



Domain decomposition and heterogeneous modeling used in solving a moving boundary problem

Bin Jiang

*Department of Mathematics and Statistics,
Portland State University, Portland, Oregon, USA, and*

John C. Bruch

*Department of Mechanical and Environmental Engineering
and Department of Mathematics, University of California,
Santa Barbara, California, USA*

Abstract

Purpose – To provide an excellent numerical method to simulate an important industrial process – wet chemical etching problem.

Design/methodology/approach – A mathematical model describing the wet chemical etching process is formulated in terms of a parabolic variational inequality and a non-overlapping domain decomposition (DD) method is proposed in the etching region, where a partial differential equation is treated in one sub-domain, while a variational inequality is considered in the second sub-domain. A Robin boundary condition with a parameter whose optimal value is to be found is enforced on the common boundary between these two sub-domains. Finite difference technique with projection and the Crank-Nicholson scheme for the time discretization are the major numerical tools utilized in this paper.

Findings – The proposed numerical method has achieved the best numerical performance for the famous wet chemical etching process among all the numerical schemes applied to this problem. It also shows the great computational power of domain splitting technique.

Research limitations/implications – Lack of parallel supercomputer system limits the authors to perform further numerical tests with extremely large data. This will be done in the future.

Practical implications – This is a very useful paper for the academic researchers and industry engineers who wish to develop more advanced numerical methods to simulate various kinds of industry processes along the research direction from this paper.

Originality/value – This paper provides a new and promising version of DD method in the field of moving boundary problems and offers a practical way to simulate wet chemical etching process for the engineers and scientists in the related field.

Keywords Variational techniques, Chemical technology, Finite difference methods

Paper type Research paper



1. Introduction

Wet chemical etching is an important industrial process (Kuiken, 1984, 1990; Kuiken *et al.*, 1986; Notten *et al.*, 1986). In this process, an impermeable mask is placed on the surface of the body to be etched. Caustic fluids are then allowed to come into contact with the body and etch into the body wherever the mask is not present. As time goes

on, the body will be cut away gradually by the fluids. Thus, this problem can be regarded as a moving boundary problem. During this process, undercutting of solid material from under the mask occurs and it can cause short-circuits and other serious problems. Thus, it is important to describe this behavior by predicting the profile of the moving boundary over time.

Moving boundary problems can be represented by a parabolic variational inequality. Its numerical simulation via the finite element method has been considered and the error estimate between the true solution and the numerical solution is provided in several places (Glowinski *et al.*, 1980; Johnson, 1976; Vuik and Cuvelier, 1990 and the references provided therein). Meanwhile, Vuik and Cuvelier (1985) presented a representative example of an etching problem of this type. Its variational inequality formulation was set-up in a fixed domain. Then this fixed domain problem was solved numerically along with a projection operator.

Bruch *et al.* (1993) found that this etching problem could be parallelized extremely well. A domain splitting technique is utilized there to reproduce the results of Vuik and Cuvelier (1985) with reasonable accuracy. The advantage of this method is that it can be parallelized on multiple CPU's very easily.

The non-overlapping domain decomposition (DD) method has shown its robustness and convenience in the field of computational science (Lions, 1990 and Deng, 1997). Recently, Jiang *et al.* (2006) applied a new version of non-overlapping DD method to the general free boundary problem and showed both its convergence property and numerical stability through the free surface seepage problem. In this paper we will apply this non-overlapping DD method to the etching problem. The etching region is split into two sub-domains where a partial differential equation is treated in one sub-domain while a variational inequality is considered in the second sub-domain. On the common boundary of these two sub-domains, a Robin boundary condition with a parameter is enforced. Meanwhile, the Crank-Nicholson scheme is utilized for the discretization of the time variable. The numerical results show the advantage of this method. Finally, we will determine the range of the parameter from the Robin boundary condition and the relaxation parameter in the successive over-relaxation (SOR) algorithm in order to achieve the best numerical performance.

The outline of the paper is as follows: in Section 2, we shall formulate the mathematical model of the physical problem. In Section 3, the new numerical algorithm is described and numerical results are presented. In Section 4, we conclude the paper and point out some new research directions.

2. Formulation of the problem

In this section, we formulate the wet chemical etching problem that was considered by Vuik and Cuvelier (1985) and Bruch *et al.* (1993). Following their physical model, we shall place two photo-resist layers (masks) onto a flat plate of length L (Figure 1). We shall allow a gap of length $2a$ between the two layers and caustic fluid to flow on top of the plate. The thickness of the mask is infinitely thin and L is large in comparison with a . Therefore, we can regard this problem as two dimensional, and only the cross section (Figure 2) is considered.

