

5-2014

Analysis of a Partial Differential Equation Model of Surface Electromigration

Selahittin Cinar

Western Kentucky University, selahittin.cinar214@topper.wku.edu

Follow this and additional works at: <http://digitalcommons.wku.edu/theses>



Part of the [Numerical Analysis and Computation Commons](#), [Partial Differential Equations Commons](#), and the [Physics Commons](#)

Recommended Citation

Cinar, Selahittin, "Analysis of a Partial Differential Equation Model of Surface Electromigration" (2014). *Masters Theses & Specialist Projects*. Paper 1368.

<http://digitalcommons.wku.edu/theses/1368>

This Thesis is brought to you for free and open access by TopSCHOLAR®. It has been accepted for inclusion in Masters Theses & Specialist Projects by an authorized administrator of TopSCHOLAR®. For more information, please contact connie.foster@wku.edu.

ANALYSIS OF A PARTIAL DIFFERENTIAL EQUATION MODEL OF
SURFACE ELECTROMIGRATION

A Thesis
Presented to
The Faculty of the Department of Mathematics
Western Kentucky University
Bowling Green, Kentucky

In Partial Fulfillment
Of the Requirements for the Degree
Master of Science

By
Selahittin Cinar

May 2014

ANALYSIS OF A PARTIAL DIFFERENTIAL EQUATION MODEL OF SURFACE
ELECTROMIGRATION

Date Recommended 4-10-2014

Mikhail Khenner

Mikhail Khenner, Director of Thesis

Mark P. Robinson

Mark Robinson

Richard Schugart

Richard Schugart

Carl A. Fero
Dean, Graduate Studies and Research

5-5-14
Date

I dedicate this thesis to my wife, Canan Cinar, who supported me each step of the way.

ACKNOWLEDGMENTS

I would like to thank to my advisor, Dr. Mikhail Khenner, for all of his help and support throughout this entire process. He has shown great patience and has always been helpful. This thesis literally would not have been done without him.

I would like to thank the other members of my committee, Dr. Mark Robinson and Dr. Richard Schugart for serving as my committee members and providing me with their feedback.

I would like to give special thanks to Dr. Ferhan Atici for her support and encouragement during my application process to WKU.

Finally, I would like to thank all my friends in the mathematics department for their support.

CONTENTS

List of Figures	vi
List of Tables	viii
Abstract	ix
Chapter 1. INTRODUCTION	1
1.1. Problem Statement	2
Chapter 2. LINEAR STABILITY ANALYSIS	8
2.1. Linearization of The Problem	8
2.2. Analysis of The Equation (2.1.7)	11
Chapter 3. METHOD OF LINES	14
3.1. Introduction to Method of Lines	14
3.2. Examples	18
3.3. Analysis of The Simplified Nonlinear PDE	24
Chapter 4. CONCLUSION AND FUTURE WORK	30
APPENDIX	32
Bibliography	37

LIST OF FIGURES

1.0.1	Horizontal electric field results in the crystal surface faceting.	2
1.1.1	Sketch of the metal film surface $h(x, t)$.	3
1.1.2	Two cases of the typical growth rate $\omega(k)$.	6
2.2.1	Characterization of the film linear stability.	13
3.2.1	This figure shows comparison of solutions for Example 3.2.1.	20
3.2.2	This figure shows the entire curve difference (absolute error) of Example 3.2.1.	21
3.2.3	Periodic BCs.	22
3.2.4	This figure shows comparison of solutions for Example 3.2.2.	24
3.2.5	This figure shows the entire curve difference (absolute error) for Example 3.2.2.	24
3.3.1	This figure shows comparison of solutions.	26
3.3.2	This figure shows the entire curve difference (absolute error) for Eq. (3.3.4) and Eq. (3.3.5).	27
3.3.3	Evolution of the graph $h(x, t)$ in time.	27
3.3.4	Last profile curve vs cosine curve.	28
3.3.5	Amplitude vs physical parameters.	29
4.0.1	Basic <i>Mathematica</i> code for the nonlinear PDE, part 1.	32
4.0.2	Basic <i>Mathematica</i> code for growth rate function, part 2.	33

4.0.3 Basic <i>Mathematica</i> code for Example 3.2.1.	33
4.0.4 Basic <i>Mathematica</i> code for Example 3.2.1 (error (a)).	34
4.0.5 Basic <i>Mathematica</i> code for Example 3.2.2.	34
4.0.6 Basic <i>Mathematica</i> code for Example 3.2.2 (error (a)).	35
4.0.7 Basic <i>Mathematica</i> code for the simplified-nonlinear PDE and error.	36

LIST OF TABLES

2.1.1 The physical parameter values used in our study.	11
--	----

ANALYSIS OF A PARTIAL DIFFERENTIAL EQUATION MODEL OF SURFACE ELECTROMIGRATION

Selahittin Cinar

May 2014

38 Pages

Directed by: Mikhail Khenner, Mark Robinson, and Richard Schugart

Department of Mathematics

Western Kentucky University

A Partial Differential Equation (PDE) based model combining surface electromigration and wetting is developed for the analysis of the morphological instability of mono-crystalline metal films in a high temperature environment typical to operational conditions of microelectronic interconnects. The atomic mobility and surface energy of such films are anisotropic, and the model accounts for these material properties. The goal of modeling is to describe and understand the time-evolution of the shape of film surface. I will present the formulation of a nonlinear parabolic PDE problem for the height function $h(x, t)$ of the film in the horizontal electric field, followed by the results of the linear stability analyses and computations of fully nonlinear evolution equation.

CHAPTER 1

INTRODUCTION

Electromigration is the mass transport of a metal due to the momentum transfer between conducting electrons and diffusing metal atoms. Although the phenomenon of electromigration has been known for over 100 years, it became of practical interest in 1966 when the first integrated circuits became commercially available. It is considered a key factor in determining the reliability of integrated circuits.

For example, nanoscale contacts manufactured from thin Ag films wetting the Si(100) substrate are described in [2]. The gap between contacts can be cyclically opened and closed. To open the contact, the electromigration current is applied at the low temperature of the contact (80 K), which enables the induced surface mass flow of atoms across the narrow bridge connecting the anode and the cathode until the bridge breaks. To close the contact, the natural surface diffusion of atoms across the gap is enabled by heating the film to the room temperature and simultaneously passing the current. See the middle and the right panels of Figure 1.0.1. Wetting in this context means the existence of an intermolecular attractive force acting between the atoms on the film surface and the substrate atoms, due to the very small thickness of the film (several tens of nanometers or less) [2].

Another application is the fabrication of structures on surfaces (called quantum wires) by inducing and tailoring surface instability through electromigration, see the left panel of Figure 1.0.1.

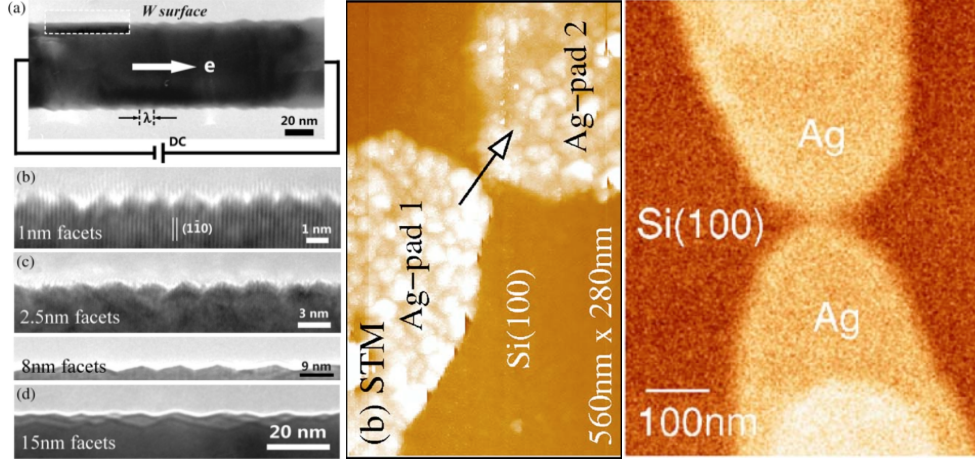


FIGURE 1.0.1. **Left:** Horizontal electric field results in the crystal surface faceting. The long ridges with triangular cross-section, formed by two adjacent facets, are the quantum wires. The figure shows the side view of such surface [10] as the size of the wires increases due to coarsening. **Middle:** A gap of around 5 nm is shown after the electromigration current has been applied [2]. **Right:** Ag structures that barely touch, but are still connected by a bridge several nanometers wide, can be made [2].

1.1. Problem Statement

Following [3] and [4], we concentrate on the case of the *horizontal* electric field, incorporating the effects of substrate wetting by the film, anisotropy of the diffusional mobility $M(\theta)$, and weak anisotropy of the surface energy $\gamma(\theta)$, where θ is the angle that the unit normal to the film surface makes with the vertical coordinate axis z . E stands for the electric field, x is the horizontal axis, z is the vertical axis, n is the normal vector, θ is the angle between the normal vector and the z -axis, E_{loc} is the local electric field, j is the atom flux vector, and $h(x, t)$ is the equation of the curve representing the film surface. (The curve $h(x, t)$ is termed the film surface in the following, despite being a one-dimensional object.) Figure 1.1.1 (a) depicts the electric field being oriented in the opposite direction. This forces the atoms downhill, making the surface more planar. Figure 1.1.1 (b) depicts a situation

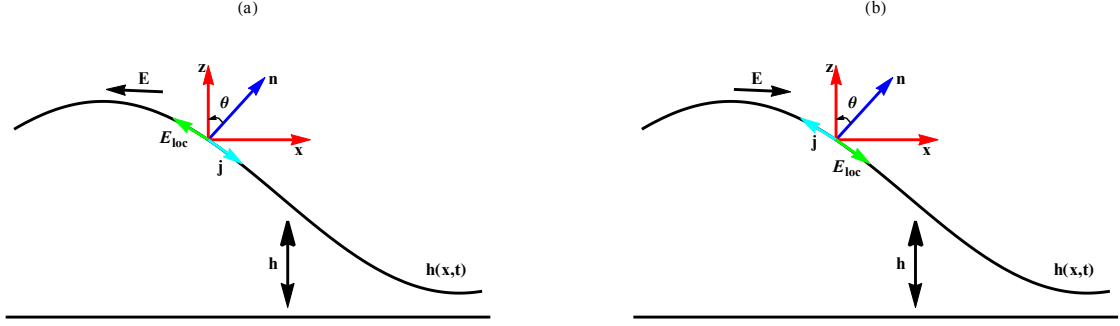


FIGURE 1.1.1. Sketch of the metal film surface $h(x, t)$ in the horizontal, constant electric field E . E_{loc} is the projection of E on the surface. The atom flux j on the surface is in the direction opposite to the direction of E_{loc} . (a) $E_{loc} = -E \cos(\theta)$, (b) $E_{loc} = E \cos(\theta)$.

in which the electric field is oriented in a manner that the atoms at the incline move uphill from the trough to the crest and therefore the surface becomes less planar.

Assuming no surface overhangs (none is expected if the field is horizontal), the key governing equation is the PDE (10) from [3] for the (non-negative) height function of the film $h(x, t)$:

$$h_t = B \left[M(h_x) (1 + h_x^2)^{-1/2} \mu_x \right]_x + A \left[M(h_x) (1 + h_x^2)^{-1/2} \right]_x. \quad (1.1.1)$$

In equation (1.1.1), A, B are positive parameters;

$$M(h_x) = \frac{1 + \beta \cos^2 [N(\arctan(h_x) + \phi)]}{1 + \beta \cos^2 [N\phi]}, \quad \beta, N, \phi = const.; \quad (1.1.2)$$

(Notice that the denominator of the fraction is a constant value for given physical values of β, N and ϕ .)

