$




(d) Final mesh for $\beta = 0.07$




(e) Solution near touchdown - $\beta = 0.15$.




(f) Solution near touchdown - $\beta = 0.07$.

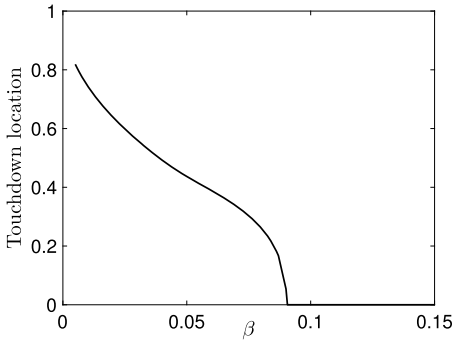
**Fig. 6.** PDE and mesh for the solution of (32) for $\beta = 0.15$ and $\beta = 0.07$ for $\min_{\mathbf{x}\in\Omega}(1+u) = 1\times 10^{-3}$. Figs. 6e–6f show the profile of the solution and the mesh in the vicinity of the touchdown point. The computations are performed with $N = 51$ (left panels) and $N = 61$ (right panels) grid points respectively.

This result is counter intuitive because the term $\beta^2\Delta^2 u$ in (32) reflects an elastic restorative force that resists the motion of the beam and so impedes the formation of singularities. However, as $\beta$ decreases below a certain threshold ($\beta \approx 0.28$), this term acts as an accelerant to quicken the onset of touchdown. A rigorous theory for this numerical observation and the determination of the correct limiting problem for $\lim_{\beta\to 0} t_c(\beta)$ are two open problems for future investigation.

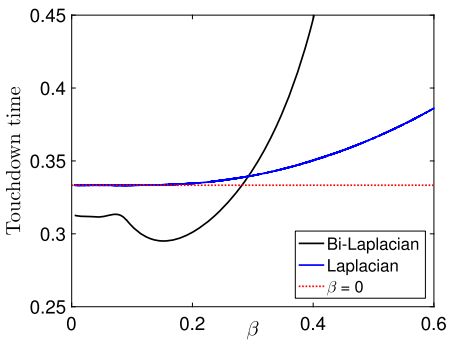
### *3.2.2. Self-similar behavior of the limiting solution*

In this section we consider the limiting form of quenching solutions to (1) for $\varepsilon = 0$ and with $\min_{\mathbf{x}\in\Omega} u(\mathbf{x}, t) \to -1$ as $t \to t_c^-$. In particular, we construct self similar profiles and integrate the PDE (1) close to singularity in order to verify convergence. The problem of determining existence and stability of self-similar blow-up profiles is a long studied problem in nonlinear PDEs arising in many diverse applications [75,35,34].

For the MEMS problem (32) in the second order case ($p = 1$), the blow up behavior was found to be approximately self-similar (cf. [45]) in a way analogous to classical semi-linear problems with power nonlinearities [35]. In the fourth order case ($p = 2$), a numerical-analytical approach (cf. [62]) in the one dimensional setting strongly suggested a self-similar singularity structure in (32). The prevalence of self similar blow up in higher order equations has also been observed in other contexts [38,37,17,36].




(a) Touchdown points against $\beta$ in the first quadrant in the bi-Laplacian ($p = 2$) case.




(b) Touchdown time against $\beta$ for the Laplacian ($p = 1$) and bi-Laplacian ($p = 2$) cases.

**Fig. 7.** The location $\mathbf{x}_c$ (left) and time $t_c(\beta)$ (right) of singularities for equation (32) on the square domain $\Omega = [-1, 1]^2$. The singularity location plotted corresponds to the first quadrant. The singularity time $t_c(\beta)$ has distinct limiting behavior as $\beta \to 0$ in the $p = 1$ and $p = 2$ cases.

Following the methodology of [75,8], we seek a two dimensional self-similar quenching profile by using the scale invariance (20) to motivate the new variables

$$u(\mathbf{x}, t) = -1 + (t_c - t)^{\frac{1}{3}} w(\mathbf{y}), \qquad \mathbf{y} = \frac{\mathbf{x} - \mathbf{x}_c}{\beta^{\frac{1}{2}} (t_c - t)^{\frac{1}{4}}}, \tag{36}$$

for $p = 2$. Substituting (36) into (32) yields the PDE for the self similar profile

$$-\Delta_{\mathbf{y}}^2 w - \frac{1}{4} \mathbf{y} \cdot \nabla_{\mathbf{y}} w + \frac{w}{2} - \frac{1}{w^2} = 0, \qquad \mathbf{y} \in \mathbb{R}^2. \tag{37}$$

Equation (37) is a nonlinear, non-variational and non-constant coefficient problem and so challenging to analyze exactly. As a simplification, we assume radial symmetry in terms of the variable $\rho = |\mathbf{y}|$ and a solution of form $v(\rho) = w(|\mathbf{y}|)$. The reduced problem satisfies the fourth-order nonlinear ordinary differential equation,

$$v'''' + \frac{2}{\rho} v''' - \frac{1}{\rho^2} v'' + \frac{1}{\rho^3} v' + \frac{1}{4} \rho\, v' - \frac{v}{3} + \frac{1}{v^2} = 0, \quad \rho > 0 \tag{38a}$$

together with the symmetry condition at the origin

$$v'(0) = v'''(0) = 0. \tag{38b}$$

The far field behavior of (38) can be found from a WKB analysis to reveal that

$$v(\rho) \sim [\alpha \rho^{4/3} + \mathcal{O}(\rho^{4/3})] + C \rho^{-16/9} \exp[-3\rho^{4/3} 2^{-8/3}] + \ldots, \quad \rho \to \infty \tag{38c}$$

where $C$ is a constant. This gives rise to the effective Robin condition

$$\frac{\rho}{4} v' - \frac{1}{3} v = 0, \qquad \rho \to \infty. \tag{38d}$$

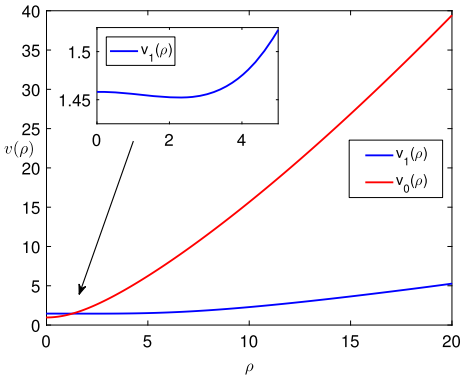
We seek a numerical solution to (38) on the interval $[0, L]$ using a finite difference approximation with the boundary conditions (38b) at $\rho = 0$ and (38d) on $\rho = L$. A damped Newton's method [3] is applied to the system obtained for $L = 20$ and iterations are initialized from the solutions of the reduced problem

$$\frac{\rho}{4} \bar{v}' - \frac{\bar{v}}{3} + \frac{1}{\bar{v}^2} = 0, \quad \Longrightarrow \quad \bar{v} = (\alpha^3 \rho^4 + 3)^{1/3}. \tag{39}$$

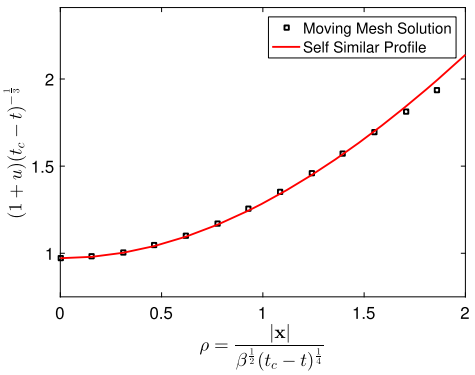
The parameter $\alpha$ plays the role of a nonlinear eigenvalue and equation (38) is expected to have solutions only at discrete values of $\alpha$.

After initializing iterations over a wide range of $\alpha$ values, convergence is observed to only two distinct profiles corresponding to $\alpha_0 = 0.7625, \alpha_1 = 0.0966$. By calculating the eigenvalues of the linearization about these profiles, we find the profile $v_0(\rho)$ corresponding to $\alpha_0$ is stable while the second profile $v_1(\rho)$, corresponding to $\alpha_1$, is unstable – see Fig. 8a. This instability of $v_1(\rho)$ is also suggested by the additional critical point of the profile. It remains an open question whether or not there exists an infinite number of these profiles for discrete values of $\alpha$, as suggested by recent exponential asymptotic approaches [26,51]. If such an infinite family does exist, those additional profiles are expected to be unstable.

We now investigate the convergence of the numerical solution to the full PDE (32) for $p = 2$ with the stable self similar profile $v_0(\rho)$, through the transformation (36). We select the value $\beta = 0.15$ which corresponds to a single singularity




(a) Two self similar profiles with zoom of $v_1(\rho)$.