The following notations are introduced in order to better understand this problem. Let D_1 be a square region in the caustic fluid which is large enough so that increasing the size of D_1 will not change the etching process. Let D_2 be a rectangular region in the

Figure 1.
The physical problem

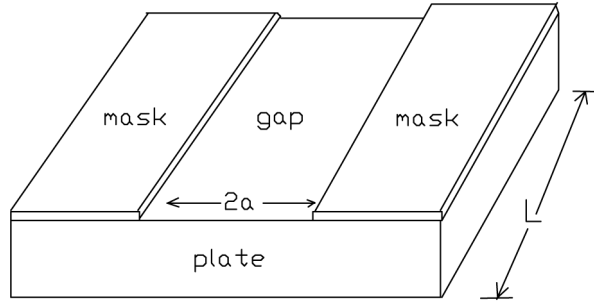
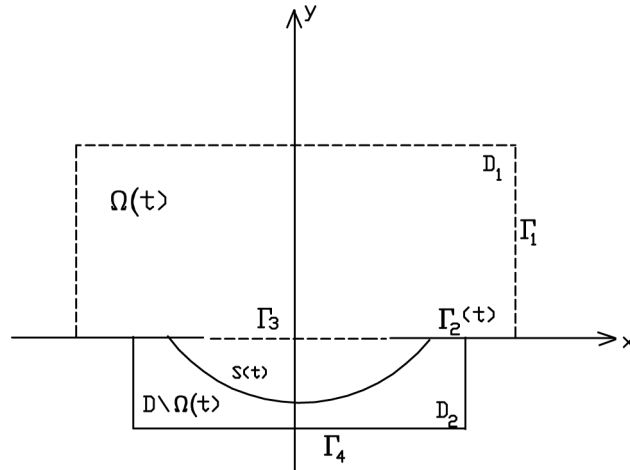


Figure 2.
The mathematical view



plate's cross section such that all the etching will occur in D_2 for $t \in [0, T]$. Therefore, the domain of the problem, D , is the union of D_1 and D_2 . Let Γ_3 be the slit of length $2a$ where the flow moves down slowly, Γ_2 be the upper and lower surfaces of the mask that are exposed to the caustic fluid, Γ_1 be the boundary of D_1 that is not along the mask and Γ_4 be the boundary of D_2 that is not along the mask. Finally, let $\Omega(t)$ be the wet region within D at time t . Then $\Omega(0) = D_1$ and $\Omega(t)$ is expanding gradually with $S(t)$ denoted as the moving boundary. The dry area within D is $D \setminus \Omega(t)$.

The problem can now be formulated as follows:

Find the concentration function $C = C(x, y, t)$ in $\Omega(t)$ and the moving boundary $S(t)$ such that:

$$\frac{\partial C}{\partial t} - \Delta C = 0 \quad \text{in } Q_\Omega = \{(x, y, t) | (x, y) \in \Omega(t), t \in (0, T)\} \quad (2.1)$$

with the following initial and boundary conditions:

$$C(x, y, 0) = 1 \quad \text{in } D_1 = \Omega(0) \quad (2.2)$$

$$C(x, y, t) = 1 \quad \text{on } \Gamma_{1t} = \Gamma_1 \times (0, T) \quad (2.3)$$

$$\frac{\partial C}{\partial n} = 0 \quad \text{on } \Gamma_{2t} = \{(x, y, t) | (x, y) \in \Gamma_2, t \in (0, T)\} \quad (2.4)$$

$$C = 0 \quad \text{on } S_t(t) = \{(x, y, t) | (x, y) \in S(t), t \in (0, T)\} \quad (2.5)$$

$$\frac{\partial C}{\partial n} = -Bv_n \quad \text{on } S_t(t) \quad (2.6)$$

where n denotes the unit normal vector on $\Gamma_2(t)$ pointing outward with respect to $\Omega(t)$, v_n is the normal velocity of the boundary $S(t)$ and B is a constant related to the etching process.

To eliminate the difficulty of dealing with moving boundary $S(t)$, the following fixed domain method using a Baiocchi type transformation has been used to transform the moving boundary problem into a fixed boundary one.

