MULTIGRID METHODS FOR INTERFACE PROBLEMS*

LOYCE ADAMS† AND ZHILIN LI‡

Abstract. New multigrid methods are developed for the maximum preserving immersed interface method applied to elliptic and parabolic interface problems. For elliptic interface problems, the multigrid solver developed in this paper works while some other multigrid solvers do not. For diffusion and reaction equations, we have developed the second order maximum preserving finite difference scheme in this paper. We use the Crank-Nicolson scheme to deal with the diffusion part, and an explicit scheme for the reaction part. Numerical examples are also presented.

Key words. interface problems, diffusion and reaction equation, multigrid method, discontinuous coefficients, quadratic optimization.

AMS subject classifications. 65N06, 65N50

1. Introduction.

In [6], the maximum principle preserving scheme is proposed for elliptic interface problems. The resulting linear system of equations from the finite difference scheme has positive or negative definite symmetric part but may not be symmetric. Available multigrid solvers may not work efficiently to solve the resulting linear system. In this paper, we propose a multigrid method which is designed particularly for interface problems.

In this paper, we also develop a maximum principle preserving scheme for diffusion and reaction equations with a fixed and closed smooth interface Γ in the solution domain, see Fig. 1 for an illustration. The governing equations are

$$u_t + \mathbf{a}(\mathbf{x},t)\cdot\nabla u = \nabla\cdot(\beta\nabla u) + f, \quad \mathbf{x}\in\Omega = \Omega^+\cup\Omega^- - \Gamma, \tag{1.1}$$

$$[u]|_\Gamma = w(s), \quad [\beta u_n]|_\Gamma = v(s) \tag{1.2}$$

$$u(\mathbf{x},0) = u_0(\mathbf{x}), \quad u(\mathbf{x},t)|_{\partial\Omega} = g(t), \tag{1.3}$$

where the coefficients $\beta(\mathbf{x},t)\geq\beta_{min}>0$, $\mathbf{a}(\mathbf{x},t,u)=(\alpha_1,\alpha_2)$, $f(\mathbf{x},t)$ are piecewise continuous but may have a finite jump across Γ, and s is the arc-length parameterization of the interface Γ. The jump is defined as the difference of the limiting values of two different sides of the interface, for example

$$[u]_{|\mathbf{X}\in\Gamma} = \lim_{\mathbf{x}\to\mathbf{X},\mathbf{X}\in\Omega^+} u(\mathbf{x},t) - \lim_{\mathbf{x}\to\mathbf{X},\mathbf{X}\in\Omega^-} u(\mathbf{x},t) = u^+ - u^-.$$

For simplicity of the discussion, we assume a Dirichlet boundary condition for the solution and a rectangular domain Ω. The natural jump condition across the interface Γ is $[u]=0$, and $[\beta u_n]=0$ as in the example of heat conduction in a composite material without a source/sink along the interface. However, we make our discussion more flexible by allowing non-homogeneous jump conditions.

The solution to the interface problem is in $H^1(\Omega)$ but not in $H^2(\Omega)$. For many applications, for example, piecewise coefficients, it is reasonable to assume that the

*The first author was supported in part by a DOE grant DE-FG03-96ER25292. The second author was supported in part by an ARO grant, 39676-MA, and an NSF grant, DMS-96-26703

†Department of Applied Mathematics, University of Washington, Seattle, WA 98195. (adams@amath.washington.edu.)

‡Center for Research in Scientific Computation & Department of Mathematics, North Carolina State University, Raleigh, NC 27695. (zhilin@math.ncsu.edu.)

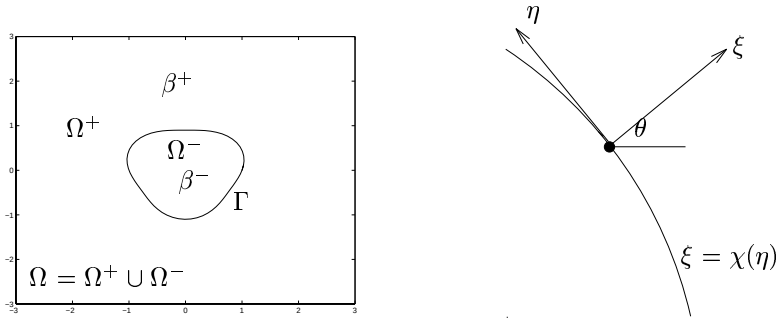


FIG. 1.1. *(a). A diagram of a rectangular domain $\Omega = \Omega^+ \cup \Omega^-$ with an immersed interface Γ. The coefficients such as $\beta(\mathbf{x})$ etc. may have a jump across the interface. (b). A diagram of the local coordinates in the normal and tangential directions, where θ is the angle between the x-axis and the normal direction.*

solution is piecewise smooth excluding the interface Γ. Typically, there is a jump in the normal derivative of the solution.

While a finite element method with a body-fitted grid can be used to solve the problem, we propose a numerical method that uses Cartesian grids. We refer the readers to the references in [6] for an incomplete overview of different numerical methods for elliptic interface problems. The potential advantage of Cartesian grid methods is to avoid the mesh generation process without sacrificing accuracy and stability. Our goal is to develop second order methods in contrast to the first order ghost fluid sharp interface method in [7].

2. The maximum principle preserving scheme for diffusion and reaction equations with an interface. The maximum principle preserving scheme for elliptic interface problems is proposed in [6] and will not be repeated here. We will explain the algorithm for the diffusion and reaction equation.

We assume the domain is $[a,\ b] \times [c,\ d]$, and use a uniform grid

$$x_i = a + ih, \quad i = 0, 1, \cdots, M; \qquad y_j = c + jh, \quad j = 0, 1, \cdots, N, \tag{2.1}$$

and denote the time step size by Δt.

2.1. The time discretization. The finite difference scheme is based on the prediction-correction Crank-Nicolson discretization

$$\frac{u^{n+1} - u^n}{\Delta t} + (\mathbf{a} \cdot \nabla_h u)^{n+\frac{1}{2}} = \frac{1}{2}\left((\nabla_h \cdot \beta \nabla_h u)^n + (\nabla_h \cdot \beta \nabla_h u)^{n+1}\right) + f^{n+\frac{1}{2}}, \tag{2.2}$$

where

$$(\mathbf{a} \cdot \nabla_h u)^{n+\frac{1}{2}} = \frac{3}{2}(\mathbf{a} \cdot \nabla_h u)^n - \frac{1}{2}(\mathbf{a} \cdot \nabla_h u)^{n-1}, \tag{2.3}$$

∇_h is the discrete gradient, and $t^{n+\frac{1}{2}} = t^n + \Delta t/2$. This discretization is second order accurate in time. The diffusion term is discretized implicitly so that we can take large time steps, while the reaction term is discretized explicitly so that second order accuracy can be achieved without affecting stability.

