Electronic Transactions on Numerical Analysis.
Volume 31, pp. 86-109, 2008.

ISSN 1068-9613.

ETNA
Kent State University
http://etna.math.kent.edu

ENHANCEMENT OF KRYLOV SUBSPACE SPECTRAL METHODS BY BLOCK LANCZOS ITERATION*

JAMES V. LAMBERS†

Dedicated to the memory of Gene H. Golub, 1932-2007

Abstract. This paper presents a modification of Krylov subspace spectral (KSS) methods, which build on the work of Golub, Meurant and others, pertaining to moments and Gaussian quadrature to produce high-order accurate approximate solutions to variable-coefficient time-dependent PDEs. Whereas KSS methods currently use Lanczos iteration to compute the needed quadrature rules, our modification uses block Lanczos iteration in order to avoid the need to compute two quadrature rules for each component of the solution, or use perturbations of quadrature rules. It will be shown that, under reasonable assumptions on the coefficients of the problem, a 1-node KSS method is unconditionally stable, and methods with more than one node are shown to possess favorable stability properties as well. Numerical results suggest that block KSS methods are significantly more accurate than their non-block counterparts.

Key words. Spectral methods, Gaussian quadrature, variable-coefficient, block Lanczos method, stability, heat equation.

AMS subject classifications. 65M12, 65M70, 65D32, 65F25.

1. Introduction.

Consider the following initial-boundary value problem in one space dimension,

$$u_t + Lu = 0, \quad \text{on } (0, 2\pi) \times (0, \infty), \tag{1.1}$$

$$u(x, 0) = f(x), \quad 0 < x < 2\pi, \tag{1.2}$$

with periodic boundary conditions

$$u(0, t) = u(2\pi, t), \quad t > 0. \tag{1.3}$$

The operator L is a second-order differential operator of the form

$$Lu = -(p(x)u_x)_x + q(x)u, \tag{1.4}$$

where $p(x)$ is a positive function and $q(x)$ is a positive smooth function. It follows that L is self-adjoint and positive definite.

In [15, 17] a class of methods, called Krylov subspace spectral (KSS) methods, was introduced for the purpose of solving time-dependent, variable-coefficient problems such as this one. These methods are based on the application of techniques developed by Golub and Meurant in [6], originally for the purpose of computing elements of the inverse of a matrix, to the elements of the matrix exponential of an operator. In these references it has been shown that KSS methods, by employing different approximations of the solution operator for each Fourier component of the solution, achieve higher-order accuracy in time than other Krylov subspace methods (see, for example, [12]) for stiff systems of ODE, and, as shown in [13], they are also quite stable, considering that they are explicit methods.

*Received December 3, 2007. Accepted August 14, 2008. Published online on January 21, 2009. Recommended by Martin H. Gutknecht.
†Department of Energy Resources Engineering, Stanford University, Stanford, CA 94305-2220 (lambers@stanford.edu).

In this paper, we consider whether these methods can be enhanced in terms of accuracy, stability, or any other measure, by using a single block Gaussian quadrature rule to compute each Fourier component of the solution, instead of two standard Gaussian rules. KSS methods take into account the solution from the previous time step only through a perturbation of initial vectors used in Lanczos iteration. While this enables KSS methods to handle stiff systems very effectively by giving individual attention to each Fourier component, and also yields high-order operator splittings (see [14]), it is worthwhile to consider whether it is best to use quadrature rules whose nodes are determined primarily by each basis function used to represent the solution, instead of the solution itself. Intuitively, a block quadrature rule that uses a basis function and the solution should strike a better balance between the competing goals of computing each component with an approximation that is, in some sense, optimal for that component in order to deal with stiffness, and giving the solution a prominent role in computing the quadrature rules that are used to evolve it forward in time.

Section 2 reviews the main properties of KSS methods, including algorithmic details and results concerning local accuracy. They use perturbations of quadratic forms to compute Fourier components of the solution, where the perturbation is in the direction of the solution from the previous time step. In Section 3, we present the modified KSS method that uses block Lanczos iteration to approximate each Fourier component of the solution by a single Gaussian quadrature rule. In Section 4, we study the convergence behavior of the block method. Numerical results are presented in Section 5. In Section 6, various extensions and future directions are discussed.

2. Krylov subspace spectral methods. We begin with a review of the main aspects of KSS methods. Let $S(t) = \exp(-Lt)$ represent the exact solution operator of the problem (1.1)-(1.3), and let $\langle \cdot, \cdot \rangle$ denote the standard inner product on $[0, 2\pi]$

$$\langle f(x), g(x) \rangle = \int_0^{2\pi} \overline{f(x)} g(x)\, dx.$$

Krylov subspace spectral methods, introduced in [15, 17], use Gaussian quadrature on the spectral domain to compute the Fourier components of the solution. These methods are time-stepping algorithms that compute the solution at times $t_1, t_2, \ldots$, where $t_n = n\Delta t$ for some choice of Δt. Given the computed solution $\tilde{u}(x, t_n)$ at time t_n, the solution at time t_{n+1} is computed by approximating the Fourier components that would be obtained by applying the exact solution operator to $\tilde{u}(x, t_n)$, i.e.,

$$\hat{u}(\omega, t_{n+1}) = \left\langle \frac{1}{\sqrt{2\pi}} e^{i\omega x}, S(\Delta t)\tilde{u}(x, t_n) \right\rangle. \tag{2.1}$$

Krylov subspace spectral methods approximate these components with higher-order temporal accuracy than traditional spectral methods and time-stepping schemes. We briefly review how these methods work.

We discretize the functions defined on $[0, 2\pi]$ on an N-point uniform grid with spacing $\Delta x = 2\pi/N$. With this discretization, the operator L and the solution operator $S(\Delta t)$ can be approximated by $N \times N$ matrices that represent linear operators on the space of grid functions, and the quantity (2.1) can be approximated by a bilinear form

$$\hat{u}(\omega, t_{n+1}) \approx \sqrt{\Delta x}\, \hat{\mathbf{e}}_\omega^H S_N(\Delta t) \mathbf{u}^n. \tag{2.2}$$

In this formula, we have

$$[\hat{\mathbf{e}}_\omega]_j = \frac{1}{\sqrt{N}} e^{i\omega j \Delta x}, \quad [\mathbf{u}^n]_j = u(j\Delta x, t_n),$$

and

$$S_N(t) = \exp(-L_N t), \quad [L_N]_{jk} = -p[D_N^2]_{jk} + q(j\Delta x), \tag{2.3}$$

where D_N is a discretization of the differentiation operator that is defined on the space of grid functions. Our goal is to approximate (2.2) by computing an approximation to

$$[\hat{\mathbf{u}}^{n+1}]_\omega = \hat{\mathbf{e}}_\omega^H \mathbf{u}^{n+1} = \hat{\mathbf{e}}_\omega^H S_N(\Delta t)\mathbf{u}^n. \tag{2.4}$$

In [6] Golub and Meurant describe a method for computing quantities of the form

$$\mathbf{u}^T f(A)\mathbf{v}, \tag{2.5}$$

where $\mathbf{u}$ and $\mathbf{v}$ are N-vectors, A is an $N \times N$ symmetric positive definite matrix, and f is a smooth function. Our goal is to apply this method with $A = L_N$, where L_N was defined in (2.3), $f(\lambda) = \exp(-\lambda t)$ for some t, and the vectors $\mathbf{u}$ and $\mathbf{v}$ are derived from $\hat{\mathbf{e}}_\omega$ and $\mathbf{u}^n$.

The basic idea is as follows: since the matrix A is symmetric positive definite, it has real eigenvalues

$$b = \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N = a > 0,$$

and corresponding orthogonal eigenvectors $\mathbf{q}_j$, $j = 1, \ldots, N$. Therefore, the quantity (2.5) can be rewritten as

$$\mathbf{u}^T f(A)\mathbf{v} = \sum_{j=1}^N f(\lambda_j)\mathbf{u}^T \mathbf{q}_j \mathbf{q}_j^T \mathbf{v}.$$

We let $a = \lambda_N$ be the smallest eigenvalue, $b = \lambda_1$ be the largest eigenvalue, and define the measure $\alpha(\lambda)$ by

$$\alpha(\lambda) = \begin{cases} 0, & \text{if } \lambda < a, \\ \displaystyle\sum_{j=i}^N \alpha_j \beta_j, & \text{if } \lambda_i \leq \lambda < \lambda_{i-1},\ i = 2, \ldots, N, \\ \displaystyle\sum_{j=1}^N \alpha_j \beta_j, & \text{if } b \leq \lambda, \end{cases} \tag{2.6}$$

with $\alpha_j = \mathbf{u}^T \mathbf{q}_j$ and $\beta_j = \mathbf{q}_j^T \mathbf{v}$. If this measure is positive and increasing, then the quantity (2.5) can be viewed as a Riemann-Stieltjes integral

$$\mathbf{u}^T f(A)\mathbf{v} = I[f] = \int_a^b f(\lambda)\, d\alpha(\lambda).$$

As discussed in [3, 4, 5, 6], the integral $I[f]$ can be bounded using either Gauss, Gauss-Radau, or Gauss-Lobatto quadrature rules, all of which yield an approximation of the form

$$I[f] = \sum_{j=1}^K w_j f(t_j) + R[f],$$

where the nodes t_j, $j = 1, \ldots, K$, as well as the weights w_j, $j = 1, \ldots, K$, can be obtained using the symmetric Lanczos algorithm if $\mathbf{u} = \mathbf{v}$, and the unsymmetric Lanczos algorithm if $\mathbf{u} \neq \mathbf{v}$; see [10].

ETNA
Kent State University
http://etna.math.kent.edu

In the case $\mathbf{u} \neq \mathbf{v}$, there is the possibility that the weights may not be positive, which destabilizes the quadrature rule; see [1] for details. Therefore, it is best to handle this case by rewriting (2.5) using decompositions such as

$$\mathbf{u}^T f(A)\mathbf{v} = \frac{1}{\delta}[\mathbf{u}^T f(A)(\mathbf{u} + \delta\mathbf{v}) - \mathbf{u}^T f(A)\mathbf{u}], \tag{2.7}$$

where δ is a small constant. Guidelines for choosing an appropriate value for δ can be found in [17, Section 2.2].

