Iterative solution of multiple radiation and scattering problems in structural acoustics using a block quasi-minimal residual algorithm[1]

Manish Malhotra
Department of Civil Engineering, Stanford University, Stanford, California 94305–4020, U.S.A.

Roland W. Freund
Bell Laboratories, Room 2C–420, 700 Mountain Avenue, Murray Hill, New Jersey 07974, U.S.A.

Peter M. Pinsky
Department of Mechanical Engineering, Stanford University, Stanford, California 94305, U.S.A.

Abstract

Finite-element discretizations of time-harmonic acoustic wave problems in exterior domains result in large sparse systems of linear equations with complex symmetric coefficient matrices. In many situations, these matrix problems need to be solved repeatedly for different right-hand sides, but with the same coefficient matrix. Recently, Freund and Malhotra have proposed a block quasi-minimal residual (BL-QMR) algorithm [11] for the iterative solution of non-Hermitian linear systems with multiple right-hand sides. The BL-QMR algorithm is a block Krylov-subspace iterative method that incorporates deflation to delete linearly and almost linearly dependent vectors in the underlying block Krylov sequences.

In this paper, we describe a **J**-symmetric variant of the BL-QMR algorithm that introduces important simplifications for the case when the coefficient matrix is symmetric with respect to a bilinear form induced by a certain matrix **J**. In particular, the **J**-symmetric variant includes the complex symmetric form of BL-QMR as a special case. We identify suitable preconditioners for the BL-QMR algorithm applied to multiple radiation and scattering problems. Our numerical tests with the preconditioned BL-QMR algorithm for such multiple linear systems show that, instead of solving each of the linear systems individually, it is significantly more efficient to employ the block version of the iterative method. Moreover, the numerical results clearly illustrate the importance of deflation and its effect on iterative convergence.

---

[1]Numerical Analysis Manuscript No. 96–11, Bell Laboratories, Murray Hill, New Jersey, May 1996. Available on WWW at `http://cm.bell-labs.com/cs/doc/96`.

1. Introduction

We are concerned with the solution of systems of linear equations with complex symmetric coefficient matrices that arise from finite-element discretizations of time-harmonic acoustic wave problems in exterior domains. In many situations, these matrix problems need to be solved repeatedly for different right-hand sides, but with the same coefficient matrix. For instance, multiple right-hand sides occur naturally in radiation problems due to multiple load cases, and in scattering problems when multiple angles of incidence of an incoming plane wave need to be considered. Similar problems also arise in other applications involving wave propagation, such as electromagnetics (e.g., antenna propagation and radar scattering problems) and seismology.

Direct solution methods, such as Gaussian elimination, are well suited for solving multiple linear systems in situations when the coefficient matrix can be stored and factorized efficiently. Since the matrix factorization is the dominating computational cost in direct methods, multiple right-hand side problems can be handled easily by performing the factorization only once, and then obtaining multiple solutions at a cost of two backsolves per right-hand side. However, since the linear systems to be solved in wave simulations can become arbitrarily large, the use of direct methods can become prohibitive, and then iterative techniques must be employed. A popular class of iterative linear-systems solvers is *Krylov-subspace methods*; we refer the reader to [10] for a survey. The class of Krylov-subspace iterations includes the classical conjugate gradient (CG) algorithm [16] for Hermitian positive definite linear systems, as well as more recent CG-type methods for non-Hermitian linear systems, such as the *generalized minimal residual* (GMRES) method [21] and the *quasi-minimal residual* (QMR) method [12, 8]. Krylov-subspace iterations are tailored to the solution of linear systems with a single right-hand side. They could be trivially used to solve multiple right-hand side problems by simply solving all these systems individually. However, it can be significantly more efficient to employ a suitable block variant of the Krylov-subspace method that generates iterates for all the multiple systems simultaneously.

Recently, Freund and Malhotra [11] have proposed the BL-QMR algorithm, which is a block variant of QMR, for the iterative solution of non-Hermitian linear systems with multiple right-hand sides. A crucial issue for the development of robust block variants of Krylov-subspace iterations is the necessity to incorporate deflation techniques to detect and delete linearly and almost linearly dependent vectors in the block Krylov sequences that underlie the block iteration. A key feature of BL-QMR is that it has a simple, but effective, built-in deflation procedure. For the case of complex symmetric coefficient matrices, the BL-QMR method can be simplified to exploit this symmetry and the resulting algorithm only requires approximately half of the work and storage needed for the general non-Hermitian case. Similarly, when the coefficient matrix is symmetric with respect to a bilinear form induced by a certain matrix $\mathbf{J}$, the BL-QMR algorithm can be simplified to obtain important savings. We remark that, for the special case of complex symmetric matrices, Boyse and Seidl [4] also developed a block variant of QMR, and applied it to finite-element models of Maxwell's equations. However, the algorithm in [4] is significantly different from our complex symmetric BL-QMR method, in that it cannot handle deflation and it also uses a different basis for the underlying block Krylov sequences.

The purpose of this paper is twofold. First, we describe the $\mathbf{J}$-symmetric variant of

BL-QMR, which includes the complex symmetric form of BL-QMR as a special case. Then, we employ various preconditioners in conjunction with the complex symmetric BL-QMR method, and conduct numerical tests to study their performance for multiple right-hand side problems in acoustics.

The remainder of this paper is organized as follows. In Section 2, we summarize the governing partial differential equations for time-harmonic structural acoustics, the finite-element discretization of these equations, and the resulting complex symmetric matrix problem. In Section 3, we describe the general form of block Krylov-subspace iterative methods, and then review the basic structure of the BL-QMR method. In Section 4, we sketch the special version of BL-QMR that exploits complex symmetry. Section 5 provides a summary of the preconditioners that we used in conjunction with BL-QMR. In Section 6, we report results of our numerical tests to illustrate the typical convergence behavior of BL-QMR applied to multiple radiation and scattering problems in structural acoustics, to identify appropriate preconditioners for these problems, and to demonstrate the importance of deflation in block Krylov-subspace methods. Finally, we make some concluding remarks in Section 7.

Throughout this paper, all vectors and matrices are allowed to have real or complex entries. Furthermore, we use the following notation. As usual, $\overline{\mathbf{M}} = \left[\overline{m_{jk}}\right]$, $\mathbf{M}^T = \left[m_{kj}\right]$, and $\mathbf{M}^H = \overline{\mathbf{M}}^T = \left[\overline{m_{kj}}\right]$ denote the complex conjugate, transpose, and the conjugate transpose, respectively, of the matrix $\mathbf{M} = \left[m_{jk}\right]$. The vector norm $\|\mathbf{x}\| := \sqrt{\mathbf{x}^H\mathbf{x}}$ is always the Euclidean norm, and $\|\mathbf{M}\| := \max_{\|\mathbf{x}\|=1} \|\mathbf{M}\mathbf{x}\|$ is the corresponding matrix norm. We denote the complex conjugate of a function $f(x)$ by an overbar, i.e., $\overline{f(x)}$. In the case of a bounded domain Ω with boundary $\partial\Omega$, however, an overbar denotes the closure of the domain, i.e., $\overline{\Omega} = \Omega \cup \partial\Omega$.

2. Exterior problems in structural acoustics

In this section, we summarize the governing differential equations for time-harmonic structural acoustics and describe their finite-element discretization. The formulation of multiple right-hand sides that arise in the resulting matrix problems is also discussed.

2.1. Governing equations

Consider a coupled fluid-structure system consisting of an elastic body submerged in an infinite fluid medium. The elastic body occupies the domain $\Omega_s \subset \mathbb{R}^d$, where d is the number of spatial dimensions. The domain interior to Ω_s is assumed to be evacuated, see Figure 1. The exterior surface of the elastic body forms the fluid-structure interface Γ_i, where the acoustic pressure loads the structure. The unbounded fluid domain exterior to the fluid-structure interface is truncated by introducing an artificial boundary, denoted by Γ_∞, in order to obtain a finite computational fluid domain, denoted by Ω_f.

Assuming harmonic time dependence of the form $e^{-\mathrm{i}\omega t}$, where $\mathrm{i} = \sqrt{-1}$ and ω is the angular frequency, the exterior structural acoustics problem in the finite computational domain $\Omega_s \cup \Omega_f$ can be stated as follows. The problem is to find the displacement vector $\mathbf{u}$

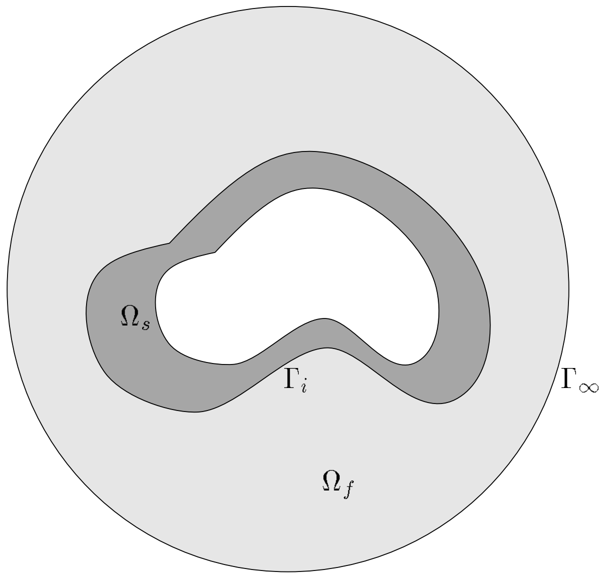


Fig. 1. Computational domain for the exterior acoustics problem.

and the acoustic pressure p satisfying the governing differential equations,

$$\mathcal{L}_s \mathbf{u} = \mathbf{f}_s \qquad \text{in } \Omega_s, \tag{1}$$

$$-\nabla^2 p - k_0^2 p = f_f \qquad \text{in } \Omega_f, \tag{2}$$

together with the Dirichlet and Neumann boundary conditions,

$$\{\mathbf{u}\}_i = \{\mathbf{g}_s\}_i \qquad \text{on } \Gamma^s_{g_i}, \quad i = 1, 2, \ldots, n_{\text{dof}}, \tag{3}$$

$$\{\mathcal{B}\mathbf{u}\}_i = \{\mathbf{h}_s\}_i \qquad \text{on } \Gamma^s_{h_i}, \quad i = 1, 2, \ldots, n_{\text{dof}}, \tag{4}$$

$$p = g_f \qquad \text{on } \Gamma^f_g, \tag{5}$$

$$\nabla p \cdot \mathbf{n} = h_f \qquad \text{on } \Gamma^f_h, \tag{6}$$

and the continuity conditions,

$$\mathcal{B}\mathbf{u} = -p\mathbf{n} \qquad \text{on } \Gamma_i, \tag{7}$$

$$\nabla p \cdot \mathbf{n} = \rho_0 \omega^2 \mathbf{u} \cdot \mathbf{n} \qquad \text{on } \Gamma_i. \tag{8}$$

In the above equations, k_0 is the acoustic wave number, ρ_0 is the density of the fluid, and $\mathbf{n}$ denotes the unit outward (from Ω_f) normal vector at Γ_i. Note that $k_0 = \omega / c_0$, where c_0 is the speed of sound in the fluid medium. The differential operator $\mathcal{L}_s$ in (1) represents the governing differential equations of the structure, which may be modeled using equations of linear elasticity (see, e.g., [1]) or, under simplifying hypotheses, using shell, plate, or beam theory. Accordingly, n_{dof} denotes the number of degrees of freedom of the structural model. The differential operator $\mathcal{B}$ in (4) and (7) is employed to enforce natural boundary conditions on $\mathbf{u}$; clearly, the definition of $\mathcal{B}$ depends on the choice of $\mathcal{L}_s$. The boundary regions Γ_g^f and Γ_h^f in (5)–(6) constitute a non-overlapping partition of Γ_i,

$$\Gamma_i = \overline{\Gamma_g^f \cup \Gamma_h^f}.$$

Similarly, the boundary of the structural domain $\partial\Omega_s$ is assumed to admit partitions

$$\partial\Omega_s = \overline{\Gamma_{g_i}^s \cup \Gamma_{h_i}^s}, \quad i = 1, 2, \ldots, n_{\text{dof}}.$$

Furthermore, $\mathbf{f}_s$, f_f, $\mathbf{g}_s$, g_f, $\mathbf{h}_s$, and h_f denote the external loads, Dirichlet data, and Neumann data prescribed on the respective domains in (1)–(6).

In addition to the equations (5)–(8) at the interface boundary Γ_i, the acoustic pressure in the infinite fluid domain also needs to satisfy the Sommerfeld radiation condition,

$$\lim_{r\to\infty} r^{\frac{d-1}{2}} \left(\frac{\partial p}{\partial r} - \mathrm{i}k_0 p \right) = 0.$$

Correspondingly, for the exterior problem restricted to the finite computational domain Ω_f, a non-reflecting boundary condition needs to be imposed on the fluid truncation boundary Γ_∞ in order to eliminate spurious reflection of waves. Here, we employ the Dirichlet-to-Neumann (DtN) condition [18] that relates the acoustic pressure on the truncation boundary to its normal derivative through a linear map $S(p) : \mathbb{C} \mapsto \mathbb{C}$, i.e.,

$$\nabla p \cdot \mathbf{n} = S(p) \qquad \text{on } \Gamma_\infty. \tag{9}$$

For example, in the case of two-dimensional problems with circular radiation boundaries Γ_∞, the function $S(p)$ is given by

$$S(p) := \sum_{n=0}^{\infty} \alpha_n \frac{k_0 H_n'(k_0 R)}{2\pi H_n(k_0 R)} \int_0^{2\pi} \cos n(\theta - \hat{\theta})\, p(R, \hat{\theta})\, \mathrm{d}\hat{\theta}, \tag{10}$$

see [18]. In (10), R denotes the radius of the circular radiation boundary Γ_∞, θ is the angular coordinate, $H_n(\cdot)$ denotes the n-th order Hankel function of the first kind and $H_n'(\cdot)$ its derivative, and

$$\alpha_n := \begin{cases} 1, & \text{if } n = 0, \\ 2, & \text{if } n > 0, \end{cases}$$

is the so-called Neumann constant.

