Semigroup Linearization for Nonlinear Parabolic Equations

James F. Epperson
*Department of Mathematical Sciences, University of Alabama,
Huntsville, Alabama 35899*

We present an algorithm for nonlinear parabolic equations that uses a linear semigroup approach to decouple the nonlinearity, thus allowing simpler techniques to be used for the solution. The basic method is applicable only to a restricted class of problems, but can be extended with little loss of utility. Stability and error estimates are given, along with example computations.

1. INTRODUCTION

In this paper we present and analyze numerical methods for nonlinear parabolic problems in either one of the following forms:

$$\begin{aligned} u_t &= Lu + f(u), & x &\in \Omega \subset \mathbb{R}^m, \quad m \geq 2 \\ u(0,t) &= u(1,t) = 0, & x &\in \Gamma = \partial\Omega \\ u(x,0) &= u_o(x), & x &\in \bar{\Omega} \end{aligned} \tag{1}$$

or

$$\begin{aligned} u_t &= L\beta(u), & x &\in \Omega \subset \mathbb{R}^m, \quad m \geq 2 \\ u(0,t) &= u(1,t) = 0, & x &\in \Gamma = \partial\Omega \\ u(x,0) &= u_o(x), & x &\in \bar{\Omega}. \end{aligned} \tag{1'}$$

Here $-L$ is a symmetric, positive definite, uniformly elliptic operator, and Ω is an open, bounded domain with smooth boundary Γ. The method uses a linear semigroup approach to decouple the nonlinearities from the diffusion operator, thus leaving us with only scalar nonlinear equations to be solved at each time step. One advantage of having only scalar nonlinear equations is that the nonlinear solution step—which is what will seriously slow down the computation—can now be done completely in parallel. Moreover, even with a nonlinear diffusion operator we are still able to employ linear techniques in the actual calculations.

The basic idea here is to use the kernel, or Green's function, for the parabolic operator $\partial_t - L$ to formally solve Eq. (1) or (1′) in an integral form, then approximate the kernel operation and the resulting integration. This is, in some sense, a development of the short notes [1] and [2], but it also is related to the nonlinear semigroup approximations of Brezis *et al.* [3], used in treating generalized Stefan problems. However, our work here

CCC 0749-159X/91/020147-17$04.00

is somewhat more general. Similar ideas result in the "nonlinear Chernoff formula" discretizations [4–7].

The scheme presented here is very similar to the method of lines, but we have used a Green's function approach because of the relationship to computational Green's function methods as developed in Refs. [1, 2, 8–10]. Moreover, the method-of-lines approach tends to obscure the fact that one can perhaps approximate the kernel of the differential operator more easily than the exponential of a difference/element matrix. Also, we feel it is important to emphasize that the methods employed here are derived mainly by using Green's functions to formally invert part of the operator, and *then* discretize the formal solution formula; this is in direct contrast to most numerical approaches, which in some sense directly approximate the differential operator and then invert the approximation to obtain the (approximate) solution. Application of this approach to equations with nonlinear convective terms is considered in Ref. [11].

Although almost all of our analysis is done in $\mathbb{R}^1$, this is for simplicity only. For the most part the techniques presented here will generalize to multiple dimensions. In this connection see Refs. [8, 12] for the dimension-separating properties of the Green's function method.

The paper is organized as follows: In Sec. 2 we present the method as applied to Eq. (1) in order to show the salient features of the algorithm. Stability and accuracy (in ℓ^2) are established (under very mild hypotheses) for this case in Theorems 1 and 2. We then extend to more general operators of the form (1′) in Sec. 3, again obtaining stability and accuracy, this time in ℓ^∞ and $\ell^{1'}$. Example computations are given in Sec. 4.

We use the following notation: $\|\cdot\|_p, p = 1, 2$, or ∞, will refer to the usual ℓ^p vector norms, as well as the corresponding induced matrix norms. Vectors of values defined on a grid of points will be denoted by a subscript h, as in

$$\varphi_h = (\varphi_1, \varphi_2, \dots, \varphi_{N-1})^T.$$

Given a function φ defined on $(0,1)$, we will use the notation $\varphi(\cdot)$ for the vector of values $[\varphi(x_1), \varphi(x_2), \dots, \varphi(x_{N-1})]^T$. Therefore, the difference $\|\varphi(\cdot) - \varphi_h\|_2$ is well defined:

$$\|\varphi(\cdot) - \varphi_h\|_2^2 = \sum_{i=1}^{N-1} (\varphi(x_i) - \varphi_i)^2.$$

While the error theory of Secs. 2 and 3—particularly assumptions $(\mathbb{H}_4)$ and $(\mathbb{H}_4')$—appears to apply only to problems with smooth solutions and smooth nonlinearities, the methods are stable for a much wider class of problems and do give good results even in the absence of smoothness (e.g., Examples 2 and 3 of Sec. 4).

All computations for this paper were performed in FORTRAN on a Sun 4/260 workstation using NCAR Graphics for the figures.

2. THE MODEL PROBLEM

Consider a simpler version of Eq. (1), with simple constant coefficient diffusion:

$$\begin{aligned} u_t &= au_{xx} + f(u), \qquad 0 < x < 1 \\ u(0,t) &= u(1,t) = 0, \\ u(x,0) &= u_o(x). \end{aligned} \tag{2}$$

Let G be the Green's function for the linear part of Eq. (2), i.e.,

$$\varphi(x,t) = \int_o^1 G(x,t;\ \xi)\varphi_o(\xi)\,d\xi \tag{3}$$

if and only if

$$\begin{aligned} \varphi_t &= a\varphi_{xx}, \qquad 0 < x < 1 \\ \varphi(0,t) &= \varphi(1,t) = 0, \\ \varphi(x,0) &= \varphi_o(x). \end{aligned} \tag{4}$$

For simplicity define the kernel integration in Eq. (3) as $\mathscr{G}(\tau)\varphi_o = \varphi$, i.e.,

$$[\mathscr{G}(\tau)\varphi_o](x,t+\tau) = \int_0^1 G(x,\tau;\ \xi)\varphi_o(\xi,t)\,d\xi,$$

so that $\mathscr{G}(\tau)\varphi_o$ is a solution of the homogeneous linear heat equation having $\varphi_o(\cdot,t)$ as the initial data.