(b) Stable profile $v_0(\rho)$ and numerics.

**Fig. 8.** Left: The self-similar profiles $v_0(\rho)$ and $v_1(\rho)$. The inset shows the region near the origin for $v_1(\rho)$ to highlight the additional critical point. Right: Comparison of stable profile $v_0(\rho)$ (red line) to the moving mesh solution of (32) with $\beta = 0.15$ evolved to $(1+u) \approx 5 \times 10^{-4}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

forming at the origin. The solution on the dynamic mesh is interpolated onto a local polar grid centered at the origin. The self similar profile $v_0(\rho)$ is then compared with a radial version of the interpolated solution formed by averaging each $\rho$ value over the angular direction. For the case of a single touchdown point, we observe in Fig. 8 good agreement between the moving mesh solution and the rescaled self similar profile when $(1+u) \approx 5 \times 10^{-4}$ corresponding to $(t_c - t) \approx 1.2 \times 10^{-10}$.

### 3.3. Case 2: adaptivity after touchdown, $\varepsilon > 0$

In this section, we apply the moving mesh method to equation (1) in the regularized regime $\varepsilon > 0$. While the parameter $m$ can take different values depending on particular substrate and surface physics being modeled, we specialize here to $m = 4$ and solve

$$\frac{\partial u}{\partial t} = -(-\Delta)^p u - \frac{\lambda}{(1+u)^2} + \frac{\lambda \varepsilon^2}{(1+u)^4}, \quad (\mathbf{x}, t) \in \Omega \times (0, T), \tag{40a}$$

$$u = \Delta u = 0, \qquad (\mathbf{x}, t) \in \partial\Omega \times (0, T). \tag{40b}$$

In contrast to the $\varepsilon = 0$ case (32), the solution of (40) is globally well posed for all time given suitable initial data [63]. After an initial touchdown event, the solution develops sharp interfaces followed by additional dynamics before eventual equilibration.

The numerical resolution of these sharp features is significantly more challenging than the $\varepsilon = 0$ case for two reasons: First, the mesh must adaptively resolve to expanding sharp interfaces, rather than fine scale behavior around discrete points. These propagating interfaces grow, interact and merge and the mesh must accommodate these dynamics. The second challenge, as seen in Fig. 2b for the equivalent 1D problem, is that each interface has a distinct multiscale structure – the notorious *triple deck* phenomenon in high Reynolds number flow [52].

In our numerical simulations of (40), we initialize with a profile that is very close to touchdown so that the number and shape of the initial contact interfaces can be carefully controlled. We build a sharp interface initial condition around the planar curve $(x, y) = r(\theta)(\cos(\theta), \sin(\theta))$ for $0 \le \theta < 2\pi$ with

$$u_0(x, y) = a[\tanh[b((x - x_0)^2 + (y - y_0)^2 - r(\arctan(y/x))^2)] - 1], \tag{41}$$

where $(x_0, y_0)$ is the center of the interface in the domain. The parameter $a$ tunes the height of the profile while $b$ controls the sharpness of the interface. As the initial condition is highly localized, the PMA equation (9a) is iterated independently of the PDE with the arc-length monitor function (17) to obtain a suitable initial mesh. The coupled system (27) is then integrated until it the solution reaches equilibrium.

In the case of a single interface solution, the motion of the contact region can be expressed in terms of a geometric evolution law derived by a singular perturbation expansion. The two interface motion laws for (40) with the shelf like initial condition (41) were derived, in the limit $\varepsilon \to 0$, to be (Sec. 3 of [60])

$$\text{Case } p = 1. \qquad \rho_t = \sqrt{\frac{4\lambda}{3\varepsilon}} \frac{1}{(1 - 3\lambda t)^{\frac{1}{3}}} - \kappa + \mathcal{O}(1); \tag{42a}$$

$$\text{Case } p = 2. \qquad \rho_t = \left(\frac{4\lambda}{3\varepsilon}\right)^{\frac{3}{4}} \frac{1}{(1 - 3\lambda t)^{\frac{1}{2}}} - \sqrt{\frac{4\lambda}{3\varepsilon}} \frac{1}{(1 - 3\lambda t)^{\frac{1}{3}}} \kappa + \mathcal{O}(1), \tag{42b}$$

where $\kappa$ is the local curvature of the interface. The interface laws (42) can be evolved by a level set method [69] and the evolution compared with that arising from the moving mesh method. These asymptotic formula provide additional checks on the validity of the numerical method for highly non-trivial examples.

After accounting for initial transient behavior seen from the numerical approximation, we find good matching between the asymptotic approximation and the extracted curve for single interface solutions (see Fig. 13 and Fig. 14). We demonstrate the method on four different examples.

*Example 1:* Case $p = 1$ and a single interface with a five fold symmetry given by

$$r(\theta) = \sqrt{0.1}(1 + \cos(5\theta)).$$

The initial profile and its evolution are shown in Fig. 9. The parameter values are $\varepsilon = 0.05$, $\lambda = 10$ and $N = 81$ grid points are used. As discussed in Section 2.4, the regularized monitor function used is

$$M(\mathbf{x}, t) = 1 + |\nabla_{\mathbf{x}} u|^2 + \frac{1}{2}\int_{\Omega_P} (1 + |\nabla_{\mathbf{x}} u|^2)\, d\mathbf{x}'. \tag{43}$$

We remark that the factor $C = 1/2$ used in the integral averaging term is smaller than $C = 1$ used in (33a) for the $\varepsilon = 0$ case. This increases the mesh density along the interface where additional resolution is required.

As predicted by the geometric evolution law (42a), the interface grows while also circularizing. A comparison between the interface predicted by (42a) and the moving mesh simulation is shown in Fig. 13 and good qualitative agreement is observed.

*Example 2:* Case $p = 1$ with an initial condition featuring four circular interfaces of radius $r = 0.175$ and centers $(\pm 0.4, \pm 0.4)$ like the example shown in Fig. 6b. The monitor function (43) is also used in this example. The interfaces expand and coalesce before eventually reaching the boundary. The parameter values are $\varepsilon = 0.05$, $\lambda = 10$ and $N = 81$. Results shown in Fig. 10.

*Example 3:* Case $p = 2$ and a single interface. Parameter values $\varepsilon = 0.05$ and $\lambda = 20000$ and $N = 81$ mesh points. This example mimics a single touchdown point at the origin with a slightly perturbed elliptical shape given by

$$r(\theta) = c_1 c_2 [c_2^2 \cos^2\theta + c_1^2 \sin^2\theta]^{-\frac{1}{2}},$$

with $c_1 = 0.1312$ and $c_2 = 0.2625$. The regularized monitor function (Sec. 2.4) used in this example is

$$M(\mathbf{x}, t) = |\Delta_{\mathbf{x}} u|^2 + \frac{1}{2}\int_{\Omega_P} |\Delta_{\mathbf{x}} u|^2\, d\mathbf{x}', \tag{44}$$

where again, less integral averaging is employed to concentrate more mesh points along the interface.

The results are shown in Fig. 11 and there are several features to remark upon. First, as also observed in 1D behavior (cf. Fig. 2b), the layer has an oscillatory overshoot, a common feature of higher order problems which lack a maximum principle. Moreover, the amplitude of this overshoot varies along the length of the interface with its magnitude modulated by the local curvature. As the evolution progresses, the interface expands and circularizes as predicted by (42b). Finally, the interface motion is arrested once it reaches the domain boundary. In Fig. 14 we show a comparison between the interface predicted from the moving mesh solution and a level-set solution of (42b) which agree well.

*Example 4:* Case $p = 2$ with four initial contact regions and parameter values $\varepsilon = 0.05$, $\lambda = 20000$ and $N = 81$ grid points. The initial configuration, similar to that of Example 2, is inspired by taking the solution shown in Fig. 6d as an initial condition and evolving under the regularized equation (40). The initial condition has four circular interfaces each of radius $r = 0.175$ and arranged symmetrically at points $(\pm 0.4, \pm 0.4)$. The four interfaces begin by independently evolving and eventually merge to create a single front which is pinned by the boundary – see results in Fig. 12. This example is particularly challenging to resolve as the oscillatory overshoots from each layer interact in a complex fashion during the merging process. A comparison between Fig. 12 and the results from Example 2 (Fig. 10) in the $p = 1$ case shows much more complexity in the solution as the interfaces interact.

