SIAM J. SCI. COMPUT.
Vol. 33, No. 2, pp. 763–785

A CONTOUR INTEGRAL METHOD FOR THE BLACK–SCHOLES AND HESTON EQUATIONS*

K. J. IN 'T HOUT† AND J. A. C. WEIDEMAN‡

Abstract. A contour integral method recently proposed by Weideman [*IMA J. Numer. Anal.*, 30 (2010), pp. 334–350] for integrating semidiscrete advection-diffusion PDEs is improved and extended for application to some of the important equations of mathematical finance. Using estimates for the numerical range of the spatial operator, optimal contour parameters are derived theoretically and tested numerically. An improvement on the existing method is the use of Krylov methods for the shifted linear systems, the solution of which represents the major computational cost of the algorithm. A parallel implementation is also considered. Test examples presented are the Black–Scholes PDE in one space dimension and the Heston PDE in two dimensions, for both vanilla and barrier options. In the Heston case efficiency is compared to ADI splitting schemes, and experiments show that the contour integral method is superior for the range of medium to high accuracy requirements.

Key words. numerical contour integration, Laplace transform, matrix exponential, Krylov methods, parallelism, financial option pricing, Black–Scholes equation, Heston equation

AMS subject classifications. 65M20, 65L05, 65F60, 65R10, 91G60

DOI. 10.1137/090776081

1. Introduction.

In this paper we focus on initial value problems for linear systems of ODEs

$$\boldsymbol{u}_t = A\,\boldsymbol{u} + \boldsymbol{b}(t) \quad (0 \le t \le T), \qquad \boldsymbol{u}(0) = \boldsymbol{u}_0, \tag{1.1}$$

that arise from semidiscretizations of some of the prominent PDEs of mathematical finance. In particular, we consider the well-known Black–Scholes model as well as its more challenging two-dimensional generalization, the Heston model, for both vanilla and barrier options.

The $m \times m$ real matrix A in (1.1) represents a discrete approximation to an elliptic operator $\mathcal{A}$, and the $m \times 1$ real vectors $\boldsymbol{u}(t)$, $\boldsymbol{u}_0$, $\boldsymbol{b}(t)$ represent, respectively, the semidiscrete solution values at time t, the initial values, and boundary contributions. For the meaning of these variables in the finance context, we refer to sections 3 and 4 below. In this paper we use semidiscretizations based on finite differences, and hence A is assumed to be large and sparse.

Linear systems such as (1.1) can be integrated in time with, for example, Runge–Kutta methods, multistep methods, or operator splitting schemes. In the last decade or two, however, an alternative technique based on Laplace transformation and contour integration has gained popularity. We refer to [5, 7, 8, 22, 23] for general systems and to [13] for a specific application to mathematical finance.

We start with a brief introduction of the method. Let $\widehat{\boldsymbol{u}}(z)$ and $\widehat{\boldsymbol{b}}(z)$ denote, respectively, the Laplace transforms of $\boldsymbol{u}(t)$ and $\boldsymbol{b}(t)$. Define I as the $m \times m$ identity

*Received by the editors November 4, 2009; accepted for publication (in revised form) January 4, 2011; published electronically March 24, 2011.
http://www.siam.org/journals/sisc/33-2/77608.html
†Department of Mathematics and Computer Science, University of Antwerp, Middelheimlaan 1, B-2020 Antwerp, Belgium (karel.inthout@ua.ac.be).
‡Applied Mathematics, Stellenbosch University, Stellenbosch, 7600, South Africa (weideman@sun.ac.za).

matrix. Then, by taking the Laplace transform of (1.1), one obtains

$$\widehat{\boldsymbol{u}}(z) = \big(zI - A\big)^{-1}\big(\boldsymbol{u}_0 + \widehat{\boldsymbol{b}}(z)\big),$$

provided z is not part of the spectrum of A. The inverse is given by the Bromwich integral

$$\boldsymbol{u}(t) = \frac{1}{2\pi i}\int_\Gamma e^{zt}\widehat{\boldsymbol{u}}(z)\,dz, \tag{1.2}$$

a formula also associated with the names of Dunford and Taylor in the PDE context. The contour Γ is a simple, closed, positively oriented curve that encloses both the spectrum of A and all the singularities of $\widehat{\boldsymbol{b}}(z)$.

Talbot was one of the first authors to suggest the numerical integration of (1.2) as a computational tool for solving PDEs [27]. The key idea was to deform the contour Γ so that it starts and ends at $\operatorname{Re} z = -\infty$, while winding counterclockwise around the spectrum of A and the singularities of $\widehat{\boldsymbol{b}}(z)$. Such a contour is effective for computation, because the factor e^{zt} causes rapid decay of the integrand in (1.2). Functions of rapid decay are known to be well suited for fast and accurate integration by the simple trapezoidal or midpoint rules [16, 17, 26].

On a suitably parameterized contour, say,

$$\Gamma:\ z = z(\phi), \qquad -\infty < \phi < \infty,$$

formula (1.2) takes the form

$$\boldsymbol{u}(t) = \frac{1}{2\pi i}\int_{-\infty}^{\infty} e^{z(\phi)t}\widehat{\boldsymbol{u}}\big(z(\phi)\big)z'(\phi)\,d\phi. \tag{1.3}$$

Approximation of this integral by the trapezoidal rule yields

$$\boldsymbol{u}(t) \approx \frac{h}{2\pi i}\sum_{k=-\infty}^{\infty} e^{z_k t} z_k'\,\widehat{\boldsymbol{u}}_k, \tag{1.4}$$

where h is the uniform node spacing, and

$$\phi_k \equiv kh, \qquad z_k \equiv z(\phi_k), \qquad z_k' \equiv z'(\phi_k), \qquad \widehat{\boldsymbol{u}}_k \equiv \widehat{\boldsymbol{u}}(z_k). \tag{1.5}$$

For the midpoint rule, use $\phi_k = \big(k + \frac{1}{2}\big)h$.

In a practical computation the series is truncated. If one further assumes that the contour Γ is symmetric with respect to the real axis, the quadrature sum (1.4) can be approximated by

$$\boldsymbol{u}(t) \approx \frac{h}{\pi}\operatorname{Im}\left\{\sum_{k=0}^{N-1}{}' e^{z_k t} z_k'\,\widehat{\boldsymbol{u}}_k\right\}. \tag{1.6}$$

The primed sigma indicates that the first term in the sum should be halved (not implemented when the midpoint rule is used). The vectors $\widehat{\boldsymbol{u}}_k$ are solved from the N linear systems

$$\big(z_k I - A\big)\widehat{\boldsymbol{u}}_k = \boldsymbol{u}_0 + \widehat{\boldsymbol{b}}(z_k), \qquad k = 0, 1, \ldots, N-1. \tag{1.7}$$

Solving these so-called shifted linear systems represents the major computational cost of the algorithm. In comparison, the cost of the vector sum (1.6) is negligible. It is therefore important to use efficient strategies for solving these linear systems.

The first strategy is based on the observation that the systems (1.7) are independent and can therefore be solved in parallel. This has been pointed out in [7, 8, 22, 23] and elsewhere. A second approach, new in this context as far as we know, is the use of methods based on Krylov subspace iteration. These methods can be efficient for shifted systems such as (1.7) since only one Krylov basis has to be constructed for all the shifted systems. This strategy works best if the right-hand sides are independent of z, which is seldom the case because of the presence of $\widehat{\boldsymbol{b}}(z)$ in (1.7). In the next paragraph, we discuss a possible modification for circumventing this difficulty. The details of this Krylov method are given in the appendix.

One situation where the right-hand side is independent of z is when $\widehat{\boldsymbol{b}}(z) = \boldsymbol{0}$, i.e., $\boldsymbol{b}(t) = \boldsymbol{0}$ for all $t > 0$. In this case the solution to (1.1) reduces to $\boldsymbol{u}(t) = \exp(At)\boldsymbol{u}_0$, and then (1.6) represents a method for computing the matrix exponential times a given vector. When $\boldsymbol{b}(t)$ is nonzero but sufficiently simple, a modification again allows one to compute essentially the matrix exponential times a vector. To demonstrate this, consider the special case that arises in our application, namely, $\boldsymbol{b}(t) = \boldsymbol{b}_1 - e^{-rt}\boldsymbol{b}_2$. Here $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ are constant vectors and r is a constant scalar. In this case the solution to (1.1) can be expressed as

$$\boldsymbol{u}(t) = \exp(At)\,\boldsymbol{c} - A^{-1}\boldsymbol{b}_1 + e^{-rt}(rI + A)^{-1}\boldsymbol{b}_2, \tag{1.8}$$

where

$$\boldsymbol{c} = \boldsymbol{u}_0 + A^{-1}\boldsymbol{b}_1 - (rI + A)^{-1}\boldsymbol{b}_2. \tag{1.9}$$

The term $\exp(At)\,\boldsymbol{c}$ can be approximated by (1.6), with the right-hand side of (1.7) replaced by $\boldsymbol{c}$. Since this right-hand side is independent of z, the Krylov iteration mentioned above can be applied directly. Another positive aspect of this reformulation is that the placement of the contour is not restricted by the location of the singularities of $\widehat{\boldsymbol{b}}(z)$, but only by the eigenvalues of A. The price one pays for this is the additional computation of $A^{-1}\boldsymbol{b}_1$ and $(rI + A)^{-1}\boldsymbol{b}_2$. Of course, if either of the matrices A or $rI + A$ is singular or poorly conditioned, this approach is not viable.

A good choice of contour, Γ, and a suitable parametrization is key to the success of the contour integral method. Talbot suggested a cotangent contour, but simpler contours such as parabolas and hyperbolas have since been proposed and analyzed [5, 6, 7, 8, 14, 15, 23]. Here we extend the work of [32], and particularly [31], by considering parabolas of type

$$\Gamma: \quad z = \alpha + \mu\big(i\phi + 1\big)^2, \qquad -\infty < \phi < \infty, \tag{1.10}$$

with α and μ real parameters, and $\mu > 0$.

