Electronic Transactions on Numerical Analysis.
Volume 40, pp. 294-310, 2013.
Copyright © 2013, Kent State University.
ISSN 1068-9613.

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

FAST ITERATIVE SOLVERS FOR CONVECTION-DIFFUSION CONTROL PROBLEMS*

JOHN W. PEARSON† AND ANDREW J. WATHEN†

Abstract. In this manuscript, we describe effective solvers for the optimal control of stabilized convection-diffusion control problems. We employ the Local Projection Stabilization, which results in the same matrix system whether the discretize-then-optimize or optimize-then-discretize approach for this problem is used. We then derive two effective preconditioners for this problem, the first to be used with MINRES and the second to be used with the Bramble-Pasciak Conjugate Gradient method. The key components of both preconditioners are an accurate mass matrix approximation, a good approximation of the Schur complement, and an appropriate multigrid process to enact this latter approximation. We present numerical results to illustrate that these preconditioners result in convergence in a small number of iterations, which is robust with respect to the step-size h and the regularization parameter β for a range of problems.

Key words. PDE-constrained optimization, convection-diffusion control, preconditioning, Local Projection Stabilization, Schur complement.

AMS subject classifications. 49M25, 65F08, 65F10, 65N30.

1. Introduction.

Convection-diffusion problems describe important physical processes such as contaminant transport. The numerical solution of such problems, in particular in the case of dominating convection, has attracted much attention, and it is now widely appreciated what role stabilization techniques have to play. In this manuscript we consider not the solution of single convection-diffusion problems (we will call this the solution of the forward problem) but the control of such problems. That is to say, we consider solution methods for control problems involving the convection-diffusion equation together with suitable boundary conditions. In particular we will describe two preconditioned iterative solution methods for the fast solution of such control problems.

Control problems, or PDE-constrained optimization problems, for various partial differential equations have been the subject of much research (see, for example, the excellent book by Tröltzsch [24]), and there has been significant recent interest in preconditioning and iterative solvers for such problems; see, for example, [19, 21]. In all such problems there arises the issue of whether to firstly perform discretization before optimization of the resulting discrete problem or to construct continuous optimality conditions and then discretize. For many PDE problems, in particular those which are self-adjoint, the two possible approaches of discretize-then-optimize and optimize-then-discretize generally give rise to the same discrete equations—that is to say the two steps commute.

For the convection-diffusion control problem, Heinkenschloss and co-workers [6, 10] have considered the quite popular SUPG stabilized finite element method of Hughes and Brooks [11] and have shown the significant extra difficulty in the case of the control problem as opposed to the forward problem. A key issue is consistency not just of the forward problem but also of the adjoint problem. The SUPG method does not satisfy such adjoint-consistency in general, though for the forward problem it yields an order of accuracy of $\mathcal{O}(h^{3/2})$ when using bilinear finite elements for instance; see [8, Theorem 3.6]. For the control problem this leads to the issue that the discretize-then-optimize approach gives rise to symmetric discrete equations in which the discrete adjoint problem is not a consistent discretization of the

*Received November 2, 2011. Accepted April 12, 2013. Published online on August 5, 2013. Recommended by V. Simoncini.
†Numerical Analysis Group, Mathematical Institute, University of Oxford, 24–29 St Giles', Oxford, OX1 3LB, UK (john.pearson@worc.ox.ac.uk, wathen@maths.ox.ac.uk).

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

continuous adjoint problem, and the optimize-then-discretize approach gives rise to different and non-symmetric discrete equations which do not therefore have the structure of a discrete optimization problem.

Here we employ the adjoint-consistent Local Projection Stabilization approach described in [1, 2, 4], which ensures that the discretize and optimize steps commute. For this approach we are able to establish preconditioned iterative solvers for the control problem which have the attractive feature of giving convergence in a number of steps independent of the parameters of the problem (including the mesh-size). With an appropriate multigrid process for the convection-diffusion problem which we describe, this leads to solvers of optimal computational complexity for PDE-constrained optimization problems involving the convection-diffusion problem.

2. Background. In this section, we summarize the theory that we will use when solving the convection-diffusion control problem. Firstly, we will detail a method for solving the forward problem, that is the convection-diffusion equation with no optimization. We will exploit aspects of this method when we wish to solve the control problem. Secondly, we will detail some properties of ideal preconditioners for saddle point systems. The convection-diffusion control problem has a saddle point structure, as we will show in Section 3, so we will need to use the theory of saddle point systems in order to develop preconditioners for this problem as in Section 4.

2.1. Solution of the convection-diffusion equation. We first consider the finite element solution of the convection-diffusion equation with Dirichlet boundary conditions

$$\begin{aligned} -\epsilon\nabla^2 y + \mathbf{w}\cdot\nabla y &= g \quad \text{in } \Omega, \\ y &= f \quad \text{on } \partial\Omega, \end{aligned} \tag{2.1}$$

where the domain $\Omega \subset \mathbb{R}^d$, $d = 2$ or 3, has boundary $\partial\Omega$, $\epsilon > 0$ represents viscosity, and $\mathbf{w}$ is a divergence-free wind vector (i.e. $\nabla\cdot\mathbf{w} = 0$).

The term $-\epsilon\nabla^2 y$ in the above equation denotes the diffusive element, and the term $\mathbf{w}\cdot\nabla y$ represents convection. As pointed out, for example in [8, Chapter 3], convection typically plays a more significant physical role than diffusion, so $\epsilon \ll \|\mathbf{w}\|$ for many practical problems. However this in turn makes the problem more difficult to solve [8, 17] as the solution procedure will need to be robust with respect to the direction of the wind $\mathbf{w}$ and any boundary or internal layers that form.

The finite element representation of the equation (2.1) is given by

$$\bar{K}\mathbf{y} = \mathbf{f}, \tag{2.2}$$

where $\mathbf{y} = \{Y_i\}_{i=1,\ldots,n}$, with Y_i denoting the coefficients of the finite element solution $y_h = \sum_{i=1}^{n+n_\partial} Y_i\phi_i$ with interior finite element basis functions $\phi_1, \ldots, \phi_n$ and boundary basis functions $\phi_{n+1}, \ldots, \phi_{n+n_\partial}$. The matrix $\bar{K}$, as stated in (2.2), is defined by

$$\begin{aligned} &\bar{K} = \epsilon K + N + T, \\ &K = \{k_{ij}\}_{i,j=1,\ldots,n}, \qquad k_{ij} = \int_\Omega \nabla\phi_i\cdot\nabla\phi_j \,\mathrm{d}\Omega, \\ &N = \{\widetilde{n}_{ij}\}_{i,j=1,\ldots,n}, \qquad \widetilde{n}_{ij} = \int_\Omega (\mathbf{w}\cdot\nabla\phi_j)\phi_i \,\mathrm{d}\Omega. \end{aligned}$$

Here, T is a matrix corresponding to the stabilization strategy used (which depends on the step-size h, a stabilization parameter δ, and an orthogonal projection operator π_h). The vector $\mathbf{f}$ corresponds to the functions f and g (and sometimes the stabilization as well). Note

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

that K is a *stiffness matrix*, a commonly occurring matrix in finite element problems. We discuss the definitions of T and $\mathbf{f}$ for two different stabilization methods in Section 3.1 (and note that $T = 0$ if no stabilization is used).

For the remainder of this section we briefly detail a method described in [8] for solving the problem (2.2) as we will use aspects of this method in our solvers for the convection-diffusion control problem in Section 4.