It is straightforward, then, to solve Eq. (2) in terms of $\mathscr{G}$ by the variation of parameters formula:

$$u(x,t) = [\mathscr{G}(t)u_o](x,t) + \left(\int_o^t \{\mathscr{G}(t-s)f[u(\cdot,s)]\}\,ds\right)(x,t).$$

Alternately, we can write the solution at time t in terms of the solution at time $t - \Delta t$ as follows:

$$u(x,t) = [\mathscr{G}(\Delta t)u(\cdot,t-\Delta t)](x,t) + \left(\int_{t-\Delta t}^t \{\mathscr{G}(t-s)f[u(\cdot,s)]\}\,ds\right)(x,t).$$

If we carefully use the trapezoid rule on the time integral, keeping in mind the δ-function behavior of G as the second argument goes to 0, then we get

$$\begin{aligned} u(x,t) - \frac{1}{2}\Delta t f[u(x,t)] = \Bigg(\mathscr{G}(\Delta t)\Bigg\{u(\cdot,t-\Delta t) \\ + \frac{1}{2}\Delta t f[u(\cdot,t-\Delta t)]\Bigg\}\Bigg)(x,t) \\ + \delta(x,t) \end{aligned} \tag{5}$$

where the truncation error $\delta(x,t)$ is defined by

$$\delta(x,t) = -\frac{1}{12}\Delta t^3 \frac{\partial^2}{\partial s^2}\left(\int_0^1 G(x,t-s;\,\xi) f[u(\xi,s)]\,d\xi\right)\Bigg|_{s=\eta}$$

for $\eta \in [t-\Delta t, t]$.

Converting Eq. (5) to a practical computational algorithm simply requires some means of approximating the kernel G, or, more fundamentally, the operator $\mathcal{G}$. While direct approximation of the kernel and/or the associated integration is worth considering [1, 2, 8, 10], it is also the case that any standard finite difference/element approximation of Eq. (4) also provides a means of approximating $\mathcal{G}$. This is what we will consider here. For simplicity, we will restrict our discussion to the finite difference case. The finite element case can be handled with only minimal changes.

Given a grid of points $\{x_i = ih \mid 0 \le i \le N, h = 1/N\}$ on $[0,1]$ and a time step $\Delta t > 0$, we can compute approximate solutions of Eq. (4) by solving the linear system associated with the difference approximation. If $v_h(0) \in \mathbb{R}^{N-1}$ is the vector of discrete initial values, then the approximate solution after a single time step is $v_h(\Delta t) = \mathcal{G}_h(\Delta t)v_h(0)$, where $\mathcal{G}_h(\Delta t)$ is the solution operator for the linear system. For the usual Crank-Nicolson–based schemes, $\mathcal{G}_h(\Delta t)$ is defined by

$$\mathcal{G}_h(\Delta t) = (I + \mu K)^{-1}(I - \mu K),$$

where K is symmetric and positive definite, and $\mu = a\Delta t/2h^2$. Consequently, we can approximate solutions of the original nonlinear problem (2) by solving the set of *decoupled, scalar* equations

$$u_i(t_{n+1}) - \frac{1}{2}\Delta t f[u_i(t_{n+1})] = g_i(t_{n+1}), \tag{6}$$

where $u_i(t_{n+1})$ approximates $u[x_i, (n+1)\Delta t]$ and $g_h(t_{n+1}) \in \mathbb{R}^{N-1}$ is defined by

$$g_h(t_{n+1}) = \mathcal{G}_h(\Delta t)v_h(t_n)$$

for

$$v_h(t_n) = u_h(t_n) + \frac{1}{2}\Delta t f[u_h(t_n)]; \tag{7}$$

here the $f(u_h)$ notation means a component-wise evaluation of the scalar function f.

The algorithm can therefore be stated quite succinctly as follows:

Given an initial vector $u_h(0)$:

1. Form $v_h(t_n)$ from $u_h(t_n)$ according to (7).
2. Compute $g_h(t_{n+1}) = \mathcal{G}_h(\Delta t)v_h(t_n)$, where $\mathcal{G}_h(\Delta t)$ is the solution matrix for a numerical approximation of Eq. (4). $(\mathbb{A})$
3. Solve Eqs. (6) to get $u_h(t_{n+1})$.
4. Return to step 1.

Note that we can carry out this algorithm using the same linear equation solver that we would use for a problem governed by the linear part of Eq. (2). Moreover, ($\mathbb{A}$) is immediately extendable to the more general problem (1):

$$\begin{aligned} u_t &= Lu + f(u), && x \in \Omega \subset \mathbb{R}^m, \quad m \geq 2 \\ u(x,t) &= 0, && x \in \Gamma = \partial\Omega \\ u(x,0) &= u_o(x), && x \in \bar{\Omega} \end{aligned}$$

where the matrix $\mathscr{G}_h(\Delta t)$ is now the solution operator for an approximation to the linear part of the new PDE.

The assumptions as to uniform grid and constant diffusion coefficient have been made heretofore mostly for the sake of simplicity. However, if they actually do hold, then the matrix K in the definition of $\mathscr{G}_h(\Delta t)$ has an especially simple form which we can take advantage of in this setting to employ FFT methods for the actual calculation. We have that

$$K = \text{tridiag}(-1, 2, -1)$$

so that the eigenvalue problem

$$Ky = \lambda y$$

can be easily solved in closed form, yielding

$$\lambda_i = 2[1 - \cos(i\pi/N)],$$

$$y_{i,j} = \frac{2 \sin(ij\pi/N)}{N}.$$

This allows us to write $\mathscr{G}_h(\Delta t)$ as

$$\mathscr{G}_h(\Delta t) = Q^T D Q,$$

where Q is the orthogonal matrix of eigenvectors and D is the diagonal matrix

$$D = \text{diag}\left(\frac{1 - \mu\lambda_i}{1 + \mu\lambda_i}\right)$$

for $\mu = a\Delta t/2h^2$. Note that, since K is symmetric and positive definite, we have $\lambda_i > 0$, so that each diagonal of D satisfies

$$|d_{ii}| < 1,$$

and this is true for more general linear operators than for those used in our example.