A parabolic contour is the natural curve when solving systems (1.1) with matrices A that satisfy a resolvent condition with respect to a region bounded by a *critical* parabola

$$\Pi: \quad x = a - by^2, \qquad x \equiv \operatorname{Re} z, \quad y \equiv \operatorname{Im} z, \tag{1.11}$$

with a and b real, and $b > 0$. In section 3 we derive such a resolvent condition for the Black–Scholes equation. This information is essential for estimating the optimal

parameters α and μ in the contour (1.10), as well as the step-size h in the quadrature rule (1.6). In the case of the Heston PDE, no such resolvent condition is known at present. We offer suggestions, however, for approximating it so that the parameters α, μ, and h are still reasonably close to optimal.

The outline of the paper is as follows. In section 2 we discuss the selection of the parameters α and μ and the step-size h. Applications to the PDEs of finance are presented in section 3 (Black–Scholes) and section 4 (Heston). In the latter case we compare the contour integral method to state-of-the-art alternating direction implicit (ADI) methods with regard to accuracy and efficiency. In section 5 we compare the method proposed here with a related contour integral method discussed in [13]. Finally, in section 6 we present conclusions and discuss various issues for future research.

2. Parameter selection. The accuracy of the contour integral method critically depends on good choices for α and μ in (1.10), as well as h in (1.6). In this section we show that α is determined by the critical parabola (1.11), while the optimal values of μ and h depend on (1.11) as well as the values of N and t.

Parameter tuning like this was done in [31, 32]. We shall follow a similar approach here, except we consider an arbitrary critical parabola (1.11). By contrast, in [31] and [32] the cases $a = 0$, $b = 1$ and $a = 0$, $b \to \infty$ were considered, respectively. Generalizing this work, we therefore consider the conformal map

$$z = \alpha + \mu\big(i\,w + 1\big)^2, \qquad w = \phi + i\,\psi, \tag{2.1}$$

with ϕ and ψ real. For fixed ψ and $-\infty < \phi < \infty$, this defines a parabola in the z-plane, symmetric with respect to the real axis. A few examples are shown in the left diagram of Figure 2.1. When $\psi = 0$ the parabola coincides with the contour of integration Γ; cf. (1.10).

2.1. Choice of α. As a preliminary step for the parameter tuning we need to map (2.1) to the critical parabola (1.11). Writing $z = x + iy$, one gets from (2.1)

$$x = \alpha + \mu\big((1-\psi)^2 - \phi^2\big), \qquad y = 2\mu\phi(1-\psi).$$

Solving for $(1-\psi)^2$ from $x = a - by^2$ gives

$$(1-\psi)^2 = \frac{\mu\,\phi^2 + a - \alpha}{\mu\big(1 + 4\,b\mu\,\phi^2\big)}. \tag{2.2}$$

The right-hand side is independent of ϕ, provided

$$\mu\,\phi^2 + a - \alpha = \gamma\big(1 + 4\,b\mu\phi^2\big)$$

for some positive constant γ. Comparing coefficients of ϕ^2 yields $\gamma = 1/(4b)$, and comparing the remaining terms yields

$$\alpha = a - \frac{1}{4b}. \tag{2.3}$$

Taking the square root in (2.2) and selecting the minus sign for the correct orientation gives

$$\psi = 1 - \frac{1}{2\sqrt{\mu b}}. \tag{2.4}$$

In summary, if α is given by (2.3) and ψ takes the constant value (2.4), then (2.1) maps to the parabola (1.11). The discussion of the next subsection should make it clear why this is the appropriate choice of α.

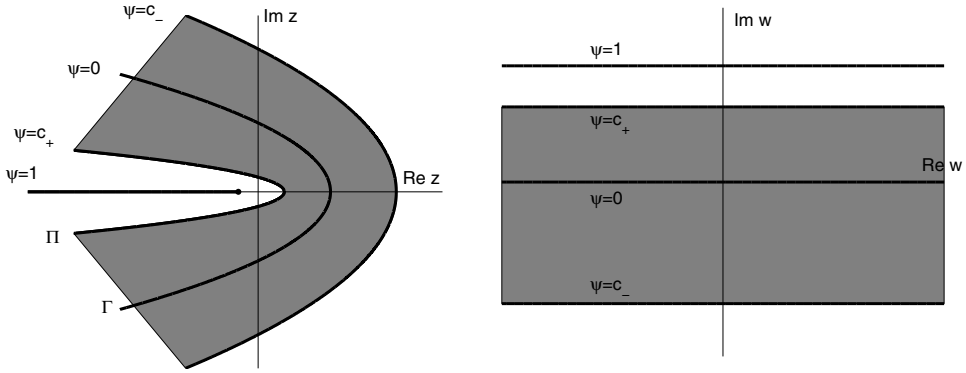

FIG. 2.1. *Schematic representation of the conformal map* (2.1). *In the left figure* Γ *refers to the parabola* (1.10), Π *refers to* (1.11), *and* c_+ *and* c_- *are the values defined by* (2.8) *and* (2.11) *respectively. The width of the strip on the right determines the size of the discretization error in the trapezoidal/midpoint rule; cf.* (2.6).

2.2. Discretization and truncation errors. The optimal values of μ and h are determined by analyticity properties of the integrand of (1.3). For this we need error estimates for the trapezoidal or midpoint rules when integrating analytic functions on the real line, a topic that has been investigated in [16, 17, 26] and elsewhere. Here we cite a theorem of Martensen [16], originally stated for the trapezoidal rule but trivially modified for the midpoint rule, namely,

$$\int_{-\infty}^{\infty} f(\phi)\,d\phi = h\sum_{k=-\infty}^{\infty} f(\phi_k) + DE(h),$$

where the *discretization error*, $DE(h)$, is bounded by

$$|DE(h)| \le \frac{2M(c)}{e^{2\pi c/h}-1}, \qquad M(c) \equiv \int_{-\infty}^{\infty} |f(\phi+ic)|\,d\phi. \tag{2.5}$$

It is assumed in [16] that $f(\phi)$ is real-valued, and its analytic continuation $f(\phi+i\psi)$ is absolutely integrable on $\psi = c$ and analytic in the strip $\psi \in (-c, c)$. Thus, the wider this strip of analyticity, the quicker the convergence.

It is relatively straightforward to extend this estimate to the case where $f(\phi)$ is complex vector-valued. This is the situation for the integral (1.3), where the integrand has different analytic properties in the upper and lower half-planes. The error bound is modified to $DE(h) = DE_+(h) + DE_-(h)$, where

$$|DE_+(h)| \le \frac{M(c_+)}{e^{2\pi c_+/h}-1}, \qquad |DE_-(h)| \le \frac{M(c_-)}{e^{-2\pi c_-/h}-1}, \tag{2.6}$$

and analyticity of the integrand is assumed in $\psi \in (c_-, c_+)$, with $c_- < 0$, $c_+ > 0$.

To illustrate how these error estimates are applied to the approximation (1.3)–(1.4), we refer to Figure 2.1, which shows a schematic representation of the conformal map (2.1). The lower half-plane $\psi < 0$ maps to the exterior of the parabola (1.10), and the strip $0 < \psi < 1$ maps to its interior. When $\psi \to 1$ the parabola degenerates into the half-line $x \le \alpha$, $y = 0$.

Estimate of DE_+*:* We apply the definition of $M(c)$ in (2.5) to (1.3) to obtain

$$M(c_+) \le \frac{1}{2\pi}\max_{-\infty<\phi<\infty}\left\{\|(z(w)I-A)^{-1}\|\|\boldsymbol{a}(z(w))\|\right\}\int_{-\infty}^{\infty}\left|e^{z(w)t}z'(w)\right|d\phi, \tag{2.7}$$

where $w = \phi + ic_+$. Here $\boldsymbol{a}$ is either the right-hand side of (1.7) or the vector $\boldsymbol{c}$ in (1.9).

To make the denominator in the error bound (2.6) large, we would like to maximize c_+. We cannot let $c_+ \to 1$, however, as the term $M(c_+)$ may become large as we do so. To see this, consider the term in braces in (2.7), and in particular the resolvent norm, $\|(z(w)I - A)^{-1}\|$. As c_+ is increased, the parabola $z(\phi + ic_+)$ will intersect the spectrum of A and the resolvent norm will blow up. If A is a normal matrix, this will determine the maximum possible value of c_+. If A is nonnormal, however, the resolvent norm may already be huge for much smaller values of c_+.

This situation was discussed in [20] and [29, sect. 12]. It was shown that for a model advection-diffusion problem there exists a critical parabola, like Π in (1.11), inside of which the resolvent norm can be large, but outside of which it is relatively small. We therefore need to restrict the value of c_+ so that the parabola $z(\phi + ic_+)$ does not enter the inner region. In order to maximize c_+, we therefore map $z(\phi + ic_+)$ to Π.

Characterizing the critical parabola Π was done in [20, 29] in terms of pseudospectra and the numerical range. For the matrix A, the *numerical range* (or *field of values*) is defined by

$$W(A) = \big\{\boldsymbol{u}^* A\boldsymbol{u} \mid \boldsymbol{u} \in \mathbb{C}^m,\ \boldsymbol{u}^*\boldsymbol{u} = 1\big\}.$$

The following bound, taken from [29, sect. 17] and [30], confirms that the resolvent norm may be large if $z \to W(A)$:

$$\|(zI - A)^{-1}\|_2 \le \frac{1}{\operatorname{dist}(z, W(A))}, \qquad z \notin W(A).$$

In the applications of the next two sections we shall estimate $W(A)$ and hence identify an appropriate Π that encloses it.

In order to map $z(\phi + ic_+)$ to Π we use the analysis of subsection 2.1 and deduce from (2.4) that the optimal choice of c_+ is

$$c_+ = 1 - \frac{1}{2\sqrt{\mu b}}. \tag{2.8}$$

The corresponding value of α is given by (2.3).

