ITERATIVE METHODS FOR PROBLEMS IN COMPUTATIONAL FLUID DYNAMICS

HOWARD C. ELMAN*, DAVID J. SILVESTER† AND ANDREW J. WATHEN‡

Report CS-TR-3675
UMIACS-TR-96-58
August 1996

Abstract. We discuss iterative methods for solving the algebraic systems of equations arising from linearization and discretization of primitive variable formulations of the incompressible Navier-Stokes equations. Implicit discretization in time leads to a coupled but linear system of partial differential equations at each time step, and discretization in space then produces a series of linear algebraic systems. We give an overview of commonly used time and space discretization techniques, and we discuss a variety of algorithmic strategies for solving the resulting systems of equations. The emphasis is on preconditioning techniques, which can be combined with Krylov subspace iterative methods. In many cases the solution of subsidiary problems such as the discrete convection-diffusion equation and the discrete Stokes equations plays a crucial role. We examine iterative techniques for these problems and show how they can be integrated into effective solution algorithms for the Navier-Stokes equations.

* Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA, e-mail: elman@cs.umd.edu. This work was supported by the U. S. National Science Foundation under grants ASC-8958544 and DMS-9423133.
† Department of Mathematics, University of Manchester Institute of Science and Technology, Manchester M601QD, UK, email: djs@lanczos.ma.umist.ac.uk
‡ Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford OX13QD, UK, email: Andy.Wathen@comlab.ox.ac.uk

ITERATIVE METHODS FOR PROBLEMS IN COMPUTATIONAL FLUID DYNAMICS

HOWARD C. ELMAN*, DAVID J. SILVESTER† AND ANDREW J. WATHEN‡

Abstract. We discuss iterative methods for solving the algebraic systems of equations arising from linearization and discretization of primitive variable formulations of the incompressible Navier-Stokes equations. Implicit discretization in time leads to a coupled but linear system of partial differential equations at each time step, and discretization in space then produces a series of linear algebraic systems. We give an overview of commonly used time and space discretization techniques, and we discuss a variety of algorithmic strategies for solving the resulting systems of equations. The emphasis is on preconditioning techniques, which can be combined with Krylov subspace iterative methods. In many cases the solution of subsidiary problems such as the discrete convection-diffusion equation and the discrete Stokes equations plays a crucial role. We examine iterative techniques for these problems and show how they can be integrated into effective solution algorithms for the Navier-Stokes equations.

1. Introduction. Our objective is to compute solutions of incompressible flow problems modelled by the Navier-Stokes equations in a flow domain $\Omega \subset \mathbb{R}^d$ ($d = 2$ or 3) with a piecewise smooth boundary $\partial\Omega$:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} - \nu\nabla^2\mathbf{u} + \nabla p = 0 \quad \text{in } \mathcal{W} \equiv \Omega \times (0,T) \tag{1.1}$$

$$\nabla\cdot\mathbf{u} = 0 \quad \text{in } \mathcal{W}. \tag{1.2}$$

together with boundary and initial conditions of the form

$$\mathbf{u}(\mathbf{x},t) = \mathbf{g}(\mathbf{x},t) \qquad \text{on } \overline{\mathcal{W}} \equiv \partial\Omega \times [0,T]; \tag{1.3}$$

$$\mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}) \qquad \text{in } \Omega. \tag{1.4}$$

Our notation is standard: $\mathbf{u}$ is the fluid velocity, p is the pressure, $\nu > 0$ is a specified viscosity parameter (in a non-dimensional setting it is the inverse of the Reynolds number), and $T > 0$ is some final time. The initial velocity field $\mathbf{u_0}$ will be assumed to satisfy the incompressibility constraint, that is, $\nabla\cdot\mathbf{u_0} = 0$. The boundary velocity field satisfies $\int_{\partial\Omega} \mathbf{g}\cdot\mathbf{n}\,\mathrm{d}s = 0$ for all time t, where $\mathbf{n}$ is the unit vector normal to $\partial\Omega$. We also assume that the pressure solution is uniquely specified e.g. by insisting that its mean value is zero.

If $\mathbf{g}$ is independent of t then the usual objective is simply to compute steady-state solutions of (1.1)–(1.2). In other cases however, time-accuracy is important and the requirements of the time discretisation will be more demanding; specifically, an accurate and unconditionally stable time-discretisation is necessary to adaptively change the timestep to reflect the dynamics of the underlying flow. Two classes of time discretisation scheme are described below, operator splitting methods and linearised implicit methods.

* Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA, e-mail: elman@cs.umd.edu. This work was supported by the U. S. National Science Foundation under grants ASC-8958544 and DMS-9423133.

† Department of Mathematics, University of Manchester Institute of Science and Technology, Manchester M601QD, UK, email: djs@lanczos.ma.umist.ac.uk

‡ Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford OX13QD, UK, email: Andy.Wathen@comlab.ox.ac.uk

1.1. Operator splitting Methods. One attractive approach ensuring stability and high accuracy is to decouple the convection and incompressibility operators using an "alternating-direction" splitting; see Glowinski & Dean [5]. Assuming uniform timesteps $\Delta\mathrm{t} = T/n$ for ease of exposition, the simplest two-stage (Peaceman-Rachford) scheme [39] is given below.

ALGORITHM 1.1. *Given* $\mathbf{u}^0$, $\theta \in [0,1]$, $\alpha \in (0,1)$, $\beta \in (0,1)$, *find* $\mathbf{u}^1, \mathbf{u}^2, \ldots, \mathbf{u}^n$ *via*

$$\begin{aligned} \frac{\mathbf{u}^{n+\theta} - \mathbf{u}^n}{\theta\Delta\mathrm{t}} - \alpha\nu\nabla^2\mathbf{u}^{n+\theta} + \mathbf{u}^* \cdot \nabla\mathbf{u}^{n+\theta} &= \beta\nu\nabla^2\mathbf{u}^n - \nabla p^n \quad \text{in } \Omega, \\ \mathbf{u}^{n+\theta} &= \mathbf{g}^{n+\theta} \quad \text{on } \partial\Omega. \end{aligned} \tag{1.5}$$

$$\begin{aligned} \frac{\mathbf{u}^{n+1} - \mathbf{u}^{n+\theta}}{(1-\theta)\Delta\mathrm{t}} - \beta\nu\nabla^2\mathbf{u}^{n+1} + \nabla p^{n+1} &= \alpha\nu\nabla^2\mathbf{u}^{n+\theta} - \mathbf{u}^* \cdot \nabla\mathbf{u}^{n+\theta} \\ \nabla \cdot \mathbf{u}^{n+1} &= 0 \quad \text{in } \Omega, \\ \mathbf{u}^{n+1} &= \mathbf{g}^{n+1} \quad \text{on } \partial\Omega. \end{aligned} \tag{1.6}$$

In practice the choice of parameters is restricted; the splitting of the diffusive terms must be done consistently i.e. $\alpha + \beta = 1$, and the "frozen" velocity in the convective term must be divergence free i.e. $\nabla \cdot \mathbf{u}^* = 0$, otherwise the skew symmetry of the convective term will not be preserved. Another important consideration with regard to the choice of $\mathbf{u}^*$ is the linearity, or otherwise, of the equation systems that must be solved at each time level. In particular, the natural choice of $\mathbf{u}^* = \mathbf{u}^n$ gives a linear method but in this case the accuracy is only first order. Second order accuracy can be achieved by setting $\mathbf{u}^* = \mathbf{u}^{n+\theta}$, but in this case a nonlinear convection-diffusion problem (1.5) must be solved at every time level, in addition to the generalised Stokes problem (1.6).

The Peaceman-Rachford splitting method has one drawback (see [5] and [46]), namely, that it is not asymptotically stable when applied to the standard model problem with an exponentially decaying solution. Thus we can expect any implementation of Algorithm 1.1 to perform poorly if the time-step is not small enough when the underlying flow exhibits fast transient behaviour. In addition, the methodology is not well suited to computing steady-state flow solutions by "pseudo-timestepping" with large timesteps. Motivated by these observations, Glowinski [5] proposed a three-stage variant of the Peaceman-Rachford scheme which has all the good features of the original method whilst retaining stability in the asymptotic limit $\Delta\mathrm{t} \to \infty$. The resulting method is commonly referred to as "Le θ-scheme".

ALGORITHM 1.2. *Given* $\mathbf{u}^0$, $\theta \in (0,1/2)$, $\alpha \in (0,1)$, $\beta \in (0,1)$, *find* $\mathbf{u}^1$, $\mathbf{u}^2, \ldots,$ $\mathbf{u}^n$ *via*

$$\begin{aligned} \frac{\mathbf{u}^{n+\theta} - \mathbf{u}^n}{\theta\Delta\mathrm{t}} - \alpha\nu\nabla^2\mathbf{u}^{n+\theta} + \nabla p^{n+\theta} &= \beta\nu\nabla^2\mathbf{u}^n - \mathbf{u}^n \cdot \nabla\mathbf{u}^n \\ \nabla \cdot \mathbf{u}^{n+\theta} &= 0 \quad \text{in } \Omega, \\ \mathbf{u}^{n+\theta} &= \mathbf{g}^{n+\theta} \quad \text{on } \partial\Omega. \end{aligned} \tag{1.7}$$

$$\begin{aligned} \frac{\mathbf{u}^{n+1-\theta} - \mathbf{u}^{n+\theta}}{(1-2\theta)\Delta\mathrm{t}} \quad - \quad & \beta\nu\nabla^2\mathbf{u}^{n+1-\theta} + \ldots \\ & \mathbf{u}^* \cdot \nabla\mathbf{u}^{n+1-\theta} = \alpha\nu\nabla^2\mathbf{u}^{n+\theta} - \nabla p^{n+\theta} \quad \text{in } \Omega, \\ \mathbf{u}^{n+1-\theta} \quad = \quad & \mathbf{g}^{n+1-\theta} \quad \text{on } \partial\Omega. \end{aligned} \tag{1.8}$$

$$\begin{aligned}
\frac{\mathbf{u}^{n+1}-\mathbf{u}^{n+1-\theta}}{\theta\Delta\mathrm{t}} &- \alpha\nu\nabla^2\mathbf{u}^{n+1}+\ldots \\
&\qquad \nabla p^{n+1} = \beta\nu\nabla^2\mathbf{u}^{n+1-\theta}-\mathbf{u}^*\cdot\nabla\mathbf{u}^{n+1-\theta} \\
\nabla\cdot\mathbf{u}^{n+1} &= 0 \qquad \text{in } \Omega, \\
\mathbf{u}^{n+1} &= \mathbf{g}^{n+1} \quad \text{on } \partial\Omega.
\end{aligned} \tag{1.9}$$

The nonlinear scheme $\mathbf{u}^*=\mathbf{u}^{n+1-\theta}$ is considered in [5], and requires the solution of two generalised Stokes problems (1.7),(1.9), and one nonlinear convection-diffusion equation (1.8) at each time step. In this case, either choosing $\alpha{=}\beta=1/2$ or else setting $\theta=1-1/\sqrt{2}$ with $\alpha+\beta=1$ gives second order accuracy as $\Delta\mathrm{t}\to 0$. In particular, setting $\theta=1-1/\sqrt{2}$, $\alpha=(1-2\theta)/(1-\theta)$ and $\beta=\theta/(1-\theta)$ gives a method which is second order accurate in time, unconditionally stable, has good asymptotic properties and has commonality between the coefficient matrices at the various substages, see [5]. The unconditional stability of the scheme in an incompressible Navier-Stokes setting was established by Klouček & Rys [31].

In [46] a linear θ-scheme is developed which retains second order accuracy (setting $\mathbf{u}^*=\mathbf{u}^{n+\theta}$ in Algorithm 1.2 reduces the accuracy to first order). The key here is to use an appropriate combination of the two convection matrices when freezing the velocity in Algorithm 1.2, viz:

$$\mathbf{u}^*=\frac{2\theta-1}{\theta}\mathbf{u}^n+\frac{1-\theta}{\theta}\mathbf{u}^{n+\theta}. \tag{1.10}$$

In this case a linear convection-diffusion problem (1.8) must be solved at each time level. Furthermore, the results in [46] show that the accuracy of the resulting method is not compromised. In contrast, making the choice $\mathbf{u}^*=\mathbf{u}^{n+\theta}$ badly impinges on accuracy as $\Delta\mathrm{t}\to 0$.

Summarising the discussion of splitting methods; both the two-stage Algorithm 1.1 in the case of small $\Delta\mathrm{t}$, and the three-stage Algorithm 1.2 in general, give accurate time discretisation of the Navier-Stokes problem (1.1)–(1.4). In sections 3 and 4 that follow we describe how the component Stokes and (linear) convection-diffusion sub-problems may be solved efficiently using contemporary preconditioned Krylov subspace iteration methods. The spatial discretisations of the convection-diffusion and Stokes subproblems which arise above are discussed in section 2.

If ν is small then an efficient alternative to operator-splitting is to use the "characteristics" of the associated hyperbolic problem (looking backwards in time to ensure stability), see Douglas & Russell [6]. Using this approach a single Stokes problem of the form (1.6) must be solved at every time-level so the discussion in section 4 is also relevant to this class of methods.

1.2. Linearised Implicit Methods. The simplest time-stepping approach for the Navier-Stokes equations is a simple one-stage finite difference discretisation. A generic (and unconditionally stable) algorithm (cf. Algorithms 1.1 and 1.2) is given below.

ALGORITHM 1.3. *Given* $\mathbf{u}^0$, $\theta\in[1/2,1]$, *find* $\mathbf{u}^1$, $\mathbf{u}^2$, …, $\mathbf{u}^n$ *via*

$$\begin{aligned}
\frac{(\mathbf{u}^{n+1}-\mathbf{u}^n)}{\Delta\mathrm{t}}+\mathbf{u}^*\cdot\nabla\mathbf{u}^{n+\theta} &- \nu\nabla^2\mathbf{u}^{n+\theta}+\nabla p^{n+\theta}=0 \\
\nabla\cdot\mathbf{u}^{n+\theta} &= 0 \qquad \text{in } \Omega, \\
\mathbf{u}^{n+\theta} &= \mathbf{g}^{n+\theta} \quad \text{on } \partial\Omega.
\end{aligned} \tag{1.11}$$

Here $\mathbf{u}^{n+\theta} = \theta\mathbf{u}^{n+1} + (1-\theta)\mathbf{u}^n$ and $p^{n+\theta} = \theta p^{n+1} + (1-\theta)p^n$. Note that p^0 is required if $\theta \neq 1$ so the Algorithm 1.3 is not self-starting in general. In this case an approximation to p^0 must be computed explicitly by manipulation of the continuum problem, or alternatively it must be approximated by taking one (very small) step of a self-starting algorithm (e.g. with $\theta = 1$ above).

Algorithm 1.3 contains the well known nonlinear schemes of backward Euler and Crank-Nicolson. These methods are given by ($\mathbf{u}^{n+\theta} = \mathbf{u}^{n+1}$, $\mathbf{u}^* = \mathbf{u}^{n+1}$), ($\mathbf{u}^{n+\theta} = \mathbf{u}^{n+\frac{1}{2}}$, $\mathbf{u}^* = \mathbf{u}^{n+\frac{1}{2}}$), and are first and second order accurate respectively. In either case, a nonlinear problem must be solved at every time-level. As a result neither of these methods is to be recommended if time-accuracy is needed. A well known linearisation strategy is to set $\mathbf{u}^* = \mathbf{u}^n$ above. This does not affect the stability properties of the time-discretisation, but it does reduce the Crank-Nicolson accuracy to first order as $\Delta t \to 0$ (the first order accuracy of backward Euler is unchanged). To retain second order accuracy in a linear scheme the Simo-Armero scheme [45] given by setting $\mathbf{u}^{n+\alpha} = \mathbf{u}^{n+\frac{1}{2}}$ with $\mathbf{u}^* = (3\mathbf{u}^n - \mathbf{u}^{n-1})/2$ in Algorithm 1.3 is recommended.

Using linearised backward Euler (or the Simo-Armero scheme) a frozen-coefficient Navier-Stokes problem (or *Oseen* problem) arises at each discrete time step. In contrast to the operator splitting case, the Oseen methodology is primarily of interest when solving steady-state problems—the linearised backward Euler method is uniquely well suited to pseudo-timestepping since it inherits the long term asymptotic dissipative behaviour of (1.1)–(1.2), see [45] for details. Alternatively, attacking the steady state version of (1.1)–(1.2) directly introduces a (steady-state) Oseen system at every iterative level. In section 5 we consider techniques for solving such Oseen problems using preconditioned Krylov subspace methods.

2. Spatial discretisation.

In this section, the spatial discretisation of the sub-problems arising from the operator splitting methods in section 1.1 are discussed. For simplicity, we only consider the *steady-state* limit of the linearised convection-diffusion and Stokes sub-problems here; for example, as would arise from setting $\Delta t \to \infty$ in (1.5) and (1.6) respectively.

2.1. The linearised convection-diffusion problem.

The problem addressed here is the following: Given some convective velocity field (or "wind") $\mathbf{w} \in \mathbb{R}^d$ such that $\nabla \cdot \mathbf{w} = 0$, find a scalar variable u (the transported quantity) satisfying

$$-\nu\nabla^2 u + \mathbf{w}\cdot\nabla u = f \qquad \text{in } \Omega, \tag{2.1}$$

with a boundary condition $u(\mathbf{x}) = g(\mathbf{x})$ on $\partial\Omega$. In practice, for example when solving (1.5), the "wind" is not actually pointwise divergence-free. Our discussion is still relevant in such cases—our starting point is then an equivalent formulation of the momentum conservation equations (1.1), with the convection term expressed in *skew-symmetric form*, see [45] for details.

Simple finite difference methods are often appropriate when spatially discretising the model problem (2.1)), especially if the geometry is straightforward and "fast solution" is the goal. Alternatively, if the flow domain is irregular or if adaptive refinement via a posteriori error control is to be included, then finite element spatial approximation is best. The theory underlying finite element approximation of (2.1) is summarised for completeness below. For further details, see for example, Quarteroni & Valli [40].

The weak formulation of (2.1) is defined in terms of the Sobolev space $H_0^1(\Omega)$ (the set of functions with derivatives in $L^2(\Omega)$ and which are zero on $\partial\Omega$). Defining the space $X \equiv H_0^1(\Omega)$, it is easy to see that the solution u satisfies

$$a(u, v) = (f, v) \qquad \forall v \in X, \tag{2.2}$$

where $a(\cdot, \cdot)$ is the bilinear form $a(u, v) = \nu(\nabla u, \nabla v) + (\mathbf{w} \cdot \nabla u, v)$, and $(\cdot, \cdot)$ denotes the usual scalar $L^2(\Omega)$ inner product.