$\mu(x, t)$ is the surface chemical potential:

$$\mu = (\gamma + \gamma_{\theta\theta}) \kappa + (\gamma_h - h_x \gamma_{h\theta}) \cos \theta, \quad \cos \theta = (1 + h_x^2)^{-1/2}, \quad \kappa = -h_{xx} (1 + h_x^2)^{-3/2}; \quad (1.1.3)$$

$\gamma(x, t)$ is the *anisotropic* film surface energy (tension):

$$\gamma(h, \theta) = 1 + \epsilon_\gamma \cos 4\theta + (G - 1 - \epsilon_\gamma \cos 4\theta) e^{-h}, \quad \theta = \arctan(h_x), \quad (1.1.4)$$

where $G, 0 \leq \epsilon_\gamma < 1/30$ are the parameters. G is the ratio of the substrate energy to the mean value of the film surface energy, and ϵ_γ is the strength of anisotropy. This anisotropy (dependence on orientation, i.e., the angle θ) is the new effect that was not accounted for in [3], and which makes the nonlinear equation structure more complex and analysis significantly more difficult.

Eq. (1.1.1) is the fourth-order, nonlinear parabolic PDE. The prototype *linear* fourth-order parabolic PDE is

$$h_t = \alpha_1 h_{xxxx} + \alpha_2 h_{xx}, \quad \alpha_1, \alpha_2 < 0. \quad (1.1.5)$$

This equation has the trivial solution $h(x, t) = h_0 = \text{const.}$ In the physical context, this solution corresponds to a film of height = *const.* for all x, t , that is a film with planar stationary surface. We call such solution an equilibrium surface. The key issue is whether the equilibrium is stable or unstable with respect to small perturbations $\xi(x, t)$. For Eq. (1.1.5), this can be settled by substituting $h = h_0 + \xi(x, t)$ and then assuming $\xi(x, t)$ has a single Fourier mode:

$\xi(x, t) = \xi_0 e^{\omega t} e^{ikx}$, where $\xi_0 = \text{const.}$, ω , and k are the amplitude, the growth rate,

and the wavenumber, respectively. (The wavelength, or the spatial period, is $\lambda = 2\pi/k$.) Then one obtains the expression for the perturbation growth rate as a function of the wavenumber, the so-called dispersion relation:

$$\omega(k) = -\alpha_2 k^2 + \alpha_1 k^4. \quad (1.1.6)$$

For small k , the first term is dominant in this expression, for large k it is the second term. Because $\alpha_2 < 0$, perturbations with small wavenumbers (large wavelengths) grow ($\omega(k) > 0$); because $\alpha_1 < 0$, perturbations with large wavenumbers (small wavelengths) decay ($\omega(k) < 0$). This is reflected in the shape of the graph of the function $\omega(k)$, see Figure 1.1.2(a), and correspondingly the instability is termed the *long-wavelength* instability. All perturbations with wavenumbers in the interval $0 < k < k_c$ grow, and all perturbation with wavenumbers $> k_c$ decay. k_c is termed the instability *cut-off wavenumber*. The surface is unstable with respect to long-wavelength perturbations, and it is stable with respect to small-wavelength perturbations. In practice, the perturbation (induced, for instance, by thermal noise) is not a single Fourier mode, however most perturbations can be represented by a superposition of Fourier modes. Thus some modes grow and some decay. Among the unstable modes, there is a mode with the largest growth rate, ω_{max} . This *most dangerous* mode will dominate over other modes shortly after the surface is destabilized, resulting in a surface deformation having the form $h(x, t) = h_0 + \xi_0 e^{\omega_{max} t} \cos k_{max} x$. Here, k_{max} is the wavenumber for which $\omega = \omega_{max}$, i.e., the maximum of $\omega(k)$ on the interval $0 < k < k_c$. In other words, k_{max} is the positive solution of $d\omega/dk = 0$.

It is easy to show that for Eq. (1.1.6) $k_{max} = \frac{k_c}{\sqrt{2}}$. First, we set the rhs of Eq. (1.1.6) to zero and solve for k , that is:

$$\omega(k) = -\alpha_2 k^2 + \alpha_1 k^4 = 0 \Rightarrow k^2(-\alpha_2 + \alpha_1 k^2) = 0 \Rightarrow k = 0 \quad \text{or} \quad k = \pm \sqrt{\frac{\alpha_2}{\alpha_1}}.$$

We only consider positive solution, so we obtain $k_c = \sqrt{\frac{\alpha_2}{\alpha_1}}$.

To determine k_{max} , we solve $d\omega/dk = 0$ for k , that is:

$$-2\alpha_2 k + 4\alpha_1 k^3 = 0 \Rightarrow 2k(-\alpha_2 + 2\alpha_1 k^2) = 0 \Rightarrow k = 0 \quad \text{or} \quad k = \pm \sqrt{\frac{\alpha_2}{2\alpha_1}}.$$

k_{max} is the positive solution. Thus, $k_{max} = \sqrt{\frac{\alpha_2}{2\alpha_1}} = \sqrt{\frac{\alpha_2}{\alpha_1}} \frac{1}{\sqrt{2}} = \frac{k_c}{\sqrt{2}}$.

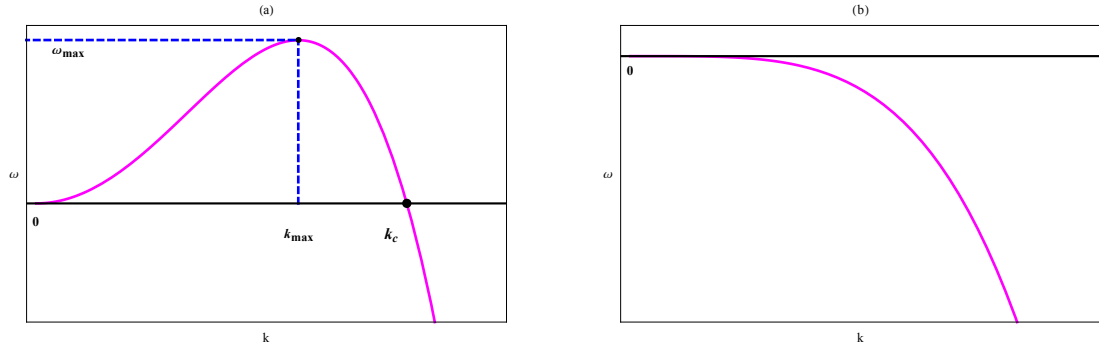


FIGURE 1.1.2. Two cases of the typical growth rate $\omega(k)$. (a) Long-wave instability; (b) $\omega(k) < 0, \forall k$: stability.

Notice that the parameter α_1 in Eq. (1.1.5) cannot be positive. (Otherwise, the short-wavelength perturbations will grow, which is not physically permissible, since in this case the surface is always unstable - such perturbations are always present in the spectrum. However, instability does not always emerge and most material surfaces remain planar. Mathematically, Eq. (1.1.5) in the case of $\alpha_1 > 0$ is ill-posed; despite its higher order (fourth vs. second) it is similar to the (ill-posed)

backward heat equation $h_t = -h_{xx}$.) However, the parameter α_2 may be positive for some physical parameters values. Then $\omega(k)$ is negative for all k (Figure 1.1.2 (b)), meaning that all perturbations decay and the surface restores its initial planar shape.

Equations such as (1.1.1) are nonlinear, thus the growth of the most dangerous mode will not continue forever. Nonlinear terms in the equation will damp growth, which usually results in a stationary, nontrivial solution which has the spatial form resembling the large-amplitude cosine curve. Determination of the stability of Eq. (1.1.1) and the form of the stationary solution are the goals of this Thesis.

CHAPTER 2

LINEAR STABILITY ANALYSIS

2.1. Linearization of The Problem

Toward our goal of determining stability of the film surface with respect to small perturbations, we first linearize $M(h_x)$ about $h_x = 0$, i.e., we write $M(h_x) = M(0) + M'(0)h_x$, where $M(0)$ and $M'(0)$ will be later calculated from Eq. (1.1.2) for given β, N and ϕ (see [3]). Then:

$$\frac{\partial M(h_x)}{\partial x} = \frac{\partial M(h_x)}{\partial h_x} h_{xx} = M'(0)h_{xx}, \quad (2.1.1)$$

and Eq. (1.1.1) takes the form:

$$\begin{aligned} h_t = & BM'(0)h_{xx} (1 + h_x^2)^{-1/2} \mu_x + B(M(0) + M'(0)h_x) \left[(1 + h_x^2)^{-1/2} \mu_x \right]_x \\ & + AM'(0)h_{xx} (1 + h_x^2)^{-1/2} + A(M(0) + M'(0)h_x) \left[(1 + h_x^2)^{-1/2} \right]_x. \end{aligned} \quad (2.1.2)$$

In order to calculate μ_x in Eq. (2.1.2), we first find $\gamma_{\theta\theta}, \gamma_h, \gamma_{h\theta}$ using Eq. (1.1.4). Next, we substitute these functions in Eq. (1.1.3), use the trigonometric identities

$$\cos 4\theta = 8 [\cos^4 \theta - \cos^2 \theta] + 1, \quad \sin 4\theta = 4 \sin \theta \cos \theta [2 \cos^2 \theta - 1]$$

(where $\cos \theta = (1 + h_x^2)^{-1/2}$, $\sin \theta = h_x (1 + h_x^2)^{-1/2}$) and finally obtain $\mu(x, t)$ in terms of h_x, h_x^2, h_{xx} , etc.

Then, we plug $\mu(x, t)$ into Eq. (2.1.2) and perform the remaining differentiations. This results in the fourth-order, nonlinear PDE for $h(x, t)$ in the form:

$$h_t = F(h_x, h_{xx}, h_{xxx}, h_{xxxx}, h_{xx}h_x^2, \dots). \quad (2.1.3)$$

We wish to obtain a simpler PDE, whose solutions have the characteristics similar to Eq. (2.1.3). To this end, anticipating a long-wave instability of Eq. (2.1.3), we introduce the stretched coordinate X through the transformation $x = X/\rho$, where $\rho \ll 1$. It follows that

$$h_x = \rho h_X, h_{xx} = \rho^2 h_{XX}, h_{xxx} = \rho^3 h_{XXX}, h_{xxxx} = \rho^4 h_{XXXX}.$$

We introduce these substitutions into the RHS of Eq. (2.1.3) and then set $\rho^i = 0, i \geq 5$. Lastly, we collect the coefficients of $\rho^j, j = 0, 1, 2, 3, 4$. and obtain the simplified nonlinear PDE:

$$h_t = (\alpha_1 + \beta_1 e^{-h}) h_{xxxx} + (\alpha_2 + \beta_2 e^{-h}) h_{xx} + (\alpha_5 + \beta_5 e^{-h}) h_{xx} h_x^2, \quad (2.1.4)$$

where

$$\begin{aligned}
\alpha_1 &= -BM(0) (1 - 15\epsilon_\gamma), \\
\alpha_2 &= AM'(0), \\
\alpha_5 &= -1.5AM'(0), \\
\beta_1 &= -BM(0) (15\epsilon_\gamma + G - 1), \\
\beta_2 &= BM(0) (-\epsilon_\gamma + G - 1), \\
\beta_5 &= -BM(0) (80\epsilon_\gamma - 29\epsilon_\gamma + 5G - 5).
\end{aligned} \tag{2.1.5}$$

Notice that we keep the original notation for the independent variable, x (not X).

