Journal of Computational Physics **167,** 316–371 (2001)
doi:10.1006/jcph.2000.6677, available online at http://www.idealibrary.com on IDEAL®

Computation for Electromigration in Interconnects of Microelectronic Devices[1]

Amir Averbuch,* Moshe Israeli,† Igor Ravve,*,[2] and Irad Yavneh†

**Department of Computer Science, School of Mathematical Sciences, Tel Aviv University, Tel Aviv 69978, Israel; and †Faculty of Computer Science, Technion, Haifa 32000, Israel*
E-mail: amir@math.tau.ac.il

Received January 20, 1999; revised November 8, 2000

Reliability and performance of microelectronic devices depend to a large extent on the resistance of interconnect lines. Voids and cracks may occur in the interconnects, causing a severe increase in the total resistance and even open circuits. In this work we analyze void motion and evolution due to surface diffusion effects and applied external voltage. The interconnects under consideration are three-dimensional (sandwich) constructs made of a very thin metal film of possibly variable thickness attached to a substrate of nonvanishing conductance. A two-dimensional level set approach was applied to study the dynamics of the moving (assumed one-dimensional) boundary of a void in the metal film. The level set formulation of an electromigration and diffusion model results in a fourth-order nonlinear (two-dimensional) time-dependent PDE. This equation was discretized by finite differences on a regular grid in space and a Runge–Kutta integration scheme in time, and solved simultaneously with a second-order static elliptic PDE describing the electric potential distribution throughout the interconnect line. The well-posed three-dimensional problem for the potential was approximated via singular perturbations, in the limit of small aspect ratio, by a two-dimensional elliptic equation with variable coefficients describing the combined local conductivity of metal and substrate (which is allowed to vary in time and space). The difference scheme for the elliptic PDE was solved by a multigrid technique at each time step. Motion of voids in both weak and strong electric fields was examined, and different initial void configurations were considered, including circles, ellipses, polygons with rounded corners, a butterfly, and long grooves. Analysis of the void behavior and its influence on the resistance gives the circuit designer a tool for choosing the proper parameters of an interconnect (width-to-length ratio, properties of the line material, conductivity of the underlayer, etc.).

[1] The research of the first two authors was supported by Israeli Ministry of Science and Technology Grant 9672-1-96-9672-3-98.

[2] Recipient of the 1997–1998 Israeli Academy of Sciences Post-Doctoral Fellowship for Research at the Computer Science Department, Tel Aviv University, Israel.

0021-9991/01 $35.00

1. INTRODUCTION

Evolution and migration of voids in the interconnects of microelectronic devices lead to considerable variations in the total resistance of a line, depending on the void shape and size and on the specific conductivities of the metal and underlayer. Unpredictable propagation and dynamics of the void may cause open circuits and other failures. Increase of the interconnect resistance can often be considered a circuit failure criterion. Analysis of the grain–void interface motion provides valuable information for a circuit designer and makes it possible to predict the circuit performance and reliability for currently available and newly emerging materials. The simulation results can be used for better choice of the circuit parameters, interconnect geometry, underlayer properties, etc.

There are several numerical approaches to track the problem of propagating interfaces (moving curves and surfaces). Bower and Freund [2] developed a finite element formulation and applied it to compute the effect of diffusion and deformation in an electrically conducting, deformable solid. They idealized an interconnect as a two-dimensional assembly of grains, with planar grain boundaries. In the absence of electrical current, the diffusion is assumed to be driven by a variation in chemical potential, associated mainly with the free energy of the surface, which in turn is related to the surface curvature. If an electric current flows down the line, it gives rise to an additional driving force for diffusion. These assumptions lead to a set of governing partial differential equations of motion of first order in time and fourth order in space, with the corresponding initial and boundary conditions. A weak formulation to solve the equations of motion employing six-node triangular elements in space and finite differences in time was developed. The finite element approach was further extended by Xia *et al.* [28] who examine the mechanisms of failure in the interconnects associated with diffusion of material along the surfaces, interfaces, or grain boundaries. The authors implemented an improved mesh adaption strategy.

The transmission-line matrix (TLM) approach was used by Gui *et al.* [4] to study the one-dimensional electromigration problem under pulsed direct current conditions. The TLM approach is a time domain numerical technique involving the use of electrical circuit analogs. In this method, established for solving diffusion and diffusion-related problems, voltage is linked to the vacancy concentration, and current is linked to the mass flux. Another method that can be used for the electromigration problem is the curve tracking procedure [5]. Front tracking is a numerical method in which surfaces of discontinuity are given explicit computational degrees of freedom. These degrees of freedom are supplemented by degrees of freedom representing continuous solution values at regular grid points.

In this paper, we develop an analytical and numerical approach to track the dynamics of the void motion under the action of a surface diffusion and an external voltage. The motion is studied using a level set method, where the closed interface between the void and the surrounding material is identified with the zero level of a special two-dimensional function which depends on space and varies in time. In the case of a single interface, separating two phases (e.g., a grain and a void inside), the central idea of the level set method [6, 7] is to follow the evolution of a special function whose zero level corresponds to the position of the moving interface. Motion of the interfaces with sharp corners is studied with the use of essentially nonoscillatory (ENO) schemes with different orders [14–16].

Consider a boundary which is either a curve in two-dimensions [9] or a surface in three dimensions [10], separating one region from another. Usually, the basic assumption is that this curve or surface moves in its normal direction with a known speed function F. The tangential motion is neglected. The normal speed depends on the local properties of the front at the given point (such as curvature and/or its derivatives, derivatives of electric field strength, etc.), on the global properties (such as integrals along the front, which, in turn, depend on the shape and position of the front), and on the independent properties (material constants, etc.). Given the initial position for an interface Γ, where Γ is a two-dimensional closed curve, the level set method views Γ as the zero level of a function $\phi(x, y, t)$. Initially, the level function is defined to be a signed distance from a given point to the interface line. The function is negative inside the front and positive outside the front. For a two-dimensional case, the level function ϕ is governed by a partial differential equation, which involves the normal velocity F of the interface curve [11]:

$$\phi_t + F \cdot \sqrt{\phi_x^2 + \phi_y^2} = 0. \tag{1.1}$$

Generally, F depends on the specific physical problem under consideration. In the present study, the front velocity includes the surface diffusion and electromigration components, depending on the second derivatives of the interface curvature K and of the voltage U, respectively [26, 31]. The motion of the interface separating the void from the metal is due to surface diffusion and therefore the interface can be viewed as "sharp" on the macro scale. The connection between sharp and diffused interface motion laws via gradient flow can be found in [33]. The derivatives are taken with respect to the interface arclength s:

$$F = BK_{ss} + \alpha U_{ss}. \tag{1.2}$$

The constant coefficients B and α express the contribution of the diffusion forces and the field forces, respectively. They depend on the properties of the material, temperature, etc. and are considered constant. B is always positive and α is positive for the migration of a void surrounded by a conducting material. The ratio between the diffusion forces and the field forces is of great importance since it defines the dynamics and stability of the void motion. In any case, the void migrates in the direction opposite to the external voltage, but its evolution varies, depending on α/B. When the diffusion forces prevail, any initial shape of the void becomes circular. For a moderately strong field, the circular shape of the void remains in equilibrium, but such equilibrium is no longer stable, and another noncircular stable equilibrium shape exists. For an extremely strong field, a void of any shape, whose area exceeds a definite threshold, ceases to be stable and splits into several slit-like smaller voids.

The governing PDE needs to be solved only in the vicinity of the interface; however it turns out that the computational bottleneck is presently the solution of the two-dimensional, elliptic, electrostatic equation. For simplicity we used initially an algorithm that solves the diffusion equation for the entire internal domain of the two-dimensional computational box; the definition of the normal velocity is extended from the interface points to all points of the box. For this, we consider an arbitrary point and draw a line of constant level that passes through this point. The level value vanishes at the interface points only; otherwise we deal with a nonzero level line. The curvature of the level line, the voltage, and their derivatives with respect to the level line arclength can be established at any point. Thus, the governing

PDE is extended to noninterface points. On the extension, K_{ss} is the second derivative of the curvature of the level line, which passes through a given point, with respect to the arclength of this level line, and U_{ss} is the second derivative of the potential with respect to the level line arclength.

The front velocity depends on the second derivative of the curvature, and the curvature, in turn, is defined by the second derivatives of the level function. Therefore the governing PDE is of fourth order in space and of first order in time. The equation is solved numerically by applying an explicit time-integration approach: finite difference discretization in space and a Runge–Kutta 2 integration scheme in time. Such a low-order Runge–Kutta scheme is justified by the relatively small magnitude of the time step, which is dictated by the stringent stability requirement. For the explicit scheme, the upper bound of the allowable step is proportional to the fourth power of the spatial grid resolution. The area which is confined by the closed front remains constant during the void motion. The applied scheme does not necessitate boundary conditions (BC). Instead of BC, we use one-sided differences to approximate the derivatives at the boundary points of the computational box.

The interface configuration is specified by a finite set of discrete points which generates a closed curve. These points do not necessarily coincide with the grid points of the computational box. A spline technique is used to restore the parametric equations of the front. We apply the standard cubic spline using cyclic boundary conditions. To estimate the new location of the front, the zero line of the level function is established. For this, we solve numerically a nonlinear equation $\phi(x, y) = 0$, moving along the gradient lines in ascending and descending directions. The previous location of the front is used as an initial approximation for its current location. At nongrid points, the level function and its derivatives are approximated by a double polynomial interpolation of fifth order.

The level function, defined initially as a distance from the given point to the interface curve, ceases to be a distance function after at least one time step. In this case, a reinitialization procedure is necessary. When a time step is completed, we find the interface configuration as the zero level line, and then we update the level function, calculating the distances between the grid nodes and the moving front. If the reinitialization procedure is employed several times per each time step, then the level function coincides with the distance to the interface curve. The number of reinitializations corresponds to the order of the Runge–Kutta scheme. For the level function, coinciding with the distance, the gradient is of unit length:

$$\phi_x^2 + \phi_y^2 = 1. \tag{1.3}$$

Then, the following relations hold for the curvature and its second derivative. These simplify the computational procedure:

$$K = \nabla^2 \phi; \quad K_{ss} = \nabla^2 K - K^3. \tag{1.4}$$

Analytic results were also obtained for the rate of change of the interface length.

The voltage distribution $U(x, y)$ is described by an approximate elliptic PDE, which is derived from the well-posed three-dimensional potential problem for the two-layer interconnect. The assumptions and derivation for the case of small aspect ratio is described later

and results in

$$\frac{\partial}{\partial x}\left(k\frac{\partial U}{\partial x}\right)+\frac{\partial}{\partial y}\left(k\frac{\partial U}{\partial y}\right)=0, \tag{1.5}$$

where $k(x, y)$ is the specific electric conductivity of the material, and it varies in space through the presence of the void. A finite difference scheme was developed and analyzed to solve Eq. (1.5). This equation requires continuous distribution of a specific conductivity. However, this distribution in the physical system under consideration is discontinuous. The conductivity inside the void differs by a finite value from that of the surrounding material. In the finite difference approach, the discontinuous distribution of the conductivity is smoothed by a special function. The finite difference discretization of the domain leads to a linear algebraic equation set with a sparse band matrix. This set is solved with an effective multigrid iterative procedure, developed specially for a boundary value elliptic problem with rapidly varying coefficients. The solution of the previous time step is used as an initial approximation for the current step which allows fast convergence. Then the total resistance of the interconnect may be established.

A specific feature of the level set procedure, related to the void motion in a strong field, was developed. The generating points of the interface curve, which were initially equally or almost equally spaced, become located more and more nonuniformly as the time steps proceed. After a number of steps, most of the generating points will be located in a fairly small portion of the interface curve, while the long portions contain only a few generating points. Therefore, the level set analysis requires an additional procedure to redistribute the generating points in a "forced" equally spaced manner after each time step or after several time steps. This redistribution is done using the spline technique.

Various simulations for void motions and evolutions were performed. These simulations assume different initial configurations for the interface: ellipse, polygons with rounded vertices (equilateral triangle and square), butterfly, and a long groove with rounded ends. The simulation results include three-dimensional plots of the conductivity and voltage distribution, the current configuration of the void, the values of the interface perimeter, the area confined inside, and the total resistance of the interconnect at current time and the graphs for the interface curvature and its derivatives vs the arclength, the voltage and its derivatives vs the arclength, and the resulting normal velocity of the front.

Thus far, existing level set finite difference formulations consider the second order (in space) governing equations where the speed function depends on the curvature only and is independent of the derivatives of the curvature. However, in diffusion problems in materials science, the normal velocity of the front is thought to be proportional to the second derivative of the curvature with respect to the interface arclength. This yields a fourth-order equation which is analyzed in this paper. The level set method is still applicable; however, the integration algorithm differs a great deal from that for a second-order equation. In particular, it was established that for central differences in space and Runge–Kutta in time, the zero level has to be computed and the distance function has to be reinitialized n times per each time step, where n is the order of the Runge–Kutta scheme.

The paper has the following structure: In Section 2 we describe the basic governing equations for the void dynamics and evolution. Section 3 describes the numerical (finite difference in space and Runge–Kutta 2 in time) algorithm for the solution of the governing PDEs. The voltage distribution is found using a special multigrid solver which is described in Section 4. Results of simulations are given in Section 5.

2. MODELING SURFACE DIFFUSION AND ELECTROMIGRATION

2.1. *Estimating the Curvature and Its Derivatives*

Consider the time-dependent function $\phi(x, y, t)$ whose zero level at any fixed time t yields an instant configuration of the moving void interface $\phi(x, y) = 0$. If the interface line $\Gamma(t)$ is specified, then the level function $\phi(x, y, t)$ may be defined as a signed distance from the given point (x, y) to the front line $\Gamma(t)$. In this work only closed fronts are considered. The normal velocity F of the moving front varies for different points of the interface. It depends on the curvature of the front, its derivatives, the distribution of electric fields along the front, and other factors. With these assumptions, the evolution of the level function is described by Eq. (1.1) where the speed function F is given by Eq. (1.2).

By integrating Eq. (1.1) we obtain $\phi(x, y, t)$, and then for any fixed $t = t_0$, we solve $\phi(x, y, t_0) = 0$ and obtain the interface $\Gamma(t_0)$). The initial position of the front $\Gamma(0)$ should be specified. However, the boundary conditions for Eq. (1.1) are not needed, provided the sizes of the computational box considerably exceed those of the domain inside the front. At the internal points of the computational box we used central differences to approximate the derivatives. At the bound lines of the box, we used one-sided differences.

Assume that $U(x, y)$ is the electric potential within the computational box, U^- and U^+ are its values on the left and right vertical boundaries, respectively, and U_n is the normal derivative on the boundary. The distribution $U(x, y)$ is governed by a static (time-independent) elliptic equation (1.5) with boundary conditions $U_n = U_y = 0$ along the horizontal bounds of the rectangular computational box. U^- and U^+ are assumed to be constant; $U^+ - U^-$ is the external voltage applied to the interconnect. $k(x, y)$ is a given function which represents the distribution of the material conductivity within the box. Usually, we assume $k = k_{out} = const$ outside the interface and $k = k_{in} = const$ inside. For a void that resides inside the interconnect we have $k_{out} > 0$, $k_{in} \geq 0$, and $k_{out} \gg k_{in}$. Usually, k_{in} is nonzero if the underlayer's conductivity is taken into account and averaged through the interconnect thickness.

Since the front moves in time, Eq. (1.5) contains the time-varying coefficient $k(x, y, t)$. Thus, its solution $U(x, y)$ is time-dependent too. Normalizing the time unit, we set the diffusion coefficient B to be equal to one, and Eqs. (1.1) and (1.2) become

$$\phi_t + (K_{ss} + \alpha U_{ss})\sqrt{\phi_x^2 + \phi_y^2} = 0. \tag{2.1}$$

If ϕ is a distance function, then its gradient is of unit length (Eq. 1.3), and Eq. (2.1) is simplified to

$$\phi_t + K_{ss} + \alpha U_{ss} = 0. \tag{2.2}$$

Our goal is to express K and K_{ss} in terms of ϕ and the Cartesian components of its spatial derivatives, and to express U_{ss} in terms of U, ϕ, and the Cartesian components of their derivatives. For the rectangular grid, these Cartesian derivatives are more convenient to use than the arclength derivatives. The above is accomplished via the following steps.

Estimating the absolute value of the distance function. First, let us describe the procedure for establishing the level function ϕ at any point in the plane when the interface contour is specified. Assume $\phi(x, y)$ is a signed distance between the fixed (but arbitrary) point

(x, y) and a specified continuous closed curve. Assume also that this closed line is defined parametrically in a two-dimensional region by the Cartesian components $f_x(r)$ and $f_y(r)$, where r is an arbitrary parameter, not necessarily the arclength. However, in particular, r may be the arclength or proportional to the arclength. In some practical cases, $f_x(r)$ and $f_y(r)$ are not specified explicitly, but the line is set by a sequence of individual points. Then, the parametric functions may be restored by applying a spline technique. The obvious way to search for a minimum distance between the given point (x, y) and the specified line is as follows:

$$\begin{aligned} \phi^2(r) &= [f_x(r) - x]^2 + [f_y(r) - y]^2 \to \min \\ H(r) &\stackrel{\text{def}}{=} [f_x(r) - x]\frac{df_x}{dr} + [f_y(r) - y]\frac{df_y}{dr} = 0. \end{aligned} \tag{2.3}$$

Estimating the distance by the Newton–Raphson iterative procedure. Equation (2.3) can be solved numerically for the unknown r by applying the Newton–Raphson iterative procedure. Assume that the nth approximation for r is already found. Then the next approximation for r is defined as

$$r^{(n+1)} = r^{(n)} - H\left[r^{(n)}\right] / H'\left[r^{(n)}\right],$$

where

$$H'(r) = \left(\frac{df_x}{dr}\right)^2 + \left(\frac{df_y}{dr}\right)^2 + [f_x(r) - x]\frac{d^2 f_x}{dr^2} + [f_y(r) - y]\frac{d^2 f_y}{dr^2}. \tag{2.4}$$

Estimating the distance by the golden section method. The Newton method converges rapidly (quadratically), but if the initial approximation is not properly chosen, it does not converge at all. In these cases we apply the standard *golden section* method to establish the distance from the given grid point to the interface line. The golden section always converges, but it converges slowly (linearly). The function $f(t)$ whose minimum is sought, can be either an unsigned distance or a square of the distance. Now assume that the proper value of r is established. Without knowing the sign, the level function $\phi(x, y)$ can be defined by Eq. (2.3).

Establishing the sign of the distance function. To establish the sign of ϕ, we have to discover whether the given point (x, y) is located inside or outside the front. For this purpose, consider the distance vector $\boldsymbol{N}$ (connecting the given point in the plane to the nearest point on the line), the outward normal vector $\boldsymbol{n}$, and the tangent vector $\boldsymbol{t}$, as described in Fig. 1. Their components are

$$\boldsymbol{N}(f_x - x,\ f_y - y); \quad \boldsymbol{t}\left(\frac{df_x}{dr}, \frac{df_y}{dr}\right); \quad \boldsymbol{n}\left(\frac{df_y}{dr}, \frac{df_x}{dr}\right). \tag{2.5}$$

Vectors $\boldsymbol{t}$ and $\boldsymbol{n}$ are not necessarily of unit length. Equation (2.5) expresses the fact that $\boldsymbol{N} \cdot \boldsymbol{t} = 0$, i.e., $\boldsymbol{N}$ is normal to $\boldsymbol{t}$. Then, it follows that the distance vector $\boldsymbol{N}$ is collinear with the normal vector $\boldsymbol{n}$ If $\boldsymbol{N} \cdot \boldsymbol{n} > 0$, their directions coincide, and this means that the considered point (x, y) is located inside the front. In this case, the distance function ϕ should be negative. For $\boldsymbol{N} \cdot \boldsymbol{n} < 0$, ϕ is positive,

$$\operatorname{sign} \phi = -\operatorname{sign}(\boldsymbol{N} \cdot \boldsymbol{n}) = \operatorname{sign}\left[(f_y - y)\frac{df_x}{dr} - (f_x - x)\frac{df_x}{dr}\right]. \tag{2.6}$$

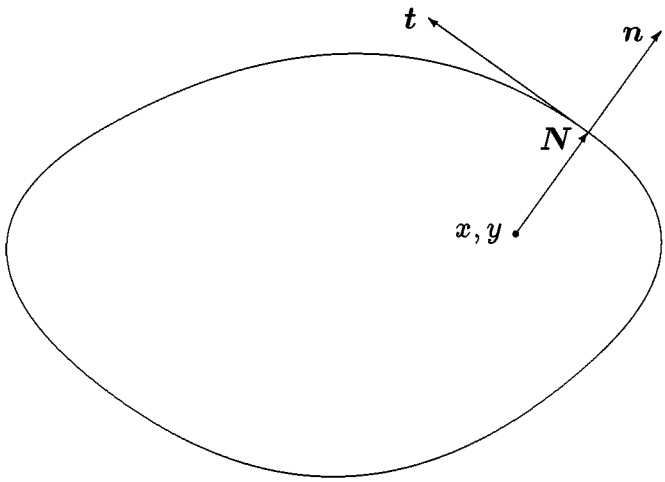


FIG. 1. Establishing the distance function sign.

Level line curvature. For an arbitrary function $\phi(x, y)$, which does not need to be a distance function, the components of the unit normal to the level line are

$$n_x = \frac{\phi_x}{\sqrt{\phi_x^2 + \phi_y^2}}, \quad n_y = \frac{\phi_y}{\sqrt{\phi_x^2 + \phi_y^2}}. \tag{2.7}$$

The curvature is a divergence of an outward unit normal [8]:

$$K = \frac{\partial n_x}{\partial x} + \frac{\partial n_y}{\partial y} = \nabla \cdot \mathbf{n} = \frac{\phi_{xx}\phi_y^2 - 2\phi_{xy}\phi_x\phi_y + \phi_{yy}\phi_x^2}{\left(\phi_x^2 + \phi_y^2\right)^{3/2}}. \tag{2.8}$$

We assume that $\phi(x, y)$ is a distance function. Then identity (1.3) is valid, and Eq. (2.8) is simplified to

$$K(x, y) = \phi_{xx} + \phi_{yy} = \nabla^2\phi, \tag{2.9}$$

i.e., the level line curvature is equal to the Laplacian of the distance function. Although relationships for the curvature (2.8) and (2.9) are analytically equivalent, provided the level function coincides with the distance, they lead to different numerical schemes. These schemes are both conditionally stable, but the stability constraints (relations between the cell size in the xy plane and the time step) prove to be different. Equation (2.9) requires the second derivatives with respect to one Cartesian coordinate only. Equation (2.8) utilizes, in addition to this data, the first derivatives and the mixed second derivative. Unfortunately, the scheme based on the simplified linear differential relation (2.9) between the curvature and the distance function required a smaller time step value in order to allow the numerical solution to proceed stably, and therefore it is not necessarily recommended for practical use. However, the information about the unit length of the gradient vector can be introduced into Eq. (2.8), and this yields

$$K = \phi_{xx}\phi_y^2 - 2\phi_{xy}\phi_x\phi_y + \phi_{yy}\phi_x^2. \tag{2.10}$$

Note that the stability constraints of the numerical schemes, approximating the curvature by second-order differences using Eq. (2.8) and Eq. (2.10), are the same.