Since Ω is bounded and $\mathbf{w}$ is divergence-free, the bilinear form $a(\cdot, \cdot)$ is coercive and bounded over X

$$a(u, u) = \nu \|\nabla u\|^2 \qquad \forall u \in X, \tag{2.3}$$

$$|a(u, v)| \leq C_{\mathbf{w}} \|\nabla u\| \|\nabla v\| \qquad \forall u \in X, \forall v \in X, \tag{2.4}$$

and the continuity constant $C_{\mathbf{w}}$ is given by

$$C_{\mathbf{w}} = \nu + C_\Omega \|\mathbf{w}\|_{L^\infty(\Omega)}$$

where C_Ω is the Poincaré constant associated with Ω. Existence and uniqueness of the solution to (2.1) then follows from the Lax-Milgram lemma.

To generate a discrete system we take a finite dimensional subspace $X_h \subset X$, where h is a representative mesh parameter, and enforce (2.2) over X_h. Specifically, we look for a function u_h such that $u_h = g_h$ on $\partial\Omega$, which solves

$$a(u_h, v) = (f_h, v) \qquad \forall v \in X_h, \tag{2.5}$$

where f_h is the $L^2(\Omega)$ orthogonal projection of f into X_h, and g_h is typically the interpolant of of the boundary data g.

Since we are using a conforming approximation, u_h is also uniquely defined, and if $g = 0$, (2.3) and (2.4) imply the following a priori error estimate

$$\|\nabla(u - u_h)\| \leq \frac{C_{\mathbf{w}}}{\nu} \inf_{v \in X_h} \|\nabla(u - v)\|. \tag{2.6}$$

Although the finite element approximation in (2.6) is of optimal order as $h \to 0$, the stability clearly depends on the ratio $C_{\mathbf{w}}/\nu$. In general, oscillatory solutions are observed if the characteristic "mesh Peclet number" is large, i.e.

$$P_e \equiv \frac{h\|\mathbf{w}\|}{2\nu} > 1,$$

for example, if there are any boundary layers which are not resolved by the mesh. In general, when convection dominates, the discrete solution "inherits" instability from the associated solution of (2.2).

An alternative to adaptive mesh refinement is to "ignore" physical boundary layers, and to stabilise the discrete problem; e.g. using some form of *upwinded* discretisation. In a finite element setting this is conveniently achieved using a Petrov-Galerkin framework [27, 28] with a "shifted" (non-conforming) test space, say,

$$a(u_h, v + \delta\mathbf{w} \cdot \nabla v) = (f_h, v + \delta\mathbf{w} \cdot \nabla v) \qquad \forall v \in X_h, \tag{2.7}$$

where δ is an appropriately chosen stabilisation/upwinding parameter, see below. Taking a standard element-wise evaluation of the non-conforming term, and using a linear P_1 (or Q_1) approximation space, the formulation (2.7) simplifies to the so-called *streamline-diffusion* method

$$(2.8)\quad b(u_h, v) \equiv a(u_h, v) + (\mathbf{w}\cdot\nabla u_h, \delta\mathbf{w}\cdot\nabla v) = (f_h, v + \delta\mathbf{w}\cdot\nabla v) \qquad \forall v \in X_h.$$

This formulation clearly has better stability properties than the original since there is additional coercivity in the local flow direction,

$$(2.9)\qquad b(u,u) = \nu\,\|\nabla u\|^2 + \delta\|\mathbf{w}\cdot\nabla u\|^2 \quad \forall u \in X_h.$$

Another appealing feature of the stabilised formulation (2.7) is that it is *consistent*— the exact solution of the differential equation (2.1) satisfies (2.7). This means that high order approximations (P_k or Q_k for $k \geq 2$) can be used without compromising accuracy.

Returning now to the choice of δ, it is possible to show that the solution of (2.7) satisfies the "best possible" error estimate (for any degree of polynomial approximation), under the assumption that δ in (2.7) is of the form

$$(2.10)\qquad \delta = \frac{\alpha h}{\|\mathbf{w}\|} \quad \text{for all } P_e > 1.$$

Here $\alpha > 0$ is a "tuning parameter", and h is the usual representative mesh parameter. For a more complete discussion, and a review of the error analysis of the streamline diffusion method, see [28]. Note that if the discretised problem is diffusion-dominated (i.e. $P_e \leq 1$) then the corresponding "best" choice above is $\delta = 0$, in which case (2.7) reduces to the standard Galerkin formulation (2.5).

In practice, determining an appropriate choice of α in (2.10) is crucial. (This issue will also arise when we consider stabilised Stokes formulations below). There are two aspects to consider here: firstly, it very easy to over-stabilise giving smooth but inaccurate solutions, secondly, the performance of iterative solvers applied to (2.8) will clearly be influenced by the choice of parameter. This second aspect will be an issue in section 3, where some experiments are presented for a model problem with constant "wind" $\mathbf{w}$, solving (2.5) and (2.7), using uniform grids of bilinear finite elements. (The associated software is freely available, see section 6 for details.) If the wind is constant, then an optimal value is known (from Fourier analysis) $\alpha^* = 1/2(1 - 1/P_e)$, which minimises the contraction rate of iterative solvers applied to (2.8), see [17] for further details. Note that $\alpha^* \to 0$ as $P_e \to 1$ so that "stabilising" the standard method is likely to adversely affect the convergence of iterative solvers if the discrete problem is diffusion dominated.

2.2. The Stokes problem. Here we consider the following problem: find the velocity vector $\mathbf{u} \in \mathbb{R}^d$ and the scalar p (the "pressure") satisfying

$$(2.11)\qquad -\nabla^2\mathbf{u} + \nabla p = f \quad \text{in } \Omega$$

$$(2.12)\qquad \nabla\cdot\mathbf{u} = 0 \quad \text{in } \Omega,$$

with specified velocity boundary conditions

$$(2.13)\qquad \mathbf{u}(\mathbf{x}) = \mathbf{g}(\mathbf{x}) \quad \text{on } \partial\Omega.$$

Note that in (2.11)–(2.12) the viscosity coefficient ν has been incorporated into the definition of the forcing function and the pressure.

The theory underlying the solution of (2.11)–(2.13) using finite element methods is outlined below. For full details see Girault & Raviart [21]. The weak formulation of (2.11)–(2.12) is defined in terms of the Sobolev spaces $H_0^1(\Omega)$ and $L_0^2(\Omega)$ (the set of functions in $L^2(\Omega)$ with zero mean value on Ω). Defining a velocity space $\mathbf{X} \equiv (H_0^1(\Omega))^d$ and a pressure space $M \equiv L_0^2(\Omega)$, it is easy to see that the solution $(\mathbf{u}, p)$ of (2.11)–(2.12) satisfies

$$(\nabla \mathbf{u}, \nabla \mathbf{v}) - (p, \nabla \cdot \mathbf{v}) = (\mathbf{f}, \mathbf{v}) \qquad \forall \mathbf{v} \in \mathbf{X} \tag{2.14}$$

$$(\nabla \cdot \mathbf{u}, q) = 0 \qquad \forall q \in M, \tag{2.15}$$

where $(\cdot, \cdot)$ denotes the usual vector or scalar $L^2(\Omega)$ inner product. Since Ω is bounded and connected there exists a constant κ satisfying the continuous *inf-sup* condition:

$$\sup_{\mathbf{w} \in \mathbf{X}} \frac{(p, \nabla \cdot \mathbf{w})}{\|\mathbf{w}\|_{\mathbf{X}}} \geq \kappa \|p\|_M \quad \forall p \in M. \tag{2.16}$$

Existence and uniqueness of solution follows, see [21].

To generate a discrete system we take finite dimensional subspaces $\mathbf{X}_h \subset \mathbf{X}$ and $M_h \subset L^2(\Omega)$, where h is a representative mesh parameter, and enforce (2.14)–(2.15) over the discrete subspaces (again specifying that functions in M_h have zero mean to ensure uniqueness). Specifically, we look for functions $\mathbf{u}_h$ and p_h such that

$$(\nabla \mathbf{u}_h, \nabla \mathbf{v}) - (p_h, \nabla \cdot \mathbf{v}) = (\mathbf{f}_h, \mathbf{v}) \qquad \forall \mathbf{v} \in \mathbf{X}_h \tag{2.17}$$

$$(\nabla \cdot \mathbf{u}_h, q) = 0 \qquad \forall q \in M_h. \tag{2.18}$$

Here, $\mathbf{f}_h$ is the $(L^2(\Omega))^d$ orthogonal projection of $\mathbf{f}$ into $\mathbf{X}_h$.

The well-posedness of (2.17)–(2.18) is not automatic since we do not have an internal approximation (i.e. functions satisfying (2.18) do not necessarily satisfy (2.15)). A sufficient condition for the existence and uniqueness of the solution to (2.17)–(2.18) is that the following *discrete inf-sup* condition is satisfied: there exists a constant γ independent of h such that

$$\sup_{\mathbf{w} \in \mathbf{X}_h} \frac{(p, \nabla \cdot \mathbf{w})}{\|\nabla \mathbf{w}\|} \geq \gamma \|p\| \quad \forall p \in M_h. \tag{2.19}$$

Note that the semi-norm $\|\nabla \mathbf{w}\|$ in (2.19) is equivalent to the norm $\|\mathbf{w}\|_{\mathbf{X}}$ used in (2.16) for functions $\mathbf{w} \in \mathbf{X}$. In the case $\mathbf{g} = \mathbf{0}$ the condition (2.19) also guarantees optimal approximation in the sense of the error estimate

$$\|\nabla(\mathbf{u} - \mathbf{u}_h)\| + \|p - p_h\| \leq C(\inf_{\mathbf{v} \in \mathbf{X}_h} \|\nabla(\mathbf{u} - \mathbf{v})\| + \inf_{q \in M_h} \|p - q\|). \tag{2.20}$$

Note that the constant C in (2.20) is inversely proportional to the inf-sup constant γ in (2.19).

The simplest example of an unstable method is the computationally convenient equal-order velocity/pressure approximation based on a single grid. The problem is that the pressure space is too rich compared to the velocity space in this case. The simplest way of constructing an equal order approximation such that (2.19) is uniformly satisfied is to introduce two grids: for example in $\mathbb{R}^2$ starting from a coarse grid of rectangles, a refined grid can be constructed by joining the mid-points of the edges. The condition (2.19) is then satisfied by taking a C^0 piecewise bilinear function on the coarse mesh for the pressure approximation, and a C^0 piecewise bilinear function on the fine mesh for each of the velocity components. Numerical results presented in sections 4 and 5 were generated using this approach—henceforth referred as the Q_1–iso–Q_2 method.

To construct the matrix analogue of (2.17)–(2.18) it is convenient to introduce discrete operators $\mathcal{A} : \mathbf{X}_h \mapsto \mathbf{X}_h$ and $\mathcal{B} : \mathbf{X}_h \mapsto M_h$ defined via

$$(\mathcal{A}\mathbf{v}_h, \mathbf{w}_h) = (\nabla \mathbf{v}_h, \nabla \mathbf{w}_h) \qquad \forall \mathbf{v}_h, \mathbf{w}_h \in \mathbf{X}_h, \tag{2.21}$$

$$(\mathcal{B}\mathbf{v}_h, q_h) = -(\nabla \cdot \mathbf{v}_h, q_h) \qquad \forall \mathbf{v}_h \in \mathbf{X}_h, \forall q_h \in M_h, \tag{2.22}$$

so that $\mathcal{B}^*$ is the adjoint of $\mathcal{B}$, i.e. $(\mathbf{v}_h, \mathcal{B}^* q_h) = (\mathcal{B}\mathbf{v}_h, q_h)$. With these definitions the discrete problem (2.17)–(2.18) can be rewritten as a matrix system:

$$\begin{pmatrix} \mathcal{A} & \mathcal{B}^* \\ \mathcal{B} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}_h \\ p_h \end{pmatrix} = \begin{pmatrix} \mathbf{f}_h \\ 0 \end{pmatrix}. \tag{2.23}$$

Furthermore, the inf-sup inequality (2.19) simplifies to

$$\gamma \|p_h\| \le \sup_{\mathbf{w}_h \in \mathbf{X}_h} \frac{(\mathcal{B}\mathbf{w}_h, p_h)}{(\mathcal{A}\mathbf{w}_h, \mathbf{w}_h)^{1/2}} \qquad \forall p_h \in M_h. \tag{2.24}$$

It is instructive to express the inf-sup condition in terms of the actual finite element matrices that arise in practice. To this end, let us explicitly introduce the finite element basis sets, say,

$$\mathbf{X}_h = \mathrm{span}\{\phi_i\}_{i=1}^n, \quad M_h = \mathrm{span}\{\psi_j\}_{j=1}^m; \tag{2.25}$$

and associate the functions $\mathbf{u}_h$, p_h, $\mathbf{f}_h$ with the vectors $u \in \mathbb{R}^n$, $p \in \mathbb{R}^m$ and $f \in \mathbb{R}^n$ of generalised coefficients, $\mathbf{u}_h = \sum_{i=1}^n u_i \phi_i$ etc. Defining the $n \times n$ "vector-stiffness matrix" $A_{ij} = (\nabla \phi_i, \nabla \phi_j)$ and also the $m \times n$ "divergence matrix" $B_{ij} = -(\nabla . \phi_j, \psi_i)$, gives the finite element version of (2.23):

$$\begin{pmatrix} A & B^t \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}. \tag{2.26}$$

Moreover, introducing the $m \times m$ pressure "mass matrix" $Q_{ij} = (\psi_i, \psi_j)$; leads to the finite element version of (2.19) or (2.24): for all $p \in \mathbb{R}^m$,

$$\gamma (p^t Q p)^{1/2} \le \max_u \frac{p^t B u}{(u^t A u)^{1/2}} \tag{2.27}$$

$$= \max_{w = A^{1/2} u} \frac{p^t B A^{-1/2} w}{(w^t w)^{1/2}} \tag{2.28}$$

$$= (p^t B A^{-1} B^t p)^{1/2}, \tag{2.29}$$

since the maximum is attained when $w = A^{-1/2} B^t p$. Thus, we have a characterisation of the inf-sup constant:

$$\gamma^2 = \min_{p \ne 0} \frac{p^t B A^{-1} B^t p}{p^t Q p}. \tag{2.30}$$

In simple terms it is precisely the square root of the smallest eigenvalue of the Schur complement preconditioned by the pressure mass matrix: $Q^{-1}BA^{-1}B^t$.

The discrete inf-sup condition is extremely restrictive. The problem is that the simplest conforming finite element methods such as Q_1–P_0 (trilinear/bilinear velocity with constant pressure) are not stable in the sense that pressure vectors $p \in M_h$ can be constructed for which the inf-sup constant tends to zero under uniform refinement. This type of instability can be difficult to detect in practice since the associated discrete systems are non-singular, (so that each of the discrete problems are uniquely solvable), however they become rapidly ill-conditioned as $h \to 0$.

The simplest way of getting such low-order methods to work in practice is to relax the discrete incompressibility condition (2.18). An efficient approach is the following *local* stabilisation method, which is based on controlling the jumps in pressure across element boundaries within an appropriate macroelement subdidvision, $\mathcal{M}$ say, as follows

$$(\nabla \mathbf{u}_h, \nabla \mathbf{v}) - (p_h, \nabla \cdot \mathbf{v}) = (\mathbf{f}_h, \mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{X}_h \tag{2.31}$$

$$(\nabla \cdot \mathbf{u}_h, q) - \beta \sum_{\substack{\mathrm{m}\in\mathcal{M}\\ e\in\Gamma_\mathrm{m}}} h_\mathrm{m} \int_e [\![p_h]\!]_e [\![q]\!]_e \mathrm{d}s = 0 \quad \forall q \in M_h. \tag{2.32}$$

In (2.32), Γ_m is the set of all edges/faces in the *interior* of the m'th macroelement, β is a positive stabilisation parameter (see below) and h_m is a local measure of the macroelement's size, see [30]. Of course, if stability is to be achieved then the number of elements in each macroelement must be sufficiently large—if every macroelement contained just one element there are no internal jump terms (i.e. $\Gamma_\mathrm{m} = \emptyset$), and (2.31)–(2.32) degenerates to the unstabilised formulation. In the motivating paper [30], it is rigorously established that as long as $\mathcal{M}$ is constructed so that each macroelement is topologically equivalent to a reference macroelement having a velocity node on every edge (or every face in three-dimensions), then there exists a minimal parameter value β_0 such that the formulation (2.31)–(2.32) is stable; i.e. there exists a constant γ_s bounded away from zero independently of h such that the following "inf-sup like" condition is satisfied

$$\sup_{\mathbf{w}\in\mathbf{X}_h} \frac{(p, \nabla \cdot \mathbf{w})}{\|\nabla \mathbf{w}\|} \geq \sqrt{2}\gamma_s \|p\| - (\beta \sum_{\substack{\mathrm{m}\in\mathcal{M}\\ e\in\Gamma_\mathrm{m}}} h_\mathrm{m} \int_e [\![p]\!]_e{}^2 \mathrm{d}s)^{\frac{1}{2}} \quad \forall p \in M_h. \tag{2.33}$$

As a result, if $\beta \geq \beta_0$ then an optimal error estimate can be established in the case $\mathbf{g} = \mathbf{0}$ (see [30])

$$\|\nabla(\mathbf{u} - \mathbf{u}_h)\| + \|p - p_h\| \leq Ch \tag{2.34}$$

where C is a constant independent of h and β (it depends only on β_0). Note that the same estimate (2.34) characterises the approximation accuracy of the Q_1–iso–Q_2 method above (with a different constant C).

Using a stabilised formulation of the form (2.31)–(2.32) leads to the following matrix system

$$\begin{pmatrix} A & B^t \\ B & -\beta S \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}. \tag{2.35}$$

where A and B are as defined in (2.26), and S corresponds to the pressure stabilisation term in (2.32). Furthermore we have an explicit representation of the stability constant γ_s in (2.33)

$$\gamma_s^2 = \min_{p \neq 0} \frac{p^t B A^{-1} B^t p + \beta p^t S p}{p^t Q p}, \tag{2.36}$$

which is the analogue of (2.30) in the unstabilised case.

One of the features of (2.31)–(2.32) is that if the discrete incompressibility constraints are added together, then the jump terms sum to zero in each macroelement (a specific example is given below). This is crucially important to the success of the method since it implies that the local incompressibility of the original method is retained after stabilisation (albeit over macroelements). The major potential limitation of this approach is that stability is only guaranteed if the stabilisation parameter β is bigger than the critical value β_0. Fortunately this does not cause any difficulty in practice, since an over-estimate of the critical parameter is easily computed if the extremal eigenvalues of the Schur complement and the stabilisation matrix are known; specifically, it is shown in [43] that $\beta_* \geq \beta_0$ if $\beta_* = \Gamma^2/\Theta^2$ with

$$\Gamma^2 = \max_{p \neq 0} \frac{p^t B A^{-1} B^t p}{p^t Q p}, \tag{2.37}$$

$$\Delta^2 = \max_{p \neq 0} \frac{p^t S p}{p^t Q p}. \tag{2.38}$$

A simple estimate of Γ is well known (see [18]): a Cauchy-Schwarz argument yields

$$\frac{|(\mathrm{div}\mathbf{v}, p)|^2}{\|\mathbf{v}\|_X^2 \|p\|_M^2} \leq \frac{\|\mathrm{div}\mathbf{v}\|^2}{\|\nabla \mathbf{v}\|^2} \leq d, \tag{2.39}$$

so for example in $\mathbb{R}^2$ we have $\sqrt{2} \geq \Gamma$. In practice, this estimate (which holds for all mixed approximations) seems to be pessimistic. In particular, in the case of the Q_1–P_0 approximation, numerical computations on quasi-uniform Cartesian grids of rectangular elements suggest that that $\Gamma \to 1$ from below, as $h \to 0$.

