A FRAMEWORK FOR POLYNOMIAL PRECONDITIONERS BASED ON FAST TRANSFORMS II: PDE APPLICATIONS*

SVERKER HOLMGREN† AND KURT OTTO†

Abstract. The solution of systems of equations arising from systems of time-dependent partial differential equations (PDEs) is considered. Primarily, first-order PDEs are studied, but second-order derivatives are also accounted for. The discretization is performed using a general finite difference stencil in space and an implicit method in time. The systems of equations are solved by a preconditioned Krylov subspace method. The preconditioners exploit optimal and superoptimal approximations by polynomials in a normal basis matrix, associated with a fast trigonometric transform. Numerical experiments for high-order accurate discretizations are presented. The results show that preconditioners based on fast transforms yield efficient solution algorithms, even for large quotients between the time and space steps. Utilizing a spatial grid ratio less than one, the arithmetic work per grid point is bounded by a constant as the number of grid points increases.

1. Introduction. We study the solution of systems of time-dependent PDEs:

$$
\left\{
\begin{array}{l}
\dfrac{\partial \mathbf{u}}{\partial t} + \mathcal{B}\mathbf{u} = \mathbf{g} \\
\text{local boundary conditions}
\end{array}
\right.
\tag{1}
$$

Here, $\mathbf{u}$ is an n_c-vector containing the solution functions, $\mathcal{B}$ is the differential operator in space, and $\mathbf{g}$ is an n_c-vector of forcing functions. In this paper, we consider linear PDEs solved on a finite region in two space dimensions, which can be smoothly mapped onto the unit square. However, as indicated later, it is possible to use the solution methods presented here also in the process of solving more general problems.

Our aim is to develop efficient, high-order accurate solution methods based on centered finite difference (or finite volume) stencils for first-order or "almost first-order" systems of PDEs. Problems of interest are the Euler and the Maxwell equations, or the Navier–Stokes equations for high Reynolds numbers. For this type of problems, artificial viscosity must often be added to damp high-frequency oscillations. Hence, it is important that the solution methods are able to handle second-order terms, even though potential viscosity terms are small.

The standard solution methods for problems of the type mentioned above often involve explicit time-marching, e.g., Runge–Kutta schemes. The stability criterion

*This work was supported by the Swedish National Board for Industrial and Technical Development (NUTEK) and by the U. S. National Science Foundation under grant ASC-8958544.

†Department of Scientific Computing, Uppsala University, Box 120, S-751 04 Uppsala, Sweden.

introduced by the explicit time-marching method demands that the maximal time step Δt is limited by some power of the minimal space step h_{min}. For a first-order PDE, a relation of the form $\Delta t < \kappa h_{min}$, where $\kappa = \mathcal{O}(1)$, must hold. For time-dependent computations, this is often not a problem. The accuracy of the time-marching method normally matches that of the spatial discretization, and the time step must be $\mathcal{O}(h_{min})$ to retain time accuracy. However, for some important problem classes, the stability criterion results in an unrealistically small time step. In that case, an implicit or semi-implicit time-marching scheme [GueGu86] must be employed. When exploiting implicit time-marching, at least one system of equations must be solved for each time level. In the work presented here, we study efficient algorithms for solving such systems.

The PDE (1) is discretized using a, possibly high-order accurate, finite difference operator on an $m_1 \times m_2$ grid and a suitable implicit time-marching method. Let $n \equiv n_c m_1 m_2$. In the time-marching procedure, the n-vector u containing the approximate solution of (1) at a new time level is obtained by solving systems of equations of the form

$$M^{-1}Bu = M^{-1}g. \tag{2}$$

Here, B is the coefficient matrix arising from the discretization of (1), and M is a preconditioner matrix.

The $n \times n$ matrix B is very sparse and highly structured. The number of nonzero elements is only $\mathcal{O}(n)$, which indicates that an iterative method should be used for solving (2). For the PDEs and the discretizations we are interested in, B is usually nonsymmetric and highly nondiagonally dominant. To attain an acceptable rate of convergence, a preconditioner must be utilized. In [Otto96], analysis and empirical results show that standard preconditioners of ILU type are not favorable for the type of problems of interest here. In [HoOtto90], [HoOtto92], [HoOtto94], and [HoOtto96], we have described semicirculant preconditioners. These preconditioners are used for problems of type (2), where B corresponds to second-order accurate discretizations of first-order time-dependent and time-independent PDEs. Empirically, the number of iterations required is found to be independent of problem size. In [Otto96] and [HoOtto94], the eigenvalues and eigenvectors of $M^{-1}B$, for the 21-discretization described in §4, are analyzed. Also, convergence estimates proving the favorable convergence properties are derived. In [Hemm96] and [HeOtto96], the same type of results are derived for semi-Toeplitz preconditioners. The purpose of this paper is to describe a wider class of preconditioners for discretizations of arbitrary order of accuracy. The preconditioners are based on the polynomial normal block (PNB) framework that is developed in Part I of this paper. As special cases, semicirculant and semi-Toeplitz preconditioners for second-order accurate approximations are included in this framework.

In §2, we use the results from §§3 and 4 of Part I for the construction of polynomial normal (PN) approximations for scalar PDEs in one space dimension. In §3, the more general problem of solving systems of PDEs in two space dimensions is considered. For that purpose, the PNB framework introduced in §7 of Part I is employed. Finally, three model problems are defined in §4, numerical experiments are presented in §5, and some concluding remarks are made in §6.

We use the notation

$$\mathrm{block}_{j,n_1;k,n_2}(\beta_{j,k}) = \begin{pmatrix} \beta_{1,1} & \cdots & \beta_{1,n_2} \\ \vdots & \ddots & \vdots \\ \beta_{n_1,1} & \cdots & \beta_{n_1,n_2} \end{pmatrix},$$

and let $\mathrm{pband}_{j,m}^{(r,-q_1:q_2)}(\beta_{j,r})$ denote an $m \times m$ periodic band matrix with bandwidth $q_1 + q_2 + 1$, as defined in §1 of Part I.

2. Approximations for scalar PDEs in one space dimension. This section serves as an introduction to the general theory, presented in Part I, applied to PDE problems. We first consider the computation of optimal and superoptimal PN approximations for matrices arising from scalar PDEs in 1D. For a problem of this type, the coefficient matrix B in (2) is narrow banded

$$B = \mathrm{pband}_{j,m}^{(r,-q_1:q_2)}(\beta_{j,r}). \tag{3}$$

The discretization in space and time may be of arbitrary order of accuracy, and the integers q_1 and q_2 are determined by the maximal width of the stencil in space. By defining B as a periodic band matrix, periodic boundary conditions are included in the framework. However, the PDE may be subject to *any* local boundary conditions, corresponding to a stencil with extension $\mathcal{O}(1)$ in space. Note that even though we study the case $q_1,\ q_2 = \mathcal{O}(1) \ll m$ here, the framework could easily be extended to spectral discretizations, where $q_1,\ q_2 = \mathcal{O}(m)$.