Arclength derivative of voltage. We now derive the expression for the derivatives of an arbitrary function U (for example, the distribution of electric potential) with respect to the arclength of the level line. The first derivative is

$$U_s = -U_x\phi_y + U_y\phi_x. \tag{2.11}$$

By using Eq. (2.8), the second derivative becomes

$$U_{ss} = -\frac{\partial U_s}{\partial x}\phi_y + \frac{\partial U_s}{\partial y}\phi_x = U_{xx}\phi_y^2 - 2U_{xy}\phi_x\phi_y + U_{yy}\phi_x^2 - KU_n. \tag{2.12}$$

Equation (2.12) may be rewritten as

$$\begin{aligned} U_{ss} + U_{nn} + KU_n &= \nabla^2 U \\ U_n &= U_x\phi_x + U_y\phi_y \\ U_{nn} &= U_{xx}\phi_x^2 + 2U_{xy}\phi_x\phi_y + U_{yy}\phi_y^2, \end{aligned} \tag{2.13}$$

where U_n and U_{nn} are the first and second derivatives with respect to the distance in the normal direction.

Arclength derivative of curvature. So far, $U(x, y)$ is an arbitrary function. In particular, Eq. (2.13) can be applied to the curvature:

$$K_{ss} + K_{nn} + KK_n = \nabla^2 K. \tag{2.14}$$

In order to simplify Eq. (2.14), we derive the relation between the curvature K and its normal derivative K_n. For this purpose we differentiate Eq. (1.3) twice with respect to x and y:

$$\begin{cases} \phi_x\phi_{xx} + \phi_y\phi_{xy} = 0 \\ \phi_x\phi_{xy} + \phi_y\phi_{yy} = 0 \end{cases} \quad \begin{cases} \phi_{xx}^2 + \phi_x\phi_{xxx} + \phi_{xy}^2 + \phi_y\phi_{xxy} = 0 \\ \phi_{xy}^2 + \phi_x\phi_{xyy} + \phi_{yy}^2 + \phi_y\phi_{yyy} = 0. \end{cases} \tag{2.15}$$

It follows from (2.15) that

$$\phi_{xx}\phi_{yy} = \phi_{xy}^2. \tag{2.16}$$

In other words, the Jacobian for ϕ_x and ϕ_y vanishes, since these two functions are dependent through Eq. (1.3). Introducing (2.16) and (2.9) into the second set of Eq. (2.15), we obtain:

$$\phi_{xx}K + \frac{\partial \phi_{xx}}{\partial n} = 0; \quad \phi_{yy}K + \frac{\partial \phi_{yy}}{\partial n} = 0. \tag{2.17}$$

Set (2.17) yields

$$K^2 + K_n = 0. \tag{2.18}$$

Now the second normal derivative of the curvature can be easily obtained:

$$K_{nn} = \frac{\partial K_n}{\partial n} = -\frac{\partial}{\partial n} K^2 = -2KK_n = 2K^3. \tag{2.19}$$

Insertion of (2.18) and (2.19) into (2.14) yields

$$K_{ss} = \nabla^2 K - K^3. \tag{2.20}$$

Note that the stability constraint for the difference scheme that uses Eq. (2.13) allows a larger step value than those that use Eq. (2.20). Therefore, the simplified relation (2.20) for the second derivative of curvature with respect to the arclength is not necessarily recommended for the numerical implementation.

Derivation of the two-dimensional electrostatic equation. We consider a conducting strip made of a thin metal film, attached to a strip of nonzero conductivity substrate. The metal film may be continuous or it may be made of conducting patches with voids in between. We allow the metal film and substrate to have variable thickness. In the present formulation we neglect the interface resistance. The strip is attached to electrodes at its ends. We may want to compute the total resistance of the strip as well as the local field strength which determines the resulting electromigration. This is a more realistic model than the model based on the assumption of a zero conductivity substrate. It also allows us to consider the behavior of a metal film with varying effective thickness at no extra cost. More details are given in the Appendix.

The three-dimensional problem. Ohm's law implies $\mathbf{j} = \sigma \mathbf{E} = -\sigma \nabla_3 \phi$, where $\mathbf{j}$ is the electric current density vector, E is the electric field vector, ϕ is the electric potential, and σ is the material conductivity. For steady fields, Maxwell's equations with vanishing space charge give

$$\nabla_3 \cdot \mathbf{j} = 0, \quad \text{where} \quad \nabla_3 = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right). \tag{2.21}$$

Hence

$$\nabla_3 \cdot (\sigma \nabla_3 \phi) = 0. \tag{2.22}$$

At all external (lateral) boundaries there is no flux in the direction of the normal, $\mathbf{n}$, so that $\mathbf{n} \cdot \mathbf{j} = 0$, and using (1.1) one gets

$$\mathbf{n} \cdot \nabla_3 \phi = 0. \tag{2.23}$$

The conditions (2.23), together with values of the potential specified at the two ends of the strip and the continuity and jump conditions at the interface, constitute boundary conditions for Eq. (2.22) in the two layers. Thus the three-dimensional potential can be found, in principle, as the solution of a well-posed three-dimensional boundary value problem. However, such a solution can be very expensive to get in the present geometry, in particular as singularities in the solution will appear at sharp geometrical corners at crystal boundaries or voids, requiring high resolution or complicated integration formulas. To avoid this (probably unrealistic) behaviour of the solution and to avoid solving three-dimensional problems many times, as required by the time development of the process, we proceed with an approximate approach suggested by (singular) perturbation theory.

The two-dimensional equation. We assume that ϕ and σ change over a characteristic length scale L in the horizontal directions x and y but over a scale H in the vertical. Furthermore we assume that $\epsilon = H/L \ll 1$. Using scaled variables in Eq. (2.22),

$$(X, Y, Z) = (x/L, y/L, z/H), \tag{2.24}$$

we get

$$\epsilon^2 \nabla_2(\sigma \nabla_2 \phi) + \frac{\partial}{\partial Z}\left(\sigma \frac{\partial \phi}{\partial Z}\right) = 0, \quad \text{where} \quad \nabla_2 = \left(\frac{\partial}{\partial X}, \frac{\partial}{\partial Y}\right). \tag{2.25}$$

Singular perturbation analysis considers an expansion

$$\phi = \phi^0 + \epsilon^2 \phi^1 + \epsilon^4 \phi^2 + \cdots, \tag{2.26}$$

where the ϕ^k are functions of order 0(1) in ϵ.

Substitution of Eq. (2.26) in Eq. (2.25) gives relations for the functions ϕ_k by grouping terms according to their order in ϵ and equating each group to zero. The zeroth-order term gives $\frac{\partial^2 \phi_0}{\partial Z^2} = 0$; thus ϕ_0 is a linear function in z for every x and y, while taking into account (2.23) kills off the z dependence, so that

$$\phi^0 = \phi^0(X, Y). \tag{2.27}$$

Thus at this stage ϕ^0 is an arbitrary function of the horizontal coordinates X and Y. The first-order equation and the boundary conditions in Z result ultimately in the approximate two-dimensional equation for ϕ^0 (see Appendix A)

$$\nabla_2(h_1\sigma_1 + h_2\sigma_2)\nabla_2 \phi^0 = 0, \tag{2.28}$$

where h_1, σ_1 and h_2, σ_2 are, respectively, the heights and conductivities of the two layers under consideration. This equation is solved with boundary conditions in the (X, Y) plane.

Note that the approximate independence of the potential ϕ on the Z coordinate also justifies the two-dimensional approach for the electromigration equation. This behaviour is a consequence of the small aspect ratio assumption and the normal derivative boundary conditions (2.23), where one must also use a small slope assumption.

Governing equations. Finally, we collect all the governing equations for the void interface evolution under the surface diffusion and electromigration:

$$\left.\begin{aligned} &\phi_t + K_{ss} + \alpha U_{ss} = 0 \\ &\nabla \cdot k \nabla U = 0 \end{aligned}\right\} \quad \text{Partial differential equations}$$

$$\left.\begin{aligned} &K = \phi_{xx}\phi_y^2 - 2\phi_{xy}\phi_x\phi_y + \phi_{yy}\phi_x^2 = \nabla^2\phi \\ &K_{ss} = \nabla^2 K - K K_n - K_{nn} = \nabla^2 K - K^3 \\ &U_{ss} = \nabla^2 U - K U_n - U_{nn} \\ &K_n = \nabla K \cdot \nabla \phi = -K^2 \\ &U_n = \nabla U \cdot \nabla \phi \\ &K_{nn} = K_{xx}\phi_x^2 + K_{xy}\phi_x\phi_y + K_{yy}\phi_y^2 = 2K^3 \\ &U_{nn} = U_{xx}\phi_x^2 + 2U_{xy}\phi_x\phi_y + U_{yy}\phi_y^2. \end{aligned}\right\} \quad \text{Algebraic relations} \tag{2.29}$$

In particular, if we take into consideration only the surface diffusion and neglect the electromigration, set (2.29) is simplified to become

$$\phi_t + K_{ss} = 0. \tag{2.30}$$

In this case, the only possible equilibrium configuration of the front is circular. Indeed, assume the equilibrium takes place: $\phi_t = 0$. Then K_{ss} also vanishes, and $K(s) = as + b$. Due to continuity of the curvature, $K(0) = K(s_{\max})$ where $s_{\max}$ is the length of the closed interface. Therefore, $a = 0$, and $K = \text{const}$. The constant curvature front is circular and $K = 1/\text{radius}$. For the front evolution governed by Eqs. (2.29) and (2.30), the area A inside the interface remains constant:

$$\frac{dA}{dt} = \oint F\,ds = \oint (K_{ss} + \alpha U_{ss})\,ds = K_s + \alpha U_s|_0^{s_{\max}} = 0. \tag{2.31}$$

This integral vanishes because of the continuity of K_s and U_s. However, the length of the interface does not remain constant. For the governing Eq. (2.30), the perimeter decreases as the arbitrary initial shape tends to a circle while keeping the area of the void constant. The constant area inside the void is in agreement with the material conservation law. However, for some other types of governing equations, the area which is confined by the front is no longer constant. Consider, for example, the second-order equation

$$\phi_t - K = 0 \quad \text{or} \quad \phi_t - \nabla^2\phi = 0, \tag{2.32}$$

where the normal speed of the front coincides with the curvature up to the sign. The rate of change of the area is

$$\frac{dA}{dt} = \oint F(K)\,ds = -\oint K\,ds = -2\pi. \tag{2.33}$$

Consequently, the confined area decreases uniformly, and in $A_0/(2\pi)$ normalized time units, the front collapses to a point. An example of the numerical simulation of such motion is presented in Fig. 2.

Equation (2.32) is solved by finite differences in space and Runge–Kutta 2 in time. Since the governing equation is of second order in space, the time step is limited, because of stability reasons, to being

$$\Delta t = \lambda h^2, \quad h = \min(\Delta x, \Delta y). \tag{2.34}$$

We set $\lambda = 1/8$ and we obtained a stable scheme. The initial front is elliptic with semi-axes 1 and 2. Thus, the initial area is 2π, and it should collapse after one time unit. The size of the computational box is 8×8, and the space grid is 60×60 nodes (or 59×59 intervals). The spatial resolution is $h = 8/59 = 0.1356$. The time step is $\Delta t = 2.298 \cdot 10^{-3}$. The internal phase should completely disappear after 435 steps, and it really does. Due to the small size of the resulting area confined by the front, it is impossible to perform further computations. In fact, after 430 steps only a few grid points have been left inside the front. This leads to considerable error in the estimation of the curvature. The derivatives of the curvature are not needed to solve the governing equations; they are calculated and plotted as a reference source. As we see, the eccentricity of the front decreases as it evolves. During its collapse, the initial elliptic front tends to a circle.

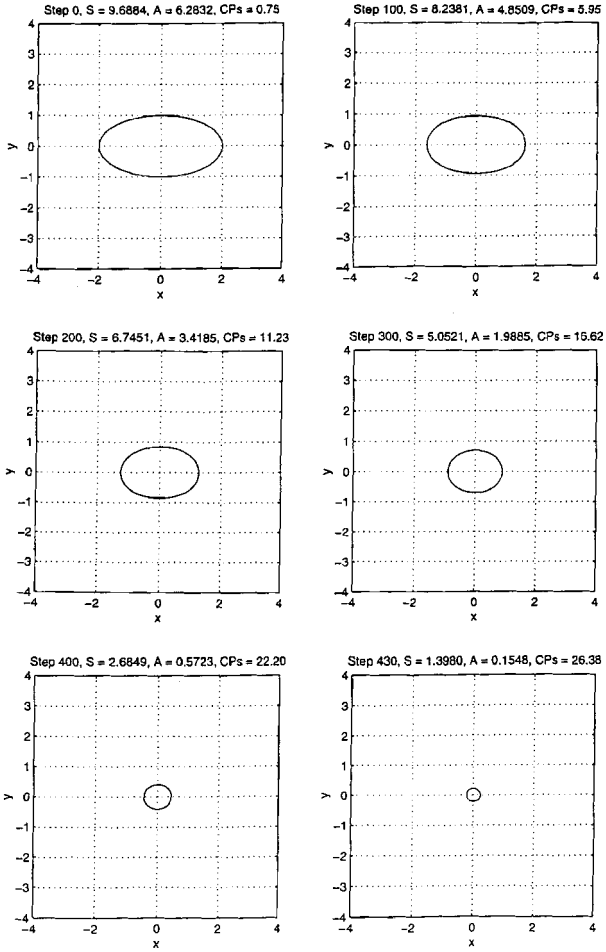


FIG. 2. Collapse of elliptic front under the second-order elliptic equation.

2.2. *Variation of Interface Length*

In this section we study the rate of change of the front length by considering the second-(2.32) and fourth-(2.30) order equations. Sethian [11] introduced the time-independent parameter p, related to the arclength $s = s(p, t)$. For example, the arclength of the initial front or the arclength of the reference configuration may be used as parameter p. He considered the metric $g(p, t) = ds/dp$ and derived the evolution equations for the metric and for the curvature:

$$g_t = KFg \tag{2.35}$$

$$K_t = -\frac{1}{g}\frac{\partial}{\partial p}\left(\frac{1}{g}\frac{\partial F}{\partial p}\right) - K^2 F. \tag{2.36}$$

Note that

$$\frac{\partial}{\partial s} = \frac{\partial}{\partial p}\frac{dp}{ds} = \frac{1}{g(p,t)} \cdot \frac{\partial}{\partial p}. \tag{2.37}$$

Then Eq. (2.36) may be rewritten in terms of the arclength s as

$$K_t = -F_{ss} - K^2 F, \tag{2.38}$$

where both curvature and speed function depend on the present arclength coordinate. Equation (2.35) yields the perimeter variation rule:

$$\frac{d}{dt}\oint ds = \frac{d}{dt}\oint g(p,t)\,dp = \oint g_t(p,t)\,dp = \oint KFg\,dp = \oint KF\,ds. \tag{2.39}$$

Now we apply this rule to two distinct speed functions of the interface.

1. The speed function is proportional to the curvature.

$$F = -AK, \quad A = const, \quad A > 0. \qquad \frac{d}{dt}\oint ds = -A\oint K^2\,ds. \tag{2.40}$$

The perimeter decreases for all time.

2. The speed function is proportional to the second derivative of the curvature with respect to the front arclength.

$$F = BK_{ss}, \quad B = const, \quad B > 0$$
$$\frac{d}{dt}\oint ds = B\oint K_{ss}K\,ds = B\oint K\,dK_s = -B\oint K_s\,dK = -B\oint K_s^2\,ds. \tag{2.41}$$

The perimeter decreases, provided the curvature is not constant along the interface. However, in the previous section it was proved that for such a speed function the confined area remains invariant. This means that any initial shape of the front tends to a circle, whose curvature $K = const$.

Next we derive the second-order terms for the interface length variation. We start with two general remarks.

Remark 1: transport theorem. Consider an arbitrary function $H = H(s,t)$ which depends on time and the arclength coordinate of the front. Note that this coordinate is time-dependent by itself. $\frac{d}{dt}\oint H(s,t)\,ds$ calculates the time derivative of the integral of H over the changing interface contour. So far, we cannot interchange the time differentiation and integration operations because the integration path depends on time. To make such a commutation possible, we switch to the time-independent parameter p defined by the metric g,

$$\frac{d}{dt}\oint H(s,t)\,ds = \frac{d}{dt}\oint H(p,t)\frac{ds}{dp}\,dp = \frac{d}{dt}\oint H(p,t)g(p,t)\,dp$$
$$= \oint H_t(p,t)\,g(p,t)\,dp + \oint H(p,t)g_t(p,t)\,dp, \tag{2.42}$$

where g_t is defined by the metric evolution rule (2.35). Therefore

$$\frac{d}{dt}\oint H\,ds = \oint H_t\,ds + \oint HKF\,ds. \tag{2.43}$$

Equation (2.43) is an analog of the Reynolds Transport Theorem in continuum mechanics [13].

Remark 2: commutation rule. Consider a mixed space–time derivative and commute the order of differentiation:

$$H_{st} = \frac{\partial H_s}{\partial t} = \frac{\partial}{\partial t}\frac{\partial H}{\partial s} = \frac{\partial}{\partial t}\frac{\partial H/\partial p}{\partial s/\partial p} = \frac{\partial}{\partial t}\frac{H_p}{g} = \frac{H_{pt}g - H_p g_t}{g^2}$$
$$= \frac{H_{tp} - H_p KF}{g} = \frac{1}{g}\frac{\partial H_t}{\partial p} - \frac{KF}{g}\frac{\partial H}{\partial p} = H_{ts} - KFH_s. \tag{2.44}$$

Note that the order interchange for the differentiation with mixed derivatives is allowed for independent variables only.

We return to our discussion on second-order terms. Due to Eq. (2.39),

$$\frac{d^2}{dt^2}\oint ds = \frac{d}{dt}\oint KF\,ds = \oint K_t F\,ds + \oint KF_t\,ds + \oint K^2F^2\,ds, \tag{2.45}$$

where rule (2.43) is used. K_t is defined by the curvature evolution law (2.38). Thus, we obtain

$$\frac{d^2}{dt^2}\oint ds = -\oint FF_{ss}\,ds + \oint KF_t\,ds = \oint F_s^2\,ds + \oint KF_t\,ds. \tag{2.46}$$

Now we consider two specific cases for the front normal velocity F.

1. The velocity is proportional to the curvature, Eq. (2.40):

$$\frac{d^2}{dt^2}\oint ds = 2A^2\oint K_s^2\,ds - A^2\oint K^4\,ds. \tag{2.47}$$

2. The velocity is proportional to the second derivative of the curvature, Eq. (2.41):

$$\frac{d}{dt^2}\oint ds = B^2\oint K_{sss}^2\,ds + B\oint KK_{sst}\,ds. \tag{2.48}$$

With the use of the commutation rule (2.44), Eq. (2.48) becomes

$$\frac{d^2}{dt^2}\oint ds = 2B^2\oint K_{sss}^2\,ds - 2B^2\oint K^2K_{ss}^2\,ds - B^2/3\oint K_s^4\,ds. \tag{2.49}$$

By expanding the interface length in a Taylor series and neglecting the high-order terms,

$$\oint ds(t+\Delta t) = \oint ds + \frac{d}{dt}\oint ds\Delta t + \frac{d^2}{dt^2}\oint ds\frac{\Delta t^2}{2} + O(\Delta t^3). \tag{2.50}$$

Various computer simulations were carried out for both second-order and fourth-order governing equations to analyze the variation of the interface length. The linear term in Eq. (2.50) appeared to have a reasonable degree of accuracy in the numerical simulations. However, this was not true for the second-order term. "Central differences in space and Runge–Kutta 2 in time" was the chosen scheme. The accuracy of the central differences is of second order. The time step was taken to be $\Delta = \lambda h^2$ for the second-order governing equation and $\Delta t = \lambda h^4$ for the fourth-order equation, where $h = \min(\Delta x, \Delta y)$ is the resolution of the spatial grid. This scheme is a good representative for studying the dynamics of moving fronts, but its accuracy is not sufficient to estimate the second-order term in (2.50).

3. FINITE DIFFERENCE FORMULATION

3.1. *Runge–Kutta Integration Scheme*

At this point our goal is to solve the set (2.29) numerically for a specified initial interface of the level function and for specified boundary conditions for the voltage. The principal time-dependent equation (2.2) describes the evolution of the level function. It is discretized by finite differences in space and by Runge–Kutta in time. The auxiliary equation (1.5) is time-independent since it does not include time derivatives. Recall, however, that the electrical conductivity $k(x, y)$ varies in time through the front motion. Although $k(x, y)$ is time-dependent, this is a known preset function at any fixed time. The elliptic equation (1.5) is solved repeatedly by finite differences. The difference scheme is discussed in Section 3.5. For now we assume that the nodal values of the voltage U are established at all times. Turning back to Eq. (2.2), $U(x, y, t)$ should be considered as a given external function. According to Eq. (2.13), U_{ss} requires evaluation of the first and second derivatives of U and ϕ. Those are calculated by central differences (applying the standard formulas of second-order accuracy), except for the points on the contour of the computational box, where the one-sided differences are used. The second derivative of the curvature with respect to arclength K_{ss} includes the fourth-order derivatives of the level function ϕ. However, due to numerical considerations, it is more convenient first to calculate the curvature K by applying second differences to the level function, and then to establish K_{ss}. And this is accomplished by applying second differences to the curvature.