We will solve Eq. (2.1.4) in Chapter 3 using Method of Lines, subject to the following IC and BCs:

$$h(x, 0) = 1 + \delta \cos(k_{\max}x), \quad u(0, t) = u\left(\frac{2\pi}{k_{\max}}, t\right), \quad u'(0, t) = u'\left(\frac{2\pi}{k_{\max}}, t\right). \tag{2.1.6}$$

To linearize, first take $h = h_0 + \xi(x, t)$ (where h_0 is the film thickness as deposited on the substrate and $\xi(x, t)$ is the *small* perturbation) and substitute with all derivatives of $h = h_0 + \xi(x, t)$ into Eq. (2.1.4). Linearizing, we omit all nonlinear contributions, such as the terms proportional to $\xi^2, \xi\xi_x$, etc.

Finally, in the resulting *linear* PDE for ξ we assume $\xi(x, t) = e^{\omega(k)t}e^{ikx}$, where ω is the growth rate and k is the wavenumber and derive the function $\omega(k)$:

$$\omega(k) = -\left[BM(0)(G - 1 - \epsilon_\gamma)e^{-h_0} + AM'(0)\right]k^2 - BM(0)\left[(G - 1 + 15\epsilon_\gamma)e^{-h_0} + 1 - 15\epsilon_\gamma\right]k^4. \tag{2.1.7}$$

Physical Parameters	Typical Values	Range	Physical Meaning
B	8	Fixed	Nominal Atomic Diffusivity
$M(0)$	1	Fixed	Atomic Mobility on the Horizontal Surface
h_0	1	$0 \leq h_0 \leq 20$	Initial Height of the Film (same for $\forall x$)
A	72	$10 \leq A \leq 10000$	Strength of the Electric Field
G	2	$0 \leq G \leq 3.5$	Ratio of Substrate to Film Surface Energy
$M'(0)$	-3	$-10 \leq M'(0) \leq 0$	Derivative of Atomic Mobility
ϵ_γ	0	$0 \leq \epsilon_\gamma \leq \frac{1}{30}$	Anisotropy Strength

TABLE 2.1.1. The physical parameter values used in our study.

2.2. Analysis of The Equation (2.1.7)

The goal of the analysis in this section is the determination of how the physical parameters of the problem affect surface stability or instability.

As was explained in section 1.1, if $\omega(k) < 0 \forall k$, then the surface is stable with respect to perturbations of any wavenumber (Figure 1.1.2 (b)). When this condition does not hold, then the surface is a long-wave unstable (Figure 1.1.2 (a)). The degree of instability is measured by the width of the domain under the dispersion curve $\omega(k)$. That is, the larger k_c is, the greater is the instability.

The short-wavelength cut-off wavenumber k_c can be determined by solving $\omega(k)=0$ for k , retaining the positive solution:

$$k_c = \sqrt{\frac{Ae^{h_0}M'(0) + BM(0)[G - 1 - \epsilon_\gamma]}{BM(0)[1 - G - 15\epsilon_\gamma + (15\epsilon_\gamma - 1)e^{h_0}]}}. \quad (2.2.1)$$

The film stability *decreases* with increasing h_0 , and this trend saturates around $h_0 = 5.12$ nm (Figure 2.2.1 (a)). This is because the film wets the substrate and thus the attractive, cohesive force between the film surface molecules and the substrate molecules is stronger for thinner films (smaller h_0). Increasing the electric

field strength A also makes the film less stable (the direction of the field and the sign of $M'(0)$ are chosen such that their combination promotes instability, as in the experiments) but increasing G makes it more stable, since the substrate energy provides stabilizing effect (Figures 2.2.1 (b)-(c)). The stability of the film *decreases* with increasing $|M'(0)|$ and ϵ_γ (Figures 2.2.1 (d)-(e)).

The most dangerous perturbation (one that destabilizes the surface faster) is one with the max growth rate ω_{max} . To find its wavenumber, we solve $\frac{\partial\omega(k)}{\partial k} = 0$ for k and retain the positive solution k_{max} . This solution is $k_{max} = \frac{k_c}{\sqrt{2}}$, where k_c is given in Eq. 2.2.1. Then, $\omega_{max} = \omega(k_{max})$:

$$\omega_{max} = \frac{1}{4} \frac{[Ae^{h_0}M'(0) + BM(0)(G-1) - BM(0)\epsilon_\gamma]^2}{BM(0)e^{h_0} [G-1 + 15\epsilon_\gamma + (1-15\epsilon_\gamma)e^{h_0}]} \quad (2.2.2)$$

The wavelength of the most dangerous perturbation is $\lambda_{max} = \frac{2\pi}{k_{max}}$.

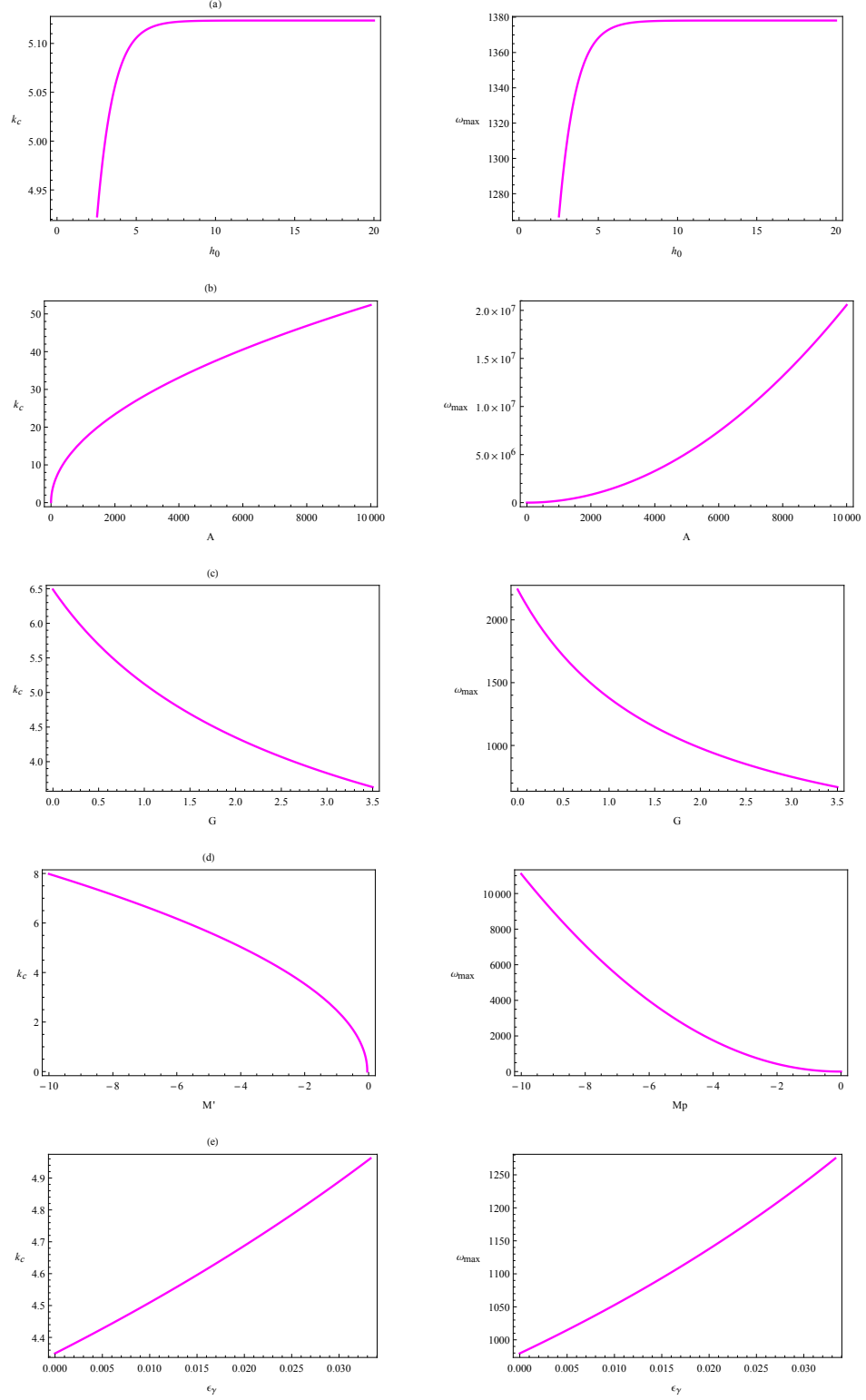


FIGURE 2.2.1. Characterization of the film linear stability. (a) k_c vs. h_0 , $A = 70$, $G = 2$, $M' = -3$, $\epsilon_\gamma = 0$. (b) k_c vs. A , $h_0 = 1$, $G = 2$, $M' = -3$, $\epsilon_\gamma = 0$. (c) k_c vs. G , $A = 70$, $h_0 = 1$, $M' = -3$, $\epsilon_\gamma = 0$. (d) k_c vs. M' , $A = 70$, $G = 2$, $h_0 = 1$, $\epsilon_\gamma = 0$. (e) k_c vs. ϵ_γ , $A = 70$, $G = 2$, $h_0 = 1$, $M' = -3$ where k_c is the instability cut-off wavenumber. ω_{max} in the right panel is plotted for the same parameters values.

CHAPTER 3

METHOD OF LINES

3.1. Introduction to Method of Lines

The numerical method of lines is a technique for solving partial differential equations by discretizing in all but one dimension, and then integrating the semi-discrete problem as a system of Ordinary Differential Equations (ODEs). A significant advantage of the method is that it allows us to use the sophisticated general-purpose methods and software that have been developed for numerically integrating ODEs. For the PDEs to which the method of lines is applicable, the method typically proves to be quite efficient.

It is necessary that the PDE problem be well posed as an initial value (Cauchy) problem in at least one dimension, since the ODE integrators used are initial value problem solvers. This rules out purely elliptic equations such as Laplace's equation, but leaves a large class of evolution equations that can be solved. This class encompasses the equations with smooth solutions in time-space, such as parabolic PDEs. Sophisticated adaptive Method of Lines (MOL) methods were developed for some hyperbolic equations [9], [11].

As a basic illustrative example of a MOL solution [6], we consider the heat equation

$$u_t = \alpha u_{xx}, \quad \alpha > 0, \tag{3.1.1}$$

where

- u dependent variable (dependent on x and t),
- t independent variable representing time,
- x independent variable representing one dimension of three-dimensional space,
- α real positive constant.

Before we consider a solution to Eq. (3.1.1), we must specify some auxiliary conditions to complete the statement of the PDE problem. The number of required auxiliary conditions is determined by the highest order derivative in each independent variable. Since Eq. (3.1.1) is *first order* in t and *second order* in x , it requires *one* auxiliary condition in t and *two* auxiliary conditions in x . t is termed an *initial value variable* and therefore requires one initial condition (IC). x is termed a *boundary value variable* and therefore requires two boundary conditions (BCs).

As examples of auxiliary conditions for Eq. (3.1.1), an IC could be

$$u(x, t = 0) = u_0(x), \quad (3.1.2)$$

where u_0 is a given function of x , and two BCs could be

$$u(x = x_0, t) = u_b \quad (3.1.3)$$

and

$$\frac{\partial u(x = x_f, t)}{\partial x} = 0, \quad (3.1.4)$$

where u_b is a given boundary (constant) value of u for all t .