Employing these quadrature rules yields the following basic process (for details see [15, 17] for computing the Fourier coefficients of $\mathbf{u}^{n+1}$ from $\mathbf{u}^n$):

for $\omega = -N/2 + 1, \ldots, N/2$
  Choose a scaling constant δ_ω
  Compute $u_1 \approx \hat{\mathbf{e}}_\omega^H S_N(\Delta t)\hat{\mathbf{e}}_\omega$
    using the symmetric Lanczos algorithm
  Compute $u_2 \approx \hat{\mathbf{e}}_\omega^H S_N(\Delta t)(\hat{\mathbf{e}}_\omega + \delta_\omega \mathbf{u}^n)$
    using the unsymmetric Lanczos algorithm
  $[\hat{\mathbf{u}}^{n+1}]_\omega = (u_2 - u_1)/\delta_\omega$
end

It is assumed that when the Lanczos algorithm (symmetric or unsymmetric) is employed, K iterations are performed to obtain the K quadrature nodes and weights. It should be noted that the constant δ_ω plays the role of δ in the decomposition (2.7), and the subscript ω is used to indicate that a different value may be used for each wave number $\omega = -N/2 + 1, \ldots, N/2$. Also, in the presentation of this algorithm in [17] a polar decomposition is used instead of (2.7), and it is applied to sines and cosines instead of complex exponential functions.

This algorithm has high-order temporal accuracy, as indicated by the following theorem. Let $BL_N([0, 2\pi]) = \text{span}\{e^{-i\omega x}\}_{\omega=-N/2+1}^{N/2}$ denote a space of bandlimited functions with at most N nonzero Fourier components.

THEOREM 2.1. *Let L be a self-adjoint m-th order positive definite differential operator on $C_p([0, 2\pi])$ with coefficients in $BL_N([0, 2\pi])$, and let $f \in BL_N([0, 2\pi])$. Then the preceding algorithm, applied to the problem (1.1)-(1.3), is consistent, i.e.,*

$$[\hat{\mathbf{u}}^1]_\omega - \hat{u}(\omega, \Delta t) = O(\Delta t^{2K}),$$

for $\omega = -N/2 + 1, \ldots, N/2$.

Proof. See [17, Lemma 2.1, Theorem 2.4]. □

The preceding result can be compared to the accuracy achieved by an algorithm, described by Hochbruck and Lubich in [12], for computing $e^{A\Delta t}\mathbf{v}$ for a given matrix A and vector $\mathbf{v}$ using the unsymmetric Lanczos algorithm. As discussed in [12], this algorithm can be used to compute the solution of some ODEs without time-stepping, but this becomes less practical for ODEs arising from a semi-discretization of problems such as (1.1)-(1.3), due to their stiffness. In this situation, it is necessary to either use a high-dimensional Krylov subspace, in which case reorthogonalization is required, or one can resort to time-stepping, in which case the local temporal error is only $O(\Delta t^K)$, assuming a K-dimensional Krylov subspace. Regardless of which remedy is used, the computational effort needed to compute the solution at a fixed time T increases substantially.

The difference between Krylov subspace spectral methods and the approach described in [12] is that in the former a different K-dimensional Krylov subspace is used for each Fourier component, instead of the same subspace for all components as in the latter. As can be seen from numerical results comparing the two approaches in [17], using the same subspace for all components causes a loss of accuracy as the number of grid points increases, whereas Krylov subspace spectral methods do not suffer from this phenomenon.

Using a perturbation of the form (2.7) is only one approach for computing bilinear forms such as (2.5) in the case where $\mathbf{u} \neq \mathbf{v}$. In [13], this approach was numerically stabilized by the use of formulas for the derivatives of the nodes and weights with respect to the parameter δ. However, two quadrature rules are needed to compute each component, as well as the unsymmetric Lanczos algorithm, which is much less well-behaved than its symmetric counterpart. A polar decomposition may be used, but that also requires two quadrature rules, although the symmetric Lanczos algorithm can be used for both. An approach that requires only one quadrature rule per component involves block Lanczos iteration. The result is a block-tridiagonal Hermitian matrix, from which the nodes and weights for the quadrature rule can be obtained. It is worthwhile to examine whether a block approach might be more effective than the original algorithm.

3. Block formulation. In this section, we describe how we can compute elements of functions of matrices using block Gaussian quadrature. We then present a modification of KSS methods that employs this block approach.

3.1. Block Gaussian quadrature. If we compute (2.5) using the formula (2.7) or the *polar decomposition*

$$\frac{1}{4}[(\mathbf{u}+\mathbf{v})^T f(A)(\mathbf{u}+\mathbf{v}) - (\mathbf{v}-\mathbf{u})^T f(A)(\mathbf{v}-\mathbf{u})], \tag{3.1}$$

then we would have to run the process for approximating an expression of the form (2.5) with two starting vectors. Instead we consider

$$\begin{bmatrix} \mathbf{u} & \mathbf{v} \end{bmatrix}^T f(A) \begin{bmatrix} \mathbf{u} & \mathbf{v} \end{bmatrix}$$

which results in the 2×2 matrix

$$\int_a^b f(\lambda)\, d\mu(\lambda) = \begin{bmatrix} \mathbf{u}^T f(A)\mathbf{u} & \mathbf{u}^T f(A)\mathbf{v} \\ \mathbf{v}^T f(A)\mathbf{u} & \mathbf{v}^T f(A)\mathbf{v} \end{bmatrix},$$

where $\mu(\lambda)$ is a 2×2 matrix function of λ, each entry of which is a measure of the form $\alpha(\lambda)$ from (2.6).

In [6] Golub and Meurant show how a block method can be used to generate quadrature formulas. We will describe this process here in more detail. The integral $\int_a^b f(\lambda)\, d\mu(\lambda)$ is now a 2×2 symmetric matrix and the most general K-node quadrature formula is of the form

$$\int_a^b f(\lambda)\, d\mu(\lambda) = \sum_{j=1}^{K} W_j f(T_j) W_j + \text{error}, \tag{3.2}$$

with T_j and W_j being symmetric 2×2 matrices. Equation (3.2) can be simplified using

$$T_j = Q_j \Lambda_j Q_j^T,$$

where Q_j is the eigenvector matrix and Λ_j the 2×2 diagonal matrix containing the eigenvalues. Hence,

$$\sum_{j=1}^{K} W_j f(T_j) W_j = \sum_{j=1}^{K} W_j Q_j f(\Lambda_j) Q_j^T W_j,$$

ETNA
Kent State University
http://etna.math.kent.edu

and, writing

$$W_j Q_j f(\Lambda_j) Q_j^T W_j = f(\lambda_1)\mathbf{z}_1\mathbf{z}_1^T + f(\lambda_2)\mathbf{z}_2\mathbf{z}_2^T,$$

with $\mathbf{z}_k = W_j Q_j \mathbf{e}_k$ for $k = 1, 2$, we get for the quadrature rule

$$\int_a^b f(\lambda)\, d\mu(\lambda) = \sum_{j=1}^{K} f(t_j)\mathbf{v}_j\mathbf{v}_j^T + \text{error},$$

where t_j is a scalar and $\mathbf{v}_j$ is a vector with two components.

We now describe how to obtain the scalar nodes t_j and the associated vectors $\mathbf{v}_j$. In [6] it is shown that there exist orthogonal matrix polynomials such that

$$\lambda\, p_{j-1}(\lambda) = p_j(\lambda)B_j + p_{j-1}(\lambda)M_j + p_{j-2}(\lambda)B_{j-1}^T,$$

with $p_0(\lambda) = I_2$ and $p_{-1}(\lambda) = 0$. We can write the last equation as

$$\lambda[p_0(\lambda), \ldots, p_{K-1}(\lambda)] = [p_0(\lambda), \ldots, p_{K-1}(\lambda)]\mathcal{T}_K + [0, \ldots, 0, p_K(\lambda)B_K],$$

with

$$\mathcal{T}_K = \begin{bmatrix} M_1 & B_1^T & & & \\ B_1 & M_2 & B_2^T & & \\ & \ddots & \ddots & \ddots & \\ & & B_{K-2} & M_{K-1} & B_{K-1}^T \\ & & & B_{K-1} & M_K \end{bmatrix}, \tag{3.3}$$

which is a block-triangular matrix. Therefore, we can define the quadrature rule as

$$\int_a^b f(\lambda)\, d\mu(\lambda) = \sum_{j=1}^{2K} f(\lambda_j)\mathbf{v}_j\mathbf{v}_j^T + \text{error}, \tag{3.4}$$

where $2K$ is the order of the matrix $\mathcal{T}_K$, λ_j is one of its eigenvalues, and $\mathbf{u}_j$ is the vector consisting of the first two elements of the corresponding normalized eigenvector.

To compute the matrices M_j and B_j, we use the block Lanczos algorithm, which was proposed by Golub and Underwood in [9]. Let X_0 be an $N \times 2$ given matrix, such that $X_1^T X_1 = I_2$. Let $X_0 = 0$ be an $N \times 2$ matrix. Then, for $j = 1, \ldots,$ we compute

$$\begin{aligned} M_j &= X_j^T A X_j, \\ R_j &= A X_j - X_j M_j - X_{j-1}B_{j-1}^T, \\ X_{j+1}B_j &= R_j. \end{aligned} \tag{3.5}$$

The last step of the algorithm is the QR decomposition of R_j (see [8]) such that X_j is $N \times 2$, with $X_j^T X_j = I_2$. The matrix B_j is 2×2 upper triangular. The other coefficient matrix M_j is 2×2 and symmetric. The matrix R_j can eventually be rank deficient, and in that case B_j is singular. The solution of this problem is given in [9].

3.2. Block KSS methods. We are now ready to describe block KSS methods. For each wave number $\omega = -N/2 + 1, \ldots, N/2$, we define

$$R_0(\omega) = \begin{bmatrix} \hat{\mathbf{e}}_\omega & \mathbf{u}^n \end{bmatrix}$$

and then compute the QR factorization

$$R_0(\omega) = X_1(\omega)B_0(\omega),$$

which yields

$$X_1(\omega) = \begin{bmatrix} \hat{\mathbf{e}}_\omega & \mathbf{u}^n_\omega/\|\mathbf{u}^n_\omega\|_2 \end{bmatrix}, \quad B_0(\omega) = \begin{bmatrix} 1 & \hat{\mathbf{e}}_\omega^H \mathbf{u}^n \\ 0 & \|\mathbf{u}^n_\omega\|_2 \end{bmatrix},$$

where

$$\mathbf{u}^n_\omega = \mathbf{u}^n - \hat{\mathbf{e}}_\omega \hat{\mathbf{e}}_\omega^H \mathbf{u}^n.$$

We then carry out the block Lanczos iteration described in (3.5) to obtain a block-tridiagonal matrix

$$\mathcal{T}_K(\omega) = \begin{bmatrix} M_1(\omega) & B_1(\omega)^H & & & \\ B_1(\omega) & M_2(\omega) & B_2(\omega)^H & & \\ & \ddots & \ddots & \ddots & \\ & & B_{K-2}(\omega) & M_{K-1}(\omega) & B_{K-1}(\omega)^H \\ & & & B_{K-1}(\omega) & M_K(\omega) \end{bmatrix}.$$

Now, we can express each Fourier component of the approximate solution at the next time step as

$$[\hat{\mathbf{u}}^{n+1}]_\omega = \left[B_0^H E_{12}^H \exp(-\mathcal{T}_K(\omega)\Delta t) E_{12} B_0\right]_{12} \tag{3.6}$$

where

$$E_{12} = \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix}.$$

The computation of $E_{12}^H \exp(-\mathcal{T}_K(\omega)\Delta t)E_{12}$ consists of evaluating the eigenvalues and eigenvectors of $\mathcal{T}_K(\omega)$, in order to obtain the nodes and weights for Gaussian quadrature, as described earlier in this section.