Define an extension \tilde{C} of C to a domain $Q_D = D \times (0, T)$ by:

$$\tilde{C}(x, y, t) = \begin{cases} C(x, y, t) & \text{in } Q_\Omega \\ 0 & \text{in } Q_D - Q_\Omega \end{cases} \quad (2.7)$$

We can also define $g(x, y, t)$ and $C_0(x, y)$ as follows:

$$g(x, y, t) = \begin{cases} 1 & \text{on } \Gamma_{1t} \\ 0 & \text{on } (\partial D \times (0, T)) - \Gamma_{1t} \end{cases} \quad (2.8)$$

$$C_0(x, y) = \begin{cases} 1 & \text{in } D_1 \\ 0 & \text{in } D_2 \end{cases} \quad (2.9)$$

Then, \tilde{C} will satisfy:

$$\frac{\partial \tilde{C}}{\partial t} - \Delta \tilde{C} = -B \frac{\partial \chi_\Omega}{\partial t} \quad \text{in } Q_D \quad (2.10)$$

$$\tilde{C} = g \quad \text{on } \partial D \times (0, T) - \Gamma_{2t} \quad (2.11a)$$

$$\frac{\partial \tilde{C}}{\partial n} = 0 \quad \text{on } \Gamma_{2t} \quad (2.11b)$$

$$\tilde{C}(x, y, 0) = C_0(x, y) \quad \text{in } D \quad (2.12)$$

where χ_Ω is defined to be the characteristic function for Q_Ω in Q_D . Then we define the following Baiocchi transformation:

$$w(x, y, t) = \int_0^t \tilde{C}(x, y, \tau) d\tau \quad (x, y) \in D, \quad t \in (0, T) \quad (2.13)$$

and w should satisfy:

$$\frac{\partial w}{\partial t} - \Delta w = C_0(x, y) + B(\chi_\Omega(x, y, 0) - \chi_\Omega(x, y, t)). \quad (2.14)$$

This leads to the following parabolic variational inequality:

$$\frac{\partial w}{\partial t} - \Delta w - f \geq 0 \quad \text{in } Q_D \quad (2.15)$$

$$w \geq 0 \quad \text{in } Q_D \quad (2.16)$$

$$\left(\frac{\partial w}{\partial t} - \Delta w - f \right) w = 0 \quad \text{in } Q_D \quad (2.17)$$

$$w(x, y, t) = \int_0^t g(x, y, \tau) d\tau = G(t) \quad \text{on } \partial D \times (0, T) - \Gamma_{2t} \quad (2.18a)$$

$$\frac{\partial w}{\partial n} = 0 \quad \text{on } \Gamma_{2t} \quad (2.18b)$$

$$w(x, y, 0) = 0 \quad \text{in } D \quad (2.19)$$

where based on equations (2.9) and (2.14),

$$f(x, y) = C_0(x, y) + B(\chi_\Omega(x, y, 0) - 1) = \begin{cases} 1 & \text{in } D_1 \\ -B & \text{in } D_2 \end{cases} \quad (2.20)$$

If w is a solution of equations (2.15)-(2.19), then $\tilde{C}(x, y, t) = \partial w / \partial t$ solves the original problem.

From equations (2.15)-(2.19), we can see that within Q_Ω (wet region), $w > 0$. Therefore, the equality part of equation (2.15) holds. Since, $D_1 \subset \Omega$, we have the following equation instead of inequality:

$$\frac{\partial w}{\partial t} - \Delta w - 1 = 0 \quad \text{in } Q_{D_1} \quad (2.21)$$

while in Q_{D_2} , the following equation holds:

$$w \left(\frac{\partial w}{\partial t} - \Delta w + B \right) = 0 \quad \text{in } Q_{D_2}, \quad (2.22)$$

which means that in D_2 , w satisfies either:

$$\frac{\partial w}{\partial t} - \Delta w + B = 0$$

in the wet region or $w = 0$ in the dry region.

Equations (2.21) and (2.22) show that the original etching problem has a different behavior in the two sub-regions D_1 and D_2 . Therefore, it is natural to construct

a DD algorithm to solve these two different sub-problems separately, as shown in Section 3.

3. Numerical schemes and results

In this section, we introduce a new numerical algorithm and apply it to the etching problem.

Since, D is the union of two non-overlapping regions D_1 and D_2 and D_1 is a region not containing the free surface, it is natural to construct an algorithm which can solve these two sub-problems, respectively.

We may have noticed that D is symmetric along the y -axis. Therefore, for pure computational purposes, we may focus on the right half region of D from now on. That is, D_1 and D_2 stand for the right upper and lower half region, respectively.

The fact that both equations (2.21) and (2.22) have a $\partial w/\partial t$ term prompts us to discretize the term via the implicit Crank-Nicholson scheme which guarantees numerical stability for any time step size. Equations (2.21) and (2.22) are discretized as follows:

Find a sequence $\{w(k\Delta t) = w(x, y, k\Delta t)\}$, where $k = 1, 2, \dots, K = T/\Delta t$ with Δt as the time step size, such that:

$$\frac{w(k\Delta t) - w((k-1)\Delta t)}{\Delta t} - \frac{1}{2}(\Delta w(k\Delta t) + \Delta w((k-1)\Delta t)) - f = 0 \quad \text{in } D_1$$

$$w(k\Delta t) \left(\frac{w(k\Delta t) - w((k-1)\Delta t)}{\Delta t} - \frac{1}{2}(\Delta w(k\Delta t) + \Delta w((k-1)\Delta t)) - f \right) = 0 \quad \text{in } D_2$$

$$w(k\Delta t) = G(k\Delta t) = \begin{cases} k\Delta t & \text{on } \Gamma_1 \\ 0 & \text{on } \Gamma_4 \end{cases}$$