The method discussed in [8] for solving (2.1) is a GMRES method preconditioned with a geometric multigrid process described by Ramage in [17]. The multigrid process contains standard prolongation and restriction operators, but there are two major differences between it and a more typical multigrid routine:

- *Construction of the coarse grid operator.* In most geometric multigrid algorithms, the construction of a coarse grid operator is carried out using the scaled Galerkin coarse grid operator (that is $\bar{K}_{\text{coarse}} = R\bar{K}_{\text{fine}}P$, where P is the projection operator and R the restriction operator). However, in the method of Ramage, the coarse grid operator is *explicitly constructed* on all grids on which it is required. This involves constructing the matrices K, N, and T on each sub-grid and incorporates different stabilization parameters δ for each grid.
- *Pre- and post-smoothing.* The smoothing strategy we employ is *block Gauss-Seidel smoothing*, applied in each direction to take account of all possible wind directions, that is to say we employ 4 (2 pre- and 2 post-) smoothing steps for a two dimensional problem and 6 smoothing steps for a three dimensional problem. This strategy is shown to be effective for a wide range of problems with our formulation as illustrated in [8, Chapter 4] and [17].

2.2. Saddle point systems. The convection-diffusion control problem that we introduce in Section 3 is of *saddle point* structure, that is, it is of the form

$$\underbrace{\begin{bmatrix} A & B^T \\ B & -C \end{bmatrix}}_{\mathcal{A}} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}, \tag{2.3}$$

where $A \in \mathbb{R}^{m\times m}$, $B \in \mathbb{R}^{q\times m}$, and $C \in \mathbb{R}^{q\times q}$, with $m \geq q$. For an overview of properties and solution methods for such systems, we refer the reader to [3].

In [14], it is demonstrated that two effective preconditioners for $\mathcal{A}$ are given by

$$\mathcal{P}_1 = \begin{bmatrix} A & 0 \\ 0 & S \end{bmatrix}, \quad \mathcal{P}_2 = \begin{bmatrix} A & 0 \\ B & -S \end{bmatrix},$$

where S is the (negative) *Schur complement* defined by $S = C + BA^{-1}B^T$. The reason these preconditioners are so potent is that the spectra of $\mathcal{P}_1^{-1}\mathcal{A}$ and $\mathcal{P}_2^{-1}\mathcal{A}$ are given by

$$\lambda(\mathcal{P}_1^{-1}\mathcal{A}) = \left\{\frac{1}{2}(1-\sqrt{5}), 1, \frac{1}{2}(1+\sqrt{5})\right\}, \quad \lambda(\mathcal{P}_2^{-1}\mathcal{A}) = \{1\},$$

in the case where $C = 0$, so long as $\mathcal{P}_1^{-1}\mathcal{A}$ and $\mathcal{P}_2^{-1}\mathcal{A}$ are nonsingular [13, 14]. In the general case $C \neq 0$, the result on $\lambda(\mathcal{P}_2^{-1}\mathcal{A})$ also holds [12]. We note that $C = 0$ in the set-up of the convection-diffusion control problem that we focus on in this article.

Now $\mathcal{P}_1^{-1}\mathcal{A}$ constructed in this way is diagonalizable but $\mathcal{P}_2^{-1}\mathcal{A}$ is not, so if we apply a Krylov subspace method with $\mathcal{A}$ preconditioned by $\mathcal{P}_1$ or $\mathcal{P}_2$, we will achieve termination in 3 and 2 iterations, respectively [14]. Of course the preconditioners $\mathcal{P}_1$ and $\mathcal{P}_2$ are not practical preconditioners as the exact inverses of A and S will need to be enforced in each case (which is particularly problematic as even when A and B are sparse, S is generally dense).

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

However, if we were able to construct effective approximations to A and S, $\widehat{A}$ and $\widehat{S}$ say, and employ the preconditioners

$$\widehat{\mathcal{P}}_1 = \begin{bmatrix} \widehat{A} & 0 \\ 0 & \widehat{S} \end{bmatrix}, \quad \widehat{\mathcal{P}}_2 = \begin{bmatrix} \widehat{A} & 0 \\ B & -\widehat{S} \end{bmatrix},$$

it is likely that we would obtain convergence of the appropriate Krylov subspace method in few iterations. In Section 4, we derive two preconditioners based on $\widehat{\mathcal{P}}_1$ and $\widehat{\mathcal{P}}_2$ for the convection-diffusion control problem.

Clearly, these preconditioners will have to be incorporated into different Krylov subspace methods. The block diagonal preconditioner $\widehat{\mathcal{P}}_1$ is symmetric positive definite, and so a natural choice is the MINRES algorithm [15, 19]. By contrast, the block triangular preconditioner $\widehat{\mathcal{P}}_2$ is neither symmetric nor positive definite, and so the same algorithm cannot be used. However as described in [5, 20, 23] for example, $\widehat{\mathcal{P}}_2^{-1}\mathcal{A}$ is symmetric positive definite in the inner product $\langle\cdot,\cdot\rangle_{\mathcal{H}}$ defined by $\langle \mathbf{u}, \mathbf{v}\rangle_{\mathcal{H}} = \mathbf{u}^T\mathcal{H}\mathbf{v}$, where

$$\mathcal{H} = \begin{bmatrix} A - \widehat{A} & 0 \\ 0 & \widehat{S} \end{bmatrix},$$

with $\widehat{A}$, $\widehat{S}$ chosen to ensure that $\mathcal{H}$ is positive definite. Hence it is possible to use a non-standard Conjugate Gradient method with the $\mathcal{H}$-inner product; this is often referred to as the *Bramble-Pasciak Conjugate Gradient* method.

3. The convection-diffusion control problem. For the remainder of this paper, we will be considering the distributed convection-diffusion control problem

$$\begin{aligned} \min_{y,u} \quad & \frac{1}{2}\left\|y-\widehat{y}\right\|^2_{L_2(\Omega)} + \frac{\beta}{2}\left\|u\right\|^2_{L_2(\Omega)} \\ \text{s.t.} \quad & -\epsilon\nabla^2 y + \mathbf{w}\cdot\nabla y = u \ \text{ in } \Omega, \\ & \qquad\qquad\qquad\quad y = f \ \text{ on } \partial\Omega, \end{aligned} \tag{3.1}$$

where y denotes the *state variable* with $\widehat{y}$ some desired state, u denotes the *control variable*, and $\beta > 0$ is a regularization parameter (sometimes known as the *Tikhonov parameter*).

We employ a finite element method to solve the problem, that is we write

$$y_h = \sum_{i=1}^{n+n_\partial} Y_i\phi_i, \quad u_h = \sum_{i=1}^{n+n_\partial} U_i\phi_i, \quad p_h = \sum_{i=1}^{n+n_\partial} P_i\phi_i,$$

where p denotes the Lagrange multiplier we use. Note that we discretize the state y, the control u, and the Lagrange multiplier p using the same basis functions here. Note also that the coefficients $Y_{n+1},...,Y_{n+n_\partial}$ are trivially obtained by considering the specified Dirichlet boundary condition $y = f$.

For the rest of this section, we define $\mathbf{y}$, $\mathbf{u}$, and $\mathbf{p}$ as follows:

$$\mathbf{y} = \{Y_i\}_{i=1,\ldots,n}, \quad \mathbf{u} = \{U_i\}_{i=1,\ldots,n}, \quad \mathbf{p} = \{P_i\}_{i=1,\ldots,n}.$$

3.1. Stabilization of the control problem. One important consideration when solving the convection-diffusion control problem (or indeed the convection-diffusion equation itself) is that of stabilizing the problem. It is well known that, without any form of stabilization, accurate solution of the convection-diffusion equation [8, 17] and the convection-diffusion

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

control problem [2, 10] is compromised due to the formation of layers in the approximate solution, potentially leading to large errors for small ϵ.