3.3. Implementation. In [18], it was demonstrated that recursion coefficients for all wave numbers $\omega = -N/2+1, \ldots, N/2$, can be computed simultaneously by regarding them as functions of ω, and using symbolic calculus to apply differential operators analytically as much as possible. As a result, KSS methods require $O(N \log N)$ floating-point operations per time step, which is comparable to other time-stepping methods. The same approach can be applied to block KSS methods. For both types of methods, it can be shown that for a K-node Gaussian rule or block Gaussian rule, K applications of the operator L_N to the previous solution $\mathbf{u}^n$ are needed.

4. Convergence analysis. We now examine the convergence of block KSS methods by first investigating their consistency and stability. As shown in [13, 17], the original KSS methods are high-order accurate in time, but are also explicit methods that possess stability properties characteristic of implicit methods, so it is desired that block KSS methods share both of these traits with their predecessors.

4.1. Consistency. As shown in [7], the error in a K-node block Gaussian quadrature rule of the form (3.4) is

$$R(f) = \frac{f^{(2K)}(\eta)}{(2K)!} \int_a^b \prod_{j=1}^{2K} (\lambda - \lambda_j)\, d\mu(\lambda). \tag{4.1}$$

It follows (see [2]) that the rule is exact for polynomials of degree up to $2K-1$. The above form of the remainder can be obtained using results from [19].

THEOREM 4.1. *Let L be a self-adjoint mth-order positive definite differential operator on $C_p([0,2\pi])$ with coefficients in $BL_N([0,2\pi])$, and let $f \in BL_N([0,2\pi])$. Assume that for each $\omega = -N/2+1,\dots,N/2$, the recursion coefficients in (3.3) are computed on a $2^K N$-point uniform grid. Then a block KSS method that uses a K-node block Gaussian rule to compute each Fourier component $[\hat{\mathbf{u}}^1]_\omega$ ($\omega = -N/2+1,\dots,N/2$) of the solution to (1.1)-(1.3), satisfies the relations*

$$\left|[\hat{\mathbf{u}}^1]_\omega - \hat{u}(\omega,\Delta t)\right| = O(\Delta t^{2K}), \quad \omega = -N/2+1,\dots,N/2,$$

where $\hat{u}(\omega,\Delta t)$ is the corresponding Fourier component of the exact solution at time Δt.

Proof. The result follows immediately from the substitution of $f(\lambda) = e^{-\lambda\Delta t}$ into the quadrature error (4.1), and the elimination of spatial error from the computation of the recursion coefficients, by refining the grid to the extent necessary to resolve all Fourier components of pointwise products of functions. □

4.2. Stability for the one-node case. When $K=1$, we simply have $\mathcal{T}_1(\omega) = M_1(\omega)$, where

$$M_1(\omega) = \begin{bmatrix} \hat{\mathbf{e}}_\omega^H L_N \hat{\mathbf{e}}_\omega & \hat{\mathbf{e}}_\omega^H L_N \mathbf{u}_\omega^n/\|\mathbf{u}_\omega^n\|_2 \\ [\mathbf{u}_\omega^n]^H L_N \hat{\mathbf{e}}_\omega/\|\mathbf{u}_\omega^n\|_2 & \mathbf{u}_\omega^H L_N \mathbf{u}_\omega^n/\|\mathbf{u}_\omega^n\|_2^2 \end{bmatrix}. \tag{4.2}$$

We now examine the stability of the 1-node method in the case where $p(x) \equiv p = \text{constant}$. We then have

$$\mathcal{T}_1(\omega) = \begin{bmatrix} p\omega^2 + \bar{q} & \hat{\mathbf{e}}_\omega^H(\tilde{\mathbf{q}} \cdot \mathbf{u}_\omega^n)/\|\mathbf{u}_\omega^n\|_2 \\ [\mathbf{u}_\omega^n]^H(\tilde{\mathbf{q}} \cdot \hat{\mathbf{e}}_\omega)/\|\mathbf{u}_\omega^n\|_2 & [\mathbf{u}_\omega^n]^H L_N \mathbf{u}_\omega^n/\|\mathbf{u}_\omega^n\|_2^2 \end{bmatrix}, \tag{4.3}$$

where the notation $(\mathbf{u}\cdot\mathbf{v})$ is used to denote component-wise multiplication of the vectors $\mathbf{u}$ and $\mathbf{v}$. We use the notation $\overline{f}$ to denote the mean of a function $f(x)$ defined on $[0,2\pi]$, and define $\tilde{q}(x) = q(x) - \overline{q}$. We denote by $\tilde{\mathbf{q}}$ the vector with components $[\tilde{\mathbf{q}}]_j = \tilde{q}(x_j)$.

Because $M_1(\omega)$ is Hermitian, we can write

$$M_1(\omega) = U_1(\omega)\Lambda_1(\omega)U_1(\omega)^H.$$

The Fourier component $[\hat{\mathbf{u}}^{n+1}]_\omega$ is then obtained as follows:

$$\begin{aligned}
[\hat{\mathbf{u}}^{n+1}]_\omega &= \left[B_0(\omega)^H \exp(-\mathcal{T}_1(\omega)\Delta t) B_0(\omega)\right]_{12} \\
&= \left[B_0(\omega)^H U_1(\omega) \exp(-\Lambda_1(\omega)\Delta t) U_1(\omega)^H B_0(\omega)\right]_{12} \\
&= \begin{bmatrix} u_{11}(\omega)e^{-\lambda_1(\omega)\Delta t} & u_{12}(\omega)e^{-\lambda_2(\omega)\Delta t} \end{bmatrix} \begin{bmatrix} \overline{u_{11}(\omega)} & \overline{u_{21}(\omega)} \\ \overline{u_{12}(\omega)} & \overline{u_{22}(\omega)} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{e}}_\omega^H \mathbf{u}^n \\ \|\mathbf{u}_\omega^n\|_2 \end{bmatrix} \\
&= [|u_{11}(\omega)|^2 e^{-\lambda_1(\omega)\Delta t} + |u_{12}(\omega)|^2 e^{-\lambda_2(\omega)\Delta t}]\hat{\mathbf{e}}_\omega^H \mathbf{u}^n \\
&\quad + [u_{11}(\omega)\overline{u_{21}(\omega)} e^{-\lambda_1(\omega)\Delta t} + u_{12}(\omega)\overline{u_{22}(\omega)} e^{-\lambda_2(\omega)\Delta t}]\|\mathbf{u}_\omega^n\|_2.
\end{aligned}$$

This simple form of the approximate solution operator yields the following result. For convenience, we denote by $\tilde{S}_N(\Delta t)$ the matrix such that $\mathbf{u}^{n+1} = \tilde{S}_N(\Delta t)\mathbf{u}^n$, for given N and Δt, and write $\tilde{S}_N(\Delta t)^n$ in place of $\left[\tilde{S}_N(\Delta t)\right]^n$.

THEOREM 4.2. *Let $q(x)$ in (1.4) belong to $BL_M([0,2\pi])$ for a fixed integer M. Then, for the problem (1.1)-(1.3), the block KSS method with $K=1$ is unconditionally stable. That is, given $T>0$, there exists a constant C_T, independent of N and Δt, such that*

$$\|\tilde{S}_N(\Delta t)^n\| \leq C_T, \tag{4.4}$$

for $0 \leq n\Delta t \leq T$.

Proof. Because $\lambda_i(\omega) > \lambda_{\text{min}}(L_N) > 0$, and the rows and columns of $U_1(\omega)$ have unit 2-norm, it follows that

$$\left[|u_{11}(\omega)|^2 e^{-\lambda_1(\omega)\Delta t} + |u_{12}(\omega)|^2 e^{-\lambda_2(\omega)\Delta t}\right] \leq e^{-\lambda_{\text{min}}(L_N)\Delta t}.$$

We now consider the remaining portion of each Fourier component,

$$\left[u_{11}(\omega)\overline{u_{21}(\omega)}e^{-\lambda_1(\omega)\Delta t} + u_{12}(\omega)\overline{u_{22}(\omega)}e^{-\lambda_2(\omega)\Delta t}\right]\|\mathbf{u}_\omega^n\|_2.$$

By the orthogonality of the rows of $U_1(\omega)$, we can rewrite this as

$$u_{11}(\omega)\overline{u_{21}(\omega)}\left[e^{-\lambda_1(\omega)\Delta t} - e^{-\lambda_2(\omega)\Delta t}\right]\|\mathbf{u}_\omega^n\|_2.$$

By direct computation of the elements of $U_1(\omega)$, whose columns are the eigenvectors of $\mathcal{T}_1(\omega)$, we obtain

$$u_{11}(\omega)\overline{u_{21}(\omega)} = -\frac{\hat{\mathbf{e}}_\omega^H(\tilde{\mathbf{q}}\cdot\mathbf{u}_\omega^n)}{\|\mathbf{u}_\omega^n\|_2\sqrt{(p\omega^2 + \bar{q} - [\mathbf{u}_\omega^n]^H L_N\mathbf{u}_\omega^n/\|\mathbf{u}_\omega^n\|_2^2)^2 + 4(\hat{\mathbf{e}}_\omega^H L_N\mathbf{u}_\omega^n)^2/\|\mathbf{u}_\omega^n\|_2^2}}.$$

Furthermore, for each integer ω and $\Delta t \geq 0$, we have

$$|e^{-\lambda_1(\omega)\Delta t} - e^{-\lambda_2(\omega)\Delta t}| \leq \Delta t|\lambda_2(\omega) - \lambda_1(\omega)|.$$

