Efficient block preconditioning for a C1 finite element discretisation of the Dirichlet biharmonic problem

Pestana, Jennifer and Muddle, Richard and Heil, Matthias and Tisseur, Francoise and Mihajlovic, Milan

2015

MIMS EPrint: **2015.23**

Manchester Institute for Mathematical Sciences
School of Mathematics

The University of Manchester

Reports available from: http://eprints.maths.manchester.ac.uk/
And by contacting: The MIMS Secretary
School of Mathematics
The University of Manchester
Manchester, M13 9PL, UK

ISSN 1749-9097

EFFICIENT BLOCK PRECONDITIONING FOR A C^1 FINITE ELEMENT DISCRETISATION OF THE DIRICHLET BIHARMONIC PROBLEM

J. PESTANA[†§], R. MUDDLE[†‡], M. HEIL[†], F. TISSEUR[†§], AND M. MIHAJLOVIĆ[‡]

Abstract. We present an efficient block preconditioner for the two-dimensional biharmonic Dirichlet problem discretised by C^1 bicubic Hermite finite elements. In this formulation each node in the mesh has four different degrees of freedom (DOFs). Grouping DOFs of the same type together leads to a natural blocking of the Galerkin coefficient matrix. Based on this block structure, we develop two preconditioners: a 2×2 block diagonal preconditioner (BD) and a block bordered diagonal (BBD) preconditioner. We prove mesh-independent bounds for the spectra of the BD-preconditioned Galerkin matrix under certain conditions. The eigenvalue analysis is based on the fact that the proposed preconditioner, like the coefficient matrix itself, is symmetric positive definite and is assembled from element matrices. We demonstrate the effectiveness of an inexact version of the BBD preconditioner, which exhibits near-optimal scaling in terms of computational cost with respect to the discrete problem size. Finally, we study robustness of this preconditioner with respect to element stretching, domain distortion and non-convex domains.

Key words. biharmonic equation; Hermite bicubic finite elements; block preconditioning; conjugate gradient method; algebraic multigrid

AMS subject classifications. 65F08, 65F10, 65N22

1. Introduction. The biharmonic operator is a key component in mathematical models of a number of important physical problems. It arises in plane strain and plane stress elasticity problems, where the solution is expressed in terms of an Airy stress function [32, p. 79], [37, p. 288], and in plate bending problems. It also occurs in the stream-function-vorticity formulation of two-dimensional Stokes flow [27].

The strong formulation of the Dirichlet biharmonic problem seeks the function $u \in C^4(\Omega)$ that satisfies

$$\nabla^4 u = f \tag{1.1}$$

in the domain $(x_1, x_2) \in \Omega \subset \mathbb{R}^2$ with piecewise smooth boundary $\partial\Omega$ and source function $f \in L_2(\Omega)$ subject to the Dirichlet boundary conditions

$$u = g_1, \quad \frac{\partial u}{\partial \hat{n}} = g_2 \ \text{ on } \ \partial\Omega, \tag{1.2}$$

where $\frac{\partial u}{\partial \hat{n}}$ denotes the outward normal derivative and g_1 and g_2 are given functions. In the context of the plate bending problem the case $g_1 = g_2 = 0$ corresponds to a clamped boundary.

Numerical schemes for solving (1.1)–(1.2) either approach the problem directly, or reformulate it as a mixed formulation (i.e., solve a system of two second-order problems). The advantages of using the former approach include better asymptotic

[†]School of Mathematics, The University of Manchester, Manchester, M13 9PL, UK, (jennifer.pestana@manchester.ac.uk, rlmuddle@googlemail.com, matthias.heil@manchester.ac.uk, francoise.tisseur@manchester.ac.uk)

[‡]School of Computer Science, The University of Manchester, Manchester, M13 9PL, UK (milan.mihajlovic@manchester.ac.uk)

[§]These authors were supported by Engineering and Physical Sciences Research Council grant EP/I005293. The work of the fourth author was supported by a Royal Society-Wolfson Research Merit Award.

accuracy for the same level of grid resolution [1, Theorem 5.4], [10, Theorems 6.1.6 and 7.1.6 and p. 392], and a symmetric positive definite coefficient matrix for the discrete problem. Conversely, for the mixed formulation, discretisation (by a finite difference or finite element method, for example) results in a linear algebraic system that is symmetric, but indefinite.

In this paper we consider a conforming C^1 finite element approach [26], for which the standard weak form is to find $u \in H^2(\Omega)$ satisfying (1.2) such that

$$\int_\Omega \nabla^2 u \, \nabla^2 v \, d\Omega = \int_\Omega f v \, d\Omega \tag{1.3}$$

holds for all test functions $v \in H_0^2(\Omega)$, where $H_0^2(\Omega) = \{v \in H^2(\Omega) \,|\, v = \frac{\partial v}{\partial \hat{n}} = 0 \text{ on } \partial\Omega\}$. The discrete weak formulation is obtained by restricting (1.3) to a finite dimensional space $S(\Omega) \subset H^2(\Omega)$, for which we adopt a basis associated with the bicubic Hermite (Bogner-Fox-Schmit) finite elements [6, p. 72]; these are formed from a tensor product of one-dimensional Hermite polynomials. The C^1 continuity across element boundaries is ensured by assigning four degrees of freedom (DOF) to each node, corresponding to four different basis functions.

The finite element approximation of (1.3) is then obtained by solving a linear system $\mathcal{A}\boldsymbol{x} = \boldsymbol{b}$, where $\mathcal{A} \in \mathbb{R}^{N\times N}$ is a large, sparse and symmetric positive definite (SPD) matrix and $\boldsymbol{b} \in \mathbb{R}^N$. Such systems are usually solved by iterative methods, with the conjugate gradient (CG) method a popular choice [16, Ch. 2]. Grouping together the unknowns corresponding to the same DOF type leads to the following natural 4×4 blocking of the coefficient matrix

$$\mathcal{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{12}^T & A_{22} & A_{23} & A_{24} \\ A_{13}^T & A_{23}^T & A_{33} & A_{34} \\ A_{14}^T & A_{24}^T & A_{34}^T & A_{44} \end{bmatrix}, \tag{1.4}$$

where $A_{ij} \in \mathbb{R}^{n\times n}$, $i, j = 1, \ldots, 4$, and $N = 4n$, where n is the number of interior nodes. Since the biharmonic operator is fourth order, the two-norm condition number of the matrix $\mathcal{A}$ behaves as $\kappa(\mathcal{A}) = O(h^{-4})$, where h is the mesh parameter (assuming uniform discretisation) and we find that mesh refinement generally has a detrimental effect on the convergence speed of the CG method. This problem can be rectified by effective preconditioning.

There are a number of preconditioning strategies for conforming C^1 discretisations of (1.1)–(1.2). The proposed methods include additive Schwarz methods [15], [40], [41], BPX preconditioning [26], Steklov-Poincaré operator-based preconditioning [23], problem-specific multigrid methods, [7], [9], [19], [33], [38] and fast auxiliary space (FASP) preconditioning [39].

Block preconditioners with multigrid components have also been considered. Aksoylu and Yeter [2] develop preconditioners with blocks based on regions of high and low bending while Bjørstad [3] uses blocks arising from a separation of variables of a related problem. Peisker and Braess [29] use a blocking based on basis function types, as we do, but their preconditioner is based on a mixed formulation of the biharmonic problem. Other preconditioners for the mixed formulation use blocks associated with different differential operators and efficient preconditioners of this sort apply multigrid to the Dirichlet Laplacian blocks [31], or the Schur complement system [17].

In this paper we propose two novel preconditioners that are fully algebraic and are assembled from the element matrices in an analogous manner to the matrix $\mathcal{A}$,

making them easy to implement. The first of these preconditioners is a 2×2 block diagonal (BD) matrix. The positive definiteness of $\mathcal{A}$ and the assembly of the preconditioner from element matrices means that analysis based on the general ideas of Wathen [34], [35], [36] can be applied to demonstrate that mesh-independent convergence is guaranteed in certain cases.

The second preconditioner introduced in this paper is a computationally cheaper block bordered diagonal (BBD) approximation of the block diagonal preconditioner, that is feasible for larger problems, and that can be implemented in a cost-effective manner. For this second preconditioner we provide some spectral analysis. We then employ numerical experiments to demonstrate mesh independent convergence rates and show that it is possible to deploy off-the-shelf multigrid approximations for certain matrix blocks.

The paper is organised as follows. In Section 2 we discuss the finite element assembly process of the matrix $\mathcal{A}$ in (1.4) and relevant aspects of the conjugate gradient method. Section 3 describes the new block diagonal preconditioner. We characterise the eigenvalues of the preconditioned matrix and give conditions for mesh-independent convergence. However, the preconditioner is costly to apply. We therefore introduce a more practical block bordered diagonal preconditioner in Section 4, and provide an eigenvalue analysis. We propose an inexact version of the BBD preconditioner, which involves matrix lumping and algebraic multigrid approximation. Finally, we present numerical experiments in Section 5 that verify the effectiveness of the inexact BBD preconditioner and investigate its robustness with respect to changes in the domain and element shape.

2. Preliminaries. In this section we describe the details of the finite element assembly process for the biharmonic problem and introduce the preconditioned conjugate gradient (PCG) method.

2.1. The finite element assembly process. The analysis of the spectra of the preconditioned matrices in later sections will be based on the fact that the finite element matrix $\mathcal{A}$ in (1.4) is assembled from element contributions. In this section we describe this assembly process.

We discretise (1.3) using C^1 Hermite finite elements, defined in a reference domain with local co-ordinates $(s_1, s_2) \in \overline{\Omega} = [-1, 1]^2$. The solution within the element is represented as

$$u(s_1, s_2) = \sum_{j=1}^{4} \sum_{k=1}^{4} U_{jk}\, \bar{\psi}_{jk}(s_1, s_2),$$

where U_{jk} are the unknown coefficients and $\bar{\psi}_{jk}$ are the reference Hermitian basis functions. The subscript j represents the node number and k enumerates the DOF type, such that at node j, U_{jk} interpolates u, $\frac{\partial u}{\partial s_1}$, $\frac{\partial u}{\partial s_2}$ and $\frac{\partial^2 u}{\partial s_1 \partial s_2}$ for $k = 1, \ldots, 4$, respectively. The same basis functions are used to isoparametrically map the reference element to the actual element Ω_e.

