Numerical Solution of Saddle Point Problems

Article *in* Acta Numerica · May 2005

DOI: 10.1017/S0962492904000212

3 authors, including:

Michele Benzi
Scuola Normale Superiore di Pisa
176 PUBLICATIONS **10,330** CITATIONS
SEE PROFILE

Jörg Liesen
Technische Universität Berlin
147 PUBLICATIONS **3,059** CITATIONS
SEE PROFILE

Technical Report

TR-2004-028

Numerical Solution of Saddle Point Problems

by

Michele Benzi, Gene H. Golub, Jorg Liesen

Mathematics and Computer Science

EMORY UNIVERSITY

NUMERICAL SOLUTION OF SADDLE POINT PROBLEMS*

MICHELE BENZI†, GENE H. GOLUB‡, AND JÖRG LIESEN§

We dedicate this paper to Gil Strang on the occasion of his 70th birthday

Abstract. Large linear systems of saddle point type arise in a wide variety of applications throughout computational science and engineering. Due to their indefiniteness and often poor spectral properties, such linear systems represent a significant challenge for solver developers. In recent years there has been a surge of interest in saddle point problems, and numerous solution techniques have been proposed for solving this type of systems. The aim of this paper is to present and discuss a large selection of solution methods for linear systems in saddle point form, with an emphasis on iterative methods for large and sparse problems.

CONTENTS

1. Introduction. In recent years, a large amount of work has been devoted to the problem of solving large linear systems in saddle point form. The reason for this interest is due to the fact that such problems arise in a wide variety of technical and scientific applications. For example, the ever increasing popularity of mixed finite element methods in engineering fields such as fluid and solid mechanics has been a major source of saddle point systems [79, 170]. Another reason for this surge in interest is due to the extraordinary success of interior point algorithms in both linear and nonlinear optimization, which require at their heart the solution of a sequence of systems in saddle point form [371, 506, 507].

Because of the ubiquitous nature of saddle point systems, methods and results on their numerical solution have appeared in a wide variety of books, journals and conference proceedings, justifying

*This draft dated 13 December 2004. To appear in *Acta Numerica 2005*.

†Department of Mathematics and Computer Science, Emory University, Atlanta, Georgia 30322, USA (benzi@mathcs.emory.edu). The work of this author was supported in part by the National Science Foundation grant DMS-0207599.

‡Scientific Computing and Computational Mathematics Program, Stanford University, Stanford, California 94305-9025, USA (golub@sccm.stanford.edu).

§Institut für Mathematik, Technische Universität Berlin, D-10623 Berlin, Germany (liesen@math.tu-berlin.de).

the need for a comprehensive survey of the subject. The purpose of this article is to review many of the most promising solution methods, with an emphasis on iterative methods for large and sparse problems. Although many of these solvers have been developed with specific applications in mind (for example, Stokes-types problems in fluid dynamics), it is possible to discuss them in a fairly general setting using standard numerical linear algebra concepts, the most prominent being perhaps the Schur complement. Nevertheless, when choosing a preconditioner (or developing a new one), knowledge of the origin of the particular problem at hand is essential. We therefore devote some space to a discussion of saddle point problems arising in a few selected applications.

It is hoped that the present survey will prove useful to practitioners who are looking for guidance in the choice of a solution method for their own application, to researchers in numerical linear algebra and scientific computing, and especially to graduate students as an introduction to this very rich and important subject.

1.1. Problem statement and classification. The subject of this paper is the solution of block 2×2 linear systems of the form

$$\begin{bmatrix} A & B_1^T \\ B_2 & -C \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}, \quad \text{or} \quad \mathcal{A}u = b, \tag{1.1}$$

$$A \in \mathbb{R}^{n\times n}, \quad B_1, B_2 \in \mathbb{R}^{m\times n}, \quad C \in \mathbb{R}^{m\times m} \quad \text{with } n \geq m. \tag{1.2}$$

It is obvious that, under suitable partitioning, *any* linear system can be cast in the form (1.1)–(1.2). We explicitly exclude the case where A or one or both of B_1, B_2 are zero. When the linear system describes a (generalized) saddle point problem, the constituent blocks A, B_1, B_2 and C satisfy one or more of the following conditions:

- C1 A is symmetric: $A = A^T$
- C2 The *symmetric part* of A, $H \equiv \frac{1}{2}(A + A^T)$, is positive semidefinite
- C3 $B_1 = B_2 = B$
- C4 C is symmetric ($C = C^T$) and positive semidefinite
- C5 $C = O$ (the zero matrix)