To see this, note that at $\Delta t = 0$, $g_\omega(\Delta t) = e^{-\lambda_1(\omega)\Delta t} - e^{-\lambda_2(\omega)\Delta t}$ has slope $g_\omega'(0) = \lambda_2(\omega) - \lambda_1(\omega)$. However, its slope becomes less steep as Δt increases from 0, because its first and second derivatives at $\Delta t = 0$ are of opposite sign. Furthermore, $g_\omega(\Delta t)$ has only one critical number and one inflection point, it approaches 0 as $\Delta t \to \infty$, and it is equal to zero at $\Delta t = 0$. It follows that for each integer ω,

$$\left|u_{11}(\omega)\overline{u_{21}(\omega)}\left[e^{-\lambda_1(\omega)\Delta t} - e^{-\lambda_2(\omega)\Delta t}\right]\|\mathbf{u}_\omega^n\|_2\right| \leq C\Delta t|\hat{\mathbf{e}}_\omega(\tilde{\mathbf{q}}\cdot\mathbf{u}^n)|,$$

where

$$C = \sup_{\omega\in\mathbb{Z}}\frac{\lambda_1(\omega) - \lambda_2(\omega)}{\sqrt{(p\omega^2 + \bar{q} - [\mathbf{u}_\omega^n]^H L_N\mathbf{u}_\omega^n/\|\mathbf{u}_\omega^n\|_2^2)^2 + 4(\hat{\mathbf{e}}_\omega^H L_N\mathbf{u}_\omega^n)^2/\|\mathbf{u}_\omega^n\|_2^2}},$$

the least upper bound of a sequence that converges to 1 as $|\omega| \to \infty$. We conclude that

$$\left\|\tilde{S}_N(\Delta t)\right\|_2 \leq e^{(-\lambda_{\text{min}}(L) + C\|\tilde{q}\|_\infty)\Delta t},$$

from which the result follows. □

ETNA
Kent State University
http://etna.math.kent.edu

We can now prove that the method converges. For convenience, we define the 2-norm of a function $u(x,t)$ to be the vector 2-norm of the restriction of $u(x,t)$ to the spatial grid:

$$\|u(\cdot,t)\|_2 = \left(\sum_{j=0}^{N-1} |u(j\Delta x, t)|^2 \right)^{1/2}. \tag{4.5}$$

We also say that a method is convergent of order (m,n) if there exist constants C_t and C_x, independent of the time step Δt and grid spacing $\Delta x = 2\pi/N$, such that

$$\|u(\cdot,t) - \mathbf{u}(\cdot,t)\|_2 \leq C_t \Delta t^m + C_x \Delta x^n, \quad 0 \leq t \leq T. \tag{4.6}$$

THEOREM 4.3. *Let the exact solution $u(x,t)$ of the problem (1.1)-(1.3) belong to $C^p([0,2\pi])$ for each t in $[0,T]$. Let $q(x)$ in (1.4) belong to $BL_M([0,2\pi])$ for some integer M. Then, the 1-node block KSS method, applied to this problem, is convergent of order $(1,p)$.*

Proof. Let $S(\Delta t)$ be the solution operator for the problem (1.1)-(1.3). As with $\tilde{S}_N(\Delta t)$, we use the notation $S(\Delta t)^n$ in place of $[S(\Delta t)]^n$ for simplicity. For any nonnegative integer n and fixed grid size N, we define

$$E_n = N^{-1/2} \|S(\Delta t)^n f - \tilde{S}_N(\Delta t)^n f\|_2. \tag{4.7}$$

Then, there exist constants C_1, C_2 and C such that

$$\begin{aligned}
E_{n+1} &= N^{-1/2} \|S(\Delta t)^{n+1} f - \tilde{S}_N(\Delta t)^{n+1} f\|_2 \\
&= N^{-1/2} \|S(\Delta t) S(\Delta t)^n f - \tilde{S}_N(\Delta t) \tilde{S}_N(\Delta t)^n f\|_2 \\
&= N^{-1/2} \|S(\Delta t) S(\Delta t)^n f - \tilde{S}_N(\Delta t) S(\Delta t)^n f \\
&\quad + \tilde{S}_N(\Delta t) S(\Delta t)^n f - \tilde{S}_N(\Delta t) \tilde{S}_N(\Delta t)^n f\|_2 \\
&\leq N^{-1/2} \|S(\Delta t) S(\Delta t)^n f - \tilde{S}_N(\Delta t) S(\Delta t)^n f\| \\
&\quad + N^{-1/2} \|\tilde{S}_N(\Delta t) S(\Delta t)^n f - \tilde{S}_N(\Delta t) \tilde{S}_N(\Delta t)^n f\|_2 \\
&\leq N^{-1/2} \|S(\Delta t) u(t_n) - \tilde{S}_N(\Delta t) u(t_n)\|_2 + \|\tilde{S}_N(\Delta t)\|_2 E_n \\
&\leq C_1 \Delta t^2 + C_2 \Delta t \Delta x^p + e^{C\Delta t} E_n,
\end{aligned}$$

where the spatial error arises from the truncation of the Fourier series of the exact solution. It follows that

$$E_n \leq \frac{e^{CT} - 1}{e^{C\Delta t} - 1} (C_1 \Delta t^2 + C_2 \Delta t \Delta x^p) \leq \tilde{C}_1 \Delta t + \tilde{C}_2 \Delta x^p, \tag{4.8}$$

for constants $\tilde{C}_1$ and $\tilde{C}_2$ that depend only on T. □

It is important to note that although stability and convergence were only shown for the case where the leading coefficient $p(x)$ is constant, it has been demonstrated that KSS methods exhibit similar stability on more general problems, such as in [13] where it was applied to a second-order wave equation with time steps that greatly exceeded the CFL limit. Furthermore, [13] also introduced homogenizing similarity transformations that can be used to extend the applicability of theoretical results concerning stability that were presented in that paper, as well as the one given here.

4.3. Stability for the multi-node case. For the case $K > 1$, we have

$$[\hat{\mathbf{u}}^{n+1}]_\omega = \hat{\mathbf{e}}_\omega^H \mathbf{u}^n \sum_{j=1}^{2K} |u_{1j}(\omega)|^2 e^{-\lambda_j(\omega)\Delta t} + \|\mathbf{u}_\omega^n\|_2 \sum_{j=1}^{2K} u_{1j}(\omega) \overline{u_{2j}(\omega)} e^{-\lambda_j(\omega)\Delta t}.$$

ETNA
Kent State University
http://etna.math.kent.edu

The first summation is bounded by $e^{-\lambda_1(L)\Delta t}$. The second summation is a K-node block Gaussian quadrature approximation of $\hat{\mathbf{e}}_\omega^H \exp(-L_N\Delta t)\mathbf{u}_\omega^n$, which is given by

$$\|\mathbf{u}_\omega^n\|_2[\exp(-\mathcal{T}_1(\omega)\Delta t)]_{12}.$$

Its Taylor expansion begins as follows:

$$\|\mathbf{u}_\omega^n\|_2 \sum_{j=1}^{2K} u_{1j}(\omega)\overline{u_{2j}(\omega)}e^{-\lambda_j(\omega)\Delta t} = -\Delta t\, \hat{\mathbf{e}}_\omega^H(\tilde{\mathbf{q}}\cdot \mathbf{u}_\omega^n) + \frac{1}{2}\Delta t^2 \hat{\mathbf{e}}_\omega^H L_N^2 \mathbf{u}_\omega^n + O(\Delta t^3).$$

Because

$$\hat{\mathbf{e}}_\omega^H L_N^2 \mathbf{u}_\omega^n = 2p\omega^2 \hat{\mathbf{e}}_\omega^H(\tilde{\mathbf{q}}\cdot \mathbf{u}_\omega^n) + \text{lower-order terms},$$

which is a negative scalar multiple of the first-order term in the Taylor series of

$$\hat{\mathbf{e}}_\omega^H \exp(-L_N\Delta t)\mathbf{u}_\omega^n,$$

it follows that for ω sufficiently large and Δt sufficiently small,

$$\left| \|\mathbf{u}_\omega^n\|_2 \sum_{j=1}^{2K} u_{1j}(\omega)\overline{u_{2j}(\omega)}e^{-\lambda_j(\omega)\Delta t} \right| \leq \Delta t|\hat{\mathbf{e}}_\omega^H(\tilde{\mathbf{q}}\cdot \mathbf{u}_\omega^n)| = \Delta t|\hat{\mathbf{e}}_\omega^H(\tilde{\mathbf{q}}\cdot \mathbf{u}^n)|. \tag{4.9}$$

However, this is not sufficient to conclude that unconditional stability is achieved as in the 1-node case. Future work will continue this analysis, but Figure 4.1 offers evidence of such unconditional stability, in the form of a "proof by MATLAB". For several values of ω, we plot the function

$$g_\omega(\Delta t) = \left| \frac{\|\mathbf{u}_\omega^n\|_2[\exp(-\mathcal{T}_1(\omega)\Delta t)]_{12}}{\hat{\mathbf{e}}_\omega^H(\tilde{\mathbf{q}}\cdot \mathbf{u}^n)} \right| \tag{4.10}$$

and observe that $g_\omega(\Delta t) \leq \Delta t$, that is, (4.9) holds, not just for sufficiently small Δx and Δt, but for *all* $\Delta x \in (0, 2\pi]$ and $\Delta t \in (0, T]$. This experiment was performed on the operator

$$Lu = -u_{xx} + E^+ f_{0,1}(x)u,$$

where the coefficient $E^+ f_{0,1}(x)$ is defined below in Section 5.1, in such a way as to have the smoothness of a function that is only once continuously differentiable.

5. Numerical results. In this section, we will present numerical results to compare the original KSS method (as described in [13]) to the new block KSS method, when applied to parabolic problems. The comparisons will focus on the accuracy of the temporal approximations employed by each method, for these reasons:

- parabolic problems tend to yield smooth solutions, so a Fourier interpolant defined on a grid of reasonable size is sufficient to accurately represent the solution;
- is not straightforward to quantify the effect of the spatial discretization error on the accuracy of Krylov subspace spectral methods, as it is two-fold. Unless grid refinement is used during the Lanczos process, such error affects the recursion coefficients, which in turn affect the quadrature nodes and weights, and therefore the accuracy of the temporal approximation of each component. The truncation of the Fourier series introduces additional spatial error, depending on the smoothness of the solution. A thorough analysis of the spatial discretization error will be deferred for future work.

For convenience, we denote by KSS(K) the original KSS method with K Gaussian nodes, and by KSS-B(K) the block KSS method with K block Gaussian nodes.