Consider now a finite element discretisation of the domain Ω consisting of M elements and let $A_e \in \mathbb{R}^{16\times 16}$, $e = 1, \ldots, M$, be the biharmonic element matrices associated with these elements. The matrices A_e are symmetric positive semidefinite and each entry is of the form

$$(A_e)_{ij} = \int_{\overline{\Omega}} \nabla^2 \bar{\psi}_{i_1 i_2} \nabla^2 \bar{\psi}_{j_1 j_2} |J_e|\, d\Omega,$$

where $i = 4(i_2-1)+i_1$, $j = 4(j_2-1)+j_1$ and J_e is the element Jacobian. Consequently, multiplying A_e by a vector $\boldsymbol{u} \in \mathbb{R}^{16}$, with elements $u_j = u_{j_1 j_2}$, is equivalent to computing integrals of linear combinations of basis vectors, that is,

$$(A_e \boldsymbol{u})_i = \int_{\overline{\Omega}} \nabla^2 \bar{\psi}_{i_1 i_2} \left(\sum_{j_1,j_2=1}^{4} \nabla^2 \left(u_{j_1 j_2} \bar{\psi}_{j_1 j_2} \right) \right) d\Omega.$$

Thus, the nullspace vectors of A_e can be thought of in terms of linear combinations of certain basis functions. These nullspace basis functions are harmonic functions, i.e., functions for which the Laplacian is zero (see (1.3)). It is straightforward to verify that a basis for these harmonic functions is

$$1,\ s_1,\ s_2,\ s_1 s_2,\ s_1^2 - s_2^2,\ s_2(s_1^2 - s_2^2/3),\ s_2(s_1^2/3 - s_2^2) \text{ and } s_1 s_2 (s_1^2 - s_2^2), \tag{2.1}$$

from which the nullspace of A_e can be computed.

Now let us describe the assembly process of (1.4) mathematically. We introduce the matrix $L_e \in \mathbb{R}^{16 \times N}$ that maps the entries of A_e to entries of $\mathcal{A}$. Then

$$\mathcal{A} = \sum_{i=1}^{M} L_e^T A_e L_e = L^T \operatorname{diag}(A_e) L \in \mathbb{R}^{N \times N}, \tag{2.2}$$

where

$$L = \begin{bmatrix} L_1^T & L_2^T & \dots & L_M^T \end{bmatrix}^T \in \mathbb{R}^{16M \times N} \tag{2.3}$$

and $\operatorname{diag}(A_e)$ is a block diagonal matrix of element matrices A_e, $i = 1, \dots, M$. The matrix $\operatorname{diag}(A_e)$ is related to the differential operator and the choice of basis functions, while L provides information about the geometry and boundary conditions.

During this assembly process, unknowns corresponding to the same DOF type are grouped together and this leads to the natural blocking of the coefficient matrix as:

$$\mathcal{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{12}^T & A_{22} & A_{23} & A_{24} \\ A_{13}^T & A_{23}^T & A_{33} & A_{34} \\ A_{14}^T & A_{24}^T & A_{34}^T & A_{44} \end{bmatrix} . \quad \begin{matrix} u \\ \frac{\partial u}{\partial s_1} \\ \frac{\partial u}{\partial s_2} \\ \frac{\partial^2 u}{\partial s_1 \partial s_2} \end{matrix} \tag{2.4}$$

The unknown vector $\boldsymbol{x}$ and the right-hand side $\boldsymbol{b}$ are blocked accordingly.

2.2. The conjugate gradient method. The conjugate gradient method (CG) is perhaps the best known Krylov subspace method for solving sparse linear systems, and is suitable for systems with an SPD coefficient matrix. The relative error after k iterations of CG is bounded by [16, p. 51]

$$\frac{\|\boldsymbol{e}^{(k)}\|_{\mathcal{A}}}{\|\boldsymbol{e}^{(0)}\|_{\mathcal{A}}} \leq 2 \left(\frac{\sqrt{\alpha(\mathcal{A})} - 1}{\sqrt{\alpha(\mathcal{A})} + 1} \right)^k,$$

where $\boldsymbol{e}^{(k)} = \boldsymbol{x} - \boldsymbol{x}^{(k)}$ and $\alpha(\mathcal{A}) = \lambda_{\max}(\mathcal{A})/\lambda_{\min}(\mathcal{A})$. Since $\mathcal{A}$ is symmetric positive definite $\alpha(\mathcal{A})$ corresponds to the 2-norm condition number $\kappa(\mathcal{A})$. As mentioned in the introduction and verified numerically in Table 2.1, $\kappa(\mathcal{A}) = O(h^{-4})$. Although this

TABLE 2.1
Extremal eigenvalues and 2-norm condition number of $\mathcal{A}$ for uniform meshes as a function of the problem size N.

Elements	4×4	8×8	16×16	32×32	64×64
N	36	196	900	3844	15876
$\lambda_{\min}$	56.20	18.45	4.94	1.26	0.32
$\lambda_{\max}$	1287	5705	23399	94179	377295
$\kappa(\mathcal{A})$	23	309	4735	74912	1.20×10^6

bound may be pessimistic, we do see a deterioration in convergence speed of the CG solver as the mesh is refined (see the computations in Section 5).

The effective condition number [4], [30]

$$\kappa_{\text{eff}} = \frac{\|\boldsymbol{b}\|_2}{\lambda_{\min}(\mathcal{A})\|\boldsymbol{x}\|_2}$$

can better describe the effect of perturbations of $\mathcal{A}$ and the right-hand side $\boldsymbol{b}$ on the solution $\boldsymbol{x}$, but does not describe the convergence rate of the conjugate gradient method (which is determined by a complex interaction between the spectrum of $\mathcal{A}$ and the right-hand side). Li, Huang and Huang [24] have shown that for the biharmonic equation and Hermite elements the effective condition number is $O(h^{-3.5})$ for general problems but can be as low as $O(1)$ for certain boundary conditions. We observe this $O(1)$ behaviour for the homogeneous Dirichlet biharmonic problem (1.1) with $f = 1$ when square, stretched or deformed elements are used (see Figure 5.2).

The problem of slow convergence rates can be alleviated by solving an equivalent preconditioned system $\mathcal{P}^{-\frac{1}{2}}\mathcal{A}\mathcal{P}^{-\frac{1}{2}}\boldsymbol{y} = \mathcal{P}^{-\frac{1}{2}}\boldsymbol{b}$ with $\boldsymbol{x} = \mathcal{P}^{-\frac{1}{2}}\boldsymbol{y}$, where $\mathcal{P} \in \mathbb{R}^{N\times N}$ is SPD. Note that the CG algorithm itself requires only a linear system solve with $\mathcal{P}$ at each iteration, i.e., the matrix $\mathcal{P}^{-\frac{1}{2}}$ is never explicitly formed. The error of the preconditioned CG iterates can be bounded by

$$\frac{\|\boldsymbol{e}^{(k)}\|_{\mathcal{A}}}{\|\boldsymbol{e}^{(0)}\|_{\mathcal{A}}} \leq 2\left(\frac{\sqrt{\alpha(\mathcal{P}^{-1}\mathcal{A})}-1}{\sqrt{\alpha(\mathcal{P}^{-1}\mathcal{A})}+1}\right)^k.$$

The error bound shows that the convergence of the CG method is accelerated when the condition number of the preconditioned matrix $\mathcal{P}^{-1}\mathcal{A}$ is small. It can also be shown that fast convergence rates are achieved when the eigenvalues belong to a small number of tightly bounded clusters (see, for example, [16, Section 3.1]). If the eigenvalues of $\mathcal{P}^{-1}\mathcal{A}$ can be bounded independently of the mesh size h (and possibly other problem parameters) then $\mathcal{P}$ is an optimal preconditioner, in the sense of convergence of the conjugate gradient method. If, in addition, linear systems involving $\mathcal{P}$ can be solved in a manner that scales linearly with the problem size then we have an optimal solver.

3. An ideal preconditioner. We first consider the block diagonal preconditioner

$$\mathcal{P}_{BD} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & \\ A_{12}^T & A_{22} & A_{23} & \\ A_{13}^T & A_{23}^T & A_{33} & \\ & & & A_{44} \end{bmatrix}. \tag{3.1}$$

Since any principal submatrix of an SPD matrix $\mathcal{A}$ is itself SPD [21, p. 397], the preconditioner $\mathcal{P}_{BD} \in \mathbb{R}^{N\times N}$ is also symmetric positive definite. Additionally, $\mathcal{P}_{BD}$

is formed from a subset of the block matrices A_{ij} of $\mathcal{A}$ and so it is possible to assemble $\mathcal{P}_{BD}$ from the element matrix contributions in a manner analogous to that described in Section 2.1. Thus,

$$\mathcal{P}_{BD} = L^T \operatorname{diag}(P_e) L, \tag{3.2}$$

where P_e is obtained from A_e, with values that would be assembled into A_{i4} or A_{i4}^T set to zero for $i = 1, 2, 3$. The element contribution to the preconditioner (henceforth, the element preconditioner) P_e, like A_e, is symmetric positive semidefinite, but it has rank 11 rather than 8. Straightforward computation shows that the nullspace of P_e is spanned by vectors corresponding to 1, s_1, s_2, $s_1^2 - s_2^2$ and $s_1^3(2s_2 - 1) + s_2^3(1 - 2s_1) + 3s_1 s_2(s_2 - s_1)$. Note that the last of these functions is a combination of the last three functions in (2.1). Consequently, the nullspace of $\operatorname{diag}(P_e)$ is contained in the nullspace of $\operatorname{diag}(A_e)$ as stated in the following lemma, which will be relevant in the subsequent analysis.

LEMMA 3.1. *Let* $\operatorname{diag}(P_e)$ *be as in* (3.2) *and* $\operatorname{diag}(A_e)$ *be as in* (2.2). *Then* $\operatorname{null}(P_e) \subset \operatorname{null}(A_e)$ *and* $\operatorname{null}(\operatorname{diag}(P_e)) \subset \operatorname{null}(\operatorname{diag}(A_e))$.

We investigate analytically the spectral properties of $\mathcal{P}_{BD}^{-1}\mathcal{A}$. For convenience we introduce the notation

$$\mathcal{A} = \left[\begin{array}{ccc|c} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{12}^T & A_{22} & A_{23} & A_{24} \\ A_{13}^T & A_{23}^T & A_{33} & A_{34} \\ \hline A_{14}^T & A_{24}^T & A_{34}^T & A_{44} \end{array}\right] = \begin{bmatrix} A & B \\ B^T & A_{44} \end{bmatrix}, \quad \mathcal{P}_{BD} = \begin{bmatrix} A & \\ & A_{44} \end{bmatrix}. \tag{3.3}$$

Then the eigenvalues of $\mathcal{P}_{BD}^{-1}\mathcal{A}$ are characterised by the following theorem.

THEOREM 3.2. *Assume that* $\operatorname{rank}(B) = r$ *in* (3.3). *Then* $\mathcal{P}_{BD}^{-1}\mathcal{A} \in \mathbb{R}^{N\times N}$, *with* $\mathcal{A}$ *and* $\mathcal{P}_{BD}$ *given by* (1.4) *and* (3.1), *respectively, has* $N - 2r$ *unit eigenvalues. The remaining* $2r$ *eigenvalues* λ *satisfy*

$$0 < 1 - \sqrt{\mu_{\max}} \le \lambda \le 1 + \sqrt{\mu_{\max}} < 2,$$

where $\mu_{\max} \in (0, 1)$ *is the largest eigenvalue of* $A_{44}^{-1} B^T A^{-1} B$.