Once the differential operators in the space (x, y) have been replaced by the corresponding difference operators, the evolution PDE that is described by Eq. (2.2) is transformed into a set of ODEs:

$$\frac{d\phi_i}{dt} = f_i(\phi, \boldsymbol{U}). \tag{3.1}$$

Each equation of the set (3.1) corresponds to a specific node i of the grid. Vectors ϕ and U in (3.1) represent the node values of the level function and voltage, respectively, according to the difference operators. Recall that $\boldsymbol{U} = \boldsymbol{U}(\phi)$, and therefore (3.1) may be presented as $d\phi_i/dt = f_i(\phi)$. This is a standard set of ODEs, and Runge–Kutta integration schemes are appropriate. The schemes of order 2, 3, and 4 were tested and produced the proper results.

Note that the distance function ϕ has to be reinitialized n times and the voltage distribution PDE (1.5) has to be solved n times per time step, where n is the order of the Runge–Kutta scheme. Therefore, the high-order schemes are computationally expensive. On the other hand, a small time step $\Delta t = \lambda h^4$ is dictated by the stability requirements. The spatial resolution $h = \min(\Delta x, \Delta y)$ should be sufficiently small to provide accurate calculations for the fourth derivatives of the moderately varying function. Consequently, Δt is usually small enough that even Runge–Kutta order 2 yields an acceptable accuracy. Therefore, the second-order scheme that is being used is:

$$\boldsymbol{k}_1 = \boldsymbol{f}(\phi, t); \quad \boldsymbol{k}_2 = f(\phi + \Delta t k_1); \quad \phi(t + \Delta t) = \phi + \frac{\Delta t}{2}(\boldsymbol{k}_1 + \boldsymbol{k}_2). \tag{3.2}$$

This scheme is accurate to order Δt^2, and the discrepancy (error generated in one step) is proportional to Δt^3. Numerical tests revealed that λ should not exceed $1/8$, or else stability is lost. It is interesting to note that this critical value of λ proved to be approximately the same for any order of Runge–Kutta scheme.

Below we present the computational steps to integrate the evolution equation (2.2). Although the algorithm assumes a Runge–Kutta order 2 integration scheme, the same steps can be also applied to the Runge–Kutta scheme of any order, provided that the obvious modifications are inserted.

1. The input data is a set of generating points (x, y) which describe the initial configuration of the interface curve. These points do not necessarily coincide with the rectangular grid points.
2. Apply the spline technique to restore the analytic parametric relationships for the interface curve: $x = f_x(r), y = f_y(r); 0 \le r \le 1$.
3. If necessary, redistribute the generating points in a uniform manner, such that the arclength between any two successive points is the same.
4. Following Eqs. (2.3–2.6), we establish the distance function at each node. We use the Newton iterative procedure to solve the nonlinear equation and the *golden section* minimization technique when the Newton procedure fails to converge.
5. Apply the difference approximations and calculate at each node the derivatives of the distance function and the curvature of the level line.
6. Apply the difference approximations and calculate at each node the derivatives of the curvature K_s and K_{ss} with respect to the arclength of the level line.
7. Using the node values of the distance function, calculate for these points the electrical conductivity. In other words, if the sign of the distance function is negative, then we are inside the void. If it is positive, we are outside the void and the conductivity of the material should be used.
8. Solve the elliptic equation (1.5) for the appropriate boundary conditions and obtain the nodal values of the voltage distribution.
9. Estimate the total resistance of the interconnect line, Eq. (3.23).
10. Apply the difference approximations and use Eqs. (2.11) and (2.13) to calculate the node values of the voltage derivatives U_s and U_{ss} with respect to the arclength of the level line.
11. Combine K_{ss} with αU_{ss} and reverse the sign, producing the node values for k_1 in the Runge–Kutta scheme, Eq. (3.2).
12. Calculate the updated node values of the level function: $\phi^{upd} = \phi + \Delta t \boldsymbol{k}_1$.
13. Find the interface curve points of the updated level function by solving the equation $\phi^{upd}(x, y) = 0$ numerically. The previous configuration of the front is used as an initial approximation.
14. Using the set of points obtained in the previous item, repeat steps 2–10 for the updated level function.
15. Combine K_{ss} with αU_{ss} and reverse the sign, producing the node values of k_2 in the Runge–Kutta scheme, Eq. (3.2).
16. Calculate the node values of the level function at $t + \Delta t$ by applying the last formula of set (3.2). Now we have $\phi = \phi(t + \Delta t)$.
17. Solve the equation $\phi(x, y) = 0$ numerically and find the interface curve points of the level function.
18. Apply the polynomial interpolation technique, calculate the values of the curvature and voltage and their arclength derivatives up to the second order at the interface points.
19. In the next time step, repeat steps 2–18.

Remark. Steps 13 and 17 require solving the equation $\phi(x, y) = 0$, i.e., finding the set of points (x, y) in the plane where the level function vanishes. Assume we do know this

set approximately. In other words, we have at our disposal a set of points (x, y) which generates a closed curve. On this set of points the level function does not necessarily vanish ($\phi(x, y) \neq 0$), but it assumes small absolute values $|\phi(x, y)|$ compared with the size of the computational box. Consider an individual point (x, y). Our goal is to find the updated values of x and y where the level function vanishes. Expand the level function into a Taylor series while neglecting the high-order terms:

$$\phi(x + \Delta x, y + \Delta y) = \phi(x, y) + \phi_x(x, y) \cdot \Delta x + \phi_y(x, y) \cdot \Delta y \approx 0. \tag{3.3}$$

We search for the point of the zero profile, which is the closest to the point (x, y) under consideration. Therefore, we move in the gradient direction for the negative values of the level function, and move in the counter-gradient direction for the positive values

$$\Delta x = w\phi_x(x, y) \quad \Delta y = w\phi_y(x, y), \tag{3.4}$$

where the sign of w should be opposite to that of $\phi(x, y)$. Substitute (3.4) into (3.3) and take into account the unit gradient length of the distance function. We obtain $w = -\phi(x, y)$. This leads to an iterative procedure which yields a point of the zero profile:

$$\begin{aligned} x^{(n+1)} &= x^{(n)} - \phi\left[x^{(n)}, y^{(n)}\right] \cdot \phi_x\left[x^{(n)}, y^{(n)}\right] \\ y^{(n+1)} &= y^{(n)} - \phi\left[x^{(n)}, y^{(n)}\right] \cdot \phi_y\left[x^{(n)}, y^{(n)}\right]. \end{aligned} \tag{3.5}$$

The procedure requires a bipolynomial interpolation to establish the values of the level function and its derivatives at an arbitrary nongrid point.

3.2. *Forward and Backward Difference Approximation for the Level Function Derivatives*

When Eq. (2.2) is solved numerically, the boundaries of the rectangular computational box should be far away from the front contour (as compared to the size of the area confined by the front). Otherwise, closeness to the boundary will affect the front motion. And conversely, if the void interface is far away from the box bounds, the boundary conditions do not matter. However, this does not mean that any boundary conditions can be applied. We realized that improper boundary conditions yield poor results, and we believe that for the closed front evolution problem, the boundary conditions should have no influence. Therefore, we need to estimate the spatial derivatives of the level function not only at the internal points of the box but also on the box contour. Central differences cannot be applied to the contour points since they require ghost values outside the box. These values, in turn, cannot be established because of the lack of boundary conditions. Therefore, we use forward and backward differences. As we have already mentioned, although the governing equation (2.2) is of fourth order in space, we do not approximate the fourth-order derivatives by differences. Instead, we apply the second differences twice. Recall that the errors of the central difference approximations adopted are of order h^2. Consequently, we have to establish the forward and backward formulas with the same order of accuracy. For an equally spaced grid x_0, x_1, x_2, x_3, the first and second derivative approximations are

$$\begin{aligned} f'(x_0) &= \frac{-3f_0 + 4f_1 - f_2}{2\Delta x} + O(\Delta x^2) \\ f''(x_0) &= \frac{2f_0 - 5f_1 + 4f_2 - f_3}{\Delta x^2} + O(\Delta x^2), \end{aligned} \tag{3.6}$$

where $f_i = f(x_i) i = 0, \ldots, 3$. To derive the formula for the mixed second derivative, we apply Eq. (3.6) repeatedly. For the corner points, we apply one-sided differences only. For the noncorner boundary points, we apply central differences in the direction of the contour line, and one-sided differences for the normal direction. In particular, the mixed second derivative is obtained by applying the central and one-sided difference sequentially.

3.3. *Far Field and Near Field Considerations*

Recall that using the Runge–Kutta 2 integration scheme, we reinitialize the level function at all grid points twice per each time step. For this, we use the values of the level function obtained by integrating the governing equation to obtain the closed zero line:

$$\phi(x, y) = 0. \tag{3.7}$$

Then at each grid point, the level function is reassigned as a signed distance from the given point to the interface line. We solve the nonlinear equation (3.7) to estimate the new locations for the generating points of the interface curve. Using their old locations as the initial approximations, we proceed in the gradient or counter-gradient direction until Eq. (3.7) is satisfied with a required reasonable degree of accuracy. For this, the non-node values of the level function at the intergrid points are necessary. This, in turn, requires the two-dimensional polynomial interpolation of $\phi(x, y)$. As mentioned above, in this work fifth-order polynomials are used, which means 36 nodal values are required to estimate a non-nodal value. In each of two Cartesian directions, we need to consider three grid points to the left and three grid points to the right from the intergrid point under consideration. This requirement means that the most remote grid point to be considered is located at the distance

$$d_b = \pm 4\sqrt{\Delta x^2 + \Delta y^2} \tag{3.8}$$

from the interface line, because three grid points to the left or to the right means up to four intervals in each direction. With the interpolation scheme adopted, other grid values provide no contribution to the interface line location, and therefore the normal front velocity $F = K_{ss} + \alpha U_{ss}$ should be established for the grid points within a relatively narrow band region inside and outside the interface line, with a front velocity bandwidth of

$$-d_b \le \phi(x, y) \le d_b. \tag{3.9}$$

Thus, the second derivative of the curvature and the voltage with respect to the arclength should be calculated within the region bounded by inequality (3.9). This means that the curvature and the voltage should be calculated through a larger region that is one grid interval wider in each direction so that the curvature and voltage bandwidth are defined by

$$\frac{-5d_b}{4} \le \phi(x, y) \le \frac{5d_b}{4}. \tag{3.10}$$

The implicit multigrid computational algorithm yields the voltage distribution at all grid points, but as we see, not all of them are necessary. Since the numerical estimation of the curvature requires first and second derivatives of the distance function, Eq. (3.9) yields the bandwidth for the distance function calculation, giving a distance and function bandwidth of

$$\frac{-3d_b}{2} \le \phi(x, y) \le \frac{3d_b}{2}. \tag{3.11}$$

Note that the bandwidth for the region, where the distance function should be calculated, is defined via the distance function itself. Thus, if at each time step we recalculate or the distance function remains unchanged according to condition (3.11) alone, then the distance function will be updated only within the *initial* near field region. The near field location will not move. The grid points that belong initially to the far field region will remain outside the near band forever, and the front motion will be confined.

To avoid such a "self-clinching loop," we recalculate the distance function within the near field (Eq. 3.11) twice per each time step: half band inside the interface line and half band outside. However, once per N_f time steps the level function is updated for all grid points of the entire rectangular computational domain.

Due to the severe stability constraints of the explicit integration scheme for the fourth-order governing equation (2.29), the time step is small and proportional to the fourth power of the cell size. Consequently, the change in the position of the interface line during one time step is pretty small compared to the cell size. The generating points of the interface line at the next time step are located in close proximity to their previous positions. Therefore, $N_f = 20$ usually suffices.

Since the boundary points of the computational box should always be far away from the void interface, those points belong to the far field region. Thus, with the use of the "far field–near field" concept, estimating the derivatives at the boundaries of the computational box (using the one-sided differences) becomes unnecessary. However, we still use the one-sided difference approximations for the derivatives, in order to calculate the electric resistance of the interconnect line. Estimation of the total resistance is described below.

3.4. *The Interface Perimeter and the Confined Area*

Assume the front is given by two parametric relationships $f_x(r)$ and $f_y(r)$, which may be specified analytically or result from a spline interpolation. Let s be the arclength and r an arbitrary parameter. To obtain the area confined by a closed front, we pass from the area integral to the integral over the interface length:

$$A = \iint_A dA = \frac{1}{2} \int_r \left(f_x \frac{df_y}{dr} - f_y \frac{df_x}{dr} \right) dr. \tag{3.12}$$

$0 \le r \le 1$ is a normalized parameter which defines the limits of integration in (3.12) and (3.13). The interface length

$$s_{\max} = \oint ds = \int_r \frac{ds}{dr}\, dr = \int_r \sqrt{(df_x/dr)^2 + (df_y/dr)^2}\, dr. \tag{3.13}$$

When $f_x(r)$ and $f_y(r)$ are given by cubic splines their derivatives are easily calculated.

3.5. *Distribution of Electric Potential*

The voltage distribution is a static boundary value problem which is described by Eq. (1.5). We consider the case in which $k = k_{in} = const$ inside the closed front and $k = k_{out} = const$ outside the front. Since the distance function is negative inside the front and positive outside, we get $k = k_{in}$ for $\phi < 0$ and $k = k_{out}$ for $\phi > 0$. Formally, Eq. (1.5) requires the existence of derivatives k_x and k_y. For this purpose, we smooth out the conductivity distribution by a

hyperbolic tangent law,

$$k(x,y)=\frac{k_{out}+k_{in}}{2}+\frac{k_{out}-k_{in}}{2}\tanh\beta\phi(x,y), \tag{3.14}$$

where β is a constant adjusting parameter. In this work, we use relatively large values of β, so that $k(x,y)$ approaches k_{in} or k_{out} even at a small distance from the interface (rapid change). Once we know the grid values of the distance function, we also know the grid values of the conductivity.

The boundary conditions are $\partial U/\partial y=0$ at the horizontal boundaries of the computational box, and the voltage is preset along the vertical boundaries: $U=U^-=const$ on the left boundary and $U=U^+=const$ on the right. We discretize Eq. (1.5) by central differences with second-order accuracy:

$$\begin{aligned}\frac{\partial}{\partial x}\left(k\frac{\partial U}{\partial x}\right)&=\frac{k_{i+1/2,j}U_{i+1,j}-\left(k_{i+1/2,j}+k_{i-1/2,j}\right)U_{ij}+k_{i-1/2,j}U_{i-1,j}}{\Delta x^2}\\ \frac{\partial}{\partial y}\left(k\frac{\partial U}{\partial y}\right)&=\frac{k_{i,j+1/2}U_{i,j+1}-\left(k_{i,j+1/2}+k_{i,j-1/2}\right)U_{ij}+k_{i,j-1/2}U_{i,j-1}}{\Delta y^2}.\end{aligned} \tag{3.15}$$

To calculate the local truncation error we expand $k(x,y)$ and $U(x,y)$ in (3.15) into Taylor series in the neighborhood of the central node (i,j) while taking into consideration Eq. (1.5):

$$\begin{aligned}E_h=&\left(\frac{\partial^3k}{\partial x^3}\frac{\partial U}{\partial x}+3\frac{\partial^2k}{\partial x^2}\frac{\partial^2U}{\partial x^2}+4\frac{\partial k}{\partial x}\frac{\partial^3U}{\partial x^3}+2k\frac{\partial^4U}{\partial x^4}\right)\frac{\Delta x^2}{24}\\&+\left(\frac{\partial^3k}{\partial y^3}\frac{\partial U}{\partial y}+3\frac{\partial^2k}{\partial y^2}\frac{\partial^2U}{\partial y^2}+4\frac{\partial k}{\partial y}\frac{\partial^3U}{\partial y^3}+2k\frac{\partial^4U}{\partial y^4}\right)\frac{\Delta y^2}{24}.\end{aligned} \tag{3.16}$$

The disadvantage of scheme (3.15) is that it requires the conductivity values in the middle of the interval between grid points: $k_{i\pm1/2,j}$ and $k_{i,j\pm1/2}$. These values may be replaced by averaging two neighbor grid points' conductivities:

$$k_{l+1/2}=\frac{k_l+k_{l+1}}{2}+O(h^2). \tag{3.17}$$

Generally, the replacement in Eq. (3.17) cannot be applied for the central differences, as the second differences are $O(h^2)$ before division by h^2. However, for this case such a replacement is justified due to the symmetry properties of the scheme (3.15), as shown below. Therefore, the difference equation becomes

$$\begin{aligned}&\frac{1}{\Delta x^2}[(k_{i+1,j}+k_{ij})U_{i+1,j}-(k_{i+1,j}+2k_{ij}+k_{i-1,j})U_{ij}+(k_{ij}+k_{i-1,j})U_{i-1,j}]+\frac{1}{\Delta y^2}\\&\times[(k_{i,j+1}+k_{ij})U_{i,j+1}-(k_{i,j+1}+2k_{ij}+k_{i,j-1})U_{ij}+(k_{ij}+k_{i,j-1})U_{i,j-1}]=0.\end{aligned} \tag{3.18}$$

To prove the validity of substitution (3.17), we estimate the local truncation error of the difference scheme (3.18)

$$\begin{aligned}E_i=&\left(\frac{\partial^3k}{\partial x^3}\frac{\partial U}{\partial x}+3\frac{\partial^2k}{\partial x^2}\frac{\partial^2U}{\partial x^2}+2\frac{\partial k}{\partial x}\frac{\partial^3U}{\partial x^3}+k\frac{\partial^4U}{\partial x^4}\right)\frac{\Delta x^2}{12}\\&+\left(2\frac{\partial^3k}{\partial y^3}\frac{\partial U}{\partial y}+3\frac{\partial^2k}{\partial y^2}\frac{\partial^2U}{\partial y^2}+2\frac{\partial k}{\partial y}\frac{\partial^3U}{\partial y^3}+k\frac{\partial^4U}{\partial y^4}\right)\frac{\Delta y^2}{12}.\end{aligned} \tag{3.19}$$

We see that the local truncation error is $O(h^2)$, i.e., the error order is the same as in the original scheme (3.15) before substitution (3.17) was applied. With a rearrangement of terms, Eq. (3.18) becomes

$$\frac{k_{i+1,j}+k_{ij}}{\Delta x^2}U_{i+1,j}+\frac{k_{ij}+k_{i-1,j}}{\Delta x^2}U_{i-1,j}+\frac{k_{i,j+1}+k_{ij}}{\Delta y^2}U_{i,j+1}+\frac{k_{ij}+k_{i,j-1}}{\Delta y^2}U_{i,j-1}$$
$$-\left(\frac{k_{i+1,j}+2k_{ij}+k_{i-1,j}}{\Delta x^2}+\frac{k_{i,j+1}+2k_{ij}+k_{i,j-1}}{\Delta y^2}\right)U_{ij}=0. \tag{3.20}$$

We now consider the derivative boundary conditions. Assume that $j=0$ and $j=M$ are the horizontal boundaries of the computational box. If we require $\partial U/\partial y=0$ at $j=0$ and $j=M$, the standard central difference yields the ghost values

$$U_{i,-1}=U_{i,0},\quad U_{i,M+1}=U_{iM}, \tag{3.21}$$

where the boundary is assumed to pass midway between grid points. The boundary condition (3.21) is substituted into the difference equation (3.20) so that the ghost values $U_{i,-1}$ and $U_{i,M+1}$ are eliminated.

Let $i=0$ and $i=N$ be vertical boundaries of the computational box. Then, the Dirichlet boundary conditions are

$$U_{0j}=U^-,\quad U_{Nj}=U^+. \tag{3.22}$$

By solving set (3.20) with boundary conditions (3.21) and (3.22), we obtain the voltage at each node. The coefficient matrix is a band matrix with a half-band of width $N+1$ or $M+1$. However, the band is sparse: each row contains only five nonzero coefficients. The linear set is solved using an efficient multigrid iterative procedure. Once the grid values of the voltage are established, the estimated total electric resistance of the interconnect is

$$R=\frac{U^+-U^-}{k_{out}\int_h \frac{\partial U}{\partial x}\,dy}, \tag{3.23}$$

where U^+-U^- is the external voltage applied to the interconnect; k_{out} is the specific electric conductivity of the material outside the front. The integral can be taken along any line $x=const$, located entirely outside the interface. We use both left and right vertical bounds. The results, of course, should be the same. The coincidence of U^+-U^- along any line is only an indication of some kind of conservation form of the numerical scheme. h is the length of the integration line, from the lower bound to the upper bound of the computational box. The resistance varies as the front evolves and it is independent of the applied voltage. We use Ohm's law to calculate the resistance.

Three-dimensional plots which describe numerical estimations for the voltage distribution are presented in Figs. 3 and 4.

Different ratios $k_{in}=k_{out}$ were considered. For $k_{in}\gg k_{out}$ we deal with a metallic island inside the front, surrounded by a fairly poor conductor. Another case, $k_{in}\ll k_{out}$, describes a void inside a good metallic conductor. Even for a void, $k_{in}0$, due to the underlayer's conductivity. A grid of size 100×100 was used for the examples.

Figure 3 presents the distribution of the conductivity k and voltage U for the circular irregularity. The circular superconductive island is considered in Fig. 3a, the circular void in Fig. 3b. Voltage inside the island and along its boundary is almost constant.