Because of the special nature of the eigenvectors, multiplication by Q (and hence, because of symmetry, Q^T) is essentially the same as a Fourier sine transform:

$$z = Qv \longleftrightarrow z_i = \frac{2}{N} \sum_{m=1}^{N-1} v_m \sin\left(\frac{im\pi}{N}\right)$$

and can therefore be carried out by an appropriately modified FFT. We now have the algorithm:

```
Given an initial vector $u_h(0)$,
1. Form $v_h(t_n)$ from $u_h(t_n)$ according to Eq. (7).
2. Compute $\mathcal{S}[v_h(t_n)]$, the Fourier sine transform of $v_h(t_n)$.
3. Compute $D\mathcal{S}[v_h(t_n)]$.                                          (𝔹)
4. Compute $g_h(t_{n+1}) = \mathcal{S}\{D\mathcal{S}[v_h(t_n)]\}$.
5. Solve Eqs. (6) to get $u_h(t_{n+1})$.
6. Return to step 1.
```

We now turn our attention to the stability and error properties of the method in either form (𝔸) or (𝔹). To this end, we will need to assume certain reasonable properties of the linear approximation represented by $\mathcal{G}_h(\Delta t)$, and some bounds on $f'(u) = \partial f/\partial u$ relative to the time step Δt:

$\mathcal{G}_h(\Delta t)$ is symmetric and $\|\mathcal{G}_h(\Delta t)\|_2 \le 1 - C_G\,\Delta t$, for a constant $C_G \in (0, 1/\Delta t)$, independent of h and Δt; $(\mathbb{H}_1)$

$\|(\mathcal{G}\varphi)(\cdot, \Delta t) - \mathcal{G}_h(\Delta t)\varphi(\cdot)\|_2 \le C_s\,\Delta t(h^2 + \Delta t^2)$, for $C_s > 0$, independent of h and Δt; here φ is a known, smooth function defined on $[0,1]$, and the $(\mathcal{G}\varphi)(\cdot, \Delta t)$ notation means the vector in $\mathbb{R}^{N-1}$ formed from the $\{(\mathcal{G}\varphi)(x_i, \Delta t)\}$ values; $(\mathbb{H}_2)$

$f'(\eta)$ satisfies $f(0) = 0$, with $-M \le f'(\eta) \le 0$, for $M > 0$, and the time step Δt is such that $\Delta t \le 4/M$; $(\mathbb{H}_3)$

There exists a positive constant C_F, independent of h and Δt, such that $\|F(\cdot, t, s)\|_2 \le C_F$, for all (t, s), where $(\mathbb{H}_4)$

$$F(x,t,s) = \frac{\partial^2}{\partial s^2}\int_0^1 G(x, t-s;\ \xi) f[u(\xi, s)]\,d\xi.$$

Both of the first two hypotheses are reasonable for the usual linear parabolic approximation methods, e.g., Crank-Nicolson. The first one says that the approximate scheme is stable and decays over time; the second says that the approximate scheme is second-order accurate (the extra factor of Δt comes in because $(\mathbb{H}_2)$ is only bounding a *single* step error). The constant C_s in $(\mathbb{H}_2)$ will depend on the smoothness of the function φ, of course. The hypothesis $(\mathbb{H}_4)$ is essentially a bound on time derivatives of functions defined by solving the linear operator $\partial_t - L$, using $f(u)$ as initial data.

Our first result establishes the stability of the method under very mild hypotheses on the nonlinearity.

Theorem 1. *Assume $(\mathbb{H}_1)$ and $(\mathbb{H}_3)$; then there exists a constant Θ, $0 < \Theta < 1 - C_G\,\Delta t$, such that the approximation satisfies*

$$\|u_h(t_{n+1})\|_2 \le \Theta\|u_h(t_n)\|_2\,.$$

Proof. The proof is very simple. In vector form, Eqs. (6) and (7) become

$$u_h(t_{n+1}) - \frac{1}{2}\Delta t f[u\ (t_{n+1})] = \mathcal{G}_h(\Delta t)\left\{u_h(t_n) + \frac{1}{2}\Delta t f[u_h(t_n)]\right\}$$

or, assuming that f is smooth enough to use the mean-value theorem,

$$\left[I - \frac{1}{2}\Delta t D_f(t_{n+1})\right] u_h(t_{n+1}) = \mathscr{G}_h(\Delta t)\left[I + \frac{1}{2}\Delta t D_f(t_n)\right] u_h(t_n),$$

where $D_f(t_n) = \operatorname{diag}\{f'[\mu_1(t_n)], f'[\mu_2(t_n)], \ldots, f'[\mu_{N-1}(t_n)]\}$ and $f[u_i(t_n)] = f'[\mu_i(t_n)]u_i(t_n)$, for each i. Therefore,

$$u_h(t_{n+1}) = \left[I - \frac{1}{2}\Delta t D_f(t_{n+1})\right]^{-1} \mathscr{G}_h(\Delta t)\left[I + \frac{1}{2}\Delta t D_f(t_n)\right] u_h(t_n)$$

so that

$$\|u_h(t_{n+1})\|_2 \leq \Theta \|u_h(t_n)\|_2$$

and ($\mathbb{H}_1$) and ($\mathbb{H}_3$) guarantee that the constant is positive and less than or equal to $1 - C_G \Delta t$. If f is not smooth enough to justify using the mean value theorem, then a limiting argument using a sequence of smooth function f_m converging uniformly to f can be applied. ∎

Next, we derive an error estimate for the algorithm.

Theorem 2. *Let ($\mathbb{H}_1 - \mathbb{H}_4$) hold, and assume Theorem 1. Then there exists a constant $C > 0$, independent of h and Δt, such that*

$$\|u(\cdot, t_n) - u_h(t_n)\|_2 \leq C(h^2 + \Delta t^2)$$

for all t_n.

Proof. From Eq. (6) we have that the approximation satisfies

$$u_h(t_{n+1}) - \frac{1}{2}\Delta t f[u_h(t_{n+1})] = \mathscr{G}_h(\Delta t)\left\{u_h(t_n) + \frac{1}{2}\Delta t f[u_h(t_n)]\right\}.$$

But the exact solution $u(x, t)$ satisfies the similar equation

$$u(\cdot, t_{n+1}) - \frac{1}{2}\Delta t f[u(\cdot, t_{n+1})] = \mathscr{G}_h(\Delta t)\left\{u(\cdot, t_n) + \frac{1}{2}\Delta t f[u(\cdot, t_n)]\right\} + \varepsilon(\cdot, t_n),$$

where $\varepsilon(x_i, t_n)$ is the truncation error associated with $\mathscr{G}_h(\Delta t)$ and the trapezoid rule integration. It quickly follows, then, that

$$\left[1 - \frac{1}{2}\Delta t D_f(t_{n+1})\right][u(\cdot, t_{n+1}) - u_h(t_{n+1})] = \mathscr{G}_h(\Delta t)\left[1 + \frac{1}{2}\Delta t D_f(t_n)\right] \times [u(\cdot, t_n) - u_h(t_n)] + \varepsilon(\cdot, t_n)$$

so that, as in the proof of the stability theorem,

$$\|u(\cdot, t_{n+1}) - u_h(t_{n+1})\|_2 \leq \Theta \|u(\cdot, t_n) - u_h(t_n)\|_2 + \|\varepsilon(\cdot, t_n)\|_2. \qquad (8)$$