Using a macroelement stabilisation, Δ in (2.38) can be computed locally. To illustrate this, consider the case of a uniform grid of $j \times j$ square Q_1–P_0 elements of side h. If j is even then local stabilisation can be based on 2×2 macroelements, and with an appropriate local numbering, the stabilisation matrix S is block diagonal with identical 4×4 blocks of the following form

$$S_{\mathcal{M}} = h^2 \begin{pmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{pmatrix}. \tag{2.40}$$

As a result the eigenvalues of S are $0, 2h^2, 2h^2, 4h^2$ (each with multiplicity equal to $j^2/4$). Furthermore, since the pressure is piecewise constant the mass matrix Q is diagonal with entries equal to h^2. Hence, $\Delta^2 = 4$ in (2.38), and $\Gamma^2 = 1$ in (2.37) so that a "good" parameter value is easily deduced, namely $\beta = 1/4$. This is important since it allows the possibility of constructing usable software built around Q_1–P_0

for discretising Stokes problems (see section 6). Some numerical results using this software/methodology are described in section 4.

Finally, we note that the discretisation of the Oseen problem (1.11), which arises using the linearised implicit time-stepping methods (see section 1.2) can be done using the Stokes methodology described above, and will give good results if the flow is diffusion-dominated in the sense that boundary layers are properly resolved by the mesh. Some numerical results using stabilised Q_1–P_0 are described in section 5. Generalising the streamline-diffusion approximations of the transport terms (cf. section 2.1) is also possible, although the characterisation of appropriate stabilisation parameters is much more difficult to do automatically in the Oseen case.

3. Solution methods for the discrete convection-diffusion equation. Discretization of the convection-diffusion equation (2.1) using finite differences or finite elements (via (2.5) or (2.8)) leads to a linear system of equations

$$Fu = f \tag{3.1}$$

where u and f are vectors in $\mathbb{R}^n$. F is a nonsymmetric matrix of the form

$$\nu A + N.$$

Here $A = -\Delta_h$, the discrete Laplacian, for the usual finite difference or Galerkin discretizations, or in the case of streamline upwinding, $A = -\Delta_h + A_w$ where A_w corresponds to the stabilizing term of (2.8). N is a skew-symmetric matrix, the discrete convection operator.

We will emphasize splitting methods for (3.1), that is, representations of the coefficient matrix in the form

$$F = Q - R$$

where Q is the nonsingular *splitting matrix.* Such a splitting can be used to produce a stationary iteration

$$u^{(k+1)} = Q^{-1}(Ru^{(k)} + f) \tag{3.2}$$

where $u^{(0)}$ is an arbitrary initial guess, or Q can be used as a preconditioner for (3.1) in combination with Krylov subspace methods. The classical analysis of the stationary method (3.2) proceeds as follows; see [1, 48, 52] for comprehensive presentations of these results and [34, 47] for concise overviews. Let $e^{(k)} = u - u^{(k)}$ denote the error at the kth step of the stationary iteration (3.2). Then

$$e^{(k)} = (Q^{-1}R)^k e^{(0)},$$

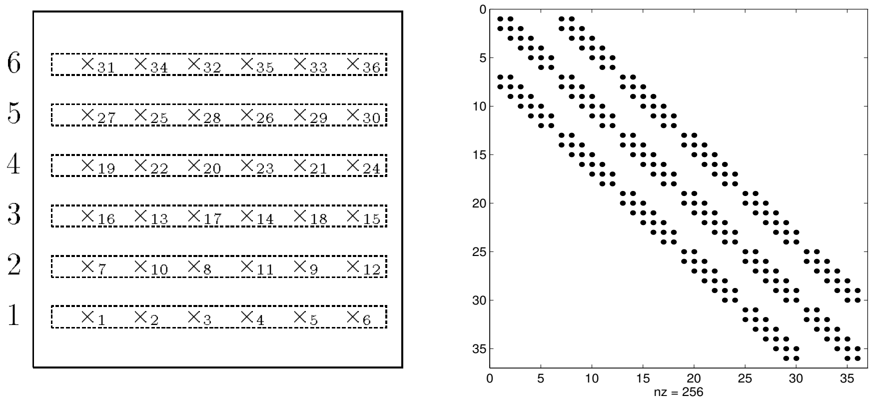
and for any consistent norm $\|\cdot\|$,

$$\|e^{(k)}\| \leq \|(Q^{-1}R)^k\| \; \|e^{(0)}\|. \tag{3.3}$$

Analysis is based on the fact that

$$\lim_{k\to\infty} \|(Q^{-1}R)^k\|^{1/k} = \rho(Q^{-1}R) \tag{3.4}$$

FIG. 1. *Natural horizontal line ordering and nonzero structure of matrix.*



where ρ denotes the spectral radius. The iteration is convergent if and only if $\rho(Q^{-1}R) < 1$, and roughly speaking, the error decreases in magnitude by a factor of $\rho(Q^{-1}R)$ at each step. Consequently, ρ is referred to as the convergence factor. The effectiveness of Krylov subspace methods depends in large part on the existence of a polynomial that takes on the values 1 at the origin and is small on the eigenvalues of $Q^{-1}A = I - Q^{-1}R$ [1, 41]; thus, it is also desirable to make $\rho(Q^{-1}R)$ as small as possible for Krylov subspace methods.

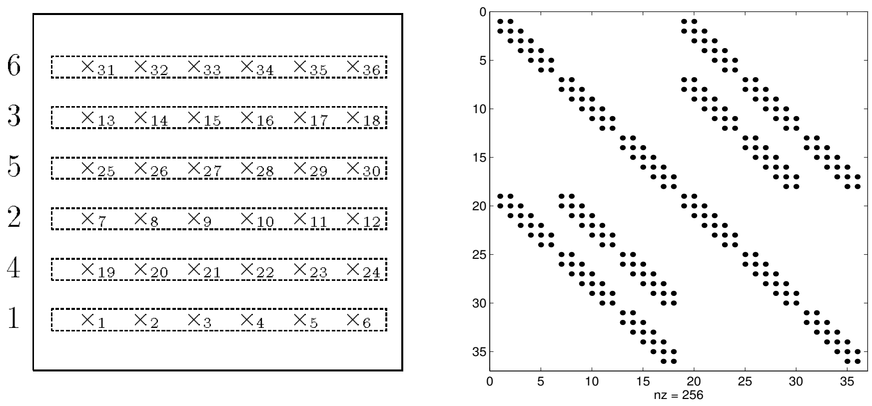
3.1. Analysis of convergence factors. We first consider versions of the classical Jacobi, Gauss-Seidel and successive over-relaxation (SOR) iterative methods and discuss analytic bounds on convergence factors for these methods. Throughout our discussion, we will use the two-dimensional version of (2.1); see [1, 34, 47, 48, 52] for general presentations. Let Ω denote the unit square $(0,1)\times(0,1)$ and assume the discretisation is performed on a uniform grid using finite differences or linear or trilinear finite elements. If the grid is ordered with a natural left-to-right bottom-to-top ordering, then the resulting matrix F has block tridiagonal form in which the block diagonal is a tridiagonal matrix. Figure 1 shows an example for a 6×6 grid ordered by horizontal lines. The nonzero structure of the matrix for a nine-point operator on this grid is shown on the right. This structure would arise from a bilinear finite element discretisation; for finite differences or linear finite elements (with unidirectional triangles), the off-diagonal blocks would be diagonal or bidiagonal, respectively. Figure 2 shows an alternative *line red-black ordering* and the structure of the corresponding matrix. Variants based on vertical orderings are defined analogously.

For any of the line orderings, let $F = D - L - U$ where D denotes the block diagonal of F, $-L$ denotes the lower triangular matrix consisting of entries below the block diagonal D, and $-U$ is the analogous upper triangular matrix. The classical stationary methods are defined by the following splittings:

$$
\begin{array}{lll}
\text{line Jacobi:} & Q = D, & R = D - F; \\
\text{line Gauss-Seidel:} & Q = D - L, & R = U; \\
\text{line SOR:} & Q = \frac{1}{\omega}(D - \omega L), & R = \frac{1}{\omega}\left[(1-\omega)D + \omega U\right].
\end{array}
$$

The matrix F arising from these orderings is block consistently ordered [52]. Consequently, the spectral radii of the line Jacobi and line Gauss-Seidel iteration matrices

FIG. 2. *Horizontal line red-black ordering and nonzero structure of matrix.*



are related by

$$\rho((D-L)^{-1}U) = \rho(D^{-1}C)^2 \tag{3.5}$$

where $C = D - F$. Moreover, if the Jacobi matrix $D^{-1}C$ has real eigenvalues and its spectral radius is less than one, then the spectral radius of the SOR iteration matrix $\mathcal{L}_\omega = (D-\omega L)^{-1}[(1-\omega)D + \omega U]$ is minimized by $\omega^* = \frac{2}{1+\sqrt{1-\rho(D^{-1}C)^2}}$ and

$$\rho(\mathcal{L}_{\omega^*}) = \omega^* - 1. \tag{3.6}$$

Thus the key to the analysis is to bound $\rho(D^{-1}C)$. We treat some specific cases individually.

The constant coefficient problem $\nu = 1$ and $\mathbf{w} = (\sigma, \tau)$ in (2.1) is the starting point for much of the analysis. Any finite difference discretisation produces a five-point operator which can be represented by a "computational molecule"

$$\begin{array}{ccccc} & & -e & & \\ & & | & & \\ -c & — & a & — & -d \\ & & | & & \\ & & -b & & \end{array} \tag{3.7}$$

In this, case, we can give an exact expression for $\rho(D^{-1}C)$. The proof depends on the fact that the block diagonal matrix D can be symmetrized using a diagonal similarity transformation, see [11].

THEOREM 3.1. *If $cd \geq 0$, then the spectral radius of the block Jacobi iteration matrix for the horizontal line ordering is*

$$\frac{2\sqrt{be}\cos(\pi h)}{a - 2\sqrt{cd}\cos(\pi h)}.$$

If $be \geq 0$, then the spectral radius of the block Jacobi iteration matrix for the vertical line ordering is

$$\frac{2\sqrt{cd}\cos(\pi h)}{a - 2\sqrt{be}\cos(\pi h)}.$$

The conditions in this theorem are satisfied if all of the off-diagonal entries b, c, d and e of F are greater than equal to zero. This is the case, for example, if centred finite differences are used to discretise the first derivatives in (2.1) on a fine enough mesh, or if upwind differencing is used [11, 12]. For example, if centred differences are used on a uniform $n \times n$ grid with $h = 1/(n+1)$, then with $\gamma = \sigma h/2$, $\delta = \tau h/2$, Theorem 3.1 is equivalent to the following result.

COROLLARY 3.2. *For centred differences, if $|\gamma| < 1$ then the spectral radius of the block Jacobi iteration matrix for the horizontal line ordering is*

$$\frac{\sqrt{|1-\delta^2|}\cos(\pi h)}{2 - \sqrt{1-\gamma^2}\cos(\pi h)}.$$

If $|\delta| < 1$, then the spectral radius of the block Jacobi iteration matrix for the vertical line ordering is

$$\frac{\sqrt{|1-\gamma^2|}\cos(\pi h)}{2 - \sqrt{1-\delta^2}\cos(\pi h)}.$$

Figure 3 shows some examples of spectral radii of Gauss-Seidel iteration matrices for centred difference discretisations and various parameters. Larger values of γ (respectively δ) correspond to increased convection in the horizontal (vertical) direction. These results indicate that it is advantageous to orient the grid lines in directions orthogonal to the dominant direction of flow, i.e., to perform the Gauss-Seidel sweep in the direction of flow.

For finite difference operators with computational molecules of type (3.7), the grid points and equations can be ordered with a red-black ordering so that every equation centred at a "red" point depends only on "black" unknowns, and every equation centred at a "black" point depends only on "red" unknowns. In matrix notation, this process corresponds to ordering the rows and columns of F so that the problem (3.1) has the form

$$\begin{pmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{pmatrix} \begin{pmatrix} u^{(r)} \\ u^{(b)} \end{pmatrix} = \begin{pmatrix} f^{(r)} \\ f^{(b)} \end{pmatrix}$$

where F_{11} and F_{22} are nonsingular diagonal matrices. Decoupling of the red points $u^{(r)}$ is equivalent to producing a smaller system

$$\hat{F} u^{(b)} = \hat{f}$$

where $\hat{F} = F_{22} - F_{21}F_{11}^{-1}F_{12}$ and $\hat{f} = f^{(b)} - F_{21}F_{11}^{-1}f^{(r)}$.

$\hat{F}$ is also a sparse matrix, and for appropriate orderings of the reduced grid $\hat{F}$ is block consistently ordered. Two examples, a natural *two-line* ordering and a red-black *two-line* ordering, are shown in figure 4. As above, analysis of the Gauss-Seidel and SOR methods uses (3.5) and (3.6). For the orderings of figure 4, we have the following bounds for the block Jacobi iteration matrices. The proof again depends on finding an appropriate symmetrization operator for $\hat{F}$; see [11, 12, 13, 14] for proofs and extensive discussions of other problems and orderings.

FIG. 3. *Spectral radii of line Gauss-Seidel iteration matrices for various parameters.*

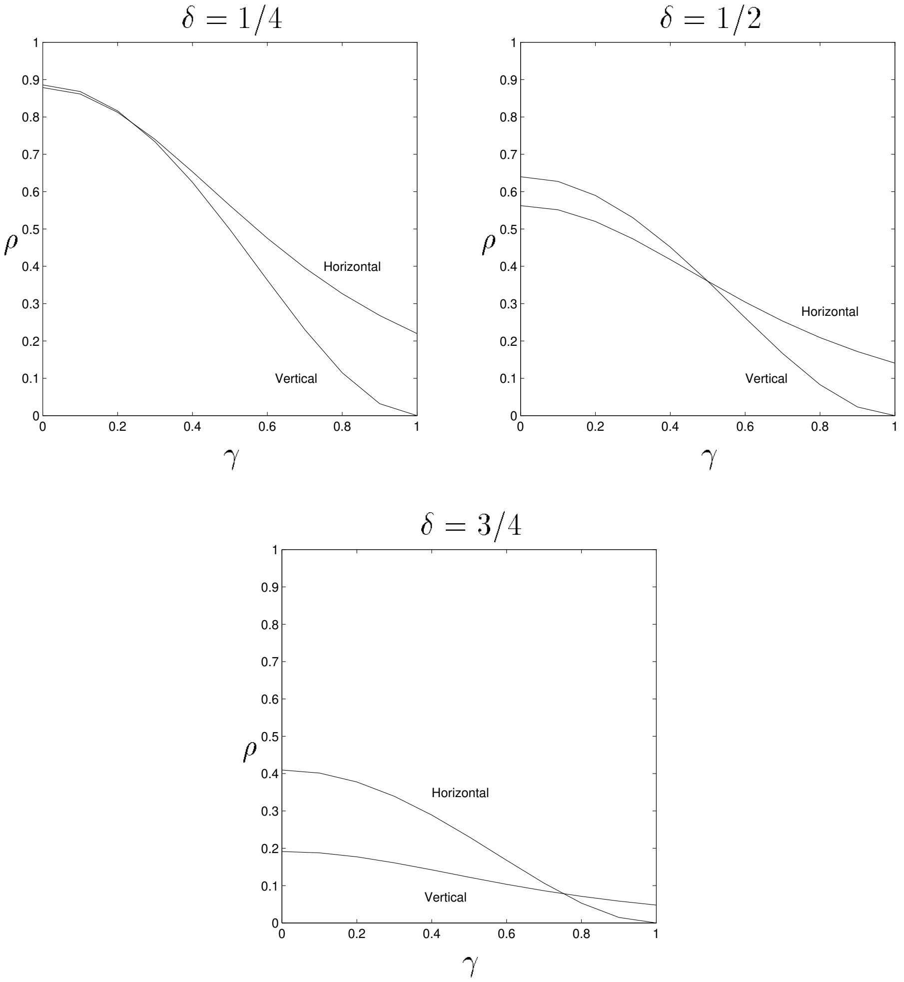
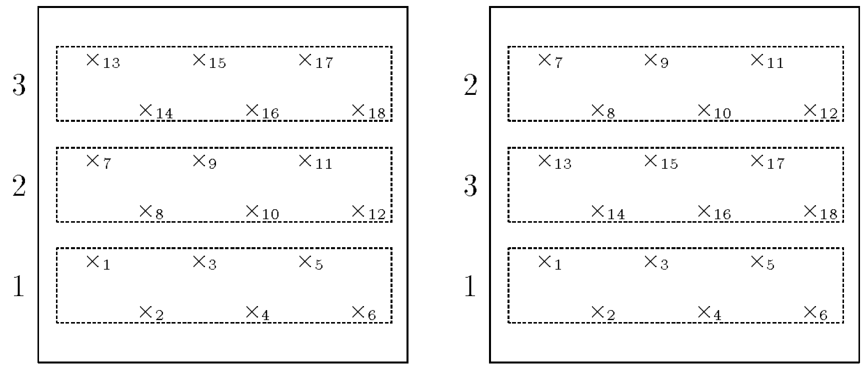


FIG. 4. *Horizontal two-line orderings of the reduced grid.*



THEOREM 3.3. *If $be \geq 0$ and $cd \geq 0$, then the spectral radius of the two-line Jacobi iteration matrix is bounded by*

$$\frac{2\,be\,\cos 2\pi h + 4\sqrt{bcde}\cos \pi h}{\begin{array}{l}[a^2 - 2(\sqrt{cd}+\sqrt{be})^2 - 2cd + \\ \qquad 4\sqrt{bcde}\,(1-\cos\pi h) + 4cd\,(1-\cos^2\pi h)]\end{array}} + o(h^2).$$

COROLLARY 3.4. *For centred differences, if $|\gamma| < 1$ and $|\delta| < 1$, then the spectral radius of the two-line block Jacobi iteration matrix for the reduced system is bounded by*

$$\frac{(1-\delta^2)\cos 2\pi h + 2\sqrt{(1-\gamma^2)(1-\delta^2)}\,\cos\pi h}{\begin{array}{l}[8 - (\sqrt{1-\gamma^2}+\sqrt{1-\delta^2})^2 - (1-\gamma^2) + \\ 2\sqrt{(1-\gamma^2)(1-\delta^2)}\,(1-\cos\pi h) + 2(1-\gamma^2)\,(1-\cos^2\pi h)]\end{array}} + o(h^2).$$

These bounds are typically stronger than those above for the unreduced system. Results for vertical two-line orderings can be established in the same way.