## 4. Conclusion

This paper has developed a dynamic mesh method for the simulation of fourth order PDEs in two dimensions via a parabolic Monge–Ampére equation. Adaptive methods are essential for resolving fine spatial scale and rapidly varying temporal behaviors in a variety of applications, including the study of elastic–electrostatic interactions in MEMS. A key detail in the implementation of this method is that the transformation between the computational and physical domains must be computed to higher order than previous approaches in the literature. This requirement for extra regularity is due to the additional derivatives that must be computed to represent fourth order derivatives in the PDE. This method has been validated on a variety of examples and the results suggest many avenues for future investigations.

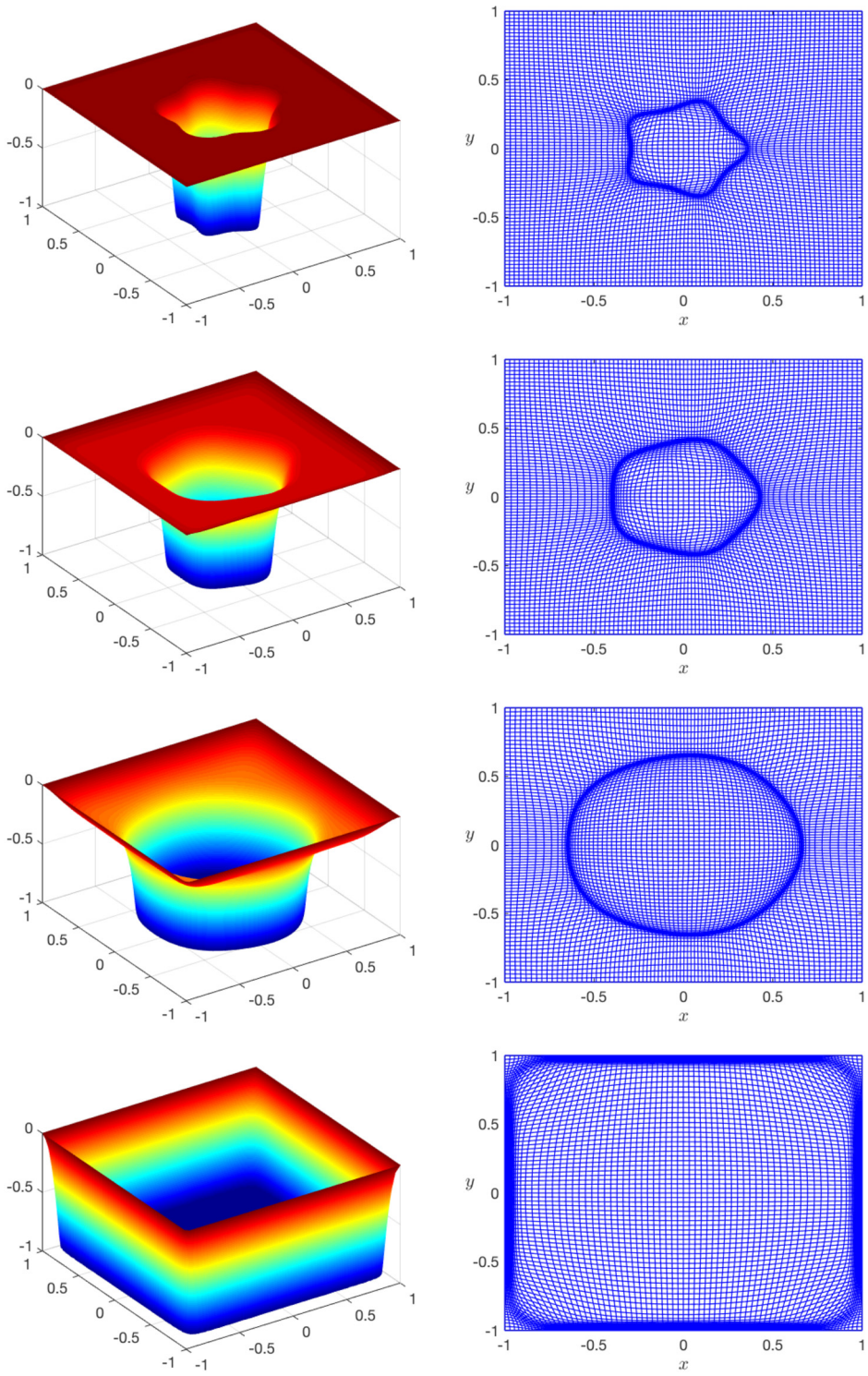



**Fig. 9.** Results for Example 1. Solution to (40) and mesh with $N = 81$ gridpoints and $\varepsilon = 0.05$, $p = 1$ and $\lambda = 10$ showing the solution of the PDE and its mesh for that respective time step.

In respect to the study of blow up in the fourth order PDE (32), we give strong numerical evidence of a stable self-similar singularity mechanism. This finding corroborates previous findings in one dimension [62] and other analytical approaches [14,38]. A rigorous proof of this limiting behavior remains a challenging open problem.

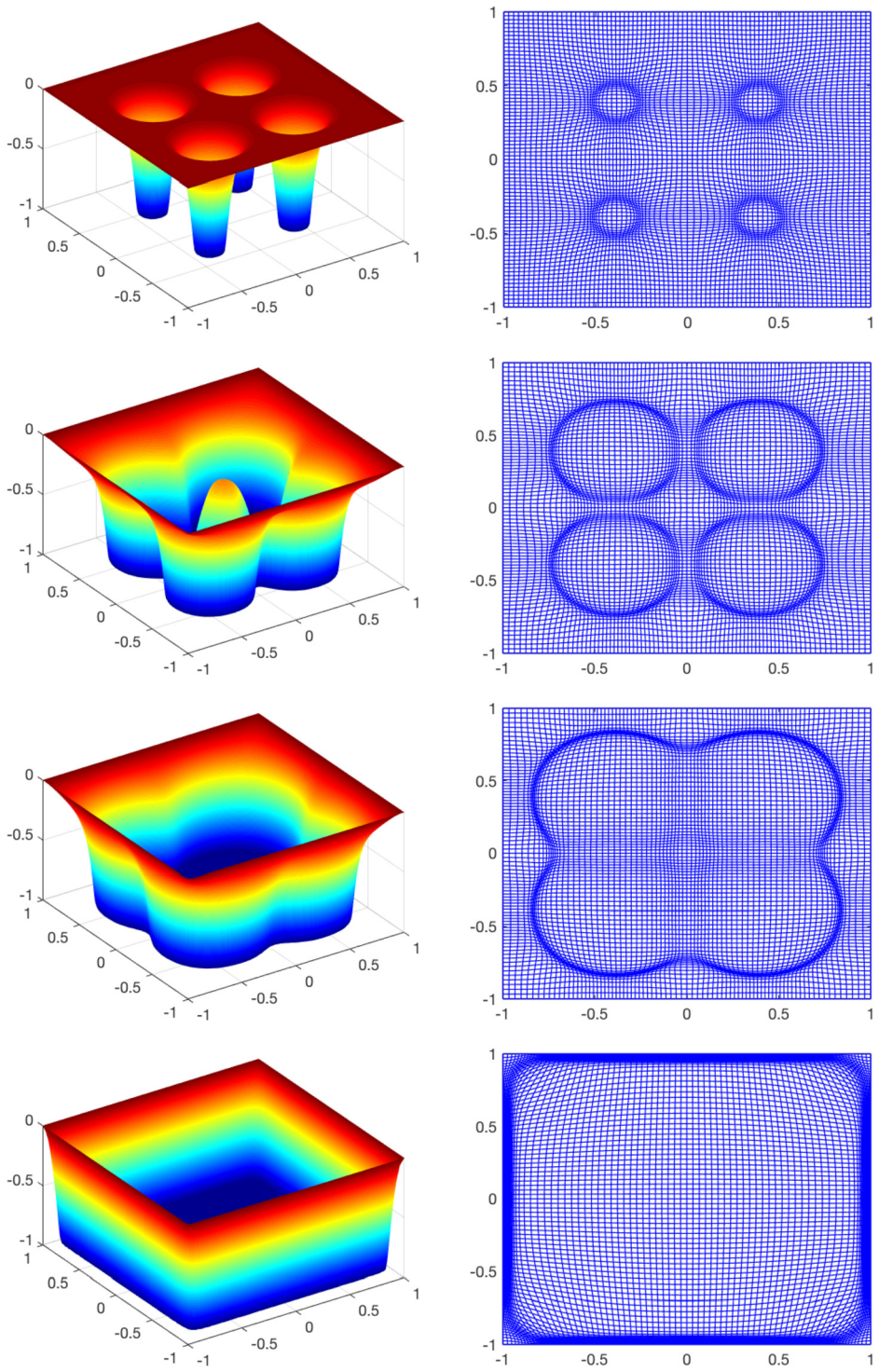


**Fig. 10.** Results for Example 2. Solutions to (40) in the Laplacian case ($p = 1$) and the corresponding mesh with $\varepsilon = 0.05$, $\lambda = 10$ and $N = 81$ gridpoints. The initial condition is four separate circular interfaces which interact, coalesce and propagate until pinned at the boundary.