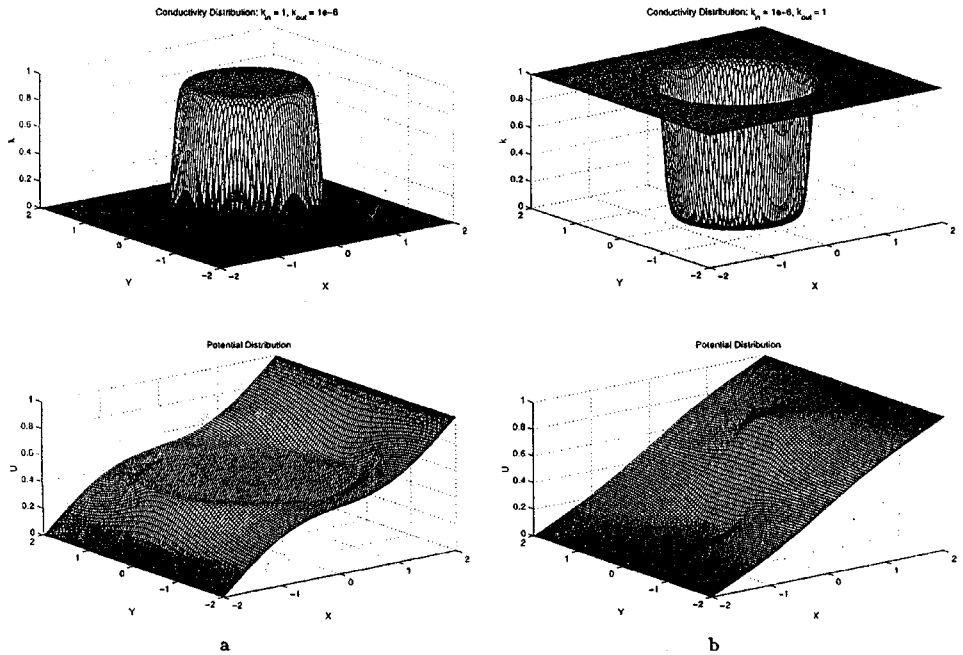


FIG. 3. Distribution of electric potential through the rectangular plate with a circular irregularity: (a) circular superconductive island; (b) circular void.

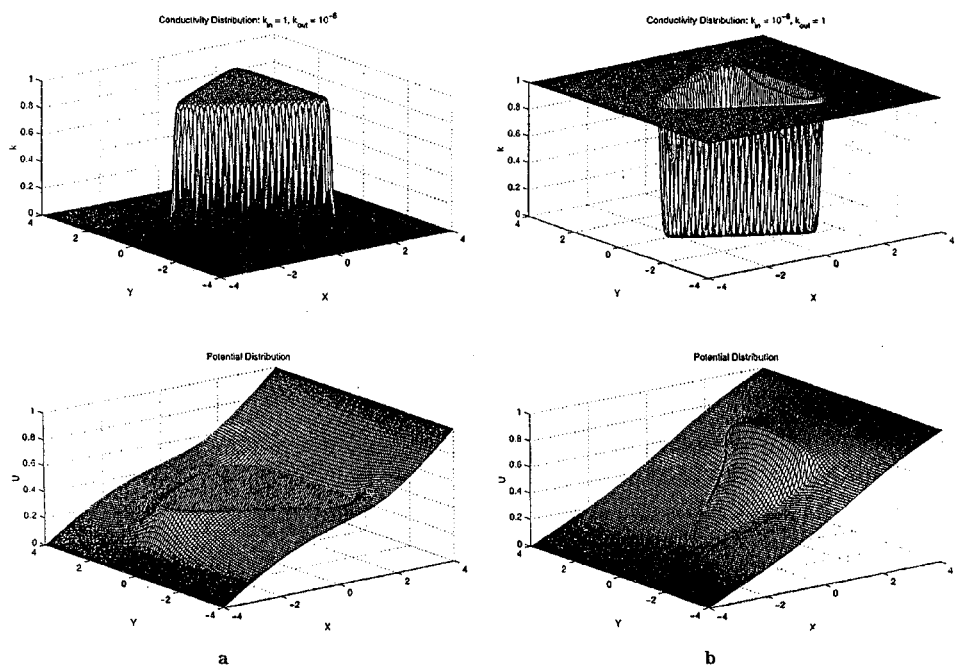


FIG. 4. Distribution of electric potential through the rectangular plate with a triangular irregularity: (a) triangular superconductive island; (b) triangular void.

Triangular island and triangular void topologies are presented in Figs. 4a and 4b, respectively. The triangle is equilateral, and its axis of symmetry coincides with the direction of the electric field at infinity (far away from the triangle).

4. ELLIPTIC BOUNDARY VALUE SOLVER

The most computationally intensive part of the algorithm described in Section 3 is step 8—solution of the elliptic boundary value problem at each time level. The finite difference discretization produces a linear system of algebraic equations. The associated matrix is banded and very sparse—only five nonzero coefficients per row—but the bandwidth is $O(\sqrt{n})$, where n is the total number of unknowns. Due to the high resolutions required, a direct solver is thus far too costly, and we need to resort to an efficient iterative solver that will produce a solution with the required accuracy in just $O(n)$ operations. Multigrid methods provide us with the necessary performance.

4.1. *Multigrid Techniques*

Efficient multigrid solvers for elliptic boundary value problems were developed in the 1970s (though conceived earlier), and the first comprehensive account of this approach and related algorithms appeared in [18]. The field then quickly developed and branched into specialized methods, "automatic" solvers that could deal with anisotropic and discontinuous coefficients, algebraic methods, applications to systems and to nonelliptic problems, and, later, nonPDE solvers. (For a basic introduction see [21]. A detailed exposition and advanced concepts appear in [24] and [19].)

The essential distinction between the various multigrid approaches for elliptic boundary value problems is in the choice of the intergrid transfers and the coarse-grid operator. For problems with smooth isotropic coefficients very simple choices perform most efficiently. However, in the present application, where the coefficients are discontinuous in certain regions of the domain, with jumps of several orders of magnitude, we must resort to specialized methods. This difficulty was first studied in [17]. These ideas were then put together in [22] in the form of a "black box" multigrid solver. This is the algorithm we employ here.

4.2. *The Black Box Multigrid (BBMG) Algorithm*

The black box multigrid (BBMG) algorithm is an automatic approach to multigrid solution of discretized elliptic boundary value problems. This means that the solver is only given the fine-grid data and the definition of the coarse grids (as well as an initial approximation to the solution), and it generates the intergrid transfers and coarse-grid operators automatically. The underlying assumption is that the coefficients of the elliptic operator may be discontinuous but the fluxes are smooth. Hence, this solver is very suitable for our present problem.

The BBMG method employs so-called Galerkin coarsening. This means that the coarse-grid operator is defined by

$$L^H = I_h^H L^h I_H^h. \tag{4.1}$$

(Note that this definition "makes sense," as it can be thought of as transferring the coarse-grid solution to the fine grid, applying the fine-grid operator, and transferring the result back

to the coarse grid. Thus, L^H is indeed a coarse-grid approximation to L^h.) Furthermore, the prolongation and restriction operators are chosen to be the transpose of each other:

$$I_h^H = \left(I_H^h\right)^T. \tag{4.2}$$

Given (4.1) and (4.2), we find that once we choose a prolongation operator, the coarse-grid correction process will be defined uniquely. The idea of the BBMG algorithm is to use the fine-grid difference operator in the definition of the prolongation operator, I_H^h. We describe it here for the 2D case, using the notation of [23] (see [22, 23] for more details). A vertex-centered coarsening is used, and the coarse-grid mesh is uniform with mesh intervals that are twice the size of those of the fine grid. We denote fine-grid and coarse-grid indices, respectively, by (IF, JF) and (IC, JC). Given the choice of coarse grid, roughly one quarter of the fine-grid points coincide with coarse-grid points. At these points we simply define

$$\left(I_H^h v^H\right)_{IF,JF} = v^H_{IC,JC}.$$

But for the remaining fine-grid points, the fine-grid template (stencil) is employed in the prolongation. Suppose that the fine-grid point $(IF+1, JF)$ lies midway between coarse-grid points (IC, JC) and $(IC+1, JC)$, and that L^h has at $(IF+1, JF)$ the template[3]

$$\begin{bmatrix} -NW & -N & -NE \\ -W & C & -E \\ -SW & -S & -SE \end{bmatrix}. \tag{4.3}$$

Then we define

$$\left(I_H^h v^H\right)_{IF+1,JF} = \frac{(NW+W+SW)v^H_{IC,JC} + (NE+E+SE)v^H_{IC+1,JC}}{C-N-S}. \tag{4.4}$$

Here, the columns of (4.3) have been summed to average out the vertical dependence. A similar formula holds for the fine-grid point $(IF, JF+1)$ that lies midway between coarse-grid points (IC, JC) and $(IC, JC+1)$ The fine-grid values at the remaining points, $(IF+1, JF+1)$, are now determined so as to satisfy the equation

$$\left(L^h I_H^h v^H\right)_{IF+1,JF+1} = 0. \tag{4.5}$$

4.3. *Application of the BBMG Algorithm to the Time-Dependent Problem*

We have not yet discussed how we choose our initial approximation to the solution at each new time level. We followed the procedure of [20], which defines a so-called modified F-cycle, which applies the well-known full multigrid (FMG) algorithm while exploiting the solution of the previous time level to obtain a good first estimate. This is equivalent to performing an FMG algorithm for only the *change* in the solution over each time level. It avoids excessive accumulation of error due to the truncation of the iterative process, which is an essential property, given the very large number of time levels. The modified F-cycle is now described.

[3] This means that the equation at node $(IF+1, JF)$ is

$$-NWu^h_{IF,JF+1} - Nu^h_{IF+1,JF+1} - NEu^h_{IF+2,JF+1} \cdots - SEu^h_{IF+2,JF-1} = f^h_{IF+1,JF}.$$

1. Beginning with the fine-grid solution of the previous time level, transfer residuals to the right-hand sides of progressively coarser grids, until right-hand sides have been defined on all the grids.

2. Solve the problem on the coarsest grid (directly or by repeated relaxation). Prolong the solution to the second-coarsest grid.

3. Perform one V-cycle on the second-coarsest grid, and prolong the resulting solution to the third-coarsest grid.

4. Continue this process until the finest grid is reached, and perform one (or more) V-cycles on the finest grid.

In all our cycles we employed the BBMG solver with $\nu_{pre} = 2$ and $\nu_{post} = 1$.

4.4. *Planned Optimization*

Although the multigrid algorithm is optimal as an elliptic boundary value problem solver, the fact that it is employed at every time level makes it still the most computation-intensive part of the algorithm. We therefore plan improvements which we expect will reduce the overall cost dramatically.

- At each time level we effectively solve for the increment in $\boldsymbol{U}$. The physical changes in $\boldsymbol{U}$ are substantial near the interface, where the coefficient k in (1.5) changes discontinuously. But away from this region the change per time level is slight (given the small time steps.) More important, this change is extremely smooth except near the interface. Therefore it can be approximated well on a far coarser mesh than that used near the interface. This can be exploited by employing local refinement techniques. However, such techniques are relatively quite complicated to implement in the framework of BBMG algorithms. Hence we shall instead only employ local processing (near the interface) during most of the integration, while freezing $\boldsymbol{U}$ far away from the interface. This procedure is justified by the fact that the time-step size is controlled by the time-dependent equation (2.2), and it is far smaller than necessary for accuracy of $\boldsymbol{U}$. Only once every few time steps do we actually need to employ the full BBMG solver.
- A second approach for the optimization of the solver is to employ a semi-implicit time-stepping approach. The principal part of (2.2) is the biharmonic operator (as seen by (2.29)). This term can be treated implicitly by employing a fairly standard multigrid algorithm. This would allow us to use much smaller time steps. We can also treat other terms implicitly, including nonlinear terms (employing the nonlinear full approximation scheme multigrid method). We expect these changes to increase the computational efficiency very significantly.

5. SIMULATION RESULTS

In this section, we present the results for two sets of simulations. The first set displays interface motion under the surface diffusion only. The second set combines the surface diffusion with electromigration forces. Different initial configurations are considered [1].

All the simulations were executed on one processor from a DEC AlphaServer 8800-5/300, with 8 CPUs, 300 MHz, 256 MB of memory, and a DEC-UNIX operating system.

5.1. *Motion under Surface Diffusion*

In this case, the steady-state configuration of the front is circular, independent of its initial shape. The area confined by the closed front remains constant for the duration of the motion.

Elliptic initial front. Consider an elliptic front

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \tag{5.1}$$

with the semi-axes $a = 2$ and $b = 1$. Equation (5.1) is presented in the parametric form as

$$f_x = a\cos r, \quad f_y = b\sin r, \quad 0 \le r \le 2\pi. \tag{5.2}$$

The initial curvature is

$$K(r) = \frac{f_y'' f_x' - f_x'' f_y'}{\left(f_x'^2 + f_y'^2\right)^{3/2}} = \frac{ab}{(a^2\sin^2 r + b^2\cos^2 r)^{3/2}} > 0, \tag{5.3}$$

where prime denotes derivative with respect to parameter r. As we see, the ellipse curvature is always positive: $K_{\max} = a/b^2$, $K_{\min} = b/a^2$. The area confined is πab, and the arclength is

$$s(r) = \int_0^r \sqrt{f_x'^2 + f_y'^2}\, dz = \int_0^r \sqrt{a^2\sin^2 z + b^2\cos z}\, dz = a[E(\varepsilon) - E(\pi/2 - r, \varepsilon)], \tag{5.4}$$

where $\varepsilon = \sqrt{a^2 - b^2}/a$ is the eccentricity of the ellipse and E is the complete elliptic integral of second kind. The entire interface length is $4aE(\varepsilon)$. Derivatives of the curvature with respect to arclength can also be presented as a function of the parameter r. The first derivative is

$$K_s(r) = \frac{dK/dr}{ds/dr} = -\frac{3/2\, ab(a^2 - b^2)\sin 2r}{(a^2\sin^2 r + b^2\cos^2 r)^{3/2}} \tag{5.5}$$

$$|K_s|_{\max} = \frac{4\sqrt{6}\, ab^2(a^2 - b^2)(a^2 - b^2/3)}{\left[(a^2 + b^2)\sqrt{a^2 - b^2/3} - (a^2 - b^2)^{3/2}\right]^3}. \tag{5.6}$$

The second derivative is

$$K_{ss} = -\frac{3ab(a^2 - b^2)}{2}\left[\frac{2\cos 2r}{(a^2\sin^2 r + b^2\cos^2 r)^{7/2}} - \frac{3(a^2 - b^2)\sin^2 2r}{(a^2\sin^2 r + b^2\cos^2 r)^{9/2}}\right]. \tag{5.7}$$

The minimum (negative) value is

$$K_{ss}^{\min} = -3\frac{a(a^2 - b^2)}{b^6}. \tag{5.8}$$

The maximum positive value of K_{ss} is reached at $r = r^*$,

$$\cos 2r^* = \frac{7\sqrt{(a^2 - b^2)^2 + 116/245\, a^2b^2} - 3(a^2 + b^2)}{4(a^2 - b^2)}, \tag{5.9}$$

provided the eccentricity ε exceeds $2/\sqrt{19}$. For the stated values of semi-axes, the relationships for K, K_s, and K_{ss} are plotted versus arclength in Fig. 5. Results of the numerical simulation are presented in Fig. 6.

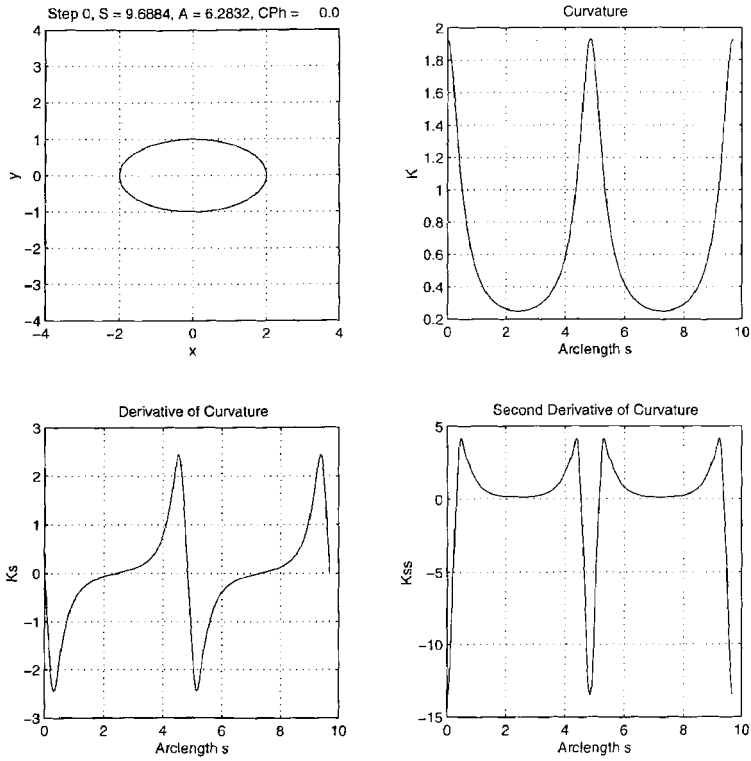


FIG. 5. Elliptic front, its curvature, and derivatives of curvature.

Note that the interface contour reaches a steady state after 30,000 steps or 1.4 h CPU time. Recall that the PDE under consideration is of fourth order in space. For the second-order PDE with the same initial configuration and the same 60×60 grid, it took less than half a minute to solve the problem; see Fig. 2. The interface configuration is set by 500 generating points. For the spatial resolution $h = \min(\Delta x; \Delta y)$, the time step is bounded by stability requirements: $\Delta t = \lambda h^2$ for the second-order equation and $\Delta t = \lambda h^4$ for the fourth-order equation. We used $\lambda = 1/8$ for all two-dimensional simulations to obtain stable results. Larger values of λ led to instability.

As part of validating our program we also analyzed the evolution of an ellipse with semi-axis under diffusion only with a semi-axis ratio of 10, in order to compare with [28] (Fig. 3, p. 1483). Our contours are drawn at the same physical problem times (see Fig. 7). It was done on a 148×148 grid, and the interface was specified by 400 generating points. About 1 million steps were required to achieve steady-state.

Triangle with rounded corners. The polygons that were considered in this study had rounded corners. The rounded equilateral triangle is shown in Fig. 8 and can be specified parametrically as

$$f_x = (2 + \cos r) \sin r, \quad f_y = -(2 - \cos r) \cos r - 1/2, \quad 0 \le r < 2\pi. \tag{5.10}$$

The interface curvature is

$$K(r) = \frac{2(1 - \cos 3r)}{(5 + 4 \cos 3r)^{3/2}}, \quad 0 \le K \le 4. \tag{5.11}$$

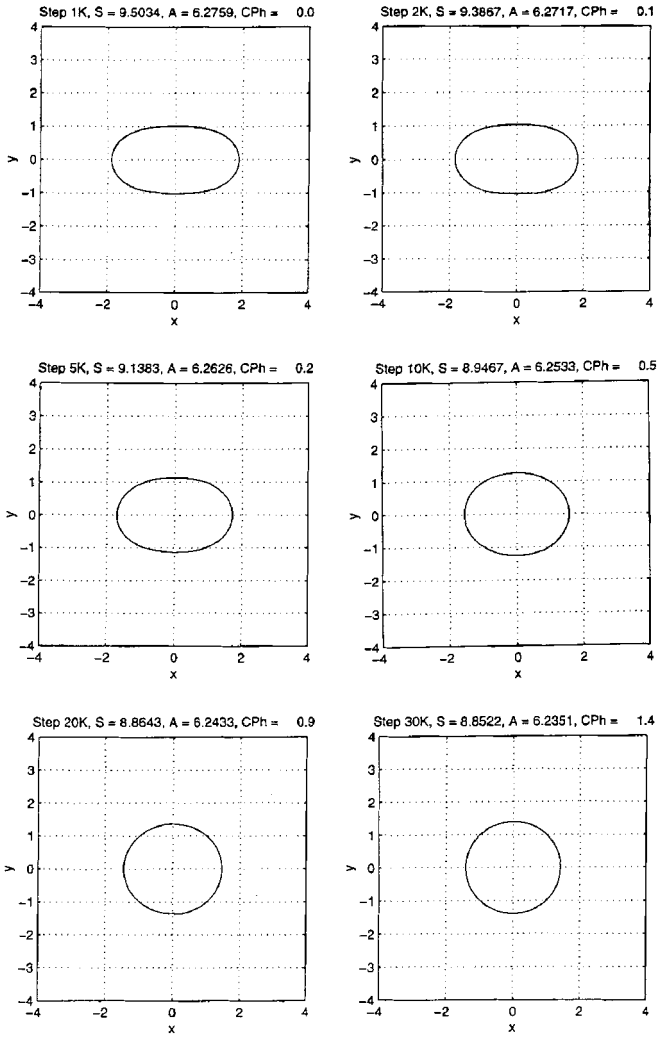


FIG. 6. Evolution of initially elliptic front under surface diffusion forces.

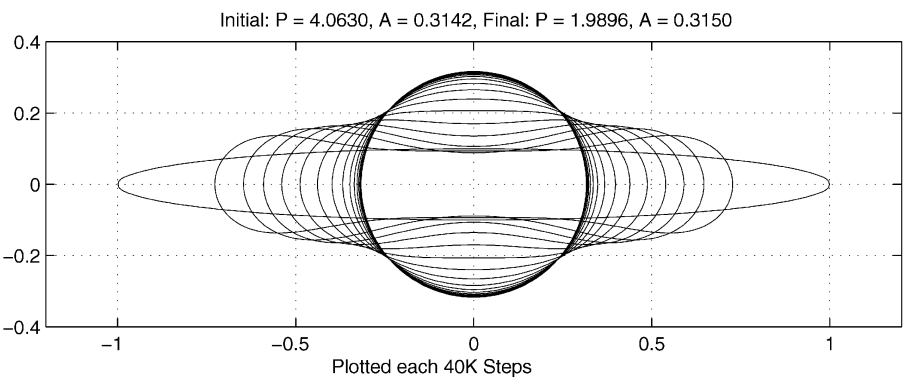


FIG. 7. Evolution of highly eccentric elliptic front under surface diffusion forces.

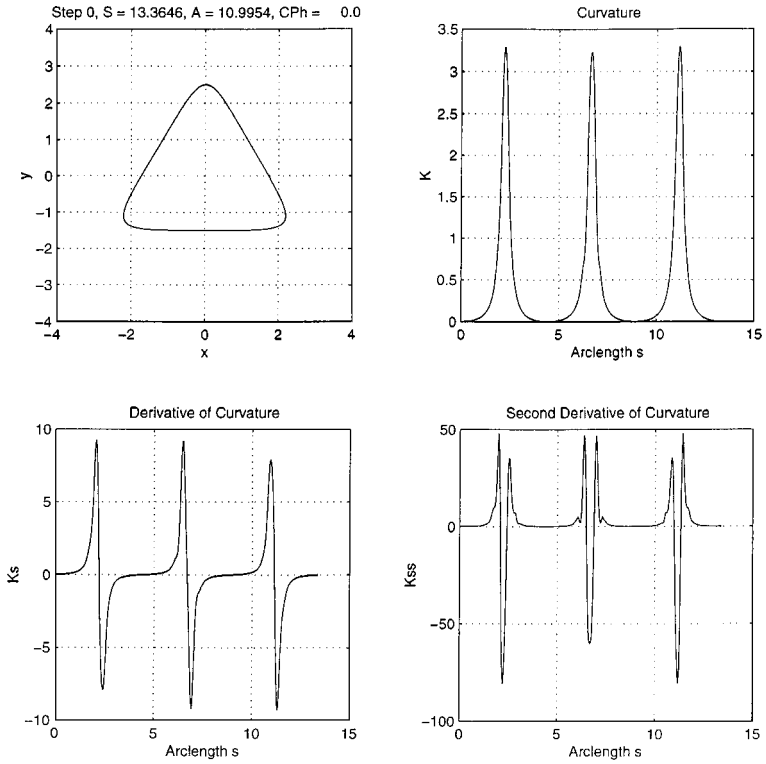


FIG. 8. Triangular front, its curvature, and derivatives of curvature.