2.2. The spatial discretization. At a regular grid point (x_i, y_j) where all the points in the centered five-point stencil are on the same side of the interface, the discreization is the standard one

$$\nabla_h U_{ij} = \left\langle \frac{U_{i+1,j} - U_{i-1,j}}{2h}, \quad \frac{U_{i,j+1} - U_{i,j-1}}{2h} \right\rangle, \tag{2.4}$$

$$\begin{aligned}(\nabla_h \cdot \beta \nabla_h)\, U_{ij} = & \frac{1}{h}\left(\beta_{i+1/2,j}\frac{(U_{i+1,j} - U_{ij})}{h} - \beta_{i-1/2,j}\frac{(U_{i,j} - U_{i-1,j})}{h}\right) \\ & + \frac{1}{h}\left(\beta_{i,j+1/2}\frac{(U_{i,j+1} - U_{i,j})}{h} - \beta_{i,j-1/2}\frac{(U_{i,j} - U_{i,j-1})}{h}\right),\end{aligned} \tag{2.5}$$

where we have omitted the time index for simplicity.

At an irregular grid point where the centered five-point stencil consists of grid points from different sides of the interface Γ, the discretization is done using a method of determined coefficients. We let

$$(\mathbf{a} \cdot \nabla_h U_{ij}) = \sum_{k=1}^{n_{s_2}} \bar{\gamma}_k U_{i+i_k, j+j_k} - Q_{ij}, \tag{2.6}$$

$$(\nabla_h \cdot \beta \nabla_h)\, U_{ij} = \sum_{k=1}^{n_{s_1}} \gamma_k U_{i+i_k, j+j_k} - C_{ij}, \tag{2.7}$$

where n_{s_1} and n_{s_2} are the number of grid points in the finite difference stencil, and U_{ij} is the approximation to the solution $u(x, y)$ in (1.1)-(1.3) at (x_i, y_j). The sum over k involves a finite number of points neighboring (x_i, y_j). So each i_k, j_k will take values in the set $\{0, \pm 1, \pm 2 \cdots\}$. The coefficients γ_k and the indices i_k, j_k will depend on (i, j), so these should really be labeled γ_{ijk}, etc., but for simplicity of notation we will concentrate on a single grid point (x_i, y_j) and drop these indices.

Our criteria is to choose the coefficients to minimize the truncation error and *maintain* the stability. The procedure to set the system of equations for the undetermined coefficients is discussed in the following subsections.

2.2.1. The local coordinates and interface relations. It is reasonable to assume that the solution is piecewise smooth. The discontinuities in the solution or/and the derivatives occur along the interface Γ.

In order to determine the finite difference coefficients γ_k, and $\bar{\gamma}_k$ at an irregular grid point (x_i, y_j), we choose the orthogonal projection (x_i^*, y_j^*) of (x_i, y_j) on the interface Γ, or any point $(x_i^*, y_j^*) \in \Gamma$ that is close to (x_i, y_j). The local coordinates in the normal and the tangential directions are

$$\begin{aligned}\xi &= (x - x_i^*)\cos\theta + (y - y_j^*)\sin\theta, \\ \eta &= -(x - x_i^*)\sin\theta + (y - y_j^*)\cos\theta.\end{aligned} \tag{2.8}$$

In a neighborhood of the point (x_i^*, y_j^*), the interface Γ can be parameterized as

$$\xi = \chi(\eta), \quad \text{with} \quad \chi(0) = 0, \quad \chi'(0) = 0. \tag{2.9}$$

The curvature of the interface at (x_i^*, y_j^*) is $\chi''(0)$.

Assume that the irregular grid point $(x_i, y_j) \in \Omega^-$. In our method, the discrete form of (2.2)-(2.3) is the approximation to the original differential equation at (x_i^*, y_j^*)

instead of (x_i, y_j). For example, $f^{n+\frac{1}{2}}$ is

$$\{f^-\}^{n+\frac{1}{2}} = \lim_{(x,y)\to(x_i^*,y_j^*),(x,y)\in\Omega^-} f(x, y, t^n + \Delta t/2).$$

By differentiating the jump conditions (1.2) along the interface with respect to the arc-length s, which is η in the local coordinates, together with (2.9), we have the following jump relations (see [4] for the derivation),

$$\begin{aligned}
&u^+ = u^- + w,\\
&u_\xi^+ = \rho\, u_\xi^- + \frac{v}{\beta^+},\\
&u_\eta^+ = u_\eta^- + w_s,\\
&u_{\eta\,\eta}^+ = u_{\eta\,\eta}^- + (1-\rho)\chi''\, u_\xi^- - \frac{\chi''}{\beta^+}\, v + w_{ss},\\
&u_{\xi\,\eta}^+ = \rho\, u_{\xi\,\eta}^- + \frac{{\beta_\eta}^- - {\beta_\eta}^+\rho}{\beta^+}\, u_\xi^- + (1-\rho)\chi''\, u_\eta^- - \frac{{\beta_\eta}^+}{(\beta^+)^2}\, v + \chi''\, w_s + \frac{v_s}{\beta^+},
\end{aligned} \tag{2.10}$$

where $\rho = \beta^-/\beta^+$, w_s and w_{ss} are the first and second order derivatives of w with respect to the arc-length s along the interface. Likewise, v_s is the first derivative of v with respect to the arc length parameter s. With the local coordinates, the original differential equation can be written as

$$u_t + \tilde{\alpha}_1 u_\xi + \tilde{\alpha}_2 u_\eta = \beta(u_{\xi\xi} + u_{\eta\eta}) + \beta_\xi u_\xi + \beta_\eta u_\eta + f, \tag{2.11}$$

where

$$\tilde{\alpha}_1 = \alpha_1 \cos\theta + \alpha_2 \sin\theta, \qquad \tilde{\alpha}_2 = -\alpha_1 \sin\theta + \alpha_2 \cos\theta, \tag{2.12}$$

and u_ξ and u_η are the directional derivative of u in the normal and tangential directions. Therefore

$$[u_t + \tilde{\alpha}_1 u_\xi + \tilde{\alpha}_2 u_\eta] = [\beta(u_{\xi\xi} + u_{\eta\eta}) + \beta_\xi u_\xi + \beta_\eta u_\eta + f]\,. \tag{2.13}$$