Since, $\overline{D_1} \cap \overline{D_2} = \Gamma_2 \cup \Gamma_3$, how to compute the values of $w(k\Delta t) = w(x, y, k\Delta t)$ on Γ_3 and then transfer the values between D_1 and D_2 interactively becomes our major concern. To this end, we will apply the efficient non-overlapping DD method from Jiang *et al.* (2006) to this etching problem at each time step k and construct a new algorithm as follows:

Assume $w_1((k-1)\Delta t) = w_1(x, y, (k-1)\Delta t)$ and $w_2((k-1)\Delta t) = w_2(x, y, (k-1)\Delta t)$ are given from the previous time step $k-1$. Let $g_1^1 = g_2^1 = 0$ on Γ_3 . For the initial iteration, $w_1^0(k\Delta t) = 0$ in D_1 ; $w_2^0(k\Delta t) = 0$ in D_2 . Find $\{w_1^n(k\Delta t) = w_1^n(x, y, k\Delta t)\}$ and $\{w_2^n(k\Delta t) = w_2^n(x, y, k\Delta t)\}$, $n = 1, 2, \dots$ such that

Problem 3.1

$$\frac{w_1^n(k\Delta t) - w_1^n((k-1)\Delta t)}{\Delta t} - \frac{1}{2}(\Delta w_1^n(k\Delta t) + \Delta w_1^n((k-1)\Delta t)) - f = 0 \quad \text{in } D_1 \quad (3.1)$$

$$\frac{\partial w_1^n(k\Delta t)}{\partial n} + \alpha w_1^n(k\Delta t) = g_1^n \quad \text{on } \Gamma_3 \quad (3.1a)$$

$$\frac{\partial w_1^n(k\Delta t)}{\partial n} = 0 \quad \text{on } \Gamma_2 \quad (3.1b)$$

$$w_1^n(k\Delta t) = k\Delta t \quad \text{on } \Gamma_1 \quad (3.1c)$$

Problem 3.2.

$$w_2^n(k\Delta t) \left(\frac{w_2^n(k\Delta t) - w_2^n((k-1)\Delta t)}{\Delta t} - \frac{1}{2}(\Delta w_2^n(k\Delta t) + \Delta w_2^n((k-1)\Delta t)) - f \right) = 0 \quad \text{in } D_2 \quad (3.2)$$

$$\frac{\partial w_2^n(k\Delta t)}{\partial n} + \alpha w_2^n(k\Delta t) = g_2^n \quad \text{on } \Gamma_3 \quad (3.2a)$$

$$\frac{\partial w_2^n(k\Delta t)}{\partial n} = 0 \quad \text{on } \Gamma_2 \quad (3.2b)$$

$$w_2^n(k\Delta t) = 0 \quad \text{on } \Gamma_4 \quad (3.2c)$$

Problem 3.3. Compute:

$$g_1^{n+1} = 2\alpha w_2^n(k\Delta t) - g_2^n, \quad g_2^{n+1} = 2\alpha w_1^n(k\Delta t) - g_1^n \quad \text{on } \Gamma_3 \quad (3.3)$$

and then solve Problem 3.1 and 3.2 iteratively with n replaced by $n + 1$.

Remark 3.1. g_1^n and g_2^n perform like a combination of the normal derivative and the function value. For each iteration, they are updated by equation (3.3) which is only concerned with the function value instead of the normal derivative. α is a parameter which can be set to any positive real value. We will conduct a series of numerical tests to choose the optimal value for the best performance.

We notice that in equations (3.1) and (3.2), we still have the continuous operator Δw . In order to obtain the full numerical scheme to the etching problem, we will apply the 2nd-order finite difference scheme to Δw and obtain the discrete formula for equation (3.1) as follows:

$$\begin{aligned} & \frac{(w_1)_{i,j,k}^n - (w_1)_{i,j,k-1}^n}{\Delta t} - \frac{1}{2} \left[\frac{(w_1)_{i+1,j,k}^n + (w_1)_{i-1,j,k}^n - 2(w_1)_{i,j,k}^n}{h^2} \right. \\ & + \frac{(w_1)_{i,j+1,k}^n + (w_1)_{i,j-1,k}^n - 2(w_1)_{i,j,k}^n}{h^2} + \frac{(w_1)_{i+1,j,k-1}^n + (w_1)_{i-1,j,k-1}^n - 2(w_1)_{i,j,k-1}^n}{h^2} \\ & \left. + \frac{(w_1)_{i,j+1,k-1}^n + (w_1)_{i,j-1,k-1}^n - 2(w_1)_{i,j,k-1}^n}{h^2} \right] - f_{i,j,k} = 0 \quad \text{in } D_1 \end{aligned} \quad (3.4)$$