The results above are derived from properties of the matrices D and C of the block Jacobi splitting. An alternative approach due to Parter [36] and Parter and Steuerwalt [38] based more closely on the differential operators reveals asymptotic convergence rates as $h \to 0$. (See also [37].) Let $\mathcal{F}$ denote the differential operator on the left side of (2.1), and assume the discretization matrix F is scaled so that F/h^2 approximates $\mathcal{F}$ with truncation error $o(1)$ at all mesh points of Ω not next to the boundary, and $O(1)$ at points next to $\partial\Omega$. Let $F = Q - R$ be a splitting.

THEOREM 3.5. *Suppose the following conditions hold for all small h:*
1. *$\rho(Q^{-1}R) < 1$.*
2. *$\rho(Q^{-1}R)$ is an eigenvalue of $Q^{-1}R$.*
3. *$\|R\|_2$ is bounded independent of h.*
4. *There is a smooth function q satisfying $q(x,y) \geq q_0 > 0$ on $\bar{\Omega}$, such that*

$$(Ru, v) = (qu, v) + E \tag{3.8}$$

where in (3.8), q refers to the vector of mesh values, and $E = he_1(u,v) + h^2e_2(u,v)$ depends on σ and τ.[1] *Then as $h \to 0$, $\rho(Q^{-1}R) = 1 - \Lambda_0 h^2 + o(h^2)$, where Λ_0 is the smallest eigenvalue of the problem*

$$\mathcal{F}u = \Lambda q u \quad in\ \Omega, \qquad u = 0\ on\ \partial\Omega. \tag{3.9}$$

This result is very easy to apply to the constant coefficient problem. The mesh function q of (3.8) is a constant obtained by inspection as the sum of the entries of the computational molecule that define R. For example, for the Jacobi splittings, $q = 2$ for the one-line ordering of the full system and $q = 3/4$ for the two-line ordering of the reduced system. If the minimal eigenvalue of (3.9) is known, then the asymptotic convergence factor is identified. On the unit square,

$$\Lambda_0 = \frac{1}{q}\left(\frac{\sigma^2}{4} + \frac{\tau^2}{4} + 2\pi^2\right).$$

[1] Here e_1 is a function of first order differences in u and v and e_2 is a function of second order differences; see [38] for a more precise statement.

For the line Jacobi splittings we have discussed here, the convergence factors are

$$1-\left(\frac{\sigma^2}{8}+\frac{\tau^2}{8}+\pi^2\right) \qquad \text{one-line ordering, full system}$$
$$1-\left(\frac{\sigma^2}{3}+\frac{\tau^2}{3}+\frac{8}{3}\pi^2\right) \qquad \text{two-line ordering, reduced system.}$$

The first of these expressions agrees with the asymptotic convergence factor obtained from Corollary 3.2 and the second one is slightly stronger than that obtained from Corollary 3.4. Note that Corollaries 3.2 and 3.4 provide insight into the nonsymptotic regime. Other examples of the use of this methodology are given in [12, 36, 38].

Finally, we note that another popular splitting method for discrete convection-diffusion equations is based on incomplete LU (ILU) factorization of the coefficient matrix. Recall that a nonsingular M-matrix B is one for which $B_{ij} \le 0$ for $i \ne j$ and $B^{-1} \ge 0$ [48]. It is well-known [32] that for any such B there is a unique ILU factorization $Q = LU$ such that L is unit lower triangular, U is upper triangular, $l_{ij} = 0$ and $u_{ij} = 0$ for $(i,j) \notin \mathcal{N}$, and $[Q-B]_{ij} = 0$ for $(i,j) \in \mathcal{N}$, where $\mathcal{N}$ is an index set containing all diagonal indices (i,i). It can be shown [3, 13, 51] that if

$$B = Q_1 - R_1 = Q_2 - R_2,$$

where $Q_1 = L_1U_1$ and $Q_2 = L_2U_2$ are incomplete factorizations such that the set of matrix indices for which $L_1 + U_1$ is permitted to be nonzero is contained in the set of indices for which L_2+U_2 is permitted to be nonzero, then $\rho(Q_2^{-1}R_2) \le \rho(Q_1^{-1}R_1)$. For the examples arising from finite differences that we have considered, both F and $\hat{F}$ are nonsingular M-matrices for a fine enough mesh. Let $Q_1 = Q$ obtained by the ILU(0) factorization (i.e., the index set $\mathcal{N}$ equals the nonzero set of of the coefficient matrix) with error matrix R, and let $Q_2 = D$ from the block Jacobi splitting. It follows that

$$\rho(Q^{-1}R) \le \rho(D^{-1}C).$$

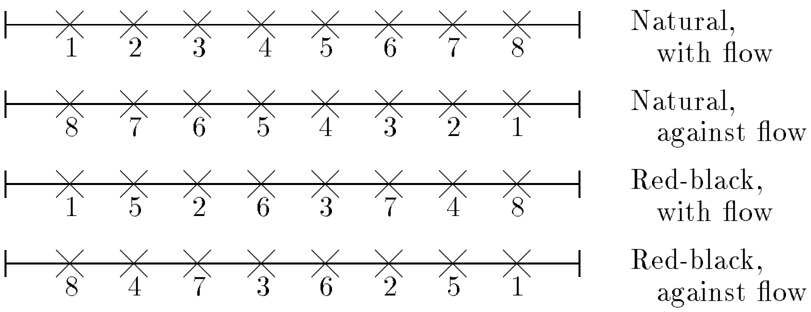
Thus, all the bounds obtained above for the block Jacobi method carry over to the ILU(0) factorization.

3.2. Ordering effects. We now turn to some issues associated with the underlying flow and the effects of ordering of the discrete grid. As noted in the discussion following Corollary 3.2, some of the analysis depends on the orientation of lines in the grid. Once that orientation is fixed, however, there is no dependence on ordering of unknowns. For example, none of results above depend on whether a "natural" or "red-black" ordering is used, and all of them are independent of the sign of the coefficients of the convection terms, which determine the direction of flow. Indeed, for a natural ordering in which relaxation is performed in a direction *opposite* the direction of flow, the bounds on convergence factor are identical.[2]

In practice, the performance of relaxation methods is sensitive to ordering. As might be expected from intuition, it is better to relax in the direction of flow than in the opposite direction, and performance for orderings such as red-black that don't bear a clear relation to flow direction is somewhere in between these extremes. The difference between the analytic results and these performance characteristics stems

[2] Theorem 3.5 has no dependence even on line orientation.

FIG. 5. *Four orderings for a one-dimensional grid, for $\sigma > 0$ and $n = 8$.*



from the difference between (3.3) and (3.4). The expression (3.4) provides insight into asymptotic behaviour as the number of iterations becomes large, but it provides no information about transient behaviour displayed before the limiting value is approached.

Many aspects of this issue can be understood from the one-dimensional version of (2.1)

$$-u'' + \sigma u' = f$$

on the unit interval $(0, 1)$ with Dirichlet boundary conditions and $\sigma > 0$. Let n denote the number of interior mesh points of a uniform grid. Finite difference and linear finite element discretization lead to a linear system (3.1) in which, for a natural ordering, the coefficient matrix F is tridiagonal of order n, with constant values on its three interior bands. Assume that F is normalized to have unit diagonal, so that it can be represented as

$$F = tri\,[-b,\, 1,\, -c\,].$$

In addition, assume $b + c = 1$ (needed for a consistent discretisation) and $b > 0$, $c > 0$ (for a nonoscillatory solution [25]). We say that the discrete problem is *convection-dominated* if b is large, i.e., close to 1. The Gauss-Seidel iteration matrix is $\mathcal{L}_1 = (I - L)^{-1}U$, where L and U are the strict lower triangular and upper triangular parts of F.

Figure 5 shows examples of four different orderings for $n = 8$. There are two natural orderings, together with two red-black orderings induced by the natural orderings. Figure 6 shows a representative example of the behaviour of relaxation for convection-dominated problems that reveals the limitations of the standard analysis. The figure plots $\|e^{(k)}\|_1$, on a logarithmic scale, against the iteration count k, for the Gauss-Seidel method corresponding to the four ordering schemes. Here, $n = 32$ and $b = 7/8$. The initial guess is a normally distributed random vector with mean 0 and variance 1, and the right hand side and solution are identically zero. The spectral radius for each of the orderings is $\rho(\mathcal{L}_1) = .434$. Figure 7 shows the norms $\|\mathcal{L}_1^k\|_1$. In both figures, the highlighted values correspond to $k = n - 1$ and $k = n/2 - 1$ for the natural ordering against the flow and red-black orderings, respectively. It is evident that the norms are closely correlated with the performance of the solution algorithm, and that the spectral radius reveals nothing about the transient behaviour.

The iteration matrices arising from different orderings will be distinguished as follows. For the left-to-right natural ordering, inducing a relaxation sweep oriented

FIG. 6. *l_1-norms of the errors in Gauss-Seidel iteration, for $n = 32$ and $b = 7/8$.*

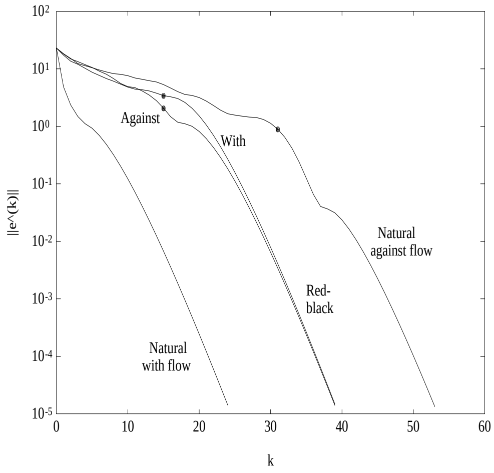
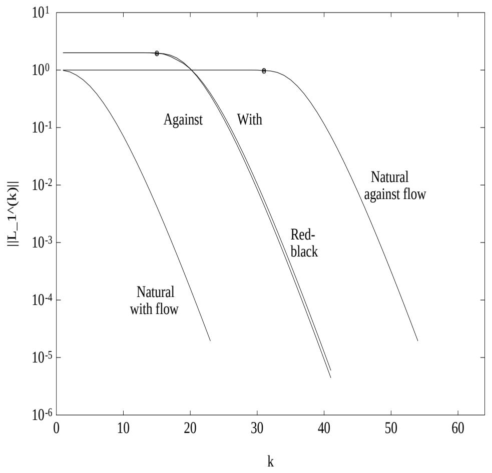


FIG. 7. $\log_{10} \|\mathcal{L}_1^k\|_1$ *for $n = 32$ and $b = 7/8$.*



with the flow, the iteration matrix is $F = (I - L)^{-1}U$, where

$$L = tri\,[\,b,\,0,\,0\,], \quad U = tri\,[\,0,\,0,\,c\,].$$

The red-black ordering induced by this natural ordering gives rise to the coefficient matrix $F = I - L_{RB} - U_{RB}$ where

$$L_{RB} = \begin{pmatrix} 0 & 0 \\ B & 0 \end{pmatrix}, \quad U_{RB} = \begin{pmatrix} 0 & C \\ 0 & 0 \end{pmatrix},$$

and

$$B = tri\,[\,0,\,b,\,c\,], \quad C = tri\,[\,b,\,c,\,0\,],$$

of dimensions $\lceil n/2 \rceil \times \lfloor n/2 \rfloor$ and $\lfloor n/2 \rfloor \times \lceil n/2 \rceil$ respectively. The iteration matrix is given by

$$F_{RB} = (I - L_{RB})^{-1}U_{RB} = (I + L_{RB})U_{RB} = \begin{pmatrix} 0 & C \\ 0 & BC \end{pmatrix}.$$

For sweeps oriented against the flow, rather than reversing the ordering, it is equivalent to use the left-to-right natural ordering and perform an "upper-triangular" sweep, i.e., with the iteration matrix $G = (D - U)^{-1}L$.

We summarize an analysis for the one-dimensional problem below. Proofs and descriptions of additional numerical experiments are given in [9]. There are three results: *lower bounds* on the values of both $\|G^k\|_1$ and $\|F_{RB}^k\|_1$, and *upper bounds* on the values of $\|F^k\|_1$. Essentially the same lower bounds apply in the l_∞-norm, and the upper bounds can be generalized to any l_p-norm.

THEOREM 3.6. *The norm $\|G^k\|_1$ for Gauss-Seidel iteration with sweeps against the flow is bounded below for $k < n$ by*

$$\|G^k\|_1 \geq (1 - c^2)^{k-1}(1 - c^{n-(k-1)}).$$

THEOREM 3.7. *For problems whose order n is divisible by four, the norm $\|F_{RB}^k\|_1$ for Gauss-Seidel iteration associated with the red-black ordering induced by a left-to-right natural ordering is bounded below as follows:*

$$\|F_{RB}^k\|_1 \geq 2 - \psi(k, c) \quad \text{for } k \leq n/2 - 1.$$

where $\psi(k, c)$ is zero for $k < n/4$ and close to zero for $n/4 \leq k < n/2$. (See [9] *for a precise definition.)*

THEOREM 3.8. *The norm $\|F^k\|_1$ for Gauss-Seidel iteration with sweeps that follow the flow is bounded above by*

$$\|F^k\|_1 \leq 1 - b^{n+k-2} \left(\sum_{j=0}^{k-1} \binom{n+k-3+j}{j} c^j \right). \tag{3.10}$$

For $k > (n-3)c/(1-2c)$, the following simpler upper bound holds:

$$\|F^k\|_1 \leq \frac{k \binom{n+2k-3}{k} b^{n-2}(bc)^k}{k(1-2c) - (n-3)c}. \tag{3.11}$$

FIG. 8. *Comparison of* $\|G^k\|_1$ *with lower bounds, for* $n = 32$ *and* $c = 1/8$.

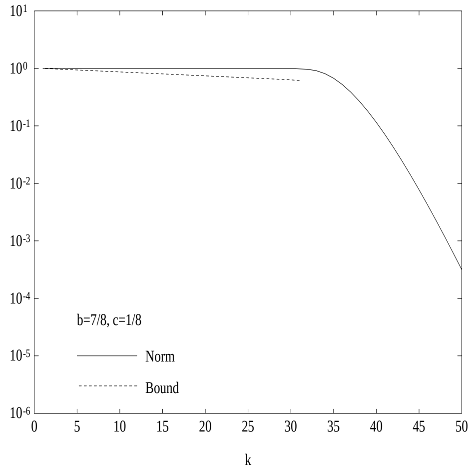


FIG. 9. *Comparison of* $\|F_{RB}^k\|_1$ *with lower bounds, for* $n = 32$ *and* $c = 1/8$.

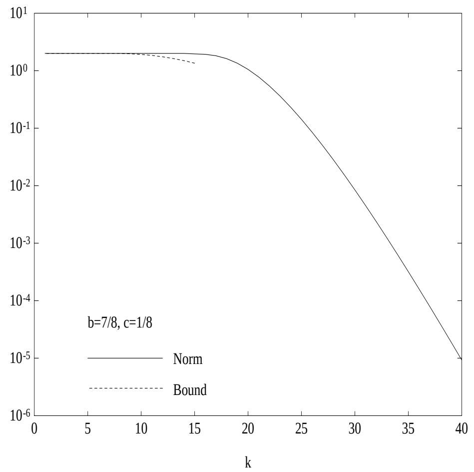
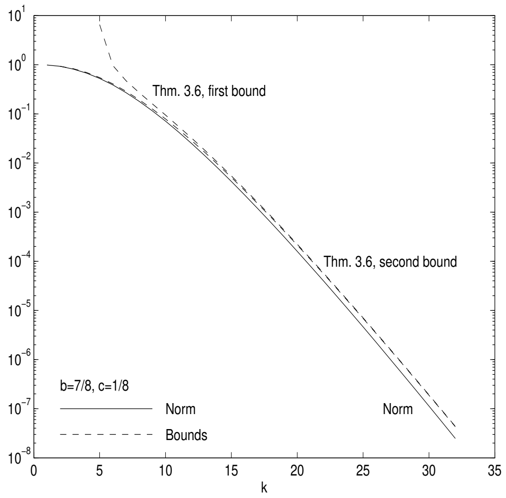


FIG. 10. *Comparison of $\|F^k\|_1$ with upper bounds, for $n = 32$ and $b = 7/8$.*



Figures 8, 9 and 10 plot norms and the bounds from each of these results, for the problem used for Figs. 6 and 7. The results of Theorems 3.6 and 3.7 indicate that if b is near 1 then the norms of the iteration matrices for sweeping against the flow and the red-black ordering are close to one for $n-1$ and $n/2-1$ steps, respectively. Consequently, these orderings incur a latency in which little reduction in the error is obtained. In contrast, Theorem 3.8 shows that the norm of the iteration matrix is small for sweeping with the flow.

It is possible to generalize the upper bounds of Theorem 3.8 to line relaxation methods applied to two-dimensional problems. In this case, the angle between the flow direction and sweep direction plays a role in both performance and bounds; details can be found in [10].

3.3. Discussion. We now discuss some practical issues associated with solving the convection-diffusion equation. We first note some limitations of the analysis cited in sections 3.1 and 3.2, namely, the results apply only to constant coefficient problems and they are limited to finite difference discretisations. The latter restriction is not transparent for Theorem 3.5 but requirement (2) of this result is typically established using the fact that the coefficient matrix is an M-matrix and applying the Perron-Frobenius theory; see [38], p. 1185. Standard finite element discretisations of the convection-diffusion equation do not produce M-matrices, and we know of little analysis for anything other than finite differences. (Cf. [10].)

Despite these limitations, we have found that the analysis above gives a good indication of the behaviour of splitting methods for simple flows or other discretisations. Examples demonstrating this for a semicircular flow are given in [13], which also contains some analysis for variable coefficient problems. As an example of behaviour for other discretisations, we consider two versions of bilinear finite elements applied to the problem (2.1) with constant convection coefficients $v = (-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$ on the unit square, $f = 0$, and Dirichlet boundary conditions $u(x,1) = u(y,0) = 0$, u(1,y)=1, $u(x,0) = 0$ for $x \le 1/2$ and $u(x,0) = 1$ for $x > 1/2$. We discretise on a uniform $n \times n$ element grid using either a pure Galerkin method (2.5) or a streamline upwind Petrov-Galerkin method (2.8). These problems were generated using the MATLAB code described in

	Galerkin			Streamline Upwinding		
ν	Hor.	Vert.	Hor. R/B	Hor.	Vert.	Hor. R/B
1/10	240	262	257	–	–	–
1/25	84	110	98	–	–	–
1/50	29	57	43	31	59	45
1/100	15	46	31	15	44	29
1/200	div.	div.	div.	6	37	22
1/250	div.	div.	div.	9	38	23

TABLE 1
Iterations for bilinear finite elements applied to the convection-diffusion equation.

section 6.