Example 1: Consider the first-order PDE:

$$\begin{cases} \dfrac{\partial \mathbf{u}}{\partial t} + \dfrac{\partial \mathbf{u}}{\partial x} = \mathbf{g}, \qquad 0 < x \leq 1, \quad t > 0 \\ \text{Dirichlet boundary condition at the inflow boundary } x = 0 \end{cases}$$

Assume that a sixth-order stencil is used in the interior and a specific set of fifth-order stencils is used at the boundaries (see the 65-discretization in §4). The resulting nonzero structure of B is depicted in Fig. 1.

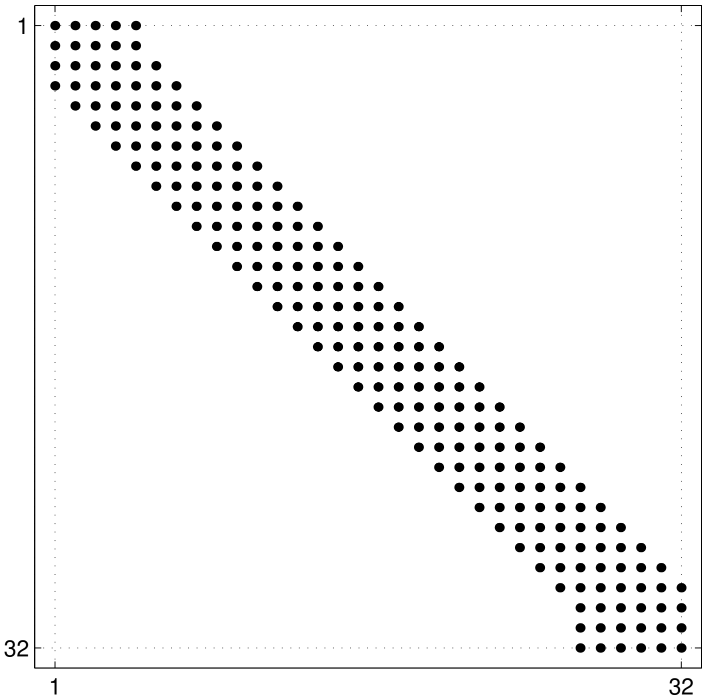

FIG. 1. *The nonzero structure of B.*

From Fig. 1, it is obvious that $q_1 = 5$ and $q_2 = 4$.

In Part I, PN approximations of B are discussed. These approximations are defined by

$$M = \sum_{r=0}^{\tilde{q}-1} \mu_r R^r, \quad \tilde{q} \leq m, \tag{4a}$$

or by

$$M^{\dotplus} = \sum_{r=0}^{\tilde{q}-1} \mu_r R^r, \quad \tilde{q} \leq m. \tag{4b}$$

Alternatively, a power series formulation corresponding to (3) in Part I could be used in (4a–b). The narrow banded normal basis matrix R corresponds to a basic stencil operation on a grid function. In §§3 and 4 of Part I, it is described how to compute the parameters μ_r, such that M and $M^{\dotplus}$ fulfill optimal or superoptimal minimization criteria, respectively.

It is not clear that optimal or superoptimal PN approximations M of B will yield good preconditioners in the sense that, when a Krylov subspace method is applied to (2), a sufficiently accurate solution u is computed in a small number of iterations. In fact, it may be considered more natural to exploit the optimal or superoptimal approximation of a *modified* matrix $B(\cdot)$ for preconditioning. The matrix $B(\cdot)$ is formed using the same computational stencil as was used for B in the interior, but the boundary conditions are altered to conform with those of the stencil corresponding to R. This is illustrated by a continued study of Example 1.

Example 1, *continued:* Assume that an approximation, where R is the periodic shift matrix C_m, should be computed. The boundary condition corresponding to C_m is periodic. Therefore, the boundary condition for the PDE in Example 1 could be changed accordingly, and the corresponding matrix $B(\text{periodic})$ is formed. Then, the PN approximation with the same sparsity structure as $B(\text{periodic})$ is computed. Thus, in the example considered here, formulation (3) in Part I is used with $\tilde{q}_1 = q_1$ and $\tilde{q}_2 = q_2$. If the difference approximation has constant coefficients then the optimal approximation M, using $R = C_m$, is the *exact* solution matrix for the PDE problem with modified boundary conditions, i.e., $M^{-1}B(\text{periodic}) = I$.

For $R = C_m$ (Case 1), the approximation is generally based on B or $B(\text{periodic})$, and $\tilde{q}_1 = q_1$ and $\tilde{q}_2 = q_2$ are chosen. For R in Cases 2, 3, 4, and 5, the boundary conditions may be changed accordingly, and $\tilde{q} = \mathcal{O}(q_1 + q_2)$ is a suitable choice. We conclude that $\tilde{q} = \mathcal{O}(1)$. From Table 2 in Part I, it follows that the total arithmetic costs to form the optimal and superoptimal approximations of B are $\mathcal{O}(m)$.

3. Approximations for systems of PDEs in two space dimensions. We use approximations of the form discussed in §7 of Part I to construct preconditioners for systems of n_c PDEs in two space dimensions.

Consider a system of equations of the form (2). The maximal extent of the computational stencil over all grid points is given by the four integers q_{11}, q_{12}, q_{21}, and q_{22}, see Fig. 2.

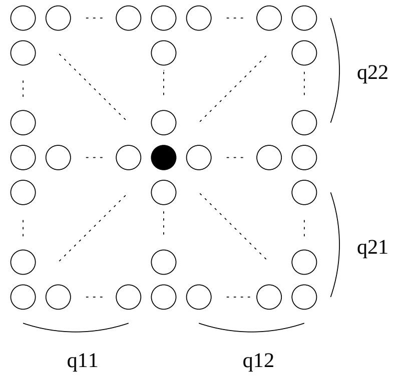

FIG. 2. *The maximal extent of the discretization stencil.*

For finite difference methods with local boundary conditions, q_{11}, q_{12}, q_{21}, and q_{22} are all $\mathcal{O}(1)$. The coefficient matrix B has a two-level (possibly periodic) band structure.

$$\begin{aligned} B &= \mathrm{pband}_{k,m_2}^{(r_2,-q_{21}:q_{22})}(B_{k,r_2}), \\ B_{k,r_2} &= \mathrm{pband}_{j,m_1}^{(r_1,-q_{11}:q_{12})}(B_{j,k,r_1,r_2}), \\ B_{j,k,r_1,r_2} &= \mathrm{block}_{j_c,n_c;k_c,n_c}(\beta_{j,k,r_1,r_2,j_c,k_c}). \end{aligned} \tag{5a–c}$$

Choose a normal $m_1 \times m_1$ basis matrix R_{m_1}, i.e.,

$$R_{m_1} = Q_{m_1}\Lambda_{m_1}Q_{m_1}^*,$$

where Λ_{m_1} is diagonal and Q_{m_1} is unitary. Define the PNB matrix M by replacing the band structure at the inner level by a "polynomial" in R_{m_1}.