With this choice of c_+ the denominator in (2.6) becomes, as $h \to 0$,

$$e^{2\pi c_+/h} - 1 = O\Big(e^{\frac{2\pi}{h}\left(1 - \frac{1}{2\sqrt{\mu b}}\right)}\Big). \tag{2.9}$$

The integral in (2.7) can be estimated as

$$\int_{-\infty}^{\infty} \big|e^{z(w)t} z'(w)\big|\, d\phi = O\big(e^{at}\big), \tag{2.10}$$

since Π intersects the real axis at $z = a$. We combine the exponential contributions in (2.9) and (2.10) and ignore the algebraic terms in (2.7). Then (2.6) yields the following estimate for the discretization error:

$$DE_+ = O\big(e^{d_+}\big), \qquad d_+ \equiv a\,t - \frac{2\pi}{h}\Big(1 - \frac{1}{2\sqrt{\mu b}}\Big).$$

We have assumed in the above analysis that the singularities of $\widehat{\boldsymbol{b}}(z)$ are contained in the interior of Π, so that they place no additional restrictions on the width of the strip. For the other cases, we suggest a simple modification at the end of subsection 2.3. Note that when the implementation (1.8) is employed this is not an issue.

Estimate of DE_-: Next, consider $\psi < 0$. Since the integrand of (1.3) is analytic in this half-plane, one can in principle let $\psi \to -\infty$. The resolvent norm is in this case not a limiting factor, as it is assumed relatively small in the region outside Π. In this case, however, the estimate analogous to (2.10) is

$$\int_{-\infty}^{\infty} \left| e^{z(w)t} z'(w) \right| d\phi = O\big(e^{\alpha t + \mu t (1-c_-)^2}\big),$$

the right-hand side of which grows unboundedly as $c_- \to -\infty$. The optimal value of c_- can be obtained by considering also the factor $e^{-2\pi c_-/h} - 1$ in the denominator of the right-hand side of (2.6), as was shown in [31]. Combining these two exponential contributions yields

$$\frac{e^{\alpha t + \mu t (1-c_-)^2}}{e^{-2\pi c_-/h} - 1} \sim e^{\,g(c_-)}, \qquad g(c_-) = \alpha t + \mu t (1-c_-)^2 + \frac{2\pi c_-}{h}.$$

The function on the right has a unique minimum at

$$c_- = 1 - \frac{\pi}{\mu t h}. \tag{2.11}$$

By computing the corresponding value of $g(c_-)$, the discretization error, DE_-, can be estimated as

$$DE_- = O\big(e^{d_-}\big), \qquad d_- \equiv \alpha\, t - \frac{\pi^2}{\mu t h^2} + \frac{2\pi}{h}.$$

Estimate of TE: An additional error, the *truncation error*, is committed in the truncation of (1.4) to (1.6). This is estimated as simply the magnitude of the first omitted term, which is approximately

$$TE = O\big(e^{d_T}\big), \qquad d_T \equiv \alpha\, t + \mu t (1 - (hN)^2).$$

2.3. Optimal strategy. For the three error components to be asymptotically equal, one requires

$$d_+ = d_- = d_T,$$

which represent two equations for the two parameters (μ, h). We sketch the solution procedure briefly.

By setting $d_- = d_T$ and solving the resulting quadratic equation for μt, one obtains

$$\mu t = \frac{\pi}{h\,(1 \pm hN)}. \tag{2.12}$$

The plus sign ensures that c_- in (2.11) is negative, and hence the formula for d_T simplifies to

$$d_T = \alpha\, t + \frac{\pi}{h} - \pi N.$$

Setting d_+ equal to this quantity yields, after simplification,

$$\frac{\pi}{\sqrt{\mu b}} = (\alpha - a)th + \pi(3 - hN). \tag{2.13}$$

Squaring both sides and making use of (2.3) and (2.12) yields a quadratic equation for h, namely,

$$\big(4\,\pi\,Nb - t\big)^2 h^2 - 8\,b\pi\,\big(5\,t + 12\,\pi\,Nb\big)h + 144\,b^2\pi^2 = 0. \tag{2.14}$$

Elementary analysis shows that for all positive t, N, and b, this equation has two positive real roots, h_1 and h_2, with $0 < h_1 < h_2$. The exception occurs when the coefficient of the quadratic term vanishes, in which case there is a single positive root, h_1. In squaring (2.13) a spurious root was introduced, and further analysis shows that it is h_2 that should be discarded. We summarize as follows.

PROPOSITION 2.1 (optimal parameter selection). *Given values t, a, b, and N, solve* (2.14). *The smaller root, $h = h_1$, defines the step-size in the quadrature method* (1.6). *Substitute this value of h into* (2.12) *and take the plus sign to obtain the parameter μ. Together with the value α given by* (2.3), *this defines the contour of integration Γ in* (1.10).

It is instructive to compare (2.14) to the results of [31, 32]. First, as $b \to \infty$ (i.e., eigenvalues on the real axis) it is readily checked that one obtains the formulas for h and μ from [32]. Second, if $b = 1$, we recover the results of [31].

It is also instructive to look at asymptotic values of these parameters. We have, for the optimal h and μ,

$$h = \frac{3}{N} + O(N^{-3/2}), \qquad \mu t = \frac{\pi\,N}{12} + O(N^{1/2}),$$

and the common value of $d_+ = d_- = d_T$ satisfies

$$d = -\frac{2}{3}\pi N + O(N^{1/2}). \tag{2.15}$$

To leading order this indicates a convergence rate of the form $O\big(\exp(-\frac{2}{3}\,\pi N)\big)$, also derived in [32]. We remark that this estimate of the convergence rate is independent of the size of the matrix A.

The implied constant in the estimate (2.15) depends on $\sqrt{t/b}$. One can expect the accuracy to deteriorate as $t \to \infty$, and similarly as $b \to 0$. On the other hand, the situation $b \to \infty$ should yield relatively higher accuracy.

Note that a numerical verification of (2.15) is not entirely straightforward, since this estimate is valid in the limit $N \to \infty$. As discussed in the next section, it is not possible to take N larger than 15 or so before roundoff error becomes a factor in a practical computation.

We conclude by recalling that in subsection 2.2 we have assumed that the singularities of $\widehat{\boldsymbol{b}}(z)$ are contained in the interior of Π. In the applications below it can happen that singularities of $\widehat{\boldsymbol{b}}(z)$ are located outside Π, but on the negative real axis. In such cases we shall simply take $a = 0$ when determining the contour of integration (1.10). That is, we translate the critical parabola Π so that it includes the singularities of $\widehat{\boldsymbol{b}}(z)$. This is justified by the fact that the optimal parameters as determined by (2.12) and (2.14) and the optimal convergence rate given by (2.15) depend only on b, not on a.

2.4. Summary of two implementations. The above discussion leads us to the following two recommendations for implementing the contour integral method. The first is useful for parallel implementation, and the second is intended for serial implementation by Krylov iteration. For both implementations we assume that the numerical range of the matrix A is contained in a parabola (1.11), and that the values of a and b can be estimated. The user supplies values of N and t in addition to a and b.

Implementation I:

1. Compute $\widehat{\boldsymbol{b}}(z)$ and its singularities. If they are not in the interior of (1.11), increase the value of a until they are.
2. Compute α, μ, and h as described in Proposition 2.1.
3. Compute the nodes z_k and the weights z'_k for $k = 0, 1, \ldots, N-1$ according to (1.5) and (1.10).
4. Solve the shifted systems (1.7) for the transforms $\widehat{\boldsymbol{u}}_k$ by some direct method. This can be done in parallel, as described in the appendix.
5. Compute the vector sum (1.6) to obtain an approximation to $\boldsymbol{u}(t)$.

In the second implementation the special situation $\boldsymbol{b}(t) = \boldsymbol{b}_1 - e^{-rt}\boldsymbol{b}_2$ is assumed. In addition, it is assumed that neither A nor $rI + A$ are ill-conditioned matrices.

Implementation II:

1. Compute $A^{-1}\boldsymbol{b}_1$, $(rI + A)^{-1}\boldsymbol{b}_2$ and $\boldsymbol{c}$ as defined in (1.9).
2. As in step 2 of Implementation I.
3. As in step 3 of Implementation I.
4. Solve the shifted systems (1.7), with right-hand side equal to $\boldsymbol{c}$. The Krylov iteration of the appendix is used for this.
5. Use the vector sum (1.6) to compute an approximation to the term $\exp(At)\boldsymbol{c}$ in (1.8) and hence an approximation to $\boldsymbol{u}(t)$.

In step 3 of the first implementation, the user has the freedom to use either the trapezoidal or midpoint nodes ϕ_k. In the second implementation, the use of the trapezoidal rule is recommended; see the appendix.

3. The Black–Scholes PDE. Our first application is the Black–Scholes PDE,

$$\frac{\partial u}{\partial t}(s,t) = \frac{1}{2}\sigma^2 s^2 \frac{\partial^2 u}{\partial s^2}(s,t) + rs\frac{\partial u}{\partial s}(s,t) - ru(s,t) \quad (s > L,\, 0 < t \le T). \tag{3.1}$$

Here $r \ge 0$ and $\sigma > 0$ denote given real constants that represent the interest rate and the volatility, respectively. $L \ge 0$ represents a given lower barrier, and $T > 0$ is the given maturity time of the option.

The Black–Scholes PDE constitutes the seminal model in financial option pricing theory; cf. [2, 24, 28, 33], for example. It is a time-dependent advection-diffusion-reaction equation, with spatial variable s. Initial and boundary conditions for (3.1) are determined by the specific option under consideration. The solution value $u(s,t)$ then represents the fair price of this option when the underlying asset price equals s at time $T - t$.

We note that (semi)analytical pricing formulas have been established for most options in the Black–Scholes framework. Nevertheless, numerical solution of equation (3.1) remains a worthwhile exercise, since it forms the starting point for many (non-trivial) generalizations in contemporary financial mathematics. We consider one such generalization in section 4.