In the numerical examples presented in Section 6, the elastic body is modeled using Timoshenko beam theory. In this case, $n_{\text{dof}} = 2$, and the displacement field is given by $\mathbf{u} = [\, u_1 \quad u_2 \,]^T$, where u_1 and u_2, respectively, denote the transverse displacement and the independent rotational degrees of freedom. Furthermore, the structural differential operator in (1) is given by

$$\mathcal{L}_s := \begin{bmatrix} -\kappa GA \frac{\partial^2}{\partial x^2} - \rho\omega^2 A & \kappa GA \frac{\partial}{\partial x} \\ -\kappa GA \frac{\partial}{\partial x} & \kappa GA - EI \frac{\partial^2}{\partial x^2} - \rho\omega^2 I \end{bmatrix}, \tag{11}$$

where κ is the shear correction factor, G is the shear modulus, E is Young's modulus, ρ is the structural mass density, A denotes the cross-sectional area of the beam, and I denotes its moment of inertia. Correspondingly, the boundary operator $\mathcal{B}$ in (4) and (7) is given by

$$\mathcal{B} := \begin{bmatrix} \kappa GA\frac{\partial}{\partial x} & -\kappa GA \\ 0 & -EI\frac{\partial}{\partial x} \end{bmatrix}.$$

2.2. Finite-element discretization

First, we define the appropriate function spaces for the structural and fluid variables. Let $L_2(\Omega)$ denote the space of square integrable functions on Ω, and $H^m(\Omega)$ be the usual Sobolev space of functions with all derivatives up to the m-th derivative in $L_2(\Omega)$. We denote the space of trial solutions and weighting functions for the structural displacement vector $\mathbf{u}$ that satisfies (1) (with $\mathcal{L}_s$ from (11)) by

$$\begin{aligned} \mathcal{S}_{\mathbf{u}} &:= \{\, \mathbf{u} \mid u_i \in H^1(\Omega_s),\ u_i = g_{s_i} \text{ on } \Gamma^s_{g_i},\ i = 1, 2, \ldots, n_{\mathrm{dof}} \,\} \\ \text{and} \quad \mathcal{V}_{\mathbf{u}} &:= \{\, \hat{\mathbf{u}} \mid \hat{u}_i \in H^1(\Omega_s),\ \hat{u}_i = 0 \text{ on } \Gamma^s_{g_i},\ i = 1, 2, \ldots, n_{\mathrm{dof}} \,\}, \end{aligned} \tag{12}$$

respectively. Similarly, let

$$\begin{aligned} \mathcal{S}_p &:= \{\, p \mid p \in H^1(\Omega_f),\ p = g_f \text{ on } \Gamma^f_g \,\} \\ \text{and} \quad \mathcal{V}_p &:= \{\, \hat{p} \mid \hat{p} \in H^1(\Omega_f),\ \hat{p} = 0 \text{ on } \Gamma^f_g \,\} \end{aligned}$$

denote the function spaces for trial solutions and weighting functions, respectively, for the acoustic pressure p.

The variational or weak form of the problem can now be expressed as follows. Given prescribed data, i.e., $\mathbf{f}_s$, $\mathbf{g}_s$, $\mathbf{h}_s$, f_f, g_f, and h_f, find $\mathbf{u} \in \mathcal{S}_{\mathbf{u}}$ and $p \in \mathcal{S}_p$ such that the relations

$$A_f(\hat{p}, p) + (\hat{p}\mathbf{n}, \rho_0\omega^2\mathbf{u})_{\Gamma_i} + (\hat{p}, S(p))_{\Gamma_\infty} = L_f(\hat{p}), \tag{13}$$

$$A_s(\hat{\mathbf{u}}, \mathbf{u}) + (\hat{\mathbf{u}}, \mathbf{n}p)_{\Gamma_i} = L_s(\hat{\mathbf{u}}), \tag{14}$$

are satisfied for all functions $\hat{\mathbf{u}} \in \mathcal{V}_{\mathbf{u}}$ and $\hat{p} \in \mathcal{V}_p$. In (13)–(14),

$$\begin{aligned} A_f(\hat{p}, p) &:= (\nabla\hat{p}, \nabla p)_{\Omega_f} - k_0^2(\hat{p}, p)_{\Omega_f}, \\ L_f(\hat{p}) &:= (\hat{p}, f_f)_{\Omega_f} + (\hat{p}, h_f)_{\Gamma^f_h}, \\ L_s(\hat{\mathbf{u}}) &:= (\hat{\mathbf{u}}, \mathbf{f}_s)_{\Omega_s} + \sum_{i=1}^{n_{\mathrm{dof}}} (\hat{u}_i, h_{s_i})_{\Gamma^s_{h_i}}. \end{aligned} \tag{15}$$

Furthermore, $(\cdot,\cdot)_\Omega$ denotes the $L_2(\Omega)$ inner product. Note that $(\cdot,\cdot)$ is conjugate linear in the first argument and linear in the second. For the case when $\mathcal{L}_s$ is given by (11), the structural operator $A_s(\cdot,\cdot)$ in (14) can be written as

$$A_s(\hat{\mathbf{u}},\mathbf{u}) := \int_{\Omega_s} \left[\overline{\left(\frac{\partial \hat{u}_1}{\partial x} - \hat{u}_2\right)} \kappa G A \left(\frac{\partial u_1}{\partial x} - u_2\right) + \overline{\frac{\partial \hat{u}_2}{\partial x}} EI \frac{\partial u_2}{\partial x}\right] \mathrm{d}x$$
$$- \int_{\Omega_s} \rho\omega^2 \left(\overline{\hat{u}}_1 A u_1 + \overline{\hat{u}}_2 I u_2\right) \mathrm{d}x. \tag{16}$$

Next, we introduce a finite-element discretization of (13)–(14). Let $\mathcal{S}^h_{\mathbf{u}} \subset \mathcal{S}_{\mathbf{u}}$, $\mathcal{V}^h_{\mathbf{u}} \subset \mathcal{V}_{\mathbf{u}}$, $\mathcal{S}^h_p \subset \mathcal{S}_p$, and $\mathcal{V}^h_p \subset \mathcal{V}_p$ be finite-element spaces of continuous piecewise linear polynomials. Using the same interpolation functions for the trial solution and weighting functions, the Galerkin formulation of the variational problem (13)–(14) can now be stated as follows. Given prescribed data, find $\mathbf{u}^h \in \mathcal{S}^h_{\mathbf{u}}$ and $p^h \in \mathcal{S}^h_p$ such that the relations

$$A_s(\hat{\mathbf{u}}^h, \mathbf{u}^h) + (\hat{\mathbf{u}}^h, \mathbf{n}p^h)_{\Gamma_i} = L_s(\hat{\mathbf{u}}^h), \tag{17}$$

$$A_f(\hat{p}^h, p^h) + (\hat{p}^h\mathbf{n}, \rho_0\omega^2\mathbf{u}^h)_{\Gamma_i} + (\hat{p}^h, S(p^h))_{\Gamma_\infty} = L_f(\hat{p}^h), \tag{18}$$

are satisfied for all $\hat{\mathbf{u}}^h \in \mathcal{V}^h_{\mathbf{u}}$ and $\hat{p}^h \in \mathcal{V}^h_p$.

Finally, by introducing the finite-element nodal basis functions in (17)–(18) in the usual way (see, e.g., [17, Chapter 2]), we obtain the matrix problem

$$\mathbf{K}\mathbf{d} = \mathbf{f}, \tag{19}$$

where

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_s - \omega^2\mathbf{M}_s & \mathbf{C}^T \\ \rho_0\omega^2\mathbf{C} & \mathbf{K}_f - k_0^2\mathbf{M}_f + \mathbf{K}_{\mathrm{DtN}} \end{bmatrix}, \quad \mathbf{d} = \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix}, \quad \text{and} \quad \mathbf{f} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \end{bmatrix}. \tag{20}$$

The vector $\mathbf{d} \in \mathbb{C}^N$ contains structural and fluid degrees of freedom, which are denoted by $\mathbf{u}$ and $\mathbf{p}$, respectively, and N is the total number of degrees of freedom. The parts $\mathbf{f}_1$ and $\mathbf{f}_2$ of the right-hand side vector $\mathbf{f}$ in (19) arise from the operators $L_s(\cdot)$ and $L_f(\cdot)$ in (17)–(18). In (20), $\mathbf{K}_s$ and $\mathbf{M}_s$ are the structural stiffness and mass matrices that arise from (16), and $\mathbf{K}_f$ and $\mathbf{M}_f$ are fluid matrices that result from the $H^1(\Omega_f)$ and $L_2(\Omega_f)$ parts, respectively, of the operator $A_f(\cdot,\cdot)$ in (18). The matrices $\mathbf{C}$ and $\mathbf{K}_{\mathrm{DtN}}$, respectively, arise from inner products defined on Γ_i and Γ_∞ in (18). We remark that the matrices $\mathbf{K}_s$, $\mathbf{M}_s$, $\mathbf{K}_f$, and $\mathbf{M}_f$ are real symmetric, while $\mathbf{K}_{\mathrm{DtN}}$ is complex symmetric. Therefore, $\mathbf{K}$ in (20) can be turned into a complex symmetric matrix by simply scaling the fluid equations by $1/\rho_0\omega^2$. In the sequel, we solve the complex symmetric formulation of (19) for nontrivial frequencies $\omega \neq 0$, i.e., we use

$$\mathbf{K} := \begin{bmatrix} \mathbf{K}_s - \omega^2\mathbf{M}_s & \mathbf{C}^T \\ \mathbf{C} & \frac{1}{\rho_0\omega^2}\left(\mathbf{K}_f - k_0^2\mathbf{M}_f + \mathbf{K}_{\mathrm{DtN}}\right) \end{bmatrix}, \tag{21}$$

together with $\mathbf{f} = \begin{bmatrix}\mathbf{f}_1^T & (1/\rho_0\omega^2)\mathbf{f}_2^T\end{bmatrix}^T$. Exploiting complex symmetry in non-Hermitian systems enables very significant savings, both in storage and computations, as it results in important simplifications in Lanczos-type Krylov-subspace iterations, such as the QMR algorithm. We will discuss this issue further in Section 4.

2.3. Multiple right-hand sides

In many situations, the matrix problem (19) needs to be solved repeatedly for different right-hand sides $\mathbf{f}^{(j)}$, $j = 1, 2, \ldots, m$, but with the same coefficient matrix $\mathbf{K}$. For instance, multiple load vectors $\mathbf{f}^{(j)}$ arise in radiation problems due to multiple load cases, which correspond to different values of the prescribed data $\mathbf{f}_s$, $\mathbf{g}_s$, $\mathbf{h}_s$, f_f, g_f, or h_f. Furthermore, in scattering problems, multiple load vectors also result when multiple angles of incidence of an incoming plane wave need to be considered.

Next, we review the formulation of multiple right-hand side problems in scattering. In this case, the total acoustic pressure can be represented as

$$p_{\text{tot}} = p_{\text{inc}} + p_{\text{sc}},$$

where p_{inc} is the known pressure field due to an incident plane wave and p_{sc} is the unknown scattered pressure. Using the fact that p_{inc} satisfies the homogeneous Helmholtz equation, the governing equations for the coupled problem can be rewritten in terms of p_{sc} and $\mathbf{u}$. Thereafter, it is easy to verify that the variational equations (13) and (14) also hold for $\mathbf{u}$ and $p = p_{\text{sc}}$, with the operators $L_s(\cdot)$ and $L_f(\cdot)$ replaced by

$$\begin{aligned} \widehat{L}_s(\hat{\mathbf{u}}) &:= L_s(\hat{\mathbf{u}}) + (\hat{\mathbf{u}}, \mathbf{n}p_{\text{inc}})_{\Gamma_i} \\ \text{and} \quad \widehat{L}_f(\hat{p}) &:= L_f(\hat{p}) - (\hat{p}, \nabla p_{\text{inc}} \cdot \mathbf{n})_{\Gamma_h^f}, \end{aligned} \tag{22}$$

respectively. Note that the modified operators $\widehat{L}_s(\cdot)$ and $\widehat{L}_f(\cdot)$ include additional terms that contain p_{inc} and, therefore, give rise to multiple right-hand sides corresponding to various angles of the incoming plane wave.

We remark that, for both radiation and scattering problems, all the different right-hand sides are available simultaneously. Therefore, the solution of the m linear systems,

$$\mathbf{K}\mathbf{d}^{(j)} = \mathbf{f}^{(j)}, \quad j = 1, 2, \ldots, m,$$

is equivalent to the solution of the block system of linear equations

$$\mathbf{K}\mathbf{D} = \mathbf{F}, \tag{23}$$

where

$$\mathbf{D} = \begin{bmatrix} \mathbf{d}^{(1)} & \mathbf{d}^{(2)} & \cdots & \mathbf{d}^{(m)} \end{bmatrix} \quad \text{and} \quad \mathbf{F} = \begin{bmatrix} \mathbf{f}^{(1)} & \mathbf{f}^{(2)} & \cdots & \mathbf{f}^{(m)} \end{bmatrix}.$$

The BL-QMR algorithm is an iterative method tailored to the solution of block systems of the type (23).