Using the jump relations in (2.10) and after some manipulations, we get

$$\begin{aligned}
u_{\xi\,\xi}^+ = {} & \rho u_{\xi\,\xi}^- + (\rho-1)\, u_{\eta\,\eta}^- + \left((\rho-1)\chi'' + \frac{\beta_\xi^- - \rho\beta_\xi^+ + \rho\tilde{\alpha}_1^+ - \tilde{\alpha}_1^-}{\beta^+}\right) u_\xi^-\\
& + \frac{[\tilde{\alpha}_2 - \beta_\eta]}{\beta^+}\, u_\eta^- + \frac{\tilde{\alpha}_2^+ - \beta_\eta^+}{\beta^+}\, w_s - w_{ss}\\
& + \frac{1}{\beta^+}\, w_t + \left(\frac{\chi''}{\beta^+} + \frac{\tilde{\alpha}_1^+ - {\beta_\xi}^+}{(\beta^+)^2}\right) v - \frac{[f]}{\beta^+}.
\end{aligned} \tag{2.14}$$

Thus we have expressed all the quantities of the solution u up to second order derivatives from $+$ side in terms of those from the $-$ side of the interface Γ.

2.2.2. Determining the discretization of the diffusion term using the maximum principle preserving scheme. As we stated earlier in this paper, if the irregular grid point $(x_i, y_j) \in \Omega^-$, we approximate the differential equation at

(x_i^*, y_j^*) instead of (x_i, y_j). The truncation error of the finite difference approximation (2.7) to the diffusion term in (1.1) is

$$T_{ij} = \sum_{k=1}^{n_{s_1}} \gamma_k \, u(x_{i+i_k}, y_{j+j_k}, t) - C_{ij} - (\nabla \cdot (\beta \nabla u))^- . \tag{2.15}$$

The truncation error is evaluated at (x_i^*, y_j^*). The grid points involved are from outside, Ω^+, and inside, Ω^-, of the closed interface Γ. Using the Taylor expansion at (x_i^*, y_j^*), we can write

$$\begin{aligned} u(x_{i+i_k}, y_{j+j_k}, t) = u(\xi_k, \eta_k, t) = u^\pm + \xi_k u_\xi^\pm + \eta_k u_\eta^\pm + \frac{1}{2}\xi_k^2 u_{\xi\xi}^\pm \\ + \xi_k \eta_k u_{\xi\eta}^\pm + \frac{1}{2}\eta_k^2 u_{\eta\eta}^\pm + O(h^3), \end{aligned}$$

where the $+$ or $-$ sign is chosen depending on whether (ξ_k, η_k) lies on the $+$ or $-$ side of Γ. Therefore the truncation error can be written as

$$\begin{aligned} T_{ij} = a_1 \, u^- + a_2 \, u^+ + a_3 \, u_\xi^- + a_4 \, u_\xi^+ + a_5 \, u_\eta^- + a_6 u_\eta^+ + a_7 \, u_{\xi\,\xi}^- + a_8 \, u_{\xi\,\xi}^+ \\ + a_9 \, u_{\eta\,\eta}^- + a_{10} \, u_{\eta\,\eta}^+ + a_{11} \, u_{\xi\,\eta}^- + a_{12} \, u_{\xi\,\eta}^+ - C_{ij} + \cdots - \Big(\nabla \cdot (\beta \nabla u) \Big)^- . \end{aligned}$$

The quantities $u^\pm$, $u_\xi^\pm$, $\cdots$, are the limiting values of the functions at (x_i^*, y_j^*) from $+$ side or $-$ side of the interface. The coefficients a_j depend only on the position of the stencil relative to the interface. They are independent of the functions u and f. If we define the index sets K^+ and K^- by

$$K^\pm = \{k : \ (\xi_k, \eta_k) \text{ is on the } \pm \text{ side of } \Gamma\},$$

then the a_j are given by

$$\begin{aligned} a_1 &= \sum_{k \in K^-} \gamma_k, & a_2 &= \sum_{k \in K^+} \gamma_k, & a_3 &= \sum_{k \in K^-} \xi_k \gamma_k, \\ a_4 &= \sum_{k \in K^+} \xi_k \gamma_k, & a_5 &= \sum_{k \in K^-} \eta_k \gamma_k, & a_6 &= \sum_{k \in K^+} \eta_k \gamma_k, \\ a_7 &= \frac{1}{2} \sum_{k \in K^-} \xi_k^2 \gamma_k, & a_8 &= \frac{1}{2} \sum_{k \in K^+} \xi_k^2 \gamma_k, & a_9 &= \frac{1}{2} \sum_{k \in K^-} \eta_k^2 \gamma_k, \\ a_{10} &= \frac{1}{2} \sum_{k \in K^+} \eta_k^2 \gamma_k, & a_{11} &= \sum_{k \in K^-} \xi_k \eta_k \gamma_k, & a_{12} &= \sum_{k \in K^+} \xi_k \eta_k \gamma_k. \end{aligned} \tag{2.16}$$