Note that C5 implies C4. The most basic case is obtained when *all* the above conditions are satisfied. In this case A is symmetric positive semidefinite and we have a *symmetric* linear system of the form

$$\begin{bmatrix} A & B^T \\ B & O \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}. \tag{1.3}$$

This system arises as the first order optimality conditions for the following equality-constrained quadratic programming problem:

$$\min \ J(x) = \frac{1}{2}x^T Ax - f^T x \tag{1.4}$$

$$\text{subject to } \ Bx = g. \tag{1.5}$$

In this case the variable y represents the vector of Lagrange multipliers. Any solution (x_*, y_*) of (1.3) is a saddle point for the Lagrangian

$$\mathcal{L}(x, y) = \frac{1}{2}x^T Ax - f^T x + (Bx - g)^T y,$$

hence the name "saddle point problem" given to (1.3). Recall that a saddle point is a point $(x_*, y_*) \in \mathbb{R}^{n+m}$ that satisfies

$$\mathcal{L}(x_*, y) \leq \mathcal{L}(x_*, y_*) \leq \mathcal{L}(x, y_*) \quad \forall x \in \mathbb{R}^n, \forall y \in \mathbb{R}^m,$$

or, equivalently,

$$\min_x \max_y \mathcal{L}(x,y) = \mathcal{L}(x_*,y_*) = \max_y \min_x \mathcal{L}(x,y)\,.$$

Systems of the form (1.3) also arise in nonlinearly constrained optimization (sequential quadratic programming and interior point methods), in fluid dynamics (Stokes problem), incompressible elasticity, circuit analysis, structural analysis, and so forth; see the next section for a discussion of applications leading to saddle point problems.

Another important special case is when conditions C1–C4 are satisfied, but not C5. In this case we have a block linear system of the form:

$$\begin{bmatrix} A & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}. \tag{1.6}$$

Problems of this kind frequently arise in the context of *stabilized* mixed finite element methods. Stabilization is used whenever the discrete variables x and y belong to finite element spaces that do not satisfy the Ladyzhenskaya–Babuška–Brezzi (or inf-sup) condition [79]. Another situation leading to a nonzero C is the discretization of the equations describing slightly compressible fluids or solids [69, Chapter 6.3]. Systems of the form (1.6) also arise from regularized, weighted least-squares problems [49] and from certain interior point methods in optimization [506, 507]. Often the matrix C has small norm compared to the other blocks.

In the literature, the phrase *generalized saddle point problem* has been used primarily to allow for the possibility of a nonsymmetric coefficient matrix $\mathcal{A}$ in (1.1). In such problems either $A \neq A^T$ (with condition C2 usually satisfied), or $B_1 \neq B_2$, or both. The most important example is perhaps that of the linearized Navier–Stokes equations, where linearization has been obtained by Picard iteration or by some variant of Newton's method. See [111, 370] and [456] for additional examples. We note that our definition of generalized saddle point problem as a linear system of the form (1.1)–(1.2) where the blocks A, B_1, B_2 and C satisfy one or more of the conditions C1-C5 is the most general possible, and it contains previous definitions as special cases.

In the vast majority of cases, linear systems of saddle point type have real coefficients, and in this paper we restrict ourselves to the real case. Complex coefficient matrices, however, do arise in some cases; see, e.g., [61, 345] and [449, page 117]. Most of the results and algorithms reviewed in this paper admit straightforward extensions to the complex case.

1.2. Sparsity, structure and size. Although saddle point systems come in all sizes and with widely different structural and sparsity properties, in this paper we are mainly interested in problems that are both large and sparse. This justifies our emphasis on iterative solvers. Direct solvers, however, are still the preferred method in optimization and other areas. Furthermore, direct methods are often used in the solution of subproblems, for example as part of a preconditioner solve. Some of the algorithms considered in this paper are also applicable if one or more of the blocks in $\mathcal{A}$ happen to be dense, as long as matrix-vector products with $\mathcal{A}$ can be performed efficiently, typically in $O(n+m)$ time. This means that if a dense block is present, it must have a special structure (e.g., Toeplitz, as in [49, 285]) or it must be possible to approximate its action on a vector with (nearly) linear complexity, as in the fast multipole method [345].