$$\begin{aligned} M &= \mathrm{pband}_{k,m_2}^{(r_2,-q_{21}:q_{22})}(M_{k,r_2}), \\ M_{k,r_2} &= \sum_{r_1=0}^{\tilde{q}-1} R_{m_1}^{r_1} \otimes M_{k,r_1,r_2}, \\ M_{k,r_1,r_2} &= \mathrm{block}_{j_c,n_c;k_c,n_c}(\mu_{k,r_1,r_2,j_c,k_c}). \end{aligned} \tag{6a–c}$$

Again, a power series formulation corresponding to (3) in Part I could be used in (6b). The methods for computing approximations described in §7 of Part I are introduced in a straightforward way. Theorem 4 in Part I proves that this approach will yield the optimal approximation M to the two-level block matrix B defined in (5a–c). When superoptimal approximations are utilized, these are computed blockwise for B_{k,r_2}. Since the problem of computing superoptimal approximations of block matrices is "nonseparable", the resulting approximation M is only blockwise superoptimal. Using the arithmetic complexities given in Part I, it is easy to derive that the total cost for forming the optimal or blockwise superoptimal preconditioner is $\mathcal{O}(m_2m_1)$.

The matrix M is easily inverted. Consider the transformation

$$N = (I_{m_2} \otimes Q^*_{m_1} \otimes I_{n_c}) M (I_{m_2} \otimes Q_{m_1} \otimes I_{n_c}).$$

This yields

$$M^{-1} = (I_{m_2} \otimes Q_{m_1} \otimes I_{n_c}) N^{-1} (I_{m_2} \otimes Q^*_{m_1} \otimes I_{n_c}). \tag{7}$$

Now

$$N = \mathrm{pband}_{k,m_2}^{(r_2,-q_{21}:q_{22})}(\Lambda_{k,r_2}),$$

where Λ_{k,r_2} are block diagonal with block size $n_c \times n_c$:

$$\begin{aligned}
\Lambda_{k,r_2} &= (Q^*_{m_1} \otimes I_{n_c}) M_{k,r_2} (Q_{m_1} \otimes I_{n_c}) \\
&= (Q^*_{m_1} \otimes I_{n_c}) \Big(\sum_{r_1=0}^{\tilde{q}-1} R^{r_1}_{m_1} \otimes M_{k,r_1,r_2} \Big) (Q_{m_1} \otimes I_{n_c}) \\
&= \sum_{r_1=0}^{\tilde{q}-1} Q^*_{m_1} R^{r_1}_{m_1} Q_{m_1} \otimes M_{k,r_1,r_2} = \sum_{r_1=0}^{\tilde{q}-1} \Lambda^{r_1}_{m_1} \otimes M_{k,r_1,r_2}.
\end{aligned}$$

Exploiting (7), it is evident that the PNB solve $u = M^{-1}g$ can be performed as

1. $v = (I_{m_2} \otimes Q^*_{m_1} \otimes I_{n_c})g$
2. solve $Nz = v$
3. $u = (I_{m_2} \otimes Q_{m_1} \otimes I_{n_c})z$

PNB SOLVE, FORMULATION 1.

Since the multiplications by $Q^*_{m_1}$ and Q_{m_1} can be realized by fast transforms with $\mathcal{O}(m_1 \log_2 m_1)$ complexities, a more algorithmic formulation is

1. Perform $n_c m_2$ independent fast transforms of length m_1.
2. Solve m_1 independent block banded systems of $n_c m_2$ equations (block bandwidth $q_{21} + q_{22} + 1$ and block size $n_c \times n_c$).
3. Perform $n_c m_2$ independent fast inverse transforms of length m_1.

PNB SOLVE, FORMULATION 2.

It is easily verified that the total complexity is $\mathcal{O}(m_2 m_1 \log_2 m_1)$. For problems where g and u are real, it is possible to exploit symmetries in the fast transforms to effectively halve the number of arithmetic operations, see, e.g., [Holm92] and [Hemm94]. Also, note that the potential for parallelization is substantial. Parallel implementations of PNB solvers will be studied in future work.

4. Model problems. As a representative model for the PDE problems of interest, we use a scalar hyperbolic PDE subject to Dirichlet boundary conditions in a time-dependent setting

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{u}}{\partial x_1} + \frac{\partial \mathbf{u}}{\partial x_2} = \mathbf{g}. \tag{8}$$

The PDE is solved on the unit square for $t > 0$. If $\mathbf{u}(0, x_2, t)$, $\mathbf{u}(x_1, 0, t)$, and $\mathbf{u}(x_1, x_2, 0)$ are supplied, (8) is a well-posed problem. Note that the solution must not be prescribed at the outflow boundaries.

The spatial discretization is performed on a uniform grid with $(m_1+1)\times(m_2+1)$ grid points. The space step in the x_d-direction is given by $h_d \equiv 1/m_d$, and the grid points are identified by the pair of numbers (j, k), $j = 0, \ldots, m_1$, $k = 0, \ldots, m_2$. Introducing the time and space discretizations described below yields systems of equations of the form (2), where

$$B \equiv I + \gamma(I_{m_2} \otimes \kappa_1 B_1 + \kappa_2 B_2 \otimes I_{m_1}). \tag{9}$$

The matrices $h_d^{-1} B_d$ are discrete approximations of $\frac{\partial}{\partial x_d}$ of the form (3), and $\kappa_d \equiv \Delta t / h_d$. Note that we are interested in solving problems where $\max_d \kappa_d \gg 1$. The scalar coefficient γ is determined by the choice of time-marching method.

Let p_i denote an even, positive integer. For the interior of the domain, p_ith-order accurate, antisymmetric, centered difference operators S_{p_i} are used for approximating the spatial derivatives in (8). General formulas for such operators may be found, e.g., in [FoSl94]. The operators $S_{2,4,6}$ are given by

$$\begin{aligned} S_2 &= h^{-1}\Delta_0, \\ S_4 &= h^{-1}\Delta_0(I - \tfrac{1}{6}\Delta_+\Delta_-), \\ S_6 &= h^{-1}\Delta_0\big(I - \tfrac{1}{6}\Delta_+\Delta_- + \tfrac{1}{30}(\Delta_+\Delta_-)^2\big), \end{aligned}$$

where

$$\Delta_0 \equiv \frac{1}{2}(\Delta_+ + \Delta_-), \qquad \Delta_+ u_j \equiv u_{j+1} - u_j, \qquad \Delta_- u_j \equiv \Delta_+ u_{j-1}.$$

Here, u_j denotes the value of a grid function u, defined on a uniform grid with space step h. The widths of the stencils defined above are $p_i + 1$, i.e., the "interior" lower and upper bandwidths of the matrices B_d are $p_i/2$. Near the boundaries numerical boundary conditions must be supplied. For each grid row and column, at least $p_i/2 - 1$ numerical boundary conditions are required at the inflow boundary. At the outflow boundary, at least $p_i/2$ extra equations are needed. This corresponds to modifications of the $p_i/2-1$ first and $p_i/2$ last rows in the matrices B_d. Numerical boundary conditions for high-order approximations to general initial-boundary value problems are discussed in, e.g., [Gust81]. If the order of accuracy for the numerical boundary conditions is denoted by p_b then, under some conditions, the global order of accuracy is $p=\min(p_b+1, p_i)$.