Using the interface relations (2.10) and (2.14), we eliminate the quantities from one side, say, the $+$ side using the quantities from the other side, say, the $-$ side, collect

terms to get an expression of the form

$$\begin{aligned} T_{ij} &= (a_1+a_2)\,u^- + \left\{ a_3 + \rho a_4 + a_8 \left((\rho-1)\chi'' + \frac{\beta_\xi^- - \rho\beta_\xi^+ + \rho\tilde{\alpha}_1^+ - \tilde{\alpha}_1^-}{\beta^+} \right) \right. \\ &\quad \left. + a_{10}\,(1-\rho)\chi'' + a_{12}\,\frac{\beta_\eta{}^- - \beta_\eta{}^+\rho}{\beta^+} - \beta_\xi^- \right\} u_\xi^- \\ &\quad + \left\{ a_5 + a_6 + a_8\frac{[\tilde{\alpha}_2 - \beta_\eta]}{\beta^+} + a_{12}(1-\rho)\,\chi'' - \beta_\eta{}^- \right\} u_\eta^- \\ &\quad + \left\{ a_7 + a_8\rho - \beta^- \right\} u_{\xi\,\xi}^- + \left\{ a_9 + a_{10} + a_8\,(\rho-1) - \beta^- \right\} u_{\eta\,\eta}^- \\ &\quad + \left\{ a_{11} + a_{12}\rho \right\} u_{\xi\,\eta}^- + (\hat{T}_{ij} - C_{ij}) + \cdots , \end{aligned} \tag{2.17}$$

where

$$\begin{aligned} \hat{T}_{ij} = a_2\,w &+ \left(a_6 + a_8\frac{\tilde{\alpha}_2^+ - \beta_\xi{}^+}{\beta^+} \right) w_s + (a_{10}-a_8)w_{ss} + \frac{a_8}{\beta^+}\,w_t \\ &+ \frac{1}{\beta^+}\left\{ a_4 + a_8\left(\chi'' + \frac{\tilde{\alpha}_1^+ - \beta_\xi{}^+}{\beta^+} \right) - a_{10}\chi'' - a_{12}\frac{\beta_\eta{}^+}{\beta^+} \right\} v \\ &+ a_{12}\,\frac{1}{\beta^+}\,v_s - a_8\frac{[f]}{\beta^+}. \end{aligned} \tag{2.18}$$

To minimize the magnitude of the truncation error, we should set

$$\begin{aligned} a_1 + a_2 &= 0 \\ a_3 + \rho a_4 + a_8\left((\rho-1)\chi'' + \frac{\beta_\xi^- - \rho\beta_\xi^+ + \rho\tilde{\alpha}_1^+ - \tilde{\alpha}_1^-}{\beta^+} \right) & \\ + a_{10}\,(1-\rho)\chi'' + a_{12}\,\frac{\beta_\eta{}^- - \beta_\eta{}^+\rho}{\beta^+} &= \beta_\xi{}^- \\ a_5 + a_6 + a_8\frac{[\tilde{\alpha}_2 - \beta_\eta]}{\beta^+} + a_{12}(1-\rho)\,\chi'' &= \beta_\eta{}^- \\ a_7 + a_8\,\rho &= \beta^- \\ a_9 + a_{10} + a_8\,(\rho-1) &= \beta^- \\ a_{11} + a_{12}\,\rho &= 0. \end{aligned} \tag{2.19}$$

If we can find $\{\gamma_k\}$ such that the linear system of equations is satisfied, we have a consistent discretization for the diffusion term but we can not guarantee the stability. The stability condition can be guaranteed, however, by enforcing the discrete maximum principle

$$\gamma_k \geq 0 \quad \text{if} \quad (i_k, j_k) \neq (0,0), \qquad \gamma_k < 0 \quad \text{if} \quad (i_k, j_k) = (0,0). \tag{2.20}$$

In order to solve (2.19)-(2.20), we set-up the following quadratic optimization problem to determine the discretization for the diffusion term

$$\min_\gamma \left\{ \frac{1}{2}\,\|\gamma - g\|_2^2 \right\}, \qquad s.t. \tag{2.21}$$

$$A\gamma = b, \quad \gamma_k \geq 0, \quad \text{if} \quad (i_k, j_k) \neq (0,0) \quad \gamma_k < 0, \quad \text{if} \quad (i_k, j_k) = (0,0), \tag{2.22}$$

where $g \in R^{n_{s_1}}$, and $A\gamma = b$ is the system of linear equations (2.19). Naturally we want to choose γ_k in a such a way that they become the coefficients of the standard five-point central difference scheme if $\beta^+ = \beta^-$ is a constant. This can be done by selecting the vector g as

$$\begin{aligned} g_k &= \frac{\beta_{i+i_k,j+j_k}}{h^2}, \quad \text{if} \quad (i_k, j_k) \in \{(-1,0),(1,0),(0,-1),(0,1)\}; \\ g_k &= -\frac{4\beta_{i,j}}{h^2}, \quad \text{if} \quad (i_k, j_k) = (0,0); \qquad g_k = 0, \quad \text{otherwise.} \end{aligned} \tag{2.23}$$

We choose $n_{s_1} = 9$ which seems to work very well. In [6], we numerically demonstrated that the solution to the optimization problem has a solution if h is sufficiently small. We use the ***QL*** code developed by K. Schittkowski [8] to solve the optimization problem.

Once we know the coefficients $\{\gamma_k\}$, the correction term then is $C_{ij} = \hat{T}_{ij}$, where $\hat{T}_{ij}$ depends on $\{\gamma_k\}$ and the jump conditions.

2.2.3. Determine the discretization for the reaction term. Borrowing an idea from the projection method for the Navier-Stokes equations, see [3] for example, we use an explicit discretization for the reaction term since it involves only first order derivatives of the solution. The truncation error of the finite difference approximation (2.6) to the reaction term at (x_i^*, y_j^*) is

$$T_{ij}^r = \sum_{k=1}^{n_{s_2}} \bar{\gamma}_k \, u(x_{i+i_k}, y_{j+j_k}, t) - Q_{ij} - (\mathbf{a} \cdot \nabla u)^- . \tag{2.24}$$

As we mentioned in [4, 6], we can require the truncation error to be $O(h)$ at irregular grid points without affecting second order accuracy. Therefore we use the standard centered five-point stencil ($n_{s_2} = 5$) and the linear system of equations is

$$\begin{aligned} a_1 + a_2 &= 0 \\ a_3 + \rho a_4 &= \tilde{\alpha}_1^- \\ a_5 + a_6 &= \tilde{\alpha}_2^- \end{aligned} \tag{2.25}$$

where we have neglected higher order terms of h. Note that the a_k's are literally defined in (2.16) with γ_k being substituted with $\bar{\gamma}_k$ and n_{s_1} being substituted with n_{s_2}. The solution to the linear system of equations is also different. The system is an under-determined system and the solution is defined as the least squares solution with the least square 2-norm. The correction term then is

$$Q_{ij} = a_2 \, w + a_6 \, w_s + \frac{a_4}{\beta^+} \, v, \tag{2.26}$$

where we have ignored higher order terms of h.