Proof. Since $\mathcal{P}_{BD}^{-1}\mathcal{A}$ is similar to $\mathcal{P}_{BD}^{-\frac{1}{2}}\mathcal{A}\mathcal{P}_{BD}^{-\frac{1}{2}}$, which is symmetric positive definite, any eigenvalue λ of $\mathcal{P}_{BD}^{-1}\mathcal{A}$ is real and positive. Using (3.3), we see that λ satisfies

$$A\boldsymbol{u} + B\boldsymbol{v} = \lambda A\boldsymbol{u}, \tag{3.4}$$

$$B^T\boldsymbol{u} + A_{44}\boldsymbol{v} = \lambda A_{44}\boldsymbol{v}, \tag{3.5}$$

where $\boldsymbol{u} \in \mathbb{R}^{3n}$ and $\boldsymbol{v} \in \mathbb{R}^n$ are not simultaneously zero and $N = 4n$.

If $\lambda = 1$ then (3.4) implies that $B\boldsymbol{v} = \boldsymbol{0}$, i.e., that $\boldsymbol{v} = \boldsymbol{0}$ or $\boldsymbol{v} \in \operatorname{null}(B)$. We can find $n - r$ linearly independent vectors in $\operatorname{null}(B)$ for which (3.4) and (3.5) are satisfied with $\boldsymbol{u} = \boldsymbol{0}$. Otherwise, $\boldsymbol{v} = \boldsymbol{0}$ and it follows from (3.5) that $\boldsymbol{u} \in \operatorname{null}(B^T)$. Since we can find $3n - r$ linearly independent vectors in $\operatorname{null}(B^T)$, we have that one is an eigenvalue of $\mathcal{P}_{BD}^{-1}\mathcal{A}$ with multiplicity $4n - 2r = N - 2r$.

If $\lambda \neq 1$ then (3.4) implies that $(\lambda - 1)^{-1} A^{-1} B\boldsymbol{v} = \boldsymbol{u}$ and substituting for $\boldsymbol{u}$ in (3.5) gives that

$$A_{44}^{-1} B^T A^{-1} B\boldsymbol{v} = (\lambda - 1)^2 \boldsymbol{v}.$$

From this we see that non-unit eigenvalues λ of $\mathcal{P}_{BD}^{-1}\mathcal{A}$ are given by $\lambda = 1 + \sqrt{\mu}$ and $\lambda = 1 - \sqrt{\mu}$, where μ is a nonzero eigenvalue of $A_{44}^{-1} B^T A^{-1} B$. Also, since $\mathcal{A}$ is positive definite, $0 < \mu < 1$ [21, Theorem 7.7.7]. The result follows. ∎

The rank of B is at most n, so at least $2n$ eigenvalues are equal to one, while the largest non-unit eigenvalue is less than two, regardless of the mesh size. The focus of the remainder of this section is to bound the smallest non-unit eigenvalue, since doing so ensures mesh-independent convergence.

To bound the smallest eigenvalue of $\mathcal{P}_{BD}^{-1}\mathcal{A}$ we adapt the analysis of Wathen [34], [35], [36] to our case. The basic idea is to determine the eigenvalues of the preconditioned element matrix $\operatorname{diag}(P_e)^{-1}\operatorname{diag}(A_e)$ and to then obtain mesh-independent bounds using Rayleigh quotients. In our case this approach is complicated by the fact that the singular matrices $\operatorname{diag}(A_e)$ and $\operatorname{diag}(P_e)$ have nullspaces of different dimensions. However, we can still apply the general methodology since we know from Lemma 3.1 that $\operatorname{null}(\operatorname{diag}(P_e)) \subset \operatorname{null}(\operatorname{diag}(A_e)) \subset \mathbb{R}^{16M}$.

To deal with the different nullspaces involved it is useful to introduce certain subspaces of $\mathbb{R}^{16M}$. Specifically, we define:

$$\begin{aligned} \mathcal{R} &:= \operatorname{range}(\operatorname{diag}(A_e)), \qquad \mathcal{Z} := \operatorname{null}(\operatorname{diag}(A_e)), \\ \mathcal{N} &:= \operatorname{null}(\operatorname{diag}(P_e)), \qquad \mathcal{M} := \mathcal{Z} \cap \mathcal{N}^{\perp}, \end{aligned} \tag{3.6}$$

where $\mathcal{N}^{\perp}$ is the space of all vectors orthogonal to vectors in $\mathcal{N}$. With these spaces, $\mathbb{R}^{16M} = \mathcal{R} + \mathcal{N} + \mathcal{M}$, with $\mathcal{N} \subset \mathcal{Z}$. Furthermore, since the matrices $\operatorname{diag}(A_e)$ and $\operatorname{diag}(P_e)$ are block diagonal, the basis vectors of $\mathcal{R}$, $\mathcal{N}$, $\mathcal{Z}$ and $\mathcal{M}$ can be constructed from their element contributions.

In addition to the spaces defined above, we require the following lemma that shows that nonzero vectors in $\mathbb{R}^N$ cannot be mapped to $\mathcal{Z}$ by the connectivity matrix.

LEMMA 3.3. *If $\boldsymbol{x} \in \mathbb{R}^N$ is a nonzero vector then $L\boldsymbol{x} \notin \mathcal{Z}$, where L, $\operatorname{diag}(A_e)$, $\operatorname{diag}(P_e)$ and $\mathcal{Z}$ are defined by* (2.3), (2.2), (3.2) *and* (3.6)*, respectively.*

Proof. Both $\mathcal{A}$ and $\mathcal{P}_{BD}$ are positive definite, which implies that for any $\boldsymbol{x} \neq \boldsymbol{0}$,

$$\mathcal{A}\boldsymbol{x} = L^T \operatorname{diag}(A_e)(L\boldsymbol{x}) \neq 0 \text{ and } \mathcal{P}_{BD}\boldsymbol{x} = L^T \operatorname{diag}(P_e)(L\boldsymbol{x}) \neq 0.$$

We know from Lemma 3.1 that $\operatorname{null}(\operatorname{diag}(P_e)) \subset \operatorname{null}(\operatorname{diag}(A_e)) = \mathcal{Z}$, so $L\boldsymbol{x} \notin \mathcal{Z}$. ∎

Both $\mathcal{A}$ and $\mathcal{P}_{BD}$ are positive definite and so $\lambda_{\min}(\mathcal{P}_{BD}^{-1}\mathcal{A})$ has the variational characterisation [28, Chapters 1 and 15]

$$\lambda_{\min}(\mathcal{P}_{BD}^{-1}\mathcal{A}) = \min_{\boldsymbol{x}\neq 0} \frac{\boldsymbol{x}^T\mathcal{A}\boldsymbol{x}}{\boldsymbol{x}^T\mathcal{P}_{BD}\boldsymbol{x}} = \min_{\substack{\boldsymbol{y}=L\boldsymbol{x}, \\ \boldsymbol{x}\neq 0}} \frac{\boldsymbol{y}^T \operatorname{diag}(A_e)\boldsymbol{y}}{\boldsymbol{y}^T \operatorname{diag}(P_e)\boldsymbol{y}}.$$

Let $\boldsymbol{y} = \boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}} + \boldsymbol{y}_{\mathcal{N}}$, where $\boldsymbol{y}_{\mathcal{R}} \in \mathcal{R}$, $\boldsymbol{y}_{\mathcal{N}} \in \mathcal{N}$, and $\boldsymbol{y}_{\mathcal{M}} \in \mathcal{M}$ with $\mathcal{R}$, $\mathcal{N}$ and $\mathcal{M}$ defined in (3.6). Lemma 3.3 shows that $\boldsymbol{y}_{\mathcal{R}} \neq \boldsymbol{0}$ and so

$$\lambda_{\min}(\mathcal{P}_{BD}^{-1}\mathcal{A}) = \min_{\substack{\boldsymbol{y}=\boldsymbol{y}_{\mathcal{R}}+\boldsymbol{y}_{\mathcal{M}} \\ \boldsymbol{y}_{\mathcal{R}}\neq\boldsymbol{0}}} \frac{\boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(A_e)\boldsymbol{y}_{\mathcal{R}}}{(\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}})^T \operatorname{diag}(P_e)(\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}})}. \tag{3.7}$$

This appears problematic because, without any restriction on the size of $\boldsymbol{y}_{\mathcal{M}}$, the smallest eigenvalue $\lambda_{\min}(\mathcal{P}_{BD}^{-1}\mathcal{A})$ could asymptotically tend to zero. To prevent this, we must somehow bound the size of the denominator of (3.7). This is achieved by the next result, provided that $\boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{R}} \geq \delta\, \boldsymbol{y}_{\mathcal{M}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{M}}$ for some $\delta \geq \delta^* > 0$, a condition that we verified numerically in Table 3.3 for the regular and stretched grids depicted in Figure 3.1.

LEMMA 3.4. *Let $\boldsymbol{y} = L\boldsymbol{x}$, $\boldsymbol{x} \in \mathbb{R}^N$, $\boldsymbol{x} \neq \boldsymbol{0}$ be decomposed as*

$$\boldsymbol{y} = \boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}} + \boldsymbol{y}_{\mathcal{N}}, \tag{3.8}$$

where $\boldsymbol{y}_{\mathcal{R}} \in \mathcal{R}$, $\boldsymbol{y}_{\mathcal{N}} \in \mathcal{N}$ *and* $\boldsymbol{y}_{\mathcal{M}} \in \mathcal{M}$ *with* $\mathcal{R}$, $\mathcal{N}$ *and* $\mathcal{M}$ *defined in* (3.6). *Additionally, assume that*

$$\boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{R}} \geq \delta\, \boldsymbol{y}_{\mathcal{M}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{M}} \tag{3.9}$$

for some $\delta \geq \delta^* > 0$. *Then*

$$(\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}})^T \operatorname{diag}(P_e)(\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}}) \leq \zeta \boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{R}}, \tag{3.10}$$

where $\zeta = 2\,(1 + 1/\delta)$.

Proof. From Lemma 3.3 we know that $\boldsymbol{y}_{\mathcal{R}} \neq \boldsymbol{0}$. Since $\operatorname{diag}(P_e)$ is symmetric positive semidefinite it has a semidefinite square root and there are vectors $\boldsymbol{a} \in \mathbb{R}^{16M}$ and $\boldsymbol{b} \in \mathbb{R}^{16M}$ for which $(\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}})^T \operatorname{diag}(P_e)(\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}}) = (\boldsymbol{a} + \boldsymbol{b})^T(\boldsymbol{a} + \boldsymbol{b})$.