2.4. Preconditioned linear system

Krylov-subspace iterations are almost always combined with a suitable preconditioning technique. The basic idea is to construct a preconditioning matrix that constitutes a "simple" approximation to the coefficient matrix of the linear system to be solved, and then to apply the iterative method to the preconditioned linear system. More precisely, let

$\mathbf{M} = \mathbf{M}_1\mathbf{M}_2$ be a a suitable preconditioning matrix for the matrix $\mathbf{K}$ of (23). Then, the Krylov-subspace iteration is applied to the preconditioned matrix problem

$$\mathbf{A}\mathbf{X} = \mathbf{B}, \tag{24}$$

where

$$\mathbf{A} = \mathbf{M}_1^{-1}\mathbf{K}\mathbf{M}_2^{-1}, \quad \mathbf{X} = \mathbf{M}_2\mathbf{D}, \quad \text{and} \quad \mathbf{B} = \mathbf{M}_1^{-1}\mathbf{F}, \tag{25}$$

rather than to the original system (23). Clearly, the linear system (24) is equivalent to (23).

In the sequel, we will discuss the iterative solution of the preconditioned matrix problem (24). The construction of suitable preconditioning matrices $\mathbf{M}$ will be discussed in Section 5.

3. Simultaneous solution of multiple linear systems

In this section, we first describe the general form of block Krylov-subspace iterative methods for the simultaneous solution of linear systems with multiple right-hand sides. Then, we review the basic structure of the BL-QMR method.

3.1. Block Krylov-subspace iterations

We are concerned with the solution of multiple linear systems (24), where $\mathbf{A} \in \mathbb{C}^{N\times N}$ is assumed to be a nonsingular matrix, and

$$\mathbf{B} = \begin{bmatrix} \mathbf{b}^{(1)} & \mathbf{b}^{(2)} & \cdots & \mathbf{b}^{(m)} \end{bmatrix} \in \mathbb{C}^{N\times m}$$

is a block of m right-hand sides. An iterative scheme for solving (24) is said to be a *block Krylov-subspace method* if it generates a sequence of block iterates

$$\mathbf{X}_\mu = \begin{bmatrix} \mathbf{x}_\mu^{(1)} & \mathbf{x}_\mu^{(2)} & \cdots & \mathbf{x}_\mu^{(m)} \end{bmatrix} \in \mathbb{C}^{N\times m}, \quad \mu = 0, 1, \ldots, \tag{26}$$

such that, for each $j = 1, 2, \ldots, m$,

$$\mathbf{x}_\mu^{(j)} \in \mathbf{x}_0^{(j)} + \mathcal{K}_\mu^{\mathrm{dl}}(\mathbf{A}, \mathbf{R}). \tag{27}$$

Here, the vectors $\mathbf{x}_0^{(j)} \in \mathbb{C}^N$, $j = 1, 2, \ldots, m$, are arbitrary initial guesses for the corresponding exact solutions $\mathbf{x}^{(j)} = \mathbf{A}^{-1}\mathbf{b}^{(j)}$ of (24), $\mathbf{R} = \mathbf{R}_0 = \mathbf{B} - \mathbf{A}\mathbf{X}_0$ is the block of initial residual vectors, and $\mathcal{K}_\mu^{\mathrm{dl}}(\mathbf{A}, \mathbf{R})$ denotes the μ-th *block Krylov subspace* of $\mathbb{C}^N$ spanned by the first μ linearly independent vectors in the block Krylov sequence

$$\mathbf{R}, \mathbf{A}\mathbf{R}, \mathbf{A}^2\mathbf{R}, \ldots, \mathbf{A}^{k-1}\mathbf{R}, \ldots . \tag{28}$$

Next, we elaborate on this notion of block Krylov subspaces, and make it more precise. In general, the block Krylov sequence (28) induced by the matrix $\mathbf{A}$ and a block of m arbitrary starting vectors $\mathbf{R}$, may have linearly dependent or almost linearly dependent vectors. In particular, note that each block $\mathbf{A}^{k-1}\mathbf{R}$, $k = 1, 2, \ldots$, in (28) consists of m vectors of length N, and thus at most N of the vectors in (28) can be linearly independent.

Hence, by scanning the vectors in (28) from left to right, and deleting each vector that is either linearly dependent or almost linearly dependent on previous vectors, we obtain the so-called *deflated* block Krylov sequence

$$\mathbf{R}^{(1)}, \mathbf{A}\mathbf{R}^{(2)}, \mathbf{A}^2\mathbf{R}^{(3)}, \ldots, \mathbf{A}^{k-1}\mathbf{R}^{(k)}, \ldots . \tag{29}$$

Here, for each $k = 1, 2, \ldots$, the block $\mathbf{R}^{(k)}$ is a submatrix of $\mathbf{R}^{(k-1)}$; for $k = 1$, we set $\mathbf{R}^{(0)} = \mathbf{R}$. Then, clearly, the first μ vectors in the deflated Krylov sequence (29) span the μ-th block Krylov subspace $\mathcal{K}_\mu^{\mathrm{dl}}(\mathbf{A}, \mathbf{R})$. We refer to the process of dropping linearly dependent (respectively, almost linearly dependent) vectors from (28) as *exact deflation* (respectively, *inexact deflation*).

In order to design an actual block Krylov-subspace method for generating iterates of the form (26)–(27), we need to address two key issues. The first one is the construction of suitable basis vectors that span the block Krylov subspaces $\mathcal{K}_\mu^{\mathrm{dl}}(\mathbf{A}, \mathbf{R})$. The second issue is the choice of the block iterates $\mathbf{X}_\mu$.

The BL-QMR method proposed in [11] is a block Krylov-subspace algorithm for solving multiple systems (24) with non-Hermitian coefficient matrices $\mathbf{A}$. The BL-QMR method uses a novel Lanczos-type process [2] to generate basis vectors for $\mathcal{K}_\mu^{\mathrm{dl}}(\mathbf{A}, \mathbf{R})$, and its block iterates are characterized by a quasi-minimal residual property. Next, we review these two stages in the general BL-QMR method.

3.2. Lanczos basis vectors

Given the matrices $\mathbf{A} \in \mathbb{C}^{N \times N}$ and $\mathbf{R} \in \mathbb{C}^{N \times m}$, the Lanczos-type algorithm described in [2] uses the columns of $\mathbf{R}$ as starting vectors and generates a sequence of basis vectors, $\mathbf{v}_1, \mathbf{v}_2, \ldots$, for the block Krylov subspaces $\mathcal{K}_j^{\mathrm{dl}}(\mathbf{A}, \mathbf{R})$, i.e.,

$$\mathrm{span}\{\, \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_j \,\} = \mathcal{K}_j^{\mathrm{dl}}(\mathbf{A}, \mathbf{R}) \quad \text{for} \quad j = 1, 2, \ldots . \tag{30}$$

In addition to the matrices $\mathbf{A}$ and $\mathbf{R}$, the Lanczos-type process also requires a second, in general arbitrary, block of vectors $\mathbf{L} \in \mathbb{C}^{N \times p}$, and it builds a second sequence of vectors, $\mathbf{w}_1, \mathbf{w}_2, \ldots$, that span the block Krylov subspaces $\mathcal{K}_j^{\mathrm{dl}}(\mathbf{A}^T, \mathbf{L})$, i.e.,

$$\mathrm{span}\{\, \mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_j \,\} = \mathcal{K}_j^{\mathrm{dl}}(\mathbf{A}^T, \mathbf{L}) \quad \text{for} \quad j = 1, 2, \ldots . \tag{31}$$

We stress that $p \geq 1$, the number of columns of $\mathbf{L}$, can be an arbitrary integer, and in particular, p can be chosen independently from m. Analogous to $\mathcal{K}_j^{\mathrm{dl}}(\mathbf{A}, \mathbf{R})$, in (31), $\mathcal{K}_j^{\mathrm{dl}}(\mathbf{A}^T, \mathbf{L})$ denotes the j-th block Krylov subspace of $\mathbb{C}^N$ spanned by the first j linearly independent vectors in the deflated block Krylov sequence generated by the matrices $\mathbf{A}^T$ and $\mathbf{L}$. Further, in the Lanczos-type process, the vectors in the $\mathbf{v}$ and $\mathbf{w}$ sequences are constructed to be *biorthogonal*, i.e.,

$$\mathbf{w}_k^T \mathbf{v}_l = 0 \quad \text{for all} \quad k \neq l = 1, 2, \ldots . \tag{32}$$

We remark that, for the case $m = p = 1$ of single starting vectors, this Lanczos-type process reduces to the classical nonsymmetric Lanczos algorithm [19]. Just as in the case of this latter algorithm, it may not always be possible to enforce the biorthogonality condition (32) in the Lanczos-type process for general m and p, as it would lead to so-called *breakdowns* or *near-breakdowns* of the algorithm due to division by zero or near-zero numbers. Such breakdowns and near-breakdowns can be avoided by performing certain *look-ahead* steps; we refer to [2] for further details. In this paper, for the sake of brevity, we assume that (32) can be enforced without any breakdowns.

The recurrences used in the Lanczos-type process to compute the first n basis vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ can be summarized compactly as follows:

$$\mathbf{A}\mathbf{V}^{(\mu)} = \mathbf{V}^{(n)}\mathbf{T}_e^{(\mu)} + \mathbf{V}_{\mathrm{dl}}^{(\mu)}, \quad \text{if} \quad \mu := n - m_{\mathrm{cr}} > 0. \tag{33}$$

Here, m_{cr} denotes the reduced size of the current block (the one from which $\mathbf{v}_n$ is constructed) in the deflated block Krylov sequence (29), $\mathbf{V}^{(n)}$ is an $N \times n$ matrix given by

$$\mathbf{V}^{(n)} := \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_n \end{bmatrix}, \tag{34}$$

and $\mathbf{T}_e^{(\mu)}$ is an $n \times \mu$ matrix whose elements are coefficients of the recursions in the Lanczos-type algorithm. In the absence of any deflations in both the $\mathbf{v}$ and $\mathbf{w}$ sequences, $\mathbf{T}_e^{(\mu)}$ is a banded matrix with lower bandwidth $m+1$ and upper bandwidth $p+1$. Each deflation step in the $\mathbf{v}$ sequence reduces the lower bandwidth of $\mathbf{T}_e^{(\mu)}$ by one, while a deflation in the $\mathbf{w}$ sequence reduces its upper bandwidth by one. In the case of an "inexact" deflation in the $\mathbf{w}$ sequence at iteration μ, additional nonzero elements are also introduced in $\mathbf{T}_e^{(\mu)}$ in row μ and columns beyond the $p+1$ upper diagonals. Furthermore, the matrix $\mathbf{V}_{\mathrm{dl}}^{(\mu)}$ in (33) consists of mostly zero column vectors and possibly a few nonzero columns, which are just the deflated $\mathbf{v}$ vectors.

Notice that (33) only holds for $\mu > 0$, or equivalently for $n > m_1$, where m_1 is the size of the first block in the sequence (29). Additionally, the recurrences for the initial block of Lanczos vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{m_1}$ can be stated as

$$\mathbf{V}^{(m_1)}\boldsymbol{\rho} + \mathbf{V}_{\mathrm{dl}}^{(0)} = \mathbf{R}. \tag{35}$$

Here, $\boldsymbol{\rho} \in \mathbb{C}^{m_1 \times m}$, and the matrix $\mathbf{V}_{\mathrm{dl}}^{(0)} \in \mathbb{C}^{N \times m}$ consists of zero vectors and deflated $\mathbf{v}$ vectors. More precisely, $\mathbf{V}_{\mathrm{dl}}^{(0)}$ is different from zero if, and only if, at least one column of the starting block $\mathbf{R}$ is deflated.

3.3. The BL-QMR method

We now discuss the choice of the block iterates (26)–(27) in the BL-QMR method. In order to construct basis vectors that span the block Krylov subspaces in (27), we choose $\mathbf{R} = \mathbf{R}_0$ as one of the starting blocks for the Lanczos-type algorithm. Since the generated vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$, $n = 1, 2, \ldots$, satisfy (30), and using (34), we have

$$\mathcal{K}_n^{\mathrm{dl}}(\mathbf{A}, \mathbf{R}_0) = \{\, \mathbf{V}^{(n)}\mathbf{z} \mid \mathbf{z} \in \mathbb{C}^n \,\}. \tag{36}$$

By (36), all possible block iterates (26)–(27) can be represented in the form

$$\mathbf{X}_\mu = \mathbf{X}_0 + \mathbf{V}^{(\mu)}\mathbf{Z}, \quad \mathbf{Z} \in \mathbb{C}^{\mu \times m}, \tag{37}$$

where $\mathbf{Z}$ is a matrix of, as yet, undetermined coefficients.