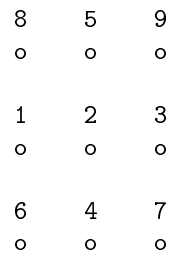
2.2.4. The CFL restriction. Since the reaction term is discretized using an explicit scheme and the maximum distance from (x_i^*, y_j^*) to (x_i, y_j) is $\sqrt{2}h$, the CFL restriction for the time step size is

$$\Delta t \le \frac{h}{\sqrt{2}\,\|\mathbf{a}\|_2}. \tag{2.27}$$

From the construction process of the method, we know that the scheme described in this section is consistent. The local truncation errors are of $O(h^2)$ at regular grid points, and of $O(h)$ at irregular grid points, or along the interface. We expect the global accuracy is of $O(h^2)$ since we can allow one order lower approximation along a boundary without affecting global accuracy. The stability of the scheme is guaranteed by the sign constraints of the scheme.

3. Multigrid method. In this section, we describe a multigrid method that is appropriate for solving the linear systems that arise from problems with internal interfaces that have been discretized using the methods described in Section 2 for parabolic problems from time t^n to t^{n+1}, and in [6] for elliptic interface problems using the maximum principle preserving schemes.

We denote the linear system to be solved on the finest grid as $A^h u^h = f^h$ where the left hand side of this equation at each grid point can be pictorially represented by the 9-point stencil with the numbering scheme given below:



The equation for the unknown at the center of the stencil is then given by

$$\gamma_1 u_1 + \gamma_2 u_2 + \gamma_3 u_3 + \gamma_4 u_4 + \gamma_5 u_5 + \gamma_6 u_6 + \gamma_7 u_7 + \gamma_8 u_8 + \gamma_9 u_9 = f_2 \tag{3.1}$$

where the γ's are the coefficients in the row of A^h corresponding to the unknown at the center of the stencil.

The components of the multigrid method are the smoother, the choice of coarse grid equations, the interpolation operator, and the restriction operator. We discuss each in turn below.

3.1. The Smoother. A simple point Gauss-Seidel smoother is used. On each grid level, the coarse grid points are smoothed first, the vertical edge fine grid points are smoothed second, the horizontal edge fine grid points are smoothed third, and the fine grid points in the center of a cell are smoothed last. The code has the option to continue smoothing (both pre- and post- smoothing) until stalling occurs or to smooth with a fixed number of pre- and a fixed number (can be a different number) of post-steps. For the problems we report in this paper, we used 2 pre- and 2 post-smooths. We found the post-smooths to be crucial to the success of the method.

3.2. The Coarse Grid Problem. Let $\tilde{u}^h$ be the approximation to u^h after pre-smoothing the problem $A^h u^h = f^h$. Then the error equation is $A^h e^h = r^h$, where the residual $r^h = f^h - A^h \tilde{u}^h$ must be approximated on a grid of size $2h$. The corresponding coarse grid equation is $A^{2h} e^{2h} = r^{2h}$ where the coarse grid operator is given by the usual Galerkin choice,

$$A^{2h} = R A^h I_{2h}^h \tag{3.2}$$

where R is the restriction operator, and I_{2h}^{h} is the interpolation operator. Likewise, the right hand side of the coarse grid equation is given by restricting the fine grid residual to the coarse grid by

$$r^{2h} = Rr^{h}. \tag{3.3}$$

3.3. The Interpolation Operator. We use two different interpolation schemes. One is for the regular grid points and one for the irregular points. Recall a grid point is termed irregular if its nine point stencil represents connections to grid points on different sides of the internal interface.

3.3.1. Interpolation at Regular Points. Many authors, (see for example Dendy, [2], and Briggs, et.al. [1] and the references therein), have developed an operator induced interpolation scheme. For two dimensional problems, given the values of the error e^{2h} at coarse grid points, we need to interpolate $I_{2h}^{h}e^{2h}$ to find an approximation to e^{h} at all fine grid points. For coarse grid points (the corners of grid cells), the value of e^{2h} is simply copied to be the value of e^{h}. For fine grid points that are on a vertical cell edge, the operator induced scheme would start with the error-residual equation associated with (3.1),

$$\gamma_1 e_1 + \gamma_2 e_2 + \gamma_3 e_3 + \gamma_4 e_4 + \gamma_5 e_5 + \gamma_6 e_6 + \gamma_7 e_7 + \gamma_8 e_8 + \gamma_9 e_9 = r_2 \approx 0, \tag{3.4}$$

where the residual, r_2, for this purpose is assumed small relative to the errors in the equation. This is a valid assumption for smooth error after the pre-relaxation step. An equation for e_2 in terms of e_4 and e_5 is desired if we are interpolating to e_2 by using the coarse grid error values at e_4 and e_5. Now, equation (3.4) could give such an approximation if all the errors on the left hand side could be estimated in terms of e_4 and e_5. At a regular point, all these neighbor grid points are on the same side of the internal interface, so it makes sense to use Taylor approximations to express e_1 and e_3 in terms of e_2, to express e_6 and e_7 in terms of e_4, and to express e_8 and e_9 in terms of e_5. This leads to the following equation for e_2 in terms of e_4 and e_5:

$$e_2 = c_4 e_4 + c_5 e_5 \tag{3.5}$$

where

$$c_4 = -(\gamma_6 + \gamma_4 + \gamma_7)/(\gamma_1 + \gamma_2 + \gamma_3) \tag{3.6}$$

and

$$c_5 = -(\gamma_8 + \gamma_5 + \gamma_9)/(\gamma_1 + \gamma_2 + \gamma_3). \tag{3.7}$$

The same strategy leads to an interpolation formula for the error at the center of a horizontal edge in terms of the coarse grid points on either side. We use

$$e_2 = c_1 e_1 + c_3 e_3 \tag{3.8}$$

where

$$c_1 = -(\gamma_6 + \gamma_1 + \gamma_8)/(\gamma_4 + \gamma_2 + \gamma_5) \tag{3.9}$$