Frequently, the matrices that arise in practice have quite a bit of structure. For instance, the A block is often block diagonal, with each diagonal block endowed with additional structure. Many of the algorithms discussed in this paper are able to exploit the structure of the problem to gain efficiency and save on storage. Sometimes the structure of the problem suggests solution algorithms

that have a high degree of parallelism. This last aspect, however, is not emphasized in this paper. Finally we mention that in most applications n is larger than m, often much larger.

2. Applications leading to saddle point problems. As already mentioned, large-scale saddle point problems occur in many areas of computational science and engineering. The following is a list of some fields where saddle point problems naturally arise, together with some references:

- Computational fluid dynamics [213, 407, 459, 469, 499]
- Constrained and weighted least squares estimation [59, 222]
- Constrained optimization [210, 506, 507]
- Economics [18, 143, 320, 456]
- Electrical circuits and networks [51, 109, 449, 467]
- Electromagnetism [67, 390, 392]
- Finance [348, 349]
- Image reconstruction [255]
- Image registration [248, 362]
- Interpolation of scattered data [342, 435]
- Linear elasticity [69, 110]
- Mesh generation for computer graphics [324]
- Mixed finite element approximations of elliptic PDEs [78, 79, 407]
- Model order reduction for dynamical systems [194, 263, 453]
- Optimal control [36, 37, 56, 57, 369]
- Parameter identification problems [86, 246, 247]

Quite often, saddle point systems arise when a certain quantity (such as the energy of a physical system) has to be minimized, subject to a set of linear constraints. In this case the Lagrange multiplier y usually has a physical interpretation and its computation is also of interest. For example, in incompressible flow problems x is a vector of velocities and y a vector of pressures. In structural mechanics x is the vector of internal forces, y represents the nodal displacements of the structure. For resistive electrical networks y represents the nodal potentials, x being the vector of currents.

In some cases, such as fluid dynamics or linear elasticity, saddle point problems result from the discretization of systems of partial differential equations with constraints. Typically the constraints represent some basic conservation law, such as mass conservation in fluid dynamics. In other cases, such as resistive electrical networks or structural analysis, the equations are discrete to begin with. Now the constraints may correspond to the topology (connectivity) of the system being studied. Because saddle point equations can be derived as equilibrium conditions for a physical system, they are sometimes called *equilibrium equations*. See [449] for a very nice discussion of equilibrium equations throughout applied mathematics. Another popular name for saddle point systems, especially in the optimization literature, is "KKT system," from the Karush-Kuhn-Tucker constraint qualification conditions; see [371, page 328] for precise definitions, and [219, 300] for historical notes.

Systems of the form (1.1)–(1.2) also arise from non-overlapping domain decomposition when interface unknowns are numbered last, as well as from FETI-type schemes when Lagrange multipliers are used to ensure continuity at the interfaces; see for instance [95, 175, 275, 408].

It is of course not possible for us to cover here all these different applications. We choose instead to give some details about three classes of problems leading to saddle point systems. The first comes from the field of computational fluid dynamics, the second from least squares estimation, and the third one from interior point methods in constrained optimization.

2.1. Incompressible flow problems. We begin with the (steady-state) Navier-Stokes equations governing the flow of a Newtonian, incompressible viscous fluid. Let $\Omega \subset \mathbb{R}^d$ $(d = 2, 3)$ be a bounded, connected domain with a piecewise smooth boundary Γ. Given a force field $\mathbf{f} : \Omega \to \mathbb{R}^d$ and boundary data $\mathbf{g} : \Gamma \to \mathbb{R}^d$, the problem is to find a velocity field $\mathbf{u} : \Omega \to \mathbb{R}^d$ and a pressure field $p : \Omega \to \mathbb{R}$ such that

$$-\nu\Delta\mathbf{u} + (\mathbf{u} \cdot \nabla)\,\mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega \tag{2.1}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \tag{2.2}$$

$$\mathcal{B}\mathbf{u} = \mathbf{g} \quad \text{on } \Gamma \tag{2.3}$$

where $\nu > 0$ is the kinematic viscosity coefficient (inversely proportional to the Reynolds number Re), Δ is the Laplace operator in $\mathbb{R}^d$, ∇ denotes the gradient, $\nabla\cdot$ is the divergence, and $\mathcal{B}$ is some type of boundary operator (e.g., a trace operator for Dirichlet boundary conditions). To uniquely determine p we may impose some additional condition, such as