Different approaches can be used for determining the extra equations. A viable option is to use "as centered as possible" stencils of order p_i-1. The widths of these stencils are p_i, implying that the lower and upper bandwidths of B_d are given by $q_1=q_2+1=p_i-1$. In [FoSl94], a general algorithm for constructing finite difference formulas of this type is described. Below, the lower right corner of the matrix B_d, corresponding to the outflow boundary, is shown for the difference schemes of the type described above. The numerical boundary conditions for the inflow boundary are trivially obtained by reflection.

21-*discretization:* $p_i=2$, $p_b=1$.

$$B_d=\begin{pmatrix} \ddots & \ddots & \ddots & \\ & -\frac{1}{2} & 0 & \frac{1}{2} \\ & & -1 & 1 \end{pmatrix}$$

43-*discretization:* $p_i=4$, $p_b=3$.

$$B_d=\begin{pmatrix} \ddots & \ddots & \ddots & \ddots & \ddots & \\ & \frac{1}{12} & -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{12} \\ & & \frac{1}{6} & -1 & \frac{1}{2} & \frac{1}{3} \\ & & -\frac{1}{3} & \frac{3}{2} & -3 & \frac{11}{6} \end{pmatrix}$$

65-*discretization:* $p_i=6$, $p_b=5$.

$$B_d=\begin{pmatrix} \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & -\frac{1}{60} & \frac{3}{20} & -\frac{3}{4} & 0 & \frac{3}{4} & -\frac{3}{20} & \frac{1}{60} \\ & & -\frac{1}{30} & \frac{1}{4} & -1 & \frac{1}{3} & \frac{1}{2} & -\frac{1}{20} \\ & & \frac{1}{20} & -\frac{1}{3} & 1 & -2 & \frac{13}{12} & \frac{1}{5} \\ & & -\frac{1}{5} & \frac{5}{4} & -\frac{10}{3} & 5 & -5 & \frac{137}{60} \end{pmatrix}$$

The time discretization is performed using A-stable singly diagonally implicit (SDI) or semiexplicit (SDSE) Runge–Kutta methods [HaiWan91]. These methods are usually described by a tableau:

$$
\begin{array}{ccccccc}
\gamma & & & & & & \\
a_{2,1} & \gamma & & & & & \\
\vdots & & \ddots & & & & \\
a_{s_i,1} & \cdots & \cdots & \gamma & & & \\
a_{s_i+1,1} & \cdots & \cdots & \cdots & 0 & & \\
\vdots & & & & & \ddots & \\
a_{s,1} & \cdots & \cdots & \cdots & \cdots & a_{s,s-1} & 0 \\
\hline
b_1 & \cdots & \cdots & \cdots & \cdots & b_{s-1} & b_s
\end{array}
$$

Here, s_i is the number of implicit stages. For an SDIRK method, $s_i = s$, while for an SDSERK method, $s_i < s$ holds. For each time level, s_i systems of equations with a coefficient matrix given by (9) must be solved. Note that all the systems have the same coefficient matrix B, since the parameter γ is the same for all the implicit stages. The following methods are considered:

Second-order accurate time-marching: The unique one-stage, second-order accurate, A-stable method is the implicit midpoint rule; hence $s_i = 1$ and $\gamma = \frac{1}{2}$.

Fourth-order accurate time-marching: A three-stage, fourth-order accurate, A-stable SDIRK method is described in [HaiWan91]. The unique γ is obtained as a root of a third degree polynomial: $\gamma = \frac{1}{\sqrt{3}}\cos(\frac{\pi}{18}) + \frac{1}{2} \approx 1.06857902130163$.

Sixth-order accurate time-marching: A sixth-order SDSERK method, where $s = 6$ and $s_i = 5$, is constructed in [CooSay79]. Here, γ is chosen as a root of a fifth degree polynomial yielding $\gamma \approx 0.473268391258295$.

5. Numerical experiments. In §4, we have defined three systems of equations of the form (2). Below, these systems of equations are solved utilizing the GMRES(20) algorithm [SaadSch86], combined with PNB preconditioners. In all the experiments, the criterion for convergence is that the preconditioned residual should be reduced by a factor of 10^{-6}. For the problems studied here, the unpreconditioned methods do not converge for large problem sizes. Consequently, it is not possible to compare, e.g., arithmetic complexities for the preconditioned and unpreconditioned methods. We are not aware of any other preconditioning technique that is successful when applied to problems of the type studied below.

Extensive numerical experiments have been performed, where many combinations of parameters and methods have been tested. It is not practical to account for all of these experiments here. We will give some important general conclusions supported by the numerical experiments. However, we will only show numerical results for a few, typical or extra interesting, cases.

Iterative method: Experiments reveal that the arithmetic performance is not very sensitive to the choice of restarting length in the GMRES algorithm; e.g., a restarting length of 10 or 30 yields comparable arithmetic performances. Also, no major difference in the number of matrix–vector multiplies and preconditioner solves required is observed when, e.g., the TFQMR algorithm [Freund93] is substituted for GMRES.

Conclusion 1: The choice of Krylov subspace method or restarting length is not very important.

Grid ratio: In [Otto96] and [HeOtto96], analytical results prove that utilizing a *grid ratio* $\beta \equiv m_2/m_1 < 1$ yields good convergence properties when semicirculant and semi-Toeplitz preconditioners are exploited for the 21-discretization. Experiments demonstrate that this holds also when the more general class of PNB preconditioners are employed for the high-order discretizations described in §4. A typical result is shown in Table 1, where systems of equations arising from the 43-discretization are solved using $R_{m_1} = C_{m_1}$ (Case 1), $\kappa_1 = 100$, $m_1 = 512$, and $m_2 = \beta m_1$.

TABLE 1. *The number of iterations required for different grid ratios β.*

β	1	63/64	31/32	15/16	7/8	3/4	1/2	1/4
# it.	91	58	43	34	27	21	14	12

Conclusion 2: When utilizing PNB preconditioners for solving first-order PDE problems, $\beta < 1$ is advantageous. In the sequel, we set $\beta = 1/2$, i.e., $m_2 = m_1/2$.

Basis matrix: Semicirculant and semi-Toeplitz preconditioners are included in the PNB framework as special cases. Here, R_{m_1} is selected according to Cases 1 and 2 in Part I, respectively. When solving first-order PDEs, these choices are natural, since they allow for an accurate approximation of the "interior part" of the matrix B_1. Cases 3–5 correspond to second-order derivatives, and one might expect that these bases will yield worse approximations of B_1. Experiments validate this claim. For all the discretizations in §4, utilizing R_{m_1} from Cases 3–5 yields extremely slow convergence. This does *not* imply that these basis matrices are completely useless. When approximating matrices corresponding to second-order derivatives, selecting R_{m_1} from Cases 3–5 is natural. In [Lars94], a preconditioner where Q_{m_1} is chosen according to Case 5 is successfully exploited to solve the Helmholtz equation.