One popular method for avoiding this problem is by using the *Streamline Upwind Petrov-Galerkin* (SUPG) stabilization, which was introduced in [11] and discussed further in literature such as [8, 10, 18]. For the forward problem, using this stabilization would result in a system of the form (2.2), with K and N as above, and

$$\begin{aligned} T &= \{\tau^{\delta}_{h,ij}\}_{i,j=1,\ldots,n}, & \tau^{\delta}_{h,ij} &= \delta \int_{\Omega} (\mathbf{w}\cdot\nabla\phi_i)(\mathbf{w}\cdot\nabla\phi_j)\ \mathrm{d}\Omega \\ &&&\quad - \epsilon\delta \sum_{k} \int_{\Delta_k} (\nabla^2\phi_i)(\mathbf{w}\cdot\nabla\phi_j)\ \mathrm{d}\Omega, \\ \mathbf{f} &= \{f_i\}_{i=1,\ldots,n}, & f_i &= \int_{\Omega} g\phi_i\ \mathrm{d}\Omega + \delta\int_{\Omega} g\mathbf{w}\cdot\nabla\phi_i\ \mathrm{d}\Omega, \end{aligned}$$

with a stabilization parameter δ, and Δ_k denoting the k-th element in our finite element discretization. Here we have taken zero Dirichlet conditions for illustrative purposes. It is well recognised that this method is effective for solving the forward problem; see, for instance, [8, Chapters 3 and 4]. However, for the convection-diffusion control problem, difficulties arise—the matrix systems that we obtain when we use the *discretize-then-optimize* and *optimize-then-discretize* formulations of Sections 3.2 and 3.3 do not commute [18, Chapter 6]. This is problematic as we would then have to choose between solving the discretize-then-optimize matrix system, which would not be strongly consistent (meaning the solutions to the optimization problem would not satisfy all the optimality conditions), or the optimize-then-discretize system, which is non-symmetric and so is not the optimality system for any finite dimensional problem. Further, the non-symmetry of the matrix system that arises when using the optimize-then-discretize approach means that we cannot apply the iterative methods introduced in Section 2.2 to solve it as these methods depend on the matrix being symmetric. It is also believed that applying SUPG to the optimal control problem will guarantee at most first-order accuracy in the solution [10].

To deal with these two problems, we now introduce the *Local Projection Stabilization* (LPS) method, which is discussed in [2, 9] for example. Applying this stabilization to the forward problem again yields a matrix system of the form (2.2), with K and N as above and

$$\begin{aligned} T &= \{\tau^{\delta}_{h,ij}\}_{i,j=1,\ldots,n}, & \tau^{\delta}_{h,ij} &= \delta\int_{\Omega} \big(\mathbf{w}\cdot\nabla\phi_i - \pi_h(\mathbf{w}\cdot\nabla\phi_i)\big) \\ &&&\qquad \times \big(\mathbf{w}\cdot\nabla\phi_j - \pi_h(\mathbf{w}\cdot\nabla\phi_j)\big)\ \mathrm{d}\Omega, \\ \mathbf{f} &= \{f_i\}_{i=1,\ldots,n}, & f_i &= \int_{\Omega} g\phi_i\ \mathrm{d}\Omega, \end{aligned} \tag{3.2}$$

where δ is again a stabilization parameter and π_h an orthogonal projection operator. We have again taken zero Dirichlet conditions for this definition. Furthermore, as we will demonstrate in Sections 3.2 and 3.3, when this stabilization is applied in the optimal control setting, the discretize-then-optimize and optimize-then-discretize systems are consistent and self-adjoint, that is the discretization and optimization steps commute.

There are a number of considerations which need to be taken into account when applying this method in the control setting with a uniform grid and bilinear basis functions, as we will do in Section 5.

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

- *Stabilization parameter* δ. We take δ to be the following as in [2]:

$$\delta = \begin{cases} 0 & \text{if Pe} < 1, \\ \frac{h}{\|\mathbf{w}\|_2} & \text{if Pe} \geq 1, \end{cases}$$

  where the mesh Péclet number Pe is defined on each element as

$$\text{Pe} = \frac{h\,\|\mathbf{w}\|_2}{\epsilon}.$$

  Clearly this means that the stabilization depends on the mesh-size, and if the step-size h is less than $\frac{\epsilon}{\|\mathbf{w}\|_2}$, then no stabilization procedure will be applied.
- *Orthogonal projection operator* π_h. We require an L_2-orthogonal projection operator defined on patches of the domain that satisfies the L_2-norm properties specified in [2, p. 4]. We will proceed by working with **Q1** elements with equally spaced nodes and divide the domain into patches consisting of 2 elements in each dimension. From this, we will take $\pi_h(v)$ (where v has support solely on that patch) to be equal to the integral of v over the patch divided by the area of the patch (in 2D this will be $4h^2$). This definition will satisfy the required properties in our formulation.
- *Error of LPS method.* In [2], it is shown that the LPS stabilization gives a rate of convergence of $\mathcal{O}(h^{3/2})$ for problems of the form (3.1) for bilinear finite elements. This further motivates the use of the LPS stabilization method for the remainder of this manuscript.

3.2. Matrix system obtained: discretize-then-optimize. We now demonstrate that, when using the LPS method described in Section 3.1, the matrix systems obtained with the discretize-then-optimize and optimize-then-discretize approaches are the same. The derivation of the matrix system when using the former approach is straightforward. We first note that the discretized version of the PDE constraint is given by

$$\bar{K}\mathbf{y} - M\mathbf{u} = \mathbf{d},$$

where $\mathbf{d}$ is stated below.

We also note that we may write the functional that we are trying to minimize, that is $\frac{1}{2}\|y-\widehat{y}\|^2_{L_2(\Omega)} + \frac{\beta}{2}\|u\|^2_{L_2(\Omega)}$, as

$$\frac{1}{2}\|y-\widehat{y}\|^2_{L_2(\Omega)} + \frac{\beta}{2}\|u\|^2_{L_2(\Omega)} = \frac{1}{2}\mathbf{y}^T M\mathbf{y} - \mathbf{b}^T\mathbf{y} + C + \frac{\beta}{2}\mathbf{u}^T M\mathbf{u},$$

where C is a constant independent of $\mathbf{y}$, M denotes the *mass matrix* defined by

$$M = \{m_{ij}\}_{i,j=1,\ldots,n}, \quad m_{ij} = \int_\Omega \phi_i\phi_j \,\mathrm{d}\Omega,$$

and $\mathbf{b}$ is given by

$$\mathbf{b} = \{b_i\}_{i=1,\ldots,n}, \quad b_i = \int_\Omega \widehat{y}\phi_i \,\mathrm{d}\Omega.$$

We therefore deduce that the Lagrangian, the stationary point of which we wish to find, is given by

$$\mathcal{L}(\mathbf{y},\mathbf{u},\mathbf{p}) = \frac{1}{2}\mathbf{y}^T M\mathbf{y} - \mathbf{b}^T\mathbf{y} + C + \frac{\beta}{2}\mathbf{u}^T M\mathbf{u} + \mathbf{p}^T(\bar{K}\mathbf{y} - M\mathbf{u} - \mathbf{d}). \tag{3.3}$$

Differentiating (3.3) with respect to $\mathbf{y}$, $\mathbf{u}$, and $\mathbf{p}$ yields the following system of equations

$$\begin{bmatrix} M & 0 & \bar{K}^T \\ 0 & \beta M & -M \\ \bar{K} & -M & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{u} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \\ \mathbf{d} \end{bmatrix}, \tag{3.4}$$

where

$$\mathbf{d} = \{d_i\}_{i=1,\ldots,n}, \quad d_i = -\sum_{j=n+1}^{n+n_\partial} Y_j \int_\Omega \nabla\phi_i \cdot \nabla\phi_j \, \mathrm{d}\Omega.$$

This system is of the saddle point form discussed in Section 2.2. We note that in the above set-up, we have reduced the matrix system to a $3n \times 3n$ system by eliminating the equations corresponding to boundary conditions. However, it is perfectly possible to solve instead a $3(n+n_\partial) \times 3(n+n_\partial)$ system by not eliminating these equations, and this is the approach we will follow in our numerical tests of Section 5.

3.3. Matrix system obtained: optimize-then-discretize. To derive the optimize-then-discretize formulation, as in [2], we need to consider a Lagrangian of the form

$$\begin{aligned} \widetilde{\mathcal{L}}(y,u,p,\widetilde{p}) = {} & \frac{1}{2}\left\|y-\widehat{y}\right\|_{L_2(\Omega)}^2 + \frac{\beta}{2}\left\|u\right\|_{L_2(\Omega)}^2 \\ & + \int_\Omega (-\epsilon\nabla^2 y + \mathbf{w}\cdot\nabla y - u)p \, \mathrm{d}\Omega + \int_{\partial\Omega} (y-f)\widetilde{p} \, \mathrm{d}s, \end{aligned}$$

where y and u relate to the weak solutions of the forward problem, and p, $\widetilde{p}$ are assumed to be sufficiently smooth. Note that the second Lagrange multiplier $\widetilde{p}$ is included in this case as we are not guaranteed to satisfy the boundary conditions as with the discretize-then-optimize approach.