Three types of BCs are:

- If the dependent variable is specified, as in BC (3.1.3), the BC is termed Dirichlet.
- If the derivative of the dependent variable is specified, as in BC (3.1.4), the BC is termed Neumann.
- If both the dependent variable and its derivative appear in the BC, it is termed BCs of the third type or Robin BCs or Mixed BCs.

In MOL, a PDE for two variables is first discretized in one variable (x or t).

In the case of Eq. (3.1.1), it is convenient to discretize the spatial variable x .

First, we need an approximation at $x = x_i$ of the second derivative u_{xx} . A commonly used second order, central approximation is

$$u_{xx} \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}, \quad (3.1.5)$$

where i is the grid point index, $\Delta x = x_{i+1} - x_i = (x_f - x_0)/N$ is the size of a subinterval, N is the number of subintervals, and $u_i = u(x_i)$. Substituting Eq. (3.1.5) into Eq. (3.1.1) gives a system of approximating ODEs

$$u'_i = \alpha \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}, \quad i = 0, 1, \dots, N. \quad (3.1.6)$$

Eq. (3.1.6) are then integrated subject to IC (3.1.2) and BCs (3.1.3) and (3.1.4). This integration in t can be done by the explicit Euler method, the implicit Euler method, or any other higher order integration method for initial value ODEs.

Before proceeding with the integration of eqs. (3.1.6), we must include BCs (3.1.3) and (3.1.4). The Dirichlet BC at $x = x_0$, Eq. (3.1.3), is merely

$$u_0 = u_b, \quad (3.1.7)$$

and therefore the ODE of eqs. (3.1.6) for $i=0$ is not required and the ODE for $i=1$ becomes

$$u'_1 = \alpha \frac{u_2 - 2u_1 + u_b}{\Delta x^2}. \quad (3.1.8)$$

At $i=N$, we have equations

$$u'_N = \alpha \frac{u_{N+1} - 2u_N + u_{N-1}}{\Delta x^2}. \quad (3.1.9)$$

Note that u_{N+1} is outside the grid in x . But we must assign a value to u_{N+1} if Eq. (3.1.9) is to be integrated. Since this requirement occurs at the boundary point $i=N$, we obtain this value by approximating BC (3.1.4) using the centered FD approximation, that is

$$u_x \approx \frac{u_{i+1} - u_{i-1}}{2\Delta x}, \quad (3.1.10)$$

and we obtain

$$u_x \approx \frac{u_{N+1} - u_{N-1}}{2\Delta x} = 0 \quad (3.1.11)$$

or

$$u_{N+1} = u_{N-1}. \quad (3.1.12)$$

3.2. Examples

EXAMPLE 3.2.1. Consider the Heat equation with IC, BCs (Dirichlet) and α as below:

$$u_t = \alpha u_{xx}, \quad (3.2.1)$$

$$u(x, 0) = f(x), \quad u(0, t) = u_b, \quad u(L, t) = u_f. \quad (3.2.2)$$

Recall Eq. (3.1.6) that reads:

$$u'_i = \alpha \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}, \quad i = 0, 1, \dots, N \quad \text{where} \quad \Delta x = \frac{L}{N}.$$

For example, for $N=6$ we get the system of ODEs below. Note that we need not evaluate u at $x = x_0 = 0$ and $x = x_6 = L$ since u_0 and u_N are given as boundary conditions. Thus, we can eliminate the first and the last equation. Also, $u_0 = u_b$ and $u_6 = u_f$ are boundary conditions. Therefore, we have 5 equations and 5 unknowns where u_b and u_f are (constant) BCs.

$$\begin{aligned}
u'_0 &= \alpha \frac{u_{-1} - 2u_0 + u_1}{\Delta x^2} \\
u'_1 &= \alpha \frac{u_0 - 2u_1 + u_2}{\Delta x^2} & u'_1 &= \alpha \frac{u_0 - 2u_1 + u_2}{\Delta x^2} & u'_1 &= \alpha \frac{u_b - 2u_1 + u_2}{\Delta x^2} \\
u'_2 &= \alpha \frac{u_1 - 2u_2 + u_3}{\Delta x^2} & u'_2 &= \alpha \frac{u_1 - 2u_2 + u_3}{\Delta x^2} & u'_2 &= \alpha \frac{u_1 - 2u_2 + u_3}{\Delta x^2} \\
u'_3 &= \alpha \frac{u_2 - 2u_3 + u_4}{\Delta x^2} & \implies u'_3 &= \alpha \frac{u_2 - 2u_3 + u_4}{\Delta x^2} & \implies u'_3 &= \alpha \frac{u_2 - 2u_3 + u_4}{\Delta x^2} \\
u'_4 &= \alpha \frac{u_3 - 2u_4 + u_5}{\Delta x^2} & u'_4 &= \alpha \frac{u_3 - 2u_4 + u_5}{\Delta x^2} & u'_4 &= \alpha \frac{u_3 - 2u_4 + u_5}{\Delta x^2} \\
u'_5 &= \alpha \frac{u_4 - 2u_5 + u_6}{\Delta x^2} & u'_5 &= \alpha \frac{u_4 - 2u_5 + u_6}{\Delta x^2} & u'_5 &= \alpha \frac{u_4 - 2u_5 + u_f}{\Delta x^2} \\
u'_6 &= \alpha \frac{u_5 - 2u_6 + u_7}{\Delta x^2}
\end{aligned} \tag{3.2.3}$$

The last ODE system in Eq. (3.2.3) can be written as

$$\vec{u}' = \frac{\alpha}{\Delta x^2} A \vec{u} + \frac{\alpha}{\Delta x^2} \vec{b}, \tag{3.2.4}$$

where

$$A = \begin{pmatrix} -2 & 1 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 1 & -2 \end{pmatrix}, \quad \vec{u}' = \begin{pmatrix} u'_1 \\ u'_2 \\ u'_3 \\ u'_4 \\ u'_5 \end{pmatrix}, \quad \vec{u} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{pmatrix} \text{ and } \vec{b} = \begin{pmatrix} u_b \\ 0 \\ 0 \\ 0 \\ u_f \end{pmatrix}.$$

Exact solution of Eq. (3.2.1) and Eq. (3.2.2) is given in [5] page 125 as:

$$u(x, t) = \sum_{n=1}^{\infty} C_n \sin\left(\frac{\pi n x}{L}\right) e^{-\frac{\pi^2 \alpha n^2 t}{L^2}}, \quad (3.2.5)$$

where C_n is

$$C_n = \frac{2}{L} \int_0^L f(x) \sin\left(\frac{\pi n x}{L}\right) dx. \quad (3.2.6)$$

For $L = 1$, $\alpha = 1/20$ and $u(x, 0) = f(x) = \sin\left(\frac{\pi x}{L}\right)$

$$C_n = \begin{cases} 1, & n = 1 \\ 0, & n \geq 2 \end{cases}$$

Therefore, the exact solution of Eq. (3.2.1) and Eq. (3.2.2) is

$$u(x, t) = \sin(\pi x) e^{-\frac{\pi^2 t}{20}}. \quad (3.2.7)$$

We used the *NDSolve* command in *Mathematica* to solve ODE system of (3.2.4) for $L=1$, $\alpha = 1/20$, $u_b = 0$, $u_f = 0$, $N=100$ and compared with exact solution. See Figure 3.2.1 for comparison and Figure 3.2.2 for error. For the basic *Mathematica* code see appendix.

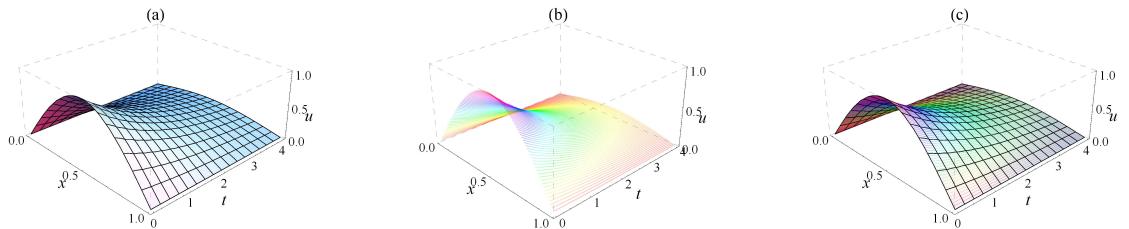


FIGURE 3.2.1. This figure shows comparison of solutions for Example 3.2.1. (a) Analytical solution, (b) MOL solution, $N=100$ (c) Analytical and MOL solution match, because the two solutions overlap.

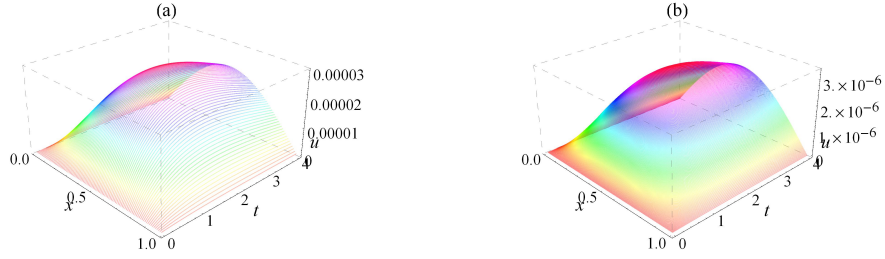


FIGURE 3.2.2. This figure shows the entire curve difference (absolute error) of Example 3.2.1. Comparison of the analytical solution with MOL solution. (a) $N=100$, (b) $N=300$.

EXAMPLE 3.2.2. Consider the PDE with IC and BCs (Periodic) as below:

$$u_t = -Bu_{xxxx}, \quad (3.2.8)$$

$$u(x, 0) = 1 + B \cos(kx), \quad u(0, t) = u(2\pi, t), \quad u'(0, t) = u'(2\pi, t). \quad (3.2.9)$$

Eq. (3.2.8) is approximated as:

$$u'_i = -B \frac{u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2}}{\Delta x^4}, \quad i = 0, 1, \dots, N. \quad (3.2.10)$$

For example, for $N=6$ we get the system of ODEs below. Note that some points are outside of our grid. These points are called *ghost points*. First and last equations have two *ghost points* and second and fifth equations have one *ghost point* each. Since BCs are periodic, we replace them with values that are in the grid. See Figure

3.2.3.

$$\begin{aligned}
 u'_0 &= -B \frac{u_{-2} - 4u_{-1} + 6u_0 - 4u_1 + u_2}{\Delta x^4} & u'_0 &= -B \frac{6u_0 - 4u_1 + u_2 + u_4 - 4u_5}{\Delta x^4} \\
 u'_1 &= -B \frac{u_{-1} - 4u_0 + 6u_1 - 4u_2 + u_3}{\Delta x^4} & u'_1 &= -B \frac{-4u_0 + 6u_1 - 4u_2 + u_3 + u_5}{\Delta x^4} \\
 u'_2 &= -B \frac{u_0 - 4u_1 + 6u_2 - 4u_3 + u_4}{\Delta x^4} & u'_2 &= -B \frac{u_0 - 4u_1 + 6u_2 - 4u_3 + u_4}{\Delta x^4} \\
 u'_3 &= -B \frac{u_1 - 4u_2 + 6u_3 - 4u_4 + u_5}{\Delta x^4} & \implies u'_3 &= -B \frac{u_1 - 4u_2 + 6u_3 - 4u_4 + u_5}{\Delta x^4} \\
 u'_4 &= -B \frac{u_2 - 4u_3 + 6u_4 - 4u_5 + u_6}{\Delta x^4} & u'_4 &= -B \frac{u_2 - 4u_3 + 6u_4 - 4u_5 + u_6}{\Delta x^4} \\
 u'_5 &= -B \frac{u_3 - 4u_4 + 6u_5 - 4u_6 + u_7}{\Delta x^4} & u'_5 &= -B \frac{u_1 + u_3 - 4u_4 + 6u_5 - 4u_6}{\Delta x^4} \\
 u'_6 &= -B \frac{u_4 - 4u_5 + 6u_6 - 4u_7 + u_8}{\Delta x^4} & u'_6 &= -B \frac{-4u_1 + u_2 + u_4 - 4u_5 + 6u_6}{\Delta x^4}
 \end{aligned} \tag{3.2.11}$$

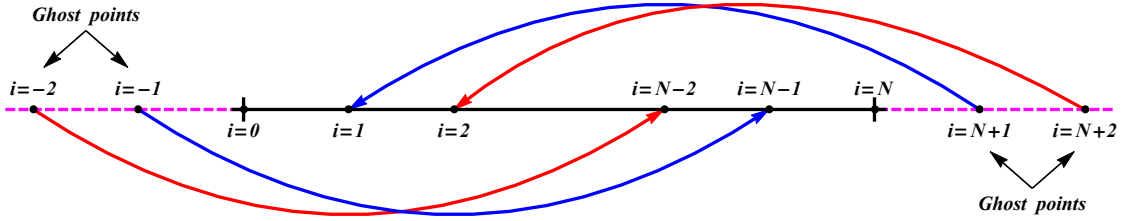


FIGURE 3.2.3. Periodic BCs. $u_{N+1} = u_1$, $u_{N+2} = u_2$, $u_{-1} = u_{N-1}$, $u_{-2} = u_N - 2$.