A new conjecture arising from our study is that the fourth order PDE (32) can blow up sooner than its ODE counterpart. The counter intuitive aspect of this result is that while the spatial term $-\beta^2\Delta^2 u$ in the PDE is generally inhibitory to blow up, it may become an accelerant if $\beta$ is small enough. This phenomena is due to the tendency for oscillations in systems that lack of a maximum principle, such as higher order PDE systems. A rigorous proof of this phenomenon is highly desirable.

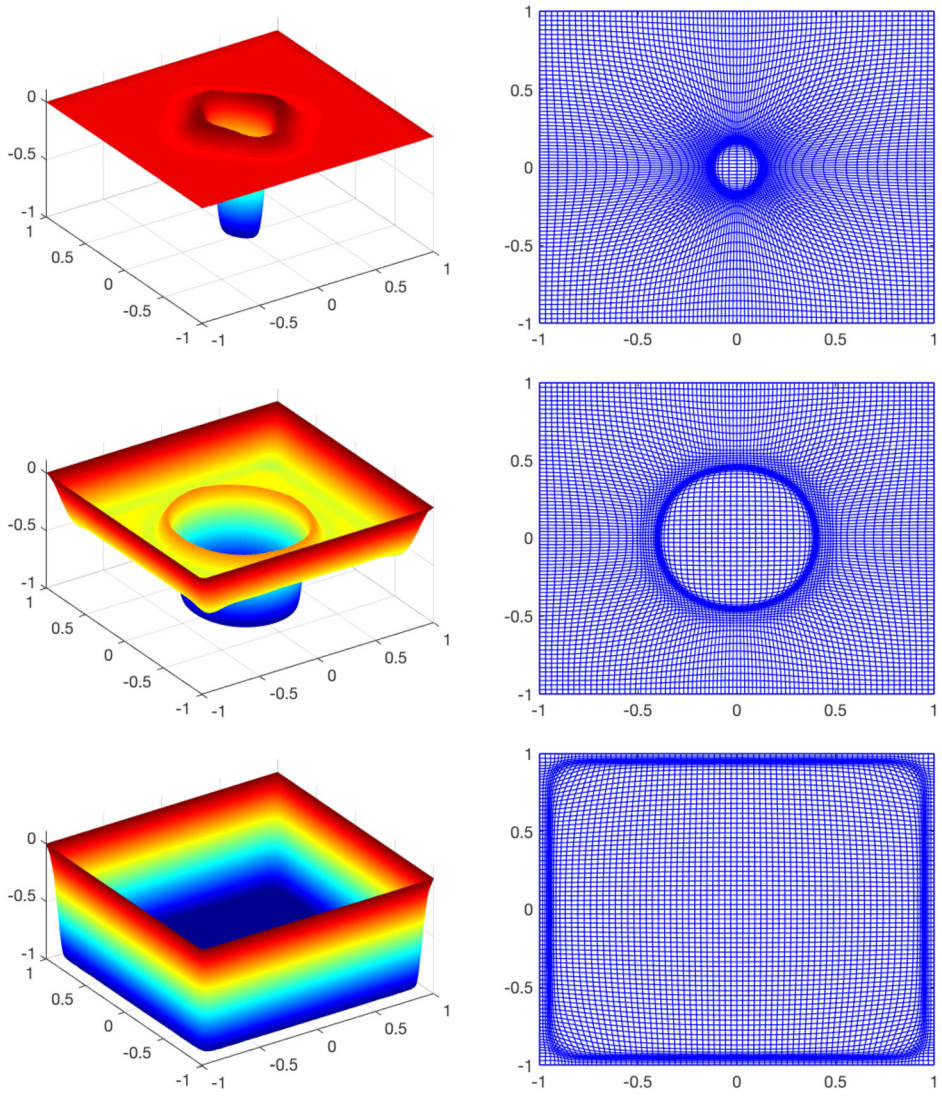


**Fig. 11.** Results for Example 3. Solutions to (40) and the corresponding mesh with $p = 2$, $\varepsilon = 0.05$, $\lambda = 20000$ and $N = 81$ grid points. In this case, the initial condition is a single elliptical interface which circularizes and expands before eventually being arrested at the boundary.

Finally, as with other quenching problems (cf. [58,61,62,76,39]), the location and multiplicity of singularities in (32) is highly dependent on the parameter $\beta$ and the shape of the domain $\Omega$. For these reasons an adaptive methodology which is able to locate and resolve singularities in arbitrary domains would be extremely valuable. This is particularly necessary for effective design MEMS since real world devices feature non-trivial geometries and may contain topological defects.

There are many potential approaches for extending the adaptive meshing techniques discussed in this paper to non-rectangular domains. One choice assumes the same curved, convex topology for the computational and physical domains and uses existing interpolation formulas for finite difference methods on curved boundaries (cf. [50] [65, Ch. 3.4, 6.4]) to evaluate the derivatives. This would keep the direct mapping between the computational and physical domains, at the cost of reduced boundary accuracy and inability to use the fast Fourier transform to invert the parabolic operator of the PMA. Another option would be to keep a logically rectangular computational domain and use a transformation (cf. [22]) to map it to a curved physical domain. In this method, the simplicity of the rectangular computation domain is preserved at the cost of an additional mapping function. Alternatively, the system (9) could be spatially discretized using finite element methods, which are more easily generalized to curved domains. Discontinuous Galerkin methods have been applied to directly solve the bi-Laplacian problem (cf. [5,41]) and mixed finite elements have been used to solve the bi-Laplacian as coupled system of second order equations (cf. [10]). However, the extension to the nonlinear MA equation is less straightforward. The use of quadratic mixed finite has been suggested by [4], but with a Dirichlet boundary condition that would need to be reformulated into the Neumann conditions of the PMA. The addition of a bi-Laplacian regularization term has also been suggested (cf. [32,33]) in order to create a finite element formulation for the MA equation.