$$\int_\Omega p\, d\,\mathbf{x} = 0.$$

Equation (2.1) represents conservation of momentum, while equation (2.2) represents the incompressibility condition, or mass conservation. Due to the presence of the convective term $(\mathbf{u}\cdot\nabla)\,\mathbf{u}$ in the momentum equations, the Navier-Stokes system is nonlinear. It can be linearized in various ways. An especially popular linearization process is the one based on Picard's iteration; see, e.g., [170, Section 7.2.2]. Starting with an initial guess $\mathbf{u}^{(0)}$ (with $\nabla \cdot \mathbf{u}^{(0)} = 0$) for the velocity field, Picard's iteration constructs a sequence of approximate solutions $(\mathbf{u}^{(k)}, p^{(k)})$ by solving the linear *Oseen problem*

$$-\nu\Delta\mathbf{u}^{(k)} + (\mathbf{u}^{(k-1)} \cdot \nabla)\,\mathbf{u}^{(k)} + \nabla p^{(k)} = \mathbf{f} \quad \text{in } \Omega \tag{2.4}$$

$$\nabla \cdot \mathbf{u}^{(k)} = 0 \quad \text{in } \Omega \tag{2.5}$$

$$\mathcal{B}\mathbf{u}^{(k)} = \mathbf{g} \quad \text{on } \Gamma \tag{2.6}$$

$(k = 1, 2, \ldots)$. Note that no initial pressure needs to be specified. Under certain conditions on ν (which should not be too small) and $\mathbf{f}$ (which should not be too large in an appropriate norm), the steady Navier-Stokes equations (2.1)–(2.3) have a unique solution $(\mathbf{u}_*, p_*)$ and the iterates $(\mathbf{u}^{(k)}, p^{(k)})$ converge to it as $k \to \infty$ for any choice of the initial velocity $\mathbf{u}^{(0)}$. We refer to [212] for existence and uniqueness results and to [292] for a proof of the global convergence of Picard's iteration.

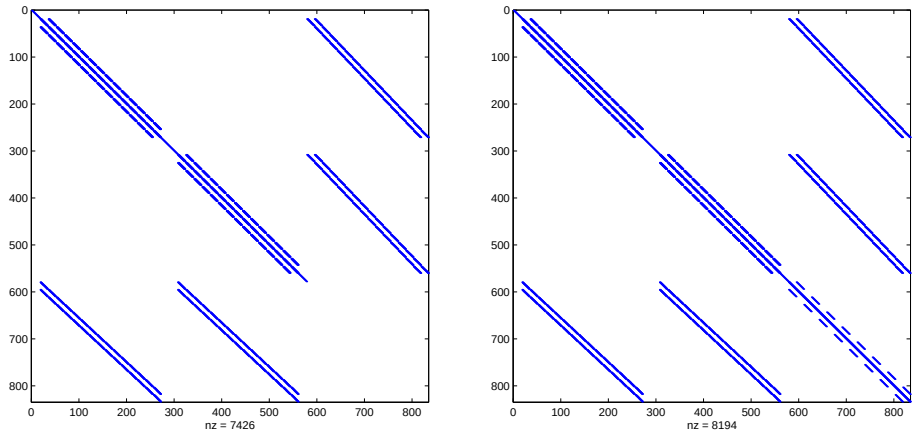
Hence, at each Picard iteration one needs to solve an Oseen problem of the form

$$-\nu\Delta\mathbf{u} + (\mathbf{v} \cdot \nabla)\,\mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega \tag{2.7}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \tag{2.8}$$

$$\mathcal{B}\mathbf{u} = \mathbf{g} \quad \text{on } \Gamma \tag{2.9}$$

with a known, divergence-free coefficient $\mathbf{v}$. Discretization of (2.7)–(2.9) using, e.g., finite differences [393] or finite elements [170, 407] results in a generalized saddle point system of the form (1.6), in which x represents the discrete velocities and y the discrete pressure. Here $A = \operatorname{diag}(A_1, \ldots, A_d)$ is a block diagonal matrix where each block corresponds to a discrete convection-diffusion operator with the appropriate boundary conditions. Note that A is nonsymmetric, but satisfies condition C2 when an appropriate (conservative) discretization is used. The rectangular matrix B^T represents the discrete gradient operator while B represents its adjoint, the (negative) divergence operator. A nonzero C may be present if stabilization is used.