The arclength $s(r)$ is given by

$$s(r) = \int_0^r \sqrt{5 + 4\cos 3z}\, dz = 2\, E(3r/2, 2\sqrt{2}/3). \tag{5.12}$$

The entire perimeter is $12\, E(2\sqrt{2}/3) \approx 13.368$. The curvature derivatives are:

$$K_s = \frac{6(11 - 2\cos 3r)\sin 3r}{(5 + 4\cos 3r)^3}, \quad |K_s| \le 13.993 \tag{5.13}$$

$$K_{ss} = \frac{18(88 + 45\cos 3r - 54\cos 6r + 2\cos 9r)}{(5 + 4\cos 3r)^{9/2}}, \quad -234 \le K_{ss} \le 57.6028. \tag{5.14}$$

The area confined by the front is

$$A = \frac{1}{2}\oint (f_x\, df_y - f_y\, df_x) = \frac{1}{2}\int_0^{2\pi} (f_x\, f_y' - f_y\, f_x')\, dr = \frac{7\pi}{2}. \tag{5.15}$$

A grid of size 60×60 was used. The relationships for the curvature and its derivatives are plotted vs arclength in Fig. 8. Results of the numerical simulation are presented in Fig. 9. The simulation took 40,000 steps (1.8 h CPU time) to reach a steady state. It is interesting to note that the initial curvature is nonnegative everywhere and vanishes at three discrete points only (at the midpoints of the triangle sides). However, the curvature becomes locally negative as the front moves. Four hundred fifty points were used to describe the interface

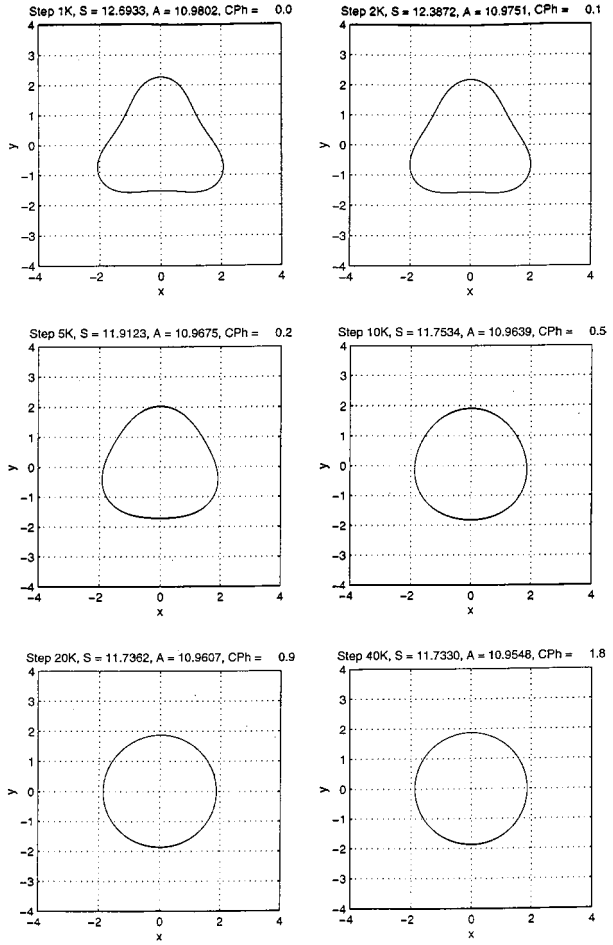


FIG. 9. Evolution of initially triangular front under surface diffusion forces.

configuration. The center of gravity of the figure given by Eq. (5.10) is located at the origin. The side of the regular circumferential triangle (with sharp corners) is $3\sqrt{3}$. The area of the figure makes up $28\pi/54\sqrt{3} \approx 94\%$ of the triangle area. The other missing 6% is due to rounded corners.

Square with rounded corners. The parametric description of a square with rounded corners, illustrated by Fig. 10, is given by

$$f_x = \frac{(5 - \cos 2r)\cos r}{4}, \quad f_y = \frac{(5 + \cos 2r)\sin r}{4}, \quad 0 \le r < 2\pi. \tag{5.16}$$

The initial curvature is

$$K(r) = \frac{4\sin^2 2r}{(4 - 3\sin^2 2r)^{3/2}}, \quad 0 \le K \le 4. \tag{5.17}$$

The arclength $s(r)$ is given by

$$s(r) = \frac{3}{4}\int_0^r \sqrt{4 - 3\sin^2 z}\, dz = \frac{3}{2}E(2r, \sqrt{3}/2). \tag{5.18}$$

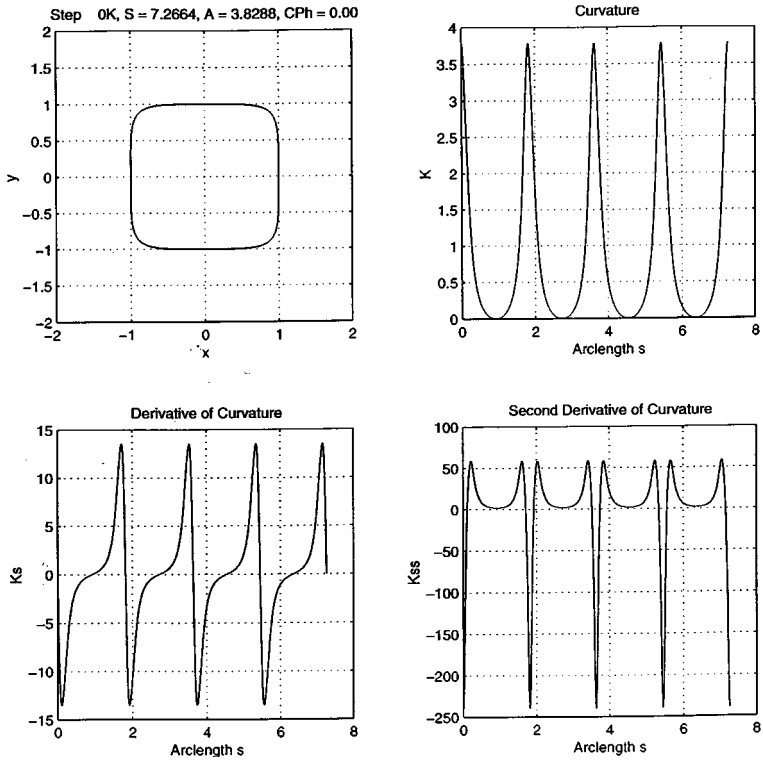


FIG. 10. Square front, its curvature, and derivatives of curvature.

The complete length of the interface is $6E(\sqrt{3}/2) \approx 7.2666$. The confined area is $39\pi/32$, and it takes approximately 96% of the area of the circumferential square with sharp corners. The derivative of the curvature with respect to the arclength is

$$K_s = \frac{4}{3}\frac{38\sin 4r - 3\sin 8r}{(4 - 3\sin^2 2r)^3}, \quad |K_s| \le 16.683. \tag{5.19}$$

The second derivative of the curvature is

$$K_{ss} = \frac{16}{9} \cdot \frac{456 + 335\cos 4r - 288\cos 8r + 9\cos 12r}{(4 - 3\sin^2 2r)^{9/2}}, \quad -312.888 \le K_{ss} \le 78.271. \tag{5.20}$$

The curvature and its derivatives are plotted vs arclength in Fig. 10. Results of the numerical simulation are presented in Fig. 11. Four hundred points were used to specify the interface curve on a grid of size 60×60. The simulation took 20,000 steps and 15 min CPU time to reach a steady state.

Butterfly contour. The butterfly contour, plotted in Fig. 12, is a solution for the ordinary differential equations

$$\frac{df_x}{ds} = \frac{z_x}{\sqrt{z_x^2 + z_y^2}}, \quad \frac{df_y}{ds} = \frac{z_y}{\sqrt{z_x^2 + z_y^2}}, \tag{5.21}$$

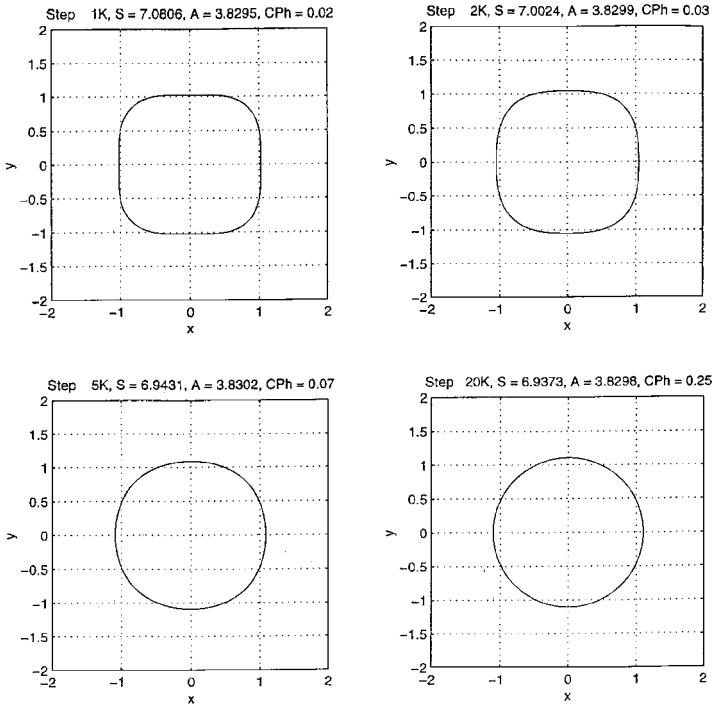


FIG. 11. Evolution of initially square front under surface diffusion forces.

where s is the arclength and z_x, z_y are functions defined as

$$z_x = -f_y^5\left(f_x^2 + f_y^2 + a^2\right), \quad z_y = f_x^5\left(f_x^2 + f_y^2 - a^2\right). \tag{5.22}$$

We set $a = 1$ and used the initial conditions at $s = 0:\ f_x = 0$ and $f_y = 0.2$. The interface length is 6.3818, and the confined area is 1.6108. The curvature and its derivatives are

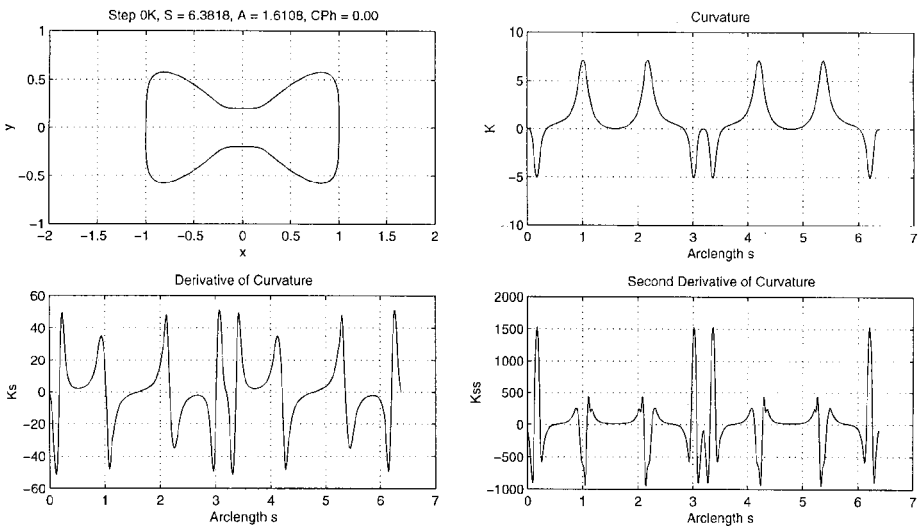


FIG. 12. Butterfly front, its curvature, and derivatives of curvature.

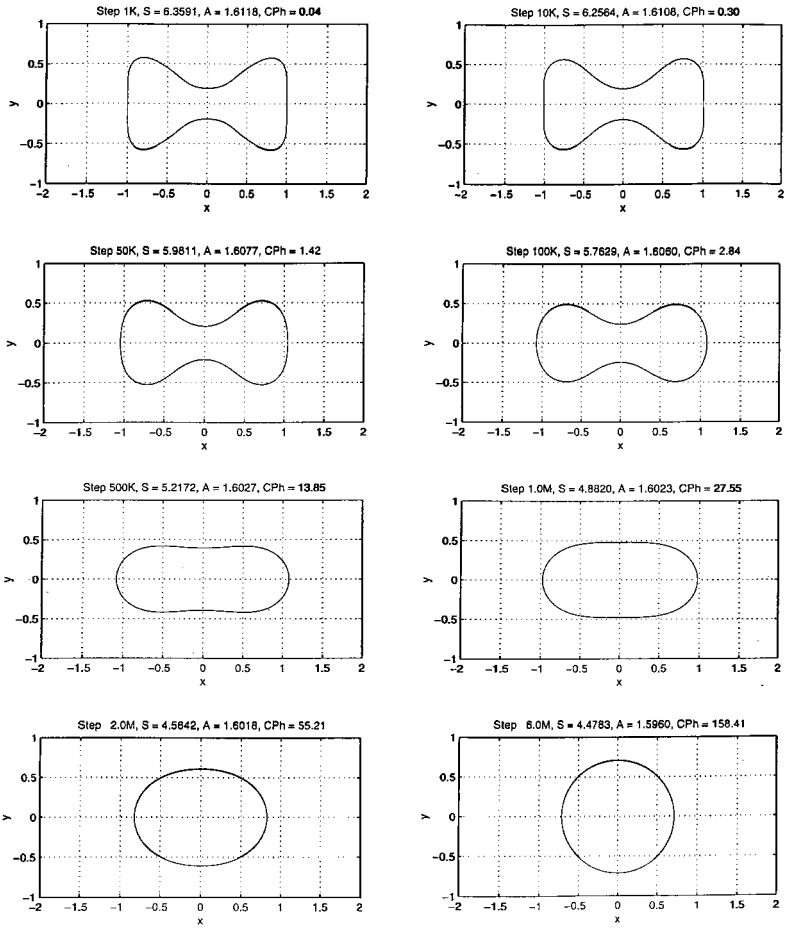


FIG. 13. Evolution of butterfly front under surface diffusion forces.

plotted vs arclength in Fig. 12. Results of the numerical simulation are presented in Fig. 13. A 100×100 grid and 1000 points that specify the interface line were used. Note that during the evolution, the butterfly (see Fig. 12) transforms first into the Cassini oval, then evolves into the ellipse, and finally reaches a steady-state circle.

Groove evolution. We consider the evolution of a long groove with rounded ends; see Fig. 14. Note that the rounding curve is not a circular arc. The circle is inappropriate for this simulation. The reason is that at the point where the circular arc joins the straight line, the curvature is discontinuous and jumps from zero to $K = 1/\text{radius}$. Thus, the curvature is a step function, its derivative with respect to arclength is a delta function, and higher derivatives are delta functions of higher orders. Such interface configurations are usually treated with essentially nonoscillatory (ENO) schemes. An ENO scheme is not used here. Therefore, we apply a special transition curve. Its curvature is a polynomial function of the arclength. At the joint points, the curvature and its derivatives (up to the second order and even higher) all vanish, to match those of a straight line. For the groove considered, the straight portions are of length 20, the transition curves are of length 8, and the groove width is 2.4. The maximum curvature of the transition curve is 1. The interface perimeter is 56, and the confined area is approximately 64.6. Although the slope angle and its derivatives

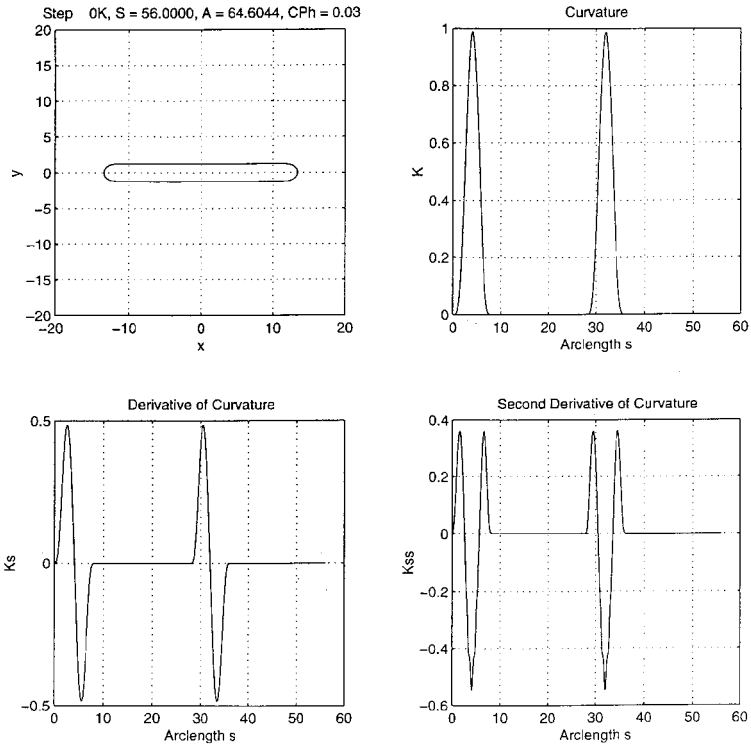


FIG. 14. Groove front, its curvature, and derivatives of curvature.

are continuous, they vary rapidly with the arclength, and the corresponding plots in Fig. 14 bear a resemblance to delta functions of different orders. The curvature and its derivatives are plotted vs arclength in Fig. 14. Results of the numerical simulation are presented in Fig. 15.

A 200×200 grid was used. Six thousand points were used to specify the interface line. The groove does not split into pieces as it evolves. It remains a unified, simply connected region.

Dumbbell evolution. In order to check accuracy and convergence in time, a dumbbell problem was run using two space resolutions: a coarse grid, 100×100, and a fine grid, 150×150. A "regular" time step of $\lambda = 1/8$ was used with both grids.

The evolution scenario for the coarse grid and regular time step (CFL) is presented in Fig. 16. The dumbbell evolved to a perfect circle. The fine grid with the regular (CFL) time step accumulated a significant error, after a large number of steps, and the computation was stopped before reaching steady-state.

The same fine-grid computation was repeated with a time step twice as small. The evolution scenario was visually the same as that for the coarse grid and regular step; it reached a steady state at the same problem time.

The area confined by the closed interface should remain constant and the error in this measure can be used as an indicator for the accuracy of the computation. Two plots are shown in Fig. 17. The upper plot shows the effect of the mesh resolution when using the same $\lambda = 1/8$. The time axis for both graphs is normalized in such a way that it is proportional to "problem time," i.e., it corresponds to the step of the coarse grid, while the fine-grid steps

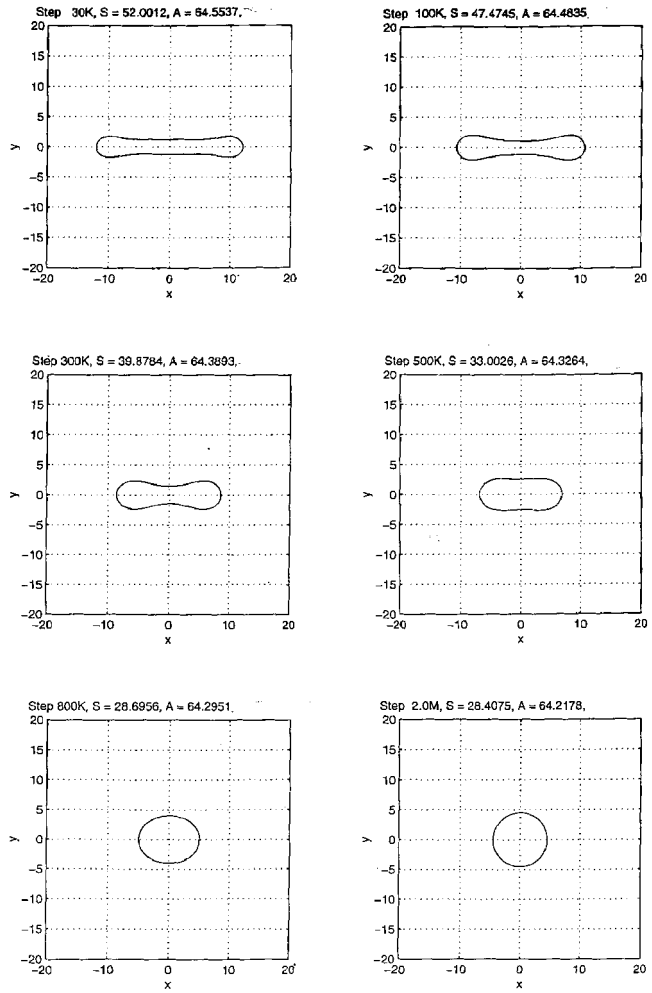


FIG. 15. Evolution of long groove under surface diffusion forces.

are adjusted by a factor of 1.54. The lower plot shows the effect of the time step when using the same fine grid. The time steps correspond to $\lambda = 1/8$ and $\lambda = 1/16$. The time axis is again normalized accordingly.

Topological changes. In order to check the ability of the present formulation to follow topological changes we performed two experiments.

The first involves the merging of two long ellipses of aspect ratio 10 with centers 0.42 apart; see Fig. 18. The ellipses become fatter and eventually touch at the symmetry axis. At this point the parametrization of the two interface curves is changed to a single curve. Subsequently, the merged void evolves to a perfect circle.

The second experiment involves the same two long ellipses but this time with centers 0.32 apart; see Fig. 19. The ellipses first touch at two points off the vertical symmetry axis. Now there is a merging of the two outer parts of the curves and simultaneously the merging of the inner parts of the curves and the splitting of an inner island of material. Subsequently, both the merged void and the inner island evolve to perfect circles. The final configuration is multiply connected.