Now, for any vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ of the same dimension

$$0 \leq \|\boldsymbol{a} - \boldsymbol{b}\|_2^2 = (\boldsymbol{a} - \boldsymbol{b})^T(\boldsymbol{a} - \boldsymbol{b}) = 2(\boldsymbol{a}^T\boldsymbol{a} + \boldsymbol{b}^T\boldsymbol{b}) - (\boldsymbol{a} + \boldsymbol{b})^T(\boldsymbol{a} + \boldsymbol{b})$$

or $(\boldsymbol{a} + \boldsymbol{b})^T(\boldsymbol{a} + \boldsymbol{b}) \leq 2(\boldsymbol{a}^T\boldsymbol{a} + \boldsymbol{b}^T\boldsymbol{b})$. Thus,

$$(\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}})^T \operatorname{diag}(P_e)(\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}}) \leq 2(\boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{M}}). \tag{3.11}$$

Combining (3.11) with (3.9) gives (3.10). ∎

We have been unable to prove that (3.9) holds for all meshes for the Dirichlet biharmonic problem, since it does not appear straightforward to remove the influence of the connectivity matrix L. However, there is strong numerical evidence to suggest that the assertion holds. In particular, let $P_{\mathcal{R}}$ and $P_{\mathcal{M}}$ be orthogonal projectors onto $\mathcal{R}$ and $\mathcal{M}$, respectively. Then for any vector $\boldsymbol{y} = L\boldsymbol{x}$, $\boldsymbol{x} \neq \boldsymbol{0}$,

$$\frac{\boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{R}}}{\boldsymbol{y}_{\mathcal{M}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{M}}} = \frac{\boldsymbol{x}^T L^T P_{\mathcal{R}}^T \operatorname{diag}(P_e) P_{\mathcal{R}} L\boldsymbol{x}}{\boldsymbol{x}^T L^T P_{\mathcal{M}}^T \operatorname{diag}(P_e) P_{\mathcal{M}} L\boldsymbol{x}} \geq \delta_{\min},$$

where

$$\delta_{\min} = \lambda_{\min}\big((L^T P_{\mathcal{M}}^T \operatorname{diag}(P_e) P_{\mathcal{M}} L)^{-1}(L^T P_{\mathcal{R}}^T \operatorname{diag}(P_e) P_{\mathcal{R}} L)\big). \tag{3.12}$$

The value of $\delta_{\min}$ is tabulated for different uniform meshes in Table 3.1. From this we see that for square elements $\delta_{\min}$ appears to tend to 1.05, so that (3.9) is satisfied for uniformly refined meshes of square elements with $\delta > 1.05$.

With these results in hand, we now bound the smallest eigenvalue of $\mathcal{P}_{BD}^{-1}\mathcal{A}$. Under the assumption (3.9), we combine the decomposition (3.8) with Lemma 3.4 to give that, for any $\boldsymbol{y} = L\boldsymbol{x}$, $\boldsymbol{x} \neq \boldsymbol{0}$,

$$\frac{\boldsymbol{y}^T \operatorname{diag}(A_e)\boldsymbol{y}}{\boldsymbol{y}^T \operatorname{diag}(P_e)\boldsymbol{y}} = \frac{\boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(A_e)\boldsymbol{y}_{\mathcal{R}}}{(\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}})^T \operatorname{diag}(P_e)(\boldsymbol{y}_{\mathcal{R}} + \boldsymbol{y}_{\mathcal{M}})} \geq \frac{1}{\zeta} \frac{\boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(A_e)\boldsymbol{y}_{\mathcal{R}}}{\boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{R}}}, \tag{3.13}$$

where by Lemma 3.3, $\boldsymbol{y}_{\mathcal{R}} \neq \boldsymbol{0}$. It follows from (3.7) that

$$\lambda_{\min}(\mathcal{P}_{BD}^{-1}\mathcal{A}) \geq \frac{\theta}{\zeta}, \quad \theta := \min_{\substack{\boldsymbol{y}_{\mathcal{R}} \in \mathcal{R} \\ \boldsymbol{y}_{\mathcal{R}} \neq \boldsymbol{0}}} \frac{\boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(A_e)\boldsymbol{y}_{\mathcal{R}}}{\boldsymbol{y}_{\mathcal{R}}^T \operatorname{diag}(P_e)\boldsymbol{y}_{\mathcal{R}}}. \tag{3.14}$$

Since $\operatorname{diag}(P_e)$, $\operatorname{diag}(A_e)$ and the projector $P_{\mathcal{R}}$ onto $\mathcal{R}$ are block diagonal, the above minimisation over all nonzero $\boldsymbol{y}_{\mathcal{R}}$ can be carried out using individual element matrices.

TABLE 3.1
Smallest ($\lambda_{\min}$) and largest ($\lambda_{\max}$) eigenvalues of the preconditioned operator $\mathcal{P}_{BD}^{-1}\mathcal{A}$, $\operatorname{rank}(B)$ from Theorem 3.2 and $\delta_{\min}$ from (3.12) for a sequence of uniformly refined grids and square elements.

Elements	4×4	8×8	16×16	32×32	64×64
N	36	196	900	3844	15876
$\lambda_{\min}$	0.72	0.64	0.61	0.60	0.60
$\lambda_{\max}$	1.28	1.36	1.39	1.40	1.40
$\operatorname{rank}(B)$	9	49	225	961	3969
$\delta_{\min}$	1.17	1.06	1.05	1.05	1.05

We computed the minimum for our element matrices and found that θ in (3.14) is larger than 0.046 for square elements. Since $\zeta < 3.91$ for square elements, we have that $\lambda_{\min}(\mathcal{P}_{BD}^{-1}\mathcal{A}) > 0.0118$. Combining (3.9) with Theorem 3.2 gives the following bounds on the eigenvalues of $\mathcal{P}_{BD}^{-1}\mathcal{A}$.

COROLLARY 3.5. *Let $\mathcal{A}$ and $\mathcal{P}_{BD}$ be as in (1.4) and (3.1) and assume that (3.9) holds. Then for square elements the eigenvalues λ of $\mathcal{P}_{BD}^{-1}\mathcal{A}$ satisfy $0.0118 < \lambda \leq 2$ and $\kappa_2(\mathcal{P}_{BD}^{-1}\mathcal{A}) < 170$ independently of the mesh spacing parameter h.*

Comparison with Table 3.1 shows that the bounds in Corollary 3.5 are pessimistic. However, combined with the high multiplicity of the unit eigenvalue, they show that we can expect fast convergence of preconditioned CG whenever (3.9) is satisfied.

We also tested assumption (3.9) for meshes of elements stretched in the x_1 direction, with a denoting the ratio of the length of the horizontal side to the length of the vertical, as shown in Figure 3.1. We see from Table 3.2 that $\delta_{\min}$ decreases as the aspect ratio increases but that, for a fixed aspect ratio, $\delta_{\min}$ seems to tend to a constant as the mesh is refined. On the other hand, θ in (3.14) actually increases with a (see Table 3.3). The net result is the eigenvalue bound θ/ζ in Table 3.3 that slowly decreases as the aspect ratio increases but is asymptotically independent of the mesh width, and that qualitatively captures the behaviour of the smallest eigenvalue.

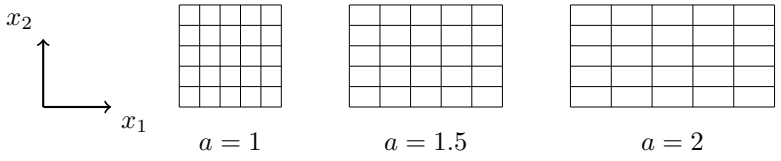

FIG. 3.1. *Stretched elements. The domain is stretched in the x_1 direction and the deformation is described by the aspect ratio a.*

4. A practical preconditioner. Although the preconditioner $\mathcal{P}_{BD}$ in (3.1) has favourable spectral properties, it is prohibitively expensive to apply for large problems, since it requires the solution of linear subsystems involving the $3n \times 3n$ matrix A. We will now investigate the block bordered diagonal (BBD) preconditioner

$$\mathcal{P}_{BBD} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & \\ A_{12}^T & A_{22} & & \\ A_{13}^T & & A_{33} & \\ & & & A_{44} \end{bmatrix} \tag{4.1}$$

that is formed by omitting A_{23} and A_{23}^T from $\mathcal{P}_{BD}$. Unlike $\mathcal{P}_{BD}$, the symmetric positive definiteness of $\mathcal{A}$ is not enough to guarantee that $\mathcal{P}_{BBD}$ is positive definite.

TABLE 3.2
Smallest ($\lambda_{\min}$) and largest ($\lambda_{\max}$) eigenvalues of the preconditioned operator $\mathcal{P}_{BD}^{-1}\mathcal{A}$ and $\delta_{\min}$ from (3.12) for stretched grids with different aspect ratios a.

	Elements	4×4	8×8	16×16	32×32	64×64
	N	36	196	900	3844	15876
	$\lambda_{\min}$	0.62	0.52	0.50	0.49	0.49
$a=1.5$	$\lambda_{\max}$	1.38	1.48	1.5	1.51	1.51
	$\delta_{\min}$	0.85	0.59	0.52	0.51	0.50
	$\lambda_{\min}$	0.47	0.38	0.35	0.34	0.34
$a=2$	$\lambda_{\max}$	1.53	1.62	1.65	1.66	1.66
	$\delta_{\min}$	0.58	0.34	0.28	0.27	0.27
	$\lambda_{\min}$	0.36	0.27	0.25	0.24	0.24
$a=2.5$	$\lambda_{\max}$	1.64	1.73	1.75	1.76	1.76
	$\delta_{\min}$	0.41	0.22	0.18	0.17	0.16

TABLE 3.3
The values of δ^* and ζ in Lemma 3.4, θ in (3.14) and the lower bound θ/ζ on $\lambda_{\min}(\mathcal{P}_{BD}^{-1}\mathcal{A})$ for stretched elements with different aspect ratios a.

a	1	1.5	2	2.5
δ^*	1.05	0.50	0.27	0.16
ζ	3.9	6.0	9.5	14
θ	0.047	0.053	0.062	0.068
θ/ζ	0.011	0.008	0.006	0.004

However, $\mathcal{P}_{BBD}$ was found to be positive definite in all the numerical experiments (performed with square, stretched and deformed meshes) presented in Section 5 below. Compared to the even simpler block Jacobi preconditioner

$$\mathcal{P}_J = \begin{bmatrix} A_{11} & & & \\ & A_{22} & & \\ & & A_{33} & \\ & & & A_{44} \end{bmatrix}, \tag{4.2}$$

$\mathcal{P}_{BBD}$ retains the coupling between u and both first derivative ($\frac{\partial u}{\partial s_1}$ and $\frac{\partial u}{\partial s_2}$) DOFs. We will see in Section 5 that this is essential to obtaining low and asymptotically constant iteration counts as the mesh is refined.

The action of $\mathcal{P}_{BBD}^{-1}$ on a vector can be computed by means of the unsymmetric UL decomposition

$$\mathcal{P}_{BBD} = UL = \begin{bmatrix} I & A_{12}A_{22}^{-1} & A_{13}A_{33}^{-1} & \\ & I & & \\ & & I & \\ & & & I \end{bmatrix} \begin{bmatrix} S_{11} & & & \\ A_{12}^T & A_{22} & & \\ A_{13}^T & & A_{33} & \\ & & & A_{44} \end{bmatrix}, \tag{4.3}$$

where

$$S_{11} = A_{11} - A_{12}A_{22}^{-1}A_{12}^T - A_{13}A_{33}^{-1}A_{13}^T. \tag{4.4}$$

Note that the solve $U\boldsymbol{w} = \boldsymbol{v}$ can be performed in a block parallel manner.

The remainder of this section is devoted to understanding the spectral properties of $\mathcal{P}_{BBD}^{-1}\mathcal{A}$ and deriving an approximation that can be implemented in a cost-optimal manner.