To render the numerical solution of (3.1) feasible, it is common to restrict the spatial domain to a bounded set $[L, S]$ with fixed S taken sufficiently large. We shall assume boundary conditions of Dirichlet type.

The Black–Scholes PDE can be written in terms of the elliptic operator $\mathcal{A}$ defined by

$$(\mathcal{A}u)(s) = \frac{1}{2}\sigma^2 s^2 u''(s) + rsu'(s) - ru(s) \quad (L \le s \le S) \tag{3.2}$$

for $u \in D(\mathcal{A}) = \{u \mid u : [L,S] \to \mathbb{C}$ is twice continuously differentiable and $u(L) = u(S) = 0\}$. Note that it is not necessary to consider the weak form here. Let $\langle \cdot, \cdot \rangle$ be the standard inner product on $D(\mathcal{A})$. To determine suitable contours of integration, we study the numerical range of $\mathcal{A}$, defined by

$$W(\mathcal{A}) = \{\langle \mathcal{A}u, u\rangle \mid u \in D(\mathcal{A}) \text{ and } \langle u, u\rangle = 1\}.$$

The following theorem shows that the numerical range of $\mathcal{A}$ is bounded by a parabola Π of the form (1.11).

THEOREM 3.1. *Let the operator $\mathcal{A}$ be given by* (3.2). *Then*

$$W(\mathcal{A}) \subset \left\{ z \mid z = x + iy\ ,\ x \le a - by^2, \quad \textit{with}\ \ a = \frac{3}{8}\sigma^2 - \frac{3}{2}r\ ,\ b = \frac{1}{2}\sigma^2/(r-\sigma^2)^2 \right\}$$

whenever $r \neq \sigma^2$. Further, if $r = \sigma^2$, then

$$W(\mathcal{A}) \subset \left\{ z \mid z = x + iy\,,\, x \le -\frac{9}{8}r\,,\, y = 0 \right\}.$$

Proof. Let $u \in D(\mathcal{A})$ with $\langle u, u\rangle = 1$. Integration by parts yields

$$\int_L^S s^2 u''(s)\bar{u}(s)\,ds = -\int_L^S s^2 |u'(s)|^2\,ds - 2\int_L^S su'(s)\bar{u}(s)\,ds.$$

Using this, we get

$$\langle \mathcal{A}u, u\rangle = -\frac{1}{2}\sigma^2 P + (r-\sigma^2)Q - r,$$

where

$$P = \int_L^S s^2 |u'(s)|^2\,ds, \quad Q = \int_L^S su'(s)\bar{u}(s)\,ds.$$

A second integration by parts implies that $Q = -\bar{Q} - 1$ and thus $\operatorname{Re} Q = -\frac{1}{2}$. Next, by the Cauchy–Schwarz inequality there holds $|Q| \le \sqrt{P}$ and, consequently, $(\operatorname{Im} Q)^2 \le P - \frac{1}{4}$. Write $x = \operatorname{Re}\langle \mathcal{A}u, u\rangle$ and $y = \operatorname{Im}\langle \mathcal{A}u, u\rangle$. Then we obtain

$$x = -\frac{1}{2}\sigma^2 P + \frac{1}{2}\sigma^2 - \frac{3}{2}r \ \text{ and } \ y^2 \le (r-\sigma^2)^2\left(P - \frac{1}{4}\right).$$

This readily leads to the result of the theorem. ∎

Further theoretical analysis complemented by numerical experiments suggest that the bound of Theorem 3.1 on $W(\mathcal{A})$ is sharp. Note, however, that in the discussion of section 2 we have considered the numerical range of the matrix A, not the operator $\mathcal{A}$. The latter is typically easier to estimate, with Theorem 3.1 being a good example. Here we shall make the assumption that for all practical semidiscretizations the numerical range of the matrix A forms a reasonable approximation to the numerical

range of the operator $\mathcal{A}$, at least in the important region near the intersection of the boundary of $W(\mathcal{A})$ and the real axis. In view of the parabolic bound of Theorem 3.1, the contour integral method with parabolic contour from sections 1 and 2 is thus naturally suited for the Black–Scholes PDE.

We consider a standard semidiscretization of the Black–Scholes PDE, defined by the usual central second-order finite difference (FD) schemes for the advection and diffusion terms. For ease of formulation, we use a uniform spatial grid $s_j = L + j \cdot \Delta s$ $(j = 1, 2, \ldots, m)$, $\Delta s = (S-L)/(m+1)$. Then a semidiscrete system (1.1) is obtained with $m \times m$ tridiagonal matrix A given by

$$A = \text{tridiag}\left(\frac{\sigma^2 s_j^2}{2(\Delta s)^2} - \frac{r s_j}{2\Delta s}\ ,\ -\frac{\sigma^2 s_j^2}{(\Delta s)^2} - r\ ,\ \frac{\sigma^2 s_j^2}{2(\Delta s)^2} + \frac{r s_j}{2\Delta s} \right). \tag{3.3}$$

As a prototype example we take a European call option. For many other more exotic options only the initial and boundary conditions change, and our method can be applied directly. We provide an example of this in section 4. If $K \in (0, S)$ denotes the given strike price, then the price of a European call option corresponds to $L = 0$ and the initial condition

$$u(s, 0) = \max(0,\ s - K) \qquad (0 \le s \le S)$$

and the Dirichlet boundary conditions

$$\begin{aligned} u(0, t) &= 0 && (0 \le t \le T), \\ u(S, t) &= S - e^{-rt} K && (0 \le t \le T). \end{aligned}$$

The initial vector $\boldsymbol{u}_0$ in (1.1) is obtained by direct evaluation of the above initial condition at the spatial grid points $s_1, s_2, \ldots, s_m$. For $0 \le t \le T$, the vector $\boldsymbol{b}(t)$ is given by

$$\boldsymbol{b}(t) = \boldsymbol{b}_1 - e^{-rt}\boldsymbol{b}_2, \qquad \boldsymbol{b}_1 = (0, 0, \ldots, 0, \gamma S)^{\mathrm{T}}, \qquad \boldsymbol{b}_2 = (0, 0, \ldots, 0, \gamma K)^{\mathrm{T}},$$

where

$$\gamma = \frac{\sigma^2 s_m^2}{2(\Delta s)^2} + \frac{r s_m}{2\Delta s}\,.$$

As a first numerical illustration we apply the method to the semidiscrete Black–Scholes PDE formulated above with $m = 200$ grid points and $N = 1, 2, 3, \ldots, 30$. We use Implementation I with the midpoint nodes; see subsection 2.4. The Laplace transform of $\boldsymbol{b}(t)$ is

$$\widehat{\boldsymbol{b}}(z) = \frac{1}{z}\boldsymbol{b}_1 - \frac{1}{z + r}\boldsymbol{b}_2\,,$$

which has singularities at $z = 0$ and $z = -r$. The singularity $z = 0$ is outside the numerical range $W(\mathcal{A})$ whenever $a = \frac{3}{8}\sigma^2 - \frac{3}{2}r < 0$. When this happens, $a = 0$ is used in our implementation, as explained in the last paragraph of subsection 2.3. For the (direct) solution of the linear systems (1.7), we employ here MATLAB's backslash operator; see the appendix.

We arbitrarily fix $r = 0.06$, $t = T = 1$, $K = 80$, $S = 200$ and consider two values for the volatility: $\sigma = 0.20$ and $\sigma = 0.05$. Figure 3.1 shows the numerical results,

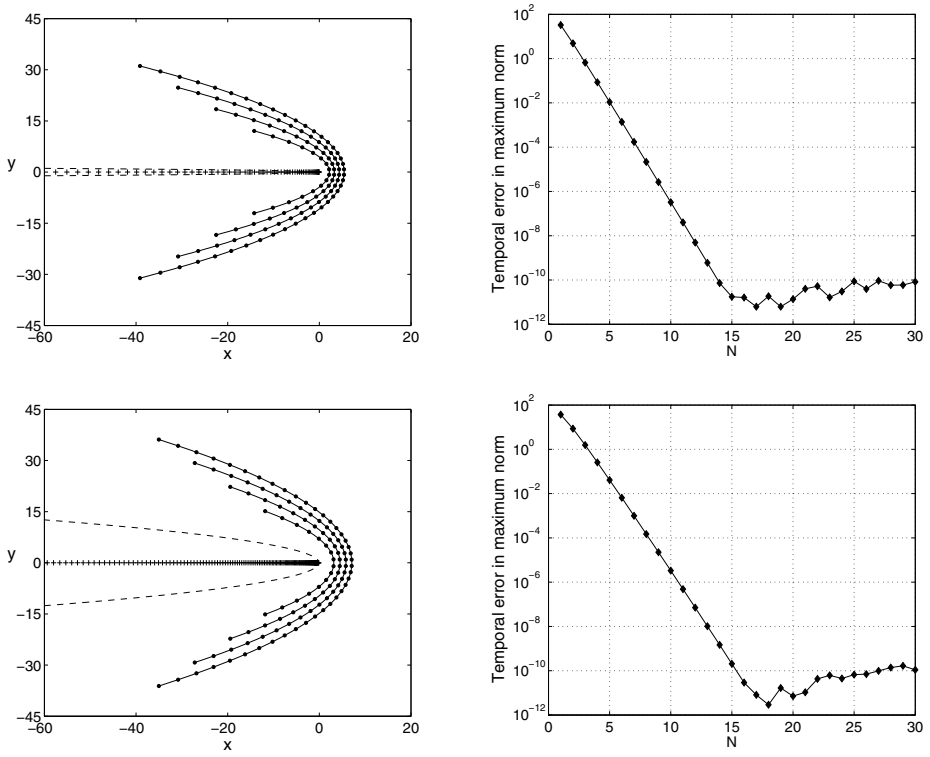

FIG. 3.1. *Contour integral method (Implementation* I, *midpoint rule) for the Black–Scholes PDE with* $r = 0.06$, $t = T = 1$, $K = 80$, $L = 0$, $S = 200$. *Number of spatial grid points:* $m = 200$. *Top row:* $\sigma = 0.20$. *Bottom row:* $\sigma = 0.05$. *Left column: eigenvalues of the semidiscrete Black–Scholes matrix* A *(pluses), analytical bound for the numerical range of the operator* $\mathcal{A}$ *(dashed line), and contours for* $N = 8, 12, 16, 20$ *(solid lines) with points* z_k *(bullets). Right column: temporal error in the maximum norm vs.* $N = 1, 2, 3, \ldots, 30$.

where the top row represents $\sigma = 0.20$ and the bottom row $\sigma = 0.05$. The left column displays the contours (solid lines) for the sample values $N = 8, 12, 16, 20$ together with the actual points z_k (bullets) used. Here, as N increases, the contours move to the right. Subsequently, the parabola from Theorem 3.1 that bounds $W(\mathcal{A})$ is included (dashed line). Finally, the eigenvalues of the semidiscrete matrix A that lie in the displayed domain are shown (pluses).