We now turn our attention to the truncation error term, $\|\varepsilon(\cdot, t_n)\|_2$. It is clear from the development of the algorithm that the truncation error has two parts, one due to the trapezoid rule integration, and one due to the error $\mathcal{G}(\Delta t) - \mathcal{G}_h(\Delta t)$:

$$\varepsilon(x_i, t_n) = \delta(x_i, t_n) + \varepsilon_G(x_i, t_n),$$

where $\delta(x_i, t_n)$ is the error due to the trapezoid rule and

$$\varepsilon_G(\cdot, t_n) = [\mathcal{G}(\Delta t) - \mathcal{G}_h(\Delta t)]\left\{u(\cdot, t_n) + \frac{1}{2}\Delta t f[u(\cdot, t_n)]\right\}.$$

Moreover, the hypothesis ($\mathbb{H}_2$) allows us to bound ε_G as

$$\|\varepsilon_G(\cdot, t_n)\|_2 \leq C_s \Delta t(h^2 + \Delta t^2).$$

Now recall that the error due to the trapezoid rule is

$$\delta(x_i, t_n) = -\frac{1}{12}\Delta t^3 \frac{\partial^2}{\partial s^2}\left\{\int_0^1 G(x_i, t_n - s; \xi) f[u(\xi, s)]\, d\xi\right\}\Bigg|_{s=\eta}$$

which, according to ($\mathbb{H}_4$), can be bounded as follows:

$$\|\delta(\cdot, t_n)\|_2 \leq \frac{\Delta t^3}{12} C_F.$$

Consequently, Eq. (8) becomes

$$\|u(\cdot, t_{n+1}) - u_h(t_{n+1})\|_2 \leq \Theta\|u(\cdot, t_n) - u_h(t_n)\|_2 + C\Delta t(h^2 + \Delta t^2),$$

where $\Theta \leq 1 - C_G \Delta t < 1$. Summing the recursion yields

$$\|u(\cdot, t_n) - u_h(t_n)\|_2 \leq \Theta^n\|u(\cdot, 0) - u_h(0)\|_2 + C\Delta t\left(\frac{1 - \Theta^{n+1}}{1 - \Theta}\right)(h^2 + \Delta t^2),$$

which becomes

$$\|u(\cdot, t_n) - u_h(t_n)\|_2 \leq \Theta^n\|u(\cdot, 0) - u_h(0)\|_2 + C(h^2 + k^2).$$

If we make the reasonable assumption that the initial error is zero, then the theorem is proved. ■

Remark. It is possible to essentially remove the condition $\Delta t \leq 4/M$, so that the method is stable and accurate for almost any time step. The condition $f' \leq 0$ remains essential, however.

3. MORE GENERAL OPERATORS

The numerical method developed above applies only to a fairly simple class of parabolic equations. In this section we extend the ideas of Section 2 to more general equations and again prove stability and error results. In contrast to the previous section, however, we will here obtain results in ℓ^∞ and ℓ^1, rather than ℓ^2. Similar methods are treated in Refs. [3–7].

Consider the evolution problem below, which is a simpler version of Eq. (1′):

$$\begin{aligned} u_t &= \beta(u)_{xx}, \qquad 0 < x < 1 \\ u(0,t) &= u(1,t) = 0, \\ u(x,0) &= u_o(x), \end{aligned} \tag{9}$$

where β is assumed to be Lipschitz with constant λ and $\beta' \geq 0$, with $\beta(0) = 0$. We can (formally) put this problem in the form (2) very simply [1], to get

$$\begin{aligned} a\beta(u)_t &= \beta(u)_{xx} + (a\beta(u)_t - u_t), \qquad 0 < x < 1 \\ u(0,t) &= u(1,t) = 0, \\ u(x,0) &= u_o(x). \end{aligned}$$

Here $a > 0$ will be specified below in order to obtain stability. From this we can produce an analog of Eq. (5) for this problem, namely:

$$\begin{aligned} \beta[u(x,t)] &= \{\mathscr{G}(\Delta t/a)\beta[u(\cdot,t - \Delta t)]\}(x,t) \\ &\quad + \frac{1}{a}\left(\int_{t-\Delta t}^{t} \{\mathscr{G}[(t - s)/a]F[u(\cdot,s)]\}\, ds\right)(x,t), \end{aligned} \tag{10}$$

where $F(u) = a\beta(u)_t - u_t$. Discretization of Eq. (10) is more delicate than was the case in Sec. 2, because the "source term" F involves time derivatives of the solution; hence, some kind of difference quotient will be required.

In order to avoid solving nonlinear *systems* of equations, we are led to a discretization having a lower-order truncation error in time; however, since the solutions to problems of the form in Eq. (9) are often of low regularity this is of less concern than if we were working with very smooth functions. Moreover, it leads to a numerical method that is stable and "nonlinearly explicit," i.e., the only systems of equations that are solved are linear systems. In any event, we opt to use the approximation

$$\psi'(t) = \frac{(\psi(t + \Delta t) - \psi(t)}{\Delta t} - \frac{1}{2}\Delta t\psi''(\tau), \qquad \tau \in [t, t + \Delta t]. \tag{11}$$

Because Eq. (11) is only first-order accurate, there is no point in using the trapezoid rule to approximate the integral; hence, we use the crude quadrature

$$\int_o^t g(s)\, ds = tg(0) + \frac{1}{2}t^2 g'(\tau), \qquad \tau \in [0,t]. \tag{12}$$

Using Eqs. (11) and (12), together with $\mathscr{G}_h(\Delta t)$ [again defined as the solution operator for an approximation of Eq. (4)], to discretize Eq. (10) yields the numerical method

$$u_i(t_{n+1}) = u_i(t_n) - a\beta[u_i(t_n)] + ag_i(t_{n+1}), \tag{13}$$

where $g_h(t_{n+1}) = [g_1(t_{n+1}), g_2(t_{n+1}), \ldots, g_{N-1}(t_{n+1})]^T$ is defined by

$$g_h(t_{n+1}) = \mathscr{G}_h(\Delta t)\beta[u_h(t_n)] \tag{14}$$

and $\beta[u_h(t_n)]$ denotes component-wise evaluation. As indicated above, the only system of equations which must be solved to compute Eq. (13) is the linear system in Eq. (14). The formal algorithm is as follows:

Given an initial function $u_h(0)$:

1. Form $\beta[u_h(t_n)]$.
2. Compute $g_h(t_{n+1}) = \mathscr{G}_h(\Delta t)\beta[u_h(t_n)]$, where $\mathscr{G}_h(\Delta t)$ is the solution matrix for an $O(h^2 + \Delta t)$ numerical approximation of Eq. (4). $(\mathbb{C})$
3. Compute $u_h(t_{n+1})$ from Eq. (13).
4. Return to step 1.