The last ODE system in Eq. (3.2.11) can be written as

$$\vec{u}' = \frac{-B}{\Delta x^4} A \vec{u}, \tag{3.2.12}$$

$$\text{where } A = \begin{pmatrix} 6 & -4 & 1 & 0 & 1 & -4 & 0 \\ -4 & 6 & -4 & 1 & 0 & 1 & 0 \\ 1 & -4 & 6 & -4 & 1 & 0 & 0 \\ 0 & 1 & -4 & 6 & -4 & 1 & 0 \\ 0 & 0 & 1 & -4 & 6 & -4 & 1 \\ 0 & 1 & 0 & 1 & -4 & 6 & -4 \\ 0 & -4 & 1 & 0 & 1 & -4 & 6 \end{pmatrix}, \quad \vec{u}' = \begin{pmatrix} u'_0 \\ u'_1 \\ u'_2 \\ u'_3 \\ u'_4 \\ u'_5 \\ u'_6 \end{pmatrix} \text{ and } \vec{u} = \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{pmatrix}.$$

Exact solution of Eq. (3.2.8) and Eq. (3.2.9) is given in [8] as:

$$u(x, t) = 1 + Be^{-k^4 t} \cos(kx), \quad \forall k > 0 \quad (3.2.13)$$

where k is the wavenumber.

We used the *NDSolve* command in *Mathematica* to solve ODE system of (3.2.12) for $k=1$, $B=1$, $N=100$ and compared with exact solution. See Figure 3.2.4 for comparison and Figure 3.2.5 for error. For the basic *Mathematica* code see appendix.

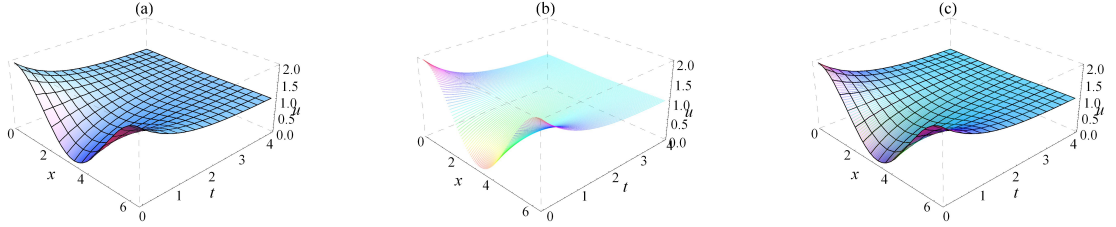


FIGURE 3.2.4. This figure shows comparison of solutions for Example 3.2.2. (a) Analytical solution, $k=1$, $B=1$ (b) MOL solution, $N=100$ (c) Analytical and MOL solutions match, because the two solutions overlap.

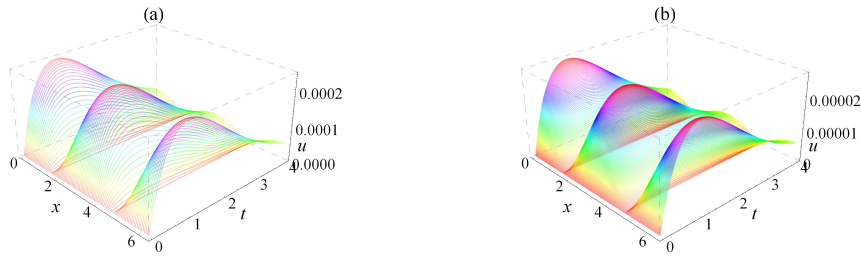


FIGURE 3.2.5. This figure shows the entire curve difference (absolute error) for Example 3.2.2. Comparison analytical with MOL. (a) $N=100$, (b) $N=300$.

3.3. Analysis of The Simplified Nonlinear PDE

In this chapter PDE (2.1.4) is solved numerically using MOL, subject to periodic boundary conditions. Such BCs are used since the goal of the numerical study is to model evolution of a finite section of a periodic, laterally infinite surface, as in Figure 1.0.1 (left panel). Evolution of a perturbation having the wavelength $\lambda_{max} = 2\pi/k_{max}$ from the linear stability analysis will be computed on the interval $0 \leq x \leq \lambda_{max}$, until the steady-state solution emerges. Properties of this solution (amplitude, etc.) are studied as a function of the physical parameters: A , G , $M'(0)$, ϵ_γ .

$$h_t = (\alpha_1 + \beta_1 e^{-h}) h_{xxxx} + (\alpha_2 + \beta_2 e^{-h}) h_{xx} + (\alpha_5 + \beta_5 e^{-h}) h_{xx} h_x^2, \quad (3.3.1)$$

$$h(x, 0) = 1 + \delta \cos(k_{\max}x), \quad u(0, t) = u\left(\frac{2\pi}{k_{\max}}, t\right), \quad u'(0, t) = u'\left(\frac{2\pi}{k_{\max}}, t\right). \quad (3.3.2)$$

where $\delta = 10^{-2}$ and

$$\begin{aligned} \alpha_1 &= -\text{BM}(0) (1 - 15\epsilon_\gamma), \\ \alpha_2 &= \text{AM}'(0), \\ \alpha_5 &= -1.5\text{AM}'(0), \\ \beta_1 &= -\text{BM}(0) (15\epsilon_\gamma + G - 1), \\ \beta_2 &= \text{BM}(0) (-\epsilon_\gamma + G - 1), \\ \beta_5 &= -\text{BM}(0) (80\epsilon_\gamma - 29\epsilon_\gamma + 5G - 5). \end{aligned} \quad (3.3.3)$$

Since $x \in [0, \frac{2\pi}{k_{\max}}]$, where $k_{\max} = \sqrt{\frac{\alpha_2}{2\alpha_1}}$ and as such it is a function of the physical parameters, every time we change a value of a parameter, the length of the computational domain changes. This is inconvenient, thus we transform it to a fixed interval $x \in [0, 2\pi]$, as follows.

Let

$$\begin{aligned} \xi &= k_{\max}x \implies \frac{d}{dx} = k_{\max} \frac{d}{d\xi}, \\ T &= k_{\max}^4 t \implies \frac{d}{dt} = k_{\max}^4 \frac{d}{dT}. \end{aligned}$$

By applying these transformations to Eq. (3.3.1), we get

$$k_{\max}^4 h_T = (\alpha_1 + \beta_1 e^{-h}) k_{\max}^4 h_{\xi\xi\xi\xi} + (\alpha_2 + \beta_2 e^{-h}) k_{\max}^2 h_{\xi\xi} + (\alpha_5 + \beta_5 e^{-h}) k_{\max}^2 h_{\xi\xi} k_{\max}^2 h_{\xi}^2$$

or

$$h_T = (\alpha_1 + \beta_1 e^{-h}) h_{\xi\xi\xi\xi} + (\alpha_2 + \beta_2 e^{-h}) k_{\max}^{-2} h_{\xi\xi} + (\alpha_5 + \beta_5 e^{-h}) h_{\xi\xi} h_{\xi}^2.$$

Going back to the original notation, that is replacing T by t and ξ by x we obtain a simplified non-linear PDE for the *height function* $h(x, t)$ of the thin film, with periodic BCs:

$$h_t = (\alpha_1 + \beta_1 e^{-h}) h_{xxxx} + (\alpha_2 + \beta_2 e^{-h}) k_{\max}^{-2} h_{xx} + (\alpha_5 + \beta_5 e^{-h}) h_{xx} h_x^2, \quad (3.3.4)$$

$$h(x, 0) = 1 + \delta \cos(x), \quad u(0, t) = u(2\pi, t), \quad u'(0, t) = u'(2\pi, t). \quad (3.3.5)$$

Recall that $k_{\max} = \sqrt{\frac{\alpha_2}{2\alpha_1}}$. Below are the first, second and fourth order centered finite difference approximations of the spatial derivatives:

$$\begin{aligned} h_x &\approx \frac{h_{i+1} - h_{i-1}}{2\Delta x}, \\ h_{xx} &\approx \frac{h_{i+1} - 2h_i + h_{i-1}}{\Delta x^2}, \\ h_{xxxx} &\approx \frac{h_{i+2} - 4h_{i+1} + 6h_i - 4h_{i-1} + h_{i-2}}{\Delta x^4}. \end{aligned} \quad (3.3.6)$$

Eq. (3.3.6) is substituted into Eq. (3.3.4), and then the system of ODEs is solved using the *NDSolve* command in *Mathematica* for $\delta = 10^{-2}$, typical parameter values and $N=100$. See Figure 3.3.1 for comparison and Figure 3.3.2 for error. For the basic *Mathematica* code see appendix.

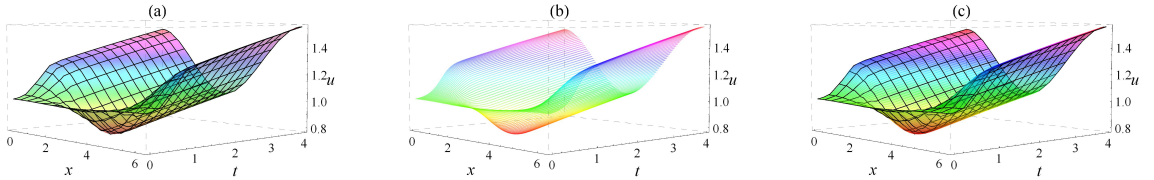


FIGURE 3.3.1. This figure shows comparison of solutions. (a) *NDSolve* command solution of *Mathematica*, (b) MOL solution of Eq. (3.3.4) and Eq. (3.3.5), for $\delta = 10^{-2}$, typical parameter values and $N=100$ (c) *NDSolve* and MOL solution match, because the two solutions overlap.