Using the software package EigTool [35], we estimated also the numerical range $W(A)$ of the actual semidiscrete matrix A, and this was found in both cases to agree well with the parabolic region given by Theorem 3.1. (To avoid clutter in Figure 3.1, we have not displayed this result.) For $\sigma = 0.05$ the parabola is much wider than for $\sigma = 0.20$, which is also obvious by considering the values of b from Theorem 3.1: these are $b \approx 0.38$ and $b = 50$, respectively. This indicates that if $\sigma = 0.05$ the operator $\mathcal{A}$ (or matrix A) deviates much more from normality than if $\sigma = 0.20$.

The right column of Figure 3.1 shows the *temporal error*, measured in the maximum norm, which is defined as the difference between the exact solution value $\boldsymbol{u}(T)$ to the semidiscrete system and its approximation obtained with our contour integral method, i.e., the error in (1.6). Note that this error does not contain the error due to semidiscretization. We are interested here in studying the error in (1.6) by itself,

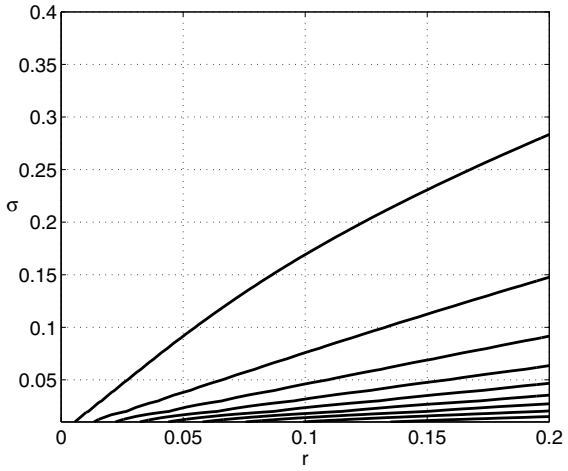

FIG. 3.2. *Temporal errors of the contour integral method (Implementation* I*, midpoint rule) when* $N = 12$ *for the Black–Scholes PDE with* $t = T = 1$*,* $K = 80$*,* $L = 0$*,* $S = 200$*. Error levels are* $10^{-8}, 10^{-7}, \ldots, 10^{0}$ *counting from top to bottom. Number of spatial grid points:* $m = 200$*.*

to validate our theory of section 2. In sections 4 and 5 we shall also incorporate the spatial error.

We computed a reference value for $\boldsymbol{u}(T)$ by formula (1.8) using the `expm` command in MATLAB to obtain the matrix $\exp(At)$ for $t = T$. We remark that this command is in general far more expensive than the contour integral approach; see the appendix.

From Figure 3.1, it is clear that the temporal errors decay rapidly when $N \leq 15$ and the convergence rate is geometric: the errors behave like $O\big(\exp(-\omega N)\big)$ with (near) constant $\omega > 0$. We compute, by least-squares approximation, that $\omega \approx 2.06$ (for $\sigma = 0.20$) and $\omega \approx 1.87$ (for $\sigma = 0.05$). The theoretical value $\omega = -(d_+/N)$ satisfies $\omega \approx 2.08$ (for $\sigma = 0.20$) and $\omega \approx 1.88$ (for $\sigma = 0.05$), and hence these are in good agreement. Subsequent experiments show that the temporal errors are essentially independent of the number of spatial grid points m as mentioned below (2.15).

Note that beyond $N = 16$ the temporal errors start to increase. This is caused by roundoff errors and the inherent ill-conditioning of the Laplace inversion formula. For financial applications this is not a critical issue, however, as this phenomenon occurs only in the region of very high accuracies which are in general not required in finance. We mention that roundoff error control could be incorporated by employing similar strategies to those in [31].

To gain further insight into the accuracy of our contour integral method, we have applied it with $N = 12$ in the case of a 40×40 grid of values $(r, \sigma) \in [0, 0.2] \times [0.01, 0.4]$. Figure 3.2 shows a contour plot of the temporal errors. We observe that the errors are always less than 10^{-4} under the mild condition that $\sigma \geq 0.05$, $r \leq 0.2$. This accuracy is sufficient for most applications in finance. Note that if σ tends to zero, for fixed $r > 0$, then the error increases. This is also expected in view of the theory of section 2 and the fact that $b = \frac{1}{2}\sigma^2/(r - \sigma^2)^2$ approaches zero.

4. The Heston PDE. As a second application we consider the Heston PDE [9]. This equation forms a prominent, two-dimensional extension of the Black–Scholes PDE. In addition to standard European call options, we also consider in this section European down-and-out barrier options. They represent a popular type of ex-

otic, path-dependent option. Closed-form analytical pricing formulas for these exotic options are not known in the literature for general parameter choices in the Heston model.

Let $u(s, v, t)$ denote the fair price of a European-style option if at time $T - t$ the underlying asset price equals s and its variance equals v. Heston's stochastic volatility model for the asset price yields that u satisfies the PDE

$$\frac{\partial u}{\partial t} = \frac{1}{2}s^2 v\frac{\partial^2 u}{\partial s^2} + \rho\sigma s v\frac{\partial^2 u}{\partial s \partial v} + \frac{1}{2}\sigma^2 v\frac{\partial^2 u}{\partial v^2} + rs\frac{\partial u}{\partial s} + \kappa(\eta - v)\frac{\partial u}{\partial v} - ru \tag{4.1}$$

for $0 < t \leq T$, $s > L$, $v > 0$; see [9, 24]. The parameter $\kappa > 0$ is the mean-reversion rate, $\eta > 0$ is the long-term mean, $\sigma > 0$ is the volatility-of-variance, $\rho \in [-1, 1]$ is the correlation between the two underlying Brownian motions, and r is the interest rate. We assume here that $2\kappa\eta > \sigma^2$, which is known as the Feller condition. Further, we have assumed without loss of generality that the market price of volatility risk is equal to zero. For a standard European call option $L = 0$, whereas for European down-and-out call options $L > 0$.

To render the numerical solution feasible, the spatial domain is restricted to a bounded set $[L, S] \times [0, V]$ with fixed S, V chosen sufficiently large. The initial condition for (4.1) becomes

$$u(s, v, 0) = \max(0,\, s - K) \qquad (L \leq s \leq S\,,\, 0 \leq v \leq V). \tag{4.2}$$

Next, Dirichlet boundary conditions are given by [9, 34]

$$u(L, v, t) = 0 \qquad (0 \leq v \leq V\,,\, 0 \leq t \leq T), \tag{4.3a}$$

$$u(S, v, t) = S - e^{-rt}K \qquad (0 \leq v \leq V\,,\, 0 \leq t \leq T), \tag{4.3b}$$

$$u(s, V, t) = s - L \qquad (L \leq s < S\ ,\ 0 \leq t \leq T). \tag{4.3c}$$

Note that at the boundary $v = 0$ no condition is specified. From the assumption $2\kappa\eta > \sigma^2$ it follows that this is an outflow boundary.

For the Heston problem (4.1)–(4.3) we consider the second-order FD discretization described in detail in [11]. Here a Cartesian grid with nonuniform meshes in both the s- and v-directions is applied such that relatively many mesh points lie in the neighborhood of $s = K$ and $v = 0$, respectively. This is motivated by the observations that the initial function (4.2) possesses a discontinuity in its first derivative at $s = K$ and that for $v \approx 0$ the Heston PDE is advection-dominated. Further, it is natural to have many grid points near $(s, v) = (K, 0)$ as in practice this is the region in the (s, v)-domain where one wishes to obtain option prices.

On the grid under consideration each spatial derivative appearing in (4.1) is replaced by its corresponding second-order central FD scheme, except for the region $v > 1$ and at the boundary $v = 0$. In the region $v > 1$, a second-order upwind scheme for $\partial u/\partial v$ is applied whenever the flow in the v-direction is towards $v = V$. This is done so as to avoid spurious oscillations in the FD solution when σ is close to zero. At the outflow boundary $v = 0$, the derivative $\partial u/\partial v$ is approximated using a second-order upwind scheme as well. All other spatial derivative terms in the v-direction vanish at $v = 0$, due to the factor v occurring in (4.1), and hence these terms do not require further treatment.