ETNA
Kent State University
http://etna.math.kent.edu

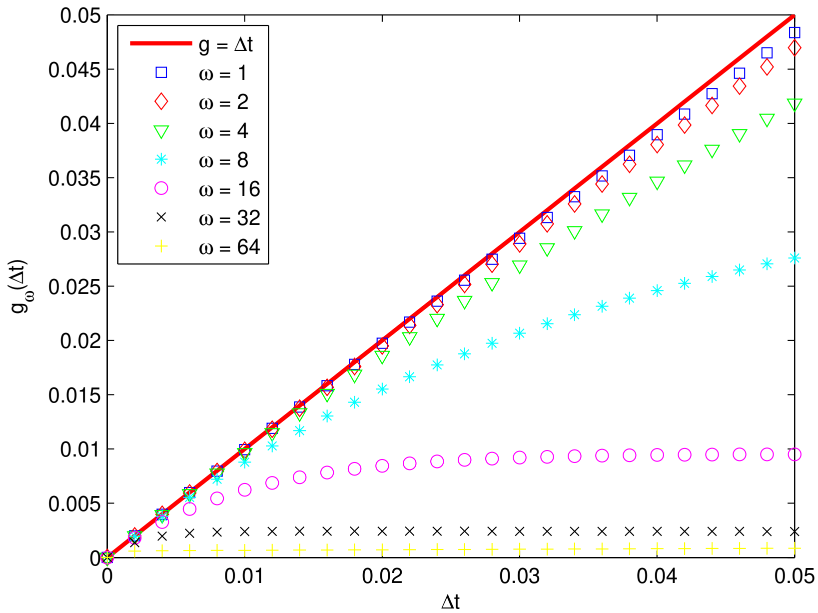

FIGURE 4.1. *Experimental verification that (4.9) holds for selected values of Δt and ω. The function $g_\omega(\Delta t)$ is defined in (4.10). The solid red line is the line $g = \Delta t$.*

5.1. Construction of test cases. We introduce some differential operators and functions that will be used in the experiments described in this section. As most of these functions and operators are randomly generated, we will denote by $R_1, R_2, \dots$ the sequence of random numbers obtained using MATLAB's random number generator `rand` after setting the generator to its initial state. These numbers are uniformly distributed on the interval $(0, 1)$.

We will make frequent use of a two-parameter family of functions, defined on the interval $[0, 2\pi]$. First, we define

$$f^0_{j,k}(x) = \mathrm{Re}\left\{ \sum_{\substack{|\omega|<N/2 \\ \omega\neq 0}} \hat{f}_j(\omega)(1+|\omega|)^{-(k+1)} e^{i\omega x} \right\}, \quad j,k = 0,1,\dots, \tag{5.1}$$

where

$$\hat{f}_j(\omega) = R_{jN+2(\omega+N/2)-1} + iR_{jN+2(\omega+N/2)}. \tag{5.2}$$

The parameter j indicates how many functions have been generated in this fashion since setting MATLAB's random number generator to its initial state, and the parameter k indicates how smooth the function is. Figure 5.1 shows selected functions from this collection.

In many cases, it is necessary to ensure that a function is positive or negative, so we define the translation operators E^+ and E^- by

$$E^+ f(x) = f(x) - \min_{x\in[0,2\pi]} f(x) + 1,$$

$$E^- f(x) = f(x) - \max_{x\in[0,2\pi]} f(x) - 1.$$

ETNA
Kent State University
http://etna.math.kent.edu

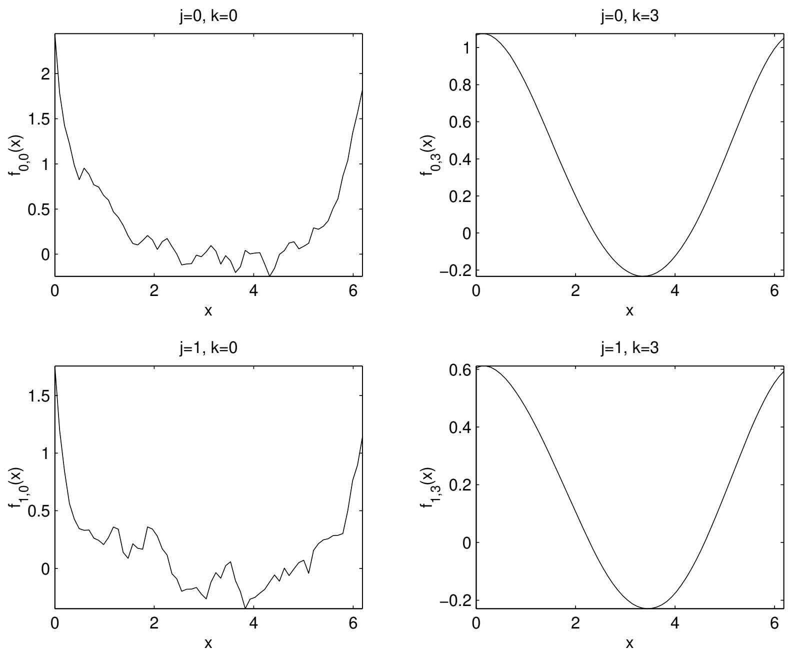

FIGURE 5.1. *Functions from the collection $f_{j,k}(x)$, for selected values of j and k.*

We define a similar two-parameter family of functions defined on the rectangle $[0, 2\pi] \times [0, 2\pi]$:

$$g_{j,k}(x, y) = \mathrm{Re}\left\{ \sum_{\substack{|\omega|,|\xi|<N/2 \\ \omega\xi\neq 0}} \hat{g}_j(\omega, \xi)(1 + |\omega|)^{-(k+1)}(1 + |\xi|)^{-(k+1)} e^{i(\omega x+\xi y)} \right\}, \tag{5.3}$$

where j and k are nonnegative integers, and

$$\begin{aligned} \hat{g}_j(\omega, \xi) = {} & R_{jN^2+2[N(\omega+N/2-1)+(\xi+N/2)]-1} \\ & + iR_{jN^2+2[N(\omega+N/2-1)+(\xi+N/2)]}. \end{aligned} \tag{5.4}$$

Figure 5.2 shows selected functions from this collection.

In all experiments, the solutions $u^{(j)}(x, t)$ are computed using time steps $\Delta t = 2^{-j}$, for $j = 0, \ldots, 5$. Unless otherwise noted, the error estimates are obtained by computing

$$\frac{\|u^{(j)}(\cdot, 1) - u^{(5)}(\cdot, 1)\|}{\|u^{(5)}(\cdot, 1)\|}.$$

This method of estimating the error assumes that $u^{(5)}(x, t)$ is a sufficiently accurate approximation to the exact solution. This assumption has been numerically verified, by comparing $u^{(5)}$ against approximate solutions computed using established methods, and by comparing $u^{(5)}$ against solutions obtained using various methods with smaller time steps.

In [18] and in Section 5.3 of this paper, errors measured by comparison to exact solutions of problems with source terms, further validate the convergence behavior. It should be noted that we are not seeking a sharp estimate of the error, but rather an indication of the rate of convergence, and for this goal it is sufficient to use $u^{(5)}$ as an approximation to the exact solution.

ETNA
Kent State University
http://etna.math.kent.edu

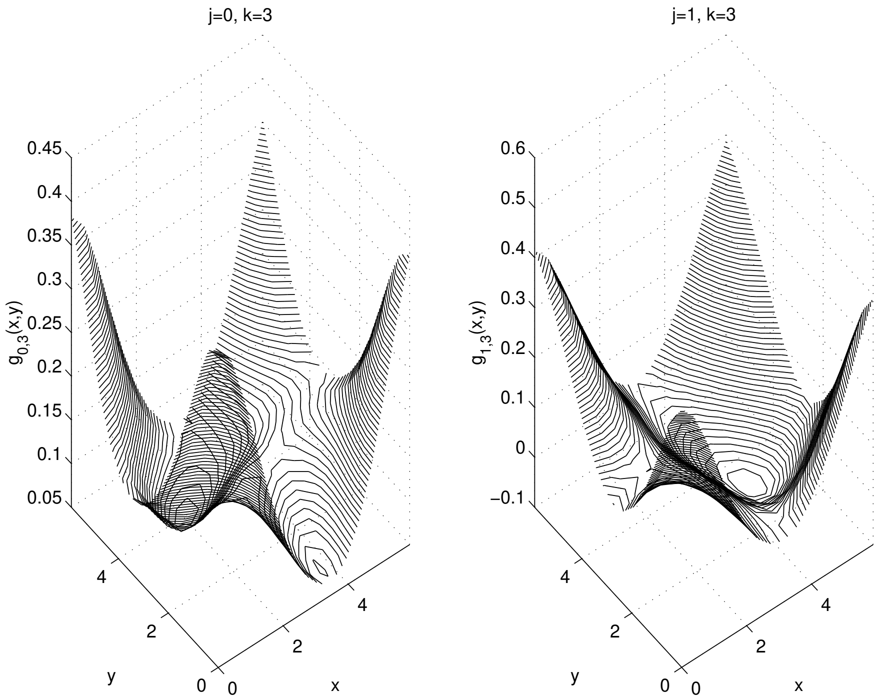

FIGURE 5.2. *Functions from the collection $g_{j,k}(x, y)$, for selected values of j and k.*

5.2. Constant leading coefficient. We now show the accuracy of our approach for higher space dimensions. We solve a variable-coefficient heat equation with randomly generated coefficients as discussed in Section 5.1.

First, we solve the parabolic problems

$$\begin{aligned} \frac{\partial u}{\partial t}(x,t) - \frac{\partial^2 u}{\partial x^2}(x,t) - E^- f_{1,2}(x)u(x,t) &= 0, \qquad x \in (0, 2\pi), \quad t > 0, \\ u(x,0) &= E^+ f_{0,3}(x), \qquad 0 < x < 2\pi, \\ u(x,t) &= u(x+2\pi,t), \qquad t > 0, \end{aligned} \tag{5.5}$$

and

$$\begin{aligned} \frac{\partial u}{\partial t}(x,y,t) - \Delta u(x,y,t) - E^- g_{3,2}(x,y)u(x,t) &= 0, \qquad (x,y) \in (0,2\pi)^2, \\ & \qquad\qquad t > 0, \\ u(x,y,0) &= E^+ g_{3,3}(x,y), \qquad (x,y) \in (0,2\pi)^2, \\ u(x,y,t) = u(x+2\pi,t) &= u(x,y+2\pi,t), \qquad t > 0, \end{aligned} \tag{5.6}$$

In [18], it is shown that the methods for efficiently computing the recursion coefficients generalizes in a straightforward manner to higher spatial dimensions. The results are shown in Figures 5.3 and 5.4, and Tables 5.1 and 5.2, and compared to those obtained using the original KSS method. In the 2-D case, the variable coefficient of the PDE is smoothed to a greater extent than in the 1-D case, because the prescribed decay rate of the Fourier coefficients is imposed in both the x- and y-directions. This results in greater accuracy in the 2-D case, which is consistent with the result proved in [17] that the local truncation error varies linearly with