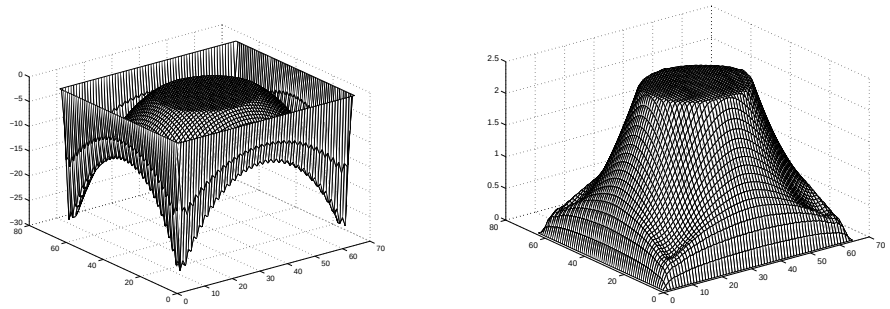


FIG. 3.1. *(a). The discrete error after pre-relax in V-cycle 1. (b). The discrete error after pre-relaxation in V-cycle 2. The PDE is given in Section 4.1 with* $b = .1$ *and* $C = 0$*. Notice the jump in the normal derivative of the error at the interface.*

and

$$c_3 = -(\gamma_7 + \gamma_3 + \gamma_9)/(\gamma_4 + \gamma_2 + \gamma_5). \tag{3.10}$$

Now, with the values determined at the coarse grid points (via copy) and the centers of vertical and horizontal edges by the formulas given above, the errors at the centers of each cell can be found by simply solving equation (3.4) for e_2 in terms of the other interpolated values. This can be written like an interpolation formula by casting the result for e_2 in terms of the four coarse grid error values at the corners of the cell.

3.3.2. Interpolation at Irregular Points. At irregular grid points, the interpolation formulas given in (3.5) and (3.8) are no longer valid since the *error* after the pre-relaxation step can also have a large jump in its normal derivative, and the approximations that were made in going from (3.4) to these formulas are not sufficient. This has been verified in practice for the test problems used in Section 4.1 where we know the exact solutions. In Figures 3.1 and 3.2, we plot the discrete error right after the pre-relax smoothing step is completed in the first, second, and tenth V-cycles. The original error has been smoothed but a large jump still remains at the interface and this continues throughout the computation. In Figure 3.2, we also plot the negative of the final discrete solution u^h which also has a jump in the normal derivative at the interface, a jump in β at the interface, but no jump in the value of βu_n.

The strategy we employ is to develop an interpolation formula for the midpoints of vertical and horizontal edges by using information about the interface and making the following assumptions about the discrete error.

1. The jump in the discrete error at the internal interface after pre-relaxation is 0. That is $[e^h] = 0$. This is reasonable since the relaxation will smooth enough for this to be true. This gives the relationship $e^+ = e^-$.
2. $[\beta e_n^h] = 0$ at the internal interface. In the problems we are solving, the value of $[\beta u_n]$ is given, and this information is built into the equations for u^h, so as the iteration proceeds the discrete solution will be an order h^2 approximation to u and will approximate this jump as well. This gives the relationship $e_\xi^+ = \frac{\beta^-}{\beta^+} e_\xi^-$.

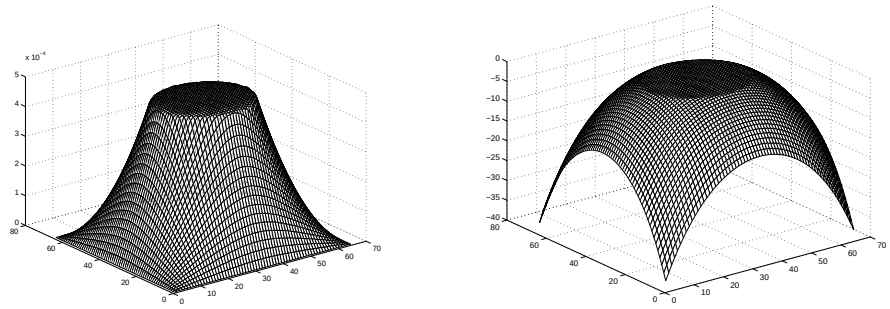


FIG. 3.2. *(a). The discrete error after pre-relax in V-cycle 10. (b). The value of* $-\tilde{u^h}$ *after V-cycle 10. The PDE is given in Section 4.1 with* $b = .1$ *and* $C = 0$*. Notice the jump in the normal derivative of the error at the interface and the jump in the solution at the interface.*

3. $[e^h_\eta] = 0$. This is also observed in practice, and the corresponding jump in the solution of the PDE, $[u_\eta]$ is also 0.
4. We can not assume that e^h_n will be smooth across the interface.

We now derive the interpolation formula for the midpoint of a vertical edge that cuts through the interface. We look for a formula for e_2 of the form (3.5). We first expand each side of (3.5) about a point (x^*, y^*) on the interface, taking care to use the values on the appropriate side of the interface. We center the coordinate system at this expansion point, and use the normal and tangential coordinates ξ and η given by (2.8). This leads to

$$e^- + \xi_2\rho_2 e^-_\xi + \eta_2 e^-_\eta = c_4(e^- + \xi_4\rho_4 e^-_\xi + \eta_4 e^-_\eta) + c_5(e^- + \xi_5\rho_5 e^-_\xi + \eta_5 e^-_\eta) \qquad (3.11)$$

to $O(h^2)$, where $\rho_i = 1$ if the grid point is inside or on the interface and $\rho_i = \frac{\beta^-}{\beta^+}$ if the grid point is outside the interface. A similar equation holds for horizontal midpoints with the subscripts 4 and 5 replaced by 1 and 3. Since the jump conditions have been applied, we now convert back to x and y coordinates using (2.8) and the relationships

$$e^-_\xi = e^-_x cos(\theta) + e^-_y sin(\theta) \qquad (3.12)$$

and

$$e^-_\eta = -e^-_x sin(\theta) + e^-_y cos(\theta). \qquad (3.13)$$

After doing this, we would like to be able to set the coefficients of e^-, e^-_x, and e^-_y to zero. This would give 3 equations but we only have the 2 unknowns (c_4 and c_5 for vertical midpoints, or c_1 and c_3 for horizontal midpoints). Before going further, we write down these three equations.

$$\begin{aligned} e^- &: c_4 + c_5 = 1 \\ e^-_x &: c_4(\xi_4\rho_4 c - \eta_4 s) + c_5(\xi_5\rho_5 c - \eta_5 s) = (\xi_2\rho_2 c - \eta_2 s) \\ e^-_y &: c_4(\xi_4\rho_4 s + \eta_4 c) + c_5(\xi_5\rho_5 s + \eta_5 c) = (\xi_2\rho_2 s + \eta_2 c) \end{aligned} \qquad (3.14)$$

In the equations above, c and s stand for $sin(\theta)$ and $cos(\theta)$, respectively.