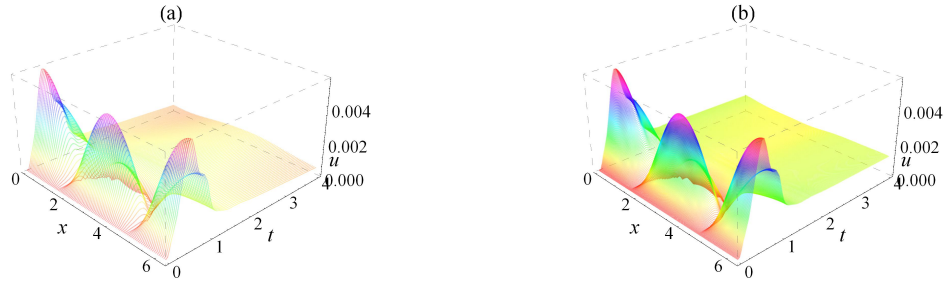


FIGURE 3.3.2. This figure shows the entire curve difference (absolute error) for Eq. (3.3.4) and Eq. (3.3.5). Comparison *NDSolve* with MOL, for $\delta = 10^{-2}$, typical parameter values (a) $N=100$, (b) $N=300$.

We plotted the first 30 profile shapes for $A = 1000$ and the typical values of the other parameters to see how $h(x,t)$ evolves in time. As it can be seen (Figure (3.3.3)) the amplitude *increases* with time and profile lines approach each other, meaning that the solution approaches the steady state solution.

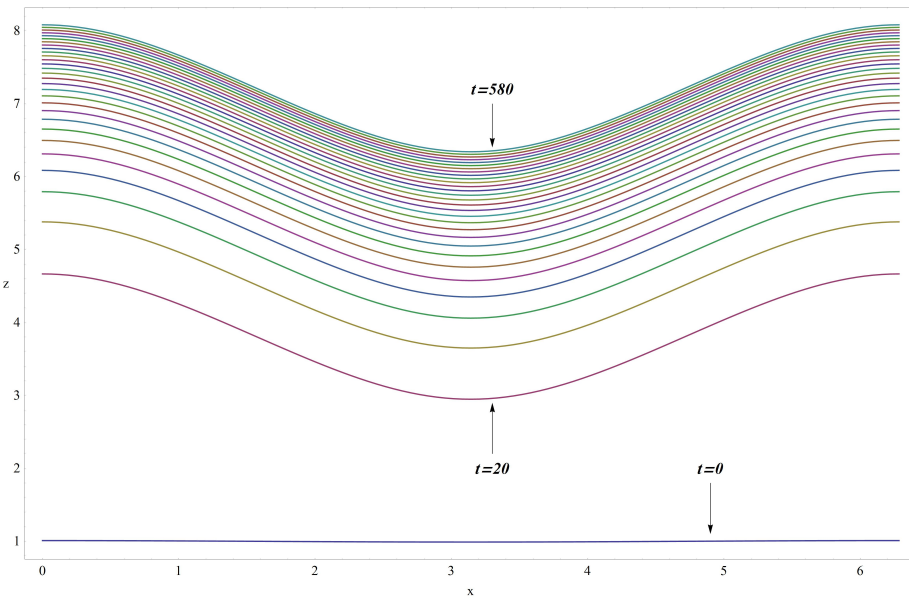


FIGURE 3.3.3. Evolution of the graph $h(x,t)$ in time.

We plotted the last profile curve and cosine curve on the same screen. Our result shows that they differ, meaning that the last profile curve is not cosine curve anymore (Figure (3.3.4)).

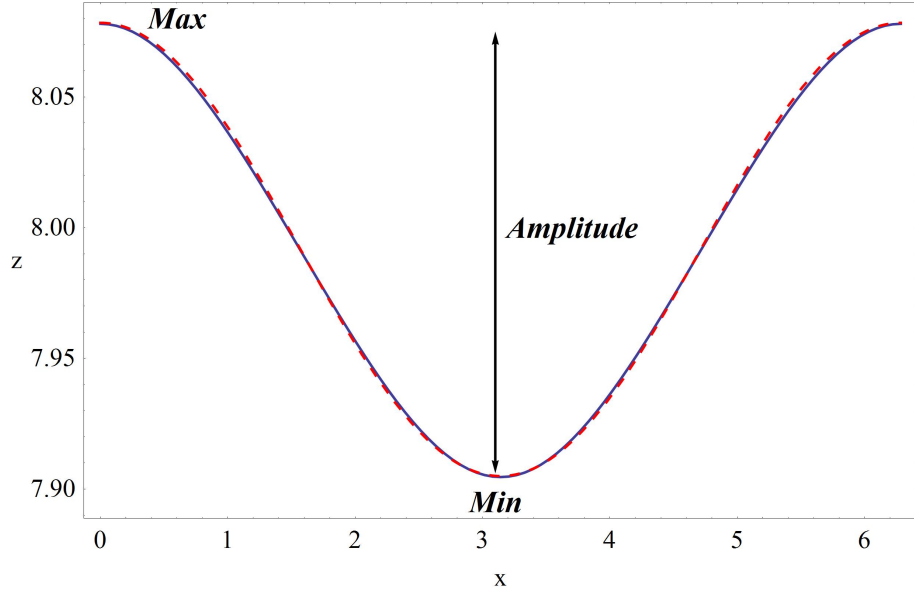


FIGURE 3.3.4. Last profile curve (solid line) vs cosine curve (dashed line).

We analyzed stability of the surface by solving Eq. (3.3.4) and Eq. (3.3.2) for the typical values of all parameters, except one parameter, and measured amplitude for ten different values of non-fixed parameter (Figure 3.3.5 (a)).

Our results show that the amplitude of the last profile curve *decreases* with *increasing* A and $|M'(0)|$ but never reaches $\delta = 10^{-2}$ which is the initial amplitude (Figures 3.3.5 (a) and (c)). That means surface is *unstable* with increasing parameter values of A and $|M'(0)|$. The amplitude of the last profile curve *does not change* with *increasing* G and *slightly changes* (decreases) with *increasing* ϵ_γ , but amplitudes are still *greater* than initial amplitude $\delta = 10^{-2}$ (Figures 3.3.5 (b) and (d)). Thus, the surface is *unstable* with increasing parameter values of G and ϵ_γ .

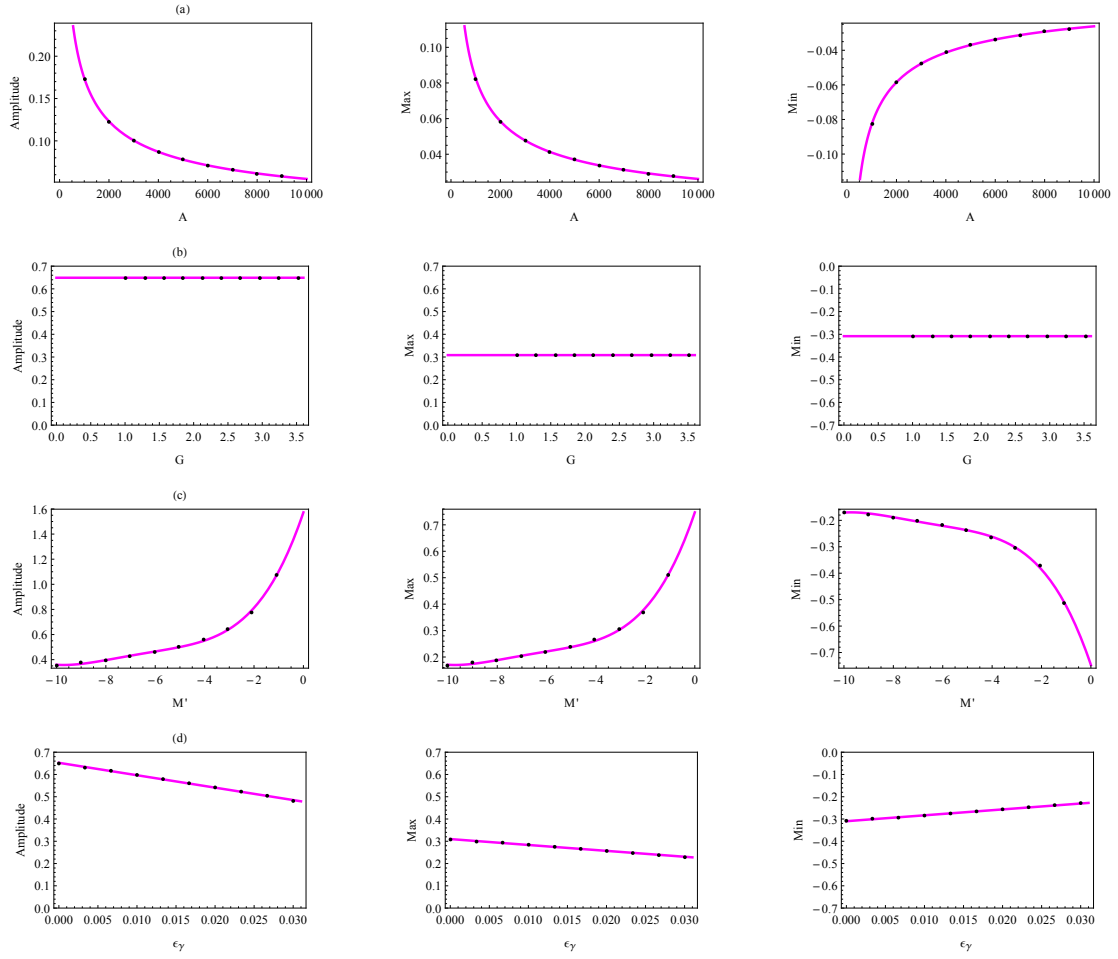


FIGURE 3.3.5. **Left:** Amplitude vs physical parameters. **Middle:** Max value vs physical parameters. **Right:** Min value vs physical parameters.

CHAPTER 4

CONCLUSION AND FUTURE WORK

In Introduction, we gave a brief definition and showed some applications of surface electromigration.

In Chapter 2, we simplified a fourth-order nonlinear PDE model of surface electromigration in the *horizontal* electric field, and we used a small perturbation around the initial film thickness to linearize the model to get the growth rate function. We then analyzed the growth rate function for different physical parameters to understand how surface stability is affected by physical parameters. Our results show that the film stability *decreases* with increasing h_0 and the electric field strength A , but the film stability *increases* with increasing G . The stability of the film *decreases* with increasing $|M'(0)|$ and ϵ_γ .

In Chapter 3, we used Method of Lines (MOL) to solve the simplified nonlinear PDE numerically and discussed how the physical parameters affect the surface stability. Our results show that the amplitude of the last profile curve *decreases* with *increasing* A and $|M'(0)|$ but never reaches $\delta = 10^{-2}$, which is the initial amplitude. That means the surface is *unstable* with increasing parameter values of A and $|M'(0)|$. The amplitude of the last profile curve *does not change* with *increasing* G and *slightly changes* (decreases) with *increasing* ϵ_γ , but the amplitudes are still *greater* than the initial amplitude $\delta = 10^{-2}$. Thus, the surface is *unstable* with increasing parameter values of G and ϵ_γ .

Our numerical studies are on the domain $x \in [0, 2\pi]$, and we used cosine curve with the small initial amplitude $\delta = 10^{-2}$ to perturb the initial height. Future

work would be to extend the domain to several wavelengths and for initial condition use a curve with a small random amplitude. In such a setup, coarsening of the initial perturbation can be studied and the predictions can be made about the pattern formation on the surface.