After each deflation in the $\mathbf{v}$ sequence, the size of the current block in the corresponding deflated block Krylov sequence (29) decreases by one. Additionally, as described in detail in [11], each time a deflation is encountered in the $\mathbf{v}$ sequence, one of the linear systems in (24) can be dropped from subsequent BL-QMR iterations. Thereafter, once convergence of the remaining linear systems in (24) is achieved, the solution of all such deflated systems can be recovered from the solutions of the converged linear systems. Therefore, in case of a deflation, one of the columns from the block matrices $\mathbf{X}_\mu$, $\mathbf{X}_0$, and $\mathbf{Z}$ in (37) can be dropped and only the remaining iterates need to be updated. Let $\mathbf{X}_\mu^{\mathrm{cr}}$ denote the current block of approximate solution vectors containing the m_{cr} columns of $\mathbf{X}_\mu$ that were retained after all deflations up to the μ-th iteration. Then, for the current block iterate $\mathbf{X}_\mu^{\mathrm{cr}}$, we can rewrite (37) as

$$\mathbf{X}_\mu^{\mathrm{cr}} = \mathbf{X}_0^{\mathrm{cr}} + \mathbf{V}^{(\mu)}\mathbf{Z}, \quad \mathbf{Z} \in \mathbb{C}^{\mu \times m_{\mathrm{cr}}}. \tag{38}$$

Using (38), (33), and (35), the residual block $\mathbf{R}_\mu^{\mathrm{cr}}$ corresponding to $\mathbf{X}_\mu^{\mathrm{cr}}$ can be represented as follows:

$$\begin{aligned} \mathbf{R}_\mu^{\mathrm{cr}} &= \mathbf{R}_0^{\mathrm{cr}} - \mathbf{V}^{(n)}\mathbf{T}_e^{(\mu)}\mathbf{Z} - \mathbf{V}_{\mathrm{dl}}^{(\mu)}\mathbf{Z} \\ &= \mathbf{V}^{(n)}\left(\begin{bmatrix} \boldsymbol{\rho}^{\mathrm{cr}} \\ \mathbf{0} \end{bmatrix} - \mathbf{T}_b^{(\mu)}\mathbf{Z}\right) - \widehat{\mathbf{V}}_{\mathrm{dl}}^{(\mu)}\mathbf{Z}. \end{aligned} \tag{39}$$

Here, the matrix $\boldsymbol{\rho}^{\mathrm{cr}} \in \mathbb{C}^{m_1 \times m_{\mathrm{cr}}}$ consists of those columns of $\boldsymbol{\rho}$ in (35) that correspond to columns of $\mathbf{X}_\mu$ retained in $\mathbf{X}_\mu^{\mathrm{cr}}$. Furthermore, $\mathbf{T}_b^{(\mu)}$ is an $n \times \mu$ banded matrix obtained from $\mathbf{T}_e^{(\mu)}$ by dropping the entries in $\mathbf{T}_e^{(\mu)}$ that result from inexact deflations. We stress that these dropped entries are small, and that

$$\mathbf{T}_b^{(\mu)} = \mathbf{T}_e^{(\mu)} + \mathcal{O}(\mathtt{dtol}),$$

where `dtol` is a suitably chosen small deflation tolerance. Finally, the matrix $\widehat{\mathbf{V}}_{\mathrm{dl}}^{(\mu)}$ in (39) contains the nonzero columns of $\mathbf{V}_{\mathrm{dl}}^{(\mu)}$ together with additional columns arising from the product in (33) corresponding to entries of $\mathbf{T}_e^{(\mu)}$ that are not included in $\mathbf{T}_b^{(\mu)}$.

Ideally, we would like to choose the free parameter matrix $\mathbf{Z}$ in (39) such that $\|\mathbf{R}_\mu\|$ is minimal. However, such a choice would result in work and storage requirements that become prohibitive as the iteration index μ increases. To obtain a computational procedure with low and constant work and storage requirements per iteration, Freund and Nachtigal [12, 8] introduced the concept of quasi-minimization of the residual norm. Adapting the QMR approach to the block case, Freund and Malhotra [11] proposed to choose $\mathbf{Z} = \mathbf{Z}^{(\mu)} \in \mathbb{C}^{\mu \times m_{\mathrm{cr}}}$ as the solution of the matrix least-squares problem

$$\left\| \begin{bmatrix} \boldsymbol{\rho}^{\mathrm{cr}} \\ \mathbf{0} \end{bmatrix} - \mathbf{T}_b^{(\mu)}\mathbf{Z}^{(\mu)} \right\| = \min_{\mathbf{Z} \in \mathbb{C}^{\mu \times m_{\mathrm{cr}}}} \left\| \begin{bmatrix} \boldsymbol{\rho}^{\mathrm{cr}} \\ \mathbf{0} \end{bmatrix} - \mathbf{T}_b^{(\mu)}\mathbf{Z} \right\|. \tag{40}$$

Observe that $\mathbf{Z} = \mathbf{Z}^{(\mu)}$ results in the minimization of the bracketed term in (39), which is just the quasi-minimal residual property of the BL-QMR iterates $\mathbf{X}_\mu$.

The problem (40) is solved using standard techniques [14, Chapter 5] based on a QR factorization

$$\mathbf{T}_b^{(\mu)} = (\mathbf{Q}^{(\mu)})^H \begin{bmatrix} \mathbf{R}^{(\mu)} \\ \mathbf{0} \end{bmatrix}$$

of the banded matrix $\mathbf{T}_b^{(\mu)}$ into a unitary matrix $(\mathbf{Q}^{(\mu)})^H$ and an upper triangular matrix $\mathbf{R}^{(\mu)}$. Since the lower bandwidth of the matrix $\mathbf{T}_b^{(\mu)}$ is at most $m+1$, its QR decomposition can be updated using m Givens rotations at each iteration of the BL-QMR method. The update of the BL-QMR iterates can then be performed by means of the following quantities. First, we define matrices $\mathbf{t}_\mu^{\mathrm{cr}} \in \mathbb{C}^{\mu \times m_{\mathrm{cr}}}$ and $\boldsymbol{\tau}_\mu^{\mathrm{cr}} \in \mathbb{C}^{m_{\mathrm{cr}} \times m_{\mathrm{cr}}}$ by setting

$$\begin{bmatrix} \mathbf{t}_\mu^{\mathrm{cr}} \\ \boldsymbol{\tau}_\mu^{\mathrm{cr}} \end{bmatrix} := \mathbf{Q}^{(\mu)} \begin{bmatrix} \boldsymbol{\rho}^{\mathrm{cr}} \\ \mathbf{0} \end{bmatrix}.$$

It is easy to see that $\mathbf{t}_\mu^{\mathrm{cr}}$ differs from the previous iteration only in an additional row, i.e.,

$$\mathbf{t}_\mu^{\mathrm{cr}} = \begin{bmatrix} \mathbf{t}_{\mu-1}^{\mathrm{cr}} \\ \mathbf{y}_\mu^T \end{bmatrix}, \quad \text{where} \quad \mathbf{y}_\mu^T \in \mathbb{C}^{1 \times m_{\mathrm{cr}}}. \tag{41}$$

We also define a sequence of direction vectors $\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_\mu$, $\mu = 1, 2, \ldots$, by the recursion

$$\mathbf{p}_\mu = \frac{1}{\zeta_\mu} \left(\mathbf{v}_\mu - \sum_{i=i(\mu)}^{\mu-1} \mathbf{p}_i \zeta_{i\mu} \right), \tag{42}$$

where $\zeta_{i\mu}$, $i = i(\mu), i(\mu)+1, \ldots, \mu$, are entries of the μ-th column of $\mathbf{R}^{(\mu)} = \begin{bmatrix} \zeta_{ik} \end{bmatrix}$. We note that $\mu - i(\mu) \leq 2m$, and thus the recurrence (42) involves at most $2m$ terms. Finally, the BL-QMR iterates can be updated by means of the relation $\mathbf{X}_\mu^{\mathrm{cr}} = \mathbf{X}_{\mu-1}^{\mathrm{cr}} + \mathbf{p}_\mu \mathbf{y}_\mu^T$. For a derivation of this update, as well as further implementation details of the BL-QMR method, we refer the reader to [11].

4. The BL-QMR method for J-symmetric systems

The BL-QMR method described in the previous section is an iterative method for solving multiple linear systems (24) with general coefficient matrices $\mathbf{A}$. The main work in the n-th iteration of BL-QMR involves the computation of two matrix-vector products

$$\mathbf{A}\mathbf{r}_{n-m_{\mathrm{cr}}} \quad \text{and} \quad \mathbf{A}^T \mathbf{l}_{n-p_{\mathrm{cr}}},$$

which are required to advance the block Krylov subspaces $\mathcal{K}_n^{\mathrm{dl}}(\mathbf{A}, \mathbf{R})$ and $\mathcal{K}_n^{\mathrm{dl}}(\mathbf{A}^T, \mathbf{L})$. However, in cases when $\mathbf{A}$ is a $\mathbf{J}$-symmetric matrix, i.e.,

$$\mathbf{A}^T \mathbf{J} = \mathbf{J}\mathbf{A} \tag{43}$$

for a given nonsingular matrix $\mathbf{J} \in \mathbb{C}^{N \times N}$, the solution algorithm simplifies (see, e.g., [13]). We note that (43) means that $\mathbf{A}$ is symmetric with respect to the bilinear form $(\mathbf{y}, \mathbf{x})_{\mathbf{J}} := \mathbf{y}^T \mathbf{J} \mathbf{x}$ induced by the matrix $\mathbf{J}$, i.e.,

$$(\mathbf{y}, \mathbf{A}\mathbf{x})_{\mathbf{J}} = (\mathbf{A}\mathbf{y}, \mathbf{x})_{\mathbf{J}}, \quad \text{for any} \quad \mathbf{x}, \mathbf{y} \in \mathbb{C}^N.$$

By exploiting the $\mathbf{J}$-symmetry of $\mathbf{A}$, the work and storage required by the simplified BL-QMR algorithm reduce to approximately half of those required for the general case.

The basic idea is to take advantage of the freedom in choosing the block of starting vectors $\mathbf{L}$ for the $\mathbf{w}$ sequence of Lanczos vectors. Indeed, if we set $\mathbf{L} = \mathbf{JR}$ and use (43), then each of the vectors in the $\mathbf{v}$ and $\mathbf{w}$ sequences are related as follows:

$$\mathbf{w}_i = \xi_i \mathbf{J} \mathbf{v}_i, \quad \text{for all} \quad i = 1, 2, \dots, \tag{44}$$

where $\xi_i \in \mathbb{C}$, $\xi_i \neq 0$, are suitable scaling factors. Therefore, while we construct the right Lanczos vectors, $\mathbf{v}_1, \mathbf{v}_2, \dots$, using (33), the left Lanczos vectors, $\mathbf{w}_1, \mathbf{w}_2, \dots$, can be computed via (44). In the n-th iteration of the resulting $\mathbf{J}$-symmetric BL-QMR method, we now need to compute a matrix-vector product with the matrix $\mathbf{J}$ instead of $\mathbf{A}^T$. So the simplification in the BL-QMR algorithm is only useful in cases where matrix-vector products with $\mathbf{J}$ are cheaper than those with $\mathbf{A}^T$. Clearly, this is true when $\mathbf{J}$ is the identity matrix, i.e., when $\mathbf{A}$ itself is complex symmetric. More generally, consider the case that $\mathbf{A} = \mathbf{M}_1^{-1} \mathbf{K} \mathbf{M}_2^{-1}$ is the preconditioned version of a complex symmetric matrix $\mathbf{K} = \mathbf{K}^T$ with complex symmetric preconditioner $\mathbf{M} = \mathbf{M}_1 \mathbf{M}_2 = \mathbf{M}^T$. One readily verifies that $\mathbf{A}$ is $\mathbf{J}$-symmetric with $\mathbf{J} := \mathbf{M}_1^T \mathbf{M}_2^{-1}$. Typically, $\mathbf{M}_1$ and $\mathbf{M}_2$ are such that products with $\mathbf{J}$ can be computed cheaply. In fact, usually $\mathbf{M}_1$ and $\mathbf{M}_2$ are connected by a relation of the form $\mathbf{M}_1 = \mathbf{M}_2^T \mathbf{D}$, where $\mathbf{D}$ is a diagonal matrix, and then $\mathbf{J} = \mathbf{D}$ is also diagonal. We remark that, for all preconditioners used in our numerical tests in Section 6, the matrix $\mathbf{J}$ is always diagonal.

Next, we sketch the $\mathbf{J}$-symmetric BL-QMR method.

Algorithm 4.1 (Sketch of the J-symmetric BL-QMR method for $\mathbf{AX} = \mathbf{B}$)

INPUT: Matrices $\mathbf{A}, \mathbf{J} \in \mathbb{C}^{N \times N}$ such that $\mathbf{A}$ is $\mathbf{J}$-symmetric.
A matrix $\mathbf{B} \in \mathbb{C}^{N \times m}$.
A deflation tolerance `dtol`.
A breakdown tolerance `btol`.

0) Choose $\mathbf{X}_0 \in \mathbb{C}^{N \times m}$ and set $\mathbf{R} = \mathbf{R}_0 = \mathbf{B} - \mathbf{A}\mathbf{X}_0$.
Set $\mu = -m$ and $m_{\text{cr}} = m$.
Set $\mathcal{I}_v = \emptyset$.

For $n = 1, 2, \dots,$ do:

1) (Build the next Lanczos vector $\mathbf{v}_n$.)

1a) Set $\mu = \mu + 1$. If $\mu = n$, then stop.