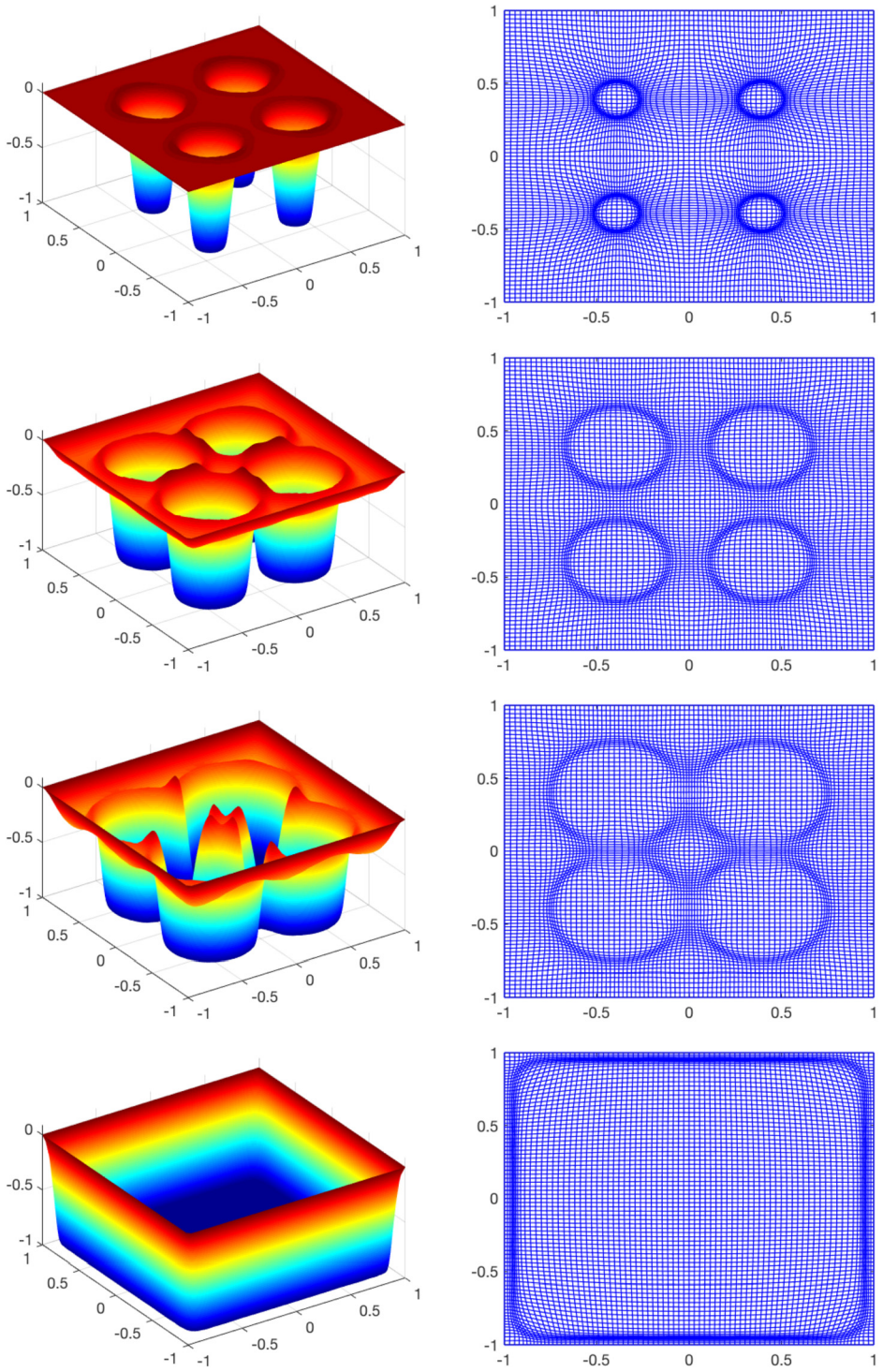


**Fig. 12.** Results for Example 4. Solutions to (40) in the bi-Laplacian case ($p = 2$) and the corresponding mesh. Parameters, $\varepsilon = 0.05$, $\lambda = 20000$ and $N = 81$ gridpoints. The initial condition is four circular interfaces which interact, coalesce and propagate until pinned at the boundary.

## Acknowledgements

A.E.L. acknowledgements support from the National Science Foundation under grant DMS-1516753. K.L.D. acknowledgements support from the National Science Foundation under grant DGE-1313583.

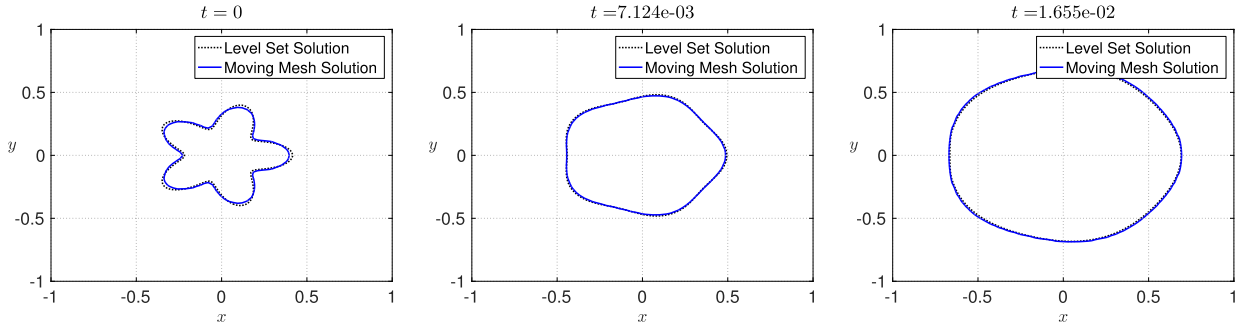



**Fig. 13.** Results for Example 1. Comparison between the geometric evolution law (42a) and the interface obtained from the moving mesh solution.

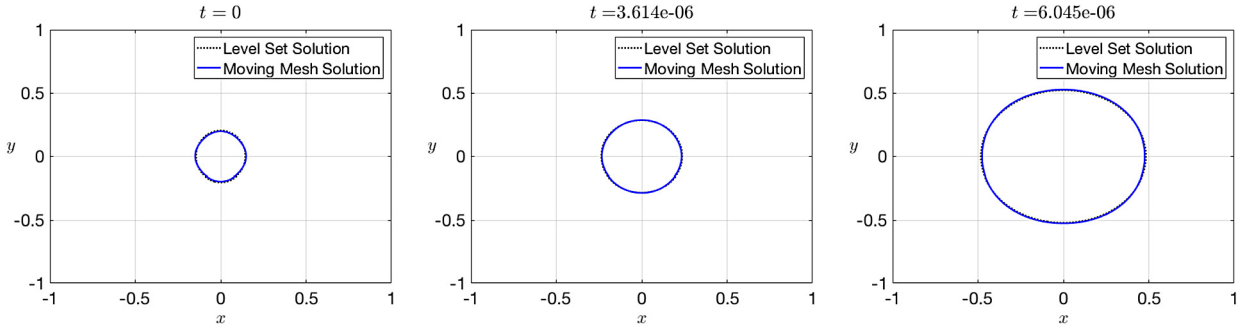



**Fig. 14.** Results for Example 3. Comparison between the geometric evolution law (42b) and the interface extraction from the moving mesh solution.

## Appendices

As discussed subsection 2.3, a fourth order discretization of the Laplacian is required to achieve desired convergence for the bi-Laplacian (particularly near the boundary). For completeness we provide the following differencing stencils on $\Omega_C = (-1,1)^2$:

- Fourth order stencils for the mesh derivatives $Q_{\xi\xi}$, $Q_{\eta\eta}$ and $Q_{\xi\eta}$, with inhomogeneous Neumann boundary conditions.
- Fourth order discretization of the Laplacian in computational space.
- A second order upwinding scheme for the coupling term.

## Appendix A. Fourth order mesh potential derivatives

### A.1. Discretizing $Q_{\xi\xi}$, $Q_{\eta\eta}$

At internal points $3 \le i \le N-2$ and $1 \le j \le N$,

$$(Q_{\xi\xi})_{i,j} = \frac{1}{12\Delta\xi^2}\Big( - Q_{i-2,j} + 16Q_{i-1,j} - 30Q_{i,j} + 16Q_{i+1,j} - Q_{i+2,j}\Big).$$

At boundary and adjacent to boundary points for $i = 1, 2$ and $1 \le j \le N$ ($i = N-1, N$ have the same stencils with the indices adjusted accordingly):

$$(Q_{\xi\xi})_{1,j} = \frac{1}{12\Delta\xi^2}\Big(\frac{-415}{16}Q_{1,j} + 96Q_{2,j} - 36Q_{3,j} + \frac{32}{3}Q_{4,j} - \frac{3}{2}Q_{5,j}\Big) + \frac{25}{6\Delta\xi},$$

$$(Q_{\xi\xi})_{2,j} = \frac{1}{12\Delta\xi^2}\Big(10Q_{1,j} - 15Q_{2,j} - 4Q_{3,j} + 14Q_{4,j} - 6Q_{5,j} + Q_{6,j}\Big).$$

The term $\frac{25}{6\Delta\xi}$ appearing in the stencil for $(Q_{\xi\xi})_{1,j}$ invokes the inhomogeneous Neumann boundary condition $Q_\xi = -1$ on $\xi = -1$ associated with the PMA (7b). The $Q_{\eta\eta}$ discretization is the same up to changes of indices.

### A.2. Discretizing $Q_{\eta\xi}$, $Q_{\xi\eta}$

On the boundary, $Q_{\xi\eta} = Q_{\eta\xi} = 0$ and so we discretize on the interior points. For the outer derivative we have for $1 \le i \le N$,

$$(Q_{\xi\eta})_{i,j} = \frac{1}{12\Delta\eta}\Big((Q_\xi)_{i,j-2} - 8(Q_\xi)_{i,j-1} + 8(Q_\xi)_{i,j+1} - (Q_\xi)_{i,j+2}\Big), \quad 3 \le j \le N-2;$$

$$(Q_{\xi\eta})_{i,j} = \frac{1}{12\Delta\eta}\Big(-3(Q_\xi)_{i,j-1} - 10(Q_\xi)_{i,j} + 18(Q_\xi)_{i,j+1} - 6(Q_\xi)_{i,j+2} + (Q_\xi)_{i,j+3}\Big), \quad j = 2;$$

$$(Q_{\xi\eta})_{i,j} = \frac{1}{12\Delta\eta}\Big(3(Q_\xi)_{i,j+1} + 10(Q_\xi)_{i,j} - 18(Q_\xi)_{i,j-1} + 6(Q_\xi)_{i,j-2} - (Q_\xi)_{i,j-3}\Big), \quad j = N-1;$$

For each of the derivatives $Q_\xi$, we use the following formulas for $1 \le j \le N$,

$$(Q_\xi)_{i,j} = \frac{1}{12\Delta\xi}\Big(Q_{i-2,j} - 8Q_{i-1,j} + 8Q_{i+1,j} - Q_{i+2,j}\Big), \quad 3 \le i \le N-2;$$

$$(Q_\xi)_{i,j} = \frac{1}{12\Delta\xi}\Big(-3Q_{i-1,j} - 10Q_{i,j} + 18Q_{i+1,j} - 6Q_{i+2,j} + Q_{i+3,j}\Big), \quad i = 2;$$

$$(Q_\xi)_{i,j} = \frac{1}{12\Delta\xi}\Big(3Q_{i+1,j} + 10Q_{i,j} - 18Q_{i-2,j} + 6Q_{i-3,j} - Q_{i-4,j}\Big), \quad i = N-1.$$

## Appendix B. Fourth order derivatives of the Laplacian

Here we describe the stencils used to discretize equation (15) with the boundary condition $u = 0$. The involves terms of the form $(Au_\xi)_\xi$, $(Bu_\eta)_\xi$, $(Cu_\xi)_\eta$, $(Du_\eta)_\eta$. We give full details for $(Au_\xi)_\xi$ and $(Bu_\eta)_\xi$ with $(Cu_\xi)_\eta$ and $(Du_\eta)_\eta$ being equivalent up to changes in indices.