As in [18] for example, we differentiate $\widetilde{\mathcal{L}}$ with respect to the state y, the control u, and the Lagrange multipliers p and $\widetilde{p}$ and study the resulting equations. Calculating the Fréchet derivative with respect to y and applying the divergence theorem and the fundamental lemma of calculus of variations along with the assumption $\nabla\cdot\mathbf{w} = 0$, as in [18], yields the *adjoint equation*. Differentiating with respect to u generates the *gradient equation* and differentiating with respect to the Lagrange multipliers p and $\widetilde{p}$ yields the *state equation*. Discretizing these three equations using the stabilization (3.2) yields the matrix system

$$\begin{bmatrix} M & 0 & \bar{K}^T \\ 0 & \beta M & -M \\ \bar{K} & -M & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{u} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \\ \mathbf{d} \end{bmatrix},$$

which is the same saddle point system as that derived using the discretize-then-optimize approach. We therefore consider the solution of this system for the remainder of this manuscript.

4. Preconditioning the matrix system. In this section, we consider how one might precondition the matrix system (3.4) for solving the convection-diffusion control problem with Local Projection Stabilization. We will use the saddle point theory of Section 2.2 in this section.

We first note that we may write (3.4) as a sparse saddle point system of the form (2.3), with $A = \begin{bmatrix} M & 0 \\ 0 & \beta M \end{bmatrix}$, $B = \begin{bmatrix} \bar{K} & -M \end{bmatrix}$, and $C = \begin{bmatrix} 0 \end{bmatrix}$. By the theory of Section 2.2,

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

we see that we may obtain an effective solver if we have a good approximation of the matrix $\begin{bmatrix} M & 0 \\ 0 & \beta M \end{bmatrix}$, as well as the Schur complement of the matrix system which is given by

$$S = \bar{K}M^{-1}\bar{K}^T + \frac{1}{\beta}M.$$

We therefore start by considering an accurate approximation of these two matrices. As discussed in [25], the Chebyshev semi-iterative method is effective for approximating mass matrices, so in our preconditioners we may approximate A by $\widehat{A}$, where

$$\widehat{A} = \begin{bmatrix} \widehat{M} & 0 \\ 0 & \beta\widehat{M} \end{bmatrix},$$

and $\widehat{M}$ denotes 20 steps of Chebyshev semi-iteration applied to M.

To find an accurate approximation of the Schur complement, we apply the result of Theorem 4.1 below. This theorem gives us a Schur complement approximation for which the eigenvalues of the Schur complement preconditioned with this approximation are bounded robustly given positive semi-definiteness of the symmetric matrix $\epsilon K + T$ and skew-symmetry of the matrix N (see [8, Chapters 3 and 5] for more details) and therefore positive semi-definiteness of the symmetric part of $\bar{K}$, $H := \frac{1}{2}(\bar{K} + \bar{K}^T)$. We note that Theorem 4.1 is an extension of the result proved in [16], which applies to symmetric operators rather than the non-symmetric operator $\bar{K}$ we are considering in this manuscript.

THEOREM 4.1. *Suppose that the symmetric part of $\bar{K}$, $H := \frac{1}{2}(\bar{K} + \bar{K}^T)$, is positive semi-definite. Then, if we approximate the Schur complement S by*

$$\widehat{S} = \left(\bar{K} + \frac{1}{\sqrt{\beta}}M\right) M^{-1} \left(\bar{K} + \frac{1}{\sqrt{\beta}}M\right)^T,$$

we can bound the eigenvalues of $\widehat{S}^{-1}S$ as follows:

$$\lambda(\widehat{S}^{-1}S) \in \left[\frac{1}{2}, 1\right].$$

Proof. We have that the eigenvalues μ and eigenvectors $\mathbf{x}$ of $\widehat{S}^{-1}S$ satisfy:

$$\begin{aligned} & \widehat{S}^{-1}S\mathbf{x} = \mu\mathbf{x} \\ \Leftrightarrow \quad & \left(\beta\bar{K}M^{-1}\bar{K}^T + M\right)\mathbf{x} = \mu\left[\beta\bar{K}M^{-1}\bar{K}^T + M + \sqrt{\beta}(\bar{K} + \bar{K}^T)\right]\mathbf{x}. \end{aligned}$$

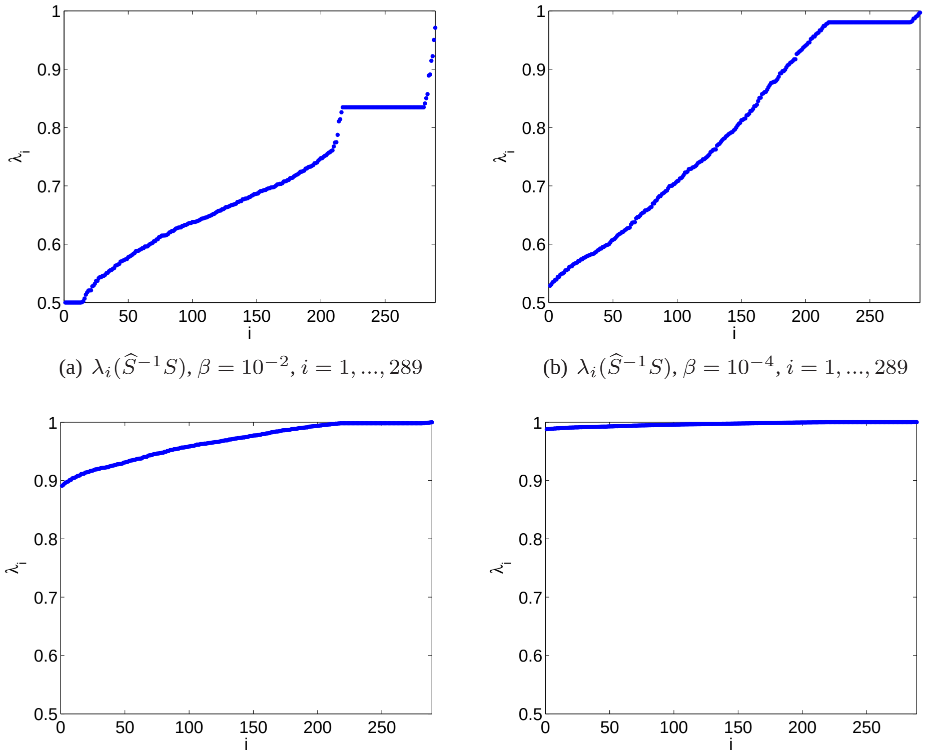
It is sufficient to show that the Rayleigh quotient $R := \frac{\mathbf{v}^T S\mathbf{v}}{\mathbf{v}^T \widehat{S}\mathbf{v}} \in \left[\frac{1}{2}, 1\right]$. To show this, we write

$$R = \frac{\mathbf{v}^T\left[\beta\bar{K}M^{-1}\bar{K}^T + M\right]\mathbf{v}}{\mathbf{v}^T\left[\beta\bar{K}M^{-1}\bar{K}^T + M + \sqrt{\beta}(\bar{K} + \bar{K}^T)\right]\mathbf{v}} = \frac{\mathbf{a}^T\mathbf{a} + \mathbf{b}^T\mathbf{b}}{(\mathbf{a} + \mathbf{b})^T(\mathbf{a} + \mathbf{b})},$$

where $\mathbf{a} = (\sqrt{\beta}\bar{K}M^{-1/2})^T\mathbf{v}$, $\mathbf{b} = (M^{1/2})^T\mathbf{v}$, and with $\mathbf{v} \neq \mathbf{0}$.