The above FD discretization yields an initial value problem for a (large) system of ODEs (1.1). The vector $\boldsymbol{b}(t)$ depends on the boundary conditions (4.3) and is again

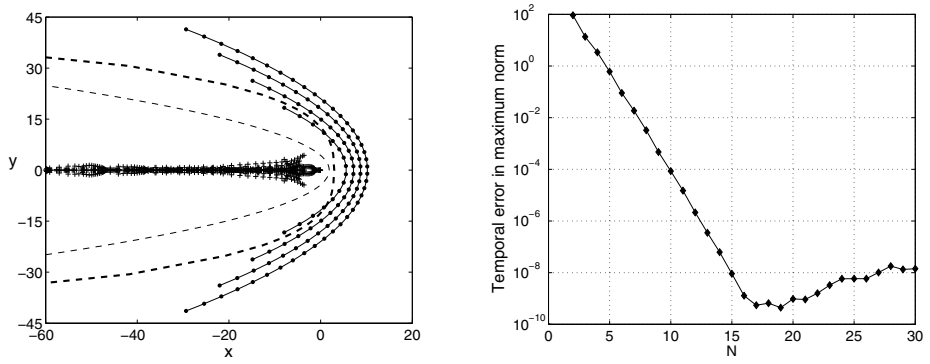

FIG. 4.1. *Contour integral method (Implementation* I*, midpoint rule) for the Heston PDE with* $\kappa = 1.5$*,* $\eta = 0.04$*,* $\sigma = 0.3$*,* $\rho = -0.9$*,* $r = 0.025$*,* $t = T = 1$*,* $K = 100$*,* $L = 0$*,* $S = 8K$*,* $V = 5$*. Number of spatial grid points:* $m = 50 \times 25$*. Left: eigenvalues of the semidiscrete Heston matrix* A *(pluses), analytical bound for the numerical range of the* $\frac{1}{2}s^2 V u_{ss}$ *term (dashed line), numerical range of* A *(bold dashed line), and contours for* $N = 8, 12, 16, 20$ *(solid lines) with points* z_k *(bullets). Right: temporal error vs.* $N = 1, 2, 3, \ldots, 30$*.*

of the type $\boldsymbol{b}_1 - e^{-rt}\boldsymbol{b}_2$ with fixed vectors $\boldsymbol{b}_1$, $\boldsymbol{b}_2$ as in section 3. As a consequence, the solution vector $\boldsymbol{u}(t)$ to the semidiscrete Heston PDE (1.1) is given by (1.8).

For the numerical illustrations throughout this section we consider the Heston parameter values from [1] given by

$$\kappa = 1.5, \quad \eta = 0.04, \quad \sigma = 0.3, \quad \rho = -0.9, \quad r = 0.025,$$

and we further take $T = 1$, $K = 100$, $S = 8K$ (if $L = 0$), $S = 14K$ (if $L > 0$), and $V = 5$ as in [11].

For the parameter selection in our contour integral method, an estimate for the numerical range of the PDE operator (or its semidiscrete version) is required. In the case of the Heston PDE such information is not available at present. By numerical experimentation we have found that a reasonable first indication in many (but not all) instances of the Heston PDE is given by the numerical range of the $\frac{1}{2}s^2 V u_{ss}$ term, which is provided by Theorem 3.1 upon setting $r = 0$ and $\sigma = \sqrt{V}$.

As an initial numerical example we consider application of the method of sections 1 and 2 to the semidiscrete Heston problem (4.1)–(4.3) outlined above for a European call option with 50×25 grid points (s, v). We employ Implementation I with the midpoint nodes and use formula (1.8), evaluated in MATLAB with the `expm` function, as a reference formula to compute temporal errors. Figure 4.1 displays results analogous to Figure 3.1. In the left subfigure we have also included the actual numerical range of A estimated with EigTool (bold dashed line); a natural scaled version of the Euclidean norm was used here that takes into account the nonuniformness of the grid. The actual numerical range is somewhat larger than the estimate we use. From the right subfigure it is clear that the convergence rate is again geometric; the temporal error behaves as $O\big(\exp(-\omega N)\big)$ with $\omega \approx 1.76$ (for $N \leq 15$). If a more accurate estimate for the numerical range of the Heston PDE operator becomes available, the parameter estimates of section 2 can be made closer to optimal and more accurate results can be expected.

As the first main numerical example we apply our contour integral method to the semidiscrete Heston problem for a European call option with 1000×500 grid

points (s, v). We consider total errors, measured in the maximum norm, versus cpu-time. The *total error* is defined as the difference between the vector of exact option prices $u(s_i, v_j, T)$ at the spatial grid points and the approximation vector obtained by the numerical method, restricted to a natural region of interest taken here as $\frac{1}{2}K < s < \frac{3}{2}K$ and $0 < v < 1$. Notice that in contrast to the temporal error, the total error also contains the error due to semidiscretization. To compute total errors, we use the closed-form analytical pricing formula of [9].

We consider here the following implementations of the contour integral method. The efficient solution of the linear systems (1.7) and other computational details are discussed in the appendix.

- *Contour–Parallel*: Implementation I executed in parallel on 8 processors. Midpoint nodes are used and the systems (1.7) are solved by MATLAB's backslash operator.
- *Contour–Krylov*: Implementation II executed on a single processor. Trapezoidal nodes are used and the systems (1.7) are solved by preconditioned Krylov iteration.
- *Contour*: Same as Contour–Parallel, except executed on a single processor.

In [11] the numerical solution of the semidiscrete Heston problem has been investigated by applying ADI time-stepping schemes. Here we compare our contour integral method to the *modified Craig–Sneyd (MCS) scheme*, with parameter $\theta = \frac{1}{3}$, which was found to be an effective method. For details and further references to this and other ADI-type schemes, we refer the reader to [11, 12]. Application of ADI schemes to the semidiscrete Heston problem leads to linear systems with matrices that are fixed over the successive time-steps and possess a small bandwidth. In view of this, these systems are efficiently solved using a priori LU factorization. We remark that designing an effective parallel version of ADI schemes is a nontrivial task, and hence we executed the MCS scheme only on a single processor.

Figure 4.2 displays total errors versus cpu-time for the various implementations. Contour–Parallel and Contour–Krylov have been applied for $2 \leq N \leq 20$ and Contour for $2 \leq N \leq 12$. The MCS scheme was applied with a range of constant step-sizes Δt satisfying $0.001 \leq \Delta t \leq 1$. We remark that damping, as discussed in [11], was not used. Note that the error curves level off at a value slightly greater than 10^{-4}. This is where the error due to the space discretization starts to dominate the temporal error.

In Figure 4.2 we observe that the implementations Contour–Parallel and Contour–Krylov are both much more efficient than the direct implementation Contour. Further, we find that Contour–Parallel is slightly more efficient than Contour–Krylov. We note that the "kink" in the Contour–Parallel curve is caused by the number of processors used; with more processors this curve will continue going straight down.

Figure 4.2 also reveals that Contour–Parallel and Contour–Krylov are both more efficient than MCS at smaller accuracies. The *break-even* point lies roughly in the range 10^{-4}–$10^{-3.5}$. Based on additional numerical evidence, this break-even point is expected to persist for larger numbers of spatial grid points. We mention that the other three Heston examples considered in [11] also yield a break-even point roughly in the range 10^{-4}–$10^{-3.5}$.

Our second main numerical example concerns a European down-and-out call option. For this exotic option a closed-form analytical pricing formula under the Heston model is not available in the literature whenever $\rho \neq 0$. We shall therefore study the efficiency of the various implementations by considering temporal errors rather than total errors. We take the same example as the preceding one, except now we select a

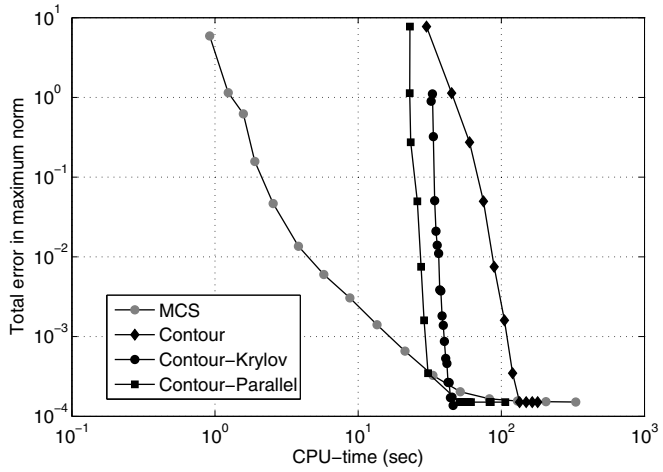

FIG. 4.2. *Total errors vs. cpu-time of the contour integral method and the MCS scheme with* $\theta = \frac{1}{3}$ *for the Heston PDE and standard European call option with* $\kappa = 1.5$, $\eta = 0.04$, $\sigma = 0.3$, $\rho = -0.9$, $r = 0.025$, $t = T = 1$, $K = 100$, $L = 0$, $S = 8K$, $V = 5$. *Number of spatial grid points:* $m = 1000 \times 500$. *Explanation of the implementations Contour, Contour–Krylov, and Contour–Parallel is given in the text.*

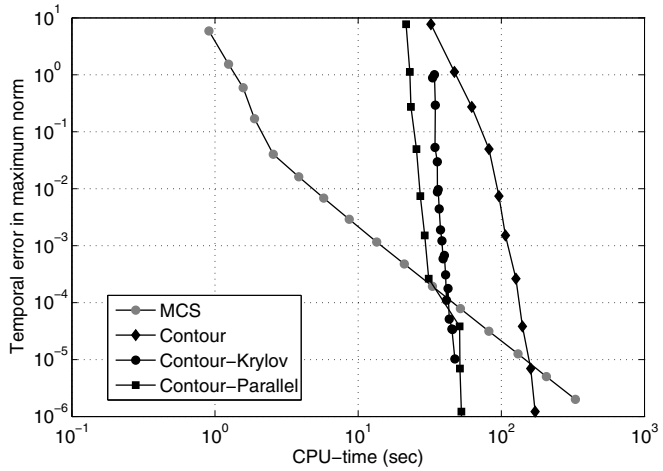

FIG. 4.3. *Temporal errors vs. cpu-time of the contour integral method and the MCS scheme with* $\theta = \frac{1}{3}$ *for the Heston PDE and European down-and-out call option with* $\kappa = 1.5$, $\eta = 0.04$, $\sigma = 0.3$, $\rho = -0.9$, $r = 0.025$, $t = T = 1$, $K = 100$, $L = 80$, $S = 14K$, $V = 5$. *Number of spatial grid points:* $m = 1000 \times 500$. *Explanation of the implementations Contour, Contour–Krylov, and Contour–Parallel is given in the text.*

barrier $L = 80$. To compute a reference value for $\boldsymbol{u}(T)$ we have used the implementation Contour–Parallel with $N = 18$. We note that temporal errors are computed here on the region of interest $\max\{\frac{1}{2}K, L\} < s < \frac{3}{2}K$ and $0 < v < 1$. Figure 4.3 displays the results. We conclude that the observations made with respect to Figure 4.2 for a standard European call option remain valid in the case of its exotic, down-and-out variant. In particular, we arrive at approximately the same break-even point, in the range 10^{-4}–$10^{-3.5}$, for Contour–Parallel and Contour–Krylov with MCS.