### B.1. Discretizing $(Au_\xi)_\xi, (Du_\eta)_\eta$

*Interior points:* We discretize the outer $\partial_\xi$ derivative on a staggered grid and the internal derivatives with standard fourth order central differences. For $4 \le i \le N-3$, and $1 \le j \le N$,

$$[(Au_\xi)_\xi]_{i,j} = \frac{1}{6\Delta\xi}\Big(A_{i-1,j}(u_\xi)_{i-1,j} - 8A_{i-\frac{1}{2},j}(u_\xi)_{i-\frac{1}{2},j} + 8A_{i+\frac{1}{2},j}(u_\xi)_{i+\frac{1}{2},j} - A_{i+1,j}(u_\xi)_{i+1,j}\Big),$$

$$(u_\xi)_{k,j} = \frac{1}{12\Delta\xi}\Big(u_{k-2,j} - 8u_{k-1,j} + 8u_{k+1,j} - u_{k+2,j}\Big), \quad k = i \pm 1,$$

$$(u_\xi)_{k,j} = \frac{1}{24\Delta\xi}\Big(u_{k-\frac{3}{2},j} - 27u_{k-\frac{1}{2},j} + 27u_{k+\frac{1}{2},j} - u_{k+\frac{3}{2},j}\Big), \quad k = i \pm \frac{1}{2},$$

$$A_{k,j} = \frac{1}{16}\Big(-A_{k-\frac{3}{2},j} + 9A_{k-\frac{1}{2},j} + 9A_{k+\frac{1}{2},j} - A_{k+\frac{3}{2},j}\Big), \quad k = i \pm \frac{1}{2}.$$

*Boundary points:* For $i = 2, 3, N-2, N-1$, the product $(Au_\xi)_\xi = Au_{\xi\xi} + A_\xi u_\xi$ is expanded and each term discretized. We give stencils for $i = 2, 3$ and $i = N-2, N-1$ are equivalent up to changes in indices. For $1 \le j \le N$,

$$[(Au_\xi)_\xi]_{2,j} = \frac{1}{12\Delta\xi^2}A_{2,j}\Big(10u_{1,j} - 15u_{2,j} - 4u_{3,j} + 14u_{4,j} - 6u_{5,j} + u_{6,j}\Big) + \frac{1}{144\Delta\xi^2}\Big(-3u_{1,j} - 10u_{2,j} + 18u_{3,j} - 6u_{4,j} + u_{5,j}\Big)\times \Big(-3A_{1,j} - 10A_{2,j} + 18A_{3,j} - 6A_{4,j} + A_{5,j}\Big),$$

$$[(Au_\xi)_\xi]_{3,j} = \frac{1}{12\Delta\xi^2}A_{3,j}\Big(-u_{1,j} + 16u_{2,j} - 30u_{3,j} + 16u_{4,j} - u_{5,j}\Big) + \frac{1}{144\Delta\xi^2}\Big(u_{1,j} - 8u_{2,j} + 8u_{4,j} - u_{5,j}\Big)\Big(A_{1,j} - 8A_{2,j} + 8A_{4,j} - A_{5,j}\Big).$$

*B.2. Discretizing $(Bu_\eta)_\xi$, $(Cu_\xi)_\eta$*

The terms $(Bu_\eta)_\xi$, $(Cu_\xi)_\eta$ are discretized using the same stencils as $Q_{\xi\eta}$, $Q_{\eta\xi}$, with the additional coefficient terms $B, C$. For example, with the term $(Bu_\eta)_\xi$ we have for $1 \le j \le N$ that

$$[(Bu_\eta)_\xi]_{i,j} = \frac{1}{12\Delta\xi}\Big(B_{i-2,j}\,(u_\eta)_{i-2,j} - 8B_{i-1,j}\,(u_\eta)_{i-1,j} + 8B_{i+1,j}\,(u_\eta)_{i+1,j} - B_{i+2,j}\,(u_\eta)_{i+2,j}\Big), \qquad 3 \le i \le N-2; \tag{46a}$$

$$[(Bu_\eta)_\xi]_{i,j} = \frac{1}{12\Delta\xi}\Big(-3B_{i-1,j}\,(u_\eta)_{i-1,j} - 10B_{i,j}\,(u_\eta)_{i,j} + 18B_{i+1,j}\,(u_\eta)_{i+1,j} - 6B_{i+2,j}\,(u_\eta)_{i+2,j} + B_{i+3,j}\,(u_\eta)_{i+3,j}\Big), \qquad i = 2; \tag{46b}$$

$$[(Bu_\eta)_\xi]_{i,j} = \frac{1}{12\Delta\xi}\Big(3B_{i+1,j}\,(u_\eta)_{i+1,j} + 10B_{i,j}\,(u_\eta)_{i,j} - 18B_{i-1,j}\,(u_\eta)_{i-1,j} + 6B_{i-2,j}\,(u_\eta)_{i-2,j} - B_{i-3,j}\,(u_\eta)_{i-3,j}\Big), \qquad i = N-1. \tag{46c}$$

In (46), we use the following stencils for $u_\eta$ for $1 \le i \le N$,

$$(u_\eta)_{i,j} = \frac{1}{12\Delta\eta}\Big(u_{i,j-2} - 8u_{i,j-1} + 8u_{i,j+1} - u_{i,j+2}\Big), \quad 3 \le j \le N-2;$$

$$(u_\eta)_{i,j} = \frac{1}{12\Delta\eta}\Big(-3u_{i,j-1} - 10u_{i,j} + 18u_{i,j+1} - 6u_{i,j+2} + u_{i,j+3}\Big), \quad j = 2;$$

$$(u_\eta)_{i,j} = \frac{1}{12\Delta\eta}\Big(3u_{i,j+1} + 10u_{i,j} - 18u_{i,j-1} + 6u_{i,j-2} - u_{i,j-3}\Big), \quad j = N-1.$$

## Appendix C. Upwinding discretization

To maintain stability on the coupling term $\nabla_{\mathbf{x}} u \cdot \nabla \mathbf{x}_t$, we use a two dimensional, second order upwinding scheme. Recall that $\mathbf{x}_t = \nabla_{\boldsymbol{\xi}} Q_t$ and we discretize the $\nabla_{\mathbf{x}} u$ term by upwinding on $Q_t$ which is calculated explicitly from (11).

Let $\nabla_{\boldsymbol{\xi}} Q_t = (a, b) = (Q_{t\xi}, Q_{t\eta})$, where the first spatial derivative is calculated using fourth order central differencing. For example, $Q_{t\xi}$ is calculated as

$$a_{i,j} = (Q_{t\xi})_{i,j} = \frac{1}{12\Delta\xi}\Big((Q_t)_{i-2,j} - 8(Q_t)_{i-1,j} + 8(Q_t)_{i+1,j} - (Q_t)_{i+2,j}\Big), \tag{48a}$$

for $3 \le i \le N-2$ and $1 \le j \le N$. At the left boundaries the formulas are

$$a_{1,j} = (Q_{t\xi})_{1,j} = \frac{1}{12\Delta\xi}\Big(-25(Q_t)_{1,j} + 48(Q_t)_{2,j} - 36(Q_t)_{3,j} + 16(Q_t)_{4,j} - 3(Q_t)_{5,j}\Big),$$

$$a_{2,j} = (Q_{t\xi})_{2,j} = \frac{1}{12\Delta\xi}\Big(-3(Q_t)_{1,j} - 10(Q_t)_{2,j} + 18(Q_t)_{3,j} - 6(Q_t)_{4,j} + (Q_t)_{5,j}\Big),$$