Heuristically, Case 1 ($R_{m_1} = C_{m_1}$) has one big advantage. A power series in the periodic shift matrix allows for an accurate approximation of the "interior part" of B_1 corresponding to *any combination* of derivatives of *general order*, discretized using a *general* finite difference approximation. When employing R_{m_1}, corresponding to

Case 2 or Cases 3–5, the approximation class is restricted to polynomials of antisymmetric or symmetric matrices, respectively. This corresponds to powers of centered first differences *or* centered second differences. As shown in §4, high-order centered difference approximations of $\frac{\partial}{\partial x}$ are constructed from the centered first-difference operator Δ_0 *and* powers of the centered second-difference operator $\Delta_+\Delta_-$. The operator S_2 contains only the first-difference Δ_0. For the 21-discretization, experiments show that comparable performances are obtained for Case 1 and Case 2. The discussion above indicates that this could not be expected for high-order accurate discretizations, containing second-difference contributions in B_1. A typical result is given in Table 2, where systems of equations arising from the 43-discretization are solved using R_{m_1} according to Cases 1 and 2, $\kappa_1 = 100$, and $m_1 = 2^p$.

TABLE 2. *The number of iterations required for different problem sizes, $\kappa_1 = 100$.*

p	4	5	6	7	8	9	10
Case 1	18	28	30	26	19	14	13
Case 2	32	80	200	391	> 2000	> 2000	> 2000

Conclusion 3: For high-order approximations of first-order PDE problems, $R_{m_1} = C_{m_1}$ (Case 1) should be used. By conjecture this holds for any problem where "the interior" of B_1 contains both first- and second-difference contributions.

Optimal or blockwise superoptimal approximations: Our numerical experiments show that utilizing optimal approximations usually yields good performance, whereas the corresponding blockwise superoptimal approximations cause very slow convergence. In Fig. 3, spectra of $M^{-1}B$ for the 43-discretization, exploiting $R_{m_1} = C_{m_1}$ (Case 1), $\kappa_1 = 100$, and $m_1 = 32$, are displayed.

From Fig. 3, it is clear that the spectrum for the blockwise superoptimal approximation is not favorable for a Krylov subspace method. A possible explanation is that, when employing superoptimal approximations for 2D problems, the minimization property holds only blockwise. However, computations show that the blockwise superoptimal approximation is closer to B^{-1} than the optimal approximation, measured in the Frobenius norm. Spectra computed for the 1D problem in §2 exhibit the same qualitative behavior as the spectra in Fig. 3. For 1D problems, both the optimal and superoptimal approximations are minimizers of the corresponding norms. The conclusion is that, as in many other cases, the convergence properties for Krylov subspace methods cannot be optimized by minimizing a scalar measure such as a norm or a condition number.

Conclusion 4: For the first-order PDE problem studied here, optimal approximations should be utilized.

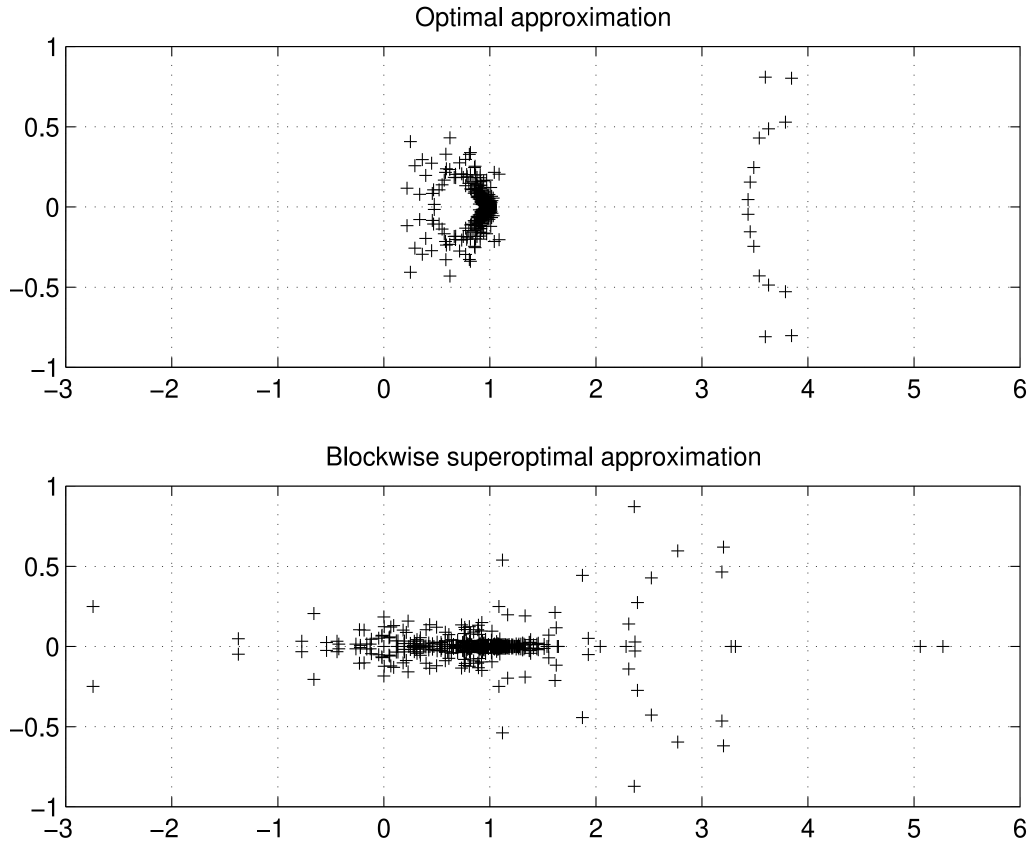

FIG. 3. *Spectra of* $M^{-1}B$, 43-*discretization.*

Boundary conditions used when forming the approximation: In §2 it is remarked that the optimal and superoptimal approximations should possibly be based on a modified matrix B, where the boundary conditions are altered to conform with those used in the basis matrix R. Numerical experiments demonstrate that altering the boundary conditions for B when forming the preconditioner often yields slightly better performance. Table 3 contains a typical result. The number of iterations needed for problems arising from the 43-discretization, where $R_{m_1} = C_{m_1}$ (Case 1), $\kappa_1 = 100$, and $m_1 = 2^p$, is shown. The optimal approximation M is based on B, where the original boundary conditions from the PDE are retained, or on B(periodic).