ETNA
Kent State University
http://etna.math.kent.edu

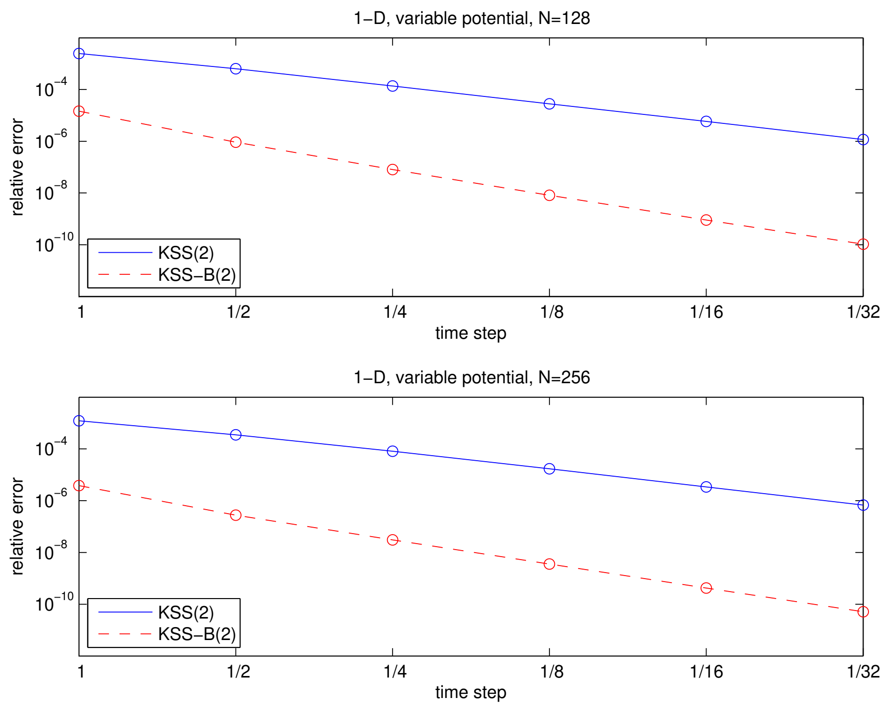

FIGURE 5.3. *(a) Top plot: estimates of relative error in the approximate solution of (5.5) at $T = 1$. Solutions are computed using the original 2-node KSS method (solid curve), and a 2-node block KSS method (dashed curve), both with $N = 128$ grid points. (b) Bottom plot: estimates of relative error in the approximate solution of the same problem, using the same methods, with $N = 256$ grid points. In both cases, the methods use time steps $\Delta t = 2^{-j}$, $j = 0, \ldots, 5$.*

the variation in the coefficients. We see that significantly greater accuracy, and temporal order of accuracy, is obtained with block KSS methods, in both one and two space dimensions.

In an attempt to understand why the block KSS method is significantly more accurate, we examine the approximate solution operator for the simple case of $K = 1$. As shown in [13], the original 1-node KSS method is equivalent to the simple splitting

$$\tilde{S}_N(\Delta t) = e^{-C\Delta t}(I - \Delta t V), \tag{5.7}$$

where $L = C + V$ is a splitting of the differential operator L, in which C is obtained by averaging the coefficients of L. On the other hand, the 1-node block KSS method is not equivalent to such a splitting, because every node and weight of the quadrature rule used to compute each Fourier component is influenced by the solution from the previous time step.

Furthermore, an examination of the nodes for both methods reveals that, for the original KSS method, all of the nodes used to compute $[\hat{\mathbf{u}}^{n+1}]_\omega$ tend to be clustered around $\hat{\mathbf{e}}_\omega^H L_N \hat{\mathbf{e}}_\omega$, whereas with the block KSS method half of the nodes are clustered near this value, and the other half are clustered near $[\mathbf{u}_\omega^n]^H L_N \mathbf{u}_\omega^n$, so the previous solution plays a much greater role in the construction of the quadrature rules. A similar effect was achieved with the original KSS method by using a Gauss-Radau rule in which the prescribed node was an approximation of the smallest eigenvalue of L_N, and while this significantly improved accuracy for parabolic problems, as shown in [17], the solution-dependent approach used by the block method makes more sense, especially if the initial data happens to be oscillatory.

ETNA
Kent State University
http://etna.math.kent.edu

TABLE 5.1

Estimates of relative error and temporal order of convergence in the approximate solution of problem (5.5) at $T = 1$, using 2-node original and block Krylov subspace spectral methods. Error is the relative difference, in the 2-norm, between approximate solutions and a solution computed using a smaller time step, since no exact solution is available. N denotes the number of grid points and Δt denotes the time step used.

Method	N	Δt	Error	Order
KSS(2)	128	1	0.0025	2.21
		1/2	0.00064	
		1/4	0.00014	
		1/8	2.8e-005	
		1/16	5.8e-006	
		1/32	1.2e-006	
	256	1	0.0012	2.16
		1/2	0.00035	
		1/4	8.2e-005	
		1/8	1.7e-005	
		1/16	3.4e-006	
		1/32	6.8e-007	
KSS-B(2)	128	1	1.5e-005	3.41
		1/2	9.3e-007	
		1/4	8e-008	
		1/8	8.1e-009	
		1/16	9e-010	
		1/32	1.1e-010	
	256	1	3.8e-006	3.24
		1/2	2.8e-007	
		1/4	3e-008	
		1/8	3.6e-009	
		1/16	4.2e-010	
		1/32	5.1e-011	

5.3. Variable leading coefficient. We now apply a 2-node block Krylov subspace spectral method to the problem

$$u_t = (a(x)u_x)_x + F(x,t), \quad x \in (0, 2\pi), \quad t > 0, \tag{5.8}$$

where

$$F(x,t) = \sin(x-t) + (a(x)\sin(x-t))_x. \tag{5.9}$$

With periodic boundary conditions and the initial condition

$$u(x,0) = \cos x, \tag{5.10}$$

the exact solution is $\cos(x-t)$. For $a(x)$, we use a smooth function $a_s(x) = E^+ f^0_{0,3}(x)$, as defined in (5.1). In [18], it was shown that the original KSS method compared favorably to the standard ODE solvers provided in MATLAB in terms of accuracy and efficiency. We now compare that method to our new block approach.

Table 5.3 lists the relative errors for various time steps and grid sizes. The errors are obtained by comparing the approximate solution to the known exact solution in the 2-norm. While the temporal order of accuracy and relative errors are comparable, it should be noted

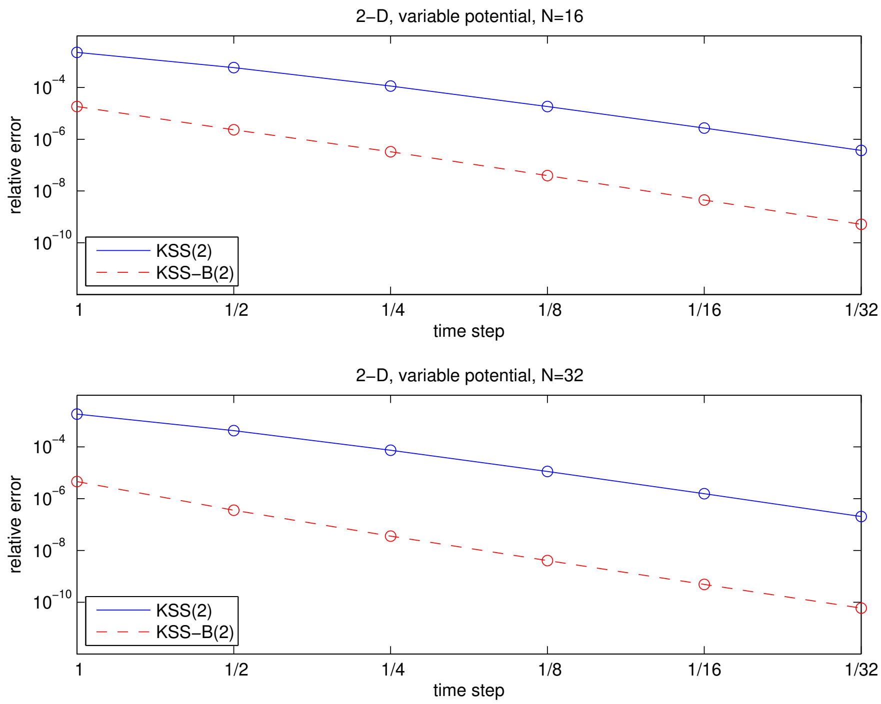

FIGURE 5.4. *(a) Top plot: estimates of relative error in the approximate solution of (5.6) at $T = 1$. Solutions are computed using the original 2-node KSS method (solid curve), and a 2-node block KSS method (dashed curve), both with $N = 16$ grid points per dimension. (b) Bottom plot: estimates of relative error in the approximate solution of the same problem, using the same methods, with $N = 32$ grid points per dimension. In both cases, the methods use time steps $\Delta t = 2^{-j}$, $j = 0, \dots, 5$.*

that for the block method, not only are the errors smaller, but at larger time steps they are much less sensitive to the increase in the number of grid points than the original KSS method, which loses some accuracy on the finer grid. The error estimates are also plotted in Figure 5.5.

5.4. Systems of coupled PDE. In [14], KSS methods were generalized to apply to systems of coupled PDE by choosing appropriate basis functions. The same basis functions can readily be employed by the block approach. We therefore apply the original and block 2-node KSS methods to the hyperbolic system

$$\begin{aligned}
\frac{\partial u}{\partial t} &= f_{0,3}^{+}(x)\frac{\partial v}{\partial x} + f_{1,3}^{-}(x)v(x,t), && x \in (0,2\pi), \quad t > 0,\\
\frac{\partial v}{\partial t} &= f_{2,3}^{+}(x)\frac{\partial u}{\partial x} + f_{3,3}^{-}(x)u(x,t), && x \in (0,2\pi), \quad t > 0,\\
u(x,0) &= f_{4,3}^{+}(x), && x \in (0,2\pi),\\
v(x,0) &= f_{5,3}^{+}(x), && x \in (0,2\pi),\\
u(x,t) &= u(x+2\pi,t), && t > 0,\\
v(x,t) &= v(x+2\pi,t), && t > 0.
\end{aligned} \tag{5.11}$$

Figure 5.6 and Table 5.4 report the results of applying the original and block 2-node KSS methods to this problem. We observe that both methods achieve the expected temporal order of convergence, but once again block KSS methods are significantly more accurate.

ETNA
Kent State University
http://etna.math.kent.edu

TABLE 5.2

Estimates of relative error and temporal order of convergence in the approximate solution of (5.6) at $T = 1$, using 2-node original and block Krylov subspace spectral methods. Error is the relative difference, in the 2-norm, between approximate solutions and a solution computed using a smaller time step. N denotes the number of grid points per dimension, and Δt denotes the time step used.