1b) Set

$$\mathbf{v} = \begin{cases} \mathbf{r}_{\mu+m}, & \text{if } \mu \leq 0, \\ \mathbf{A}\mathbf{v}_\mu, & \text{otherwise.} \end{cases}$$

1c) Set

$$i_v = \begin{cases} 1, & \text{if } \mu \leq 0, \\ \max\{1, \mu_\mu\}, & \text{otherwise.} \end{cases}$$

Compute

$$t_{i,\mu} = \frac{\mathbf{v}_i^T \mathbf{J} \mathbf{v}}{\delta_i} \quad \text{for all } i \text{ with } i_v \le i \le n-1 \text{ or } i \in \mathcal{I}_v.$$

Set

$$\mathbf{v} = \mathbf{v} - \sum_{i=i_v}^{n-1} \mathbf{v}_i t_{i,\mu} - \sum_{\substack{i \in \mathcal{I}_v, \\ i < i_v.}} \mathbf{v}_i t_{i,\mu}.$$

$1d)$ If $\|\mathbf{v}\| > \mathtt{dtol}$, then do the following :
Set

$$\mathbf{v}_n = \frac{\mathbf{v}}{t_{n,\mu}}, \quad \text{where} \quad t_{n,\mu} = \|\mathbf{v}\|.$$

Set $\mu_n = \mu$.

$1e)$ If $\|\mathbf{v}\| \le \mathtt{dtol}$, then deflate the vector $\mathbf{v}$ and do the following :

$i)$ If $\mu \le 0$, then do the following:
Delete one column vector from $\mathbf{X}_0^{\mathrm{cr}}$, $\mathbf{R}_0^{\mathrm{cr}}$, $\boldsymbol{\rho}^{\mathrm{cr}}$.
Set $m_{\mathrm{cr}} = m_{\mathrm{cr}} - 1$.
Repeat Step 1).

$ii)$ If $\mu > 0$ and the deflated vector $\mathbf{v}$ is nonzero, then
set $\mathcal{I}_v = \mathcal{I}_v \cup \{\mu\}$.

2) If $\mu > 0$, then do the following (Perform BL-QMR update of $\mathbf{X}_\mu^{\mathrm{cr}}$.) :

$2a)$ Update the QR factorization of $\mathbf{T}_b^{(\mu)}$ and $\mathbf{t}_\mu^{\mathrm{cr}}$.

$2b)$ Compute

$$\mathbf{X}_\mu^{\mathrm{cr}} = \mathbf{X}_{\mu-1}^{\mathrm{cr}} + \mathbf{p}_\mu \mathbf{y}_\mu^T,$$

where $\mathbf{p}_\mu$ and $\mathbf{y}_\mu^T$ are given by (41) and (42).

$2c)$ Check if all solution vectors in $\mathbf{X}_\mu^{\mathrm{cr}}$ have converged.
If yes, then recover solution vectors corresponding to deflated linear systems and stop.

$2d)$ Check if we need to deflate in $\mathbf{A}\mathbf{X}^{\mathrm{cr}} = \mathbf{B}^{\mathrm{cr}}$.
If yes, then delete one column vector from $\mathbf{X}_\mu^{\mathrm{cr}}$, $\mathbf{R}_\mu^{\mathrm{cr}}$, and $\tilde{\boldsymbol{\tau}}^{\mathrm{cr}}$.
Set $\boldsymbol{\tau}^{\mathrm{cr}} = \tilde{\boldsymbol{\tau}}^{\mathrm{cr}}$.
Set $m_{\mathrm{cr}} = m_{\mathrm{cr}} - 1$.
Repeat Steps 1) and 2).

3) (Compute δ_n and check for breakdown.)
Set

$$\delta_n = \mathbf{v}_n^T \mathbf{J} \mathbf{v}_n.$$

If $|\delta_n| \le \mathtt{btol}$, then stop.

5. Preconditioning approaches

In this section, we first motivate a crude criterion for the choice of suitable preconditioners for the BL-QMR method, and then we sketch the approaches used in our numerical study of preconditioned BL-QMR for problems in structural acoustics.

5.1. Convergence of QMR iterations

As in Section 2.4, we denote by $\mathbf{K}$ the matrix of the original linear system to be solved and by $\mathbf{M} = \mathbf{M}_1\mathbf{M}_2$ the preconditioner. Recall from (24)–(25) that preconditioning transforms the matrix $\mathbf{K}$ into the preconditioned coefficient matrix

$$\mathbf{A} = \mathbf{M}_1^{-1}\mathbf{K}\mathbf{M}_2^{-1}. \tag{45}$$

Krylov-subspace methods require the computation of matrix-vector products $\mathbf{A}\cdot\mathbf{v}$. Because of the inverses in (45), computing $\mathbf{A}\cdot\mathbf{v}$ in general involves solutions of auxiliary linear systems with $\mathbf{M}_1$ and $\mathbf{M}_2$ as coefficient matrices. Clearly, the preconditioner $\mathbf{M}$ needs to be "simple" enough so that the solution of these auxiliary systems is easy, compared to the task of solving the original system with matrix $\mathbf{K}$. On the other hand, the preconditioner $\mathbf{M}$ needs to be such that the employed Krylov-subspace method converges in considerably fewer iterations when applied to the preconditioned linear system, compared to the original system. Hence, for the choice of a suitable preconditioner it is crucial to understand the convergence behavior of the Krylov-subspace method that is used to solve the preconditioned system.

We now briefly discuss the convergence behavior of the QMR method. To keep the discussion short, we restrict ourselves to the case of the QMR method [12] for solving systems $\mathbf{A}\mathbf{x} = \mathbf{b}$ with a single right-hand side. Starting from an arbitrary initial guess $\mathbf{x}_0$ for the exact solution $\mathbf{A}^{-1}\mathbf{b}$, the QMR algorithm generates a sequence of iterates $\mathbf{x}_\mu$, $\mu = 0, 1, 2, \ldots$, that satisfy the error bound

$$\frac{\|\mathbf{b} - \mathbf{A}\mathbf{x}_\mu\|}{\|\mathbf{b} - \mathbf{A}\mathbf{x}_0\|} \;\leq\; \sqrt{\mu+1}\;\|\mathbf{U}\|\cdot\|\mathbf{U}^{-1}\|\cdot \min_{\varphi\in\mathcal{P}_\mu:\;\varphi(0)=1}\;\max_{\lambda\in\lambda(\mathbf{A})}|\varphi(\lambda)|, \tag{46}$$

see [9]. Here, $\mathbf{U}$ is a certain eigenvector matrix, $\mathcal{P}_\mu$ denotes the set of all complex polynomials of degree at most μ, and $\lambda(\mathbf{A})$ denotes the eigenspectrum of $\mathbf{A}$, i.e., $\lambda(\mathbf{A})$ is the set of all (in general, complex) eigenvalues of $\mathbf{A}$. The factor $\sqrt{\mu+1}$ of the right-hand side in (46) reflects the fact that QMR quasi-minimizes the residual norm. For a method based on a true residual minimization, such as GMRES [21], an error bound (46) with $\sqrt{\mu+1}$ replaced by 1 holds. We stress that true minimal-residual methods are, however, considerably more expensive than QMR, in terms of both computational cost and storage requirements.

The dominating factor in (46) is the value of the min-max problem

$$\Bigl(\varepsilon_\mu(\mathbf{A}) :=\Bigr) \quad \min_{\varphi\in\mathcal{P}_\mu:\;\varphi(0)=1}\;\max_{\lambda\in\lambda(\mathbf{A})}|\varphi(\lambda)|. \tag{47}$$

We remark that the quantity (47) is typical for convergence estimates of Krylov-subspace iterations based on a minimization property; in fact, it also appears in error bounds for CG and GMRES. Clearly, by (47), the factor $\varepsilon_\mu(\mathbf{A})$ is small if, and only if, there exists a polynomial φ of degree at most μ and with normalization $\varphi(0) = 1$ that is small for *all* eigenvalues of $\mathbf{A}$. Moreover, in view of (46), convergence of QMR will be the faster the smaller the degree μ of such a polynomial $\varphi(0) = 1$ that is small on the eigenspectrum $\lambda(\mathbf{A})$. For symmetric positive definite matrices $\mathbf{A}$, all eigenvalues of $\mathbf{A}$ are real and positive, and it is well known that the worst-case behavior of $\varepsilon_\mu(\mathbf{A})$ is characterized by the condition number of $\mathbf{A}$. More precisely, the smaller the condition number of $\mathbf{A}$, the faster the convergence one can expect. For general non-Hermitian matrices $\mathbf{A}$, however, the worst-case behavior of $\varepsilon_\mu(\mathbf{A})$ cannot be characterized by a single number alone, and in fact, it depends on the distribution of the eigenvalues of $\mathbf{A}$ in the complex plane. Roughly speaking, for non-Hermitian matrices $\mathbf{A}$, the more the eigenspectrum $\lambda(\mathbf{A})$ is clustered about some point in the complex plane and bounded away from the normalization point 0 in (47), the smaller the quantity $\varepsilon_\mu(\mathbf{A})$ and thus the faster the convergence one can expect.

Motivated by these considerations, preconditioners should be chosen such that the eigenspectrum of the preconditioned matrix is clustered and bounded away from the origin in the complex plane. Typical eigenspectra for preconditioned matrices in acoustic scattering problems are shown in Figure 4 below.

5.2. SSOR preconditioning

A simple, but often very efficient, preconditioner is SSOR [3]. Let $\mathbf{K} \in \mathbb{C}^{N\times N}$ be a given matrix. We denote by $\mathbf{L}$, $\mathbf{\Delta}$, and $\mathbf{U}$, the strictly lower triangular, diagonal, and strictly upper triangular part of $\mathbf{K}$, respectively, so that we have

$$\mathbf{K} = \mathbf{L} + \mathbf{\Delta} + \mathbf{U}. \tag{48}$$

Assuming that the diagonal part $\mathbf{\Delta}$ is nonsingular, the SSOR preconditioner is now defined as follows:

$$\mathbf{M}_{\mathrm{SSOR}} = \mathbf{M}_{\mathrm{SSOR}}(w) = \mathbf{M}_1\mathbf{M}_2, \tag{49}$$

where

$$\mathbf{M}_1 = \big(\mathbf{\Delta} + w\mathbf{L}\big)\mathbf{\Delta}^{-1/2} \quad \text{and} \quad \mathbf{M}_2 = \mathbf{\Delta}^{-1/2}\big(\mathbf{\Delta} + w\mathbf{U}\big). \tag{50}$$

Furthermore, $w \geq 0$ is a parameter. Mostly, SSOR is used with $w = 1$. However, for linear systems arising in structural acoustics, we found that the best choice of w depends on the actual problem, and that often faster convergence can be obtained by using parameter values $w \neq 1$. Note that, for $w = 0$, the preconditioner (49)–(50) reduces to simple diagonal preconditioning:

$$\mathbf{M}_{\mathrm{D}} = \mathbf{\Delta} = \text{the diagonal of } \mathbf{K}.$$

In the sequel, we now always assume that $w > 0$ for the SSOR preconditioner.

An important property of $\mathbf{M}_{\mathrm{SSOR}}$ is that it is an almost "free" preconditioner. More precisely, by means of the so-called Eisenstat trick [7], one can compute each matrix-vector product with the SSOR-preconditioned coefficient matrix $\mathbf{A} = \mathbf{M}_1^{-1}\mathbf{K}\mathbf{M}_2^{-1}$ at the expense of one matrix-vector product with the original matrix $\mathbf{K}$ plus a few multiplications with a diagonal matrix. The key observation is the relation

$$\begin{aligned}
\mathbf{A} &= \mathbf{M}_1^{-1}\mathbf{K}\mathbf{M}_2^{-1} \\
&= \mathbf{\Delta}^{1/2}\big(\mathbf{\Delta} + w\mathbf{L}\big)^{-1}\big(\mathbf{L} + \mathbf{\Delta} + \mathbf{U}\big)\big(\mathbf{\Delta} + w\mathbf{U}\big)^{-1}\mathbf{\Delta}^{1/2} \\
&= \frac{1}{w}\mathbf{\Delta}^{1/2}\Big(\big(\mathbf{\Delta} + w\mathbf{U}\big)^{-1} + \big(\mathbf{\Delta} + w\mathbf{L}\big)^{-1}\big(\mathbf{I} + (w-2)\mathbf{\Delta}(\mathbf{\Delta} + w\mathbf{U})^{-1}\big)\Big)\mathbf{\Delta}^{1/2},
\end{aligned}$$

which readily follows from (48) and (50). Using this relation, the result $\mathbf{r} = \mathbf{A}\cdot\mathbf{v}$ of a matrix-vector product with the preconditioned matrix $\mathbf{A}$ can be obtained as follows.

Algorithm 5.1 (Eisenstat trick for SSOR preconditioning)

INPUT: A vector $\mathbf{v} \in \mathbb{C}^N$.
OUTPUT: The vector $\mathbf{r} = \mathbf{A} \cdot \mathbf{v}$.

1) Set $\mathbf{t} = \mathbf{\Delta}^{1/2}\,\mathbf{v}$.
2) Solve $\left(\mathbf{\Delta} + w\mathbf{U}\right)\hat{\mathbf{t}} = \mathbf{t}$ for $\hat{\mathbf{t}}$.
3) Set $\hat{\mathbf{t}} := \mathbf{t} + (w-2)\,\mathbf{\Delta}\,\hat{\mathbf{t}}$.
4) Solve $\left(\mathbf{\Delta} + w\mathbf{L}\right)\tilde{\mathbf{t}} = \hat{\mathbf{t}}$ for $\tilde{\mathbf{t}}$.
5) Set $\mathbf{r} = \frac{1}{w}\mathbf{\Delta}^{1/2}\left(\mathbf{t} + \tilde{\mathbf{t}}\right)$.

For the numerical results with the SSOR preconditioner reported in Section 6 below, we always used Algorithm 5.1 to compute the matrix-vector products with $\mathbf{A}$.