(a) Without stabilization (b) With stabilization

FIG. 2.1. *Sparsity patterns for two-dimensional Stokes problem (leaky lid driven cavity) using Q1-P0 discretization.*

The important special case $\mathbf{v} = \mathbf{0}$ corresponds to the (steady-state) Stokes equations:

$$-\Delta \mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega \tag{2.10}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \tag{2.11}$$

$$\mathcal{B}\mathbf{u} = \mathbf{g} \quad \text{on } \Gamma\,. \tag{2.12}$$

Note that without loss of generality we have set $\nu = 1$, since we can always divide the momentum equation by ν and rescale the pressure p and the forcing term $\mathbf{f}$ by ν. The Stokes equations can be interpreted as the Euler–Lagrange partial differential equations for the constrained variational problem

$$\min \; J(\mathbf{u}) = \frac{1}{2} \int_\Omega ||\nabla \mathbf{u}||_2^2 \, d\,\mathbf{x} - \int_\Omega \mathbf{f} \cdot \mathbf{u} \, d\,\mathbf{x} \tag{2.13}$$

$$\text{subject to } \; \nabla \cdot \mathbf{u} = 0 \tag{2.14}$$

(see, e.g., [239, page 636]). Here the pressure p plays the role of the Lagrange multiplier. The Stokes equations describe the flow of a slow-moving, highly viscous fluid. They also arise as subproblems in the numerical solution of the Navier-Stokes equations by operator splitting methods [214, 407] and as the first step of Picard's iteration when the initial guess used is $\mathbf{u}^{(0)} = \mathbf{0}$.

Appropriate discretization of the Stokes system leads to a symmetric saddle point problem of the form (1.3) where now A is a block diagonal matrix, and each of its d diagonal blocks is a discretization of the Laplace operator $-\Delta$ with the appropriate boundary conditions. Thus, A is now symmetric and positive (semi-)definite. Again, a nonzero C may be present if stabilization is used. Typical sparsity patterns for $\mathcal{A}$ are displayed in Fig. 2.1.

An alternative form of the Oseen system, popularized in recent years in papers by Olshanskii and co-workers [338, 375, 376] (but see also [238, 239]) is the following:

$$-\nu \Delta \mathbf{u} + \mathbf{w} \times \mathbf{u} + \nabla P = \mathbf{f} \quad \text{in } \Omega \tag{2.15}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \tag{2.16}$$

$$\mathcal{B}\mathbf{u} = \mathbf{g} \quad \text{on } \Gamma \tag{2.17}$$

where, for the two-dimensional case:

- $(\mathbf{w}\times) = \begin{pmatrix} 0 & w \\ -w & 0 \end{pmatrix}$
- $w = \nabla \times \mathbf{v} = -\frac{\partial v_1}{\partial x_2} + \frac{\partial v_2}{\partial x_1}$
- $P = p + \frac{1}{2}||\mathbf{v}||_2^2$ (the so-called *Bernoulli pressure*)

Here the divergence-free vector field $\mathbf{v}$ again denotes the approximate velocity from the previous Picard iteration. See [375] for the three-dimensional case. Note that when the "wind" function $\mathbf{v}$ is irrotational ($\nabla \times \mathbf{v} = 0$), equations (2.15)–(2.17) reduce to the Stokes problem. This so-called *rotation form* of the Oseen problem, when discretized, leads to a generalized saddle point system which has certain advantages over the standard (convective) form. We will return on this in section 10.3.

A related problem, also leading to large sparse linear systems in saddle point form upon discretization, is the potential fluid flow problem in porous media, often used to to model groundwater contamination [38, 351]. This consists of a boundary value problem for a system of first order partial differential equations representing, respectively, Darcy's Law for the velocity field $\mathbf{u}$ and the continuity equation:

$$K\mathbf{u} + \nabla p = 0 \quad \text{in } \Omega \tag{2.18}$$

$$\nabla \cdot \mathbf{u} = q \quad \text{in } \Omega \tag{2.19}$$

$$p = p_D \quad \text{on } \Gamma_D, \quad \mathbf{u} \cdot \mathbf{n} = u_N \quad \text{on } \Gamma_N\,, \tag{2.20}$$