5. Comparisons with a related method. A contour integral method similar to (1.6) has been proposed for the (generalized) Black–Scholes equation by Lee and Sheen [13]. We briefly summarize the similarities and differences between these two methods.

The method of [13] is essentially identical to the method (1.6), as both methods are based on the ideas of [14, 23, 32] and others. The difference lies in the choice of contour and the parameter selection for this contour. In [13] a hyperbolic contour, originally proposed in [23], was used. The parameter selection for this contour was based on the formulas of [32].

We are not convinced, however, that the parameter selection strategy used in [13] is optimal. This strategy was based on the formulas of [32], but those formulas were derived for a hyperbolic contour with a different parameterization. It should be possible to use the ideas presented here and in [31, 32] to obtain the optimal parameters for that parameterization. This is not entirely straightforward, however, as the mapping used in [13] is more complicated than our (2.1).

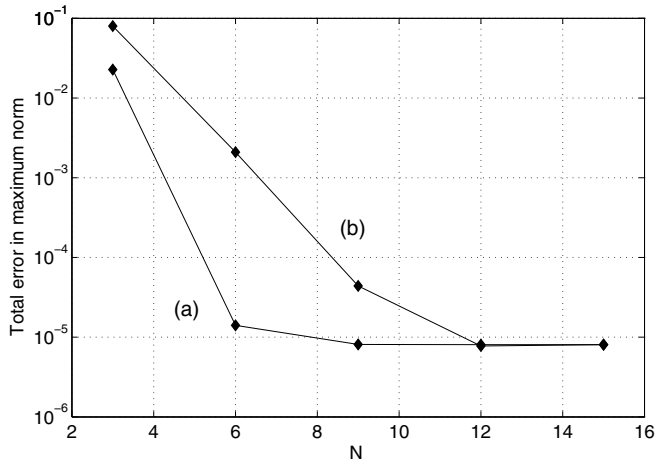

FIG. 5.1. *Total error in the contour integral method* (a) *proposed here (Implementation* I*, trapezoidal rule) and* (b) *proposed in* [13]*. The problem is the Black–Scholes PDE with* $r = 0.05$*,* $\sigma = 0.3$*,* $t = T = 1$*,* $K = 50$*,* $L = 0$*,* $S = 200$*. Number of spatial grid points:* $m = 4000$*.*

Similarly, it is not clear that a hyperbolic contour is preferable to a parabolic one. In [13] the hyperbolic contour was used based on the assumption that the resolvent of the operator can be bounded with respect to a sectorial region. Certainly in the case of the Black–Scholes operator (3.2) the results of Theorem 3.1 show that a parabolic region is more natural. Moreover, when one looks at the proposed contours in [13] they move unboundedly into the right half-plane as the volatility $\sigma \to 0$, which will cause a loss of accuracy and increased sensitivity to roundoff errors.

We offer a numerical comparison of the two methods in Figure 5.1. The test example corresponds to Example 4.1 in [13], a European put option problem with parameters listed in the caption of the figure. The parameters in the method of [13] were chosen according to Table 3 of that paper. We used Implementation I of the method proposed here with the trapezoidal rule; see subsection 2.4.

The explicit solution of the problem under consideration is known [2] and was used to compute the total error in the method. With the given parameter values the

grid spacing amounts to $\Delta s = 0.05$, and with this spacing the spatial error is just less than 10^{-5}, as can be seen in the figure. The method proposed here achieves this accuracy using $N \geq 7$, while $N \geq 11$ is required for the method of [13]. As mentioned above, however, improved parameter selection may reduce the error shown in curve (b), but that has yet to be accomplished.

6. Conclusions. In this paper we considered the numerical solution of linear systems of ODEs derived from semidiscretization of time-dependent PDEs, with emphasis on applications in mathematical finance. Our method is similar to a recently proposed method by Lee and Sheen [13], but based on a parabolic rather than a hyperbolic contour. Using a parabolic contour is natural when the numerical range of the PDE operator can be bounded suitably by a parabola. Indeed, numerical comparisons presented in section 5 revealed higher accuracy than the results reported in [13].

One contribution of this paper is the generalization of the results of [31, 32] for the estimation of the optimal parameters that determine the parabolic contour plus the step-size of the quadrature rule. These results are quite general and cover many pricing problems of mathematical finance, with various exotic options. The only requirement is that the problem can be modeled by a PDE, the semidiscretization of which leads to a linear system of ODEs (1.1) with matrix A whose numerical range can be bounded by a parabola. Of course, many PDEs outside the field of mathematical finance are also covered by this theory. In addition, the semidiscretization can be based on second-order finite differences (as considered here), higher-order differences, finite elements, or spectral methods. In fact, if the discretization is based on higher-order methods, then one can expect a better correspondence between the numerical ranges of the matrix A and the operator $\mathcal{A}$, at least in the critical region around the real axis. This should justify even better the practice of using the numerical range of $\mathcal{A}$ (instead of A) in our parameter selection strategy. Indeed, in [31] a spectral method was used to solve a model advection-diffusion equation with excellent results.

A second contribution of the paper is an investigation into efficient methods for solving the shifted linear systems (1.7), which represent the major computational cost of the method. The fact that these systems are independent and can be solved in parallel has been a major selling point for the contour integral method; cf. [7, 8, 22, 23]. Here we went one step further: by actually implementing the method on a multiprocessor machine we could quantify the advantage; see Figures 4.2 and 4.3. Similarly, the fact that Krylov iteration methods could be efficient for solving the shifted systems (1.7) has been mentioned in the literature. As far as we know, however, the algorithm described in the appendix is the first attempt at actually incorporating a version of Krylov iteration into the contour integral method to solve PDEs.

In our numerical tests we considered both the Black–Scholes and Heston equations. For both PDEs we approximately established the geometric convergence rate predicted by the theory of section 2. For the Heston PDE we performed further comparisons with recently studied ADI methods [11]. It was found that the contour integral method, with both parallel and Krylov implementations, is more efficient in terms of execution time than the ADI methods at higher accuracy requirements. In the Heston application, the actual break-even point was found to be in the error range 10^{-4}–$10^{-3.5}$, which is within the typical accuracy specifications of financial applications. In other applications, where higher accuracy is required, the contour integral method will fare relatively better in such timing comparisons. For lower accuracies,

the ADI methods are superior. More conventional methods, such as ADI, are also expected to perform better for strongly advection-dominated problems.

With further research the contour method is likely to be improved so that the break-even point could occur at even lower accuracies. On the numerical side, the Krylov method of the appendix is just a starting point and improvements can be expected. On the theoretical side, better estimates of the numerical range of the Heston operator may lead to better parameter selection (as described in section 2) and hence increased accuracy.

Appendix. Here we discuss strategies for the efficient solution of the linear systems (1.7), which represent the major computational cost of the contour integral method.

First, one could use a sparse direct method based on some preordering algorithm. For a small improvement, note that the preordering algorithm plus accompanying symbolic analysis need to be executed only once, since the sparsity pattern of the coefficient matrices in (1.7) does not change. As pointed out in the introduction, this approach is suitable for implementation as a parallel for-loop on a multiprocessor machine, since there is no exchange of information between these linear systems. In the numerical experiments reported here, we have used the sparse direct solver for unsymmetric systems in T. Davis' package UMFPACK [3, 4], accessed via the backslash operator in MATLAB (version R2010a). For the parallel implementation, we have used MATLAB's Parallel Toolbox on an 8 processor machine (Intel Xeon E5462 / 2.8 GHz processors, RAM 31.5 GB).

An alternative, serial implementation is to use iterative rather than direct methods for the solution of (1.7). One of the advantages of iterative methods is that they can be stopped when a required error tolerance is reached; by contrast, direct methods have to run to completion regardless of accuracy requirements. Financial applications typically require only low to moderate accuracy, which makes an iterative method more attractive.

It is further known that the class of solvers based on Krylov subspace iteration is effective for shifted linear systems such as (1.7). This is because all the matrices $z_k I - A$ share the same Krylov basis, and hence this has to be constructed only once; see [18, 25]. Technically this statement is true only if all the systems have the same right-hand side, which is why we proposed the alternate implementation (1.8). We shall therefore consider the shifted systems in the form

$$\big(z_k I - A\big)\, \boldsymbol{u}_k = \boldsymbol{c}, \qquad k = 0, 1, \ldots, N-1, \tag{A.1}$$

where $\boldsymbol{c}$ does not depend on k. In our application, $\boldsymbol{c}$ is defined in (1.9). We have dropped the hat on $\widehat{\boldsymbol{u}}_k$ to simplify notation.

For most iterative methods, including Krylov iteration, some form of preconditioning is required to accelerate convergence. Following [18], we use $z_0 I - A$ as a common preconditioner for the systems (1.7), i.e.,

$$(z_0 I - A)^{-1}(z_k I - A)\, \boldsymbol{u}_k = (z_0 I - A)^{-1} \boldsymbol{c}.$$

It is slightly more efficient to use a real matrix as preconditioner as opposed to one with a complex diagonal. For this implementation we therefore recommend the use of the trapezoidal rule (real z_0) as opposed to the midpoint rule (complex z_0).