where D_1 is divided into a rectangular mesh with mesh size $\Delta x = \Delta y = h$, i, j are the row and column mesh point numbers, respectively. $(w_1)_{i,j,k}$, $1 \leq i \leq I_1$, $1 \leq j \leq J_1$, $1 \leq k \leq K$ stands for the w_1 values at time step k on all the mesh points (i, j) where $(0, 0)$ means the upper right corner of D_1 and (I_1, J_1) means

the lower left corner of D_1 . I_1 and J_1 will later be specified based on the actual size of D_1 .

Similarly, the boundary conditions equations (3.1a)-(3.1c) can also be discretized using the forward finite difference scheme on $\Gamma_2 \cup \Gamma_3$ as follows:

$$\frac{(w_1)_{i-1,j,k}^n - (w_1)_{i,j,k}^n}{h} + \alpha(w_1)_{i,j,k}^n = (g_1^n)_{i,j} \quad \text{on } \Gamma_3 \quad (3.4a)$$

$$\frac{(w_1)_{i-1,j,k}^n - (w_1)_{i,j,k}^n}{h} = 0 \quad \text{on } \Gamma_2 \quad (3.4b)$$

$$(w_1)_{i,j,k}^n = k\Delta t \quad \text{on } \Gamma_1 \quad (3.4c)$$

We take care of equations (3.2) and (3.2a)-(3.2c) in D_2 in a similar way and obtain:

$$\begin{aligned} (w_2)_{i,j,k}^n & \left(\frac{(w_2)_{i,j,k}^n - (w_2)_{i,j,k-1}^n}{\Delta t} - \frac{1}{2} \left[\frac{(w_2)_{i+1,j,k}^n + (w_2)_{i-1,j,k}^n - 2(w_2)_{i,j,k}^n}{h^2} \right. \right. \\ & + \frac{(w_2)_{i,j+1,k}^n + (w_2)_{i,j-1,k}^n - 2(w_2)_{i,j,k}^n}{h^2} + \frac{(w_2)_{i+1,j,k-1}^n + (w_2)_{i-1,j,k-1}^n - 2(w_2)_{i,j,k-1}^n}{h^2} \\ & \left. \left. + \frac{(w_2)_{i,j+1,k-1}^n + (w_2)_{i,j-1,k-1}^n - 2(w_2)_{i,j,k-1}^n}{h^2} \right] - f_{i,j,k} \right) = 0 \quad \text{in } D_2 \quad (3.5) \end{aligned}$$

$$\frac{(w_2)_{i+1,j,k}^n - (w_2)_{i,j,k}^n}{\Delta t} + \alpha(w_2)_{i+1,j,k}^n = (g_2^n)_{i,j} \quad \text{on } \Gamma_3 \quad (3.5a)$$

$$\frac{(w_2)_{i+1,j,k}^n - (w_2)_{i,j,k}^n}{\Delta t} = 0 \quad \text{on } \Gamma_2 \quad (3.5b)$$

$$(w_2)_{i,j,k}^n = 0 \quad \text{on } \Gamma_4 \quad (3.5c)$$

where D_2 is divided into a rectangular mesh with mesh size $\Delta x = \Delta y = h$, $(w_2)_{i,j,k}$, $1 \leq i \leq I_2$, $1 \leq j \leq J_2$, $1 \leq k \leq K$ stands for the w_2 values at time step k on all the mesh points (i, j) where $(0, 0)$ means the upper right corner of D_2 and (I_2, J_2) means the lower left corner of D_2 . I_2 and J_2 will later be specified based on the actual size of D_2 .

By using the above schemes, we can carry out our computation at each time step k as follows:

At first we scan through D_1 point-wisely from right to left and top to bottom. As mentioned above, D_1 stands for the upper right half region due to symmetry with the y -axis, and D_2 stands for the lower right half region. At each internal mesh point, equation (3.4) is used to compute $(w_1)_{i,j,k}^n$. When we reach the common boundary $\Gamma_2 \cup \Gamma_3$, equations (3.4a) and (3.4b) are then used for $(w_1)_{i,j,k}^n$ values. Meanwhile, we also scan through D_2 from left to right and top to bottom. Similarly, the $(w_2)_{i,j,k}^n$ values on the common boundary need to be found from equations (3.5a) and (3.5b) while those from the internal mesh points will be computed from equation (3.5), whose

implementation is different from equation (3.4) and will be specified shortly. This finishes one iteration at time step k .