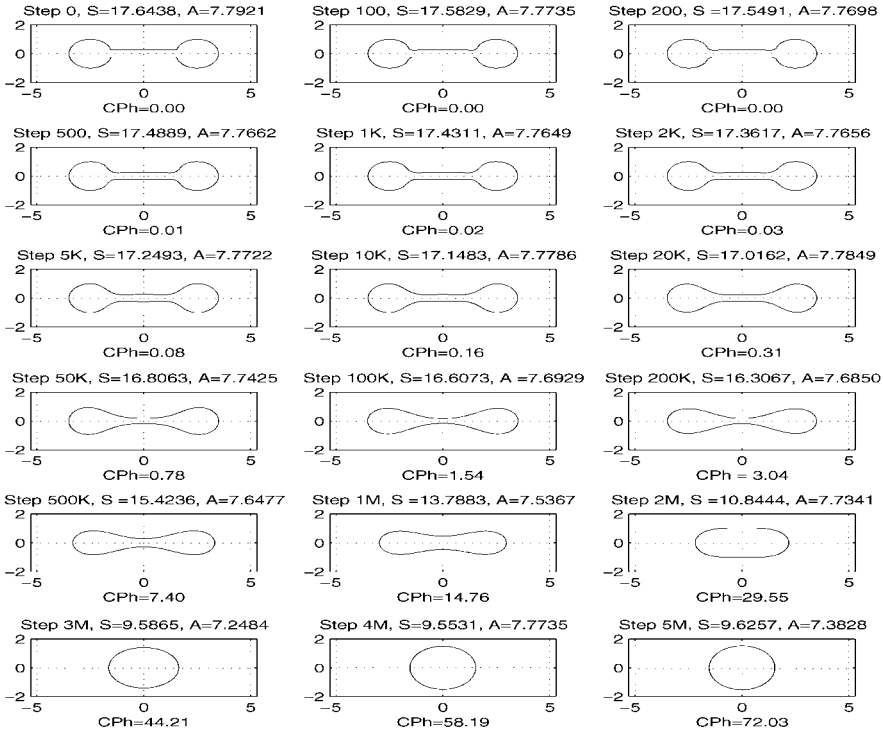


FIG. 16. Evolution of dumbbell front under surface duffusion forces.

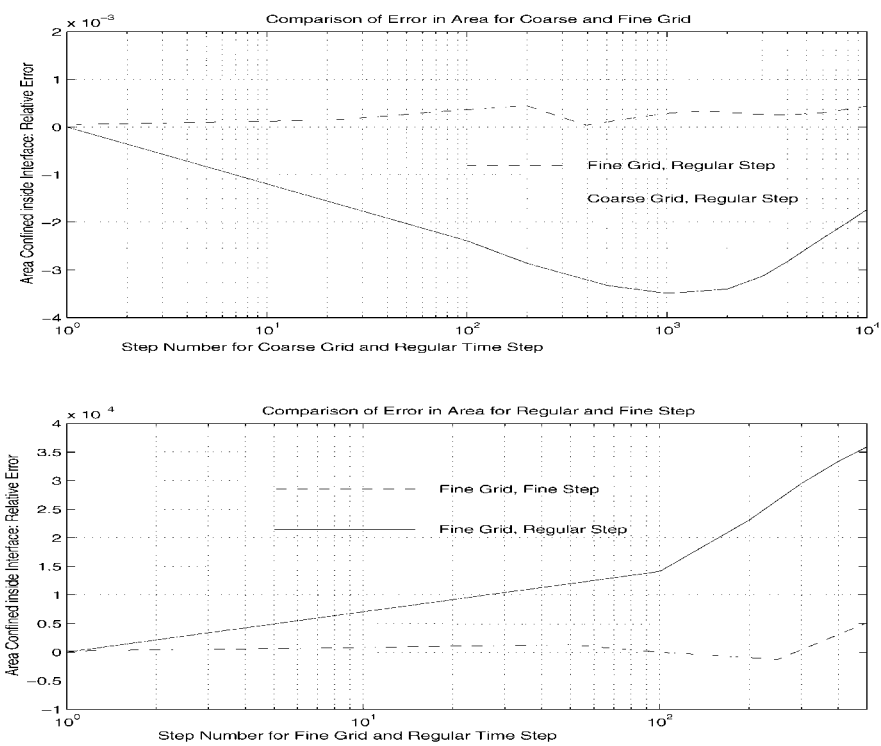


FIG. 17. Accuracy of numerical scheme for a dumbbell evolution: Coarse and fine grid; regular and fine time step.

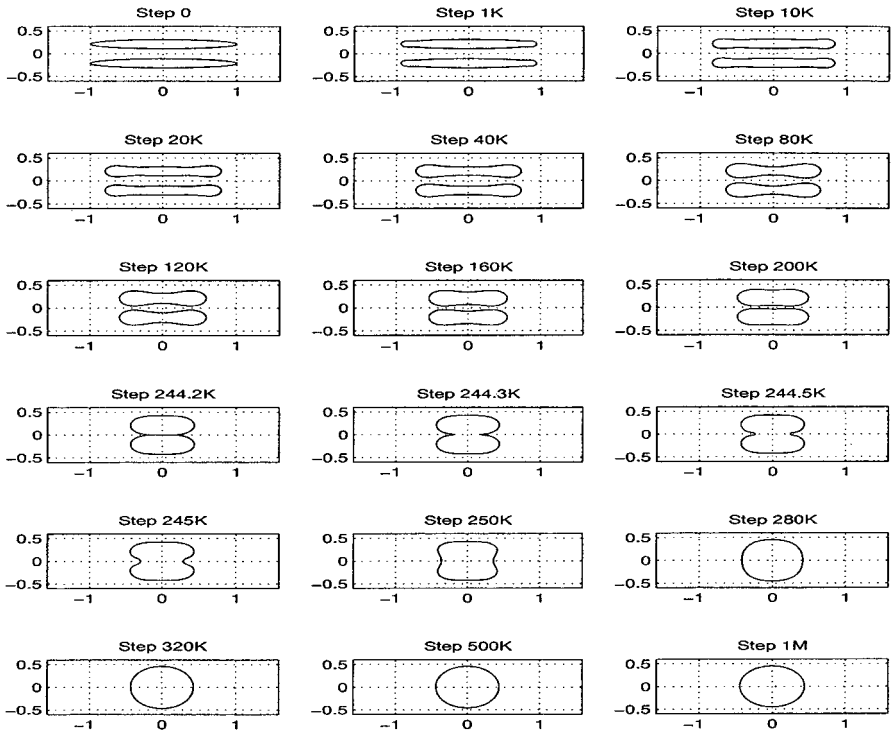


FIG. 18. Merging of two elliptic fronts under surface diffusion forces, resulting in a simply connected region.

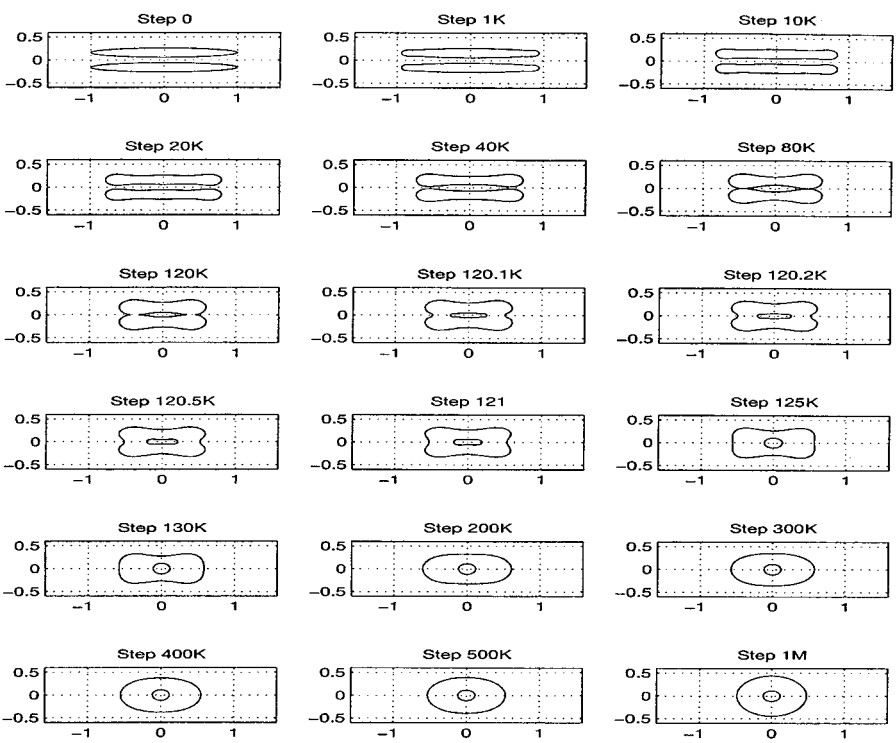


FIG. 19. Merging of two elliptic fronts under surface diffusion forces, resulting in a multiply connected region.

5.2. *Motion under Surface Diffusion and Electromigration Forces*

In this section we examine three major cases of ratio between field forces and diffusion forces. In our study they are classified as weak, moderate, and strong electric fields.

The interface motion under governing equation (2.2) was studied, by varying α, the ratio between the electric field forces and the surface diffusion forces.

Weak electric field. In this series we assume $\alpha = 1$, which corresponds to a relatively weak field. Due to the field forces, the void migrates in the direction of the field, but since the diffusion forces prevail, the steady state configuration of the void contour is a circle. For all the tests, the conductivity of the void is $k_{in} = 10^{-6}$ while the conductivity of the surrounding material is $k_{out} = 1$. The gradient of the external voltage (in the x direction) was assumed to be one. Figs. 20–29 present the void initial configuration, evolution, and migration. For the initial configuration, we present the voltage and its derivatives vs arclength of the void and the resulting normal velocity of the front. We also specify the total electric resistance of the interconnect which is given by Eq. (3.23).

First, migration of a circular void was studied. The initial circular shape does not change. According to Eq. (2.20), the velocity of the migration of the circular void is $2\alpha E/R$, where R is the void radius and E is the field strength.

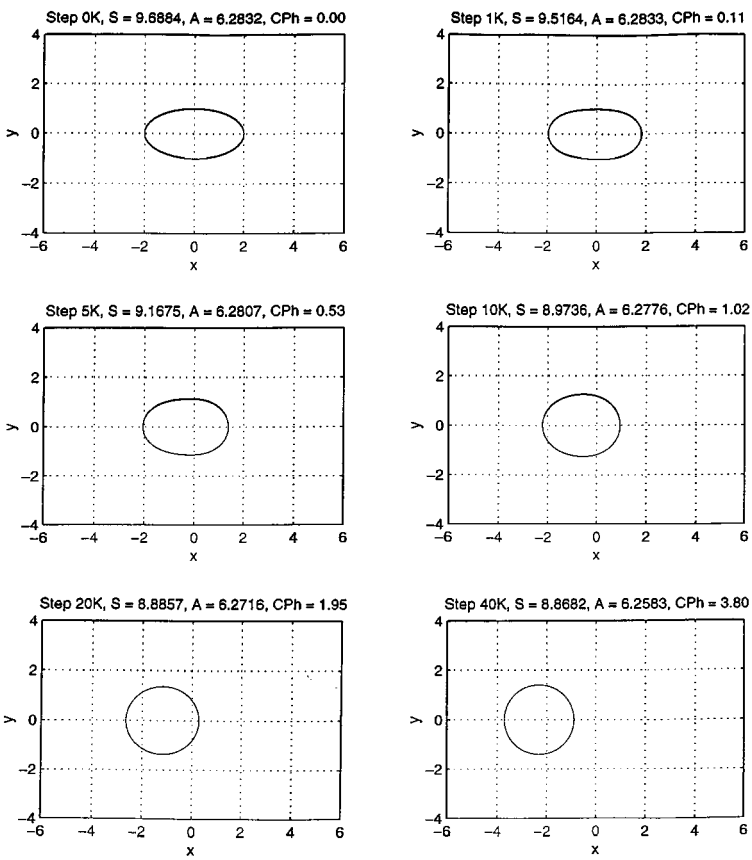


FIG. 20. Evolution and migration of elliptic void in a weak electric field (with the large axis of the ellipse coinciding with the field direction).

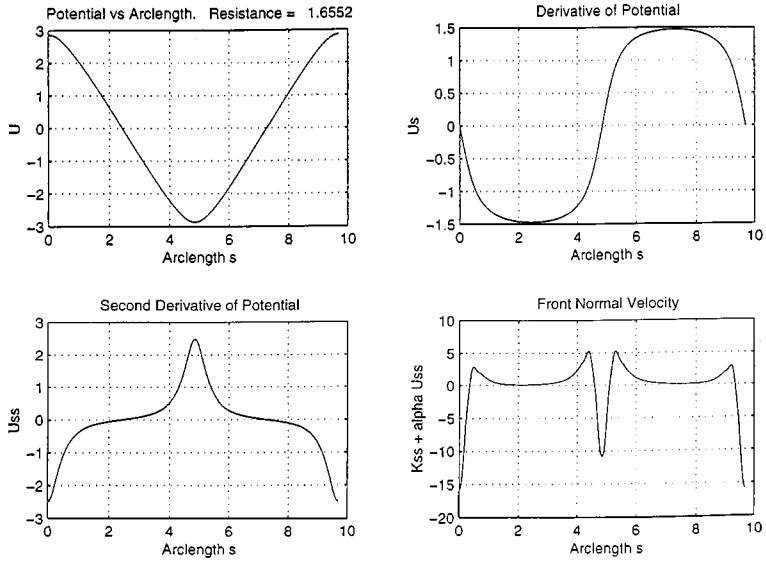


FIG. 21. Distribution of electric potential, its derivatives, and normal velocity for elliptic void (with the large axis of the ellipse coinciding with the field direction).

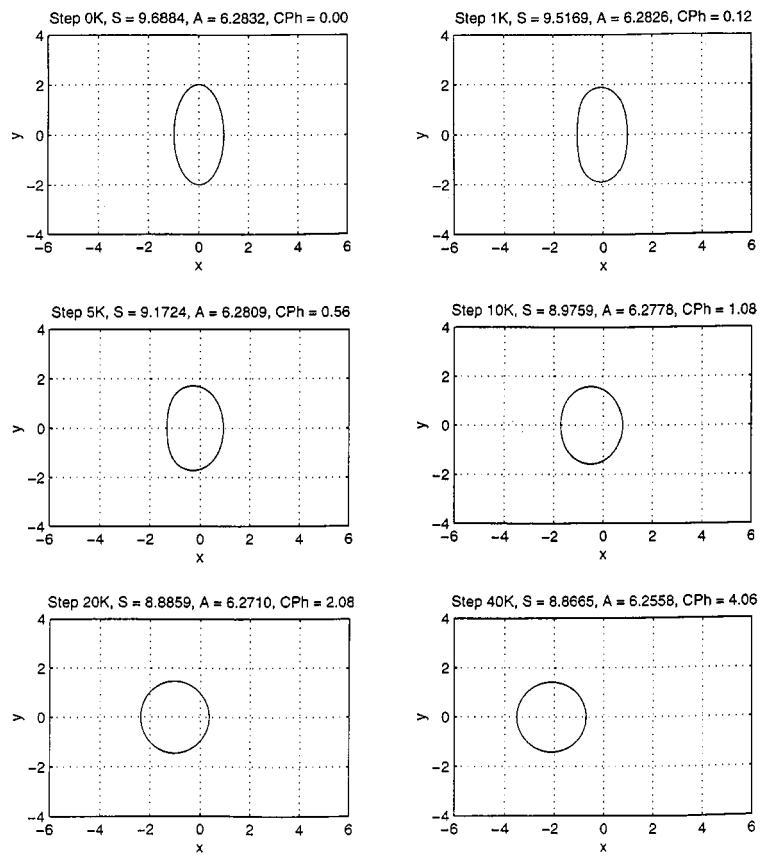


FIG. 22. Evolution and migration of elliptic void in a weak electric field (with the large axis of the ellipse normal to the field direction).

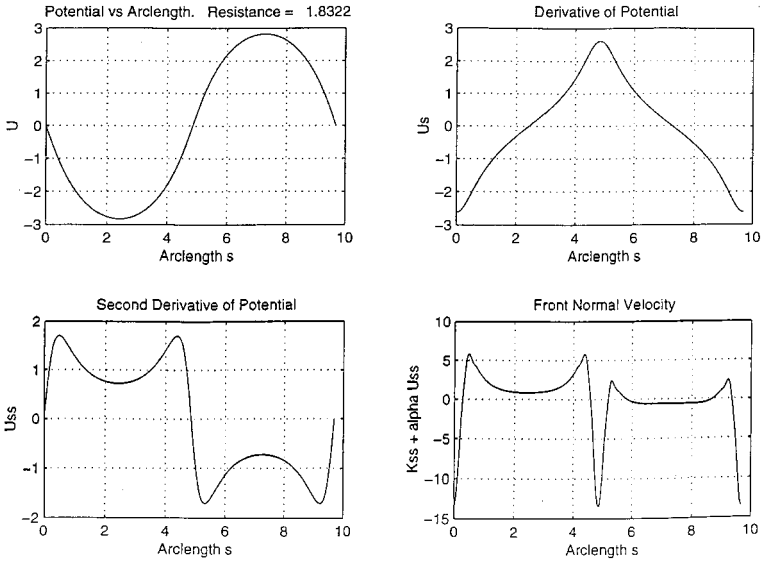


FIG. 23. Distribution of electric potential, its derivatives, and normal velocity for elliptic void (with the large axis of the ellipse normal to the field direction).

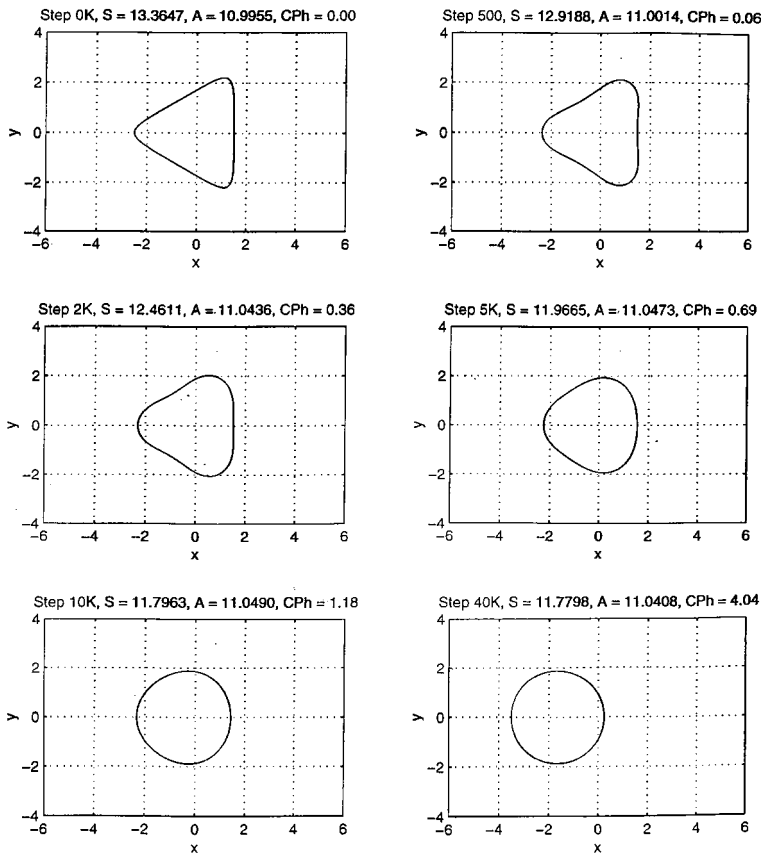


FIG. 24. Evolution and migration of triangular void in a weak electric field (with the large axis of the triangle coinciding with the field direction).

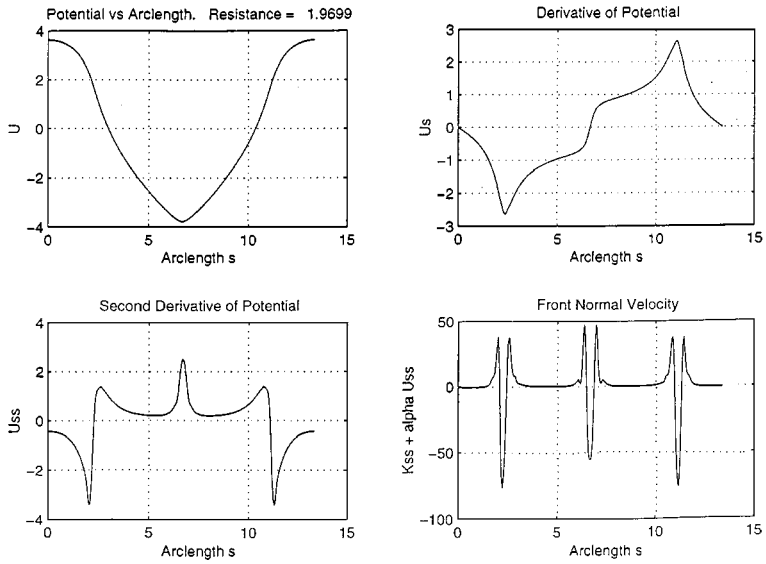


FIG. 25. Distribution of electric potential, its derivatives, and normal velocity for triangular void (with the large axis of the triangle coinciding with the field direction).