4.1. Eigenvalue analysis. The block structure of $\mathcal{P}_{BBD}$ and, in particular, the indefiniteness of the element matrices used in its assembly prevent us from applying the previously introduced analysis to bound the spectrum of $\mathcal{P}_{BBD}^{-1}\mathcal{A}$. Instead, we consider the eigenvalues of $\mathcal{P}_{BBD}^{-1}\mathcal{P}_{BD}$ and then apply the bounds

$$\lambda_{\min}(\mathcal{P}_{BBD}^{-1}\mathcal{P}_{BD})\lambda_{\min}(\mathcal{P}_{BD}^{-1}\mathcal{A}) \leq \lambda(\mathcal{P}_{BBD}^{-1}\mathcal{A}) \leq \lambda_{\max}(\mathcal{P}_{BBD}^{-1}\mathcal{P}_{BD})\lambda_{\max}(\mathcal{P}_{BD}^{-1}\mathcal{A}), \tag{4.5}$$

which follow from the Courant-Fischer theorem [21, Theorem 4.2.11], in conjunction with the bounds in Corollary 3.5. The eigenvalues of $\mathcal{P}_{BBD}^{-1}\mathcal{P}_{BD}$ are given in the following lemma.

LEMMA 4.1. *Let* $\operatorname{rank}(A_{23}) = s$. *Then 1 is an eigenvalue of* $\mathcal{P}_{BBD}^{-1}\mathcal{P}_{BD}$ *with multiplicity* $N - 2s$ *while the remaining* $2s$ *eigenvalues* η *satisfy*

$$(G - FA_{11}^{-1}F^T)\boldsymbol{v} = \eta(\widetilde{G} - FA_{11}^{-1}F^T)\boldsymbol{v}, \tag{4.6}$$

where $\boldsymbol{v} \neq \mathbf{0}$,

$$F^T = \begin{bmatrix} A_{12} & A_{13} \end{bmatrix}, \quad G = \begin{bmatrix} A_{22} & A_{23} \\ A_{23}^T & A_{33} \end{bmatrix} \text{ and } \widetilde{G} = \begin{bmatrix} A_{22} & \\ & A_{33} \end{bmatrix}. \tag{4.7}$$

Proof. In the notation of (3.3)

$$\mathcal{P}_{BBD} = \begin{bmatrix} \widetilde{A} & \\ & A_{44} \end{bmatrix}, \quad \widetilde{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{12}^T & A_{22} & \\ A_{13}^T & & A_{33} \end{bmatrix}$$

and

$$\mathcal{P}_{BBD}^{-1}\mathcal{P}_{BD} = \begin{bmatrix} \widetilde{A}^{-1}A & \\ & I_n \end{bmatrix},$$

where I_n is the identity matrix of dimension n. This shows that 1 is an eigenvalue of $\mathcal{P}_{BBD}^{-1}\mathcal{P}_{BD}$ with multiplicity at least n.

To obtain the remaining $3n$ eigenvalues let us further partition $\widetilde{A}$ and A as

$$\widetilde{A} = \begin{bmatrix} A_{11} & F^T \\ F & \widetilde{G} \end{bmatrix}, \qquad A = \begin{bmatrix} A_{11} & F^T \\ F & G \end{bmatrix},$$

where F and G are as in (4.7). Then, the result is obtained by a straightforward extension of Theorem 3.1 of Dollar et al. [13] to the case of rank-deficient F, which we sketch out for completeness.

The eigenvalues η of $\widetilde{A}^{-1}A$ satisfy

$$A_{11}\boldsymbol{u} + F^T\boldsymbol{v} = \eta A_{11}\boldsymbol{u} + \eta F^T\boldsymbol{v}, \tag{4.8}$$

$$F\boldsymbol{u} + G\boldsymbol{v} = \eta F\boldsymbol{u} + \eta\widetilde{G}\boldsymbol{v}, \tag{4.9}$$

where $\boldsymbol{u} \in \mathbb{R}^n$ and $\boldsymbol{v} \in \mathbb{R}^{2n}$ are not simultaneously zero. From (4.8) we see that either $\eta = 1$ or $A_{11}\boldsymbol{u} + F^T\boldsymbol{v} = \mathbf{0}$. If $\eta = 1$ then, letting $\boldsymbol{v} = [\boldsymbol{v}_1^T \ \boldsymbol{v}_2^T]^T$ with $\boldsymbol{v}_1, \boldsymbol{v}_2 \in \mathbb{R}^n$, we find that there are $n - s$ linearly independent vectors $\boldsymbol{v}_1 \in \operatorname{null}(A_{23})$ for which (4.8) and (4.9) are satisfied with $\boldsymbol{v}_2 = \boldsymbol{u} = \mathbf{0}$. Similarly, there are $n-s$ linearly independent vectors $\boldsymbol{v}_2 \in \operatorname{null}(A_{23}^T)$ for which (4.8) and (4.9) are satisfied with $\boldsymbol{v}_1 = \boldsymbol{u} = \mathbf{0}$. Otherwise, $\boldsymbol{v} = \mathbf{0}$ and we can find n linearly independent vectors $\boldsymbol{u} \neq \mathbf{0}$. Combining

TABLE 4.1
Computed smallest ($\lambda_{\min}$) and largest ($\lambda_{\max}$) eigenvalues of the preconditioned operator $\mathcal{P}_{BBD}^{-1}\mathcal{A}$ as well as $\operatorname{rank}(A_{23})$ *for a sequence of uniformly refined grids of square elements.*

Elements	4×4	8×8	16×16	32×32	64×64
N	36	196	900	3844	15876
$\lambda_{\min}$	0.72	0.62	0.58	0.56	0.55
$\lambda_{\max}$	1.27	1.38	1.40	1.41	1.41
$\operatorname{rank}(A_{23})$	8	48	224	960	3968

these results shows that $\eta = 1$ with multiplicity $3n-2s$. If $\eta \neq 1$ then $\boldsymbol{u} = -A_{11}^{-1}F^T\boldsymbol{v}$ and substituting into (4.9) gives (4.6). □

Similarly to Theorem 3.2 we see that $\lambda = 1$ is an eigenvalue of $\mathcal{P}_{BBD}^{-1}\mathcal{P}_{BD}$ with high multiplicity. However, we have been unable to bound the remaining $2s$ eigenvalues. In spite of this, combining Lemma 4.1 with Corollary 3.5 and (4.5) shows that most of the eigenvalues of $\mathcal{P}_{BBD}^{-1}\mathcal{A}$ lie in a bounded interval.

COROLLARY 4.2. *When square elements are used and* (3.9) *is satisfied at least $N-2s$ eigenvalues of $\mathcal{P}_{BBD}^{-1}\mathcal{A}$ lie in* $(0.0118, 2)$. *Any remaining eigenvalues lie in* $(0.0118\eta_{min}, 2\eta_{max})$, *where η_{min} and η_{max} are the smallest and largest eigenvalues of the generalised eigenvalue problem* (4.6).

REMARK 1. Analogous results hold for stretched elements if we replace 0.0118 in Corollary 4.2 by the appropriate bound θ/ζ on $\lambda_{\min}(\mathcal{P}_{BD}^{-1}\mathcal{A})$ in Table 3.3.

The extreme eigenvalues of $\mathcal{P}_{BBD}^{-1}\mathcal{A}$ are given in Table 4.1 as a function of the problem size N. From this we see that these eigenvalues do not differ greatly from the extreme eigenvalues of $\mathcal{P}_{BD}^{-1}\mathcal{A}$ (see Table 3.1), and in practice little is lost in terms of the asymptotic convergence speed by using a more practical preconditioner. The numerical evidence in Table 4.1 suggests that the extreme eigenvalues of $\mathcal{P}_{BBD}^{-1}\mathcal{A}$ appear to be bounded under mesh refinement, although we have been unable to prove this analytically.

4.2. Further simplifications. Although the block decomposition (4.3) allows the efficient application of $\mathcal{P}_{BBD}$, to achieve a preconditioner with optimal cost we require optimal solvers for linear systems involving the principal diagonal blocks S_{11}, A_{22}, A_{33} and A_{44}.

First, we consider spectrally equivalent approximations of A_{22}, A_{33} and A_{44}.

LEMMA 4.3. *Let*

$$L_{22} = \operatorname{lump}(A_{22}), \quad L_{33} = \operatorname{lump}(A_{33}), \tag{4.10}$$

where $\operatorname{lump}(H) = \{h_{ij}\}$ *with*

$$h_{ij} = \begin{cases} \sum_{k=1}^{n} h_{ik}, & i = j, \\ 0, & i \neq j. \end{cases}$$

Then for uniformly refined meshes of square elements the eigenvalues of $L_{22}^{-1}A_{22}$ and $L_{33}^{-1}A_{33}$ are contained in $[1/3, 1]$ *while the eigenvalues of* $\operatorname{diag}(A_{44})^{-1}A_{44}$ *are contained in* $[0.43, 1.24]$.

Proof. The matrices A_{22}, A_{33} and A_{44} are assembled from 4×4 submatrices $A_e^{(22)}$, $A_e^{(33)}$ and $A_e^{(44)}$ of the element matrix A_e. Additionally, the approximations L_{22} and L_{33} are assembled from lumped versions of $A_e^{(22)}$ and $A_e^{(33)}$, while $\operatorname{diag}(A_{44})$ is assembled from the diagonal of $A_e^{(44)}$. All six of these element matrices are SPD.

As a result, we can use the approach of Wathen [34] to prove the result. (Recall that a similar result was used in Section 3, where we had to deal with singular element matrices.) ∎

REMARK 2. This result is not surprising since the spectrum of $A_{22} = A_{33}$ resembles that of a scaled mass matrix, and for such matrices lumping often gives spectrally equivalent operators. In particular, for our problem $\lambda(A_{22}) = \lambda(A_{33}) \sim O(h^{-4})\lambda(M)$, where M is a mass matrix; for uniform grids this can be verified using Fourier analysis, similarly to the approach in [14, Section 1.6]. Additionally, on a uniform mesh all entries of A_{22} and A_{33} are nonnegative.

REMARK 3. For the stretched grids used in the numerical experiments in Section 5 below, L_{22}, L_{33} and $\mathrm{diag}(A_{44})$ are still spectrally equivalent to A_{22}, A_{33} and A_{44}. However, the spectral equivalence bounds deteriorate as the aspect ratio increases. For example, when $a = 2.5$ the eigenvalues of $L_{22}^{-1}A_{22}$ lie in $[0.04, 1]$, the eigenvalues of $L_{33}^{-1}A_{33}$ lie in $[0.21, 1]$ and the eigenvalues of $\mathrm{diag}(A_{44})^{-1}A_{44}$ lie in $[0.28, 2.4]$. Some off-diagonal elements of A_{22} and A_{33} were found to be negative even for an aspect ratio of $a = 1.5$.