for $1 \le j \le N$. The stencils for $i = N-1, N$ are equivalent after a change of sign and reversal of index order. The differencing formulas for $b = Q_{t\eta}$ are identical after indices are reversed. The coupling term is discretized by

$$(\nabla_{\mathbf{x}} u \cdot \nabla \mathbf{x}_t)_{i,j} = a_{i,j}(u_x)_{i,j} + b_{i,j}(u_y)_{i,j}\,,$$

where $u_x$, $u_y$ are broken into cases based on the direction of the mesh flow,

$$(u_x)_{i,j} = \begin{cases} (u_x^{bf})_{i,j}, & a_{i,j} < 0,\ b_{i,j} > 0; \\ (u_x^{bb})_{i,j}, & a_{i,j} < 0,\ b_{i,j} < 0; \\ (u_x^{ff})_{i,j}, & a_{i,j} > 0,\ b_{i,j} > 0; \\ (u_x^{fb})_{i,j}, & a_{i,j} > 0,\ b_{i,j} < 0. \end{cases} \qquad (u_y)_{i,j} = \begin{cases} (u_y^{bf})_{i,j}, & a_{i,j} < 0,\ b_{i,j} > 0; \\ (u_y^{bb})_{i,j}, & a_{i,j} < 0,\ b_{i,j} < 0; \\ (u_y^{ff})_{i,j}, & a_{i,j} > 0,\ b_{i,j} > 0; \\ (u_y^{fb})_{i,j}, & a_{i,j} > 0,\ b_{i,j} < 0. \end{cases} \tag{49}$$

In computational coordinates $u_x$ and $u_y$ depend on both $u_\xi$ and $u_\eta$ which must be upwinded. For example, the term $u_x^{bf}$ is the $u_x$ derivative with backward differencing in the $\xi$ direction and forward differencing in the $\eta$ direction. Applying the formula (13a), the discretization of this particular example has the following form:

$$u_x^{bf} = \bar{J}\,(Q_{\eta\eta}u_\xi^b - Q_{\xi\eta}u_\eta^f),$$

$$(u_\xi^b)_{i,j} = \frac{1}{2\Delta\xi}\Big(u_{i-2,j} - 4u_{i-1,j} + 3u_{i,j}\Big),$$

$$(u_\eta^f)_{i,j} = \frac{1}{2\Delta\eta}\Big(-3u_{i,j} + 4u_{i,j+1} - u_{i,j+2}\Big).$$

The other terms in (49) are discretized similarly using the second order forward and backward differencing stencils above.

## References


[1] Ronald D. Haynes, Alexander Bihlo, Emily J. Walsh, Stochastic domain decomposition for time dependent adaptive mesh generation, J. Math. Study 45 (2015) 106–124.
[2] Anubhav Arora, Mark Prausnitz, Samir Mitragotri, Micro-scale devices for transdermal drug delivery, Int. J. Pharm. 364 (2) (2008) 227–236.
[3] U. Ascher, R. Mattheij, R. Russell, Numerical Solution of Boundary Value Problems for Ordinary Differential Equations, Society for Industrial and Applied Mathematics, 1995.
[4] Gerard Awanou, Quadratic mixed finite element approximations of the Monge–Ampère equation in 2D, Calcolo 52 (4) (2015) 503–518.
[5] Garth A. Baker, Finite element methods for elliptic equations using nonconforming elements, Math. Comput. 31 (137) (1977) 45–59.
[6] Romesh C. Batra, Maurizio Porfiri, Davide Spinello, Effects of Van der Waals force and thermal stresses on pull-in instability of clamped rectangular microplates, Sensors 8 (2) (2008) 1048.
[7] G. Beckett, J.A. Mackenzie, Convergence analysis of finite difference approximations on equidistributed grids to a singularly perturbed boundary value problem, Appl. Numer. Math. 35 (2) (2000) 87–109.
[8] Andrew J. Bernoff, Thomas P. Witelski, Stability and dynamics of self-similarity in evolution equations, J. Eng. Math. 66 (1) (2010) 11–31.
[9] Yann Brenier, Polar factorization and monotone rearrangement of vector-valued functions, Commun. Pure Appl. Math. 44 (4) (1991) 375–417.
[10] Franco Brezzi, Michel Fortin, Mixed and Hybrid Finite Element Methods, vol. 15, Springer Science & Business Media, 2012.
[11] C.J. Budd, J.F. Williams, Parabolic Monge–Ampére methods for blow-up problems in several spatial dimensions, J. Phys. A, Math. Gen. 39 (19) (2006) 5425.
[12] C.J. Budd, J.F. Williams, Moving mesh generation using the parabolic Monge–Ampére equation, SIAM J. Sci. Comput. 31 (5) (2009) 3438–3465.
[13] C.J. Budd, J.F. Williams, How to adaptively resolve evolutionary singularities in differential equations with symmetry, J. Eng. Math. 66 (1) (2010) 217–236.
[14] C.J. Budd, J.F. Williams, V.A. Galaktionov, Self-similar blow-up in higher-order semilinear parabolic equations, SIAM J. Appl. Math. 64 (5) (2004) 1775–1809.
[15] Chris J. Budd, Weizhang Huang, Robert D. Russell, Adaptivity with moving grids, Acta Numer. 18 (2009) 111.
[16] C.J. Budd, M.J.P. Cullen, E.J. Walsh, Monge–Ampére based moving mesh methods for numerical weather prediction, with applications to the Eady problem, J. Comput. Phys. 236 (2013) 247–270.
[17] C.J. Budd, V.A. Galaktionov, J.F. Williams, Self-similar blow-up in higher-order semilinear parabolic equations, SIAM J. Appl. Math. 64 (5) (2004) 1775–1809.
[18] C.J. Budd, B. Leimkuhler, M.D. Piggott, Scaling invariance and adaptivity, Appl. Numer. Math. 39 (3) (2001) 261–288.
[19] C.J. Budd, R.D. Russell, E. Walsh, The geometry of r-adaptive meshes generated using optimal transport methods, J. Comput. Phys. 282 (2015) 113–137.
[20] Luis A. Caffarelli, Interior $W^{2,p}$ estimates for solutions of the Monge–Ampére equation, Ann. Math. 131 (1) (1990) 135–150.
[21] Luis A. Caffarelli, Boundary regularity of maps with convex potentials, Commun. Pure Appl. Math. 45 (9) (1992) 1141–1151.
[22] Donna A. Calhoun, Christiane Helzel, Randall J. LeVeque, Logically rectangular grids and finite volume methods for PDEs in circular and spherical domains, SIAM Rev. 50 (4) (2008) 723–752.
[23] Weiming Cao, Weizhang Huang, Robert D. Russell, A study of monitor functions for two-dimensional adaptive mesh generation, SIAM J. Sci. Comput. 20 (6) (1999) 1978–1994.
[24] Hector Ceniceros, A semi-implicit moving mesh method for the focusing nonlinear Schrodinger equation, Commun. Pure Appl. Anal. 1 (4) (2002) 1–18.
[25] Hector D. Ceniceros, Thomas Y. Hou, An efficient dynamically adaptive mesh for potentially singular solutions, J. Comput. Phys. 172 (2) (2001) 609–639.
[26] S. Jonathan Chapman, Philippe H. Trinh, Thomas P. Witelski, Exponential asymptotics for thin film rupture, SIAM J. Appl. Math. 73 (1) (2013) 232–253.
[27] Shibin Dai, Keith Promislow, Geometric evolution of bilayers under the functionalized Cahn–Hilliard equation, Proc. R. Soc. A, Math. Phys. Eng. Sci. 469 (2153) (2013).
[28] Carl de Boor, Good Approximation by Splines with Variable Knots. II, Springer Berlin Heidelberg, Berlin, Heidelberg, 1974, pp. 12–20.
[29] G.L. Delzanno, L. Chacón, J.M. Finn, Y. Chung, G. Lapenta, An optimal robust equidistribution method for two-dimensional grid adaptation based on Monge–Kantorovich optimization, J. Comput. Phys. 227 (23) (2008) 9841–9864.
[30] T.J. Dodwell, M.A. Peletier, C.J. Budd, G.W. Hunt, Self-similar voiding solutions of a single layered model of folding rocks, SIAM J. Appl. Math. 72 (1) (2012) 444–463.
[31] P. Esposito, N. Ghoussoub, Y. Guo, Mathematical Analysis of Partial Differential Equations Modeling Electrostatic MEMS, Courant Lecture Notes, 2010.
[32] Xiaobing Feng, Michael Neilan, Mixed finite element methods for the fully nonlinear Monge–Ampére equation based on the vanishing moment method, SIAM J. Numer. Anal. 47 (2) (2009) 1226–1250.
[33] Xiaobing Feng, Michael Neilan, Analysis of Galerkin methods for the fully nonlinear Monge–Ampère equation, J. Sci. Comput. 47 (3) (2011) 303–327.
[34] Gadi Fibich, Self-focusing: past and present, Top. Appl. Phys. 114 (2009) 413–438.
[35] Stathis Filippas, Robert V. Kohn, Refined asymptotics for the blowup of $u_t - \delta u = u^p$, Commun. Pure Appl. Math. XLV (1992) 821–869.
[36] V.A. Galaktionov, Five types of blow-up in a semilinear fourth-order reaction diffusion equation: an analytical-numerical approach, Nonlinearity 22 (7) (2009) 1695.
[37] V.A. Galaktionov, Five types of blow-up in a semilinear fourth-order reaction–diffusion equation: an analytical-numerical approach, Nonlinearity 22 (2009) 1695–1741.
[38] V.A. Galaktionov, J.F. Williams, Blow-up in a forth-order semilinear parabolic equation from explosion–convection theory, Eur. J. Appl. Math. 14 (2003) 745–764.
[39] Victor A. Galaktionov, Juan-Luis Vázquez, The problem of blow-up in nonlinear parabolic equations, Discrete Contin. Dyn. Syst. 8 (2) (2002) 399–433.
[40] Nir Gavish, Gurgen Hayrapetyan, Keith Promislow, Li Yang, Curvature driven flow of bi-layer interfaces, Physica D 240 (7) (2011) 675–693.
[41] Emmanuil H. Georgoulis, Paul Houston, Discontinuous Galerkin methods for the biharmonic problem, IMA J. Numer. Anal. 29 (3) (2009) 573.
[42] Karl B. Glasner, Alan E. Lindsay, The stability and evolution of curved domains arising from one-dimensional localized patterns, SIAM J. Appl. Dyn. Syst. 12 (2) (2013) 650–673.