Next update g_1^n and g_2^n on Γ_3 from equation (3.3) and repeat to scan through D_1 and D_2 as before. The iteration will stop at each time step k if the convergence criterion is met when:

$$\max_{i,j} |(w_1)_{i,j,k}^{n+1} - (w_1)_{i,j,k}^n| < \in \quad \text{and} \quad \max_{i,j} |(w_2)_{i,j,k}^{n+1} - (w_2)_{i,j,k}^n| < \in \quad (3.6)$$

where \in is some fixed positive constant.

During the computation, finite difference SOR is utilized in D_1 where a partial differential equation holds, while in D_2 which contains the moving boundary part, finite difference SOR with projection is used to make sure the w_2 value at each point is always non-negative.

Therefore, when applying the SOR in D_1 , equation (3.4) becomes:

$$(w_1)_{i,j,k}^{n-(1/2)} = \left(\frac{1}{\Delta t} + \frac{2}{h^2} \right)^{-1} \left\{ \frac{(w_1)_{i,j,k-1}}{\Delta t} + \frac{1}{2} \left[\frac{(w_1)_{i+1,j,k}^{n-1} + (w_1)_{i-1,j,k}^n}{h^2} \right. \right. \\ \left. \left. + \frac{(w_1)_{i,j+1,k}^{n-1} + (w_1)_{i,j-1,k}^n}{h^2} + \frac{(w_1)_{i+1,j,k-1} + (w_1)_{i-1,j,k-1} - 2(w_1)_{i,j,k-1}}{h^2} \right. \right. \\ \left. \left. + \frac{(w_1)_{i,j+1,k-1} + (w_1)_{i,j-1,k-1} - 2(w_1)_{i,j,k-1}}{h^2} \right] + f_{i,j,k} \right\} \quad (3.7a)$$

$$(w_1)_{i,j,k}^n = (w_1)_{i,j,k}^{n-1} + \bar{\beta} \left((w_1)_{i,j,k}^{n-\frac{1}{2}} - (w_1)_{i,j,k}^{n-1} \right) \quad (3.7b)$$

where $\bar{\beta}$ is the relaxation parameter.

Similarly, when applying the SOR with projection in D_2 , equation (3.5) becomes:

$$(w_2)_{i,j,k}^{n-(1/2)} = \left(\frac{1}{\Delta t} + \frac{2}{h^2} \right)^{-1} \left\{ \frac{(w_2)_{i,j,k-1}}{\Delta t} + \frac{1}{2} \left[\frac{(w_2)_{i+1,j,k}^{n-1} + (w_2)_{i-1,j,k}^n}{h^2} \right. \right. \\ \left. \left. + \frac{(w_2)_{i,j+1,k}^{n-1} + (w_2)_{i,j-1,k}^n}{h^2} + \frac{(w_2)_{i+1,j,k-1} + (w_2)_{i-1,j,k-1} - 2(w_2)_{i,j,k-1}}{h^2} \right. \right. \\ \left. \left. + \frac{(w_2)_{i,j+1,k-1} + (w_2)_{i,j-1,k-1} - 2(w_2)_{i,j,k-1}}{h^2} \right] + f_{i,j,k} \right\} \quad (3.8a)$$

$$(w_2)_{i,j,k}^n = \max \left(0, (w_2)_{i,j,k}^{n-1} + \bar{\beta} \left((w_2)_{i,j,k}^{n-(1/2)} - (w_2)_{i,j,k}^{n-1} \right) \right) \quad (3.8b)$$

The following data is used for the numerical example:

Let $\alpha = 1$, $\Delta x = \Delta y = h = 0.0125$, $I_1 = 321$, and $J_1 = 321$ for D_1 , $I_2 = 161$, and $J_2 = 161$ for D_2 . Therefore, the whole upper right and lower right regions have the size of 4×4 and 2×2 , respectively. Meanwhile, the middle slit opening Γ_3 has the size of 1, the etching constant $B = 10$, time step size $\Delta t = 0.25$, relaxation parameter $\bar{\beta} = 1.75$ and the tolerance is $\in = 1 \times 10^{-4}$. The free surface is defined

at time t to be the first value less than the above tolerance moving in a direction outward from $\Omega(t)$ in D_2 .

Figure 3 shows the moving boundary for $t = 2, 4, 8, 16, 20$ in D_2 obtained through the new algorithm. It exactly matches the numerical results from Bruch *et al.* (1993).

However, the combination of the new non-overlapping DD method in the etching region and the Crank-Nicholson scheme generates a better performance than those from Bruch *et al.* (1993). Table I shows the required number of iterations at each time $t = 2, 4, 8, 16, 20$ for our current algorithm and the algorithm from Bruch *et al.* (1993). We can see that the performance has been improved over Bruch *et al.* (1993).