For midpoints of vertical edges, we use the first and third equations; thereby forcing the coefficients of e^- and e_y^- to vanish. Note that we are hoping for the second equation to be nearly satisfied without its enforcement. This is definitely the case when no interface is present (all $\rho_i = 1$). This gives the following values for c_4 and c_5:

$$c_5 = \frac{(\rho_4 s^2 + c^2)(y^* - y_4) + (\rho_2 s^2 + c^2)(y_2 - y^*)}{(\rho_4 s^2 + c^2)(y^* - y_4) + (\rho_5 s^2 + c^2)(y_5 - y^*)} \tag{3.15}$$

and $c_4 = 1 - c_5$. The values s^2 and c^2 are $sin^2(\theta)$ and $cos^2(\theta)$, respectively.

For midpoints of horizontal edges, we use the first and second equations; thereby forcing the coefficients of e^- and e_x^- to vanish. Note that we are hoping for the third equation to be nearly satisfied without its enforcement. This is definitely the case when no interface is present. The values obtained for c_1 and c_3 are:

$$c_1 = \frac{(\rho_3 c^2 + s^2)(x^* - x_3) + (\rho_2 c^2 + s^2)(x_2 - x^*)}{(\rho_3 c^2 + s^2)(x^* - x_3) + (\rho_1 c^2 + s^2)(x_1 - x^*)} \tag{3.16}$$

and $c_3 = 1 - c_1$. The values s^2 and c^2 are $sin^2(\theta)$ and $cos^2(\theta)$, respectively.

This interpolation scheme allows the error to have jumps in the direction normal to the interface. The error is not assumed to vary strictly with y (for vertical interpolation) or strictly with x (for horizontal interpolation). The jump conditions relative to ξ and η are imposed.

The only remaining issue is how to interpolate the centers of the cells. Since we now have a formula for the cell corners (copy the coarse value) and the vertical and horizontal midpoints, we can solve (3.4) to find the value of e_2 for cell centers. This is the same strategy adopted for cell centers for regular points. Hence the overall scheme takes advantage of the interface information as well as that of the PDE operator.

3.4. The Restriction Operator. By discretizing the PDEs as described in Section 2, we guarantee that the problem A^h on the finest grid is diagonally dominant. This does not however, guarantee that A^{2h} is diagonally dominant. For problems with large jumps at the internal interface, the choice of $R = (I_{2h}^h)^T$ will not lead to a diagonally dominant problem or to a problem that is solved quickly with multigrid. What is true, however, is that simple injection will led to a diagonally dominant A^{2h} matrix. The proof is very simple and will not be presented here. We have found this choice for R, combined with our interpolation scheme to be very effective for these interface problems.

4. Numerical results. We have performed a number of numerical experiments. The computations are done using either a Sun's Ultra-1 or a Dec Alpha workstation. The linear system of equations is solved using the multigrid method. The interface is a closed curve in the solution domain and is expressed in terms of the periodic cubic spline interpolation developed in [5]. The implementation of the methods is sequential and not optimized. In this section, the following notations are used: $m = n$ is the number of grid lines in the x- and y- directions; n_1 is the number of control points used in the spline interpolation to represent the interface Γ; n_{coarse} and n_{finest} are

the number of the coarsest and finest grid lines respectively when the multigrid solver is used; n_l is the number of levels used for the multigrid method.

4.1. Numerical results for elliptic problems. First we present our multigrid results for elliptic interface problems. We consider the elliptic equation

$$\nabla \cdot (\beta(x,y)\nabla u(x,y)) \;=\; f(x,y). \tag{4.1}$$

In this example, the interface is the circle $x^2 + y^2 = \frac{1}{4}$ within the computation domain $-1 \le x,\, y \le 1$. The value of β is

$$\beta(x,y) = \begin{cases} x^2+y^2+1, & \text{if } x^2+y^2 \le \frac{1}{4}, \\ b, & \text{if } x^2+y^2 > \frac{1}{4}. \end{cases} \tag{4.2}$$

The source term is

$$f(x,y) = 8(x^2+y^2) + 4. \tag{4.3}$$

The true solution is

$$u(x,y) = \begin{cases} r^2, & \text{if } r \le \frac{1}{2}, \\ (1 - \frac{1}{8\,b} - \frac{1}{b})/4 + (\frac{r^4}{2} + r^2)/b + C\,\log(2r)/b, & \text{if } r > \frac{1}{2}. \end{cases} \tag{4.4}$$

In this example, we have *variable* and discontinuous coefficients. The jump conditions are

$$[u] = 0, \qquad [\beta u_n] = 2C, \qquad [\beta] = 5/4 - b, \qquad [f] = 0. \tag{4.5}$$

The difficulty of the problem can be controlled by decreasing the value of b since the jump in the normal derivative of the solution is given by

$$[u_n] = (2C + 5/4)/b - 1. \tag{4.6}$$

We use Dirichlet boundary conditions.

Tables 4.1, 4.2, and 4.3 show the results of the multigrid solver for various levels n_l and values of b. Note that in Table 4.2 where $b = 0.1$, both the solution and the flux are continuous, but β has a finite jump. The maximum error over all grid points,

$$\| E_n \|_\infty = \max_{i,j} \mid u(x_i, y_j) - U_{ij} \mid,$$

is presented. The order of convergence is computed from

$$\text{order} = \left| \frac{\log\left(\| E_{n_1} \|_\infty / \| E_{n_2} \|_\infty\right)}{\log(n_1/n_2)} \right|,$$

which is the solution of the equation

$$\| E_n \|_\infty = C\, h^{\text{order}}$$

with two different n's. The fourth column in the Tables gives the number of V-cycles to reach the convergence criteria, and the fifth column gives the average rate of multigrid convergence calculated as

$$\text{rate}(k) = e^{\frac{\log(||r^k||_2 / ||r^0||_2)}{k}}. \tag{4.7}$$

TABLE 4.1

Multigrid results. The parameters are $n_{coarse} = 8$, $C = 0.0$, $b = 1.25$. The discrete system was solved to a 2-norm residual tolerance of 10^{-6}. Constant number of iterations with grid size is confirmed.