where p is a piezometric potential (fluid pressure), K is the symmetric and uniformly positive definite second rank tensor of hydraulic permeability of the medium and q represents density of potential sources (or sinks) in the medium. Here Γ_D and Γ_N are subsets of the boundary Γ of the bounded connected flow domain Ω, with $\Gamma = \bar{\Gamma}_D \cup \bar{\Gamma}_N$, $\Gamma_D \neq \emptyset$, and $\Gamma_D \cap \Gamma_N = \emptyset$; $\mathbf{n}$ is the outward normal vector defined (a.e.) on Γ. When discretized by mixed finite elements (Raviart-Thomas elements being a very popular choice for this problem), a linear system of the type (1.3) is obtained. The symmetric positive definite matrix A is now a discretization of the linear operator "multiplication by K," a zeroth-order differential operator. The conditioning properties of A are independent of the discretization parameter h (for most discretizations), and depend only on properties of the hydraulic permeability tensor K. The matrix $-B$ represents, again, a discrete divergence operator and B^T a discrete gradient. We note that modeling the interaction between surface and subsurface flows leads to coupled (Navier) Stokes and Darcy systems [135, 136]. Problem (2.18)–(2.20) is just one example of a first-order system formulation of a second order linear elliptic PDE [79]. Saddle point sytsems also arise from mixed formulation of fourth order (biharmonic) elliptic problems [216].

In the course of this brief discussion we have restricted ourselves to stationary (steady-state) problems. The unsteady (time-dependent) case leads to sequences of saddle point systems when fully implicit time-stepping schemes are used, for example when the time derivative $\mathbf{u}_t$ is discretized using backward Euler or Crank-Nicolson schemes; see. e.g., [469, Chapter 2]. In the case of Stokes and Oseen, the resulting semi-discrete systems are often referred to as *generalized* Stokes and Oseen problems. The literature on numerical methods for incompressible flow problems is vast; see, e.g., [170, 189, 214, 239, 244, 407, 459, 469, 499].

2.2. Constrained and weighted least squares. Linear systems of saddle point type commonly arise when solving least squares problems. Consider the following least squares problem with linear equality constraints:

$$\min_x \ ||s - Gx||_2 \tag{2.21}$$

$$\text{subject to } \ Bx = g\,, \tag{2.22}$$

where $s \in \mathbb{R}^p$, $G \in \mathbb{R}^{p\times n}$, $x \in \mathbb{R}^n$, $B \in \mathbb{R}^{m\times n}$, $g \in \mathbb{R}^m$ and $m < n$. Problems of this kind arise for instance in curve or surface fitting when the curve is required to interpolate certain data points; see [59, Chapter 5]. Since

$$||s - Gx||_2^2 = (s - Gx)^T(s - Gx) = x^T G^T Gx - 2s^T Gx + s^T s,$$

it is clear that problem (2.21)–(2.22) is equivalent to a constrained quadratic programming problem of the type (1.4)–(1.5) with $A = G^T G$ and $f = G^T s$.

Next we consider the generalized linear least squares problem

$$\min_x (f - Gx)^T W^{-1}(f - Gx) \tag{2.23}$$

where $f \in \mathbb{R}^n$, $G \in \mathbb{R}^{n\times m}$ with $m < n$, and the (scaled) covariance matrix $W \in \mathbb{R}^{n\times n}$ is symmetric positive definite; see [59, Chapter 4]. It is readily seen that (2.23) is equivalent to a standard saddle point system of the form (1.3) with $A = W$, $B = G^T$ and $g = 0$. This saddle point formulation of the generalized least squares problem is often referred to as the *augmented system* formulation. The matrix W is diagonal when the errors in the observations are uncorrelated. When $W = I$ (the $n \times n$ identity matrix) we have the usual linear least squares problem. An advantage of the augmented system formulation over the standard one is that the former allows for the case in which W is singular, which is important in some applications. In other words, W could be positive semidefinite rather than definite. Some applications even lead to an indefinite weight matrix W; see [64, 96, 257].

Finally, if the original problem is ill-posed and Tikhonov regularization is applied, one obtains a saddle point system of the form (1.6) with $C = \gamma L^T L$ where $\gamma > 0$ is the regularization parameter and L is either the $m \times m$ identity or some type of smoothing operator, such as a first-order finite difference operator. See [49, 226] for applications in image processing.

2.3. Saddle point systems from interior point methods. Here we show how saddle point systems arise when interior point methods are used to solve constrained optimization problems. Our presentation is based on the nice synopsis given in [50]. Consider a convex nonlinear programming problem

$$\min \ f(x) \tag{2.24}$$

$$\text{subject to} \ \ c(x) \le 0\,, \tag{2.25}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $c : \mathbb{R}^n \to \mathbb{R}^m$ are convex and twice differentiable. Introducing a nonnegative slack variable $z \in \mathbb{R}^m$, we can write the inequality constraint as the system of equalities $c(x) + z = 0$, and we can introduce the associated barrier problem:

$$\min \ f(x) - \mu \sum_{i=1}^{m} \ln z_i \tag{2.26}$$

$$\text{subject to} \ \ c(x) + z = 0\,. \tag{2.27}$$

The corresponding Lagrangian is

$$\mathcal{L}(x, y, z; \mu) = f(x) + y^T(c(x) + z) - \mu \sum_{i=1}^{m} \ln z_i\,.$$