APPENDIX

```

γ[h_, θ_] := 1 + eγ Cos[4 θ] + (G - 1 - eγ Cos[4 θ]) e-h
μ = (γ[h, θ] + D[γ[h, θ], θ, θ]) (-h''[x] (1 + h'[x]^2)-3/2) +
      (D[γ[h, θ], h] - m * h'[x] * D[γ[h, θ], h, θ]) (1 + h'[x]^2)-3/2 /.
      {e-h -> e-h[x], Cos[4 θ] -> 8 ((1 + h'[x]^2)-3/2)4 - ((1 + h'[x]^2)-3/2)2} + 1,
      Sin[4 θ] -> 4 h'[x] (1 + h'[x]^2)-3/2 (1 + h'[x]^2)-3/2 (2 ((1 + h'[x]^2)-3/2)2 - 1)};
μ = Simplify[μ];
w1 = Simplify[D[μ, x]];
rhs1 = Simplify[B * Mp * h''[x] (1 + h'[x]^2)-3/2 * w1];
w2 = Simplify[D[(1 + h'[x]^2)-3/2] * D[μ, x], x];
rhs2 = Simplify[B * (M + Mp * h'[x]) * w2];
rhs3 = Simplify[A * Mp * h''[x] * ((1 + h'[x]^2)-3/2)]];
w3 = D[(1 + h'[x]^2)-3/2], x];
rhs4 = Simplify[A * (M + Mp * h'[x]) * w3];
rhs = Simplify[rhs1 + rhs2 + rhs3 + rhs4];

rhs = rhs /. {h'[x] -> ρ * h'[x], h''[x] -> ρ2 * h''[x], h'''[x] -> ρ3 * h'''[x], h''''[x] -> ρ4 * h''''[x]};
t1 =  $\frac{-A \rho^3 h'[x] (M + Mp \rho h'[x]) h''[x]}{(1 + \rho^2 h'[x]^2)^{3/2}}$ ; t2 =  $\frac{A Mp \rho^2 h''[x]}{\sqrt{1 + \rho^2 h'[x]^2}}$ ; t3 =  $\frac{1}{(1 + \rho^2 h'[x]^2)^5}$ ;
t4 =  $\frac{1}{(1 + \rho^2 h'[x]^2)^6}$ ;
s1 = Series[ $\frac{-A \rho^3 h'[x] (M + Mp \rho h'[x]) h''[x]}{(1 + \rho^2 h'[x]^2)^{3/2}}$ , {ρ, 0, 4}] // Normal;
s2 = Series[ $\frac{A Mp \rho^2 h''[x]}{\sqrt{1 + \rho^2 h'[x]^2}}$ , {ρ, 0, 4}] // Normal; s3 = Series[ $\frac{1}{(1 + \rho^2 h'[x]^2)^5}$ , {ρ, 0, 4}] // Normal;
s4 = Series[ $\frac{1}{(1 + \rho^2 h'[x]^2)^6}$ , {ρ, 0, 4}] // Normal;
rhs = Expand[rhs /. {t1 -> s1, t2 -> s2, t3 -> s3, t4 -> s4}];
Collect[rhs, Table[ρi, {i, 2, 17}]];
rhs = rhs /. Table[ρi -> 0, {i, 5, 17}];
rhs = rhs /. ρ -> 1;
Collect[rhs, {h''[x], h''[x] h'[x]^2, h(4)[x]}];

```

FIGURE 4.0.1. Basic *Mathematica* code for the nonlinear PDE, part 1.

```

rhs = rhs /. { $\rho \rightarrow 1$ ,  $e^{-h[x]} \rightarrow e^{-h_0} (1 - \xi[x])$ };

rhs = rhs /. { $h'[x] \rightarrow \xi'[x]$ ,  $h''[x] \rightarrow \xi''[x]$ ,  $D[h[x], \{x, 3\}] \rightarrow D[\xi[x], \{x, 3\}]$ ,
   $D[h[x], \{x, 4\}] \rightarrow D[\xi[x], \{x, 4\}]$ };

rhs = Simplify[rhs];

rhs = Expand[rhs];

rhs = rhs /. Table[ $\xi[x]^i \rightarrow 0$ , {i, 1, 4}];

rhs = rhs /. Table[ $\xi'[x]^i \rightarrow 0$ , {i, 1, 4}];

rhs = rhs /. Table[ $\xi''[x]^i \rightarrow 0$ , {i, 2, 4}];

rhs = rhs /. { $\xi''[x] \rightarrow e^{\omega t} (I k)^2 e^{I k x}$ ,  $\xi''''[x] \rightarrow e^{\omega t} (I k)^4 e^{I k x}$ };

lhs =  $\omega[k] * e^{i k x + \omega t}$ ;

 $\omega[k] = \text{Simplify}\left[\frac{\text{rhs}}{e^{i k x + \omega t}}\right]$ ;

 $\omega[k] = \text{Expand}[\omega[k]]$ ;

 $\omega[k] = \text{Collect}[\omega[k], \{k^2, k^4\}]$ ;

```

FIGURE 4.0.2. Basic *Mathematica* code for growth rate function, part 2.

```

n = 100; h =  $\frac{1}{n}$ ;  $\alpha = \frac{1}{20}$ ; L = 1; bc1 = 0; bc2 = 0;

U[t_] = Table[u_i[t], {i, 0, n}];
initc = Thread[U[0] = Table[Sin[ $\pi i$ ], {i, 0., 1,  $\frac{1-0}{n}}$ ]];
initc = Delete[initc, {{1}, {n+1}}];

A =
 $\frac{\alpha}{h^2}$  SparseArray[{Band[{1, 1}, {n+1, n+1}]  $\rightarrow$  -2, Band[{1, 2}, {n, n}]  $\rightarrow$  1, Band[{3, 2}, {n+1, n+1}]  $\rightarrow$  1},
  {n+1, n+1}];

b = SparseArray[{{2, 1}  $\rightarrow$  bc1, {n, 1}  $\rightarrow$  bc2}, {n+1, 1}];

F = Flatten[A.U[t] + b];

system = Thread[D[U[t], t] = Flatten[F]];
system = Delete[system, {{1}, {n+1}}];

sol = NDSolve[{system, initc}, Delete[U[t], {{1}, {n+1}}], {t, 0, 4}];

p1 = ParametricPlot3D[Evaluate[Table[{i h, t, First[u_i[t] /. sol]}, {i, 1, n-1}], {t, 0, 4},
  BoxRatios  $\rightarrow$  {4, 4, 2}, BoxStyle  $\rightarrow$  {Dashed}, ColorFunction  $\rightarrow$  Hue, PlotStyle  $\rightarrow$  {Opacity[0.3]},
  ViewPoint  $\rightarrow$  {2.20414, -2.08663, 1.49591}, ImageSize  $\rightarrow$  {230.951, 158.571},
  AxesLabel  $\rightarrow$  {Style[x, Medium], Style[t, Medium], Style[u, Medium]}, LabelStyle  $\rightarrow$  Directive[Small],
  PlotLabel  $\rightarrow$  "(b)", PlotRange  $\rightarrow$  All];

u[x_, t_] := Sin[ $\frac{\pi x}{L}$ ] e[ $-\frac{\alpha \pi^2 t}{L^2}$ ]

p2 = Plot3D[u[x, t], {x, 0, 1}, {t, 0, 4}, BoxStyle  $\rightarrow$  {Dashed}, ViewPoint  $\rightarrow$  {2.20414, -2.08663, 1.49591},
  ImageSize  $\rightarrow$  {230.951, 158.571}, BoxStyle  $\rightarrow$  {Dashed},
  AxesLabel  $\rightarrow$  {Style[x, Medium], Style[t, Medium], Style[u, Medium]}, LabelStyle  $\rightarrow$  Directive[Small],
  PlotLabel  $\rightarrow$  "(a)", PlotRange  $\rightarrow$  All];

p3 = Show[{p2, p1}, PlotLabel  $\rightarrow$  "(c)"];

GraphicsGrid[{{p2, p1, p3}}, Spacings  $\rightarrow$  {Scaled[.3], Scaled[0.01]}]

```

FIGURE 4.0.3. Basic *Mathematica* code for Example 3.2.1.

```

n = 100; h =  $\frac{1}{n}$ ;  $\alpha = \frac{1}{20}$ ; L = 1; bc1 = 0; bc2 = 0;

U[t_] = Table[u_i[t], {i, 0, n}];
initc = Thread[U[0] == Table[Sin[ $\pi i$ ], {i, 0., 1,  $\frac{1-0}{n}$ }}];

initc = Delete[initc, {{1}, {n+1}}];

A =
 $\frac{\alpha}{h^2}$  SparseArray[{Band[{1, 1}, {n+1, n+1}]  $\rightarrow$  -2, Band[{1, 2}, {n, n}]  $\rightarrow$  1, Band[{3, 2}, {n+1, n+1}]  $\rightarrow$  1},
{n+1, n+1}];

b = SparseArray[{{2, 1}  $\rightarrow$  bc1, {n, 1}  $\rightarrow$  bc2}, {n+1, 1}];

F = Flatten[A.U[t] + b];

system = Thread[D[U[t], t] == Flatten[F]];

system = Delete[system, {{1}, {n+1}}];

sol = NDSolve[{system, initc}, U[t], {t, 0, 4}];

u[x_, t_] := Sin[ $\frac{\pi x}{L}$ ] e $-\frac{\alpha \pi^2 t}{L^2}$ 

p1 = ParametricPlot3D[Evaluate[Table[{i+h, t, Abs[First[u_i[t] /. sol] - u[i+h, t]]}, {i, 1, n-1}],
{t, 0, 4}], BoxStyle  $\rightarrow$  Dashed, PlotRange  $\rightarrow$  All, BoxRatios  $\rightarrow$  {2, 2, 1}, ColorFunction  $\rightarrow$  Hue,
PlotStyle  $\rightarrow$  {Opacity[0.3]}, ViewPoint  $\rightarrow$  {2.20414, -2.08663, 1.49591}, ImageSize  $\rightarrow$  {230.951, 158.571},
AxesLabel  $\rightarrow$  {Style[x, Medium], Style[t, Medium], Style[u, Medium]}, PlotLabel  $\rightarrow$  "(a)", PlotRange  $\rightarrow$  All]

```

FIGURE 4.0.4. Basic *Mathematica* code for Example 3.2.1 (error (a)).

```

B = 1; k = 1;
f[x_, t_] := 1 + B * e $-k^4 t$  * Cos[k * x];

p1 = Plot3D[f[x, t], {x, 0, 2 *  $\pi$ }, {t, 0, 4}, PlotRange  $\rightarrow$  All, BoxStyle  $\rightarrow$  Dashed, PlotLabel  $\rightarrow$  "(a)",
AxesLabel  $\rightarrow$  {Style[x, Medium], Style[t, Medium], Style[u, Medium]}, PlotRange  $\rightarrow$  All,
ViewPoint  $\rightarrow$  {2.20414, -2.08663, 1.49591}, ImageSize  $\rightarrow$  {230.951, 158.571}]

n = 100; L = 2. *  $\pi$ ; h =  $\frac{L}{n}$ ;

U[t_] = Table[u_i[t], {i, 0, n}];
initc = Thread[U[0] == Table[1 + B Cos[k i], {i, 0, 2  $\pi$ ,  $\frac{2. \pi - 0}{n}$ }}];

A =
 $\frac{-1}{h^4}$  SparseArray[{Band[{1, 1}]  $\rightarrow$  6, Band[{2, 1}]  $\rightarrow$  -4, Band[{1, 2}]  $\rightarrow$  -4, Band[{3, 1}]  $\rightarrow$  1,
Band[{1, 3}]  $\rightarrow$  1, {1, 3}  $\rightarrow$  1, {1, n}  $\rightarrow$  -4, {1, n-1}  $\rightarrow$  1, {2, n}  $\rightarrow$  1, {n, 2}  $\rightarrow$  1, {n, n-1}  $\rightarrow$  1,
{n, n-1}  $\rightarrow$  1, {n+1, 2}  $\rightarrow$  -4, {n+1, 3}  $\rightarrow$  1}, {n+1, n+1}];

system = Thread[D[U[t], t] == Flatten[A.U[t]]];

sol = NDSolve[{system, initc}, U[t], {t, 0, 4}, MaxSteps  $\rightarrow$  20 000];

p2 = ParametricPlot3D[Evaluate[Table[{i h, t, First[u_i[t] /. sol]}, {i, 0, n}], {t, 0, 4},
BoxRatios  $\rightarrow$  {4, 4, 2}, PlotRange  $\rightarrow$  All, ColorFunction  $\rightarrow$  Hue, PlotStyle  $\rightarrow$  {Opacity[0.3]},
BoxStyle  $\rightarrow$  Dashed, PlotLabel  $\rightarrow$  "(b)",
AxesLabel  $\rightarrow$  {Style[x, Medium], Style[t, Medium], Style[u, Medium]}, PlotRange  $\rightarrow$  All,
ViewPoint  $\rightarrow$  {2.20414, -2.08663, 1.49591}, ImageSize  $\rightarrow$  {230.951, 158.571}]

p3 = Show[p1, p2, PlotLabel  $\rightarrow$  "(c)"]

GraphicsGrid[{{p1, p2, p3}}, Spacings  $\rightarrow$  {Scaled[.3], Scaled[0.1]}]