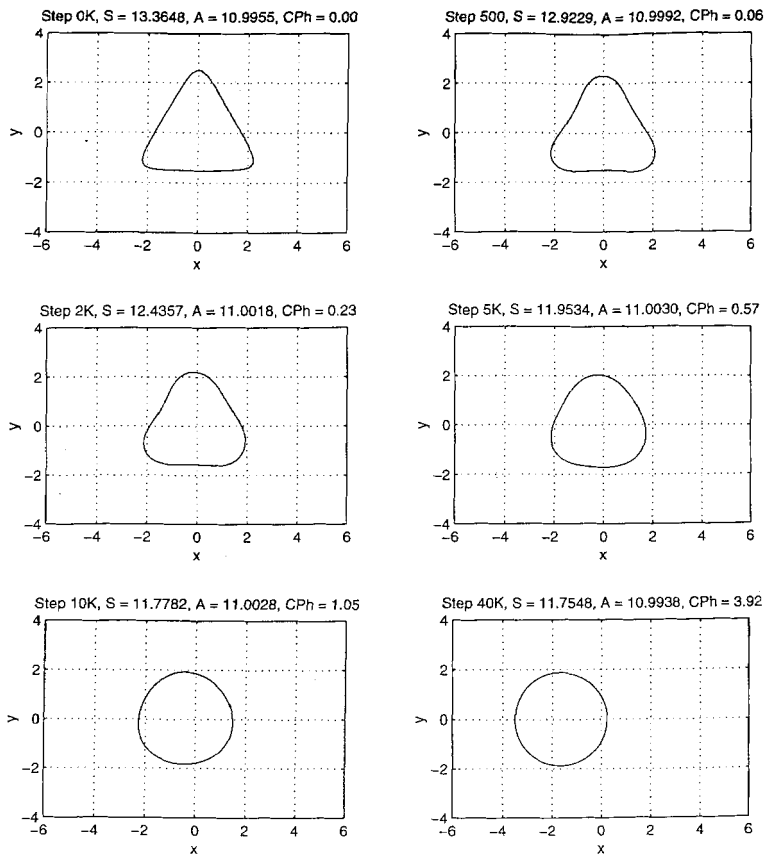


FIG. 26. Evolution and migration of triangular void in a weak electric field (with the large axis of the triangle normal to the field direction).

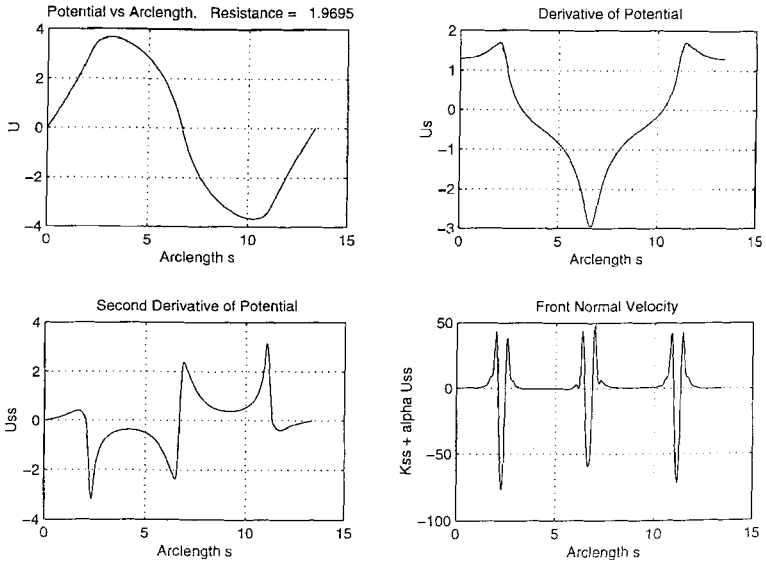


FIG. 27. Distribution of electric potential, its derivatives, and normal velocity for triangular void (with the large axis of the triangle normal to the field direction).

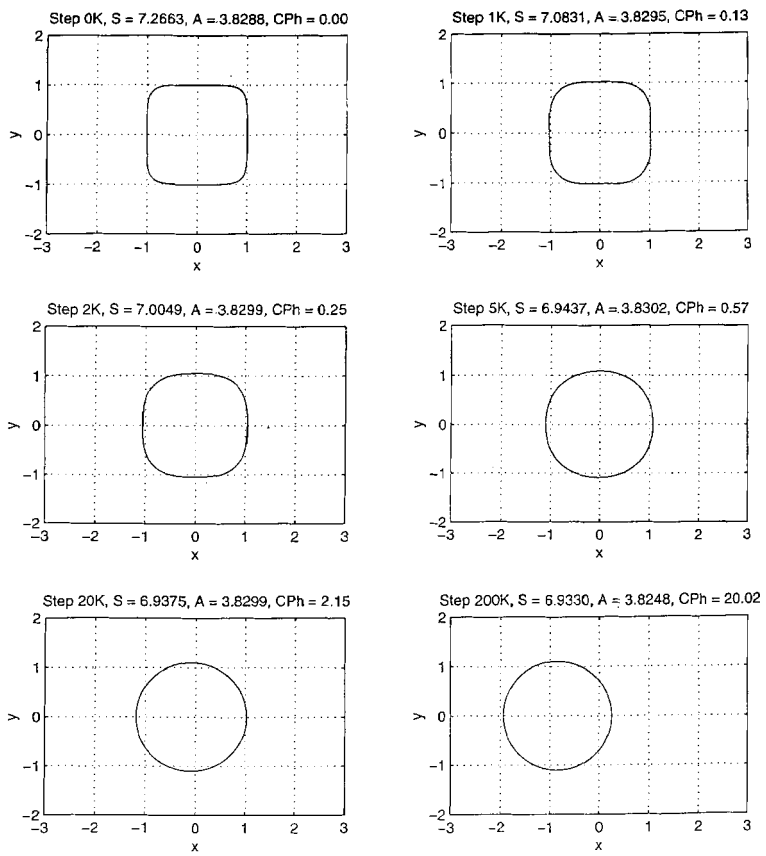


FIG. 28. Evolution and migration of square void in a weak electric field.

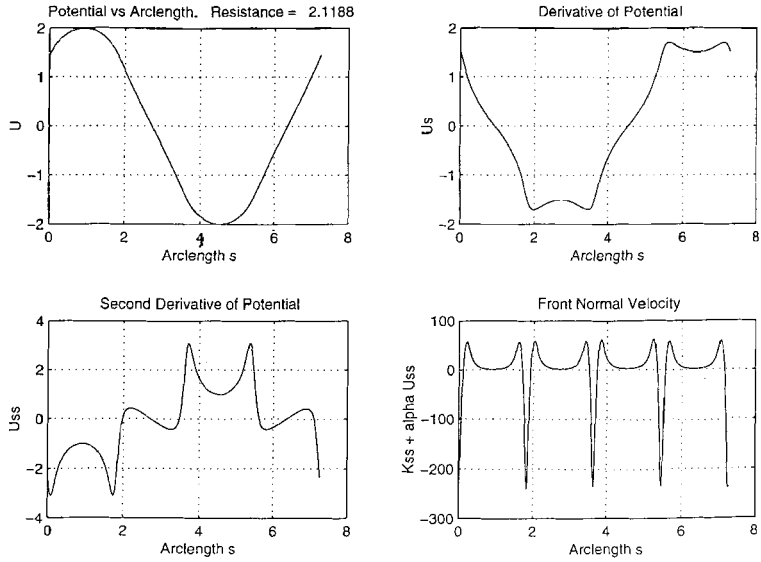


FIG. 29. Distribution of electric potential, its derivatives, and normal velocity for square void.

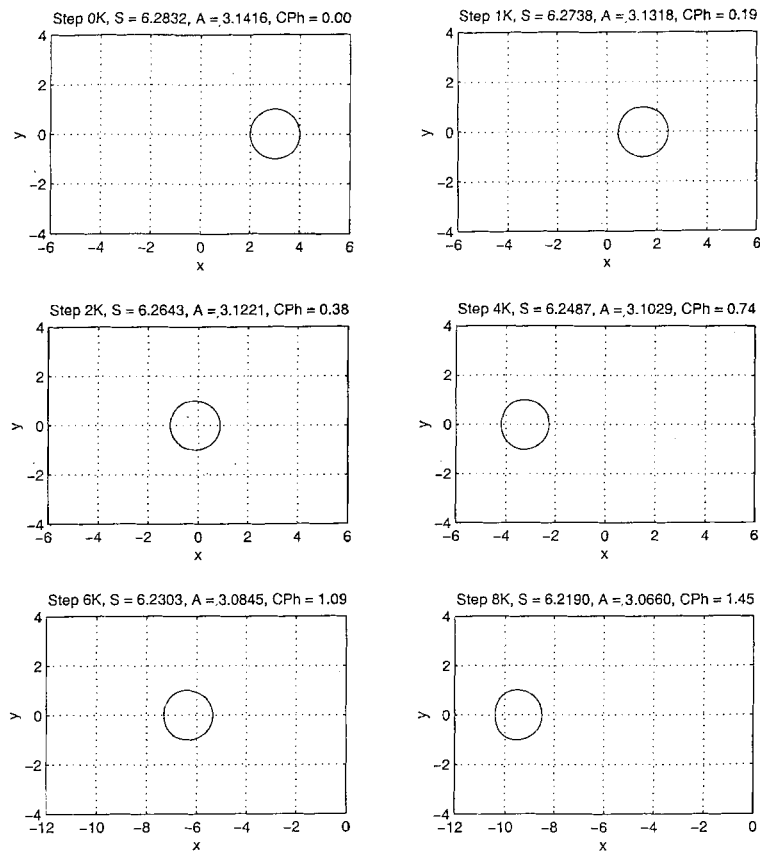


FIG. 30. Evolution and migration of circular void in a moderate electric field.

Migration and evolution of an elliptic void is simulated in Fig. 20. The larger axis of the ellipse is parallel to the strength $\boldsymbol{E}$ of the external field. Figure 21 presents the voltage, its derivatives, and the normal velocity of the initial front. A similar problem, where the larger axis is normal to the field strength, is presented in Fig. 22. The initial characteristics of the front are presented in Fig. 23.

It is interesting to note that in the first case, the initial resistance of the interconnect is less than its steady state resistance. In the latter case, the initial resistance exceeds that of the steady state.

Next we consider migration and evolution of a triangular void. The triangle is equilateral with rounded corners. We distinguish between two cases: 1. the axis of symmetry of the triangle is parallel to the elective field (see Fig. 24 for the evolution and motion and Fig. 25 for the initial characteristics); 2. the axis of symmetry is normal to the field (Figs. 26 and 27, respectively). The motion and initial characteristics of a square void with rounded corners are presented in Figs. 28 and 29.

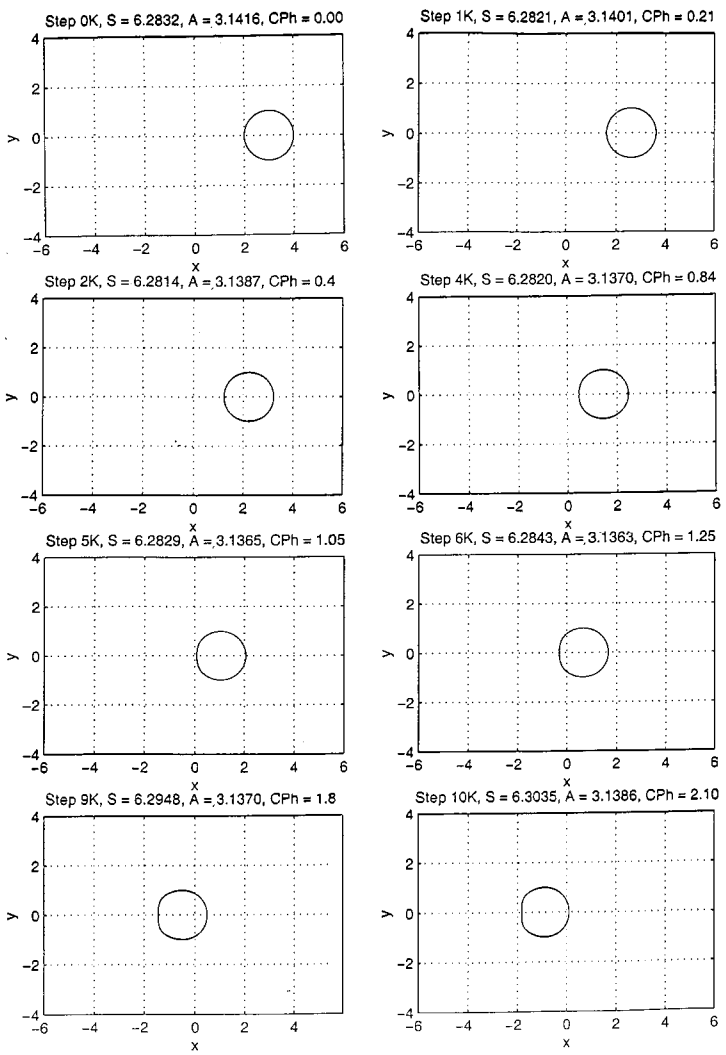


FIG. 31. Evolution and migration of circular void in a strong electric field.

Moderately strong field. Motions caused by a strong electric field ($\alpha \gg 1$) were considered. It is known [2] that when the parameter α exceeds a definite critical value, the circular equilibrium form of the void boundary becomes unstable, and other (stable) equilibrium forms evolve. Kraft and Arzt [25] found that electromigration-induced failure of metallic interconnects in integrated circuits occurs when rounded voids deform into narrow slit-like voids, which are often transgranular. The mechanism of this shape change is examined in [25] where the numerical simulation is applied for both finite element and finite difference approaches. These authors use a finite element method to find the distribution of the density of electric current and temperature in the vicinity of the void. The authors take into consideration the nonuniform Joule heating and the dependence of the surface diffusion coefficient on the temperature.

When we consider the motion of a void in a strong electric field, the computational procedure needs additional work. As the number of iterations increases, the location of the generating points on the interface line becomes more and more nonuniform. Even when the

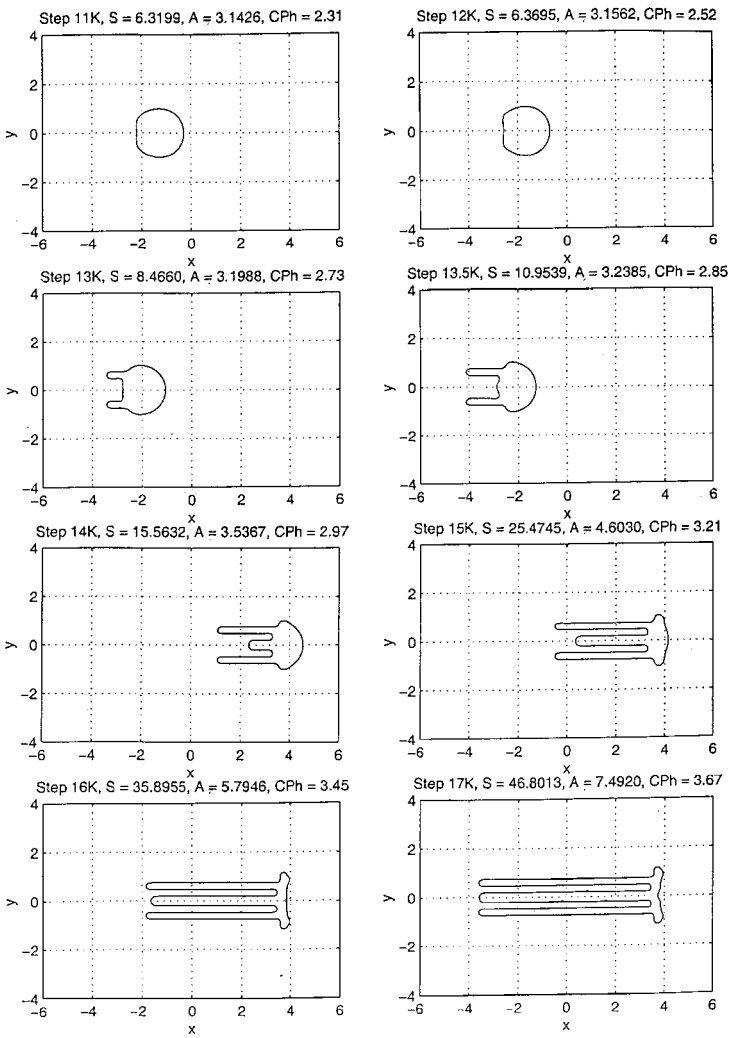


FIG. 32. Evolution and migration of circular void in a strong electric field (continued).

initial configuration of the zero level line is generated by equally spaced points (or nearly equally spaced), after some time there is a high density of generating points within relatively short sectors of the interface line, and there are long sectors where these points are sparse. To overcome this problem, a special procedure was developed to redistribute the generating points in a uniform equally spaced manner. This redistribution is accomplished by spline interpolation. The redistribution procedure should not necessarily be applied at each time step. It may be used, say, every 10 or 100 time steps, depending on the "condensation rate" of the generating points.

$\alpha = 20$ is used to simulate a moderately strong field. In this case, the round void does not remain round. It approaches a special steady state configuration where the arc of initial curvature is placed in the "plus" side of the electric field. On the other hand, the arc of a reduced or even negative curvature is placed in the "minus" side of the field. Results of this numerical simulation are presented in Fig. 30.

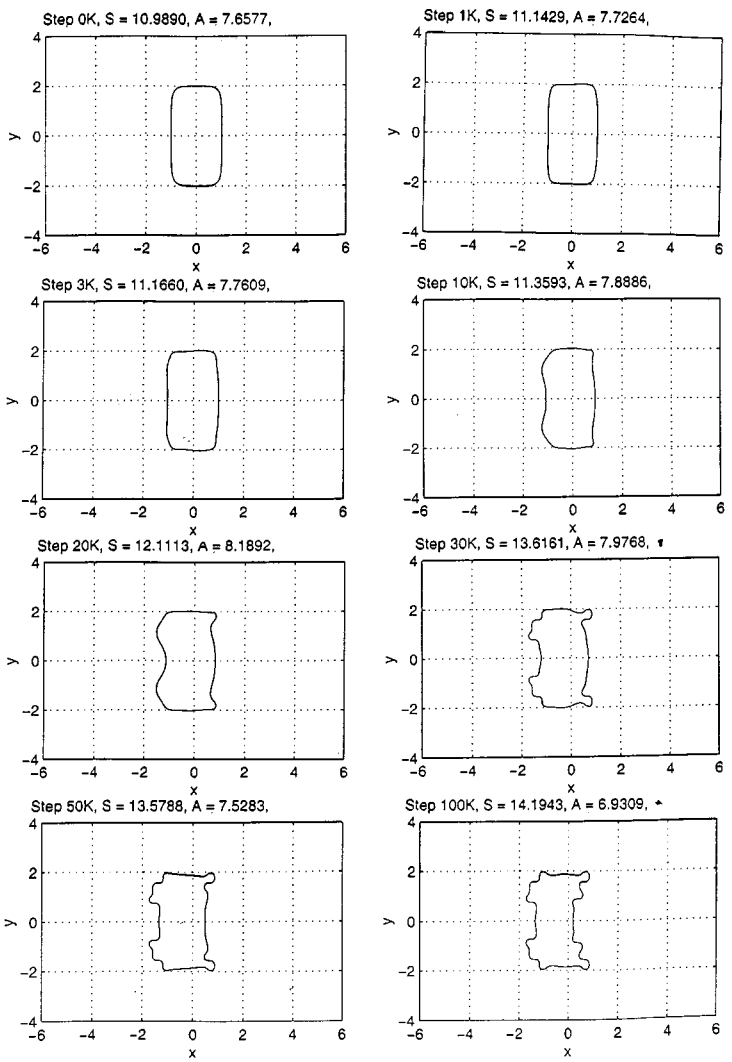


FIG. 33. Evolution and migration of rectangular void in a strong electric field.

Extremely strong field. $\alpha = 50$ is used to simulate an extremely strong field. Initially, the void reaches the shape described above for the moderately strong field. But now this is not a steady state. The void continues its evolution. It deforms and reaches a two "fingers" shape, exactly as was shown in [2]. Then, a third "finger" appears. Finally, it splits into several parts. Thus, for a very strong field, sometimes the steady state does not exist at all for large voids, and exists only for small ones. Stability depends on the area of the void. Results of the numerical simulation are presented in Figs. 31 and 32.

As we see, there is a considerable loss of accuracy at the last stages of the evolution before the split of the void occurs. At these stages, the curvature varies rapidly within a wide range which resembles a delta function. Its derivatives are delta functions of higher order. As a result, the area confined by the void varies, although it should remain constant following the material conservation law. In our numerical example the area more than doubles before the split. We conjecture that a finer spatial grid and a smaller time step may reduce the error. Note that even with this error, the general picture seems to be correct.

Dynamics of migration and evolution of the rectangular void in the extremely strong field is presented in Fig. 33. The initial sizes are 1 and 2 (1 in the field direction and 2 in the normal to the field direction). The corners of the initial rectangle are rounded.

Motion of the triangular void is presented in Fig. 34. The initial shape of the void is an equilateral triangle with rounded corners. Its axis of symmetry coincides with the field

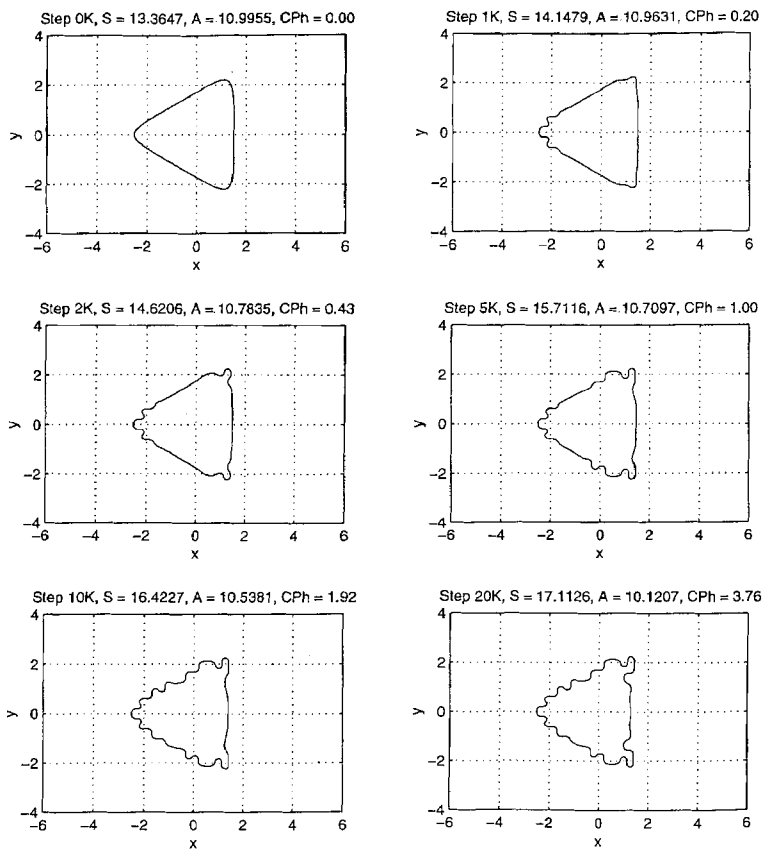


FIG. 34. Evolution and migration of triangular void in a strong electric field.