5.3. Hierarchical basis preconditioning

While SSOR is a purely algebraic preconditioner that is defined for any matrix $\mathbf{K}$ with nonsingular diagonal $\mathbf{\Delta}$, hierarchical basis preconditioning is specifically tailored to linear systems resulting from finite-element discretizations of differential equations. The key observation here is that the set of basis functions that span finite-element spaces, such as $\mathcal{S}^h_{\mathbf{u}}$, $\mathcal{V}^h_{\mathbf{u}}$ in (12) and $\mathcal{S}^h_p$, $\mathcal{V}^h_p$ in (13), can be chosen in several different ways. For Krylov-subspace methods, the choice of basis functions becomes important since it affects the eigenvalue distribution of the discrete equations, and hence the convergence behavior.

Typically, finite-element formulations are based on nodal basis functions that consist of Lagrange polynomials in one dimension and of products thereof in higher dimensions. However, for certain classes of problems, switching to hierarchical basis functions yields matrix problems that are "easier" to solve by Krylov-subspace methods; see, e.g., [5, 24]. The hierarchical basis (HB) preconditioner [22] is a preconditioning approach that uses transformations between the nodal basis functions and a certain set of hierarchical basis functions. In particular, for two-dimensional boundary-value problems arising from self-adjoint elliptic partial differential equations, the HB preconditioner [22] is nearly optimal in a certain sense. While preconditioners based on this approach have primarily been constructed for symmetric positive definite linear systems, indefinite problems have been studied for the special case of the Helmholtz equation with simplified boundary conditions [23]. Motivated by numerical results reported with the HB preconditioner for exterior scattering problems in [20], we also tested this approach in conjunction with BL-QMR for solving multiple right-hand side problems in acoustics. Next, we give a brief overview of the HB preconditioner.

5.3.1. Transformations from nodal to hierarchical basis

Let the function $u(x) \in H^1$ be approximated on a given finite-element mesh with N unknowns (or degrees of freedom) using the standard piecewise-continuous linear nodal basis functions. An alternative choice is to use a set of hierarchical basis functions defined in such a way that they are equivalent to the nodal basis and satisfy the usual completeness requirements for finite-element basis functions; we refer to [20, 22] for their actual definition.

Let us denote the column vectors containing nodal and hierarchical basis functions, respectively, by $\Phi(x) = [\,\phi_1(x) \;\; \phi_2(x) \;\; \ldots \;\; \phi_N(x)]^T$ and $\Psi(x) = [\,\psi_1(x) \;\; \psi_2(x) \;\; \ldots \;\; \psi_N(x)]^T$. Then, $u^h(x)$, the approximation of $u(x)$, can be represented as

$$u^h(x) = \sum_{i=1}^{N} \alpha_i \, \phi_i(x) = \sum_{i=1}^{N} \beta_i \, \psi_i(x) \,, \tag{51}$$

where $\{\alpha_i\}$ and $\{\beta_i\}$, $i = 1, 2, \ldots, N$, are the unknown coefficients (or degrees of freedom) in the respective bases. Furthermore, at any point $x \in \Omega_f \bigcup \Omega_s$, the nodal basis and the hierarchical basis functions are related by

$$\Psi(x) = \mathbf{P} \cdot \Phi(x), \tag{52}$$

where $\mathbf{P}$ is an $N \times N$ matrix whose elements are constant and independent of x. In fact, the matrix $\mathbf{P}$ only depends on the mesh topology and the associated definition of the hierarchical basis functions. Now, we consider the partition of vectors $\Phi = [\,\Phi_s^T \;\; \Phi_f^T\,]^T$ and $\Psi = [\,\Psi_s^T \;\; \Psi_f^T\,]^T$, such that $\Phi_s, \Psi_s \in \mathbb{R}^{N_s}$ and $\Phi_f, \Psi_f \in \mathbb{R}^{N_f}$ contain basis functions for structural and fluid degrees of freedom, respectively. Using this reordering of unknowns, $\mathbf{P}$ becomes a block-diagonal matrix, and thus, (52) can be rewritten as follows:

$$\begin{bmatrix} \Psi_s \\ \Psi_f \end{bmatrix} = \mathbf{P} \begin{bmatrix} \Phi_s \\ \Phi_f \end{bmatrix}, \quad \text{where} \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_s & 0 \\ 0 & \mathbf{P}_f \end{bmatrix}. \tag{53}$$

5.3.2. Formulation of the preconditioner

First, we represent the functions in the variational formulation (17)–(18) of the acoustic problem in terms of the hierarchical basis, as shown in (51). Thereafter, using (53), we introduce nodal basis functions Φ in place of Ψ to obtain a relation between the nodal basis finite-element coefficient matrix $\mathbf{K}$ in (19), (21) and its equivalent representation, $\hat{\mathbf{K}}$, in the hierarchical basis. The result is the relation

$$\hat{\mathbf{K}} = \mathbf{P} \begin{bmatrix} \mathbf{K}_s - \omega^2 \mathbf{M}_s & \mathbf{C}^T \\ \mathbf{C} & \frac{1}{\rho_0 \omega^2} \left(\mathbf{K}_f - k_0^2 \mathbf{M}_f + \mathbf{K}_{\mathrm{DtN}} \right) \end{bmatrix} \mathbf{P}^T = \mathbf{P}\mathbf{K}\mathbf{P}^T. \tag{54}$$

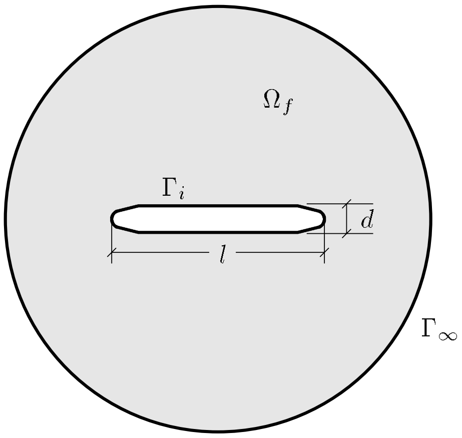
Since linear systems with the matrix $\hat{\mathbf{K}}$ can be expected to be "easier" to solve by Krylov-subspace methods, we now choose $\mathbf{A} = \hat{\mathbf{K}}$ as the preconditioned matrix. In view of (54) and (45), this choice corresponds to setting $\mathbf{M}_1^{-1} = \mathbf{P}$ and $\mathbf{M}_2^{-1} = \mathbf{P}^T$. Therefore, the hierarchical basis preconditioner can be stated as

$$\mathbf{M}_{\mathrm{HB}} = \mathbf{M}_1 \mathbf{M}_2 = (\mathbf{P}^T \mathbf{P})^{-1}. \tag{55}$$

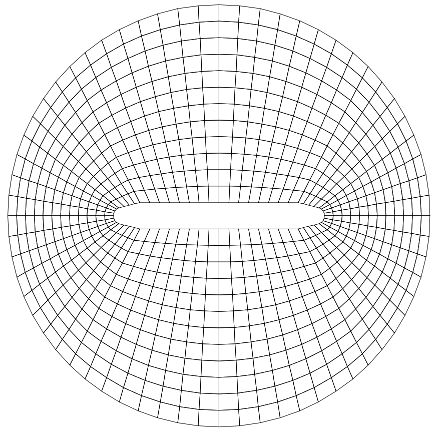
Typically, the hierarchical basis approach (55) can be enhanced further by combining it with a diagonal scaling of $\mathbf{K}$, resulting in the preconditioner

$$\mathbf{M}_{\mathrm{HBDS}} = (\mathbf{P}^T [\mathrm{diag}(\mathbf{K})]^{-1} \mathbf{P})^{-1}. \tag{56}$$

In fact, for the numerical tests reported in the next section, we always used the variant (56) of the hierarchical basis preconditioner. Finally, we stress that, for the actual computations



(a) Computational domain



(b) Finite-element mesh

Fig. 2. Scattering from a rigid cylinder with spherical-to-conical end caps

with the preconditioner (56), the transformation matrix $\mathbf{P}$ need not be formed or stored explicitly. Instead, at each preconditioning step, the required matrix-vector products $\mathbf{P} \cdot \mathbf{r}$ and $\mathbf{P}^T \cdot \mathbf{r}$, are computed directly using mesh connectivity data. When implemented in this form, preconditioning with $\mathbf{M}_{\text{HBDS}}$ requires $\mathcal{O}(N)$ operations and $\mathcal{O}(N)$ words of additional storage; for further details, we refer the reader to [22, 20].

6. Numerical examples

In this section, we present numerical tests with the preconditioned $\mathbf{J}$-symmetric BL-QMR method applied to multiple right-hand side problems in acoustics.

For all our tests, we used `ctol` $= 10^{-6}$ as the convergence tolerance, i.e., we stop iterations when residuals for the preconditioned system (24) satisfy

$$\|\mathbf{r}_\mu^{(j)}\| \;\leq\; \|\mathbf{r}_0^{(j)}\| \cdot 10^{-6}, \quad j = 1, 2, \ldots, m.$$

Furthermore, we used the deflation tolerance `dtol` $= 10^{-6}$ and the breakdown tolerance `btol` $= 10^{-14}$ in Algorithm 4.1.

Example 6.1 (Plane-wave scattering from a rigid cylinder)
In this example, we consider the scattering of plane waves in two dimensions from a rigid cylinder. For the case of rigid scattering, the boundary-value problem described in Section 2.1 simplifies, and it reduces to equations (2), (6), and (9), with $f_f := 0$, $h_f := 0$, $\Gamma_h^f \equiv \Gamma_i$, and $\Gamma_g^f = \emptyset$.

The computational domain for the scattering problem is shown in Figure 2(a). We consider a cylindrical scatterer with conical-to-spherical end caps and a large aspect ratio of $l/d = 8$. The artificial boundary Γ_∞ is taken to be a circle of radius $R = l$ centered

Tab. 1. Comparison of BL-QMR with individual solution of each right-hand side using QMR

Number of RHS (m)	1	2	3	4	5	6	7
Cumulative QMR iterations	114	280	439	601	763	934	1094
BL-QMR iterations	114	224	246	293	314	333	346

about the cylinder. The incident pressure field, corresponding to a plane wave with unit amplitude and incident at angle θ to the positive $\mathbf{x}$ axis, is given by

$$p_{\text{inc}}(\mathbf{x}) = \exp(\mathrm{i}\,\mathbf{k}\cdot\mathbf{x}), \tag{57}$$

where $\mathbf{k} = \begin{bmatrix} k\cos\theta & k\sin\theta \end{bmatrix}$ is the wave vector and $\mathbf{x} = \begin{bmatrix} x & y \end{bmatrix}^T$ is the position vector. Inserting (57) into (22), together with (15), it follows that

$$\widehat{L}_f(\hat{p}) = -\left(\ \hat{p},\ \mathrm{i}\,\mathbf{k}\cdot\mathbf{n}\exp(\mathrm{i}\,\mathbf{k}\cdot\mathbf{x})\ \right)_{\Gamma_h^f}. \tag{58}$$

Next, by introducing finite-element nodal basis functions in the weak formulation for this problem, one obtains the linear system

$$\left[\mathbf{K}_f - k_0^2\,\mathbf{M}_f + \mathbf{K}_{\text{DtN}}\right]\mathbf{p} = \mathbf{f},$$

where the matrices $\mathbf{K}_f$, $\mathbf{M}_f$, and $\mathbf{K}_{\text{DtN}}$ arise as described in Section 2.2, and $\mathbf{f}$ is obtained from (58). In our numerical tests, we used multiple right-hand side vectors $\mathbf{f}^{(j)}$, $j = 1, 2, \ldots, m$, resulting from varying angles of incidence $\theta_j = (j-1)\,\Delta\theta$ in (58).

We begin by comparing the block solution approach using BL-QMR with the individual solution of each of the m linear systems in a given block. We set $k_0 d = \pi/6$, $\Delta\theta = 10°$, and use $\mathbf{X}_0 = \mathbf{0}$ as initial guess. In all the numerical tests conducted to solve the problem at $k_0 d = \pi/6$, we employ the discretization of the computational domain shown in Figure 2(b), which has 832 nodes and 768 bilinear quadrilateral elements, and results in a problem size with $N = 832$ unknowns. In each column of Table 1, we compare cumulative iteration counts required to individually solve each of the m linear systems using complex symmetric QMR [8] with the number of BL-QMR iterations required to solve all the m systems simultaneously. Note that, as m increases, the BL-QMR method requires significantly fewer iterations than individual solves. For instance, for $m = 7$, the BL-QMR method needed about 3.2 times fewer iterations; see Table 1. We remark that both the complex symmetric QMR algorithm for single right-hand sides and the BL-QMR Algorithm 4.1 for multiple right-hand sides require one matrix-vector product $\mathbf{A}\cdot\mathbf{v}$ per iteration. Since, for large values of N, these products represent the dominating computational cost in both algorithms, the block solution approach can be expected to reduce overall solution times significantly.