As in Sec. 2, the extension of this algorithm to problems governed by PDEs of the form $u_t - L\beta(u) = 0$ (where $-L$ is a symmetric, uniformly elliptic, positive definite operator) is straightforward.

To develop the stability and error theory for (C), we must slightly modify the assumptions from Sec. 2. In particular, we assume that $\mathscr{G}_h(\Delta t)$ comes from a discretization that is only $\mathbb{O}(h^2 + \Delta t)$ but which is monotone in a certain sense.

$\mathscr{G}_h(\Delta t) = (I + \mu K)^{-1}$ for K symmetric and positive definite, where $\mu = \Delta t/ah^2$; moreover, $\mathscr{G}_h(\Delta t)$ satisfies $\mathscr{G}_h(\Delta t) \geq 0$ and $\|\mathscr{G}_h(\Delta t)\|_\infty \leq 1$. $(\mathbb{H}_1')$

$\|\mathscr{G}\varphi)(\cdot, \Delta t) - \mathscr{G}_h(\Delta t)\varphi(\cdot)\|_\infty \leq C_s \Delta t(h^2 + \Delta t)$, for $C_s > 0$, independent of h and Δt. $(\mathbb{H}_2')$

β is Lipschitz continuous with constant λ and nondecreasing with $\beta(0) = 0$, $0 \leq a\lambda \leq 1$. $(\mathbb{H}_3')$

There exists a positive constant C_B, independent of h and Δt, such that $\|B(\cdot, t, s)\|_1 \leq C_B$ for all (t, s), where $(\mathbb{H}_4')$

$$B(x, t, s) = \frac{\partial}{\partial s}\int_0^1 G[x, (t - s)/a;\ \xi][a\beta(u)_t - u_t](\xi, s)\, d\xi.$$

Remark. The $(\mathbb{H}_1')$ hypothesis is much more specific than $(\mathbb{H}_1)$, but it still holds—as does $(\mathbb{H}_2')$—for the usual fully implicit schemes for linear parabolic equations, as we shall prove in Lemma 1, below. The hypothesis $(\mathbb{H}_4')$, similar to its analog in Sec. 2, essentially bounds derivatives of functions defined by solving the linear operator $\partial_t - a^{-1}L$, using $\beta_t(u)$ as the initial data.

Lemma 1. *Let $\mathscr{G}_h(\Delta t)$ be the solution operator for the fully implicit difference scheme as applied to the linear heat equation with diffusion coefficient $1/a$, on $[0, 1]$. Then $(\mathbb{H}_1')$ and $(\mathbb{H}_2')$ both hold.*

Proof. The PDE problem is

$$\begin{aligned} \varphi_t &= a^{-1}\varphi_{xx}, \\ \varphi(0,t) &= \varphi(1,t) = 0, \\ \varphi(x,0) &= \varphi_0(x). \end{aligned} \tag{15}$$

The fully implicit discretization applied to this problem is

$$(I + \mu K)\varphi_h(t_{n+1}) = \varphi_h(t_n), \tag{16}$$

where

$$\begin{aligned} K &= \text{tridiag}(-1,2,-1), \\ \mu &= \Delta t/ah^2, \\ \varphi_i(t_n) &\approx \varphi(x_i,t_n). \end{aligned}$$

Define $A = I + \mu K$, and note that A is obviously symmetric and positive definite. Moreover, the off-diagonal elements are all nonpositive. If we define $B = I - \text{diag}(A)^{-1}A$, where $\text{diag}(A)$ denotes the diagonal part of A, then it is easy to show that the spectral radius of B, $\rho(B)$, satisfies $\rho(B) < 1$. Therefore, it follows [13] that A is a Stieltjes matrix, hence $\mathcal{G}_h(\Delta t) = A^{-1}$ is non-negative.

Now define $u = Av$ for a pair of vectors u and v in $\mathbb{R}^{N-1}$. Therefore,

$$-\mu v_{i-1} + (1 + 2\mu)v_i - \mu v_{i+1} = u_i,$$

and suppose $|v_i| = \|v\|_\infty$. Then

$$(1 + 2\mu)|v_i| \le |u_i| + 2\mu\|v\|_\infty$$

so that

$$\|v\|_\infty = |v_i| \le |u_i| \le \|u\|_\infty,$$

which implies that $\|A^{-1}\|_\infty = \|\mathcal{G}_h(\Delta t)\|_\infty \le 1$.

Finally, we note that $(\mathbb{H}_2')$ is nothing more than a single-step error estimate for the scheme (16) applied to Eq. (15). The truncation error for Eq. 16 is $\mathcal{O}(h^2 + \Delta t)$, and the norm result from $(\mathbb{H}_1')$ constitutes a stability estimate, hence $(\mathbb{H}_2')$ follows immediately. ■

Remark. Our proof of Lemma 1 depended upon the choice of uniform grid over the interval (0, 1); however, the same argument applies to any grid over any domain as long as the stiffness matrix K is symmetric and positive definite with negative off-diagonal elements.

We can now prove the following stability theorem:

Theorem 3. *If $(\mathbb{H}_1')$ and $(\mathbb{H}_3')$ hold, then there exists a positive constant $\Theta \le 1$, such that $\|\beta(u_h(t_n))\|_\infty \le \Theta^n\|\beta(u_h(0))\|_\infty$; if β' is strictly positive, then Θ is strictly less than one.*

Proof. It follows from Lemma 1 that

$$0 \le 1 - \sum_{j=1}^{N-1} g_{ij} \le 1 \tag{17}$$

for all i, where $\{g_{ij}\}$ are the components of $\mathcal{G}_h(\Delta t)$. To establish the desired result, write the numerical scheme (13)–(14) in vector form as

$$u_h(t_{n+1}) = u_h(t_n) - a\beta[u_h(t_n)] + a\mathcal{G}_h(\Delta t)\beta[u_h(t_n)]. \tag{18}$$

Now assume that β is smooth enough to use the mean value theorem (if it is not, then a limiting argument can be used) and define the diagonal matrix D by

$$D = \operatorname{diag}[\beta'(\mu_1), \beta'(\mu_2), \ldots, \beta'(\mu_{N-1})],$$

where the $\beta'(\mu_i)$ satisfy

$$\beta[u_i(t_{n+1})] - \beta[u_i(t_n)] = \beta'(\mu_i)[u_i(t_{n+1}) - u_i(t_n)].$$