The preconditioned system can be written as

$$\big((z_k - z_0)B + I\big)\, \boldsymbol{u}_k = \boldsymbol{d}, \tag{A.2}$$

where

$$B = (z_0 I - A)^{-1}, \qquad \boldsymbol{d} = (z_0 I - A)^{-1} \boldsymbol{c}.$$

By applying ℓ ($\leq m$) steps of the standard Arnoldi procedure to the $m \times m$ matrix B with initial vector $\boldsymbol{d}$, a relation of the form

$$V_\ell^T B\, V_\ell = H_\ell \tag{A.3}$$

is obtained, where V_ℓ and H_ℓ are real $m \times \ell$ and $\ell \times \ell$ matrices, respectively [21, p. 155]. H_ℓ is upper-Hessenberg, and V_ℓ has orthonormal columns, with the normalized version of $\boldsymbol{d}$ as its first column.

The full orthogonalization method enforces the following projection condition on (A.2):

$$V_\ell^T \big((z_k - z_0)B + I\big)\, V_\ell\, \boldsymbol{y}_k = V_\ell^T \boldsymbol{d}.$$

Here $V_\ell\, \boldsymbol{y}_k$ represents the approximation to $\boldsymbol{u}_k$; for details, see [21, p. 159]. By making use of (A.3) and the orthogonality properties of V_ℓ, one obtains

$$\big((z_k - z_0)H_\ell + I\big)\, \boldsymbol{y}_k = \delta \boldsymbol{e}_1,$$

where $\delta = \|\boldsymbol{d}\|_2$ and $\boldsymbol{e}_1$ is the unit vector in the first coordinate direction. When $\ell \ll m$ this $\ell \times \ell$ Hessenberg system is solved cheaply for $k = 1, \ldots, N-1$, and the required approximations to $\boldsymbol{u}_k$ are computed as $V_\ell\, \boldsymbol{y}_k$.

In the Arnoldi iteration that leads to (A.3), it is necessary to compute matrix-vector products of the form $B\,\boldsymbol{v}$, which means solving $(z_0 I - A)\boldsymbol{w} = \boldsymbol{v}$ for several $\boldsymbol{v}$. This is done by an a priori LU factorization of $z_0 I - A$, again using the algorithms from UMFPACK.

We remark that while the matrix $z_0 I - A$ is a good preconditioner for $z_k I - A$ when k is 1 or 2, it may not be the case for larger values of k. However, it is not necessary to compute the systems corresponding to large k to the same absolute accuracy as the systems corresponding to small k. This is because of the damping factor $e^{z_k t}$ in (1.6). Similarly, it is not clear what the best stopping criterion is for the Arnoldi iteration in the present context. By experimentation we have found that the simple choice $\ell = N + 1$ is in many cases sufficient to guarantee that the error in the solution of the shifted linear systems is on the same order of magnitude (or less) as the approximation error of the contour integral method discussed in section 2. A theoretical justification is still lacking, however.

We recall that our implementation based on (1.8) is effectively a method for computing the matrix exponential times a given vector. The question arises why one does not use a standard method for the computation of the matrix exponential, such as one of the methods described in [10, 19]. First, we point out that several of these methods, including the one implemented in MATLAB's `expm` function, cannot utilize the sparsity of the matrix. Second, there exist Krylov-based methods that can exploit the sparsity, but the issue of preconditioning does not seem fully resolved yet. We have tried, for example, the (unpreconditioned) Krylov method described in [19], and it was not as effective as the method outlined above.

Acknowledgments. The authors would like to thank two anonymous referees for many good suggestions that led to substantial improvements to the original version of this paper. Included in these was the encouragement to actually implement a Krylov

iteration method, something that was only hinted at in the original draft. The advice of two experts on Krylov methods, Karl Meerbergen and Valeria Simoncini, proved to be vital in this context. Iain Duff gave advice on the direct solvers. The authors also thank Sven Foulon for an implementation of the closed-form analytical formula for European call option prices in the Heston model.

REFERENCES

[1] H. Albrecher, P. Mayer, W. Schoutens, and J. Tistaert, *The little Heston trap*, Wilmott Mag., January 2007, pp. 83–92.
[2] F. Black and M. Scholes, *The pricing of options and corporate liabilities*, J. Polit. Econ., 81 (1973), pp. 637–654.
[3] T. A. Davis, *Algorithm* 832*: UMFPACK V*4.3*—an unsymmetric-pattern multifrontal method*, ACM Trans. Math. Software, 30 (2004), pp. 196–199.
[4] T. A. Davis, *A column pre-ordering strategy for the unsymmetric-pattern multifrontal method*, ACM Trans. Math. Software, 30 (2004), pp. 165–195.
[5] I. P. Gavrilyuk, W. Hackbusch, and B. N. Khoromskij, *H-matrix approximation for the operator exponential with applications*, Numer. Math., 92 (2002), pp. 83–111.
[6] I. P. Gavrilyuk, W. Hackbusch, and B. N. Khoromskij, *Data-sparse approximation to a class of operator-valued functions*, Math. Comp., 74 (2005), pp. 681–708.
[7] I. P. Gavrilyuk and V. L. Makarov, *Exponentially convergent parallel discretization methods for the first order evolution equations*, Comput. Methods Appl. Math., 1 (2001), pp. 333–355.
[8] I. P. Gavrilyuk and V. L. Makarov, *Exponentially convergent algorithms for the operator exponential with applications to inhomogeneous problems in Banach spaces*, SIAM J. Numer. Anal., 43 (2005), pp. 2144–2171.
[9] S. L. Heston, *A closed-form solution for options with stochastic volatility with applications to bond and currency options*, Rev. Finan. Stud., 6 (1993), pp. 327–343.
[10] N. J. Higham, *Functions of Matrices: Theory and Computation*, SIAM, Philadelphia, PA, 2008.
[11] K. J. in 't Hout and S. Foulon, *ADI finite difference schemes for option pricing in the Heston model with correlation*, Int. J. Numer. Anal. Model., 7 (2010), pp. 303–320.
[12] K. J. in 't Hout and B. D. Welfert, *Stability of ADI schemes applied to convection-diffusion equations with mixed derivative terms*, Appl. Numer. Math., 57 (2007), pp. 19–35.
[13] S. Lee and D. Sheen, *Laplace transformation method for the Black–Scholes equation*, Int. J. Numer. Anal. Model., 6 (2009), pp. 642–658.
[14] M. López-Fernández and C. Palencia, *On the numerical inversion of the Laplace transform of certain holomorphic mappings*, Appl. Numer. Math., 51 (2004), pp. 289–303.
[15] M. López-Fernández, C. Palencia, and A. Schädle, *A spectral order method for inverting sectorial Laplace transforms*, SIAM J. Numer. Anal., 44 (2006), pp. 1332–1350.
[16] E. Martensen, *Zur numerischen Auswertung uneigentlicher Integrale*, ZAMM Z. Angew. Math. Mech., 48 (1968), pp. T83–T85.
[17] J. McNamee, *Error-bounds for the evaluation of integrals by the Euler-Maclaurin formula and by Gauss-type formulae*, Math. Comp., 18 (1964), pp. 368–381.
[18] K. Meerbergen, *The solution of parametrized symmetric linear systems*, SIAM J. Matrix Anal. Appl., 24 (2003), pp. 1038–1059.
[19] C. Moler and C. Van Loan, *Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later*, SIAM Rev., 45 (2003), pp. 3–49.
[20] S. C. Reddy and L. N. Trefethen, *Pseudospectra of the convection-diffusion operator*, SIAM J. Appl. Math., 54 (1994), pp. 1634–1649.
[21] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, Philadelphia, PA, 2003.
[22] D. Sheen, I. H. Sloan, and V. Thomée, *A parallel method for time-discretization of parabolic problems based on contour integral representation and quadrature*, Math. Comp., 69 (2000), pp. 177–195.
[23] D. Sheen, I. H. Sloan, and V. Thomée, *A parallel method for time discretization of parabolic equations based on Laplace transformation and quadrature*, IMA J. Numer. Anal., 23 (2003), pp. 269–299.
[24] S. E. Shreve, *Stochastic Calculus for Finance* II, Springer-Verlag, New York, 2004.
[25] V. Simoncini and D. B. Szyld, *Recent computational developments in Krylov subspace methods for linear systems*, Numer. Linear Algebra Appl., 14 (2007), pp. 1–59.

[26] F. Stenger, *Numerical Methods Based on Sinc and Analytic Functions*, Springer-Verlag, New York, 1993.
[27] A. Talbot, *The accurate numerical inversion of Laplace transforms*, J. Inst. Math. Appl., 23 (1979).
[28] D. Tavella and C. Randall, *Pricing Financial Instruments*, Wiley, New York, 2000.
[29] L. N. Trefethen and M. Embree, *Spectra and Pseudospectra*, Princeton University Press, Princeton, NJ, 2005.
[30] J. L. M. van Dorsselaer, J. F. B. M. Kraaijevanger, and M. N. Spijker, *Linear stability analysis in the numerical solution of initial value problems*, Acta Numer., 2 (1993), pp. 199–237.
[31] J. A. C. Weideman, *Improved contour integral methods for parabolic PDEs*, IMA J. Numer. Anal., 30 (2010), pp. 334–350.
[32] J. A. C. Weideman and L. N. Trefethen, *Parabolic and hyperbolic contours for computing the Bromwich integral*, Math. Comp., 76 (2007), pp. 1341–1356.
[33] P. Wilmott, *Derivatives*, reprinted edition, Wiley, Chichester, England, 1999.
[34] G. Winkler, T. Apel, and U. Wystup, *Valuation of options in Heston's stochastic volatility model using finite element methods*, in Foreign Exchange Risk, J. Hakala and U. Wystup, eds., Risk Books, London, 2002, pp. 283–303.
[35] T. G. Wright, *EigTool*, http://www.comlab.ox.ac.uk/pseudospectra/eigtool, 2002.