Using (4.3) and Lemma 4.3 we approximate $\mathcal{P}_{BBD}$ by

$$\widetilde{\mathcal{P}}_{BBD} = \begin{bmatrix} I & A_{12}L_{22}^{-1} & A_{13}L_{33}^{-1} & \\ & I & & \\ & & I & \\ & & & I \end{bmatrix} \begin{bmatrix} \overline{S}_{11} & & & \\ A_{12}^T & L_{22} & & \\ A_{13}^T & & L_{33} & \\ & & & \mathrm{diag}(A_{44}) \end{bmatrix}, \tag{4.11}$$

where

$$\overline{S}_{11} = A_{11} - A_{12}L_{22}^{-1}A_{12}^T - A_{13}L_{33}^{-1}A_{13}^T. \tag{4.12}$$

The block $\overline{S}_{11}$ in (4.12) is a sparse approximation of the Schur complement S_{11} from (4.4), owing to the diagonal approximations (4.10), and can be assembled cheaply.

To apply the preconditioner $\widetilde{\mathcal{P}}_{BBD}$ within the preconditioned CG algorithm we must solve systems with $\overline{S}_{11}$, L_{22}, L_{33} and $\mathrm{diag}(A_{44})$. The last three matrices are diagonal, and hence trivial to invert. For systems with $\overline{S}_{11}$ we consider two approaches: an LU factorisation, which yields an exact solution but is not computationally optimal, or two V(2,2)-cycles of classical algebraic multigrid (AMG) with point Gauss-Seidel smoothing and Ruge-Stüben coarsening [25], which has optimal cost but leads to an inexact solution (cf. Table 5.1). Using these approximations for the Schur complement subsystem, we obtain the preconditioners $\widetilde{\mathcal{P}}_{BBD}^{[LU]}$ and $\widetilde{\mathcal{P}}_{BBD}^{[AMG]}$ in which the Schur complement subsidiary system is solved using an LU factorisation and AMG, respectively. In Table 4.2 we present the spectral properties of the preconditioned operators $\left(\widetilde{\mathcal{P}}_{BBD}^{[LU]}\right)^{-1}\mathcal{A}$ and $\left(\widetilde{\mathcal{P}}_{BBD}^{[AMG]}\right)^{-1}\mathcal{A}$. These results suggest that the spectrum of $\left(\widetilde{\mathcal{P}}_{BBD}^{[LU]}\right)^{-1}\mathcal{A}$ is bounded under mesh refinement, as we might expect from the spectral equivalence bounds in Lemma 4.3, although the eigenvalues are not as tightly clustered as those of $\mathcal{P}_{BBD}^{-1}\mathcal{A}$ in Table 3.1. However, the smallest eigenvalue of $\left(\widetilde{\mathcal{P}}_{BBD}^{[AMG]}\right)^{-1}\mathcal{A}$ decreases with mesh refinement; that is, the AMG approximation is not spectrally equivalent to $\overline{S}_{11}$.

5. Numerical experiments. In this section we examine the effectiveness of the preconditioners $\mathcal{P}_{BD}$, $\mathcal{P}_{BBD}$ and $\widetilde{\mathcal{P}}_{BBD}$ at reducing the number of conjugate gradient iterations and the computational time. Additionally, we investigate the robustness of

TABLE 4.2

Smallest ($\lambda_{\min}$) and largest ($\lambda_{\max}$) eigenvalues of $\left(\widetilde{\mathcal{P}}_{BBD}^{[LU]}\right)^{-1}\mathcal{A}$ and $\left(\widetilde{\mathcal{P}}_{BBD}^{[AMG]}\right)^{-1}\mathcal{A}$ for a sequence of uniformly refined grids. For the AMG solver we use the HSL routine MI_20 [5, 22]. Note that we were unable to obtain the eigenvalues of the largest $\widetilde{\mathcal{P}}_{BBD}^{[AMG]}$ preconditioned matrix because of memory constraints.

	Elements	4×4	8×8	16×16	32×32	64×64
	N	36	196	900	3844	15876
$\left(\widetilde{\mathcal{P}}_{BBD}^{[LU]}\right)^{-1}\mathcal{A}$	$\lambda_{\min}$	0.40	0.33	0.30	0.29	0.28
	$\lambda_{\max}$	1.25	1.30	1.31	1.32	1.32
$\left(\widetilde{\mathcal{P}}_{BBD}^{[AMG]}\right)^{-1}\mathcal{A}$	$\lambda_{\min}$	0.40	0.31	0.21	0.13	—
	$\lambda_{\max}$	1.25	1.30	1.31	1.32	—

their performance with respect to stretching of the finite elements as well as deformations and non-convexity of the domain. Throughout, we choose the homogeneous Dirichlet boundary conditions $g_1 = g_2 = 0$ in (1.2). Our default domain is the unit square domain $\Omega = [0,1]^2$ discretised by a uniform grid of square elements. Although we note that for finite element problems the stopping criterion for CG should be tied to the discretisation error, to demonstrate mesh independence we terminate the preconditioned CG method when the residual decreases in norm by six orders of magnitude, that is, $\|\boldsymbol{r}^{(k)}\|_2 \leq 10^{-6}\|\boldsymbol{r}^{(0)}\|_2$.

All AMG results in this section are obtained with two V(2,2)-cycles using Ruge-Stüben coarsening and point Gauss-Seidel smoothing. We note that different AMG methods may give different results. However, our aim is to develop an effective preconditioner that is easy to implement, and so we choose off-the-shelf codes that generally work well for finite element problems [5]. Thus, for the smaller experiments in Tables 5.1 and 5.2 we use the HSL code MI20 [5, 22] with default options, except that we change the coarsening criterion `c_fail` from 1 to 2 (and alter the number of V-cycles). For all the other experiments we use HYPRE's BOOMERAMG [20].

We first compare preconditioned CG iterations for $\mathcal{P}_{BD}$, $\mathcal{P}_{BBD}$, $\widetilde{\mathcal{P}}_{BBD}^{[LU]}$ and $\widetilde{\mathcal{P}}_{BBD}^{[AMG]}$ for smaller problems using MATLAB. For comparison, we also present iteration counts for the block Jacobi preconditioner (4.2) and AMG applied as a preconditioner to the entire block re-ordered matrix $\mathcal{A}$ from (1.4). We stress that preconditioners $\mathcal{P}_{BBD}$ in (4.3) and $\widetilde{\mathcal{P}}_{BBD}$ in (4.11) are parallelisable, like the block Jacobi preconditioner $\mathcal{P}_J$ in (4.2), as discussed in Section 4. Since the problems considered here are of relatively small dimension, in addition to measuring the norm of the residual we computed the relative error $\|\boldsymbol{x} - \boldsymbol{x}^{(k)}\|_{\mathcal{A}}/\|\boldsymbol{x}\|_{\mathcal{A}}$ in the energy norm at termination, which we found to be uniformly smaller than 1.8×10^{-7}. Computations were performed with different right-hand-sides $\boldsymbol{b}$: we used the right-hand-side from the finite element discretisation of (1.1) for $f = 1$ and a random right-hand side $\boldsymbol{b}$. Both choices result in similar behaviour, which shows that the convergence behaviour does not depend on the regularity of the forcing term. Consequently, only results for $f = 1$ are presented.

The results are given in Table 5.1, from which we see that without preconditioning the number of CG iterations increases rapidly, and appears to grow as $O(h^{-2})$. The application of the AMG and block Jacobi preconditioners reduces the number of iterations somewhat, but convergence is still mesh dependent. This is not surprising since the condition numbers of these preconditioned matrices increase as the mesh is refined, as shown in Table 5.2. (Note that we were unable to obtain the eigenvalues of the largest AMG preconditioned matrix because of memory constraints.) Conversely, the number of iterations required for $\mathcal{P}_{BD}$, $\mathcal{P}_{BBD}$ and $\widetilde{\mathcal{P}}_{BBD}^{[LU]}$ do not increase

TABLE 5.1

CG iteration counts for the unpreconditioned system, and preconditioned CG iterations counts for several different preconditioners: AMG applied to the whole matrix $\mathcal{A}$ in (1.4), $\mathcal{P}_J$, $\mathcal{P}_{BD}$, $\mathcal{P}_{BBD}$ and the two inexact versions $\widetilde{\mathcal{P}}_{BBD}^{[LU]}$,and $\widetilde{\mathcal{P}}_{BBD}^{[AMG]}$.

Elements	4×4	8×8	16×16	32×32	64×64	128×128
N	36	196	900	3844	15876	64516
Unpreconditioned	6	29	74	216	741	2741
AMG	3	9	27	82	272	864
$\mathcal{P}_J$	6	19	51	113	232	480
$\mathcal{P}_{BD}$	3	9	10	11	11	11
$\mathcal{P}_{BBD}$	4	10	11	12	13	14
$\widetilde{\mathcal{P}}_{BBD}^{[LU]}$	5	14	16	17	18	19
$\widetilde{\mathcal{P}}_{BBD}^{[AMG]}$	8	14	18	24	33	46

TABLE 5.2

Smallest ($\lambda_{\min}$) and largest ($\lambda_{\max}$) eigenvalues of the AMG and Jacobi preconditioned matrices.

	Elements	4×4	8×8	16×16	32×32	64×64
	N	36	196	900	3844	15876
$\mathcal{P}_{AMG}^{-1}\mathcal{A}$	$\lambda_{\min}$	0.92	0.17	0.01	0.0008	—
	$\lambda_{\max}$	1.00	1.00	1.00	1.00	—
$\mathcal{P}_J^{-1}\mathcal{A}$	$\lambda_{\min}$	0.18	0.04	0.009	0.002	0.0005
	$\lambda_{\max}$	1.80	2.02	2.07	2.09	2.10

markedly with mesh refinement, and our experiments later in this section for larger problems indicate asymptotically mesh-independent convergence. This is in line with the spectral analysis in previous sections and the computed eigenvalues in Tables 3.1, 4.1 and 4.2.

To explore the asymptotic behaviour of $\widetilde{\mathcal{P}}_{BBD}$ for larger problems, Figure 5.1 shows the number of iterations required for convergence of preconditioned CG, and solution times, for $\mathcal{P}_{BD}$ and the three block bordered diagonal preconditioners ($\mathcal{P}_{BBD}$, $\widetilde{\mathcal{P}}_{BBD}^{[LU]}$ and $\widetilde{\mathcal{P}}_{BBD}^{[AMG]}$). These results were obtained using a C++ implementation in OOMPH-LIB [18] with SUPERLU [12] for the direct solver. Note that times for the unpreconditioned system, and for the block Jacobi and full AMG preconditioned systems, were similar to or larger than those of the direct method, and had poor asymptotic behaviour. For this reason timings for these preconditioners are not shown in Figure 5.1.

We see from Figure 5.1 that $\mathcal{P}_{BD}$ and $\mathcal{P}_{BBD}$ give mesh independent convergence and that for both preconditioners the time to solution is lower than for the direct method. The use of $\widetilde{\mathcal{P}}_{BBD}^{[LU]}$ instead of $\mathcal{P}_{BBD}$ leads to a slight increase in iteration counts but mesh independence is retained. Furthermore, the solution times in Figure 5.1 show that this increase in iterations is more than compensated for by the drastically reduced computational cost of applying the preconditioner. A further improvement can be achieved by replacing $\widetilde{\mathcal{P}}_{BBD}^{[LU]}$ by $\widetilde{\mathcal{P}}_{BBD}^{[AMG]}$ – Figure 5.1 shows that although the iteration count increases significantly, as expected from the eigenvalue computations in Table 4.2, using AMG still reduces the solution times. This is due to the optimal cost of the AMG solver. Moreover, the solution times for $\widetilde{\mathcal{P}}_{BBD}^{[LU]}$ and $\widetilde{\mathcal{P}}_{BBD}^{[AMG]}$ scale approximately linearly with the problem size.