The upper bound follows from the fact that $\sqrt{\beta}\mathbf{v}^T(\bar{K} + \bar{K}^T)\mathbf{v} = 2\sqrt{\beta}\mathbf{v}^T H\mathbf{v} \geq 0$ by the assumption of positive semi-definiteness of H, as well as the positivity of $\mathbf{b}^T\mathbf{b} = \mathbf{v}^T M\mathbf{v}$ (which ensures that both the numerator and denominator of R are strictly positive).

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



(a) $\lambda_i(\widehat{S}^{-1}S)$, $\beta = 10^{-2}$, $i = 1, ..., 289$

(b) $\lambda_i(\widehat{S}^{-1}S)$, $\beta = 10^{-4}$, $i = 1, ..., 289$

(c) $\lambda_i(\widehat{S}^{-1}S)$, $\beta = 10^{-6}$, $i = 1, ..., 289$

(d) $\lambda_i(\widehat{S}^{-1}S)$, $\beta = 10^{-8}$, $i = 1, ..., 289$

FIG. 4.1. *Spectra of* $\widehat{S}^{-1}S$ *for* $\beta = 10^{-2}$, $\beta = 10^{-4}$, $\beta = 10^{-6}$, *and* $\beta = 10^{-8}$ *for an evenly spaced grid on* $\Omega = [-1, 1]^2$ *with* $h = 2^{-3}$, $\epsilon = \frac{1}{100}$, *and* $\mathbf{w} = \left(\sin\frac{\pi}{6}, \cos\frac{\pi}{6}\right)^T$.

To show that $R \geq \frac{1}{2}$, we proceed as follows noting again that $\mathbf{b}^T\mathbf{b} > 0$:

$$
\begin{aligned}
R \geq \frac{1}{2} &\Leftrightarrow \mathbf{a}^T\mathbf{a} + \mathbf{b}^T\mathbf{b} \geq \frac{1}{2}\left[\mathbf{a}^T\mathbf{a} + \mathbf{b}^T\mathbf{b} + \mathbf{a}^T\mathbf{b} + \mathbf{b}^T\mathbf{a}\right] \\
&\Leftrightarrow \frac{1}{2}\left[\mathbf{a}^T\mathbf{a} + \mathbf{b}^T\mathbf{b} - \mathbf{a}^T\mathbf{b} - \mathbf{b}^T\mathbf{a}\right] \geq 0 \\
&\Leftrightarrow (\mathbf{a} - \mathbf{b})^T(\mathbf{a} - \mathbf{b}) \geq 0.
\end{aligned}
$$

As $(\mathbf{a} - \mathbf{b})^T(\mathbf{a} - \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_2^2 \geq 0$ is clearly satisfied, the result is proved. ◻

Illustrations of the eigenvalue distribution of $\widehat{S}^{-1}S$ for a variety of values of β in a particular practical case are shown in Figure 4.1.

Therefore, by Theorem 4.1, we may obtain an effective Schur complement approximation if we can find a good way of approximating the matrices $\bar{K} + \frac{1}{\sqrt{\beta}}M$ and $\left(\bar{K} + \frac{1}{\sqrt{\beta}}M\right)^T$. The method we use for approximating these matrices is the geometric multigrid process described for the forward problem in Section 2.1: with the coarse grid matrices formed explicitly rather than by the use of prolongation and restriction operators and with block Gauss-Seidel smoothing.

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

So, as we now have good approximations of the matrices A and S, we can propose two effective preconditioners of the form

$$\widehat{\mathcal{P}}_1 = \begin{bmatrix} \widehat{A} & 0 \\ 0 & \widehat{S} \end{bmatrix}, \quad \widehat{\mathcal{P}}_2 = \begin{bmatrix} \widehat{A} & 0 \\ B & -\widehat{S} \end{bmatrix},$$

described in Section 2.2.

Unlike the forward problem, the convection-diffusion control problem is symmetric with our (symmetric) stabilization, and so $\widehat{\mathcal{P}}_1$ is symmetric positive definite. Therefore, our first method for solving the matrix system (3.4) would be to apply a MINRES method with preconditioner

$$\widehat{\mathcal{P}}_1 = \begin{bmatrix} \widehat{M} & 0 & 0 \\ 0 & \beta\widehat{M} & 0 \\ 0 & 0 & \widehat{S} \end{bmatrix}. \tag{4.1}$$

In our preconditioner, $\widehat{M}$ denotes 20 steps of Chebyshev semi-iteration to approximate the mass matrix M, and $\widehat{S}$ denotes the approximation to the Schur complement discussed above.

Our second method involves applying the Bramble-Pasciak Conjugate Gradient method as described in Section 2.2 with preconditioner

$$\widehat{\mathcal{P}}_2 = \begin{bmatrix} \gamma\widehat{M} & 0 & 0 \\ 0 & \beta\gamma\widehat{M} & 0 \\ \bar{K} & -M & -\widehat{S} \end{bmatrix} \tag{4.2}$$

and inner product given by

$$\mathcal{H} = \begin{bmatrix} M-\gamma\widehat{M} & 0 & 0 \\ 0 & \beta\left(M-\gamma\widehat{M}\right) & 0 \\ 0 & 0 & \widehat{S} \end{bmatrix},$$

where γ is a constant which can be chosen a priori to ensure that $M-\gamma\widehat{M}$ is positive definite; results for a 2D **Q1** mass matrix which may be applied to the test problems of Section 5 are provided in [20].

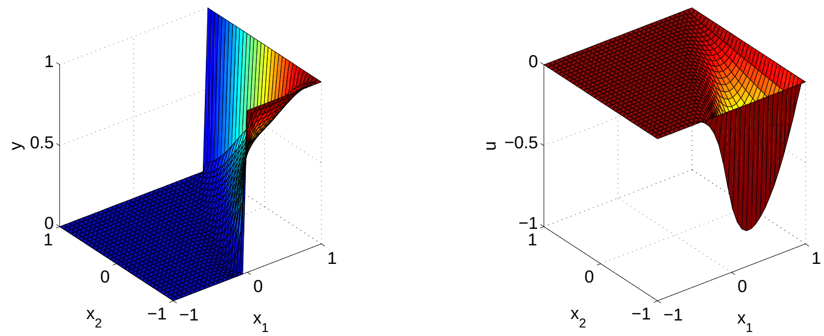
At this juncture, we make two points about our preconditioning strategy and its applicability:

1. The matrix system (3.4) for the distributed convection-diffusion control problem could potentially be reduced to the following system of equations by elimination of the discretized gradient equation

$$\begin{bmatrix} M & \bar{K}^T \\ \bar{K} & -\frac{1}{\beta}M \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ \mathbf{d} \end{bmatrix}, \quad \mathbf{p} = \beta\mathbf{u}.$$

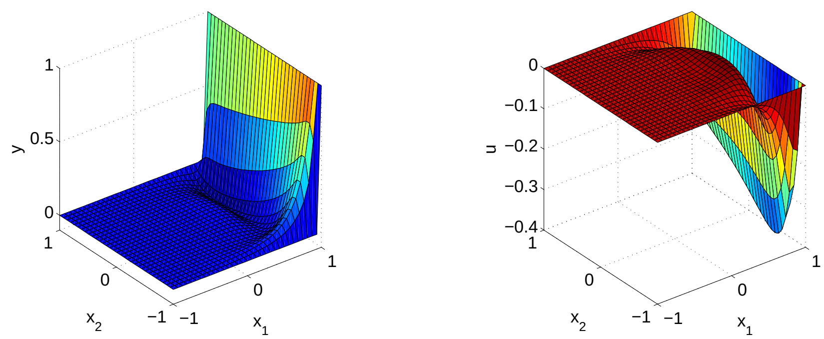
   We note that our preconditioning strategy could also be applied to this problem as we still obtain a saddle point system of the structure discussed in Section 2.2, so we will again need to implement a Chebyshev semi-iteration process to approximate M and enact the approximation of the Schur complement S, which remains the same as for the system (3.4). We avoid reducing the matrix system in this way here as we wish to keep the system in a form as general as possible—for example, if boundary control problems or problems involving control on a subdomain are considered, reducing the matrix system is not as simple. We note that results obtained when reducing the matrix system are similar to the case where it is not reduced.