TABLE 3. *The number of iterations required for different problem sizes,* $\kappa_1 = 100$.

p	4	5	6	7	8	9	10
M based on B	22	28	30	25	20	18	14
M based on B(periodic)	18	28	30	26	19	14	13

In Fig. 4, spectra of $M^{-1}B$ and $M^{-1}B$(periodic) are displayed for the case $m_1 = 32$. The spectrum of $M^{-1}B$ is less scattered than that of $M^{-1}B$(periodic). On the other hand, the number of distinct eigenvalues is much smaller when the boundary conditions are altered. As shown in Table 3, the iterative method converges in the same number of steps for both cases. In the sequel, the boundary conditions are altered to conform with those of R_{m_1}.

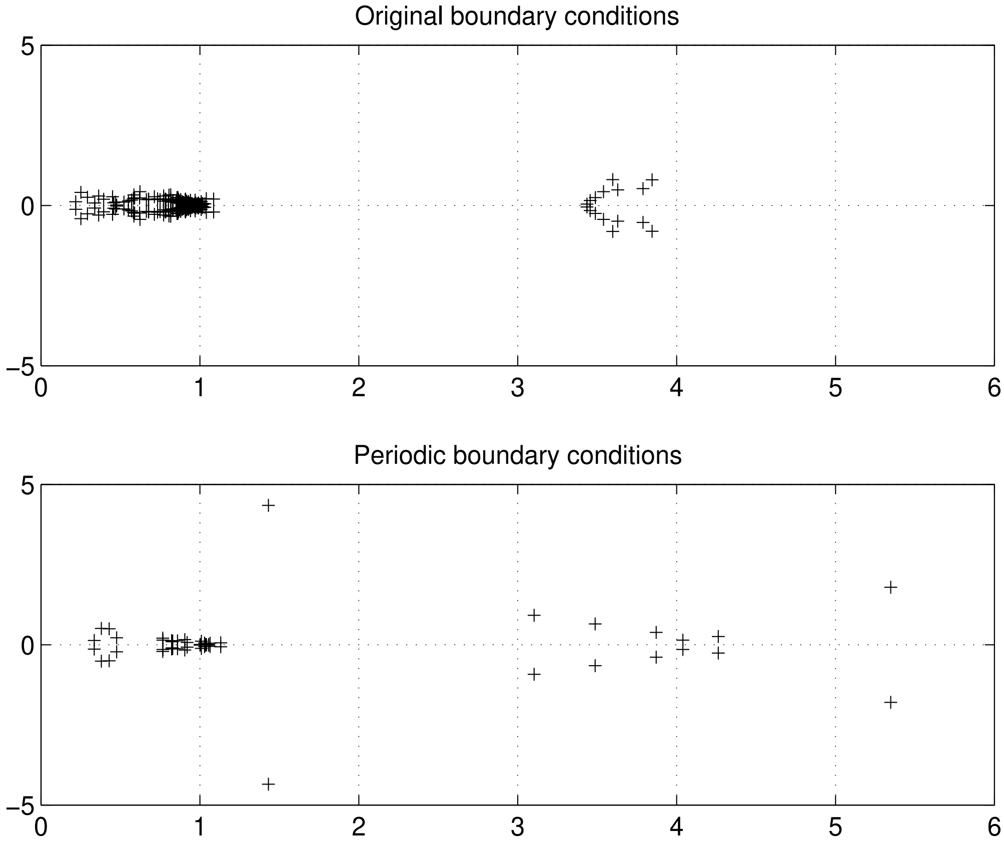

FIG. 4. *Spectra of* $M^{-1}B$ *and* $M^{-1}B$(periodic), 43-*discretization.*

Conclusion 5: When forming M, the boundary conditions in B may be altered to conform with those of R_{m_1}, or the boundary conditions from the PDE may be retained. For the problems studied here and for theoretical analysis, the boundary conditions in B should conform with those in R_{m_1}. However, the difference in convergence rate between the two approaches is very small.

We now examine the arithmetic performance when $R_{m_1} = C_{m_1}$ (Case 1) is chosen and optimal approximations are used. In Table 4, the number of iterations required for problems, where $\kappa_1 = 100$ and $m_1 = 2^p$, is presented for the 21-, 43-, and 65-discretizations.

TABLE 4. *The number of iterations required for different problem sizes,* $\kappa_1 = 100$.

p	4	5	6	7	8	9	10
21-discr.	17	18	16	11	9	7	7
43-discr.	18	28	30	26	19	14	13
65-discr.	20	32	43	60	48	46	44

From Table 4 it is clear that, for all the discretizations, the number of iterations *decreases* when the number of unknowns is increased. In Fig. 5, the number of arithmetic operations per unknown is shown for the same experiments. The solid, dashed, and dash-dot lines represent the 21-, 43-, and 65-discretizations, respectively.

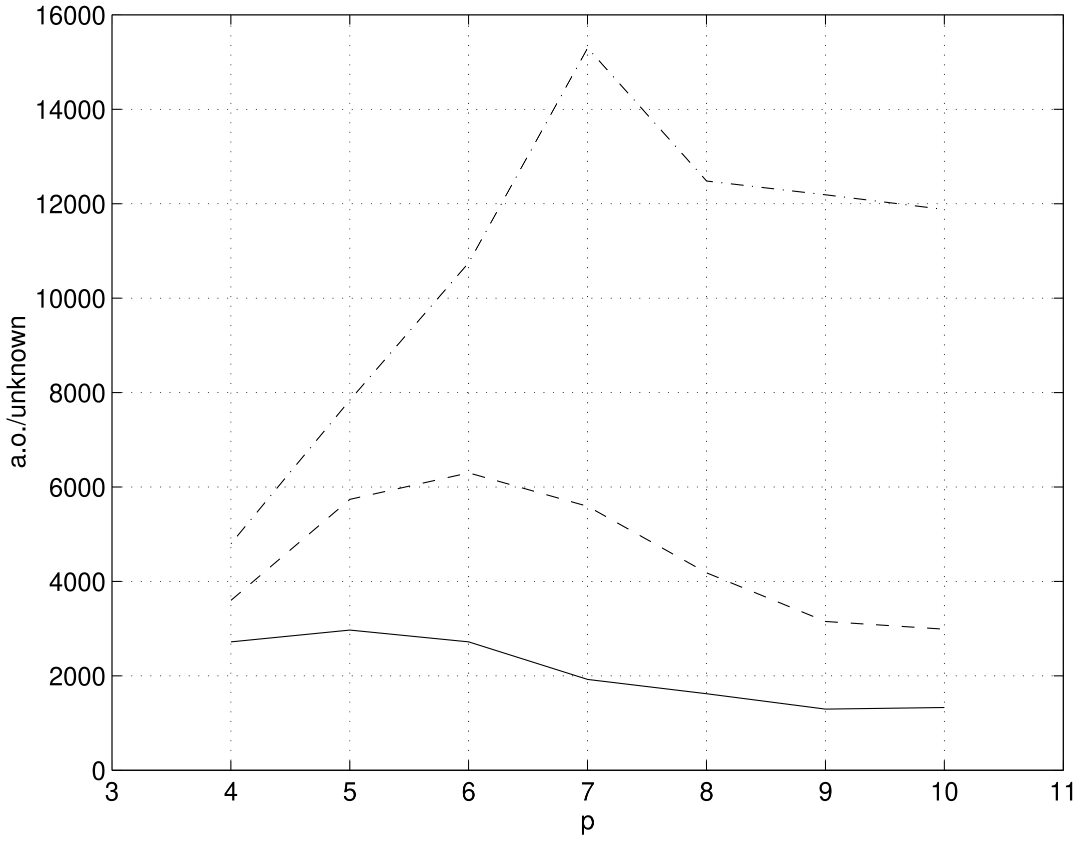

FIG. 5. *Arithmetic work per unknown for different problem sizes,* $\kappa_1 = 100$.