[43] A.C.R. Grayson, R.S. Shawgo, A.M. Johnson, N.T. Flynn, Yawen Li, M.J. Cima, R. Langer, A BioMEMS review: MEMS technology for physiologically integrated devices, Proc. IEEE 92 (1) (January 2004) 6–21.
[44] Jian-Gang Guo, Ya-Pu Zhao, Influence of Van der Waals and Casimir forces on electrostatic torsional actuators, J. Microelectromech. Syst. 13 (6) (2004) 1027–1035.
[45] Yujin Guo, Zhenguo Pan, M.J. Ward, Touchdown and pull-in voltage behavior of a MEMS device with varying dielectric properties, SIAM J. Appl. Math. 66 (1) (2005) 309–338.
[46] Weizhang Huang, Measuring mesh qualities and application to variational mesh adaptation, SIAM J. Sci. Comput. 26 (5) (2005) 1643–1666.
[47] Weizhang Huang, Benedict Leimkuhler, The adaptive Verlet method, SIAM J. Sci. Comput. 18 (1) (1997) 239–256.
[48] Weizhang Huang, Robert D. Russell, Moving mesh strategy based on a gradient flow equation for two-dimensional problems, SIAM J. Sci. Comput. 20 (3) (1998) 998–1015.
[49] Weizhang Huang, Robert D. Russell, Adaptive mesh movement – the MMPDE approach and its applications, J. Comput. Appl. Math. 128 (1–2) (2001) 383–398.
[50] Bruce Hunt, Finite difference approximation of boundary conditions along irregular boundaries, Int. J. Numer. Methods Eng. 12 (2) (1978) 229–235.
[51] Hangjie Ji, Thomas P. Witelski, Finite-time thin film rupture driven by modified evaporative loss, Physica D 342 (2017) 1–15.
[52] A. Kluwick, M. Kornfeld, Triple-deck analysis of transonic high Reynolds number flow through slender channels, Philos. Trans. R. Soc. Lond., Ser. A, Math. Phys. Eng. Sci. 372 (2020) (2014).
[53] B. Lai, On the partial differential equations of electrostatic MEMS with effects of Casimir force, Ann. Henri Poincaré 16 (1) (2015) 239–253.
[54] Ruo Li, Tao Tang, Pingwen Zhang, Moving mesh methods in multiple dimensions based on harmonic maps, J. Comput. Phys. 170 (2) (2001) 562–588.
[55] Shengtai Li, Linda Petzold, Moving mesh methods with upwinding schemes for time-dependent PDEs, J. Comput. Phys. 131 (2) (1997) 368–377.
[56] Shengtai Li, Linda Petzold, Yuhe Ren, Stability of moving mesh systems of partial differential equations, SIAM J. Sci. Comput. 20 (2) (1998) 719–738.
[57] Xianping Li, Anisotropic mesh adaptation for image representation, Int. J. Image Video Process. 2016 (1) (September 2016) 26.
[58] A.E. Lindsay, An asymptotic study of blow up multiplicity in fourth order parabolic partial differential equations, Discrete Contin. Dyn. Syst., Ser. B 19 (1) (2014) 189–215.
[59] A.E. Lindsay, Regularized model of post-touchdown configurations in electrostatic MEMS: bistability analysis, J. Eng. Math. 99 (1) (2016) 65–77.
[60] A.E. Lindsay, J. Lega, K.B. Glasner, Regularized model of post-touchdown configurations in electrostatic MEMS: interface dynamics, IMA J. Appl. Math. 80 (6) (2015) 1635–1663.
[61] A.E. Lindsay, J. Lega, F.J. Sayas, The quenching set of a MEMS capacitor in two-dimensional geometries, J. Nonlinear Sci. 23 (5) (2013) 807–834.
[62] A.E. Lindsay, J. Lega, Multiple quenching solutions of a fourth order parabolic PDE with a singular nonlinearity modeling a MEMS capacitor, SIAM J. Appl. Math. 72 (3) (2012) 935–958.
[63] A.E. Lindsay, J. Lega, K.B. Glasner, Regularized model of post-touchdown configurations in electrostatic MEMS: equilibrium analysis, Physica D 280–281 (2014) 95–108.
[64] H.-W. Lu, K. Glasner, A.L. Bertozzi, C.-J. Kim, A diffuse-interface model for electrowetting drops in a Hele–Shaw cell, J. Fluid Mech. 590 (2007) 411–435.
[65] K.W. Morton, D.F. Mayers, Numerical Solution of Partial Differential Equations: An Introduction, ISBN 0521607930, Cambridge University Press, New York, NY, USA, 2005.
[66] John A. Pelesko, Mathematical modeling of electrostatic MEMS with tailored dielectric properties, SIAM J. Appl. Math. 62 (3) (2002) 888–908.
[67] Joy C. Perkinson, Michael J. Aziz, Michael P. Brenner, Miranda Holmes-Cerfon, Designing steep, sharp patterns on uniformly ion-bombarded surfaces, Proc. Natl. Acad. Sci. USA 113 (41) (2016) 11425–11430.
[68] R.D. Russell, J.F. Williams, X. Xu, Movcol4: a moving mesh code for fourth-order time-dependent partial differential equations, SIAM J. Sci. Comput. 29 (1) (2007) 197–220.
[69] Peter Smereka, Semi-implicit level set methods for curvature and surface diffusion motion, J. Sci. Comput. 19 (1) (2003) 439–456.
[70] Mohamed H.M. Sulman, Optimal mass transport-based adaptive mesh method for phase-field models of two-phase fluid flows, Comput. Math. Appl. 72 (9) (2016) 2181–2193.
[71] Mohamed M. Sulman, J.F. Williams, Robert D. Russell, An efficient approach for the numerical solution of the Monge–Ampère equation, Appl. Numer. Math. 61 (3) (2011) 298–307.
[72] Tao Tang, Moving mesh methods for computational fluid dynamics, Contemp. Math. 383 (2001) 141–174.
[73] J.T.M. van Beek, R. Puers, A review of MEMS oscillators for frequency reference and timing applications, J. Micromech. Microeng. 22 (1) (2012) 013001.
[74] Robert D. Russell Weizhang Huang, Analysis of moving mesh partial differential equations with spatial smoothing, SIAM J. Numer. Anal. 34 (3) (1997) 1106–1126.
[75] Thomas P. Witelski, Andrew J. Bernoff, Stability of self-similar solutions for Van der Waals driven thin film rupture, Phys. Fluids 11 (9) (1999) 2443–2445.
[76] Thomas P. Witelski, Andrew J. Bernoff, Dynamics of three-dimensional thin film rupture, Physica D 147 (1) (2000) 155–176.
[77] R. Zhang, L. Cai, On the semi linear equations of electrostatic NEMS devices, Z. Angew. Math. Phys. 65 (2014) 1207–1222.