5.1. Robustness of the preconditioner. So far we have shown that the block bordered diagonal preconditioners $\widetilde{\mathcal{P}}_{BBD}^{[LU]}$ and $\widetilde{\mathcal{P}}_{BBD}^{[AMG]}$ are nearly optimal in terms of wall clock time for a simple test problem. We will now evaluate the robustness of our

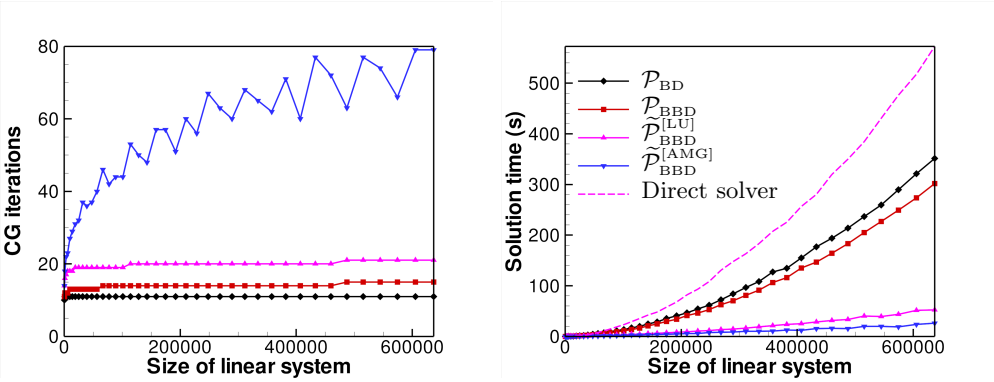

FIG. 5.1. *Number of preconditioned CG iterations (left) and solution times (in seconds). The execution time of the direct method* SUPERLU *applied to the system with coefficient matrix* (1.4) *is presented for comparison. The legend applies to both plots.*

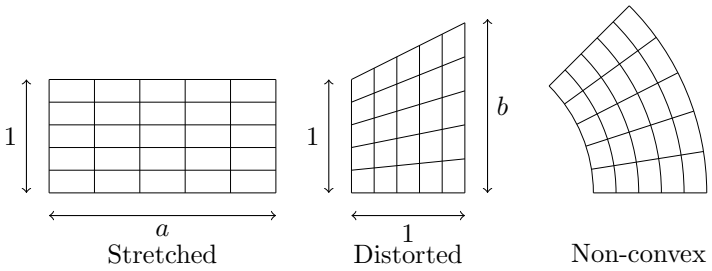

FIG. 5.2. *Robustness tests. Stretched elements (left): the domain is stretched in the x_1 direction and the deformation is described by the aspect ratio a. Distorted domain and elements (middle): the top right corner is stretched upwards in the x_2 direction with the ratio of the heights of the vertical boundaries parameterised by b. Curved non-convex domain (right).*

preconditioners for problems with stretched grids and domains that are non-square and non-convex. Stretched grids are needed, for example, for accurate computations of biharmonic eigenfunctions near the corners of the domain (see [8]). Figure 5.2 illustrates the three tests considered:

1. **Stretched Elements.** The domain is stretched in the x_1 direction and the deformation is described by the aspect ratio a.
2. **Distorted Elements.** The top right corner is stretched upwards in the x_2 direction; the ratio of the heights of the vertical boundaries is parameterised by b.
3. **Curved Domain.** The rectangular domain is isoparametrically deformed to form a non-convex curved domain.

In all tests we used the same number of elements in each coordinate direction.

We start by examining the effect of stretching the grid, as in the left of Figure 5.2, on the preconditioners. The analysis in Section 3 (for $\mathcal{P}_{BD}$ and $\mathcal{P}_{BBD}$) suggests that an increase in the element aspect ratio is likely to have a detrimental effect on the effectiveness of the preconditioners. This is confirmed by Figure 5.3 which shows the iteration counts and solution times for a stretch ratio of $a = 2.5$. Element stretching leads to a slight increase in the iteration counts and the solution times for $\mathcal{P}_{BD}$ and $\mathcal{P}_{BBD}$, though the asymptotic convergence rates obtained with these preconditioners

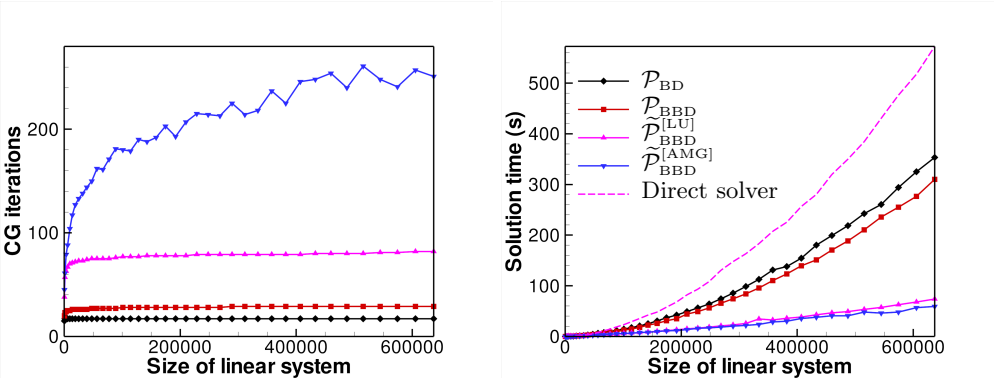

FIG. 5.3. *Number of CG iterations (left) and solution time in seconds (right) for robustness test 1 (stretched elements) with aspect ratio* $a = 2.5$. *The execution time of the direct method* SUPERLU *applied to the system with coefficient matrix* (1.4) *is presented for comparison. The legend applies to both plots.*

remain mesh independent. As expected, the two inexact implementations of the block-bordered preconditioner, $\widetilde{\mathcal{P}}^{[LU]}_{BBD}$ and $\widetilde{\mathcal{P}}^{[AMG]}_{BBD}$, are affected more strongly. We attribute this to the fact that for sufficiently large stretch ratios A_{22} and A_{33} have negative entries, which implies that the use of lumping and diagonal approximations in these preconditioners is less effective (cf. Remark 3). $\widetilde{\mathcal{P}}^{[AMG]}_{BBD}$ is most sensitive because this preconditioner is also affected by the behaviour of AMG on stretched meshes [11]. However, despite the noticeable increase in iteration counts, the plot of the solution times shows that the two inexact preconditioners remain significantly faster and scale better than the two exact preconditioners or the direct solver. In fact, over the range of problem sizes considered here, $\widetilde{\mathcal{P}}^{[AMG]}_{BBD}$ performs best.

Figure 5.4 illustrates the effect of element stretching on the CG convergence histories. For all four preconditioners, an increase in the element aspect ratio can be seen to lead to a decrease in the convergence rates (again consistent with the eigenvalue computations in Sections 3 and 4). In all cases the norm of the scaled residual starts with a value of one but jumps to a much larger value during the first CG iteration. Subsequently it decreases approximately linearly on a semi-log scale as the CG iteration proceeds. This implies that a reduction in the CG convergence tolerance will result in a controlled increase in the number of iterations required to achieve a solution of the desired accuracy.

Figure 5.5 shows the iteration counts and solution times for the deformed domain shown in the middle of Figure 5.2 ($b = 1.5$). In this case $\mathcal{P}_{BD}$, $\mathcal{P}_{BBD}$, and $\widetilde{\mathcal{P}}^{[LU]}_{BBD}$ yield mesh independent convergence rates, whereas the number of iterations obtained with $\widetilde{\mathcal{P}}^{[AMG]}_{BBD}$ appears to increase linearly with the problem size. While $\widetilde{\mathcal{P}}^{[AMG]}_{BBD}$ is still much faster than the exact preconditioners and the direct solver, $\widetilde{\mathcal{P}}^{[LU]}_{BBD}$ now yields the shortest execution times.

Finally, Figure 5.6 illustrates the performance of the preconditioners for the case of the curved, non-convex domain shown in Figure 5.2. Here the trends are similar to those observed for the case of stretched grids. In particular, we observe mesh-independent convergence for $\widetilde{\mathcal{P}}^{[LU]}_{BBD}$ but with higher iteration counts than for square elements. This again suggests issues with lumping/diagonal approximations for the matrix blocks A_{22}, A_{33} and A_{44}. The iteration counts obtained with $\widetilde{\mathcal{P}}^{[AMG]}_{BBD}$ show

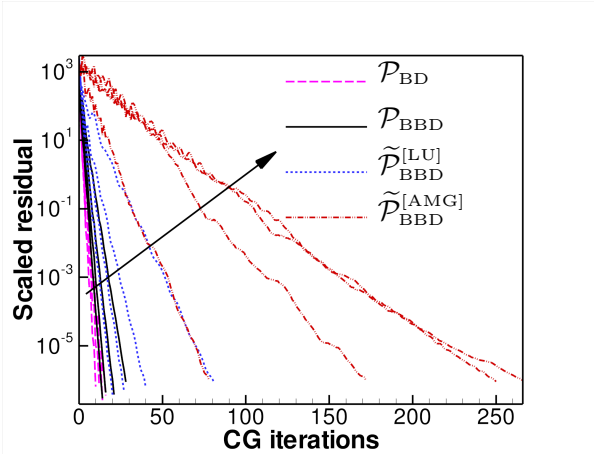

FIG. 5.4. *Convergence histories for the various preconditioners for stretch ratios* $a = 1.0, 1.5, 2.0$ *and* 2.5*, increasing in the direction of the arrow. In all cases, the domain was discretised with* 400×400 *elements.*

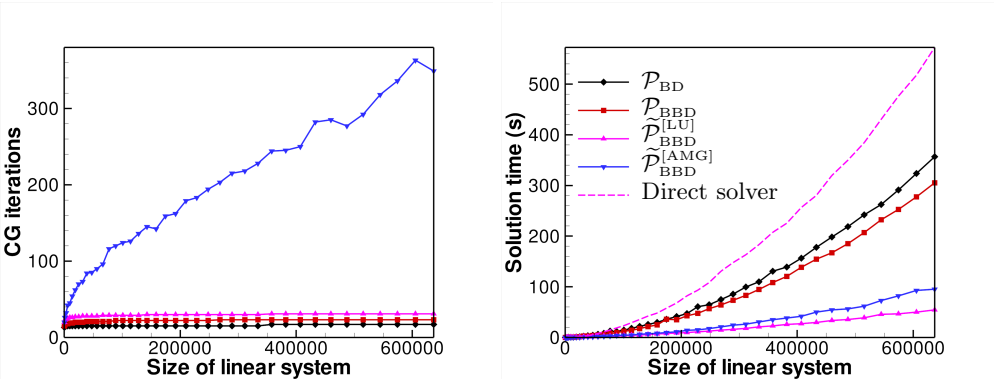

FIG. 5.5. *Number of CG iterations (left) and solution time in seconds (right) for robustness test 2 (distorted elements) with* $b = 1.5$*. The execution time of the direct method* SUPERLU *applied to the system with coefficient matrix* (1.4) *is presented for comparison. The legend applies to both plots.*

some signs of saturation, and this preconditioner leads to the shortest execution times overall, closely followed by $\widetilde{\mathcal{P}}^{[LU]}_{BBD}$.