Then Eq. (18) becomes

$$\beta[u_h(t_{n+1})] = \beta[u_h(t_n)] - aD[I - \mathcal{G}_h(\Delta t)]\beta[u_h(t_n)]$$

so that

$$\|\beta[u_h(t_{n+1})]\|_\infty \le \|I - aD[I - \mathcal{G}_h(\Delta t)]\|_\infty \|\beta[u_h(t_n)]\|_\infty .$$

Therefore,

$$\Theta = \|I - aD[I - \mathcal{G}_h(\Delta t)]\|_\infty . \tag{19}$$

If $a\lambda \le 1$, then Eq. (19), together with Eq. (17), implies that $\Theta \le 1$. Equality is possible because $\beta'(\mu_i) = 0$ for all i is possible. However, if β' is strictly positive for at least one i, then we have $\Theta < 1$. ∎

Corollary. For any diagonal matrix D satisfying $0 \le \max_i ad_{ii} \le 1$, we have

$$\|I - aD[I - \mathcal{G}_h(\Delta t)]\|_\infty \le 1$$

with strict inequality if $\min_i d_{ii} > 0$.

A rigorous error estimate now follows:

Theorem 4. *If we assume* ($\mathbb{H}_1' - \mathbb{H}_4'$) *and Theorem 3 all hold, then for any $T > 0$, there exists a positive constant $C_T > 0$, independent of h and Δt, such that, for all $t_n \le T$,*

$$\|\beta[u(\cdot, t_n)] - \beta[u_h(t_n)]\|_1 \le C_T(h^2 + \Delta t).$$

Proof. The method of proof is analogous to that used in Theorem 2, modified slightly because of the use of ℓ^∞ and ℓ^1. The exact solution satisfies

$$u(x_i, t_{n+1}) = u(x_i, t_n) - a\beta[u(x_i, t_n)] + ag(x_i, t_{n+1}) + \varepsilon(x_i, t_{n+1}), \tag{20}$$

where $g(\cdot, t_{n+1}) = \mathcal{G}_h(\Delta t)\beta[u(\cdot, t_n)]$ and ε is the truncation error defined by

$$\begin{aligned}\varepsilon_i(x_i, t_{n+1}) &= \frac{1}{2}\Delta t^2\left[\frac{\partial}{\partial s}\int_0^1 G\left[x, \frac{(t-s)}{a}; \xi\right][a\beta(u)_t - u_t](\xi, s)\, d\xi\right] \\ &\quad - \frac{1}{2}\Delta t^2\{a\beta[u(x_i,\tau)] - u(x_i,\tau)\}_{ttt} \\ &\quad + a[\mathcal{G}(\Delta t) - \mathcal{G}_h(\Delta t)]\beta[u(\cdot, t_n)].\end{aligned}$$

Then, we can subtract Eqs. (13) and (20) to get, in vector form,

$$\begin{aligned}u(\cdot, t_{n+1}) - u_h(t_{n+1}) &= u(\cdot, t_n) - u_h(t_n) - a\{\beta[u(\cdot, t_n)] - \beta[u_h(t_n)]\} \\ &\quad + a\mathcal{G}_h(\Delta t)\{\beta[u(\cdot, t_n)] - \beta[u_h(t_n)]\} + \varepsilon(\cdot, t_{n+1}).\end{aligned}$$

For simplicity, define the vectors

$$\begin{aligned}\Delta u(t_n) &= u(\cdot, t_n) - u_h(t_n), \\ \Delta\beta(t_n) &= \beta[u(\cdot, t_n)] - \beta[u_h(t_n)]\end{aligned}$$

so that we have

$$\Delta u(t_{n+1}) = \Delta u(t_n) - a\,\Delta\beta(t_n) + a\mathcal{G}_h(\Delta t)\,\Delta\beta(t_n) + \varepsilon(\cdot, t_{n+1}) \tag{21}$$

We again assume that β is smooth enough to employ the mean value theorem, relying on a limiting argument should this not be the case. Define, then, the positive semidefinite diagonal matrix mapping from $\Delta u(\cdot)$ to $\Delta\beta(\cdot)$ by

$$\Delta\beta(t_n) = D(t_n)\,\Delta u(t_n)$$

so that Eq. (21) becomes

$$\Delta\beta(t_{n+1}) = D(t_{n+1})\{I - a[I - \mathcal{G}_h(\Delta t)]D(t_n)\}\Delta u(t_n) + D(t_{n+1})\varepsilon(\cdot, t_{n+1}).$$

For each n, define the matrix $\mathcal{P}_n = I - a[I - \mathcal{G}_h(\Delta t)]D(t_n)$ and solve the recursion backwards to get

$$\Delta\beta(t_n) = D(t_n)\left\{\prod_{i=0}^{n-1}\mathcal{P}_i\right\}\Delta u(0) + D(t_n)\varepsilon(\cdot, t_n) + D(t_n)\sum_{i=1}^{n-1}\left\{\prod_{j=1}^{n-1}\mathcal{P}_j\right\}\varepsilon(\cdot, t_i).$$

It follows from the Corollary to Theorem 1 that

$$\|\mathcal{P}_i\|_1 = \|\mathcal{P}_i^T\|_\infty = \|I - aD(t_i)[I - \mathcal{G}_h(\Delta t)]\|_\infty \leq 1$$

so that we have

$$\|\Delta\beta(t_n)\|_1 \leq \lambda[\|\Delta u(0)\|_1 + n\ \max_i\|\varepsilon(\cdot, t_i)\|_1]$$

and it remains only to bound the error term. The hypotheses $(\mathbb{H}_2')$ and $(\mathbb{H}_4')$, together with Eq. (11), imply that

$$\|\varepsilon(\cdot, t_i)\|_1 \leq C\Delta t(h^2 + \Delta t)$$

for all i, hence

$$n\|\varepsilon(\cdot, t_i)\|_1 \leq Ct_n(h^2 + \Delta t)$$

for all i, which is sufficient to complete the proof. ■

Remark. If β is strictly increasing, then the initial error term is multiplied by Θ^n for some positive constant $\Theta < 1$. Moreover, in this case the factor of t_n does not appear in the error bound so that the estimate is uniform over time.

4. EXAMPLE COMPUTATIONS

In this section we illustrate the methods described above by applying them to several simple examples.