The above numerical test uses the default value $\alpha = 1$ which means the normal derivative and the function value are treated with the same weight from the Robin boundary condition. It provides the most convenient computation but doesn't guarantee the best numerical performance. Next, we will conduct a series of numerical tests to pick up the best range of α with which the algorithm will converge with the minimum number of iterations.

We solve the above problem with the same numerical scheme but with different α values. By conducting numerous tests, we find that the best numerical performance can

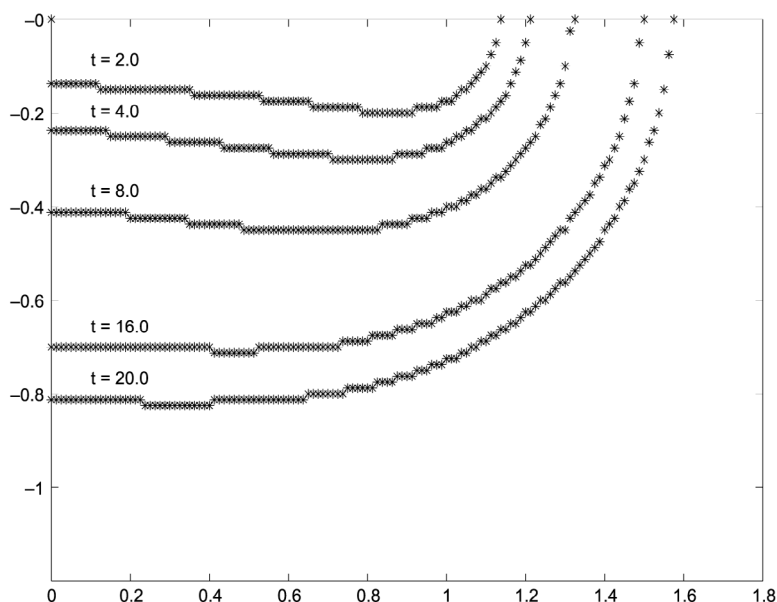


Figure 3.
Moving boundary at various times

| Time (t) | Current algorithm | Algorithm of Bruch <i>et al.</i> (1993) |
|--------------|-------------------|---|
| 2 | 656 | 790 |
| 4 | 637 | 786 |
| 8 | 631 | 789 |
| 16 | 691 | 807 |
| 20 | 704 | 818 |

Table I.
Comparison of required number of iterations to go from $t - 0.25$ to t between the two algorithms

be achieved when $2 < \alpha < 50$. When α is out of this range, the performance becomes poor. Therefore, only with moderate α values, the normal derivative and the function value can both make a good contribution towards the Robin boundary condition on the common boundary. The good performance can then be achieved due to the balance of these two factors. A too large or too small α value will significantly reduce the effect of either the normal derivative or the function value in the Robin boundary condition, which then leads to poor numerical performance. The result is verified from Table II where the numerical performance is compared when $\alpha = 0.5, 1, 2, 10, 50, 60$.

The following numerical tests also show that relaxation parameter $0 < \bar{\beta} < 2$ in the SOR algorithm can be set to 1.85 or its neighboring values for the best numerical performance. Table III compares the performance data for $\bar{\beta} = 0.5, 0.90, 1.50, 1.75, 1.85$ where $\alpha = 1$ is used.

Finally, let us discuss briefly the Crank-Nicholson scheme whose implicitness guarantees numerical stability for any time step size Δt . Since, the numerical error of Crank-Nicholson scheme is $O(\Delta t)^2 + h^2$, smaller Δt may generate smaller numerical error and thus faster convergence at each time step. However, it requires more time steps to reach the designated time, such as $t = 8, 16, 20$ in our problem. Therefore, our computation will not benefit from smaller Δt . This can be verified in Table IV where $\Delta t = 0.25, 0.125, 0.0625, 0.03125$. We can see that when Δt is reduced, the required number of iterations at each time step is also reduced significantly. However, the total number of required iterations will not be reduced at all due to the increased number of time steps.