6. Conclusions. We have presented effective preconditioners for the C^1 finite element discretisation of the Dirichlet biharmonic problem using Hermitian bicubic elements. The preconditioners are easy to set up as they only involve operations on blocks that are readily extracted from the full system; these blocks can also be computed from element matrices. On uniform meshes both the block diagonal and block bordered diagonal preconditioners appear to give mesh independent convergence. Moreover, we analysed the spectrum of block diagonal and block bordered diagonal preconditioners and showed that, under a certain condition, the block diag-

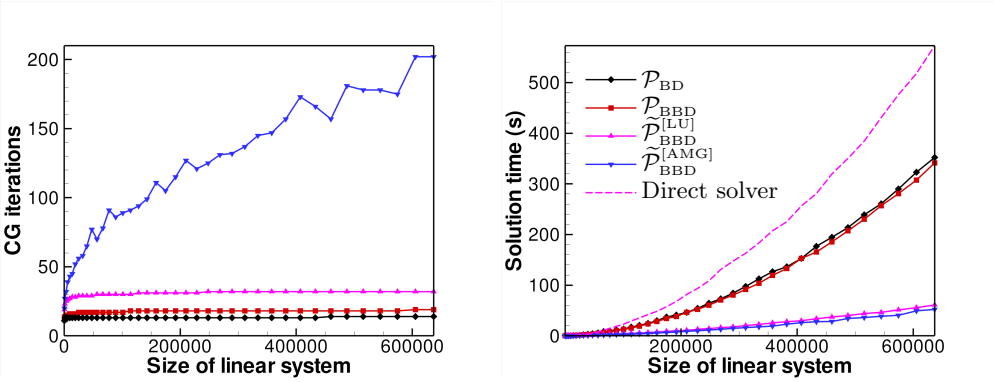

FIG. 5.6. *Number of CG iterations (left) and solution time in seconds (right) for robustness test 3 (curved non-convex domain). The execution time of the direct method* SUPERLU *applied to the system with coefficient matrix* (1.4) *is presented for comparison. The legend applies to both plots.*

onal preconditioner $\mathcal{P}_{BD}$ gives mesh independent convergence; the required condition holds for the uniform and stretched meshes tested here. Our analysis of the block diagonal preconditioner uses the approach of Wathen [34], [35], [36], which assumes that the coefficient matrix and preconditioner are symmetric positive definite and are assembled from element matrices. As such, Wathen's appealing technique is applicable to other finite element discretisations, differential operators and preconditioners.

To obtain a cost-optimal implementation, we further simplified the block bordered diagonal preconditioner $\mathcal{P}_{BBD}$ by lumping certain block matrices and using AMG for the approximate solution of a sparse Schur complement subsidiary linear system. We tested this approximate preconditioner on square, stretched and distorted elements and on non-convex domains. In all cases we observed mesh-independent convergence for $\mathcal{P}_{BD}$, $\mathcal{P}_{BBD}$ and $\widetilde{\mathcal{P}}_{BBD}^{[LU]}$. Although $\widetilde{\mathcal{P}}_{BBD}^{[AMG]}$ does not give mesh independent convergence, in many cases it gives the fastest execution time. However, stretching or distorting elements increased both the iteration counts and wall clock times, particularly for the AMG version of the preconditioner. An alternative to the current AMG solver could alleviate this issue. For example, in geometric multigrid, line smoothing is known to improve performance in the case of stretched or distorted meshes. It would be interesting to investigate this issue further.

Acknowledgements The authors would like to thank Andy Wathen for fruitful discussions, and the referees for helpful comments and references.

REFERENCES

[1] R. AITBAYEV, *A quadrature finite element Galerkin scheme for a biharmonic problem on a rectangular polygon*, Numer. Methods Partial Differential Equations, 24 (2008), pp. 518–534.

[2] B. AKSOYLU AND Z. YETER, *Robust multigrid preconditioners for the high-contrast biharmonic plate equation*, Numer. Lin. Algebra Appl., 18 (2011), pp. 733–750.

[3] P. BJØRSTAD, *Fast numerical solution of the biharmonic Dirichlet problem on rectangles*, SIAM J. Numer. Anal., 20 (1983), pp. 59–71.

[4] C. D. BOOR AND H.-O. KREISS, *On the condition of the linear systems associated with discretized BVPs of ODEs*, SIAM J. Numer. Anal., 23 (1986), pp. 936–939.

[5] J. BOYLE, M. MIHAJLOVIĆ, AND J. SCOTT, *HSL_MI20: An efficient AMG preconditioner for finite element problems in 3D*, Int. J. Numer. Methods Eng., 82 (2010), pp. 64–98.

[6] D. Braess, *Finite elements: Theory, fast solvers, and applications in solid mechanics*, Cambridge University Press, Cambridge, UK, 2007.
[7] J. Bramble and X. Zhang, *Multigrid methods for the biharmonic problem discretized by conforming C^1 finite elements on non-nested meshes*, Numer. Funct. Anal, and Optimiz., 16 (1995), pp. 835–846.
[8] B. M. Brown, E. B. Davis, P. K. Jimack, and M. D. Mihajlović, *A numerical investigation of the solution of a class of fourth-order eigenvalue problems*, Proc. Roy. Soc. (London), A 456 (2000), pp. 1505–1521.
[9] Q. Chang, Y. S. Wong, and H. Fu, *On the algebraic multigrid method*, J. Comput. Phys., 125 (1996), pp. 279–292.
[10] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, SIAM, Philadelphia, 2002.
[11] A. J. Cleary, R. D. Falgout, V. E. Henson, J. E. Jones, T. A. Manteuffel, S. F. McCormick, G. N. Miranda, and J. W. Ruge, *Robustness and Scalability of Algebraic Multigrid*, SIAM J. Sci. Comput., 21 (2000), pp. 1886–1908.
[12] J. W. Demmel, S. C. Eisenstat, J. R. Gilbert, X. S. Li, and J. W. H. Liu, *A supernodal approach to sparse partial pivoting*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 720–755.
[13] H. S. Dollar, N. I. M. Gould, W. H. A. Schilders, and A. J. Wathen, *Implicit-factorization preconditioning and iterative solvers for regularized saddle-point systems*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 170–189.
[14] H. Elman, D. Silvester, and A. Wathen, *Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluid Dynamics*, Oxford University Press, 2nd ed., 2014.
[15] X. Feng and T. Rahman, *A Non-Overlapping Additive Schwarz Method for the Biharmonic Equation*, Tech. Report 1554, University of Minnesota, 1998.
[16] A. Greenbaum, *Iterative methods for solving linear systems*, SIAM, Philadelphia, 1997.
[17] M. R. Hanisch, *Multigrid preconditioning for the Dirichlet biharmonic equation*, SIAM J. Numer. Anal., 30 (1993), pp. 184–214.
[18] M. Heil and A. L. Hazel, `oomph-lib` – *an object-oriented multi-physics finite-element library*, in Fluid-Structure Interaction, M. Schäfer and H.-J. Bungartz, eds., Springer, 2006, pp. 19–49. `oomph-lib` is available as open-source software at `http://www.oomph-lib.org`.
[19] S. Henn, *A multigrid method for a fourth-order diffusion equation with application to image processing*, SIAM J. Sci. Comput., 27 (2005), pp. 831–849.
[20] V. E. Henson and U. M. Yang, *BoomerAMG: A parallel algebraic multigrid solver and preconditioner*, Appl. Numer. Math., 41 (2002), pp. 155–177.
[21] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, 1990.
[22] HSL, *A collection of Fortran codes for large scale scientific computation.* http://www.hsl.rl.ac.uk.
[23] B. Khoromskij and G. Schmidt, *A fast interface solver for the biharmonic Dirichlet problem on polygonal domains*, Numer. Math., 78 (1998), pp. 577–596.
[24] Z. C. Li, H.-T. Huang, and J. Huang, *Effective condition number of the Hermite finite element methods for biharmonic equations*, Appl. Numer. Math., 58 (2008), pp. 1291–1308.
[25] S. F. McCormick, *Multigrid Methods*, SIAM, Philadelphia, 1994.
[26] P. Oswald, *Multilevel preconditioners for discretizations of the biharmonic equation by rectangular finite elements*, Numer. Lin. Algebra Appl., 2 (1995), pp. 487–505.
[27] R. L. Panton, *Incompressible Flow*, John Wiley & Sons, New York, second ed., 1996.
[28] B. N. Parlett, *The Symmetric Eigenvalue Problem*, SIAM, Philadelphia, 1998.
[29] P. Peisker and D. Braess, *A conjugate gradient method and a multigrid algorithm for Morley's finite element approximation of the biharmonic equation*, Numer. Math., 50 (1987), pp. 567–586.
[30] J. R. Rice, *Matrix Computations and Mathematical Software*, McGraw-Hill, New York, 1981.
[31] D. J. Silvester and M. D. Mihajlović, *A black-box multigrid preconditioner for the biharmonic equation*, BIT, 44 (2004), pp. 151–163.
[32] S. Timoshenko, *Theory of Plates and Shells*, The McGraw-Hill Book Company, New York, 1st ed., 1959.
[33] P. Vanek, J. Mandel, and M. Brezina, *Algebraic multigrid by smoother aggregation for second and fourth order elliptic problems*, Computing, 56 (1996), pp. 179–196.
[34] A. J. Wathen, *Realistic eigenvalue bounds for the Galerkin mass matrix*, IMA J. Numer. Anal., 7 (1987), pp. 449–457.
[35] ———, *Spectral bounds and preconditioning methods using element-by-element analysis for Galerkin finite element equations*, in The Mathematics of Finite Elements and Applications VI, MAFELAP 1987, J. R. Whiteman, ed., Academic Press, London, UK, 1988, pp. 157–168.

[36] ———, *Singular element preconditioning for the finite element method*, in Iterative Methods in Linear Algebra, R. Beauwens and P. de Groen, eds., North Holland, Amsterdam, 1992, pp. 531–540.

[37] G. Wempner and T. Demosthenes, *Mechanics of Solids and Shells: Theories and Approximations*, CRC Press, Florida, 2003.

[38] S. Zhang, *An optimal order multigrid method for biharmonic, C^1 finite element equations*, Numer. Math., 56 (1989), pp. 613–624.

[39] S. Zhang and J. Xu, *Optimal solvers for fourth-order PDEs discretized on unstructured grids*, SIAM J. Numer. Anal., 52 (2014), pp. 282–307.

[40] X. Zhang, *Multilevel Schwarz methods for the biharmonic Dirichlet problem*, SIAM J. Sci. Comput., 15 (1994), pp. 621–644.

[41] ———, *Two-level Schwarz methods for the biharmonic problem discretized conforming C^1 elements*, SIAM J. Numer. Anal., 33 (1996), pp. 555–570.