n_{finest}	n_1	n_l	V's	Rate	$\| E_n \|_\infty$	Order
32	40	3	4	.02	$6.0420\ 10^{-3}$	
64	80	4	5	.02	$1.4554\ 10^{-3}$	2.0536
128	160	5	5	.02	$3.5598\ 10^{-4}$	2.0315
256	320	6	5	.02	$8.7707\ 10^{-5}$	2.0210
512	640	7	5	.02	$2.2204\ 10^{-5}$	1.9819

TABLE 4.2

Multigrid results. The parameters are $n_{coarse} = 8$, $C = 0.0$, $b = 0.10$. The discrete system was solved to a 2-norm residual tolerance of 10^{-6}. Constant number of iterations with grid size is confirmed.

n_{finest}	n_1	n_l	V's	Rate	$\| E_n \|_\infty$	Order
64	80	4	12	.21	$2.5960\ 10^{-2}$	
128	160	5	12	.21	$6.3981\ 10^{-3}$	2.0206
256	320	6	13	.21	$1.6322\ 10^{-3}$	1.9708
512	640	7	14	.21	$4.2681\ 10^{-4}$	1.9352

TABLE 4.3

Multigrid results. The parameters are $n_{coarse} = 16$, $C = 0.0$, $b = .005$. The discrete system was solved to a 2-norm residual tolerance of 10^{-6}. Constant number of iterations with grid size is confirmed.

n_{finest}	n_1	n_l	V's	Rate	$\| E_n \|_\infty$	Order
64	80	3	17	.29	$6.063\ 10^{-1}$	
128	160	4	21	.36	$1.449\ 10^{-1}$	2.065
256	320	5	32	.47	$3.619\ 10^{-2}$	2.002
512	640	6	42	.54	$9.148\ 10^{-3}$	1.983

We also solved the problem in Table 4.3 with an algebraic multigrid solver that used an incomplete Cholesky smoother. The number of iterations required to solve the problem was around 275 for all the levels, much worse than the results we obtained. Also, our multigrid solver was able to solve the problem when $b = 10^{-6}$ in 32 iterations with a rate of .43 on a grid of size 256×256 and in 36 iterations on a grid of size 256×256; whereas, the algebraic multigrid solver failed for this value of b. For a given value of $\beta(x, y)$, the V-cycle rates are constant as the problem size increases. Our experiments show the number of V-cycles is not quite constant as the ratio β^-/β^+ increases, but tends to vary linearly with the logarithm of the ratio. That is, V-cycles $= c_1 \log_{10}(\beta^-/\beta^+) + c_2$.

4.2. Numerical results for diffusion and reaction equations. In this example, the interface is again the circle $x^2 + y^2 = \frac{1}{4}$ within the computation domain $-1 \le x, y \le 1$. The heat conductivity β is the same as (4.2). Let $g(x, y)$ be the piecewise function defined in (4.4) and

$$f(x, y) = h(t) \left(\alpha_1 \frac{\partial g}{\partial x} + \alpha_2 \frac{\partial g}{\partial y} - 8 \left(x^2 + y^2\right) - 4 \right) + h'(t)\, g, \tag{4.8}$$

where $h(t)$ is a function of our choice. The jump conditions are

$$[u] = 0, \qquad [\beta u_n] = h(t)\, 2C. \tag{4.9}$$

The solution to the diffusion and reaction equations (1.1)-(1.3) is

$$u(x, y) = h(t) g(x, y). \tag{4.10}$$

We use Dirichlet boundary conditions and the initial condition from the exact solution. Table 4.4 shows the results of the grid refinement analysis of our method for different grid sizes. The linear system at each time step was solved using both the multigrid method described in Section 3 and an algebraic multigrid method that uses incomplete Cholesky smoothing. Both multigrid methods solved the systems equally well (requiring between 3 and 5 V-cycles). These systems were more diagonally dominant that the elliptic problems reported in Section 4.1.

TABLE 4.4

The grid refinement analysis. The parameters are $h(t) = \cos t$, $\alpha_1 = 100$, $\alpha_2 = 1$, $n_{coarse} = 9$, $C = 0.1$, $b = 10$, *the final time is* $t = 1.0$. *Second order convergence is confirmed.*

n_{finest}	n_1	n_l	$\| E_n \|_\infty$	order
32	40	3	$7.5602\ 10^{-4}$	
64	80	4	$1.3792\ 10^{-4}$	5.48216
128	160	5	$2.8496\ 10^{-5}$	2.2750
256	320	6	$6.5788\ 10^{-6}$	2.1147

A remark. The method here is recommended for modest convection coefficient $\|\mathbf{a}\|$ since this term was treated explicitly. If $\|\mathbf{a}\|$ is very large, then the time step is small. In this case, some may prefer to use a upwind scheme to deal with the convection term. But then the upwind method only has first order accuracy.

5. Conclusions. In this paper, we have developed a second order method that preserves the discrete maximum principle for a diffusion and reaction equation involving a fixed interface and presented a multigrid method for solving the discrete system of equations obtained from the maximum principle preserving scheme both for elliptic and parabolic partial differential equations.

REFERENCES

[1] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial.* SIAM Publication, 2000.

[2] Jr. J. E. Dendy. Black box multigrid. *J. Comput. Phys.*, 48:336–386, 1982.

[3] M-C. Lai and Z. Li. The immersed interface method for the navier-stokes equations with singular forces. *J. Comput. Phys.*, in press, 2001.

[4] R. J. LeVeque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.*, 31:1019–1044, 1994.

[5] Z. Li. *The Immersed Interface Method — A Numerical Approach for Partial Differential Equations with Interfaces.* PhD thesis, University of Washington, 1994.

[6] Z. Li and K. Ito. Maximum principle preserving schemes for interface problems with discontinuous coefficients. SIAM J. Sci. Comput., in press.

[7] X. Liu, R. Fedkiw, and M. Kang. A boundary condition capturing method for Poisson's equation on irregular domain. *J. Comput. Phys.*, 160:151–178, 2000.

[8] K. Schittkowski. QL-quadratic Programming, version 1.5, 1991. http://www.uni-bayreuth.de/departments/math/~kschittkowski/ql.htm.