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



(a) State y (b) Control u

FIG. 5.1. *Solutions of state and control for Problem 1 using* **Q1** *basis functions with* $\epsilon = \frac{1}{100}$ *and* $\beta = 1$.



(a) State y (b) Control u

FIG. 5.2. *Solutions of state and control for Problem 2 using* **Q1** *basis functions with* $\epsilon = \frac{1}{100}$ *and* $\beta = 1$.

2. We believe that other similar methods could be devised to solve the convection-diffusion control problem based on the framework discussed in this section. For instance, we see no reason why a preconditioner of the form

$$\widehat{\mathcal{P}}_3 = \begin{bmatrix} \widehat{A} & B^T \\ B & B\widehat{A}^{-1}B^T - \widehat{S} \end{bmatrix} = \begin{bmatrix} I & 0 \\ B\widehat{A}^{-1} & I \end{bmatrix} \begin{bmatrix} \widehat{A} & B^T \\ 0 & -\widehat{S} \end{bmatrix},$$

which was discussed in the context of the Poisson control problem in [21], could not be applied to this problem using our approximations $\widehat{A}$ and $\widehat{S}$.

5. Numerical results. In this section, we provide numerical results to illustrate the effectiveness of our suggested methods. In our numerical tests, we discretize the state y, the control u, and the adjoint p using **Q1** finite element basis functions.[*][†]

The two problems that we consider are stated below with plots of their solutions shown in Figures 5.1 and 5.2, respectively.

[*]We construct the relevant matrices for our two test problems in the same way as is done in the Incompressible Flow & Iterative Solver Software (IFISS) package [7, 22].

[†]All results are generated using a tri-core 2.5 GHz workstation.

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

TABLE 5.1

Number of MINRES *iterations with the 'ideal' block diagonal preconditioner* (4.1) *and Bramble-Pasciak* CG *iterations with the 'ideal' block triangular preconditioner* (4.2) *needed to solve Problem 1. Results are given for a range of values of* $\frac{h}{2}$ *(which is equal to the inverse of the number of steps in space in each coordinate) and* β*, where* $\epsilon = \frac{1}{250}$ *and* $\mathbf{Q1}$ *basis functions are used to approximate the state, control and adjoint.*

$\epsilon = \frac{1}{250}$		MINRES				BPCG			
		β				β			
$\frac{h}{2}$	SIZE	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-2}	10^{-4}	10^{-6}	10^{-8}
2^{-2}	75	13	7	5	3	11	9	6	6
2^{-3}	243	13	9	5	3	12	10	7	6
2^{-4}	867	13	11	5	3	12	13	9	7
2^{-5}	3267	13	12	7	3	13	14	10	7
2^{-6}	12675	13	12	7	4	13	14	12	8
2^{-7}	49923	12	11	9	5	13	15	15	10

- PROBLEM 1: We wish to solve the following distributed convection-diffusion control problem on $\Omega = [-1, 1]^2$

$$\min_{y,u} \frac{1}{2} \|y\|^2_{L_2(\Omega)} + \frac{\beta}{2} \|u\|^2_{L_2(\Omega)}$$
$$\text{s.t.} -\epsilon\nabla^2 y + \mathbf{w} \cdot \nabla y = u \ \text{ in } \Omega,$$
$$y = \begin{cases} 1 & \text{on } \partial\Omega_1 := ([0,1] \times \{-1\}) \cup (\{1\} \times [-1,1]), \\ 0 & \text{on } \partial\Omega \backslash \partial\Omega_1, \end{cases}$$

  where $\mathbf{w} = \left(\sin \frac{\pi}{6}, \cos \frac{\pi}{6}\right)^T$. This is an optimal control problem involving a constant wind $\mathbf{w}$; forward problems of this form have previously been considered in literature such as [8, 18].
- PROBLEM 2: We wish to solve the following distributed convection-diffusion control problem on $\Omega = [-1, 1]^2$

$$\min_{y,u} \frac{1}{2} \|y\|^2_{L_2(\Omega)} + \frac{\beta}{2} \|u\|^2_{L_2(\Omega)}$$
$$\text{s.t.} -\epsilon\nabla^2 y + \mathbf{w} \cdot \nabla y = u \ \text{ in } \Omega,$$
$$y = \begin{cases} 1 & \text{on } \partial\Omega_2 := \{1\} \times [-1,1], \\ 0 & \text{on } \partial\Omega \backslash \partial\Omega_2, \end{cases}$$

  where $\mathbf{w} = \left(\frac{1}{2}x_2(1 - x_1^2), -\frac{1}{2}x_1(1 - x_2^2)\right)^T$ and $\mathbf{x} = (x_1, x_2)^T$ denotes the spatial coordinates. This is an optimal control formulation of the *double-glazing problem* discussed in [8, p. 119]: a model of the temperature in a cavity with recirculating wind $\mathbf{w}$. We note that we have chosen the wind so that the maximum value of $\|\mathbf{w}\|_2$ on Ω is equal to 1.

We first provide a proof-of-concept that our proposed preconditioners are effective ones. In Table 5.1, we present iteration numbers for solving Problem 1 with $\epsilon = \frac{1}{250}$ and a range of h and β using 'ideal' versions of our two preconditioners (specifically, where we invert $\bar{K} + \frac{1}{\sqrt{\beta}} M$ and its transpose directly in the preconditioners rather than using a multigrid method). The results shown illustrate that in theory our preconditioners are highly potent for a range of parameters. All other results presented are thus generated using the geometric multigrid procedure previously described.

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

TABLE 5.2

Number of MINRES *iterations with block diagonal preconditioner* (4.1) *needed to solve Problem 1 and computation times taken to do so (in seconds). Results are given for a range of values of* $\frac{h}{2}$ *(and hence problem size) and* β *with* $\epsilon = \frac{1}{100}$ *and* $\epsilon = \frac{1}{500}$, *where* **Q1** *basis functions are used to approximate the state, control, and adjoint.*

MINRES $\epsilon = \frac{1}{100}$		β 10^{-2}		10^{-4}		10^{-6}		10^{-8}	
$\frac{h}{2}$	SIZE	ITER.	TIME	ITER.	TIME	ITER.	TIME	ITER.	TIME
2^{-2}	75	13	0.070	7	0.051	5	0.040	3	0.038
2^{-3}	243	13	0.11	9	0.092	5	0.072	3	0.063
2^{-4}	867	13	0.20	11	0.17	5	0.078	3	0.064
2^{-5}	3267	13	0.54	12	0.50	7	0.29	3	0.23
2^{-6}	12675	13	2.36	13	2.24	7	1.52	5	1.53
2^{-7}	49923	13	14.1	11	12.9	9	11.1	5	8.10

MINRES $\epsilon = \frac{1}{500}$		β 10^{-2}		10^{-4}		10^{-6}		10^{-8}	
$\frac{h}{2}$	SIZE	ITER.	TIME	ITER.	TIME	ITER.	TIME	ITER.	TIME
2^{-2}	75	13	0.072	7	0.054	5	0.044	3	0.038
2^{-3}	243	13	0.13	9	0.098	4	0.066	3	0.060
2^{-4}	867	13	0.27	11	0.15	5	0.084	3	0.062
2^{-5}	3267	13	0.58	12	0.52	7	0.42	3	0.27
2^{-6}	12675	13	2.93	12	2.73	7	1.76	4	1.21
2^{-7}	49923	12	15.2	11	15.1	9	10.2	5	9.51

In Table 5.2, we present the number of MINRES iterations and computation times (including the time taken to construct the relevant matrices on sub-grids) required to solve Problem 1 with $\epsilon = \frac{1}{100}$ and $\epsilon = \frac{1}{500}$ using the preconditioner $\widehat{\mathcal{P}}_1$ to a tolerance of 10^{-6}.[‡] In Table 5.3 we show how many Bramble-Pasciak CG iterations are required to solve the same problem to the same tolerance with the preconditioner $\widehat{\mathcal{P}}_2$ and with $\gamma = 0.95$.[§] We observe that both our solvers generate convergence in a small number of iterations for both values of the viscosity. The convergence rate actually improves as β decreases, probably because our Schur complement approximation becomes better for smaller β as illustrated by Figure 4.1. Although we take the wind $\mathbf{w} = \left(\sin\frac{\pi}{6}, \cos\frac{\pi}{6}\right)^T$ and specific values of ϵ, we find, in other computations not presented here, that the results are similar for any constant wind with vector 2-norm equal to 1 for a wide range of ϵ. We note that altering the boundary conditions or target function $\widehat{y}$ would not change the matrix within the system being solved, so our solvers seem to be very robust for problems involving constant winds and values of β which are of computational interest.