Table II.
Comparison of required number of iterations to go from $t - 0.25$ to t for different α values

| Time (t) | $\alpha = 0.5$ | $\alpha = 1$ | $\alpha = 2$ | $\alpha = 10$ | $\alpha = 50$ | $\alpha = 60$ |
|--------------|----------------|--------------|--------------|---------------|---------------|---------------|
| 2 | 831 | 656 | 545 | 545 | 545 | 1,500 |
| 4 | 809 | 637 | 541 | 541 | 540 | 1,346 |
| 8 | 827 | 631 | 541 | 541 | 541 | 1,467 |
| 16 | 887 | 691 | 550 | 550 | 550 | 1,609 |
| 20 | 910 | 704 | 555 | 556 | 556 | 1,680 |

Table III.
Comparison of iterations numbers to go from $t - 0.25$ to t for different relaxation parameters

| Time (t) | $\bar{\beta} = 0.50$ | $\bar{\beta} = 0.90$ | $\bar{\beta} = 1.50$ | $\bar{\beta} = 1.75$ | $\bar{\beta} = 1.85$ |
|--------------|----------------------|----------------------|----------------------|----------------------|----------------------|
| 2 | 1,851 | 1,915 | 1,140 | 656 | 427 |
| 4 | 1,927 | 1,963 | 1,099 | 637 | 416 |
| 8 | 2,756 | 2,018 | 1,090 | 631 | 417 |
| 16 | 3,439 | 2,443 | 1,223 | 691 | 452 |
| 20 | 3,538 | 2,528 | 1,256 | 704 | 462 |

Table IV.
Comparison of iterations numbers to go from $t - \Delta t$ to t for different Δt

| Time (t) | $\Delta t = 0.25$ | $\Delta t = 0.125$ | $\Delta t = 0.0625$ | $\Delta t = 0.03125$ |
|--------------|-------------------|--------------------|---------------------|----------------------|
| 2 | 656 | 332 | 179 | 106 |
| 4 | 637 | 322 | 177 | 118 |
| 8 | 631 | 332 | 197 | 132 |
| 16 | 691 | 363 | 214 | 145 |
| 20 | 704 | 370 | 219 | 149 |

4. Conclusion

In this paper, we solved the chemical etching problem by using a new non-overlapping DD method and a finite difference technique with projection in a fixed domain and the implicit Crank-Nicholson scheme for the time discretization. The numerical results have shown the advantage of this approach. Because the original problem can be solved in its two sub-domains, respectively, we can easily implement the parallelized version of this numerical method on multiple CPU's and gain further improvement. Our future work will be concerned with providing a convergence analysis for the non-overlapping DD method applied to the parabolic variational inequality, which has not been reported in the literature.

References

- Bruch, J.C. Jr, Papadopoulos, C.A. and Sloss, J.M. (1993), "Parallel computing used in solving wet chemical etching semiconductor fabrication problems", *GAKUTO International Series, Mathematical Sciences and Applications*, Vol. 1, pp. 281-92.
- Deng, Q. (1997), "An analysis for a nonoverlapping domain decomposition iterative procedure", *SIAM Journal of Scientific Computation*, Vol. 18 No. 5, pp. 1517-25.
- Glowinski, R., Lions, J.L. and Tremolieres, R. (1980), *Numerical Analysis of Variational Inequalities*, North-Holland, Amsterdam.
- Jiang, B., Bruch, J.C. Jr and Sloss, J.M. (2006), "A non-overlapping domain decomposition method for variational inequalities derived from free boundary problems", *Numerical Methods for Partial Differential Equations*, Vol. 22 No. 1, pp. 1-17.
- Johnson, C. (1976), "A convergence estimate for an approximation of parabolic variational inequality", *SIAM Journal of Numerical Analysis*, Vol. 13 No. 4, pp. 599-606.
- Kuiken, H.K. (1984), "Etching through a slit", *Proceedings of Royal Society of London, A396*, pp. 95-117.
- Kuiken, H.K. (1990), "Mathematical modeling of etching processes", in Hoffmann, K.H. and Sprekels, J. (Eds), *Free Boundary Problems: Theory and Applications*, Vol. 1, Longman, Essex, pp. 89-109.
- Kuiken, H.K., Kelly, J.J. and Notten, P.H.L. (1986), "Etching profiles at resist edges, I. Mathematical models for diffusion-controlled cases", *Journal of Electrochemical Society*, Vol. 133 No. 6, pp. 1217-26.
- Lions, P.L. (1990), "On the Schwarz alternating method, III. A variant for non-overlapping subdomains", *Domain Decomposition Methods for Partial Differential Equations*, SIAM, Philadelphia, PA, pp. 202-23.
- Notten, P.H.L., Kelly, J.J. and Kuiken, H. (1986), "Etching profiles at resist edges, II. Experimental confirmation of models using GaAs", *Journal of Electrochemical Society*, Vol. 133 No. 6, pp. 1226-32.
- Vuik, C. and Cuvelier, C. (1985), "Numerical solution of an etching problem", *Journal of Computational Physics*, Vol. 59, pp. 247-63.
- Vuik, C. and Cuvelier, C. (1990), "An L^2 -error estimate for an approximation of the solution of a parabolic variational inequality", *Numerische Mathematik*, Vol. 57, pp. 453-71.

Corresponding author

Bin Jiang can be contacted at: bjiang@pdx.edu