Figure 5 reveals that the arithmetic work per unknown stays bounded when the number of unknowns is increased. The decrease in the number of iterations more than cancels the logarithmic term in the arithmetic complexity for the PNB solver. In Table 5 and Fig. 6, the same type of results are given for $\kappa_1 = 1000$.

TABLE 5. *The number of iterations required for different problem sizes, $\kappa_1 = 1000$.*

p	4	5	6	7	8	9	10
21-discr.	17	25	37	38	34	19	11
43-discr.	19	30	46	81	84	70	37
65-discr.	20	35	149	649	1146	691	190

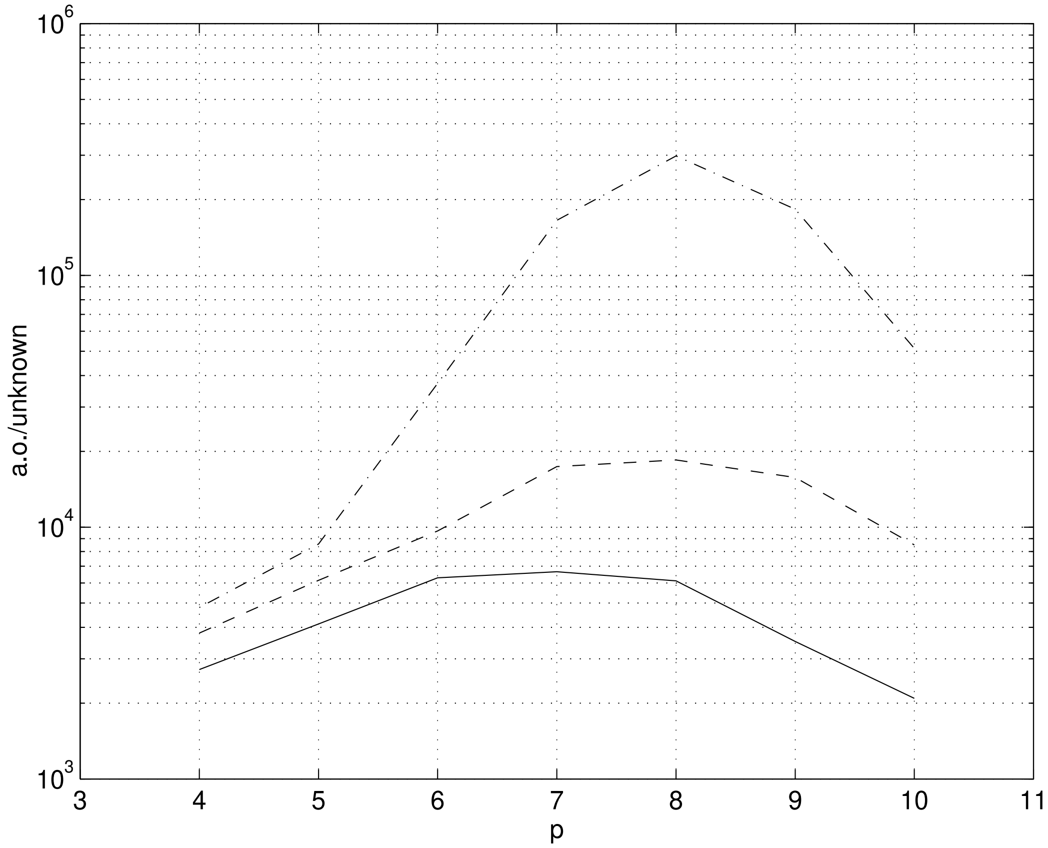

FIG. 6. *Arithmetic work per unknown for different problem sizes, $\kappa_1 = 1000$.*

From Fig. 6, it is clear that the favorable properties are retained also for larger values of κ_1. However, the number of iterations needed for the 65-discretization is large for medium-size problems.

Finally, in Fig. 7, problems where $m_1 = 1024$ are solved for different values of κ_1. As before, the solid, dashed, and dash-dot lines represent the 21-, 43-, and 65-discretizations, respectively.

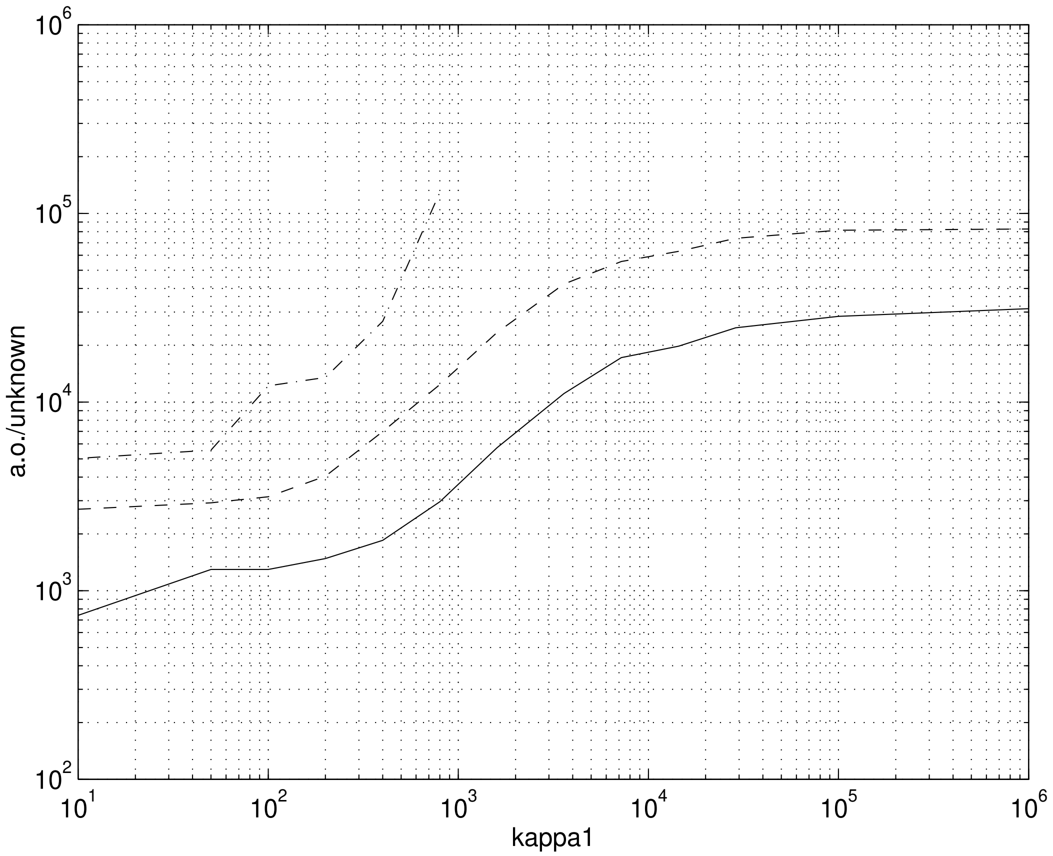

FIG. 7. *Arithmetic work per unknown for different* κ_1.