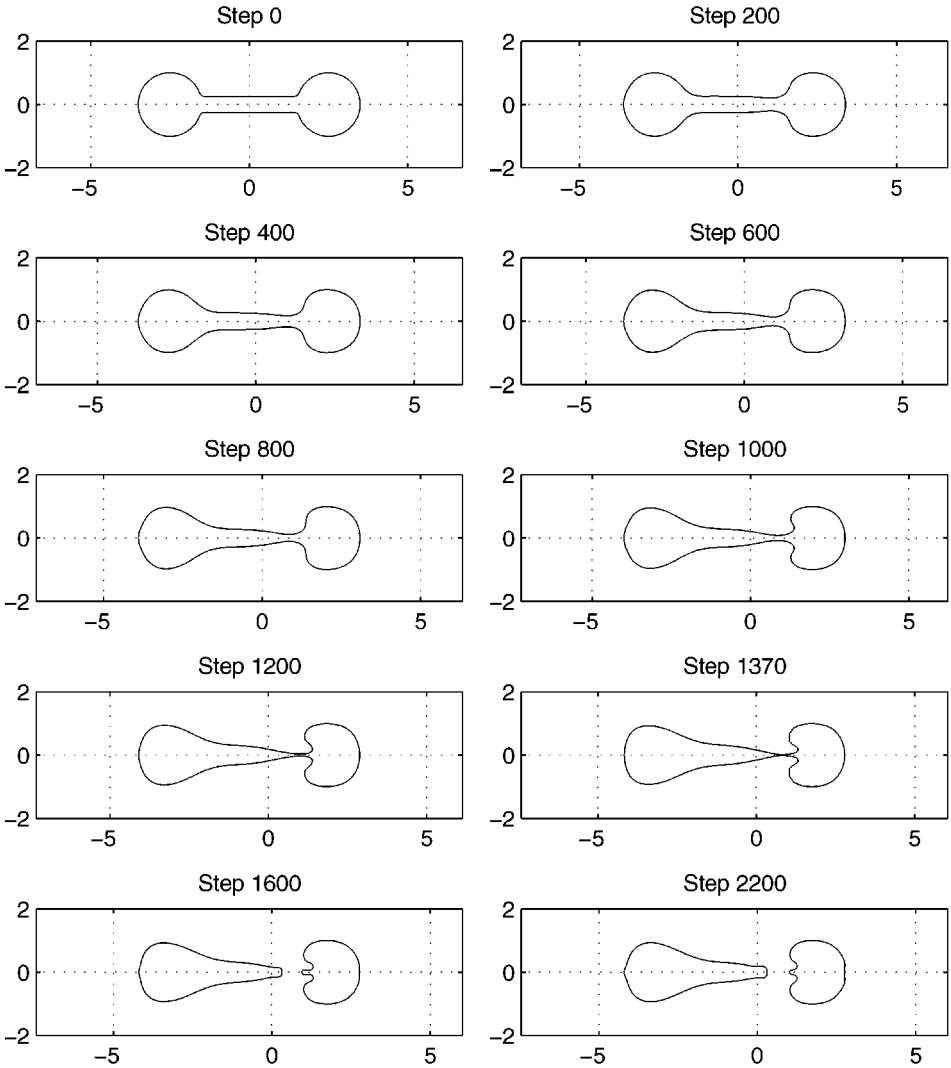


FIG. 35. Separation of dumbbell void in a strong electric field. The grid is 60×50 with 400 generating points for the interface line.

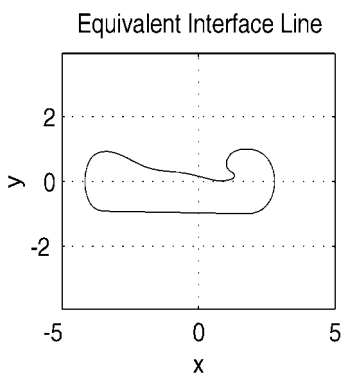


FIG. 36. Equivalent interface line for diffusion component of velocity for upper part of dumbbell.

direction. The rounded corner, located on this axis, points toward the negative of the external voltage. Figure 35 shows a separation example which is a motion under both diffusion and electric field forces (if there is diffusion only, then no separation occurs, as any initial shape becomes circular).

The interface in the dumbbell example is symmetric. At the bottleneck, the level function ceases to be a distance function because two branches become too close to each other. This leads to a distortion of the level function since we have overlapping of two different distance functions. In order to calculate the fourth derivatives numerically we need at least five cells.

To avoid this situation, the lower branch of the dumbbell was replaced by another curve. This curve was obtained with a tangent line to the circles of the dumbbell in the lower half-plane, as demonstrated in Fig. 36.

This curve was used to calculate the diffusion component of the velocity for the upper half-plane. It is correct because the diffusion component at the given point of the interface depends only on local conditions, i.e., on the shape of the interface near this point. On the other side, this modified interface does not include a bottleneck and has no problems with the distance function overlap.

Hovewer, this is not true for the electric field component of the velocity. Distribution of the voltage and its derivatives at the given point depends on the values of conductivity at all points of the domain, so the true shape with the bottleneck is used to calculate the electric component of the velocity. However, the distance function is not needed for this electric component, so the overlap problems do not arise. After we find the interface line on each time step, the true shape is an upper branch + symmetric lower branch.

6. CONCLUSIONS

A computational methodology was developed and tested for the simulation of electromigration phenomena in the interconnects of microelectronic circuits. A void motion was studied first under surface diffusion only, and then under both diffusion and electric field forces. Under pure surface diffusion, any initial configuration of the front reaches a steady state which is in a circular form. The area, confined by the closed front, remains constant during the evolution. In the weak electric field, an arbitrary initial shape becomes a circle, and in addition, the void moves along the field direction. The superconductive metallic island does not move. Under the strong electric field, the circular equilibrium form of the void becomes unstable and it is transformed into slit-like configurations.

In order to solve the fourth-order governing PDE a finite difference discretization in space and a Runge–Kutta 2 procedure in time were applied. The low order of Runge–Kutta scheme is justified since the value of the time step is fairly small for stability requirements.

The initial location of the front should be specified to the governing PDE as a finite set of discrete points which generates a closed curve. However, there are no boundary conditions at all. Instead the forward and backward differences are applied to approximate the spatial derivatives along the boundaries of the computational box.

To solve the static elliptic PDE for the voltage distribution, a finite difference scheme was applied. The resulting set of linear equations which have a sparse band matrix is solved by a special multigrid procedure, which speeds up the computations.

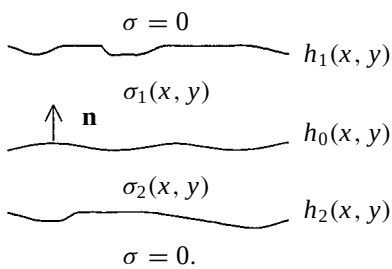
Variation of the total resistance of the interconnect was examined. This value can be further used as an interconnect failure criterion.

A number of original analytic relationships were developed for the curvature and its derivatives, which simplify the computational procedure.

APPENDIX: DERIVATION FOR THE TWO-LAYER CASE

We consider a conducting strip made of a thin metal film, attached to a strip of nonzero conductivity substrate. The metal film may be continuous or it may be made of conducting patches with voids in between. We allow the metal film and substrate to have variable thickness. In the present formulation we neglect the interface resistance. The strip is attached to electrodes at its ends. We may want to compute the total resistance of the strip as well as the local field strength which determines the resulting electromigration. This is a more realistic model as opposed to the model assuming a zero-conductivity substrate. This model also allows us to consider the behavior of a metal film with varying effective thickness at no extra cost.

We will consider a two-layer case where the subscript 1 designates the top layer (metal film) and the subscript 2 the substrate. The interface between the layers is denoted $h_0(x, y)$. Let H be the scale of the elevations and let L be the horizontal scale. We will assume that $\varepsilon = H/L \ll 1$ and that the slopes of h_i, $i = 0, 1, 2$, are small.



Ohm's law implies

$$\mathbf{j} = \sigma \mathbf{E}, \tag{7.23}$$

$$\mathbf{E} = -\nabla_3 \phi, \quad \text{where} \quad \nabla_3 = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right). \tag{7.24}$$

Here, $\mathbf{j}$ is the electric current density vector, $\mathbf{E}$ is the electric field vector, ϕ is the electric potential, and σ is the material conductivity. We assume that σ is independent of z in each layer, i.e., $\sigma_1 = \sigma_1(x, y)$, $\sigma_2 = \sigma_2(x, y)$.

For steady fields, Maxwell's equations with vanishing space charge yield

$$\nabla_3 \cdot \mathbf{j} = 0, \tag{7.25}$$

hence,

$$\nabla_3 \cdot (\sigma \mathbf{E}) = \nabla_3 \cdot (\sigma \nabla_3 \phi) = 0. \tag{7.26}$$

In the scaled coordinates $(X, Y, Z) = (x/L, y/L, z/H)$, Eq. (7.26) becomes

$$\varepsilon^2 \nabla \cdot (\sigma \nabla \phi) + \frac{\partial}{\partial Z} \left(\sigma \frac{\partial \phi}{\partial Z} \right) = 0, \tag{7.27}$$

where

$$\nabla \equiv \nabla_{horizontal} = \left(\frac{\partial}{\partial X}, \frac{\partial}{\partial Y} \right).$$

The following relations describe normal and tangential continuity at the interface between the layers:

$$\nabla_3 \cdot (\sigma \mathbf{E}) = 0 \Rightarrow \mathbf{n} \cdot (\sigma_2 \mathbf{E}_2 - \sigma_1 \mathbf{E}_1) = 0, \tag{7.28}$$

$$\nabla_3 \times \mathbf{E} = 0 \Rightarrow \mathbf{n} \times (\mathbf{E}_2 - \mathbf{E}_1) = 0, \tag{7.29}$$

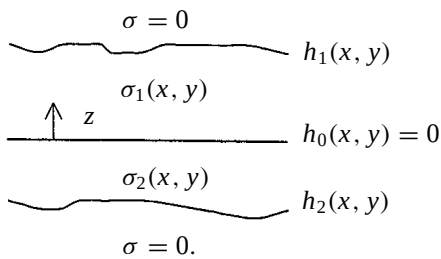
where $\mathbf{n}$ is a vector normal to the interface. Hence,

$$\mathbf{n} \cdot (\sigma_2 \nabla_3 \phi_2 - \sigma_1 \nabla_3 \phi_1) = 0, \tag{7.30}$$

and

$$\mathbf{n} \times (\nabla_3 \phi_2 - \nabla_3 \phi_1) = 0. \tag{7.31}$$

For simplicity we will assume that the material interface h_0 is horizontal at $z = 0$.



We therefore have at $Z = 0$,

$$\sigma_1 \frac{\partial \phi_1}{\partial Z}(X, Y, 0) = \sigma_2 \frac{\partial \phi_2}{\partial Z}(X, Y, 0), \tag{7.32}$$

$$\nabla \phi_1 = \nabla \phi_2. \tag{7.33}$$

The top and bottom surfaces are given by

$$S_j(x, y, z) = 0 = z - h_j(x, y) = H(Z - H_j(X, Y)). \tag{7.34}$$

Since

$$\frac{\partial \phi}{\partial n} = \frac{\nabla_3 \phi \cdot \nabla_3 S_j}{|\nabla_3 S_j|} = 0,$$

then at the top and bottom surfaces

$$\frac{\partial \phi}{\partial z} = H \nabla_3 \phi \cdot \nabla_2 H_j \left(\frac{x}{L}, \frac{y}{L} \right), \tag{7.35}$$

or, equivalently,

$$\frac{\partial \phi}{\partial Z} = \varepsilon^2 \nabla \phi \cdot \nabla H_j. \tag{7.36}$$

We can perform a singular perturbation analysis by assuming the expansion

$$\phi(X, Y, Z) = \sum_{m=0}^{\infty} \varepsilon^{2m} \phi^{(m)}(X, Y, Z), \tag{7.37}$$

where the $\phi^{(k)}$ are functions of order $O(1)$ in ε. Substitution of (7.37) into (7.27), setting $\varepsilon = 0$, produces

$$\frac{\partial^2 \phi^{(0)}}{\partial Z^2} = 0, \tag{7.38}$$

$$\phi^{(0)} = A_2 Z + B_2 \text{ (layer 2)}, \tag{7.39}$$

$$\phi^{(0)} = A_1 Z + B_1 \text{ (layer 1)}. \tag{7.40}$$

By (7.36) we have at the top

$$\frac{\partial \phi_2^{(0)}}{\partial Z} = 0 \Rightarrow \phi_2^{(0)} = B_2(X, Y), \tag{7.41}$$

and at the bottom

$$\frac{\partial \phi_1^{(0)}}{\partial Z} = 0 \Rightarrow \phi_1^{(0)} = B_1(X, Y). \tag{7.42}$$

Therefore the continuity condition (7.33) implies (apart from a moot constant)

$$B_2 = B_1. \tag{7.43}$$

Thus, at this order,

$$\phi_2^{(0)} = \phi_1^{(0)} = \phi^{(0)}(X, Y). \tag{7.44}$$

Now we consider the next-order terms in (7.36). At the jth interface (top and bottom)

$$\frac{\partial \phi_j^{(1)}}{\partial Z} = \nabla \phi^{(0)} \cdot \nabla H_j, \quad j = 1, 2. \tag{7.45}$$

From (7.32), using (7.41) and (7.42), we get

$$\sigma_1 \frac{\partial \phi_1^{(1)}}{\partial Z} = \sigma_2 \frac{\partial \phi_2^{(1)}}{\partial Z}. \tag{7.46}$$

Now, at this order in Eq. (7.27),

$$\sigma \frac{\partial^2 \phi^{(1)}}{\partial Z^2} = -\nabla \cdot \left(\sigma \nabla \phi^{(0)}\right). \tag{7.47}$$

We integrate with respect to Z in both layers and obtain

$$\sigma_1 \frac{\partial \phi_1^{(1)}}{\partial Z} = -\nabla \cdot \left(\sigma_1 \nabla \phi_1^{(0)}\right) Z + A_1(X, Y), \tag{7.48}$$

and

$$\sigma_2 \frac{\partial \phi_2^{(1)}}{\partial Z} = -\nabla \cdot \left(\sigma_2 \nabla \phi_2^{(0)}\right) Z + A_2(X, Y). \tag{7.49}$$

At $Z = 0$ we get

$$\sigma_1 \frac{\partial \phi_1^{(1)}}{\partial Z} = A_1(X, Y), \tag{7.50}$$

and

$$\sigma_2 \frac{\partial \phi_2^{(1)}}{\partial Z} = A_2(X, Y); \tag{7.51}$$

therefore, by (7.46),

$$A_1 = A_2. \tag{7.52}$$

Consider (7.48) at $Z = H_1$:

$$\sigma_1 \frac{\partial \phi_1^{(1)}}{\partial Z} = -\nabla \cdot \left(\sigma_1 \nabla \phi_1^{(0)}\right) H_1 + A_1(X, Y). \tag{7.53}$$

Substituting $\frac{\partial \phi_1^{(1)}}{\partial Z}$ from (7.45) leads to the following expression for $A_1(X, Y)$:

$$A_1(X, Y) = \left(\sigma_1 \nabla \phi_1^{(0)}\right) \cdot \nabla H_1 + \nabla \cdot \left(\sigma_1 \nabla \phi_1^{(0)}\right) H_1. \tag{7.54}$$

Hence,

$$A_1(X, Y) = \nabla \cdot \left(\sigma_1 H_1 \nabla \phi_1^{(0)}\right). \tag{7.55}$$

Similarly, at $Z = H_2$ we obtain

$$A_2(X, Y) = \nabla \cdot \left(\sigma_2 H_2 \nabla \phi_2^{(0)}\right). \tag{7.56}$$

From (7.44), (7.52), (7.55), and (7.56), we now have

$$\nabla \cdot \left((\sigma_1 H_1 - \sigma_2 H_2) \nabla \phi^{(0)}\right) = 0. \tag{7.57}$$

Multiplying (7.57) by H and noting that h_2 is negative (so that $|h_2|$ is the thickness of the lower layer), we finally obtain

$$\nabla \left((\sigma_1 h_1 + \sigma_2 |h_2|) \nabla \phi^{(0)}\right) = 0. \tag{7.58}$$

A careful presentation of a similar derivation, with motivation and details concerning the treatment of boundary conditions, can be found in the monograph [3].

ACKNOWLEDGMENT

The authors are grateful to the reviewers for helpful comments which improved the presentation and substance of the paper.

REFERENCES

1. A. Averbuch, M. Israeli, I. Ravve, and I. Yavneh, *Fast Computation for Electro-Migration in Interconnects of Micro-Electronics Devices*, Research Report (Tel Aviv University, 1998).
2. A. F. Bower and L. B. Freund, Finite element analysis of electro-migration and stress-induced diffusion in deformable solids, in *Materials Research Society Symposium Proceedings* **391** (1995), pp. 177–188.
3. J. D. Cole, *Perturbation Methods in Applied Mathematics* (Blaisdell, Boston, 1968), p. 203.
4. X. Gui, S. Dew, and M. Brett, Numerical solution of electro-migration boundary value problem under pulse DC conditions, *J. Appl. Phys.* **80**, 4948 (1996).
5. H. Glimm, C. Klingenberg, O. McBryan, B. Plohr, D. Sharp, and S. Yaniv, Front tracking and two-dimensional Riemann problems, *Adv. Appl. Math.* **6**, 259 (1990).
6. H. Zhao, T. Chan, B. Merriman, and S. Osher, A variational level set approach to multiphase motion, *J. Comput. Phys.* **127**, 179 (1996).
7. J. A. Sethian and D. Adalsteinsson, An overview of level set methods for etching, de-position, and lithography development, *IEEE Trans. Semicond. Manuf.* **10**(N1), 167 (1997).
8. J. A. Sethian, Curvature and evolution of fronts, *Commun. Math. Phys.* **101**, 487 (1985).
9. D. Adalsteinsson and J. A. Sethian, A level set approach to a unified model for etching, deposition and lithography I: Algorithms and two-dimensional simulations, *J. Comput. Phys.* **120**, 128 (1995).
10. D. Adalsteinsson and J. A. Sethian, A level set approach to a unified model for etching, deposition and lithography II: Three-dimensional simulations, *J. Comput. Phys.* **122**, 348 (1995).
11. J. A. Sethian, *Level Set Methods. Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Science* (Cambridge Univ. Press, Cambridge, UK, 1996).
12. J. A. Sethian, *Level Set Methods and Fast Marching Methods* (Cambridge Univ. Press, Cambridge, UK, 1999).
13. R. Aris, *Vectors, Tensors and the Basic Equations of Fluid Mechanics* (Prentice Hall, London, 1962).
14. S. Osher and C. Shu, High-order essentially nonoscillatory schemes for Hamilton–Jacobi equations, *SIAM J. Numer. Anal.* **28**(N4), 907 (1991).
15. S. Osher and C. Shu, Efficient implementation of essentially non-oscillatory shock-capturing schemes I, *J. Comput. Phys.* **77**, 439 (1988).
16. S. Osher and C. Shu, Efficient implementation of essentially non-oscillatory shock-capturing schemes II, *J. Comput. Phys.* **83**, 32 (1989).
17. R. E. Alcouffe, A. Brandt, J. E. Dendy, Jr., and J. W. Painter, The multi-grid methods for the diffusion equation with strongly discontinuous coefficients, *SIAM J. Sci. Comput.* **2**, 430 (1981).
18. A. Brandt, Multi-level adaptive solutions to boundary-value problems, *Math. Comput.* **31**, 333 (1977).
19. A. Brandt, *1984 Multigrid Guide with Applications to Fluid Dynamics*, Monograph, GMD-Studie 85 (St. Augustin, West Germany, 1985).
20. A. Brandt and J. Greenwald, Parabolic multigrids revisited, in *Multigrid Methods III*, edited by W. Hackbusch and U. Trottenberg, Int. Ser. Numer. Math. **98** (Birkhauser-Verlag, Basel, 1991), pp. 143–154.
21. W. L. Briggs, *A Multigrid Tutorial* (Soc. for Industr. & Appl. Math., Philadelphia, 1987).
22. J. E. Dendy, Jr., Black box multigrid, *J. Comput. Phys.* **48**, 366 (1982).
23. J. E. Dendy, Jr., Black box multigrid for nonsymmetric problems, *Appl. Math. Comput.* **13**, 261 (1983).
24. W. Hackbusch and U. Trottenberg, Eds., *Multigrid Methods*, Lecture Notes in Mathematics (Springer-Verlag, Berlin, 1982), Vol. 960.
25. O. Kraft and E. Arzt, Numerical simulation of electromigration-induced shape changes of voids in bamboo lines, *Phys. Lett.* **66**(16), 2063 (1995).

26. L. Klinger and L. Levin, Morphological stability of a heterophase interface under electromigration conditions, *J. Appl. Phys.* **79**, 6834 (1996).
27. Z. Li, H. Zhao, and H. Gao, A numerical study of electro-migration voiding by evolving level set functions on a fixed cartesian grid, *J. Comput. Phys.* **152**(1), 281 (1999).
28. L. Xia, A. F. Bower, Z. Suo, and C. F. Shih, A finite element analysis of the motion and evolution of voids due to strain and electromigration-induced surface diffusion, *J. Mech. Phys. Solids* **45**, 1473 (1997).
29. M. Sussman, P. Smerekka, and S. Osher, A level set approach for computing solutions to incompressible two-phase flow, *J. Comput. Phys.* **114**, 146 (1994).
30. M. Sussman and E. Fatemi, An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow, *SIAM J. Sci. Comput.* **20**(4), 1165 (1999).
31. L. M. Klinger, X. Chu, W. W. Mullins, and C. L. Bauer, Grain-boundary slit propagation in an electric field, *J. Appl. Phys.* **80**, 6670 (1996).
32. T. McInerney and D. Terzopoulos, Topologically adaptable snakes, in *Proceedings of the Fifth International Conference on Computer Vision, Cambridge, MA, 1995*, pp. 840–845.
33. J. E. Taylor and J. W. Cahn, Linking anisotropic sharp and diffuse surface motions laws via gradient flow, *J. Stat. Phys.* **77**, 183 (1994).