Table 1 shows the iteration counts needed by three variants of line relaxation (horizontal, vertical and horizontal red-black) to solve the discrete problems with $n = 32$, with stopping criterion

$$\|f - Fu^{(k)}\|_2/\|f\|_2 \leq 10^{-6}.$$

Note that the flow direction forms a $-45°$ angle with the horizontal axis. Consequently, the direction of horizontal line relaxation contains a large component in the direction of flow whereas vertical line relaxation is essentially sweeping against the flow. The results show that as the flow becomes stronger (i.e. as the viscosity ν decreases), the differences among the methods are essentially as predicted in section 3.2: sweeping against the flow incurs a latency of approximately n steps and the red-black ordering incurs a latency of approximataly $n/2$ steps. Moreover, the iteration counts decrease dramatically as convection becomes more dominant, as the results of section 3.1 predict.

We also note, however, that the use of these ideas for more complex flows and on large-scale parallel computers lead to some open questions. For example, for the circular flow arising in the driven cavity problem (see section 5), there are portions of the domain where neither a horizontal or vertical line orientation produces a sweep in the direction of flow, and it may be necessary to use more sophisticated strategies to handle such flows. We expect complex three-dimensional flows to add additional difficulties. On parallel architectures, it is known that red-black and multi-color orderings lead to higher parallel efficiencies than natural orderings. However, the latency associated with red-black orderings shows that these reorderings may have limitations that need to be overcome to produce effective solution methods.

4. Solution methods for the discrete Stokes equations. The stability issue associated with mixed approximation of the Stokes problem is of central importance when it comes to finding fast and reliable iterative solution methods. In this section we will develop the theory and present computational results for only one class of method, namely those based on preconditioned Minimum Residual iteration, but see [8] for a comparison of various competitive techniques. The relevant theory for any of the alternative approaches is based on the key result (2.30) which is a direct consequence of the stability required to ensure accuracy properties of the underlying approximation.

In our examples, we concentrate on two particular mixed finite elements: the stable Q_1–*iso*–Q_2 and locally stabilised $Q_1 - P_0$ approximations. Our aim is to illustrate the general structure for stable and stabilised mixed spaces with these convenient and popular choices. Since we wish to concentrate on the stability issue, we consider here the steady state Stokes problem as mentioned in section 2. For consideration of the additional issue arising with time-dependent problems see [4].

4.1. Statement of the problem. As in section 2, we can express the discrete Stokes problem as

$$\mathcal{A}x := \begin{pmatrix} A & B^t \\ B & -\beta S \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix} \tag{4.1}$$

where A is the discrete vector-Laplacian, B is the discrete gradient so that its adjoint B^t is the discrete negative divergence and S is the stabilisation matrix with β being the non-negative stabilisation parameter. The vector u contains the velocity coefficients in terms of the selected basis and p correspondingly for the pressure. Certainly A is symmetric and it will also be positive definite with the usual Dirichlet boundary conditions, B will be full rank except that the vector $p = (1, 1, \ldots, 1)^t$ representing hydrostatic (constant) pressure will be in the null space (unless it is explicitly removed) and S will be symmetric and positive semi-definite ($S = 0$ in the case of an unstabilised approximation). Any body forces are represented in the vector f.

Employing the Sylvester Law of Inertia ([22] pp. 274), the congruence transform

$$\begin{pmatrix} A & B^t \\ B & -\beta S \end{pmatrix} =$$

$$\begin{pmatrix} I & 0 \\ BA^{-1} & I \end{pmatrix} \begin{pmatrix} A & 0 \\ 0 & -\beta S - BA^{-1}B^t \end{pmatrix} \begin{pmatrix} I & A^{-1}B^t \\ 0 & I \end{pmatrix}$$

reveals that $\mathcal{A}$ has n_u positive eigenvalues and n_p negative eigenvalues for a stable or stabilised method. This observation follows directly from the discrete stability condition (2.36).

The indefiniteness of the Stokes system is thus clear. Note that if the mesh size h is reduced and the discrete problem size correspondingly increased, both the number of positive and negative eigenvalues increases: some authors refer to such systems as being highly (or strongly) indefinite. It is the solution of such linear systems which we address in this section.

4.2. The MINRES method. There are two applicable Krylov subspace iterative methods for such symmetric and indefinite systems: SYMMLQ and MINRES, both based on the symmetric Lanczos procedure and both due to Paige and Saunders [35]. Here we concentrate on the MINRES methods since it possesses a minimisation property. We comment that to our knowledge, SYMMLQ has not been tried on discrete Stokes problems.

In the generic context of solving the symmetric and indefinite matrix system

$$\mathcal{A}x = b,$$

MINRES is characterised by the following. The kth iterate x_k lies in the (affine) Krylov subspace

$$x_0 + \mathrm{span}\{r_0, \mathcal{A}r_0, \mathcal{A}^2 r_0, \ldots, \mathcal{A}^{k-1} r_0\}$$

where $r_0 = b - \mathcal{A}x_0$ is the initial residual. Correspondingly for the kth residual vector we have

$$r_k \in r_0 + \mathrm{span}\{\mathcal{A}r_0, \mathcal{A}^2 r_0, \ldots, \mathcal{A}^k r_0\},$$

the defining condition being that $\|r_k\|_2$ is minimum from this space. Thus if Π_k is the set of all real polynomials of degree less than or equal to k then $r_k = p(\mathcal{A})r_0$, with $p(0) = 1$ and p being optimal in the above sense. Employing a spectral (eigenvector) expansion

$$r_0 = \sum \alpha_i v_i \quad , \quad \mathcal{A}v_i = \lambda_i v_i$$

we have

$$r_k = p(\mathcal{A})\sum \alpha_i v_i = \sum \alpha_i p(\lambda_i) v_i$$

so that

$$\begin{aligned}
\|r_k\|_2 &= \min_{p\in\Pi_k, p(0)=1} \|\sum \alpha_i p(\lambda_i) v_i\|_2 \\
&= \min_{p\in\Pi_k, p(0)=1} \left(\sum \alpha_i^2 p(\lambda_i)^2 v_i^t v_i\right)^{\frac{1}{2}} \\
&\leq \min_{p\in\Pi_k, p(0)=1} \max_i |p(\lambda_i)| \; \left(\sum \alpha_i^2 v_i^t v_i\right)^{\frac{1}{2}}
\end{aligned}$$

or

$$\frac{\|r_k\|_2}{\|r_0\|_2} \leq \min_{p\in\Pi_k, p(0)=1} \max_{\lambda\in\Lambda(\mathcal{A})} |p(\lambda)|$$

where $\Lambda(\mathcal{A})$ denotes the eigenvalue spectrum. Note that the orthogonality of the eigenvectors which is a consequence of the symmetry of $\mathcal{A}$ is important here. Also if one were interested in positive definite symmetric matrices $\mathcal{A}$, this convergence estimate would be the same as that for the 'classical' Conjugate Gradient method (which requires fewer operations per iteration) except that $\|r_k\|_2$ would be replaced by $\sqrt{r_k^t \mathcal{A}^{-1}\, r_k} = \sqrt{(x - x_k)^t \mathcal{A}\,(x - x_k)} \stackrel{\mathrm{def}}{=} \|x - x_k\|_{\mathcal{A}}$.

In order to achieve rapid convergence, preconditioning will be as important here as in the symmetric and positive definite case. Also it is desirable to ensure that any preconditioner does not destroy the underlying symmetry of the original problem else more general non-symmetric iterative methods such as GMRES ([42] or see the paper by Van der Vorst in this volume) would have to be employed. Such methods are generally less efficient than their symmetric counterparts (see for example [16]). In order to preserve symmetry in the preconditioned system we employ a symmetric and positive definite preconditioner $\mathcal{M}$ which for theoretical purposes only we factor as $\mathcal{M} = \mathcal{M}^{\frac{1}{2}}\mathcal{M}^{\frac{1}{2}}$. (A Cholesky factorisation could equally be used). We are then interested in applying MINRES to the preconditioned system

$$\mathcal{M}^{-\frac{1}{2}}\mathcal{A}\mathcal{M}^{-\frac{1}{2}}(\mathcal{M}^{\frac{1}{2}}x) = \mathcal{M}^{-\frac{1}{2}}b$$

or

$$\widetilde{\mathcal{A}}\,\widetilde{x} = \widetilde{b}$$

say. Now the corresponding preconditioned residual is

$$\widetilde{r} = \widetilde{b} - \widetilde{\mathcal{A}}\,\widetilde{x} = \mathcal{M}^{-\frac{1}{2}}(b - \mathcal{A}x) = \mathcal{M}^{-\frac{1}{2}}r$$

so that

$$\|\widetilde{r_k}\|_2^2 = \widetilde{r_k}^t\widetilde{r_k} = r_k^t\mathcal{M}^{-1}r_k = \|r_k\|_{\mathcal{M}^{-1}}^2.$$

The preconditioned MINRES convergence estimate therefore becomes

$$\frac{\|r_k\|_{\mathcal{M}^{-1}}^2}{\|r_0\|_{\mathcal{M}^{-1}}^2} \leq \min_{p\in\Pi_k, p(0)=1} \max_{\lambda\in\Lambda(\mathcal{M}^{-1}\mathcal{A})} |p(\lambda)| := \widehat{\rho}_k. \tag{4.2}$$

Note that the use of a positive definite preconditioner was necessary as $\|\cdot\|_{\mathcal{M}^{-1}}$ does not define a norm for indefinite $\mathcal{M}$. A consequence is that preconditioning can not alter the inertia of the original system since $\mathcal{M}^{-\frac{1}{2}}\mathcal{A}\mathcal{M}^{-\frac{1}{2}}$ is a congruence transform and the Sylvester Law of Inertia applies. That is, any symmetric and indefinite matrix preconditioned by a positive definite matrix is necessarily left with the same number of positive and negative eigenvalues. The role of preconditioning in this case is therefore to cluster both the positive and the negative eigenvalues so that the polynomial approximation error $\widehat{\rho}_k$ in (4.2) is small for low number of iterations, k.

A second point is that (unlike in the case of the conjugate gradient method) reduction of the residual in the preconditioned MINRES algorithm is in a norm which is dependent on the preconditioner. Thus one must be careful not to select a preconditioner which simply distorts this norm. We will return to this point later.

At each MINRES iteration we will require the solution of a system of equations with the preconditioner as coefficient matrix. Thus from the point of view of practicality, this must be readily achieved.

4.3. Preconditioning. The convergence estimate (4.2) shows that convergence depends on the eigenvalues of the preconditioned system: our goal now is to estimate these eigenvalues. In particular for a partial differential equation problem such as the Stokes problem, we are interested in the rate of MINRES convergence for large discrete problems, i.e. for discretisations on fine meshes which lead to very large dimensional matrix systems. It is therefore appropriate to consider how the rate of convergence depends on the representative mesh-size, h, as $h \to 0$. The best case will be if the number of iteration required to achieve convergence to a given tolerance does not depend on h.

Since this preserves the underlying block structure of the coefficient matrix, we are interested in block diagonal preconditioning matrices of the form

$$\begin{pmatrix} P & 0 \\ 0 & M \end{pmatrix} \tag{4.3}$$

where both P and M are symmetric and positive definite. The eigenvalues we wish to estimate are therefore the eigenvalues λ of

$$\begin{pmatrix} A & B^t \\ B & -\beta S \end{pmatrix}\begin{pmatrix} u \\ p \end{pmatrix} = \lambda\begin{pmatrix} P & 0 \\ 0 & M \end{pmatrix}\begin{pmatrix} u \\ p \end{pmatrix}.$$

We readily see that if $P = A$ then $\lambda = 1$ is an eigenvalue of multiplicity $n_u - n_p$ corresponding to any eigenvector $[u, 0]^t$ with $Bu = 0$. (The multiplicity comes simply from the size of the right null space of the rectangular matrix B). In the stable case ($S = 0$), if also $M = BA^{-1}B^t$, the remaining eigenvalues satisfy

$$(1-\lambda)Au = -B^t p \qquad \text{and} \qquad Bu = \lambda BA^{-1}B^t p$$

or by eliminating u,

$$(\lambda^2 - \lambda + 1)BA^{-1}B^t p = 0.$$

Thus since the assumed inf-sup stability in this case ensures that $BA^{-1}B^t$ is positive definite, we deduce that $\lambda = 1/2 \pm \sqrt{5}/2$ are the remaining eigenvalues each with multiplicity n_p. This is an ideal situation from the point of view of convergence of MINRES: since the preconditioned matrix $\mathcal{M}^{-\frac{1}{2}}\mathcal{A}\mathcal{M}^{-\frac{1}{2}}$ has only three distinct eigenvalues the convergence bound (4.2) will be zero for $k = 3$ as there is a cubic polynomial with these three roots. That is, MINRES will terminate with the exact solution after three iterations regardless of the size of the discrete problem.

Unfortunately use of the Schur complement $BA^{-1}B^t$ in the preconditioner is not desirable since it is in general a dense matrix which is not easy to construct let alone to invert (or rather solve a system) at each MINRES iteration. But this is where the discrete inf-sup stability condition (2.30) and (2.37) provides the key: the pressure mass matrix Q is spectrally equivalent to $BA^{-1}B^t$ and so we lose little by selecting $M = Q$. The analysis with this choice is similar to the above: we have

$$\begin{aligned} Au + B^t p &= \lambda Au \\ Bu &= \lambda Qp. \end{aligned}$$

The case $\lambda = 1$ arises with the same eigenvectors (and thus multiplicity) as above, and for $\lambda \neq 1$ eliminating u using the first of these equations gives

$$BA^{-1}B^t p = \lambda(\lambda - 1)Qp.$$

Thus for each eigenvalue μ of $Q^{-\frac{1}{2}}BA^{-1}B^tQ^{-\frac{1}{2}}$ there are a pair of eigenvalues

$$\lambda = \frac{1}{2} - \frac{1}{2}\sqrt{1+4\mu} < 0 \quad \text{and} \quad \lambda = \frac{1}{2} + \frac{1}{2}\sqrt{1+4\mu} > 0$$

of the original problem. Now since discrete inf-sup stability and boundedness imply $\gamma^2 \leq \mu \leq \Gamma^2$, we see that

$$\lambda \in \left[\frac{1-\sqrt{1+4\Gamma^2}}{2}, \frac{1-\sqrt{1+4\gamma^2}}{2}\right] \cup \{1\} \cup \left[\frac{1+\sqrt{1+4\gamma^2}}{2}, \frac{1+\sqrt{1+4\Gamma^2}}{2}\right]$$

for every eigenvalue. That is, the multiple eigenvalue $\lambda = 1$ is retained and the remaining eigenvalues are pairwise symmetric about $\frac{1}{2}$ and lie in small intervals which are uniformly bounded and uniformly bounded away from the origin. In this situation the convergence of MINRES will not take only three iterations, but nevertheless it will be fast and (crucially) will be independent of the size of the discrete problem. For an unstable approximation we have $\gamma = 0$ (or $\gamma \to 0$ under mesh refinement), so the negative eigenvalues would not be bounded away from the origin and poor convergence results.

Before proceding to more general theory, we motivate other approximations which will preserve the effective form of this 'ideal' preconditioner but which lead to a more practical overall preconditioner.

It is apparent that preconditioning with $\mathcal{M}$ as above requires at each MINRES iteration the solution of two systems of equations of size n_u and n_p and with coefficient matrices P and M respectively. The 'ideal' choice $P = A$ thus requires an exact solution of a Poisson equation for each of the velocity components since A is the vector Laplacian coming from approximation of the viscous terms. Conveniently there has been much analysis of preconditioners for the Laplacian (see for example the papers by Xu and Chan in this volume).

We do not need to use an inner preconditioned conjugate gradient iteration to effect an exact solution, but are in a position to simply take P to be a domain decomposition or multilevel preconditioner for example. That is we simply let P be a preconditioner for the Laplacian. By applying a suitable scaling to P if necessary we will assume that

$$\alpha \leq \frac{u^t A u}{u^t P u} \leq 1 \qquad \text{for all } u. \tag{4.4}$$

If we use a powerful preconditioner such as a multigrid cycle, then α will be near 1 independently of the discrete problem size (usually expressed in terms of inverse powers of the mesh size parameter, h) and we might expect that only a few more MINRES iterations will be required than if we made the more expensive choice $P = A$. If we use a weaker preconditioner such as diagonal scaling for which $\alpha = O(h^2)$ then more MINRES iterations will be needed for convergence.

It is a much simpler matter to approximate the ideal choice of M further by approximating the pressure mass matrix Q without significantly affecting the convergence of MINRES. The simplest choice $M = \mathrm{diag}(Q)$ is proved to be a good approximation to Q in [49]. Specifically we will assume

$$\theta^2 \leq \frac{p^t Q p}{p^t M p} \leq \Theta^2 \qquad \text{for all } p. \tag{4.5}$$

Using a continuous P_1 pressure approximation for example, replacing the mass matrix Q by its diagonal is very convenient computationally, and furthermore (4.5) is satisfied in this case with $\theta = 1/\sqrt{2}$ and $\Theta = \sqrt{2}$.

4.4. Eigenvalue bounds. Our analysis proceeds with the assumptions (4.4), (4.5) and the further assumption of boundedness of the stabilisation matrix S:

$$\frac{p^t S p}{p^t Q p} \leq \Delta^2 \qquad \text{for all } p. \tag{4.6}$$

Using the locally stabilised $Q_1 - P_0$ mixed approximation described in section 2, we choose $M = Q$ in (4.5) since Q is a diagonal matrix and in this case we know that $\Delta = 2$ on a uniform mesh.

For the eigenvalue analysis it is convenient to consider the symmetrically preconditioned system:

$$\mathcal{M}^{-\frac{1}{2}}\mathcal{A}\mathcal{M}^{-\frac{1}{2}} = \begin{pmatrix} P^{-\frac{1}{2}}AP^{-\frac{1}{2}} & P^{-\frac{1}{2}}B^tM^{-\frac{1}{2}} \\ M^{-\frac{1}{2}}BP^{-\frac{1}{2}} & -\beta M^{-\frac{1}{2}}SM^{-\frac{1}{2}} \end{pmatrix}$$

$$= \begin{pmatrix} \widetilde{A} & \widetilde{B}^t \\ \widetilde{B} & -\beta\widetilde{S} \end{pmatrix} = \widetilde{\mathcal{A}}. \tag{4.7}$$

In the following, we denote by $\sigma_{\max}$ the largest singular value of $\widetilde{B}$ (i.e. the largest eigenvalue of $\widetilde{B}\widetilde{B}^t$).