Method	N	Δt	Error	Order
KSS(2)	16	1	0.0023	2.52
		1/2	0.00059	
		1/4	0.00011	
		1/8	1.8e-005	
		1/16	2.7e-006	
		1/32	3.7e-007	
	32	1	0.0018	2.63
		1/2	0.00042	
		1/4	7.5e-005	
		1/8	1.1e-005	
		1/16	1.6e-006	
		1/32	2.1e-007	
KSS-B(2)	16	1	1.8e-005	3.02
		1/2	2.3e-006	
		1/4	3.3e-007	
		1/8	4e-008	
		1/16	4.5e-009	
		1/32	5.2e-010	
	32	1	4.6e-006	3.25
		1/2	3.6e-007	
		1/4	3.6e-008	
		1/8	4.1e-009	
		1/16	4.9e-010	
		1/32	5.9e-011	

Although it is straightforward to apply the block approach to systems of PDE as well as scalar equations, the essential tasks of efficiently implementing the block algorithm, and analyzing its stability for hyperbolic systems, are less straightforward. Furthermore, the spatial discretization of such a system yields a system of ODEs of the form $\mathbf{u}_t = A\mathbf{u}$ where A is not necessarily symmetric, and while KSS methods have been successfully applied to such systems (see [15]), it is necessary to investigate the limitations of this applicability, as the unsymmetric Lanczos algorithm is much more susceptible to premature breakdown than its symmetric counterpart.

5.5. The matrix exponential. We now consider the problem of computing $\mathbf{w} = e^{-A}\mathbf{v}$ for a given symmetric positive definite matrix A and vector $\mathbf{v}$. One approach, described in [12], is to apply the Lanczos algorithm to A with initial vector $\mathbf{v}$ to obtain, at the end of the jth iteration, an orthogonal matrix X_j and a tridiagonal matrix T_j such that $X_j^T A X_j = T_j$. Then, we can compute the approximation

$$\mathbf{w}_j = e^{-A}\mathbf{v} \approx \|\mathbf{v}\|_2 X_j e^{-T_j}\mathbf{e}_1. \tag{5.12}$$

However, the effectiveness of this approach, for general $\mathbf{v}$, depends on the eigenvalues of A: if the eigenvalues are not clustered, which is the case if A arises from a stiff system of ODEs,

ETNA
Kent State University
http://etna.math.kent.edu

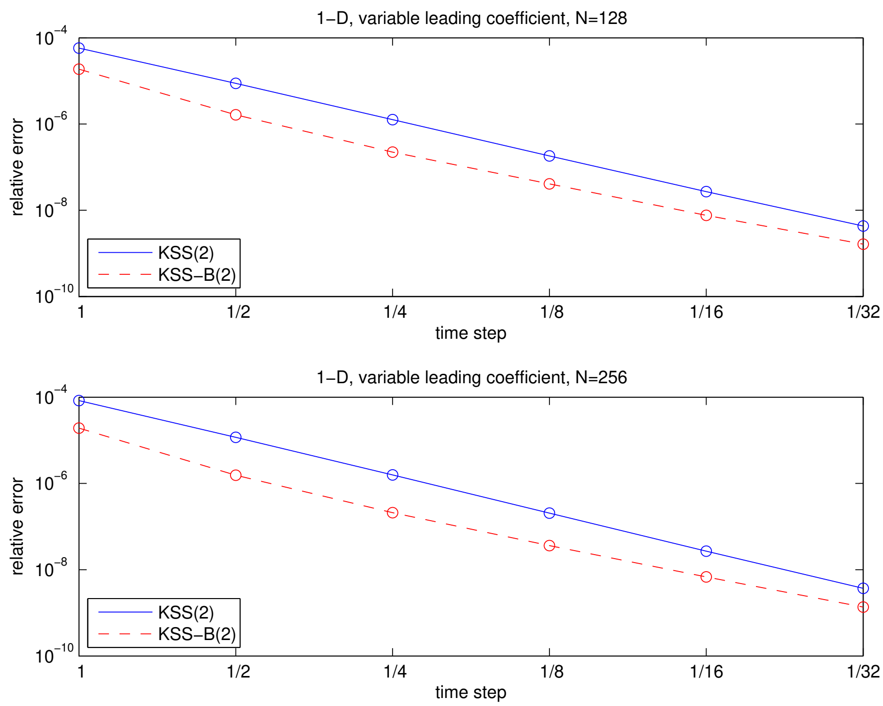

FIGURE 5.5. *(a) Top plot: estimates of relative error in the approximate solution of (5.8), (5.10), (1.3), at $T = 1$. Solutions are computed using the original 2-node KSS method (solid curve), and a 2-node block KSS method (dashed curve). Both methods use $N = 128$ grid points. (b) Bottom plot: estimates of relative error in the approximate solution of the same problem at $T = 1$. Solutions are computed using the same methods, with $N = 256$ grid points. In both cases, the methods use time steps $\Delta t = 2^{-j}$, $j = 0, \ldots, 5$.*

a good approximation cannot be obtained using a small number of iterations. This can be alleviated using an outer iteration of the form

$$\mathbf{w}_j^{m+1} \approx e^{-A\Delta t}\mathbf{w}_j^m, \quad m = 0, 1, \ldots, \quad \mathbf{w}_j^0 = \mathbf{v}, \tag{5.13}$$

for some $\Delta t \ll 1$, where the total number of outer iterations M satisfies $M\Delta t = 1$. However, this approach is not practical if Δt must be chosen very small.

An alternative is to use the approach employed by KSS methods, computing each component of $\mathbf{w}$ using its own approximation to the matrix exponential. Specifically, for the original KSS method, we apply the Lanczos algorithm to A with initial vectors $\mathbf{e}_j$ and $\mathbf{e}_j + \delta\mathbf{v}$, for some small constant δ, where $\mathbf{e}_j$ is the jth standard basis vector. For the block KSS method, we apply the block Lanczos algorithm to A with initial blocks $R_0 = \begin{bmatrix} \mathbf{e}_j & \mathbf{v} \end{bmatrix}$. We then use block Gaussian quadrature as described in Section 3.1.

Figure 5.7 and Table 5.5 describe the results of applying all three methods in the case where A is a $N \times N$ symmetric positive definite matrix, with $N = 64$ and $N = 128$. The elements of A are given by

$$a_{ij} = -\rho^{|i-j|} b_{ij}, \tag{5.14}$$

where $\rho = 0.2$, the matrix B is defined by $B = C^T C$, and the c_{ij} are random numbers uniformly distributed between 0 and 1, so that A is, with high probability, diagonally dominant. We use time steps $\Delta t = 2^{-j}$, for $j = 1, 2, \ldots, 6$, and compare our approximations to the

ETNA
Kent State University
http://etna.math.kent.edu

TABLE 5.3

Estimates of relative error and temporal order of convergence in the approximate solution of (5.8), (5.10), (1.3), using 2-node original and block Krylov subspace spectral methods. Error is the relative difference, in the 2-norm, between the exact solution $u(x,t) = \cos(x-t)$ and the computed solution at $T = 1$. N denotes the number of grid points and Δt denotes the time step used.

Method	N	Δt	Error	Order
KSS(2)	128	1	5.8e-005	2.74
		1/2	8.7e-006	
		1/4	1.3e-006	
		1/8	1.8e-007	
		1/16	2.7e-008	
		1/32	4.3e-009	
	256	1	8.3e-005	2.89
		1/2	1.2e-005	
		1/4	1.6e-006	
		1/8	2e-007	
		1/16	2.7e-008	
		1/32	3.7e-009	
KSS-B(2)	128	1	1.9e-005	2.7
		1/2	1.6e-006	
		1/4	2.2e-007	
		1/8	4.1e-008	
		1/16	7.6e-009	
		1/32	1.6e-009	
	256	1	1.9e-005	2.76
		1/2	1.5e-006	
		1/4	2.1e-007	
		1/8	3.6e-008	
		1/16	6.8e-009	
		1/32	1.4e-009	

vector $\mathbf{w}$ obtained by using the MATLAB function `expm`. We see not only that the block KSS method is significantly more accurate than both the original KSS method and the Lanczos approximation given by (5.12), but it achieves the higher order of convergence, and the original KSS method does not even converge at all, in fact diverging for the the larger matrix.

6. Discussion. In this concluding section, we consider various generalizations of the problems and methods considered in this paper.

6.1. Higher space dimension. In [18], it is demonstrated how to compute the recursion coefficients α_j and β_j for operators of the form $Lu = -p\Delta u + q(x,y)u$, and the expressions are straightforward generalizations of those given in Section 4.3 for the one-dimensional case. It is therefore reasonable to suggest that, for operators of this form, the consistency and stability results given here for the one-dimensional case generalize to higher dimensions. This will be investigated in the near future.

6.2. Discontinuous coefficients. As shown in [18], discontinuous coefficients reduce the accuracy of KSS methods, because they introduce significant spatial discretization error into the computation of recursion coefficients. Furthermore, for the stability result reported in this paper, the assumption that the coefficients are bandlimited is crucial. Regardless, this result does not apply to problems in which the coefficients are particularly rough or

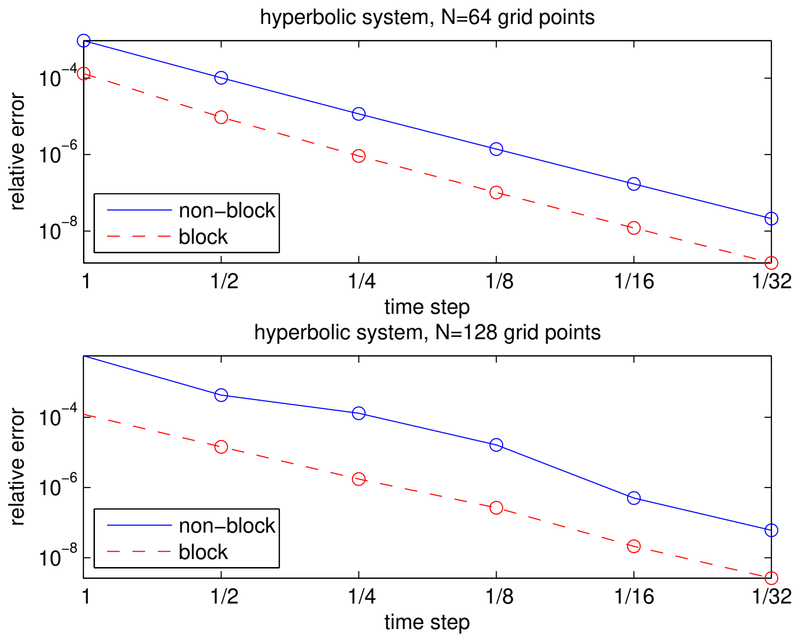

FIGURE 5.6. *(a) Top plot: estimates of relative error in the approximate solution of (5.11) at $T = 1$. Solutions are computed using the original 2-node KSS method (solid curve), and a 2-node block KSS method (dashed curve). Both methods use $N = 64$ grid points for each unknown. (b) Bottom plot: estimates of relative error in the approximate solution of the same problem at $T = 1$. Solutions are computed using the same methods, with $N = 128$ grid points for each unknown. In both cases, the methods use time steps $\Delta t = 2^{-j}$, $j = 0, \ldots, 5$.*

discontinuous, because Gibbs' phenomenon prevents their discrete Fourier transforms from being uniformly bounded for all N. Ongoing work, described in [16], involves the use of the polar decomposition (3.1) to alleviate difficulties caused by such coefficients.