To find a stationary point of the Lagrangian we set

$$\nabla_x \mathcal{L}(x, y, z; \mu) = \nabla f(x) + \nabla c(x)^T y = 0 \tag{2.28}$$

$$\nabla_y \mathcal{L}(x, y, z; \mu) = c(x) + z = 0 \tag{2.29}$$

$$\nabla_z \mathcal{L}(x, y, z; \mu) = y - \mu Z^{-1} e = 0\,, \tag{2.30}$$

where $Z^{-1} = \operatorname{diag}(z_1^{-1}, z_2^{-2}, \ldots, z_m^{-1})$. Introducing the diagonal matrix $Y = \operatorname{diag}(y_1, y_2, \ldots, y_m)$ and the vector $e = [1\,1\,\ldots\,1]^T$, the first order optimality conditions for the barrier problem become

$$\nabla f(x) + \nabla c(x)^T y = 0 \tag{2.31}$$

$$c(x) + z = 0 \tag{2.32}$$

$$YZe = \mu e \tag{2.33}$$

$$y, z \geq 0\,. \tag{2.34}$$

This is a nonlinear system of equations with nonnegativity constraints and it can be solved by Newton's method. The barrier parameter μ is gradually reduced so as to ensure convergence of the iterates to the optimal solution of problem (2.24)–(2.25). At each Newton iteration, it is necessary to solve a linear system of the form

$$\begin{bmatrix} H(x,y) & B(x)^T & O \\ B(x) & O & I \\ O & Z & Y \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix} = \begin{bmatrix} \nabla f(x) - B(x)^T y \\ -c(x) - z \\ \mu e - YZe \end{bmatrix}, \tag{2.35}$$

where

$$H(x,y) = \nabla^2 f(x) + \sum_{i=1}^{m} y_i \nabla^2 c_i(x) \in \mathbb{R}^{n\times n} \quad \text{and} \quad B(x) = \nabla c(x) \in \mathbb{R}^{m\times n}\,.$$

Here $\nabla^2 f(x)$ denotes the Hessian of f evaluated at x. The linear system (2.35) can be reduced to one of smaller dimensions by using the third equation to eliminate $\delta z = \mu Y^{-1} e - Ze - ZY^{-1}\delta y$ from the second equation. The resulting system is

$$\begin{bmatrix} -H(x,y) & B(x)^T \\ B(x) & ZY^{-1} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \end{bmatrix} = \begin{bmatrix} \nabla f(x) + B(x)^T y \\ -c(x) - \mu Y^{-1} e \end{bmatrix}. \tag{2.36}$$

Apart from the sign, (2.36) is a saddle point system of the form (1.6). If the objective function $f(x)$ and the constraints $c_i(x)$ are convex, the symmetric matrix $H(x,y)$ is positive semidefinite, and it is positive definite if $f(x)$ is strictly convex. The diagonal matrix ZY^{-1} is obiously positive semidefinite. The coefficient matrix in (2.36) depends on the current approximation (x,y) and changes at each Newton step.

Similar linear systems arise when interior point methods are used to solve linear and quadratic programming problems. Now the systems to be solved at each Newton iteration are of the form

$$\begin{bmatrix} -H - D & B^T \\ B & O \end{bmatrix} \begin{bmatrix} \delta x \\ \delta y \end{bmatrix} = \begin{bmatrix} \xi \\ \eta \end{bmatrix},$$

where the $n \times n$ matrix H is symmetric positive semidefinite if the problem is convex and D is a (positive) diagonal scaling matrix. Now H and B remain constant ($H \equiv O$ in linear programming), while D changes at each Newton iteration.

There are many interesting linear algebra problems arising from the use of interior point methods; see in particular [50, 127, 187, 192, 193, 198, 208, 371, 374, 506, 507].

3. Properties of saddle point matrices. This section is devoted to establishing basic algebraic properties of the saddle point matrix $\mathcal{A}$ such as existence of various factorizations, invertibility, spectral properties, and conditioning. Knowledge of these properties is important when developing solution algorithms.