Lemma 4.1. *All negative eigenvalues λ of $\widetilde{\mathcal{A}}$ satisfy*

$$\frac{1}{2}\left(\alpha - \beta\Delta^2\Theta^2 - \sqrt{(\alpha+\beta\Delta^2\Theta^2)^2 + 4\sigma_{\max}^2}\right) \leq \lambda \tag{4.8}$$

and

$$\lambda \leq \frac{1}{2}\left(\alpha - \sqrt{\alpha^2 + 4\gamma^2\theta^2\alpha}\right) \tag{4.9}$$

and all positive eigenvalues λ of $\widetilde{\mathcal{A}}$ satisfy

$$\alpha \leq \lambda, \tag{4.10}$$

and

$$\lambda \leq \frac{1}{2}\left(1 + \sqrt{1 + 4\sigma_{\max}^2}\right). \tag{4.11}$$

Proof. If λ is an eigenvalue of $\widetilde{\mathcal{A}}$ then there are vectors u, p not both zero satisfying

$$\widetilde{A}u + \widetilde{B}^tp = \lambda u \tag{4.12}$$

$$\widetilde{B}u - \beta\widetilde{S}p = \lambda p. \tag{4.13}$$

If $\lambda > 0$ then $u \neq 0$ since otherwise (4.13) implies $p = 0$ as $\widetilde{S}$ is positive semi-definite. If $\lambda < 0$ then $p \neq 0$ since otherwise (4.12) implies $u = 0$ as $\widetilde{A}$ is positive definite.

Taking the scalar product of (4.12) with u and the scalar product of (4.13) with p and subtracting gives

$$u^t\widetilde{A}u + \beta p^t\widetilde{S}p = \lambda\, u^tu - \lambda\, p^tp$$

which using (4.4) and the positive semi-definiteness of $\beta\widetilde{S}$ gives

$$(\alpha - \lambda)u^tu \leq -\lambda p^tp$$

leading to (4.10) for positive λ since $u \neq 0$ in this case.

Further for $\lambda > 0$, substituting for p from (4.13) into the scalar product of u with (4.12) gives

$$u^t\widetilde{A}u + \frac{1}{\lambda}u^t\widetilde{B}^t\left(I + \frac{\beta}{\lambda}\widetilde{S}\right)^{-1}\widetilde{B}u = \lambda u^tu$$

where the stated matrix inverse certainly exists because $\beta, \lambda > 0$ and $\widetilde{S}$ is positive semi-definite. Moreover the maximum eigenvalue of $(I + \frac{\beta}{\lambda}\widetilde{S})^{-1}$ is 1 thus

$$\lambda u^t \widetilde{A} u + u^t \widetilde{B}^t \widetilde{B} u \geq \lambda^2 u^t u$$

from which follows

$$0 \geq \lambda^2 - \lambda - \sigma_{\max}^2.$$

This gives (4.11).

For $\lambda < 0$, $\widetilde{A} - \lambda I$ is invertible, so we can take the scalar product of (4.13) with p and substitute for u from (4.12) to obtain

$$p^t \widetilde{B}(\widetilde{A} - \lambda I)^{-1} \widetilde{B}^t p + \beta p^t \widetilde{S} p = -\lambda p^t p. \tag{4.14}$$

Considering (4.14), if $\lambda < 0$ is an eigenvalue of $\widetilde{\mathcal{A}}$ then

$$p^t \widetilde{B} \widetilde{A}^{-\frac{1}{2}} (I - \lambda \widetilde{A}^{-1})^{-1} \widetilde{A}^{-\frac{1}{2}} \widetilde{B}^t p + \beta p^t \widetilde{S} p = -\lambda p^t p$$

where $p \neq 0$. Because the eigenvalues of $(I - \lambda \widetilde{A}^{-1})^{-1}$ are

$$(1 - \lambda/\alpha)^{-1} \leq \ldots \leq (1 - \lambda)^{-1},$$

we have

$$(1 - \lambda/\alpha)^{-1} p^t \widetilde{B} \widetilde{A}^{-1} \widetilde{B}^t p + \beta p^t \widetilde{S} p \leq -\lambda p^t p,$$

and since $0 \leq (1 - \lambda/\alpha)^{-1} \leq 1$ there follows

$$(1 - \lambda/\alpha)^{-1} \Big(p^t \widetilde{B} \widetilde{A}^{-1} \widetilde{B}^t p + \beta p^t \widetilde{S} p \Big) \leq -\lambda p^t p.$$

Using (4.7) to express this in terms of the blocks of the original unpreconditioned Stokes matrix (4.1) this is

$$(1 - \lambda/\alpha)^{-1} p^t M^{-\frac{1}{2}} (B A^{-1} B^t + \beta S) M^{-\frac{1}{2}} p \leq -\lambda p^t p.$$

Now using the stability property (2.36) this implies

$$\gamma^2 (1 - \lambda/\alpha)^{-1} p^t M^{-\frac{1}{2}} Q M^{-\frac{1}{2}} p \leq -\lambda p^t p$$

which by employing (4.5) further implies

$$\gamma^2 \theta^2 (1 - \lambda/\alpha)^{-1} p^t p \leq -\lambda p^t p.$$

Since $p \neq 0$ this gives

$$0 \leq \lambda^2 - \alpha\lambda - \alpha\gamma^2\theta^2$$

from which (4.9) easily follows.

To derive (4.8) we use (4.6) and (4.5) in (4.14) to obtain

$$(\alpha - \lambda)^{-1} \sigma_{\max}^2 + \beta \Delta^2 \Theta^2 \geq -\lambda$$

or

$$0 \geq \lambda^2 + (\beta \Delta^2 \Theta^2 - \alpha)\lambda - \sigma_{\max}^2 - \beta \Delta^2 \Theta^2 \alpha$$

which yields the result. □

It is convenient to remove $\sigma_{\max}$ from these bounds since estimates for this quantity are not readily available.

Lemma 4.2.

$$\sigma_{\max} \leq \Gamma\Theta \tag{4.15}$$

Proof. For all p we have

$$\begin{aligned} p^t \widetilde{B}\widetilde{B}^t p &= p^t M^{-\frac{1}{2}} B P^{-1} B^t M^{-\frac{1}{2}} p \\ &\leq p^t M^{-\frac{1}{2}} B A^{-1} B^t M^{-\frac{1}{2}} p \end{aligned}$$

using (4.4). So given that (2.37) holds in the stable or stabilised case we have

$$\begin{aligned} p^t \widetilde{B}\widetilde{B}^t p &\leq \Gamma^2 \; p^t M^{-\frac{1}{2}} Q M^{-\frac{1}{2}} p \\ &\leq \Gamma^2 \; \Theta^2 \; p^t p \end{aligned}$$

where we have further used (4.5). We have thus proved

$$\sigma_{\max}^2 \leq \Gamma^2 \; \Theta^2$$

and hence (4.15). □

Employing Lemma 4.2, the bounds (4.8) and (4.11) become

$$\frac{1}{2}\left(\alpha - \beta\Delta^2\Theta^2 - \sqrt{(\alpha + \beta\Delta^2\Theta^2)^2 + 4\Gamma^2 \; \Theta^2}\right) \leq \lambda \tag{4.16}$$

and

$$\lambda \leq \frac{1}{2}\left(1 + \sqrt{1 + 4\Gamma^2 \; \Theta^2}\right). \tag{4.17}$$

Regarding (4.16),(4.9),(4.10) and (4.17) as the best bounds which we can estimate, we are now in a position to find an upper bound on the convergence rate of the preconditioned MINRES algorithm by considering the approximation problem in (4.2). Before doing so let us just point out the dependencies of the relevant quantities α, γ, Γ, θ, Θ and Δ as well as the stabilisation parameter β (which arises only in a stabilised formulation):

- α: depends on how well the preconditioning block P approximates the discrete Laplacian A
- γ: stability constant—bounded above zero independently of the mesh
- Γ: boundedness constant: $\Gamma \leq \sqrt{d}$ for any domain $\Omega \subset \mathbb{R}^d$ (see (2.39))
- θ, Θ: positive constants independent of problem size even for the simple choice $M = \mathrm{diag}(Q)$. For such a choice of the preconditioning block M, these constants are tabulated in ([49]) for many different finite elements types
- Δ: upper bound on the stabilisation matrix S—an $O(1)$ constant.
- β: positive stabilisation parameter optimally chosen to be just large enough to achieve stability (see [43]).

Any of these parameters may depend on the geometry of the domain and/or the computational grid, *BUT it is only α which can depend explicity on the size of the discrete problem.* That is, the only way that mesh-size dependence arises in the eigenvalues bounds for the preconditioned Stokes coefficient matrix $\widetilde{\mathcal{A}}$ is through a dependence of α on the representative mesh-size, h. Therefore, provided that some simple approximation of the pressure mass matrix is used so that (4.5) is satisfied and provided a suitable stable or stabilised formulation is employed so that (2.30) or (2.36) and (4.6) hold then the convergence of the preconditioned MINRES algorithm will be essentially determined by the quality of the Laplacian preconditioner, P. Let us illustrate with a few examples.

Example 1: $P = \mathrm{diag}(A)$ and M is any suitable choice (such as $\mathrm{diag}(Q)$) which satisfies (4.5).

For this case we have $\alpha = ch^2 + \mathrm{O}(h^4)$ for some constant c independent of h (see for example [2], pp. 240). By considering the leading asymptotic term for small h it is apparent that the eigenvalue bounds (4.16),(4.9),(4.10) and (4.17) define a pair of eigenvalue inclusion intervals of the form

$$\Lambda(\mathcal{M}^{-\frac{1}{2}}\mathcal{A}\mathcal{M}^{-\frac{1}{2}}) \subset [-a, -bh] \,\cup\, [ch^2, d]. \tag{4.18}$$

The constants a, b, c and d are defined in terms of γ, Γ, θ, Θ, Δ and β by the above formulae *but* they do not depend on h. (c is exactly as above because of the simple form of (4.10)). We demonstrate the asymptotic manipulation for the least obvious bound (4.9):

$$\begin{aligned}
\lambda &\leq \frac{1}{2}\left(ch^2 + \mathrm{O}(h^4) - \sqrt{(ch^2 + \mathrm{O}(h^4))^2 + 4\gamma^2\theta^2(ch^2 + \mathrm{O}(h^4))}\right) \\
&= \frac{1}{2}\left(ch^2 + \mathrm{O}(h^4) - 2c^{\frac{1}{2}}\theta\gamma h \left(1 + \mathrm{O}(h^4)\right)^{\frac{1}{2}}\right) \\
&= -c^{\frac{1}{2}}\gamma\theta h + \frac{1}{2}ch^2 + \mathrm{O}(h^4).
\end{aligned}$$

The important point to note here is that as $h \to 0$ the negative eigenvalues approach the origin at only half the rate at which the positive eigenvalues can approach from above.

h	$\lambda^-_{\max}$	$\lambda^-_{\min}$	$\lambda^+_{\min}$	$\lambda^+_{\max}$
1/8	−0.7547	−0.1556e0	0.2747e0	2.0640
1/16	−0.7701	−0.9500e-1	0.7444e-1	2.1347
1/32	−0.7740	−0.5253e-1	0.1902e-1	2.1531
1/64	−0.7749	−0.2770e-1	0.4783e-2	2.1577
1/128	−0.7752	−0.1427e-1	0.1198e-2	2.1589

TABLE 2
Extreme eigenvalues: $Q_1 - P_0$ element with diagonal preconditioning

In table 2 we show the results of eigenvalue computations on the diagonally preconditioned Stokes coefficient matrix as above for a driven cavity flow problem (see section 6 for associated software). We show the extreme eigenvalues of $\mathcal{M}^{-\frac{1}{2}}\mathcal{A}\mathcal{M}^{-\frac{1}{2}}$ for a sequence of regular grids refined by bisection and using the locally stabilised $Q_1 - P_0$ element with the 'optimal' stabilisation parameter value $\beta = 0.058$ (see [43]). The driven cavity problem was solved on only half of the flow domain by using the natural symmetry about the centreline. The most positive and most negative eigenvalues ($\lambda^+_{\max}$ and $\lambda^-_{\max}$ respectively) clearly approach constant values as h is reduced, the negative eigenvalue nearest to the origin ($\lambda^-_{\min}$) is approximately halved and the smallest positive eigenvalue ($\lambda^+_{\min}$) reduces by approximately a quarter as h is halved:

h	$\mu_{\min}$	$\mu_{\max}$	$\lambda_{\min}(P^{-1}A)$	$\lambda_{\max}(P^{-1}A)$
1/8	0.1686	0.9340	0.8519	1.0000
1/16	0.1655	0.9862	0.8220	1.0000
1/32	0.1642	0.9967	0.8090	1.0000

TABLE 3

Extreme eigenvalues of $Q^{-1}BA^{-1}B^t$ and $P^{-1}A$: Q_1–iso–Q_2 element with multigrid preconditioning

these results therefore show that (4.18) is descriptive and is not just providing crude bounds.

We note that example 1 is illustrative of the generic situation: if P is chosen such that $\alpha = \mathrm{O}(h^r)$ and M is an appropriate approximation of the pressure mass matrix then

$$\Lambda(\mathcal{M}^{-\frac{1}{2}}\mathcal{A}\mathcal{M}^{-\frac{1}{2}}) \subset [-a, -bh^{r/2}] \cup [ch^r, d]. \tag{4.19}$$

That is the negative eigenvalues always approach the origin at half of the rate of the positive eigenvalues. The analysis given here therefore applies to a wide range of Laplacian preconditioners including, for example, the modified incomplete cholesky factorisation ([32], [26]) for which $r = 1$.

Example 2: P is a multigrid cycle for A (see for example the paper by Xu in this volume) and $M = Q$ (or some approximation).

This is actually the easiest situation from the view point of the analysis as α is bounded away from zero independently of h. In table 3 we give the computed extremal eigenvalues of $P^{-1}A$ for our test problem employing the stable Q_1–*iso*–Q_2 element on a sequence of refined meshes. The preconditioner P represents a single multigrid V-cycle with an 'optimally' damped Jacobi smoother. It is apparent that $\alpha \approx 0.8$ in this situation. Also tabulated in 3 are the extreme eigenvalues, $\mu_{\min}$ and $\mu_{\max}$ of $Q^{-1}BA^{-1}B^t$: these show that $\gamma \approx 0.16$ and $\Gamma \approx 1$ for this element. It follows that the bounds (4.16),(4.9),(4.10) and (4.17) are all independent of h.

An interesting point arises with the use of spectrally equivalent preconditioners such as in this example, namely convergence of MINRES occurs in norm which is naturally associated with the problem, see [44].

4.5. The rate of convergence of MINRES. In the case of a spectrally equivalent Laplacian preconditioner P such as a multigrid cycle, we may simply note that since all of the eigenvalues are bounded away from infinity and away from the origin independently of the mesh-size h, then $\widehat{\rho}_k$ in (4.2) is also independent of h. The convergence of MINRES in this case should therefore be independent of problem size. This is clearly displayed in table 4 where we present some preconditioned MINRES iteration counts.

The problem is again the leaky lid driven cavity but solved on only half of the domain by using the natural symmetry. For these results the stable Q_1–*iso*–Q_2 element was used and the convergence criterion was a reduction by 10^{-6} in the $\mathcal{M}^{-1}$-norm of the residual. The preconditioner A_{MG1} represents a single multigrid V-cycle: as above an 'optimally' damped Jacobi smoother was employed. Iteration and total flop counts using the 'ideal' block preconditioner $P = A$, the diagonally scaled MINRES method of example 1 above, and the block preconditioner based on a Modified Incomplete

h	$P = A$ $M = Q$	$P = A_{MG1}$ $M = Q$	$P = A_{MIC}$ $M = Q$	$P = \mathrm{diag}(A)$ $M = \mathrm{diag}(Q)$
1/8	23	27 (0.44)	28 (0.19)	41 (0.23)
1/16	25	28 (1.97)	38 (1.22)	94 (2.25)
1/32	27	30 (8.10)	53 (7.69)	206(20.63)
1/64	27	31 (36.33)	78 (54.89)	427(175.04)

TABLE 4
MINRES iterations (Megaflops): Q_1–iso–Q_2 element

Cholesky factorisation (MIC) are also included for comparison. Note that the cost of the incomplete factorisation is not included in the flop counts given in the table; preconditioning is via sparse upper and lower triangular matrix solves in this case. The use of $P = A$ is expensive in operation counts since a full factorisation is needed in this case, so only the MINRES iteration counts are included for comparison. The computations were done on a Sun Sparcstation-10 using MATLAB 4.1.

Note that use of $P = A$ or of the more practical multigrid cycle as a preconditioner for the Laplacian does indeed imply that the number of MINRES iterations does not depend on the discrete problem size. Use of the multigrid preconditioner rather than the 'ideal' choice $P = A$ is seen to increase the number of iterations only slightly: it is nearly ideal, but much more efficient overall. Indeed the multigrid preconditioner gives an 'optimal' Stokes solver: the total number of floating point operation increases by a factor of approximately four each time the grid is refined to create four times as many discrete variables. This is a very desirable property.

It remains to analyse the convergence of MINRES for preconditioners such as those in the two right hand columns of table 4 above which do not involve multigrid or some other spectrally equivalent Laplacian preconditioner.

Having estimated the eigenvalue spectrum in the form $\Lambda(\widetilde{\mathcal{A}}) \subset E$ where E comprises two intervals of the form $[-a, -b] \cup [c, d]$ with a, b, c and d being positive, our attention therefore turns to the approximation problems

$$\widehat{\rho}_k = \min_{p \in \Pi_k, p(0)=1} \max_{\lambda \in \Lambda(\widetilde{\mathcal{A}})} |p(\lambda)| \tag{4.20}$$

$$\leq \min_{p \in \Pi_k, p(0)=1} \max_{x \in E} |p(x)| := \rho_k. \tag{4.21}$$

We know from (4.2) that $\widehat{\rho}_k$ bounds the relative reduction in the MINRES residual after k iterations; if little is lost in the inequality above then it is more tractable to deal with the approximation problem (4.21) on intervals rather than (4.20) on the discrete eigenvalue set. A rapidly decreasing sequence ρ_k will still indicate fast convergence.

In fact, when a single number is desired to represent convergence, it is convenient to consider the *asymptotic convergence factor*

$$\rho := \lim_{k \to \infty} \rho_k^{1/k}$$

which represents a bound on the average contraction in the residual per iteration.