In Table 5.4, we present the number of preconditioned MINRES iterations and CPU times required to solve Problem 2, a harder problem, to the same tolerance, when $\epsilon = \frac{1}{100}$

[‡]In our numerical experiments, we set the viscosity to be of the same order as for the numerical tests for the forward problem in [17], however we note that our solvers are often very effective when ϵ is even smaller.

[§]We wish to choose γ reasonably close to 1 in order that the approximation of the $(1,1)$-block is effective but also far enough away from 1 to ensure that the inner product we work with is clearly positive definite. We find that the value $\gamma = 0.95$ meets these criteria in practice. Similar issues are discussed in [20] in the context of solving Poisson control problems.

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

TABLE 5.3

Number of Bramble-Pasciak CG *iterations with block triangular preconditioner* (4.2) *needed to solve Problem 1 and computation times taken to do so (in seconds). Results are given for a range of values of* $\frac{h}{2}$ *(and hence problem size) and* β *with* $\epsilon = \frac{1}{100}$ *and* $\epsilon = \frac{1}{500}$*, where* **Q1** *basis functions are used to approximate the state, control, and adjoint.*

BPCG $\epsilon = \frac{1}{100}$		β							
		10^{-2}		10^{-4}		10^{-6}		10^{-8}	
$\frac{h}{2}$	SIZE	ITER.	TIME	ITER.	TIME	ITER.	TIME	ITER.	TIME
2^{-2}	75	10	0.056	9	0.050	6	0.040	6	0.044
2^{-3}	243	12	0.11	10	0.11	7	0.084	6	0.075
2^{-4}	867	12	0.20	13	0.22	9	0.17	7	0.13
2^{-5}	3267	13	0.60	14	0.62	10	0.46	7	0.38
2^{-6}	12675	13	2.89	15	2.99	12	2.60	9	2.31
2^{-7}	49923	13	14.5	15	16.0	15	15.8	11	11.6

BPCG $\epsilon = \frac{1}{500}$		β							
		10^{-2}		10^{-4}		10^{-6}		10^{-8}	
$\frac{h}{2}$	SIZE	ITER.	TIME	ITER.	TIME	ITER.	TIME	ITER.	TIME
2^{-2}	75	11	0.057	8	0.048	6	0.047	6	0.043
2^{-3}	243	12	0.11	10	0.10	7	0.080	6	0.079
2^{-4}	867	12	0.22	13	0.22	9	0.16	7	0.14
2^{-5}	3267	13	0.52	14	0.55	10	0.45	7	0.36
2^{-6}	12675	13	2.91	14	2.96	12	2.68	8	2.01
2^{-7}	49923	13	13.7	15	14.8	14	14.2	9	10.5

and $\epsilon = \frac{1}{500}$; the number of preconditioned Bramble-Pasciak CG iterations required to solve this problem is shown in Table 5.5. Once more, for this problem and a wide range of values of β, our solvers are effective with convergence achieved in a very small number of iterations. We find that for this harder problem (with non-constant wind), the iteration numbers may rise very slightly for smaller ϵ in some cases (see Tables 5.4 and 5.5), however the iteration numbers in all cases are very reasonable.

We can see that the MINRES and Bramble-Pasciak CG methods are very competitive, and the results for both methods are similar. Whereas MINRES tends to converge in fewer iterations, the Bramble-Pasciak CG method is computationally cheaper for a fixed number of iterations. We note that the computation times for Bramble-Pasciak CG seem to be better for larger β (in particular for smaller h) and that the MINRES solver works better for smaller β due to the lower iteration numbers. We note that when β is small compared to h, as observed in Figure 4.1, the eigenvalues of the preconditioned Schur complement are highly clustered—consequently for smaller β the iteration numbers are particularly low for larger h and increase slightly as h is decreased. However the analysis of Section 4 and these results illustrate that the iteration count should be bounded by a low number for these problems as h decreases.

The results in this section illustrate that the solvers we have proposed are potent ones for a number of convection-diffusion control problems, a class of problems which, as for the convection-diffusion equation itself, is fraught with numerical difficulties. The number of iterations required to solve these problems is small, and the convergence of the solvers improves rather than degrades as β is decreased. As observable from the computation times shown in Tables 5.2–5.5, the convergence is close to linear with respect to the size of the

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

TABLE 5.4

Number of MINRES *iterations with block diagonal preconditioner* (4.1) *needed to solve Problem 2 and computation times taken to do so (in seconds). Results are given for a range of values of* $\frac{h}{2}$ *(and hence problem size) and* β *with* $\epsilon = \frac{1}{100}$ *and* $\epsilon = \frac{1}{500}$*, where* **Q1** *basis functions are used to approximate the state, control, and adjoint.*

MINRES $\epsilon = \frac{1}{100}$		β 10^{-2}		10^{-4}		10^{-6}		10^{-8}	
$\frac{h}{2}$	SIZE	ITER.	TIME	ITER.	TIME	ITER.	TIME	ITER.	TIME
2^{-2}	75	13	0.071	7	0.050	4	0.044	3	0.039
2^{-3}	243	15	0.13	7	0.063	4	0.061	3	0.059
2^{-4}	867	13	0.19	7	0.13	5	0.076	3	0.065
2^{-5}	3267	13	0.52	9	0.42	5	0.32	3	0.25
2^{-6}	12675	13	2.39	11	2.14	7	1.49	3	1.06
2^{-7}	49923	13	13.9	11	13.2	9	10.8	5	8.32

MINRES $\epsilon = \frac{1}{500}$		β 10^{-2}		10^{-4}		10^{-6}		10^{-8}	
$\frac{h}{2}$	SIZE	ITER.	TIME	ITER.	TIME	ITER.	TIME	ITER.	TIME
2^{-2}	75	15	0.074	7	0.053	5	0.041	3	0.040
2^{-3}	243	21	0.20	7	0.085	4	0.071	3	0.060
2^{-4}	867	19	0.35	9	0.17	5	0.085	3	0.064
2^{-5}	3267	12	0.55	9	0.47	5	0.33	3	0.28
2^{-6}	12675	12	2.81	9	2.34	5	2.10	3	1.17
2^{-7}	49923	12	15.4	11	14.7	5	8.92	3	7.71

matrix system—we find that the only part of the solvers that does not scale linearly in time is the construction of matrices on the sub-grids.

6. Conclusions. In this manuscript we have first given an overview of a GMRES approach for solving the convection-diffusion equation, as well as summarizing some general properties of saddle point systems and some possible solution methods for such systems.

We then introduced the convection-diffusion control problem and illustrated that, with a suitable stabilization technique (the Local Projection Stabilization), the same saddle point system arises whether the discretize-then-optimize approach or the optimize-then-discretize approach is used for solving the control problem.

We proposed two effective solvers for solving the convection-diffusion control problem: one involving a MINRES solver with a block diagonal preconditioner and one involving a Bramble-Pasciak Conjugate Gradient approach with a block triangular preconditioner. The key components of each of these preconditioners are a good approximation of the mass matrix, a powerful approximation of the Schur complement of the matrix system, and a geometric multigrid process which enables us to enact that Schur complement approximation.