```

FIGURE 4.0.5. Basic *Mathematica* code for Example 3.2.2.

```

B = 1; k = 1;
f[x_, t_] := 1 + B * e-k4*t * Cos[k*x];
p1 = Plot3D[f[x, t], {x, 0, 2*π}, {t, 0, 4}, PlotRange → All, PlotRange → All,
  ViewPoint → {2.20414, -2.08663, 1.49591}, ImageSize → {230.951, 158.571}]

n = 100; L = 2.*π; h =  $\frac{L}{n}$ ;
U[t_] = Table[ui[t], {i, 0, n}];
initc = Thread[U[0] == Table[1 + B Cos[k i], {i, 0, 2 π,  $\frac{2. π - 0}{n}$ }]];

A =
 $\frac{-1}{h^4}$  SparseArray[{Band[{1, 1}] → 6, Band[{2, 1}] → -4, Band[{1, 2}] → -4, Band[{3, 1}] → 1,
  Band[{1, 3}] → 1, {1, 3} → 1, {1, n} → -4, {1, n-1} → 1, {2, n} → 1, {n, 2} → 1, {n, n-1} → 1,
  {n, n-1} → 1, {n+1, 2} → -4, {n+1, 3} → 1}, {n+1, n+1}];

system = Thread[D[U[t], t] == Flatten[A.U[t]]];

sol = NDSolve[{system, initc}, U[t], {t, 0, 4}, MaxSteps → 20 000];

p1 = ParametricPlot3D[Evaluate[Table[{i*h, t, Abs[First[ui[t] /. sol] - f[i*h, t]]}, {i, 0, n}],
  {t, 0, 4}, PlotRange → All, BoxRatios → {2, 2, 1}, ColorFunction → Hue, PlotStyle → {Opacity[0.3]},
  BoxStyle → Dashed, PlotLabel → "(a)",
  AxesLabel → {Style[x, Medium], Style[t, Medium], Style[u, Medium]}, PlotRange → All,
  ViewPoint → {2.20414, -2.08663, 1.49591}, ImageSize → {230.951, 158.571}]

```

FIGURE 4.0.6. Basic *Mathematica* code for Example 3.2.2 (error (a)).

```

h_t = (alpha_1 h^(4)[x] + beta_1 e^-h[x] h^(4)[x]) + (alpha_2 h''[x] + beta_2 e^-h[x] h''[x]) + (alpha_5 h''[x] h'[x]^2 + beta_5 e^-h[x] h''[x] h'[x]^2);
B = 8; M = 1; A = 72; Mp = -3; epsilon_y = 0; G = 2; m = 1; alpha_1 = -B*M*(1 - 15*epsilon_y); alpha_2 = A*Mp; alpha_5 = -1.5*A*Mp;
beta_1 = -B*M*(G - 1 + 15*epsilon_y); beta_2 = B*M*(G - 1 - epsilon_y); beta_5 = -B*M*(-5 + 5G - 29*epsilon_y + 80*m*epsilon_y);
w = k_max = N[Sqrt[alpha_2 / (2*alpha_1)]];

n = 100; L = N[2*pi]; h = L/n; k = 1; delta = 10^-2;
U[t_] = Table[u_i[t], {i, 0, n}]; V[t_] = Table[e^-u_i[t], {i, 0, n}];
initc = Thread[U[0] == Table[1 + delta Cos[k i], {i, 0., 2 pi, 2 pi/n}]];

A =
1/h^4
SparseArray[{Band[{1, 1}] -> 6, Band[{2, 1}] -> -4, Band[{1, 2}] -> -4, Band[{3, 1}] -> 1,
Band[{1, 3}] -> 1, {1, 3} -> 1, {1, n-1} -> 1, {1, n} -> -4, {2, n} -> 1, {n, 2} -> 1, {n, n-1} -> 1,
{n+1, n-1} -> 1, {n+1, 2} -> -4, {n+1, 3} -> 1, {n+1, n+1}}.U[t];

A1 = alpha_1 A;
A2 = Thread[Times[V[t], beta_1 A]];

B =
1/h^2 SparseArray[{Band[{1, 1}] -> -2, Band[{2, 1}] -> 1, Band[{1, 2}] -> 1, {1, n} -> 1, {n+1, 2} -> 1,
{n+1, n+1}}.U[t];

B1 = alpha_2 B;
B2 = Thread[Times[V[t], beta_2 B]];

F = 1/(2 h) SparseArray[{Band[{2, 1}] -> -1, Band[{1, 2}] -> 1, {1, n} -> -1, {n+1, 2} -> 1, {n+1, n+1}}.U[t];
F1 = Thread[Times[alpha_5 B, F^2]];
F2 = Thread[Times[V[t], beta_5 B, F^2]];

system = Thread[D[U[t], t] == Flatten[A1 + A2 + B1 + B2 + F1 + F2]];
sol1 = NDSolve[{system, initc}, U[t], {t, 0, 4}, MaxSteps -> 50000];

p1 = ParametricPlot3D[Evaluate[Table[{i h, t, First[u_i[t] /. sol1}], {i, 0, n}], {t, 0, 4},
BoxRatios -> {4, 4, 2}, PlotRange -> All, ColorFunction -> Hue, PlotStyle -> {Opacity[0.3]},
BoxStyle -> Dashed, ViewPoint -> {3.13682, -1.13812, 0.561281},
AxesLabel -> {Style[x, Medium], Style[t, Medium], Style[u, Medium]}, LabelStyle -> Directive[Small],
PlotLabel -> "(b)", PlotRange -> All]

h_t = (alpha_1 + beta_1 e^-h[x]) h^(4)[x] + (alpha_2 + beta_2 e^-h[x]) h''[x] + (alpha_5 + beta_5 e^-h[x]) h''[x] h'[x]^2;

sol2 =
NDSolve[
{D[u[x, t], t] == (alpha_1 + beta_1 e^-u[x,t]) D[u[x, t], {x, 4}] + (alpha_2 + beta_2 e^-u[x,t]) D[u[x, t], {x, 2}] +
(alpha_5 + beta_5 e^-u[x,t]) D[u[x, t], {x, 2}] (D[u[x, t], x])^2, u[x, 0] == 1 + delta Cos[k x], u[0, t] == u[2 pi, t],
(D[u[x, t], x] /. x -> 0) == (D[u[x, t], x] /. x -> 2 pi)}, u, {x, 0, 2 pi}, {t, 0, 4}, MaxSteps -> 50000];

p2 = Plot3D[Evaluate[u[x, t] /. sol2], {x, 0, 2 pi}, {t, 0, 4}, PlotRange -> All, BoxStyle -> Dashed,
ViewPoint -> {3.13682, -1.13812, 0.561281}, BoxRatios -> {4, 4, 2},
AxesLabel -> {Style[x, Medium], Style[t, Medium], Style[u, Medium]}, LabelStyle -> Directive[Small],
PlotLabel -> "(a)", PlotRange -> All]

p3 = Show[{p1, p2}, PlotLabel -> "(c)"]

GraphicsGrid[{{p2, p1, p3}}, Spacings -> {Scaled[.3], Scaled[0.01]}]

error =
ParametricPlot3D[Evaluate[Table[Abs[{i*h, t, First[u_i[t] /. sol1] - u[i*h, t] /. sol2}], {i, 0, n}],
{t, 0, 4}, PlotRange -> All, BoxRatios -> {2, 2, 1}, ColorFunction -> Hue, PlotStyle -> {Opacity[0.3]},
BoxStyle -> Dashed, ViewPoint -> {3.13682, -1.13812, 0.561281}, BoxRatios -> {4, 4, 2},
AxesLabel -> {Style[x, Medium], Style[t, Medium], Style[u, Medium]}, LabelStyle -> Directive[Small],
PlotRange -> All]

```

FIGURE 4.0.7. Basic *Mathematica* code for the simplified-nonlinear PDE and for error.

BIBLIOGRAPHY

- [1] V. G. GANZHA, AND E. V. VOROZHTSOV, *Numerical Solutions for Partial Differential Equations: Problem Solving Using Mathematica*. Boca Raton, FL: CRC (1996).
- [2] G. GARDINOWSKI, J. SCHMEIDEL, H. PFNR, T. BLOCK and C. TEGENKAMP, *Switchable Nanometer Contacts: Ultrathin Ag Nanostructures on Si(100)*, Applied Physics Letters 89, 063120 (2006).
- [3] M. KHENNER, *Analysis of A Combined Influence of Substrate Wetting and Surface Electromigration on A Thin Film Stability and Dynamical Morphologies*, in the special issue "Trends and perspectives in solid-state wetting" of the journal Comptes Rendus Physique 14, 607-618 (2013).
- [4] M. KHENNER, *Comparative Study of A Solid Film Dewetting in An Attractive Substrate Potentials with the Exponential and the Algebraic Decay*, Math. Model. Nat. Phenom. 3(5), 16-29 (2008).
- [5] P. K. KYTHE, P. PURI and M. R. SCHÄFERKOTTER, *Partial Differential Equations and Mathematica*, Boca Raton, FL: CRC (1997).
- [6] W. E. SCHIESSER, AND G. W. GRIFFITHS, *A Compendium of Partial Differential Equation Models: Method of Lines Analysis with Matlab*. Cambridge UK: Cambridge University Press, (2009).
- [7] I. P. STAVROULAKIS, and S. A. TERSIAN, *Partial Differential Equations: An Introduction with Mathematica and MAPLE*. 2nd ed. River Edge, NJ: World Scientific (2004).
- [8] K. WOODS, *An Analysis of the Electromigration of Atoms Along A Crystal Surface Using the Method of Lines*, Senior Research Seminar (MATH 498) Fall (2011).
- [9] A. VANDE WOUWER, Ph. SAUCEZ, and W. E. SCHIESSER, *Adaptive Method of Lines*. Boca Raton: Chapman and Hall/CRC (2001).
- [10] J. ZHAO, R. YU, S. DAI, and J. ZHU, *Kinetic Faceting of the Low Index W Surfaces Under Electrical Current*, Surface Science 625, 10-15 (2014).

[11] <https://reference.wolfram.com/mathematica/tutorial/NDSolvePDE.html>, Last Accessed April 2014.