Firstly we require some results from Approximation Theory to characterise polynomials $p \in \Pi_k, p(0) = 1$ which solve the minimax problem (4.21) for different sets E (see for example [33] for these results). Note that $\{\rho_k\}$ must be a decreasing (non-negative) sequence as each successive iteration simply increases the allowable degree

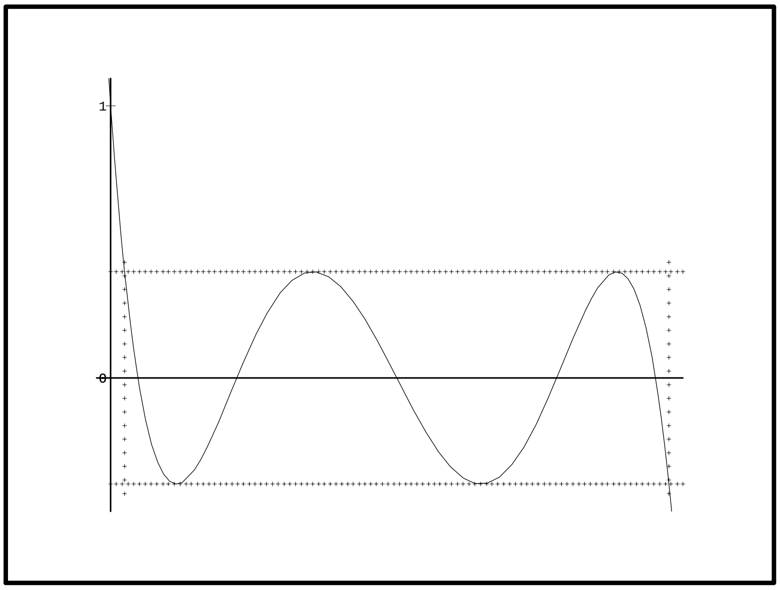


FIG. 11. *Optimal polynomial on a single interval*

of p by one. Existence and uniqueness of the solution is known and a characterisation is expressed in terms of the number of points in the set E at which ρ_k is attained.

Let us consider first the simpler problem when $E = [c, d]$ with $c > 0$ such as would arise if $\widetilde{A}$ were symmetric and *positive definite*. In this case the optimal polynomial $p \in \Pi_k$ satisfies $|p(x_j)| = \rho_k$ for $k+1$ distinct points $a = x_0 < x_1 < \ldots < x_{k-1} < x_k = b$. Moreover $p(x_j) = -p(x_{j-1})$ for $j = 1, 2, \ldots, k$. It is then a straightforward matter to see that p must be as sketched in figure 11.

We now take the unusual step of writing down an ordinary differential equation initial value problem which must be satisfied by the polynomial p. Noting that $p = \pm\rho_k$ at the points $x_1, \ldots, x_{k-1}$ where the derivative p' vanishes as well as at the endpoints of the interval $[c, d]$ we have

$$k^2(p^2(x) - \rho_k^2) = (p'(x))^2(x - c)(x - d) \tag{4.22}$$

where the constant scaling term k^2 comes from equating the leading coefficient (of x^{2k}) on both sides of this equation. The 'initial' value is $p(0) = 1$.

The nonlinear ordinary differential equation (4.22) can now be differentiated to give

$$2k^2pp' = 2p'p''(x - c)(x - d) + (p')^2(2x - c - d)$$

so the common factor p' can be cancelled to reveal the linear ordinary differential equation

$$(x - c)(x - d)p'' + (x - (c + d)/2)p' - k^2p = 0. \tag{4.23}$$

This is the classical Chebyshev equation (see for example [24], pp. 1033) the solutions of which are the well known Chebyshev polynomials $T_k(x) = \cos k\phi,\ x = \cos\phi$ suitably shifted to the interval $[c, d]$ and scaled to satisfy the side condition $p(0) = 1$. (This may be discovered by seeking a series solution).

This is a rather unusual way to show the well-known result that the solution of the polynomial approximation problem (4.21) is

$$p(x) = T_k\left(\frac{2x-c-d}{d-c}\right) \Big/ T_k\left(\frac{c+d}{c-d}\right)$$

(see for example [2]).

Using the definition in terms of the cosine it follows that $-1 \leq T_k \leq 1$ for the relevant argument and so the preconditioned MINRES convergence estimate (4.2) becomes

$$\frac{\|r_k\|}{\|r_0\|} \leq \rho_k = 1 \Big/ T_k\left(\frac{c+d}{c-d}\right). \tag{4.24}$$

If c and/or d are defined asymptotically in terms of h then use can be made of the asymptotics of Chebyshev polynomials to give asymptotic formulae for ρ_k in terms of h. For example if $c = O(h^r), d = O(1)$ then

$$\lim_{k\to\infty} \rho_k^{1/k} = 1 - O(h^{r/2})$$

(see [2]).

We use this non-standard approach here because it is actually more general since it extends to various situations where $\mathcal{M}^{-\frac{1}{2}}\mathcal{A}\mathcal{M}^{-\frac{1}{2}}$ is indefinite.

The first indefinite case we consider is $E = [-d,-c] \cup [c,d]$. We say that a symmetric matrix $\widetilde{\mathcal{A}}$ with $\lambda \in \Lambda(\widetilde{\mathcal{A}}) \Rightarrow -\lambda \in \Lambda(\widetilde{\mathcal{A}})$ is 'symmetrically indefinite': such a matrix necessarily leads to consideration of an inclusion set of this form. In this case the optimal polynomials p in (4.21) must inherit the symmetry of the inclusion set and so must be of the form sketched in figure 12. We see that p' vanishes at the origin as well as at the points where p attains $\pm\rho_k$, so in a similar manner to the above we obtain the ordinary differential equation

$$\begin{aligned} k^2x^2(p^2(x)-\rho_k^2) &= (p'(x))^2(x-c)(x-d)(x+c)(x+d) \\ &= (p'(x))^2(x^2-c^2)(x^2-d^2) \end{aligned}$$

which is necessarily satisfied by the optimal polynomial.

Making the change of variable $y = x^2$ and setting $p(x) = q(y)$ so that $p'(x) = 2xq'(y)$ we obtain

$$k^2x^2(q^2(y)-\rho_k^2) = 4x^2(q'(y))^2(y-c^2)(y-d^2)$$

or

$$(k/2)^2(q^2(y)-\rho_k^2) = (q'(y))^2(y-c^2)(y-d^2).$$

This is precisely in the form of (4.22) and so proceding as above we obtain q as the Chebyshev polynomial of degree $k/2$ shifted to the interval $[c^2, d^2]$ and scaled to satisfy $q(0) = 1$:

$$q(y) = T_{k/2}\left(\frac{2y-c^2-d^2}{d^2-c^2}\right) \Big/ T_{k/2}\left(\frac{c^2+d^2}{c^2-d^2}\right).$$

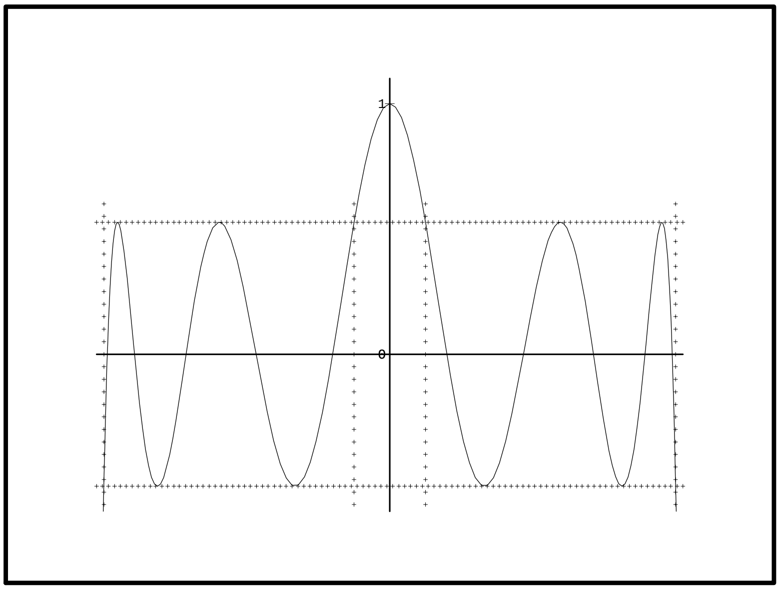


FIG. 12. *Optimal polynomial on two intervals symmetric about the origin*

Thus in terms of the optimal polynomial p of degree k we have

$$p(x) = T_{k/2}\left(\frac{2x^2 - c^2 - d^2}{d^2 - c^2}\right) \Big/ T_{k/2}\left(\frac{c^2 + d^2}{c^2 - d^2}\right)$$

and so for symmetrically indefinite systems the MINRES convergence estimate is

$$\frac{\|r_k\|}{\|r_0\|} \leq \rho_k = 1 \Big/ T_{k/2}\left(\frac{c^2 + d^2}{c^2 - d^2}\right). \tag{4.25}$$

In a partial differential equation situation where $c = O(h^r), d = O(1)$ as above, we would thus have

$$\lim_{k \to \infty} \rho_k^{1/k} = 1 - O(h^r)$$

It is instructive to compare this with the convergence that would be achieved by an iterative method such as MINRES (or Conjugate Gradients) applied to the symmetric and positive definite 'normal equations' (NE), $\widetilde{\mathcal{A}}^2\,\widetilde{x} = \widetilde{\mathcal{A}}\,\widetilde{b}$. For this system $\Lambda(\widetilde{\mathcal{A}}^2) \subset [c^2, d^2]$ so that we can estimate convergence using (4.24) to obtain

$$\frac{\|r_k^{NE}\|}{\|r_0^{NE}\|} \leq \rho_k^{NE} = 1 \Big/ T_k\left(\frac{c^2 + d^2}{c^2 - d^2}\right). \tag{4.26}$$

Comparing (4.25) with (4.26) we see that we can expect that MINRES for the original indefinite symmetric problem will take twice the number of iterations as MINRES (or the more efficient Conjugate Gradient method) for the normal equations to achieve the same reduction of residual. Since for the normal equations *two* matrix-vector multiplies will be required at each iteration compared to only one for the indefinite system so that the normal equation method will be twice as expensive per iteration, we deduce that there is essentially nothing to choose between these two approaches in this case. That is, the iterative solution of 'symmetrically indefinite' systems using a method such as MINRES is no better than the much more generally applicable normal equations approach. A more precise statement of this result is given by Freund [19].

When proposing the use of preconditioned MINRES for a class of indefinite systems it is therefore important to show that the eigenvalues are *not* symmetric about the origin. For the Stokes problem the results of the previous section establish a precise non-symmetry in the eigenvalues: for non-optimal preconditioners, the negative eigenvalues approach the origin at half of the rate of the positive eigenvalues under mesh refinement.

In this situation, the approach employing ordinary differential equations as above can still be employed to derive a convergence estimate, though the details are rather more involved (see [50]). We quote only the result: If the eigenvalues of $\widetilde{\mathcal{A}}$ are contained in a set of the form

$$[-a, -bh^{r/2}] \,\cup\, [ch^r, d]$$

then

$$\lim_{k\to\infty} \rho_k^{1/k} = 1 - O(h^{3r/4}).$$

That is the convergence of MINRES on the Stokes problem is at a rate precisely half way between that achived for a symmetric positive definite problem such as the Laplacian and that achieved for the corresponding normal equations.

5. Solution methods for the discrete Oseen equations. In this section we examine methods for solving the steady-state Navier-Stokes equations that combine and build on the techniques of sections 3 and 4. The methods are designed for the steady-state Oseen equations ($\Delta t \to \infty$ in (1.11)). These equations also arise from a nonlinear iteration for solving the Navier-Stokes equations in which $\mathbf{u}^*$ represents the iterate from a given step and the solution $\mathbf{u}$ is the iterate for the next step. See [29] for a convergence analysis.

Discretisation leads to a matrix problem

$$\begin{pmatrix} F & B^t \\ B & -\beta S \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}, \tag{5.1}$$

where u and p now represent discrete versions of velocity and pressure, respectively. F is a discrete vector convection-diffusion operator and B represents the coupling between the discrete velocity u and the pressure p. For simplicity of presentation we only present results for the unstabilised case $S = 0$.

5.1. Preconditioning I: Convection-diffusion solves. We first describe two preconditioning techniques developed in [7] that generalise the methods of section 4 essentially by replacing the approximation P to the vector Laplacian operator in (4.3) with an approximation to the vector convection-diffusion operator F. It is easiest to describe the ideas using the exact operator F. Thus, consider the block diagonal preconditioner

$$\begin{pmatrix} F & 0 \\ 0 & \frac{1}{\nu}Q \end{pmatrix} \tag{5.2}$$

where Q is the pressure mass matrix. As in section 4, the eigenvalues of the preconditioned system are the solutions of the generalised eigenvalue problem

$$\begin{pmatrix} F & B^t \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \lambda \begin{pmatrix} F & 0 \\ 0 & \frac{1}{\nu}Q \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix}.$$

These are given by $\lambda = 1$ or

$$\lambda = \frac{1 \pm \sqrt{1+4\mu}}{2}$$

where μ comes from the generalised eigenvalue problem for the Schur complement system,

$$BF^{-1}B^t p = \mu \left(\frac{1}{\nu}Q\right) p\,. \tag{5.3}$$

The following result provides a bound on μ.

THEOREM 5.1. *The eigenvalues of the generalised Schur complement problem* (5.3) *for the Oseen operator are contained in a rectangular box in the right half plane of the form*

$$\left[\frac{\gamma^2\nu^2}{\delta^2+\nu^2}, \Gamma^2\right] \times i \left[\frac{\Gamma^2}{2}, \frac{\Gamma^2}{2}\right].$$

where γ *and* Γ *are as in* (2.30) *and* (2.37), *and* $\delta = \rho(A^{-1}N)$.

This is proved [7] by bounding the eigenvalues of the symmetric part of $BF^{-1}B^t$ (with respect to $\frac{1}{\nu}Q$) and the skew-symmetric part of $BF^{-1}B^t$, and then applying Bendixson's theorem ([47], p. 418). But γ and Γ are independent of the mesh size h of the discretisation. Moreover, since N and A are first-order and second-order operators, respectively, δ is also independent of h [15]. Consequently, the box containing the generalised eigenvalues of (5.3) are independent of the discretisation mesh size. A bound on the eigenvalues of the preconditioned Oseen operator is an immediate consequence.

COROLLARY 5.2. *The eigenvalues of the discrete Oseen operator* (5.1) *preconditioned by* (5.2) *consist of* $\lambda = 1$ *of multiplicity* $n_u - n_p$, *together with four sets consisting of points of the form* $1 + (a \pm bi)$ *and* $-a \pm bi$. *These sets can be enclosed in two rectangular regions that are symmetric with respect to* $\Re(\lambda) = \frac{1}{2}$ *whose borders are bounded independently of* h.

The preconditioned system can be solved using any Krylov subspace method. The convergence behavior of such methods depends implicitly on finding a polynomial that is small on the spectrum of the coefficient matrix. (Again see [42] or the paper by Van der Vorst in this volume.) The fact that the eigenvalues for the preconditioned system derived from (5.2) lie on both sides of the imaginary axis is a potential disadvantage of this preconditioner. An alternative that avoids this problem is the block triangular preconditioning operator

$$\begin{pmatrix} F & B^t \\ 0 & -\frac{1}{\nu}Q \end{pmatrix}. \tag{5.4}$$

For this choice, the associated generalised eigenvalue problem is

$$\begin{pmatrix} F & B^t \\ B & O \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \lambda \begin{pmatrix} F & B^t \\ 0 & -\frac{1}{\nu}Q \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix}. \tag{5.5}$$

As above, one solution is $\lambda = 1$, now of multiplicity n_u. If $\lambda \neq 1$, then premultiplying the first block row of (5.5) by BF^{-1} and using the relation $Bu = -\lambda\,(\frac{1}{\nu}Q)p$ leads to the equation (5.3) for the other eigenvalues. Thus, we have the following result.

THEOREM 5.3. *The eigenvalues of the discrete Oseen operator preconditioned by* (5.4) *consist of* $\lambda = 1$ *together with the generalised eigenvalues of* S *in* (5.3). *Therefore, the eigenvalues are bounded independently of* h *and they all have positive real part.*

The analysis in [16] shows that for a particular starting guess the i'th GMRES polynomial derived from the triangular preconditioning (5.4) is identical to the $(2i-1)st$ GMRES polynomial for the diagonal preconditioning (5.2). Experimental results for both GMRES and the quasi-minimal residual method (QMR) [20] indicate that this analysis is predictive for arbitrary initial guesses, i.e., the triangular method requires roughly half the iterations to converge [7]. Moreover, the inverse of the block triangular preconditioner can be expressed in factored form as

$$\begin{pmatrix} F & B^t \\ 0 & -\frac{1}{\nu}Q \end{pmatrix}^{-1} = \begin{pmatrix} F^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & B^t \\ 0 & -I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & \nu Q^{-1} \end{pmatrix},$$

so that the only overhead associated with using (5.4) instead of (5.2) is a matrix multiplication by B^t. Therefore, this preconditioner is typically more effective for the Oseen problem.

Since the eigenvalues for either of these preconditioners are independent of the mesh size, the asymptotic convergence rate of GMRES is also independent h [42]. Table 5 shows the iterations required by GMRES and QMR to solve the driven cavity problem on $\Omega = (-1,1)\times(-1,1)$ using the $Q_1\!-\!iso\!-\!Q_2$ discretization with an $n \times n$ non-uniform grid of elements for velocities. The initial guess was identically zero and the stopping criterion was

$$\frac{\left\| \begin{pmatrix} f \\ 0 \end{pmatrix} - \begin{pmatrix} F & B^t \\ B & 0 \end{pmatrix} \begin{pmatrix} u_k \\ p_k \end{pmatrix} \right\|_2}{\left\| \begin{pmatrix} f \\ 0 \end{pmatrix} \right\|_2} \leq 10^{-6}.$$

These results, which come from [7], indicate that the iteration counts are independent of the mesh size. (This is less evident for the smallest value $\nu = 1/100$ considered here; we believe that this is because finer meshes are needed for the asymptotic behavior to be displayed in this case.) See [7] for additional experimental results.

5.2. Preconditioning II: Stokes solves. An alternative approach considered in [23] builds on the ideas of section 4 in a different way, by using a symmetric operator as a preconditioner for the Oseen equations. Here we consider one example from [23], the symmetric part of (5.1). This is a discrete Stokes operator

$$\begin{pmatrix} \nu A & B^t \\ B & 0 \end{pmatrix}. \tag{5.6}$$

Thus, using this with a Krylov subspace method entails solving the discrete Stokes equations at each step. See [23] for other examples of symmetric preconditioners as well as a discussion of their use for stationary iterative methods.

Iterations of GMRES

Grid	$\nu = 1$	$\nu = 1/10$	$\nu = 1/50$
16×16	18	25	45
32×32	19	31	69
64×64	17	32	93
128×128	14	31	110

Iterations of QMR

Grid	$\nu = 1$	$\nu = 1/10$	$\nu = 1/50$	$\nu = 1/100$
16×16	22	28	51	73
32×32	22	36	78	126
64×64	22	39	112	189
128×128	16	36	127	253

TABLE 5

Iterations for Q_1–iso–Q_2 finite elements applied to the Oseen equation with block triangular preconditioning.