The results presented in Fig. 7 show that, for the 21- and 43-discretizations, the arithmetic complexity per unknown grows only slowly as κ_1 is augmented. Even for very large values of κ_1, corresponding to very large time steps, the arithmetic work required for solving the system of equations (2) is quite reasonable. For the 65-discretization, the arithmetic complexity grows faster when κ_1 is increased, and when $\kappa_1 \geq 1600$, more than 1000 iterations are needed.

Conclusion 6: When utilizing optimal PNB approximations, exploiting R_{m_1} according to Case 1, as preconditioners for the systems of equations arising from the high-order approximations of a first-order PDE problem defined in §4, the arithmetic work per unknown stays bounded as the problem size increases, even for large values of κ_1. For the 21- and 43-discretizations, the arithmetic complexity grows very slowly when κ_1 is augmented. Hence, the solution procedure is efficient for problems where very long time steps are allowed.

6. Concluding remarks. The numerical experiments presented in §5 show that, for the systems of equations in §4, PNB preconditioners yield efficient solution methods. Even for large quotients between the time and space steps, the arithmetic complexity per unknown stays bounded as the number of unknowns is increased. Also, for a fixed number of unknowns, the arithmetic complexity grows only slowly as this quotient is augmented.

The systems of equations defined in §4 arise from discretizations of a model problem, i.e., a scalar first-order PDE in two space dimensions. Note that the definition of the PNB preconditioner in §3 is given for a system of n_c PDEs. Furthermore, using a generalization of the technique presented in an appendix of [Otto93], the framework could easily be extended to 3D. The arithmetic complexity for forming the preconditioner is $\mathcal{O}(m_2 m_1)$, and the memory requirement is at most $\mathcal{O}(m_2 m_1)$. An implementation requiring only $\mathcal{O}(m_2)$ memory positions is possible at the expense of a slightly larger arithmetic complexity. Hence, it is possible to apply the technique described in this paper to large-scale application problems. For second-order discretizations, this has been exemplified in [HoOtto96] and [Hemm96], where the Navier–Stokes equations have been sucessfully solved in different geometries. In future work, PNB preconditioners will be exploited for solving systems of equations arising from high-order discretizations of application problems in CFD, CEM, and acoustics.

The difference operators in §4 have constant coefficients. In applications, variable coefficients, nonlinear contributions in the PDEs, and nonuniform grids usually result in difference operators with variable coefficients. In the preconditioner M, the difference approximation in the x_1-direction is replaced by an optimal or superoptimal constant-coefficient approximation. The effect of variable coefficients for the 21-discretization is studied in [HoOtto96]. For weakly variable coefficients in the x_1-direction, the solution method retains the favorable convergence properties described above. Note that strongly variable coefficients in the x_2-direction do not present a problem, since these are included in the approximation.

For a problem with variable coefficients, the degeneracy of the spectrum for $M^{-1}B$(periodic) will normally disappear. However, this degeneracy is crucial for convergence analyses of high-order discretizations, using the technique developed in [Otto96], [HoOtto94], and [HeOtto96] for the 21-discretization. Such analyses will be pursued in future work.

REFERENCES

[CooSay79] G. J. Cooper and A. Sayfy, *Semiexplicit A-stable Runge–Kutta methods*, Math. Comp., 33 (1979), pp. 541–556.

[FoSl94] B. Fornberg and D. M. Sloan, *A review of pseudospectral methods for solving partial differential equations*, Acta Numerica, 3 (1994), pp. 203–267.

[Freund93] R. W. Freund, *A transpose-free quasi-minimal residual algorithm for non-Hermitian linear systems*, SIAM J. Sci. Comput., 14 (1993), pp. 470–482.

[GueGu86] J. Guerra and B. Gustafsson, *A semi-implicit method for hyperbolic problems with different time-scales*, SIAM J. Numer. Anal., 23 (1986), pp. 734–749.

[Gust81] B. Gustafsson, *The convergence rate for difference approximations to general mixed initial boundary value problems*, SIAM J. Numer. Anal., 18 (1981), pp. 179–190.

[HaiWan91] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations* II, Springer-Verlag, Berlin, 1991.

[Hemm94] L. Hemmingsson, *A fast modified sine transform for solving block-tridiagonal systems with Toeplitz blocks*, Numer. Algorithms, 7 (1994), pp. 375–389.

[Hemm96] ———, *Toeplitz preconditioners with block structure for first-order PDEs*, Numer. Linear Algebra Appl., 3 (1996), to appear.

[HeOtto96] L. Hemmingsson and K. Otto, *Analysis of semi-Toeplitz preconditioners for first-order PDEs*, SIAM J. Sci. Comput., 17 (1996), to appear.

[Holm92] S. Holmgren, *CG-like iterative methods and semicirculant preconditioners on vector and parallel computers*, Report No. 148, Dept. of Scientific Computing, Uppsala Univ., Uppsala, Sweden, 1992.

[HoOtto90] S. Holmgren and K. Otto, *A comparison of preconditioned iterative methods for nonsymmetric block-tridiagonal systems of equations*, Report No. 123 (revised), Dept. of Scientific Computing, Uppsala Univ., Uppsala, Sweden, 1990.

[HoOtto92] ———, *Iterative solution methods and preconditioners for block-tridiagonal systems of equations*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 863–886.

[HoOtto94] ———, *Semicirculant preconditioners for first-order partial differential equations*, SIAM J. Sci. Comput., 15 (1994), pp. 385–407.

[HoOtto96] ———, *Semicirculant solvers and boundary corrections for first-order partial differential equations*, SIAM J. Sci. Comput., 17 (1996), to appear.

[Lars94] E. Larsson, *Numerical solution of the Helmholtz equation*, UPTEC report No. 94 083E, Inst. of Technology, Uppsala Univ., Uppsala, Sweden, 1994.

[Otto93] K. Otto, *Construction and Analysis of Preconditioners for First-order PDE*, Ph.D. thesis, Dept. of Scientific Computing, Uppsala Univ., Uppsala, Sweden, 1993.

[Otto96] K. Otto, *Analysis of preconditioners for hyperbolic partial differential equations*, SIAM J. Numer. Anal., 33 (1996), to appear.

[SaadSch86] Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 856–869.