Example 1. Consider the evolution problem (2)

$$u_t = u_{xx} - \frac{Vu}{K + u}, \qquad 0 < x < \pi$$

$$u(0,t) = u(\pi,t) = 0,$$

$$u(x,0) = u_o(x),$$

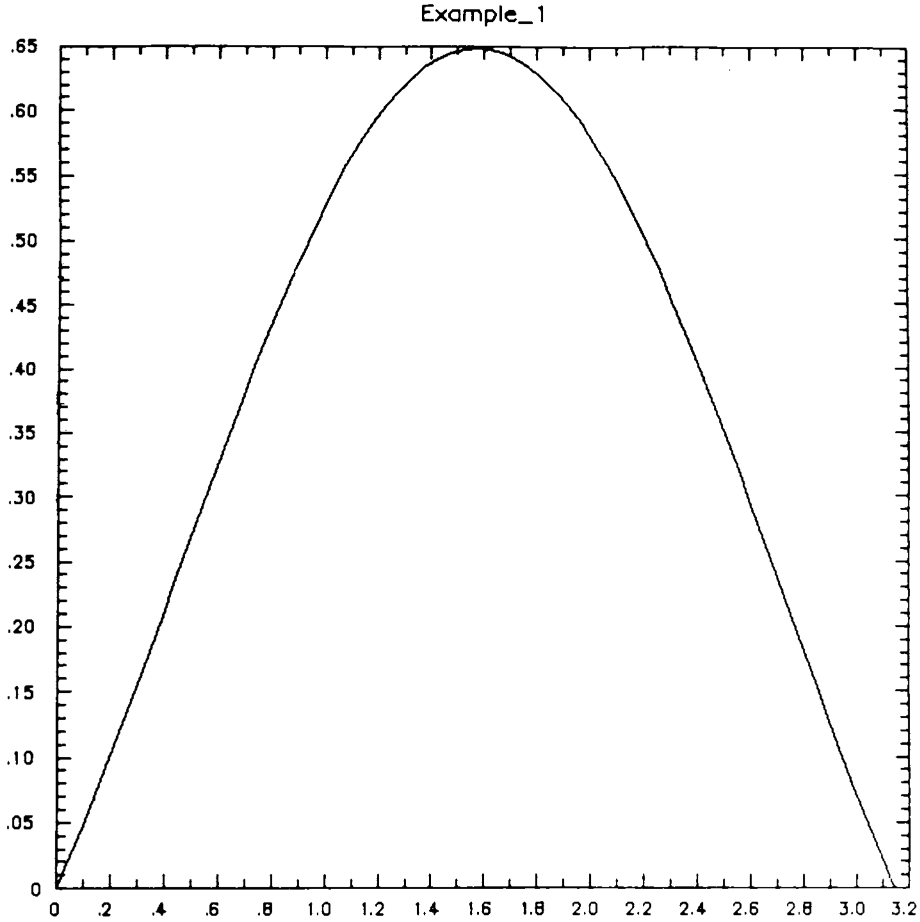

FIG. 1.

which has been used as a model of enzyme kinetics [14]. Application of our technique is quite straightforward; for $h = \pi/64$, $\Delta t = h/4$, and $u_o(x) = \sin x$, Fig. 1 gives the solution profile for $t = \pi/16$, using $V = 0.95$, $K = 0.05$.

Example 2. Consider the evolution problem (9), posed on $(0, \pi)$, with $\beta(u)$ defined as

$$\beta(u) = \begin{cases} u - 1, & 2 \le u \\ 1, & 1/2 \le u \le 2 \\ u/2, & u \le 1/2 \end{cases}$$

For this kind of β, Eq. (9) is an enthalpy formulation of the classical Stefan problem [15–17], which is a type of moving boundary problem. In this formulation, the temperature is β and the enthalpy is u. The melting temperature is $\beta(u) = 1$. Figure 2 shows both u and β, for $h = \pi/64$, $\Delta t = h/4$, and

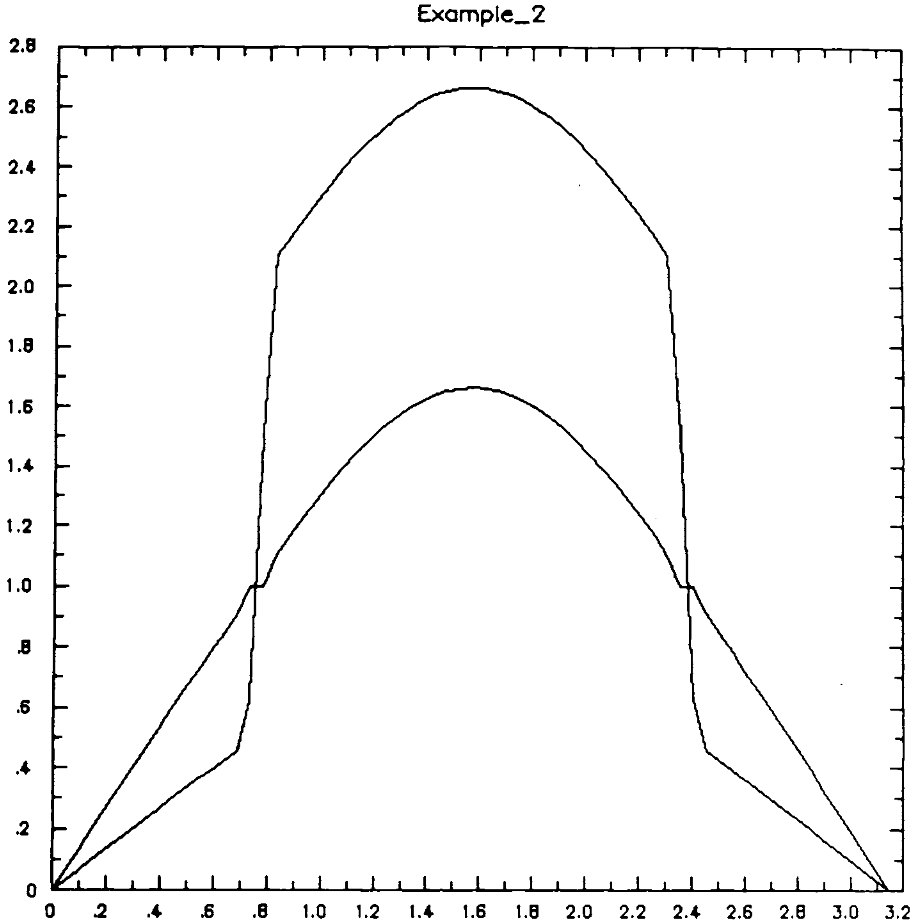

FIG. 2.