An analysis of the Stokes preconditioning is as follows. Once again, we have a generalised eigenvalue problem,

$$\begin{pmatrix} F & B^t \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \lambda \begin{pmatrix} \nu A & B^t \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix}.$$

One solution is $\lambda = 1$, which has eigenvectors of the form $(u,p)^t$ where $Nu = 0$ and p is arbitrary. Any remaining eigenvalues satisfy

$$Fu + B^t p = \lambda \left((\nu A)u + B^t p\right)$$

where u is such that $Bu = 0$. If $(u,p)^t$ is any eigenvector, then taking the inner product with u leads to the expression for the corresponding eigenvalue

$$\lambda = 1 + \frac{(u, Nu)}{(u, (\nu A)u)}.$$

(Note that if u exists it will be complex.) It follows that

$$|\Im(\lambda)| \le \frac{1}{\nu}\rho(A^{-1}N).$$

Thus, we have established the following result.

THEOREM 5.4. *The eigenvalues of the discrete Oseen operator* (5.1) *preconditioned by* (5.6) *consist of $\lambda = 1$ of multiplicity at least n_p together with at most $n_u - n_p$ eigenvalues of the form $1 \pm i\eta/\nu$ where $|\eta| \le \rho(A^{-1}N)$.*

These eigenvalues lie on a vertical line segment in the complex plane with real part equal to 1. As noted in section 5.1, $\rho(A^{-1}N)$ is independent of the mesh size, so that the asymptotic convergence rate of GMRES will also be independent of h [23, 42].

Table 6 shows the results of numerical experiments with the Stokes preconditioner [23] applied to the driven cavity problem. Here the discretization is locally stabilised $Q_1 - P0$ with $\beta = 1/4$. (See section 6.) The stopping criterion and initial guess are as in section 5.2.

Iterations of GMRES

Grid	$\nu = 1$	$\nu = 1/10$	$\nu = 1/100$
8×8	5	11	26
16×16	4	12	39
32×32	4	12	45
64×64	4	12	45

Iterations of QMR

Grid	$\nu = 1$	$\nu = 1/10$	$\nu = 1/50$	$\nu = 1/100$
8×8	7	12	27	45
16×16	5	14	40	67
32×32	5	14	47	83
64×64	6	13	47	89

TABLE 6

Iterations for locally stabilised $Q_1 - P_0$ finite elements (with $\beta = 1/4$) applied to the Oseen equation with Stokes preconditioning.

5.3. Discussion. We conclude this section with a brief discussion comparing the two classes of ideas presented here. Each of the approaches requires the solution of a key subproblem, the discrete convection-diffusion equation for the methods of section 5.1 and the discrete Stokes equations for the method of section 5.2.[3] We have not made a systematic comparison of these approaches and will refrain from making a recommendation here. For a practical computation we would expect the solution of either of the subproblems to be replaced by an approximate solution obtained using an iterative method. These computations could be done using the techniques of sections 3 or 4. This issue adds to the difficulty in making a comparison of the two approaches.

Finally, we point out that although both methodologies discussed here produce asymptotic convergence rates that are independent of the mesh size, they are dependent on the viscosity ν. This is seen in the lower bound of ν^2 for the real parts of the eigenvalues in Theorem 5.1 (which is shown to be tight in [7]) and the upper bound of $1/\nu$ in Theorem 5.3. In both cases the iteration counts appear to grow linearly in $1/\nu$, and therefore we expect these ideas to be most suitable problems with relatively high viscosity, i.e., low Reynolds numbers.

6. Test Problems and Software. In this section, the test problems used to illustrate the methodology in sections 3–5 are described. These problems can be constructed (and the solutions plotted) using MATLAB software which is available by anonymous `ftp` in the `tar` files
`ftp://ftp.ma.man.ac.uk/pub/narep/convdiff.tar`
`ftp://ftp.ma.man.ac.uk/pub/narep/oseen.tar`
The three test problems that are built-in are described below.

[3] As described, the techniques of section 5.1 also require the action of the inverse of the mass matrix. However, as we observed in section 4, this can be replaced with a less expensive computation using, say, the diagonal of the mass matrix, without affecting asymptotic convergence properties. Indeed, this choice was used for the results of section 5.1.

6.1. The Convection-Diffusion Problem. The directory `/convdiff/` contains two driver routines; `square_grid` and `stretch_grid`. These generate solutions to (2.1) using square or rectangular bilinear Q_1 elements. The "wind" $\mathbf{w}$ is defined within the function `transprt.m`, and for the test problem (see section 3.3) it is set to a constant vector $(-\sqrt{2}/2, \sqrt{2}/2)$. The boundary conditions are defined in the function `skewx.m`. In the test problem, the solution satisfies $u = 1$ on part of the bottom boundary and on the right-hand wall, and $u = 0$ on the remainder.

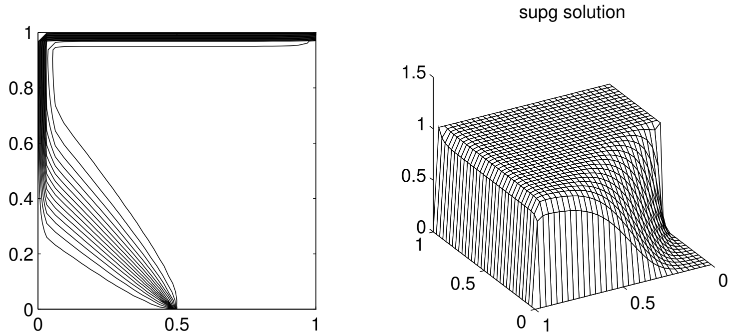


FIG. 13. *Convection skew to the mesh*

Reducing the viscosity parameter ν increases the relative strength of the wind, and if ν is "small" there is an internal layer generated by the discontinuity on the inflow boundary, and a boundary layer at the left hand wall and along the top. The case $\nu = 1/100$ is illustrated in figure 13. This shows a uniform 32×32 grid solution corresponding to the streamline diffusion formulation (2.8), and was generated via `square_grid`. Note that for this combination of ν and h the standard Galerkin solution is oscillatory, unless stretched grids are used to resolve the boundary layer (via the routine `stretch_grid`).

6.2. The Stokes Problem. The directory `/oseen/` contains two driver routines; `square_mesh` and `stretch_mesh`. These generate finite element matrices associated with the Oseen operator using square (or rectangular) Q_1–P_0 elements. These matrices are "saved" on the datafile `system_nobc.mat`.

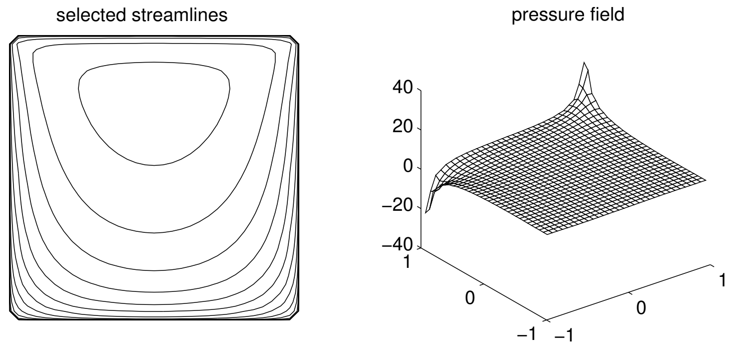


FIG. 14. *Stokes driven cavity flow*

Having set up the system matrices, the Stokes flow test problem (see sections 4.4 and 4.5) can be solved using the driver `stokes`. The "leaky driven cavity" boundary conditions are defined in the function `ldcavf.m`; the vertical velocity is set to zero everywhere, whereas the horizontal velocity is set to unity on the lid, and is zero on the other boundaries. One of the interesting features of the problem is that the pressure is singular at the top corners, i.e. where the imposed velocity is discontinuous. Without convection the flow is (anti–)symmetric about the line $x = 0$, where the pressure must be identically zero. This feature can be exploited when generating the flow solution (see section 4). A typical flow is illustrated in figure 14. This shows a uniform 32×32 grid solution of the stabilised system (2.35) with the "optimal" stabilisation parameter $\beta = 0.058$. Using Q_1–P_0 the pressure solution becomes increasingly oscillatory as $\beta \to 0$, although a realistic velocity solution is obtained for this test problem without stabilisation (this is not true in general). If stretched grids are used (via the routine `stretch_mesh`) then secondary recirculations (so called "Moffatt eddies") can be observed in the bottom two corners.

6.3. The Oseen Problem. Having set up the system matrices as above (using `square_mesh` or `stretch_mesh`), the Oseen flow test problem (see sections 5.1 and 5.2) can be solved using the driver `osn`. Unlike the Stokes case where there is no convection, in the Oseen problem there is a "wind" which is defined within the function `wind.m`. For the test problem the wind is the "divergence-free vortex" $\mathbf{w} = (2y(1-x^2), -2x(1-y^2))$. As in the Stokes case, the "leaky cavity" boundary conditions are defined in the function `ldcavf.m`.

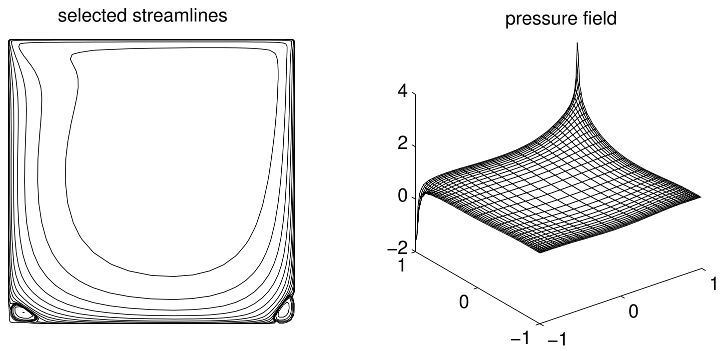


FIG. 15. *Oseen driven cavity flow*

Reducing the viscosity parameter ν increases the relative strength of the wind, and if ν is "small" the centre of primary recirculation (which is on the line $x = 0$ in the Stokes case) is moved significantly to the right. The case $\nu = 1/50$ is illustrated in figure 15. This shows a stretched grid 32×32 grid solution of the stabilised system (5.1) with a stabilisation parameter $\beta = 1/4$. The secondary recirculations can be clearly observed here.

REFERENCES

[1] O. AXELSSON, *Iterative Solution Methods*, Cambridge University Press, New York, 1994.
[2] O. AXELSSON AND V. A. BARKER, *Finite Element solution of boundary value problems: Theory and Computation*, Academic Press, New York, 1984.
[3] R. BEAUWENS, *Factorization iterative methods, M-operators and H-operators*, Numer. Math., 31 (1979), pp. 335–357.

[4] J. Bramble and J. Pasciak, *Iterative techniques for time dependent Stokes problems*, in Solution Techniques for Large-Scale CFD Problems, W. Habashi, ed., John Wiley, 1995, pp. 201–216.

[5] E. Dean and R. Glowinski, *On some finite element methods for the numerical solution of incompressible viscous flow*, in Incompressible Computational Fluid Dynamics, M. Gunzburger and R. Nicolaides, eds., 1993.

[6] J. Douglas and T. Russell, *Numerical methods for convection dominated diffusion problems based on combining the method of characteristics with finite elements or finite differences*, SIAM J. Numer. Anal., 19 (1982), pp. 871–885.

[7] H. Elman and D. Silvester, *Fast nonsymmetric iterations and preconditioning for Navier-Stokes equations*, SIAM J. Sci. Comput., 17 (1996), pp. 33–46.

[8] H. C. Elman, *Multigrid and Krylov subspace methods for the discrete Stokes equations*, Int. J. Numer. Meth. Fluids, 227 (1996), pp. 55–770.

[9] H. C. Elman and M. P. Chernesky, *Ordering effects on relaxation methods applied to the discrete one-dimensional convection-diffusion equation*, SIAM J. Numer. Anal., 30 (1993), pp. 1268–1290.

[10] ———, *Ordering effects on relaxation methods applied to the discrete convection-diffusion equation*, in Recent Advances in Iterative Methods, G. H. Golub, A. Greenbaum, and M. Luskin, eds., Springer-Verlag, New York, 1994, pp. 45–57.

[11] H. C. Elman and G. H. Golub, *Iterative methods for cyclically reduced non-self-adjoint linear systems*, Math. Comp., 54 (1990), pp. 671–700.

[12] ———, *Iterative methods for cyclically reduced non-self-adjoint linear systems, II*, Math. Comp., 56 (1991), pp. 215–242.

[13] ———, *Line iterative methods for cyclically reduced convection-diffusion problems*, SIAM J. Sci. Stat. Comput., 13 (1992), pp. 339–363.

[14] ———, *On the convergence of line iterative methods for cyclically reduced nonsymmetrizable linear systems*, Numer. Math., 67 (1994), pp. 177–190.

[15] H. C. Elman and M. H. Schultz, *Preconditioning by fast direct methods for nonselfadjoint nonseparable elliptic problems*, SIAM J. Numer. Anal, 23 (1986), pp. 44–57.

[16] B. Fischer, A. Ramage, D. Silvester, and A. Wathen, *Minimum residual methods for augmented systems*, Tech. Report No. 15, Mathematics Dept., Strathclyde University, June 1995. submitted to SIAM J. Matrix Anal. Appl.

[17] B. Fischer, A. Ramage, D. J. Silvester, and A. J. Wathen, *Optimal streamline upwinding for advection diffusion problems*, tech. report, Manchester Centre for Computational Mathematics, June 1996. in preparation.

[18] M. Fortin and R. Pierre, *Stability analysis of discrete generalised Stokes problems*, Numer. Meth. Partial Diff. Eq., 8 (1992), pp. 303–323.

[19] R. Freund, *On polynomial preconditioning for indefinite hermitian matrices*, Tech. Report 89.32, Research Institute for Advanced Computer Science, NASA Ames Research Center, August 1989.

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

[21] V. Girault and P. Raviart, *Finite Element Methods for Navier-Stokes Equations*, Springer-Verlag, Berlin, 1986.

[22] G. H. Golub and C. F. Van Loan, *Matrix Computations*, John Hopkins, Baltimore, 1st ed., 1983.

[23] G. H. Golub and A. J. Wathen, *An Iteration for Indefinite Systems and its Application to the Navier-Stokes Equations*, Tech. Report AM-95-09, Mathematics Department, University of Bristol, 1995. To appear in SIAM J. Sci. Stat. Comput.

[24] I. S. Gradshteyn and I. M. Ryzhik, *Tables of Integrals, Series and Products*, Academic Press, London, 4th ed., 1965. translated by A. Jeffrey.

[25] P. M. Gresho and R. L. Lee, *Don't suppress the wiggles – they're telling you something*, Computers and Fluids, 9 (1981), pp. 223–253.

[26] I. Gustafsson, *A class of first order factorisation methods*, BIT, 18 (1978), pp. 142–156.

[27] C. Johnson, *Numerical Solution of Partial Differential Equations by the Finite Element Method*, Cambridge University Press, New York, 1987.

[28] ———, *The streamline diffusion finite element method for compressible and incompressible fluid flow*, in Numerical Analysis 1989, D. F. Griffiths and G. Watson, eds., Pitman Research Notes in Mathematics Series, 1989.

[29] O. A. Karakashian, *On a Galerkin-Lagrange multiplier method for the stationary Navier-Stokes equations*, SIAM. J. Numer. Anal., 19 (1982), pp. 909–923.

[30] N. Kechkar and D. Silvester, *Analysis of locally stabilised mixed finite element methods for the Stokes problem*, Math. Comp., 58 (1992), pp. 1–10.

[31] P. Klouček and F. Rys, *Stability of the fractional step θ-scheme for the nonstationary Navier-Stokes equations*, SIAM J. Numer. Anal., 31 (1994), pp. 1312–1335.

[32] J. A. Meijerink and H. A. van der Vorst, *An iterative solution method for linear systems of which the coefficient matrix is a symmetric m-matrix*, Math. Comp., 31 (1977), pp. 148–162.

[33] G. Meinardus, *Approximation of functions: theory and numerical methods*, Springer, New York, 1967.

[34] J. Ortega, *Numerical Analysis: A Second Course*, Academic Press, New York, 1972.

[35] C. C. Paige and M. A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.

[36] S. V. Parter, *On estimating the "rates of convergence" of iterative methods for elliptic difference operators*, Trans. Amer. Math. Soc., 114 (1965), pp. 320–354.

[37] ———, *Iterative methods for elliptic problems and the discovery of "q"*, SIAM Review, 28 (1986), pp. 153–175.

[38] S. V. Parter and M. Steuerwalt, *Block iterative methods for elliptic and parabolic difference equations*, SIAM J. Numer. Anal., 19 (1982), pp. 1173–1195.

[39] D. Peaceman and A. Rachford, *The numerical solution of parabolic and elliptic differential equations*, SIAM J., 3 (1955), pp. 28–41.

[40] A. Quarteroni and A. Valli, *Numerical Approximation of Partial Differential Equations*, Springer-Verlag, Berlin, 1994.

[41] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing, Boston, 1996.

[42] Y. Saad and M. Schultz, *GMRES: a generalised minimal residual algorithm for solving non-symmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869.

[43] D. Silvester, *Optimal low order finite element methods for incompressible flow*, Comp. Meths. Appl. Mech. Engrg., 111 (1994), pp. 357–368.

[44] D. J. Silvester and A. J. Wathen, *Fast and robust solvers for time-discretised incompressible Navier-Stokes equations*, in Numerical Analysis: proceedings of the 1995 Dundee biennial conference, D. F. Griffiths and G. A. Watson, eds., Longman, 1996. Pitman Research Notes in Mathematics Series 344.

[45] J. Simo and F. Armero, *Unconditional stability and long-term behavior of transient algorithms for the incompressible Navier-Stokes and Euler equations*, Comp. Meth. Appl. Mech. Eng., 111 (1994), pp. 111–154.

[46] A. Smith and D. J. Silvester, *Implicit algorithms and their linearisation for the transient Navier-Stokes equations*, Tech. Report 290, Manchester Centre for Computational Mathematics, June 1996.

[47] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*, Springer-Verlag, New York, second ed., 1993.

[48] R. S. Varga, *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, New Jersey, 1962.

[49] A. J. Wathen, *Realistic eigenvalue bounds for the Galerkin mass matrix*, IMA J. Numer. Anal., 7 (1987), pp. 449–457.

[50] A. J. Wathen, B. Fischer, and D. J. Silvester, *The convergence rate of the minimum residual method for the Stokes problem*, Numer. Math., 71 (1995), pp. 121–134.

[51] Z. Woźnicki, *Two-Sweep Iterative Methods for Solving Large Linear Systems and their Application to the Numerical Solution of Multi-Group Multi-Dimensional Neutron Diffusion Equation*, PhD thesis, Institute of Nuclear Research, Swierk, Poland, 1973. Report N^{0}1447/CYFRONET/PM/A.

[52] D. M. Young, *Iterative Solution of Large Linear Systems*, Academic Press, New York, 1970.