We have shown theoretically that in an ideal case our preconditioners should be effective ones. Numerical results given in Section 5 indicate that our solvers do indeed perform well in practice for the problems we have tested, yielding fast and close to linear convergence as the problem size is increased; this rate of convergence improves as the regularization parameter β is decreased. We proved that the convergence rate cannot worsen as β is decreased if exact solves are used within a preconditioner and have illustrated numerically that the Chebyshev

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

TABLE 5.5

Number of Bramble-Pasciak CG *iterations with block triangular preconditioner* (4.2) *needed to solve Problem 2 and computation times taken to do so (in seconds). Results are given for a range of values of* $\frac{h}{2}$ *(and hence problem size) and* β *with* $\epsilon = \frac{1}{100}$ *and* $\epsilon = \frac{1}{500}$*, where* **Q1** *basis functions are used to approximate the state, control, and adjoint.*

BPCG $\epsilon = \frac{1}{100}$		β							
		10^{-2}		10^{-4}		10^{-6}		10^{-8}	
$\frac{h}{2}$	SIZE	ITER.	TIME	ITER.	TIME	ITER.	TIME	ITER.	TIME
2^{-2}	75	10	0.056	7	0.050	6	0.040	6	0.044
2^{-3}	243	12	0.10	8	0.097	6	0.078	6	0.077
2^{-4}	867	12	0.19	10	0.18	7	0.14	6	0.12
2^{-5}	3267	13	0.58	12	0.52	9	0.44	7	0.38
2^{-6}	12675	13	2.93	15	3.02	11	2.38	8	2.10
2^{-7}	49923	13	14.2	15	15.6	15	15.5	10	10.4

BPCG $\epsilon = \frac{1}{500}$		β							
		10^{-2}		10^{-4}		10^{-6}		10^{-8}	
$\frac{h}{2}$	SIZE	ITER.	TIME	ITER.	TIME	ITER.	TIME	ITER.	TIME
2^{-2}	75	12	0.061	7	0.046	6	0.045	6	0.043
2^{-3}	243	16	0.13	8	0.091	6	0.071	6	0.075
2^{-4}	867	17	0.25	9	0.16	7	0.13	6	0.13
2^{-5}	3267	13	0.54	11	0.45	7	0.38	6	0.34
2^{-6}	12675	13	2.86	13	2.88	9	2.28	7	1.85
2^{-7}	49923	13	13.6	15	15.4	11	12.7	7	9.14

semi-iteration and multigrid methods used show robustness in practice. We have observed that our solution methods work well whether SUPG or LPS stabilization is used. The methods also work well with no stabilization at all when such an approach is reasonable; for such diffusion-dominated problems, it is likely that more standard methods (including multigrid) could also be effective. If new stabilization methods are discovered for this problem, we might predict that our proposed preconditioners will again prove to be potentially useful for its solution.

Acknowledgements. The authors would like to thank two anonymous referees for their helpful comments and advice. The first author was supported for this work by the Engineering and Physical Sciences Research Council (UK), Grant EP/P505216/1.

REFERENCES

[1] R. BECKER AND M. BRAACK, *A finite element pressure gradient stabilization for the Stokes equations based on local projections*, Calcolo, 38 (2001), pp. 173–199.

[2] R. BECKER AND B. VEXLER, *Optimal control of the convection-diffusion equation using stabilized finite element methods*, Numer. Math., 106 (2007), pp. 349–367.

[3] M. BENZI, G. H. GOLUB, AND J. LIESEN, *Numerical solution of saddle point problems*, Acta Numer., 14 (2005), pp. 1–137.

[4] M. BRAACK AND E. BURMAN, *Local projection stabilization for the Oseen problem and its interpretation as a variational multiscale method*, SIAM J. Numer. Anal., 43 (2006), pp. 2544–2566.

[5] J. H. BRAMBLE AND J. E. PASCIAK, *A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems*, Math. Comp., 50 (1988), pp. 1–17.

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

[6] S. S. COLLIS AND M. HEINKENSCHLOSS, *Analysis of the streamline upwind/Petrov Galerkin method applied to the solution of optimal control problems*, Technical Report CAAM TR02-01, Dept. of Computational and Applied Math., Rice University, Houston, 2002.
[7] H. C. ELMAN, A. RAMAGE, AND D. SILVESTER, *Algorithm 866: IFISS, a Matlab Toolbox for Modelling Incompressible Flow*, ACM Trans. Math. Software, 33 (2007), Art. 14 (18 pages).
[8] H. C. ELMAN, D. J. SILVESTER, AND A. J. WATHEN, *Finite Elements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics*, Oxford University Press, New York, 2005.
[9] J.-L. GUERMOND, *Stabilization of Galerkin approximations of transport equations by subgrid modeling*, M2AN Math. Model. Numer. Anal., 33 (1999), pp. 1293–1316.
[10] M. HEINKENSCHLOSS AND D. LEYKEKHMAN, *Local error estimates for SUPG solutions of advection-dominated elliptic linear-quadratic optimal control problems*, SIAM J. Numer. Anal., 47 (2010), pp. 4607–4638.
[11] T. J. R. HUGHES AND A. BROOKS, *A multidimensional upwind scheme with no crosswind diffusion*, in Finite Element Methods for Convection Dominated Flows, T. J. R. Hughes, ed., AMD 34, Amer. Soc. Mech. Engrs., New York, 1979, pp. 19–35.
[12] I. C. F. IPSEN, *A note on preconditioning non-symmetric matrices*, SIAM J. Sci. Comput., 23 (2001), pp. 1050–1051.
[13] Y. A. KUZNETSOV, *Efficient iterative solvers for elliptic finite element problems on nonmatching grids*, Russian J. Numer. Anal. Math. Modelling, 10 (1995), pp. 187–211.
[14] M. F. MURPHY, G. H. GOLUB, AND A. J. WATHEN, *A note on preconditioning for indefinite linear systems*, SIAM J. Sci. Comput., 21 (2000), pp. 1969–1972.
[15] C. C. PAIGE AND M. A. SAUNDERS, *Solutions of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.
[16] J. W. PEARSON AND A. J. WATHEN, *A new approximation of the Schur complement in preconditioners for PDE-constrained optimization*, Numer. Linear Algebra Appl., 19 (2012), pp. 816–829.
[17] A. RAMAGE, *A multigrid preconditioner for stabilised discretisations of advection-diffusion problems*, J. Comput. Appl. Math., 110 (1999), pp. 187–203.
[18] T. REES, *Preconditioning iterative methods for PDE constrained optimization*, Ph.D. Thesis, New College, University of Oxford, Oxford, 2010.
[19] T. REES, H. S. DOLLAR, AND A. J. WATHEN, *Optimal solvers for PDE-constrained optimization*, SIAM J. Sci. Comput., 32 (2010), pp. 271–298.
[20] T. REES AND M. STOLL, *Block triangular preconditioners for PDE-constrained optimization*, Numer. Linear Algebra Appl., 17 (2010), pp. 977–996.
[21] J. SCHÖBERL AND W. ZULEHNER, *Symmetric indefinite preconditioners for saddle point problems with applications to PDE-constrained optimization problems*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 752–773.
[22] D. SILVESTER, H. ELMAN, AND A. RAMAGE, *Incompressible Flow and Iterative Solver Software (IFISS)*, version 3.1, 2011. http://www.manchester.ac.uk/ifiss/
[23] M. STOLL AND A. WATHEN, *Combination preconditioning and the Bramble-Pasciak$^+$ preconditioner*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 582–608.
[24] F. TRÖLTZSCH, *Optimal Control of Partial Differential Equations: Theory, Methods and Applications*, AMS, Providence, 2010.
[25] A. J. WATHEN AND T. REES, *Chebyshev semi-iteration in preconditioning for problems including the mass matrix*, Electron. Trans. Numer. Anal., 34 (2008/09), pp. 125–135.
http://etna.mcs.kent.edu/vol.34.2008-2009/pp125-135.dir