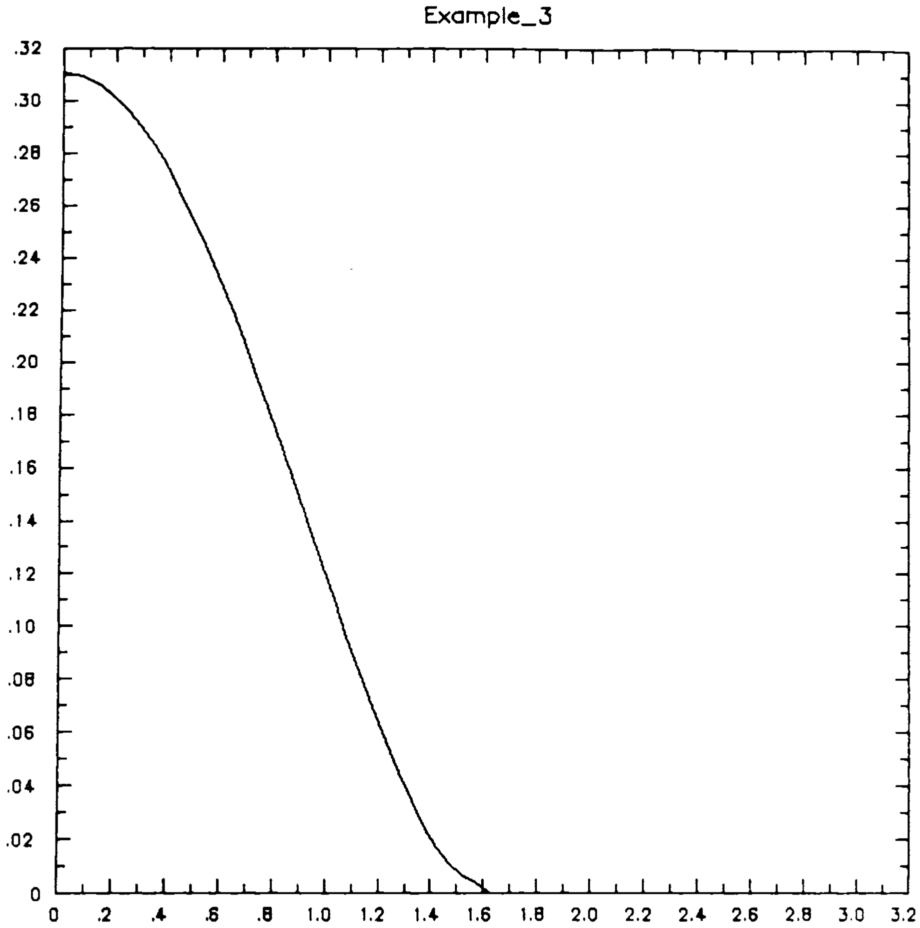

FIG. 3.

$t = \pi/16$. The initial enthalpy was taken to be

$$u_o(x) = \begin{cases} 0, & 0 \leq x \leq \pi/3 \\ 5, & \pi/3 < x < 2\pi/3 \\ 0, & 2\pi/3 \leq x \leq \pi \end{cases}$$

Note the small flat spots, or "mushy zones," in the graph of the enthalpy.

Example 3. We now pose an example with (homogeneous) Neumann data; in principle this is no different from the previous examples, since all of the changes are internal to the definition of the $\mathscr{G}_h(\Delta t)$ operator.

$$\begin{aligned} u_t &= u_{xx} - f(u), \qquad 0 < x < \pi \\ u'(0,t) &= u'(\pi,t) = 0, \qquad 0 < t \\ u(x,0) &= u_o(x), \end{aligned}$$

where

$$f(u) = \begin{cases} 1, & 0 < u \\ 0, & u \leq 0 \end{cases}$$

and

$$u_o(x) = \begin{cases} \cos^2 x, & 0 \leq x \leq 0.5\pi \\ 0, & 0.5\pi < x \leq \pi \end{cases}$$

This is an example of a diffusion-absorption problem similar to those studied in Ref. [14]. Figure 3 shows the solution profile for $h = \pi/64$, $\Delta t = h/2$, and $t = \pi/8$.

References

[1] J. Epperson, "On the use of Green's functions for approximating nonlinear parabolic PDE's," *Appl. Math. Lett.*, **2**(3), 293 (1989).

[2] J. Epperson, "Efficient computation with heat equation kernels," *SIAM J. Sci. Stat. Comput.*, to appear.

[3] H. Brezis, J. Rogers, and A. Berger, "A numerical method for solving the problem $u_t - \Delta f(u) = 0$," *RAIRO*, **13**(4), 297 (1979).

[4] E. Magenes, "Numerical approximation of non-linear evolution equations," preprint, 1988.

[5] E. Magenes, "A time discretization scheme approximating the nonlinear evolution equation $u_t + ABu = 0$," in F. Colombini, et al., Eds., *Partial Differential Equations and the Calculus of Variations,* Boston, Birhauser, 1989, pp. 743–766.

[6] E. Magenes and C. Verdi, "The semigroup approach to the two phase Stefan problem with nonlinear flux conditions," in A. Bossavit, D. Damlamian, and M. Fremond, Eds., *Free Boundary Problems: Applications and Theory, Vol. III,* Boston, Pitman, 1985, pp. 28–39.

[7] R. H. Nochetto and C. Verdi, "The combined use of a nonlinear Chernoff formula with a regularization procedure for two-phase Stefan problems," *Numer. Funct. Anal. and Opt.*, **9**, 1177 (1987–88).

[8] C. Brooks, Master's Thesis, University of Georgia, 1987.

[9] J. Epperson, "Computational Green's function methods for nonlinear parabolic PDE's," (unpublished).

[10] J. Epperson, "Applications of the fast Gauss transform to nonlinear parabolic equations," (unpublished).

[11] J. Epperson, "Kernel-based methods for parabolic equations with nonlinear convective terms," *J. Appl. Math. Comp.*, to appear.

[12] J. Epperson, "All parabolic equations are (numerically) one dimensional," (unpublished).

[13] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables,* Academic, San Diego, 1970.

[14] A. Berger, M. Ciment, and J. Rogers, "Numerical solution of a diffusion consumption problem with a free boundary," *SIAM J. Numer. Anal.*, **16**, 646 (1975).

[15] D. Atthey, "A finite difference method for melting problems," *J. Inst. Math. Appl.*, **13**, 353 (1974).

[16] G. Meyer, "Multidimensional Stefan problems," *SIAM J. Numer. Anal.*, **10**, 552 (1973).

[17] I. Rubinstein, *The Stefan Problem,* Providence, RI, American Mathematical Society, 1972.