Next, we compare the performance of various preconditioners in conjunction with BL-QMR. Table 2 shows the iteration counts and the number of deflations performed (in parenthesis) for different preconditioning approaches; here, $\mathbf{M}_{\text{I}} = \mathbf{I}$ (the $N \times N$ identity matrix) is used to denote the unpreconditioned system. The hierarchical basis preconditioner $\mathbf{M}_{\text{HBDS}}$

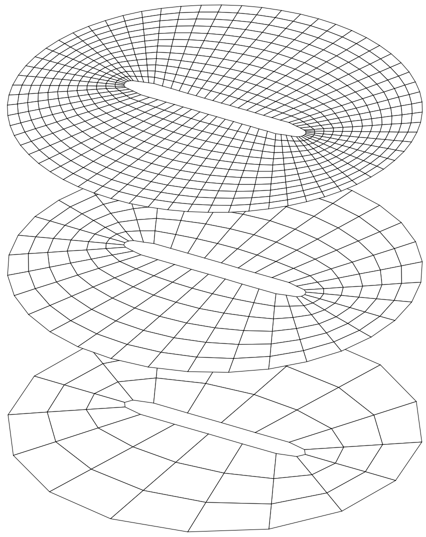


Fig. 3. Hierarchical grid levels employed to construct the $\mathbf{M}_{\mathrm{HBDS}}$ preconditioner for the mesh shown in Figure 2(b)

Tab. 2. Performance of various preconditioners with BL-QMR for the scattering problem with $k_0 d = \pi/6$, $N = 832$, $\Delta\theta = 1.5°$, and $\mathbf{X}_0 = \mathbf{0}$

Number of RHS (m)	Preconditioner				
	$\mathbf{M}_{\mathrm{I}}$	$\mathbf{M}_{\mathrm{D}}$	$\mathbf{M}_{\mathrm{HBDS}}$	$\mathbf{M}_{\mathrm{SSOR}}(1.0)$	$\mathbf{M}_{\mathrm{SSOR}}(1.4)$
1	114 (0)	108 (0)	47 (0)	51 (0)	41 (0)
5	348 (0)	284 (0)	181 (0)	115 (0)	98 (0)
10	337 (3)	336 (3)	248 (3)	156 (3)	128 (3)
20	462 (11)	467 (11)	332 (10)	201 (11)	169 (11)
40	708 (27)	727 (27)	503 (27)	380 (27)	366 (27)

Tab. 3. Performance of preconditioned BL-QMR for the scattering problem with $k_0 d = \pi/6$, $N = 832$, $\Delta\theta = 1.5°$, and random $\mathbf{X}_0$

Number of RHS (m)	Preconditioner				
	$\mathbf{M}_{\mathrm{I}}$	$\mathbf{M}_{\mathrm{D}}$	$\mathbf{M}_{\mathrm{HBDS}}$	$\mathbf{M}_{\mathrm{SSOR}}(1.0)$	$\mathbf{M}_{\mathrm{SSOR}}(1.4)$
1	141	137	52	49	41
5	322	302	172	118	100
10	432	404	301	174	163
20	576	572	522	267	282
40	1119	756	809	428	479

uses the hierarchical grid levels shown in Figure 3. Note that for any given preconditioner, the BL-QMR method requires fewer and fewer iterations per right-hand side as m increases. However, results in Table 2 also indicate that the choice of a good preconditioner in the context of block solution methods is more difficult than what first appears. In particular, observe that although the hierarchical basis preconditioner $\mathbf{M}_{\mathrm{HBDS}}$ seems to be about as effective as the SSOR preconditioners for the case $m = 1$, their performance differs considerably for $m \geq 5$. For problems of acoustic radiation and scattering, we have found that SSOR outperforms the hierarchical basis preconditioner for all block sizes $m > 1$. In Figure 4, we plot the eigenspectra (in blue) of the preconditioned matrix $\mathbf{A}$ for various preconditioners, together with the eigenspectrum (in red) of the original unpreconditioned matrix $\mathbf{K}$. Observe that the SSOR preconditioners are more effective in contracting the spectrum of $\mathbf{A}$ in the complex plane, and away from the origin, than $\mathbf{M}_{\mathrm{HBDS}}$ or $\mathbf{M}_{\mathrm{D}}$. Finally, note from Table 2 that as m increases, progressively larger number of deflations are performed. These deflations were required to ensure linear independence of the vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{m_1}$ in the initial block. We stress that it is crucial to strictly enforce the deflation of linearly dependent and almost linearly dependent vectors in block solution methods. For example, if we do not enforce deflation in the tests shown in Table 2 with $\mathbf{M}_{\mathrm{SSOR}}(1.4)$, then for the case $m = 10$ convergence is achieved only after 316 iterations and for $m = 20$ no convergence is achieved even after 900 iterations. In fact, in the latter case, the residuals corresponding to the 20 right-hand sides stagnate with

$$\min_{\mu=0,1,\ldots} \|\mathbf{r}_\mu^{(j)}\| \;\geq\; (2.88 \times 10^{-1}) \cdot \|\mathbf{r}_0^{(j)}\| \quad \text{for all} \quad j = 1, 2, \ldots, m.$$

Next, we consider the effect of choosing the initial block $\mathbf{X}_0 \in \mathbb{C}^{N \times m}$ as a random matrix whose entries are uniformly distributed between -1.0 and 1.0, and in Table 3, we show the resulting iteration counts. A comparison of these results with those in Table 2 indicates that although deflations are avoided by using random starting vectors, in some cases the iteration counts can be significantly higher than those for $\mathbf{X}_0 = \mathbf{0}$.

Next, we fix the number of right-hand sides to be $m = 10$ and consider the effect of varying the increment $\Delta\theta$ of the incident angles $\theta_j = (j-1)\,\Delta\theta$, $j = 1, 2, \ldots, 10$. Table 4

Tab. 4. Convergence of BL-QMR for fixed block size $m = 10$ and various values of $\Delta\theta$ ($k_0 d = \pi/6$, $N = 832$, and $\mathbf{M} = \mathbf{M}_{\mathrm{SSOR}}(1.4)$)

$\Delta\theta$	$\mathbf{X}_0 = \mathbf{0}$		Random $\mathbf{X}_0$	
	Number of Deflations	Number of Iterations	Number of Deflations	Number of Iterations
0.5	5	98	0	163
1.0	4	114	0	163
2.0	3	128	0	163
5.0	0	157	0	164

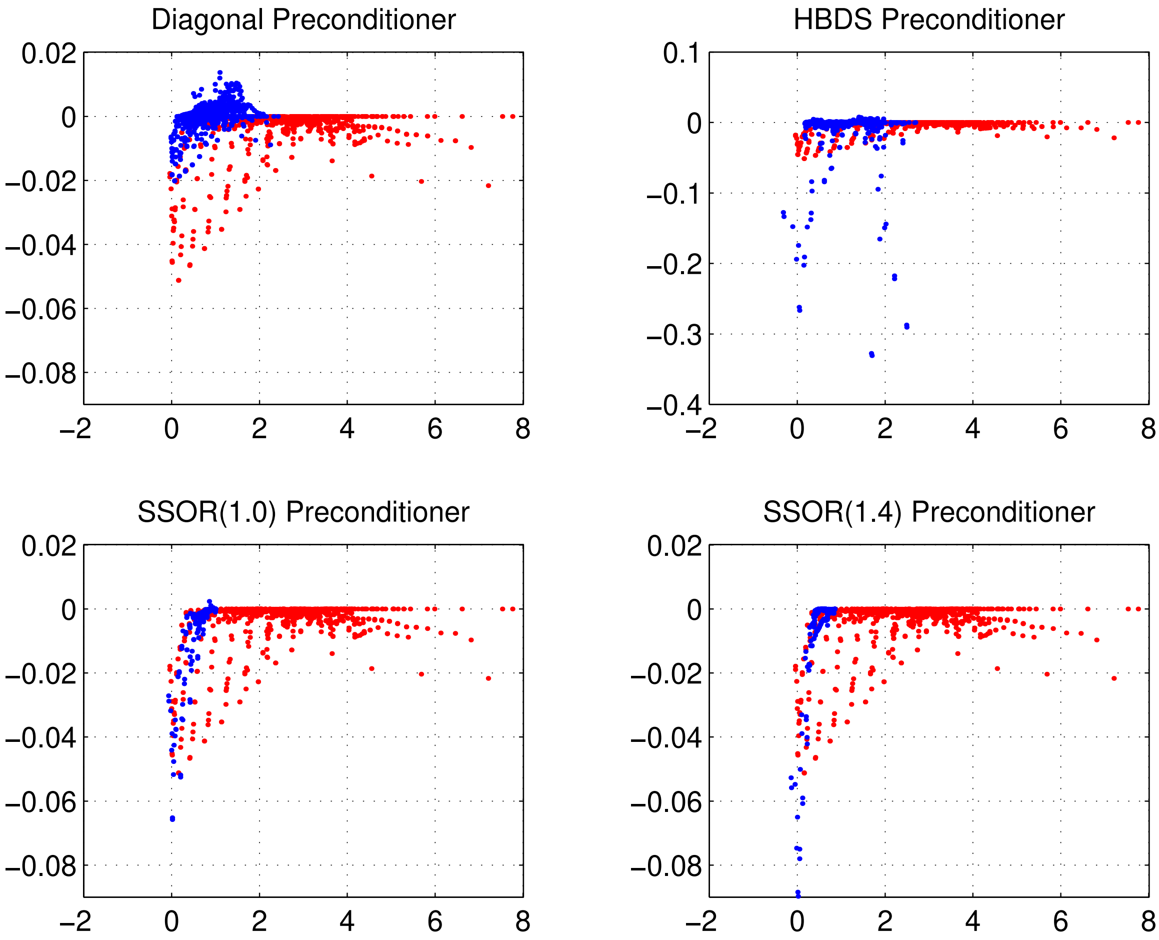


Fig. 4. Eigenspectra (in blue) of the preconditioned matrices $\mathbf{A}$ and eigenspectrum (in red) of the unpreconditioned matrix $\mathbf{K}$ for plane-wave scattering with $N = 832$ and $k_0 d = \pi/6$

Tab. 5. Iteration counts for BL-QMR for the scattering problem at $k_0 d = \pi$, with $\Delta\theta = 1.5°$, $\mathbf{M} = \mathbf{M}_{\mathrm{SSOR}}(1.2)$, and random $\mathbf{X}_0$

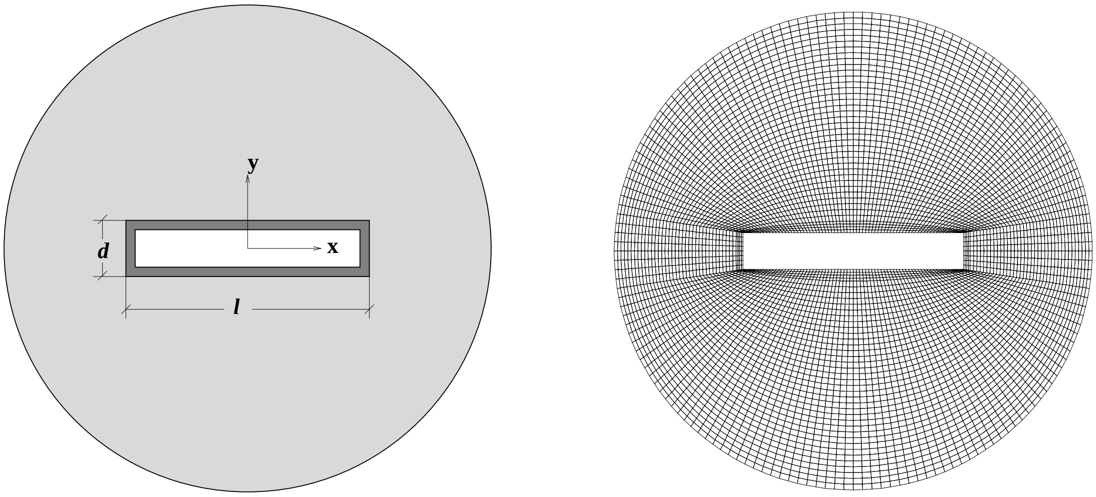
Number of RHS (m)	Number of Iterations (n_t)	Iterations/RHS (n_t/m)
1	309	309
10	1291	129.1
20	1506	75.3
30	1621	54.3

shows the resulting iteration counts. Note that as $\Delta\theta$ decreases, the solution of multiple problems becomes faster if $\mathbf{X}_0 = \mathbf{0}$ and deflations are enforced. However, as one would expect, iteration counts are almost independent of $\Delta\theta$ if a random initial block $\mathbf{X}_0$ is used instead.

Finally, in Table 5, we present iteration counts for solving multiple scattering problems at a higher frequency that corresponds to $k_0 d = \pi$. We now use a finite-element mesh, with 12544 nodes and 12288 elements, which is obtained by uniformly refining the mesh shown in Figure 2(b). Iteration counts for increasing values of m clearly indicate the advantage of employing preconditioned BL-QMR to solve multiple right-hand side problems in acoustic wave scattering.

Example 6.2 (Radiation from a fluid-loaded box frame)
In this example, we consider a two-dimensional coupled fluid-structure interaction problem that is excited by externally applied point loads. The elastic radiating structure is chosen as a rectangular configuration consisting of four interconnected elastic plates; the computational domain for this problem is illustrated in Figure 5(a). The aspect ratio of the rectangular configuration is taken to be $l/d = 6$, with the shorter dimension $d = 1.0$cm, and the thickness of each plate is 0.15cm. The material properties for the steel plates are as follows: $E = 210 \times 10^{10}\text{dyne/cm}^2$, $\nu = 0.29$, $\kappa = 5/6$, and $\rho = 7.8\,\text{g/cm}^3$. The plates are connected rigidly with continuity of displacement and rotations at the four corner joints. Longitudinal motion has been added into the finite-element model, although we did not discuss its implementation in Section 2.2; instead, we refer the reader to [6] for details. The discretization of this problem using Galerkin and Galerkin least-squares type finite-element approximations was studied recently by Grosh [15]. In our numerical tests, we employ the Galerkin approach, and choose our finite-element grids to satisfy the resolution guidelines proposed in [15].