6.3. The wave equation. In [11], KSS methods were applied to the second-order wave equation with Dirichlet boundary conditions, thus demonstrating that they are applicable to "true" IBVP, as opposed to the problems discussed in this paper that include periodic boundary conditions. It was shown that each Fourier component of the solution is computed with $O(\Delta t^{4K})$ accuracy. The block KSS methods presented in this paper can easily be applied to the wave equation, as only the integrands differ, not the quadrature rules. In [13], it was shown that the 1-node KSS method for the wave equation, which is 3rd-order accurate in time, is unconditionally stable when the leading coefficient is constant. Stability analysis for the block KSS methods applied to this problem still remains to be carried out.

6.4. Summary. We have proved that for parabolic variable-coefficient PDE, block KSS methods are capable of computing Fourier components of the solution with greater accuracy than the original KSS methods, and they possess similar stability properties. By pairing the solution from the previous time step with each trial function in a block and applying the Lanczos algorithm to them together, we obtain a block Gaussian quadrature rule that is better suited to approximating a bilinear form involving both functions, than the approach of perturbing Krylov subspaces in the direction of the solution. While the latter approach has already shown much promise for the problem of computing the product of a function of a matrix and a vector by employing componentwise approximations, we see that the new

TABLE 5.4

Estimates of relative error and temporal order of convergence in the approximate solution of problem (5.11) at $T = 1$, using 2-node original and block Krylov subspace spectral methods. Error is the relative difference, in the 2-norm, between approximate solutions and a solution computed using a smaller time step, since no exact solution is available. N denotes the number of grid points and Δt denotes the time step used.

Method	N	Δt	Error	Order
KSS(2)	64	1	0.00096	3.02
		1/2	0.0001	
		1/4	1.2e-005	
		1/8	1.4e-006	
		1/16	1.7e-007	
		1/32	2.1e-008	
	128	1	0.0057	3.06
		1/2	0.00043	
		1/4	0.00013	
		1/8	1.6e-005	
		1/16	5e-007	
		1/32	6e-008	
KSS-B(2)	64	1	0.00013	3
		1/2	9.5e-006	
		1/4	9.2e-007	
		1/8	1e-007	
		1/16	1.2e-008	
		1/32	1.5e-009	
	128	1	0.00012	3.01
		1/2	1.4e-005	
		1/4	1.7e-006	
		1/8	2.7e-007	
		1/16	2.1e-008	
		1/32	2.6e-009	

block approach shows even more promise, and for more general problems of this type.

REFERENCES

[1] K. ATKINSON, An Introduction to Numerical Analysis, 2nd ed., Wiley, 1989.
[2] S. BASU AND N. K. BOSE, *Matrix Stieltjes series and network models*, SIAM J. Math. Anal., 14 (1983), pp. 209–222.
[3] G. DAHLQUIST, S. C. EISENSTAT, AND G. H. GOLUB, *Bounds for the error of linear systems of equations using the theory of moments*, J. Math. Anal. Appl., 37 (1972), pp. 151–166.
[4] G. H. GOLUB, *Some modified matrix eigenvalue problems*, SIAM Rev., 15 (1973), pp. 318–334.
[5] ———, *Bounds for matrix moments*, Rocky Mountain J. Math., 4 (1974), pp. 207–211.
[6] G. H. GOLUB AND G. MEURANT, *Matrices, Moments and Quadrature*, in Proceedings of the 15th Dundee Conference, June-July 1993, D. F. Griffiths and G. A. Watson, eds., Longman Scientific & Technical, 1994.
[7] ———, Matrices, Moments and Quadrature with Applications, Princeton University Press, in press.
[8] G. H. GOLUB AND C. F. VAN LOAN, Matrix Computations, 3rd ed., Johns Hopkins University Press, 1996.
[9] G. H. GOLUB AND R. UNDERWOOD, *The block Lanczos method for computing eigenvalues*, in Mathematical Software III, J. Rice Ed., 1977, pp 361–377.
[10] G. H. GOLUB AND J. WELSCH, *Calculation of Gauss quadrature rules*, Math. Comp., 23 (1969), 221–230.
[11] P. GUIDOTTI, J. V. LAMBERS, AND K. SØLNA, *Analysis of 1-D wave propagation in inhomogeneous media*, Num. Funct. Anal. Opt., 27 (2006), pp. 25–55.

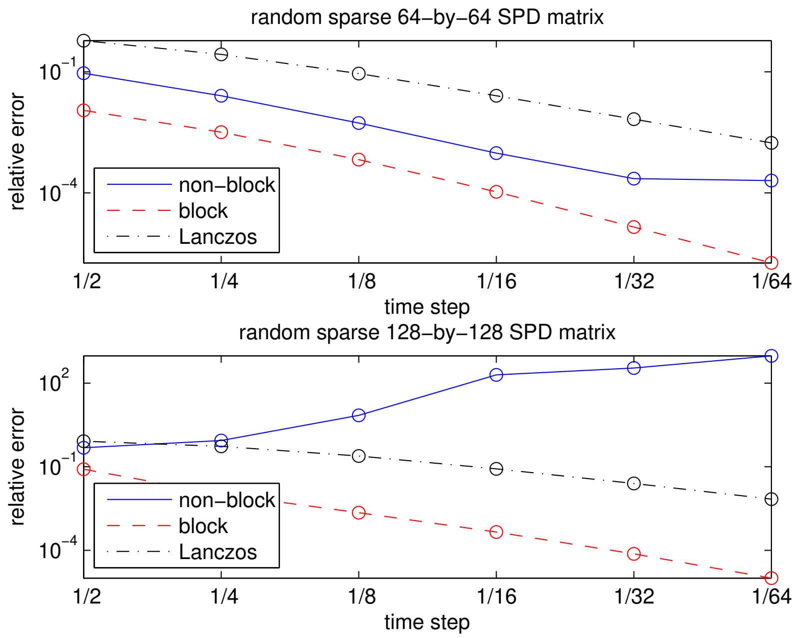

FIGURE 5.7. *Estimates of relative error in the computation of* $\mathbf{w} = e^{-At}\mathbf{v}$ *using original and block 3-node KSS methods, and the Lanczos algorithm as described in (5.12) with* $j = 3$*, where* A *is an* $N \times N$ *matrix defined in (5.14), and* $\mathbf{v}$ *is a discretization of the function* $v(x) = 1 + \sin x + \sin 2x$ *on a uniform* N*-point grid, for* $N = 64$ *and* $N = 128$*. The vector* $\mathbf{w}$ *is computed using the outer iteration (5.13), with* $\Delta t = 2^{-j}$*, for* $j = 1, 1, \ldots, 6$*.*

[12] M. HOCHBRUCK AND C. LUBICH, *On Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., 34 (1996), pp. 1911–1925.

[13] J. V. LAMBERS, *Derivation of high-order spectral methods for time-dependent PDE using modified moments*, Electron. Trans. Numer. Anal., 28 (2008), pp. 114–135, http://etna.math.kent.edu/vol.28.2007-2008/pp114-135.dir/.

[14] ———, *Implicitly defined high-order operator splittings for parabolic and hyperbolic variable-coefficient PDE using modified moments*, Int. J. Computational Sci., 2 (2008), pp. 376–401.

[15] ———, *Krylov Subspace Methods for Variable-Coefficient Initial-Boundary Value Problems*, Ph.D. Thesis, Stanford University, SCCM Program, 2003.
Available at http://sccm.stanford.edu/pub/sccm/theses/James_Lambers.pdf.

[16] ———, *Krylov subspace spectral methods for the time-dependent Schrödinger equation with non-smooth potentials*, submitted.

[17] ———, *Krylov subspace spectral methods for variable-coefficient initial-boundary value problems*, Electron. Trans. Numer. Anal., 20 (2005), pp. 212–234, http://etna.math.kent.edu/vol.20.2005/pp212-234.dir/.

[18] ———, *Practical implementation of Krylov subspace spectral methods*, J. Sci. Comput., 32 (2007), pp. 449–476.

[19] J. STOER AND R. BURLISCH, Introduction to Numerical Analysis, 2nd ed., Springer Verlag, 1983.

ETNA
Kent State University
http://etna.math.kent.edu

TABLE 5.5
Estimates of relative error and temporal order of convergence in the computation of $\mathbf{w} = e^{-At}\mathbf{v}$ *using original and block 3-node KSS methods, and the Lanczos algorithm as described in (5.12) with* $j = 3$*, where* A *is an* $N \times N$ *matrix defined in (5.14), and* $\mathbf{v}$ *is a discretization of the function* $v(x) = 1 + \sin x + \sin 2x$ *on a uniform* N*-point grid, for* $N = 64$ *and* $N = 128$*. The vector* $\mathbf{w}$ *is computed using the outer iteration (5.13), with* $\Delta t = 2^{-j}$*, for* $j = 1, 1, \ldots, 6$*.*

Method	N	Δt	Error	Order
KSS(3)	64	1/2	0.092	1.58
		1/4	0.025	
		1/8	0.0054	
		1/16	0.00096	
		1/32	0.00023	
		1/64	0.0002	
	128	1/2	0.48	-2.37
		1/4	0.88	
		1/8	7	
		1/16	2e+002	
		1/32	3.5e+002	
		1/64	9.6e+002	
KSS-B(3)	64	1/2	0.011	2.83
		1/4	0.0032	
		1/8	0.00067	
		1/16	0.00011	
		1/32	1.4e-005	
		1/64	1.8e-006	
	128	1/2	0.082	2.6
		1/4	0.01	
		1/8	0.0023	
		1/16	0.00046	
		1/32	7.5e-005	
		1/64	1e-005	
Lanczos(3)	64	1/2	0.59	1.9
		1/4	0.27	
		1/8	0.09	
		1/16	0.025	
		1/32	0.0067	
		1/64	0.0017	
	128	1/2	0.83	1.71
		1/4	0.54	
		1/8	0.24	
		1/16	0.084	
		1/32	0.025	
		1/64	0.0069	