In many practical situations, it is of interest to solve the radiation problem for multiple locations of an excitation force. In our numerical tests, we consider the two sets of point forces shown in Table 6, which lead to multiple right-hand side problems with $m = 1, 2, \ldots, 7$. We remark that Load Case I consists of point forces of similar magnitude and direction applied on adjacent node locations in the finite-element mesh, while Load



(a) Computational domain

(b) Finite-element mesh

Fig. 5. Radiation from box-frame under point excitations

Tab. 6. Point load excitations and their locations considered in Load Case I and Load Case II of Example 6.2

RHS vector	Load case I		Load case II	
	Node location	Force	Node location	Force
1	(0.0,0.5)	(1,0,0)	(3.0,0.0)	(0,1,0)
2	(-0.08,0.5)	(1,0,0)	(3.0,0.25)	(1,0,0)
3	(-0.16,0.5)	(1,0,0)	(3.0,0.5)	(0,0,1)
4	(-0.25,0.5)	(1,0,0)	(2.83,0.5)	(1,0,0)
5	(-0.33,0.5)	(1,0,0)	(2.67,0.5)	(1,0,0)
6	(-0.41,0.5)	(1,0,0)	(2.5,0.5)	(0,1,0)
7	(-0.5,0.5)	(1,0,0)	(2.33,0.5)	(1,0,0)

Tab. 7. Iteration counts for convergence of BL-QMR for Load Case I with $k_0 d = 1.0$ and $N = 7040$

Number of RHS (m)	Preconditioner		
	$\mathbf{M}_{\mathrm{D}}$	$\mathbf{M}_{\mathrm{HBDS}}$	$\mathbf{M}_{\mathrm{SSOR}}(1.0)$
1	715 (0)	368 (0)	557 (0)
2	1529 (0)	701 (0)	517 (0)
3	1559 (1)	756 (1)	521 (1)
4	1557 (2)	697 (2)	522 (2)
5	1588 (3)	721 (3)	524 (3)
6	1961 (4)	740 (4)	533 (4)
7	1610 (5)	767 (5)	524 (5)

Case II assumes an arbitrary combination of point forces and their point of application. Based on the resolution requirements suggested in [15, Chapter 4], we set $k_0 = 1.0\text{cm}^{-1}$ and use a finite-element discretization consisting of 6720 nodes, such that the number of structural and fluid unknowns, respectively, are $N_s = 480$ and $N_f = 6560$. The resulting coefficient matrix $\mathbf{K}$ is complex symmetric and of order $N = 7040$.

Table 7 presents iteration counts and the number of deflations required by the preconditioned BL-QMR method for multiple right-hand sides corresponding to Load Case I. In this example, we observe that the simultaneous solution of multiple linear systems is obtained very efficiently by using $\mathbf{X}_0 = \mathbf{0}$ and enforcing deflations. While in most cases the iteration counts increase only modestly as m increases, in some cases the number of iterations actually decreases when additional linear systems are solved simultaneously! The explanation of this phenomenon is that BL-QMR becomes less prone to loss of biorthogonality as the block size m increases. More precisely, in the underlying Lanczos-type algorithm, biorthogonality of the Lanczos vectors is explicitly enforced only among segments of $2m$ consecutive vectors. The remaining biorthogonality conditions are satisfied in exact arithmetic, but are gradually lost in finite-precision arithmetic due to round-off. While this loss of biorthogonality does not prevent convergence of the BL-QMR method, it typically increases the convergence counts. Consequently, as m increases, more biorthogonality is enforced explicitly and, for BL-QMR, convergence counts per right-hand side can actually decrease.

To illustrate the loss of biorthogonality for this example, in Figure 6, we plot a measure of biorthogonality of Lanczos vector $\mathbf{v}_{50}$ against all other vectors for the two block sizes of $m = 1$ and $m = 5$. Similarly, Figure 7 shows the biorthogonality between $\mathbf{v}_{200}$ and all other Lanczos vectors for $m = 1$ and $m = 5$. Note that, in both figures, the biorthogonality measure is several orders lower among a larger set of Lanczos vectors for the case $m = 5$ than the case $m = 1$.

Finally, in Table 8 we show the iteration counts and the number of deflations for solving multiple right-hand side problems of Load Case II. Once again, we observe that additional

Tab. 8. Iteration counts for convergence of BL-QMR for Load Case II with $k_0d = 1.0$, $N = 7040$, and $\mathbf{M} = \mathbf{M}_{\mathrm{SSOR}}(1.0)$

Number of RHS (m)	1	2	3	4	5	6	7
Number of iterations	555	550	563	565	581	598	601
Number of deflations	0	0	0	0	0	0	1

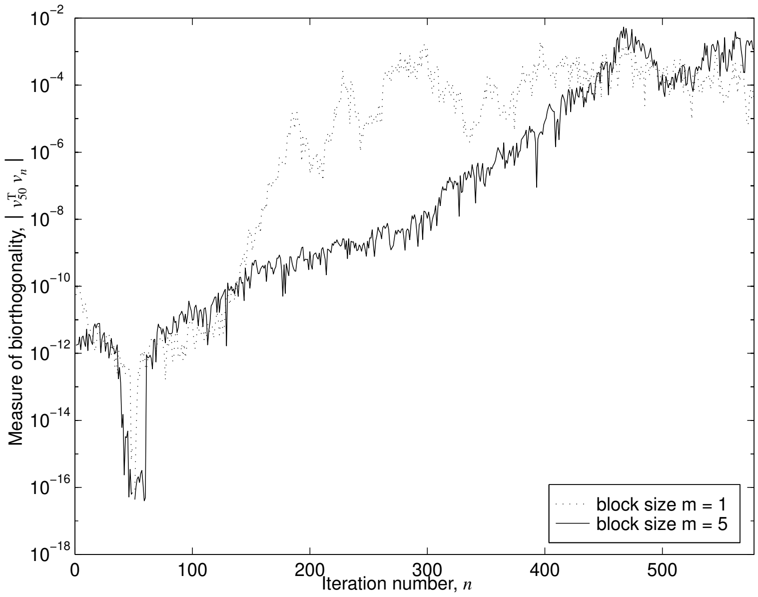


Fig. 6. Biorthogonality of Lanczos vector $\mathbf{v}_{50}$ against all other vectors for block sizes $m = 1$ and $m = 5$

systems could be solved in this example with very little increase in iteration counts.

7. Concluding remarks

We have considered the iterative solution of multiple right-hand side problems arising in structural acoustics using the BL-QMR method. In order to take advantage of the complex symmetry of the coefficient matrices of these multiple linear systems, we used a $\mathbf{J}$-symmetric variant of the BL-QMR method. Our numerical results show that, for both radiation and scattering problems, multiple linear systems can be solved very efficiently using the BL-QMR method. In fact, for multiple radiation problems considered in our numerical study, we have observed close to optimal speed-ups over solving the systems individually. Further, we have demonstrated the importance of deflation in block Krylov-subspace methods.

We have compared the performance of SSOR and hierarchical basis preconditioners, used in conjunction with BL-QMR, for the block solution of acoustics problems. Our numerical

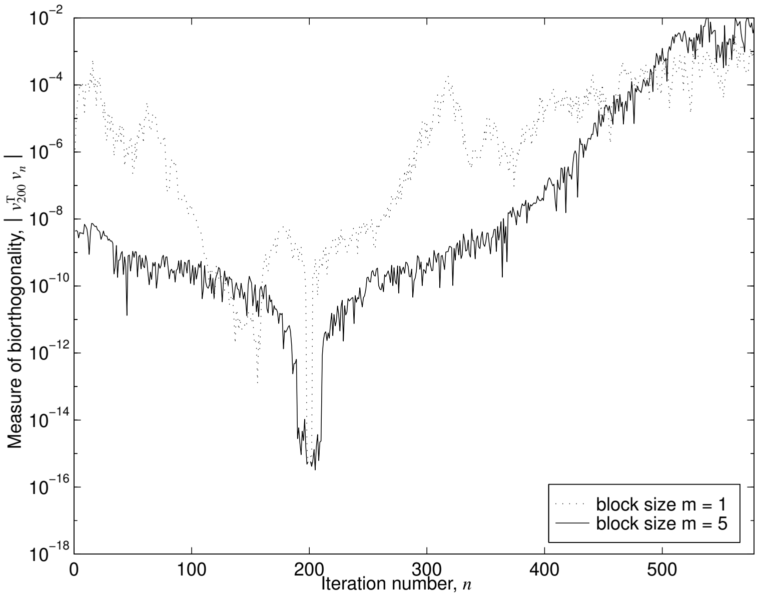


Fig. 7. Biorthogonality of Lanczos vector $\mathbf{v}_{200}$ against all other vectors for block sizes $m = 1$ and $m = 5$

tests indicate that SSOR outperforms the hierarchical basis preconditioner for block sizes greater than one. While these tests help to identify appropriate preconditioners for these problems, a theoretical understanding of this behavior needs to be developed.

Multiple linear systems also arise in the analysis of wave propagation problems at varying frequencies. However, unlike the multiple right-hand side problem, the resulting linear systems now involve coefficient matrices that change with the frequency. Therefore, block Krylov-subspace methods of the type studied in this paper are no longer applicable. There is a need to develop new types of iterative techniques for the efficient solution of such multiple linear systems with both varying coefficient matrices and right-hand sides.

Acknowledgements

The first and third authors were supported in part by ONR grant N00014-92-J-1774. Part of this research was done while the first author was visiting AT&T Bell Laboratories, Murray Hill, New Jersey.

References

[1] N.N. Abboud, A mixed finite element formulation for the transient and harmonic exterior fluid-structure interaction problem, Ph.D. Thesis, Department of Civil Engineering, Stanford University (Stanford, California, 1990).

[2] J.I. Aliaga, D.L. Boley, R.W. Freund and V. Hernández, A Lanczos-type algorithm for multiple starting vectors, Numerical Analysis Manuscript No. 96-18, Bell Laboratories (Murray Hill, New Jersey, 1996).

[3] O. Axelsson, A survey of preconditioned iterative methods for linear systems of algebraic equations, BIT 25 (1985) 166–187.

[4] W.E. Boyse and A.A. Seidl, A block QMR method for computing multiple simultaneous solutions to complex symmetric systems, SIAM J. Sci. Comput. 17 (1996) 263–274.

[5] G.F. Carey and E. Barragy, Basis function selection and preconditioning high degree finite element and spectral methods, BIT 29 (1989) 794–804.

[6] R.D. Cook, D.S. Malkus and M.E. Plesha, Concepts and Applications of Finite Element Analysis (John Wiley & Sons, New York, 1989).

[7] S.C. Eisenstat, Efficient implementation of a class of preconditioned conjugate gradient methods, SIAM J. Sci. Statist. Comput. 2 (1981), 1–4.

[8] R.W. Freund, Conjugate gradient-type methods for linear systems with complex symmetric coefficient matrices, SIAM J. Sci. Comput. 13 (1992) 425–448.

[9] R.W. Freund, Quasi-kernel polynomials and convergence results for quasi-minimal residual iterations, in: D. Braess and L. Schumaker, eds., Numerical Methods of Approximation Theory, Vol. 9 (Birkhäuser, Basel, 1992) 77–95.

[10] R.W. Freund, G.H. Golub and N.M. Nachtigal, Iterative solution of linear systems, Acta Numerica 1 (1992) 57–100.

[11] R.W. Freund and M. Malhotra, A block-QMR algorithm for non-Hermitian linear systems with multiple right-hand sides, Numerical Analysis Manuscript No. 95-09, AT&T Bell Laboratories (Murray Hill, New Jersey, 1995). (Available on WWW at `http://cm.bell-labs.com/cs/doc/95`)

[12] R.W. Freund and N.M. Nachtigal, QMR: a quasi-minimal residual method for non-Hermitian linear systems, Numer. Math. 60 (1991) 315–339.

[13] R.W. Freund and N.M. Nachtigal, Software for simplified Lanczos and QMR algorithms, Appl. Numer. Math. 19 (1995) 319–341.

[14] G.H. Golub and C.F. Van Loan, Matrix Computations, 2nd Edition (The Johns Hopkins University Press, Baltimore, 1989).

[15] K. Grosh, Design and Analysis of Computational Methods for Structural Acoustics, Ph.D. Thesis, Department of Mechanical Engineering, Stanford University (Stanford, California, 1994).

[16] M.R. Hestenes and E. Stiefel, Methods of conjugate gradients for solving linear systems, J. Res. Nat. Bur. Standards 49 (1952) 409–436.

[17] T.J.R. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis (Prentice-Hall, Englewood Cliffs, New Jersey, 1987).

[18] J.B. Keller and D. Givoli, Exact non-reflecting boundary conditions, J. Comput. Phys. 82 (1989) 172–192.

[19] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, J. Res. Nat. Bur. Standards 45 (1950) 255–282.

[20] M. Malhotra and P.M. Pinsky, Parallel preconditioning based on h-hierarchical finite elements with application to acoustics, Manuscript No. SCCM-95-05, Scientific Computing and Computational Mathematics Program, Stanford University (Stanford, California, 1995).

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

[22] H. Yserentant, On the multi-level splitting of finite element spaces, Numer. Math. 49 (1986) 379–412.

[23] H. Yserentant, Preconditioning indefinite discretization matrices, Numer. Math. 54 (1990) 719–734.

[24] O.C. Zienkiewicz, D.W. Kelly, J. Gago and I. Babuška, Hierarchical finite element approaches, error estimates and adaptive refinement, in: J.R. Whiteman, ed., The Mathematics of Finite Elements and Applications, IV (Academic Press, London, 1982) 314–346.