3.1. Block factorizations and the Schur complement. If A is invertible, the saddle point matrix $\mathcal{A}$ admits the following block triangular factorization:

$$\mathcal{A} = \begin{bmatrix} A & B_1^T \\ B_2 & -C \end{bmatrix} = \begin{bmatrix} I & O \\ B_2A^{-1} & I \end{bmatrix} \begin{bmatrix} A & O \\ O & S \end{bmatrix} \begin{bmatrix} I & A^{-1}B_1^T \\ O & I \end{bmatrix}, \tag{3.1}$$

where $S = -(C + B_2A^{-1}B_1^T)$ is the *Schur complement* of A in $\mathcal{A}$. A number of important properties of the saddle point matrix $\mathcal{A}$ can be derived on the basis of (3.1): we do this in the next three subsections.

Also useful are the equivalent factorizations

$$\mathcal{A} = \begin{bmatrix} A & O \\ B_2 & S \end{bmatrix} \begin{bmatrix} I & A^{-1}B_1^T \\ O & I \end{bmatrix} \tag{3.2}$$

and

$$\mathcal{A} = \begin{bmatrix} I & O \\ B_2A^{-1} & I \end{bmatrix} \begin{bmatrix} A & B_1^T \\ O & S \end{bmatrix}. \tag{3.3}$$

The assumption that A is invertible may appear to be rather restrictive, since A is singular in many applications; see, e.g., [246]. However, one can use augmented Lagrangian techniques [191, 215, 219, 237] to replace the original saddle point system with an equivalent one having the same solution but in which the $(1,1)$ block A is now nonsingular. Hence, no great loss of generality is incurred. We shall return to augmented Lagrangian techniques in subsection 3.5 below.

Besides being useful for deriving theoretical properties of saddle point matrices, the decompositions (3.1)–(3.2) are also the basis for many of the most popular solution algorithms for saddle point systems, as we shall see.

3.2. Invertibility conditions. Assuming A is nonsingular, it readily follows from any of the block decompositions (3.1)–(3.2) that $\mathcal{A}$ is nonsingular if and only if S is. Unfortunately, very little can be said in general about the invertibility of the Schur complement $S = -(C + B_2A^{-1}B_1^T)$. It is necessary to place some restrictions on the matrices A, B_1, B_2 and C.

Symmetric case. We begin with the standard saddle point system (1.3), where A is symmetric positive definite, $B_1 = B_2$ and $C = O$. In this case the Schur complement reduces to $S = -BA^{-1}B^T$, a symmetric negative semidefinite matrix. It is obvious that S, and thus $\mathcal{A}$, is invertible if and only if B^T has full column rank (hence, if and only if $\operatorname{rank}(B) = m$), since in this case S is symmetric negative definite. Then both problems (1.3) and (1.4)–(1.5) have a unique solution: if (x_*, y_*) is the solution of (1.3), x_* is the unique solution of (1.4)–(1.5). It can be shown that x_* is the A-orthogonal projection of the solution $\hat{x} = A^{-1}f$ of the unconstrained problem (1.4) onto the constraint set $\mathcal{C} = \{x \in \mathbb{R}^n \,|\, Bx = g\}$. Here A-orthogonal means orthogonal with respect to the inner product $\langle v, w\rangle_A \equiv w^TAv$. We will discuss this in more detail in section 3.3 below.

Next we consider the case where A is symmetric positive definite, $B_1 = B_2 = B$, and $C \neq O$ is symmetric positive semidefinite. Then again $S = -(C + BA^{-1}B^T)$ is symmetric negative semidefinite, and it is negative definite (hence, invertible) if and only if $\ker(C) \cap \ker(B^T) = \{0\}$. Obvious sufficient conditions for invertibility are that C be positive definite or that B^T have full column rank. We can summarize our discussion so far in the following theorem.

THEOREM 3.1. *Assume A is symmetric positive definite, $B_1 = B_2 = B$, and C is symmetric positive semidefinite. If $\ker(C) \cap \ker(B^T) = \{0\}$, then the saddle point matrix $\mathcal{A}$ is nonsingular. In particular, $\mathcal{A}$ is invertible if B has full rank.*

Now we relax the condition that A be positive definite. If A is indefinite, the following simple example shows that $\mathcal{A}$ may be singular, even if B has full rank:

$$\mathcal{A} = \left[\begin{array}{cc|c} 1 & 0 & -1 \\ 0 & -1 & 1 \\ \hline -1 & 1 & 0 \end{array}\right] = \left[\begin{array}{cc} A & B^T \\ B & O \end{array}\right].$$

However, $\mathcal{A}$ will be invertible if A is positive definite on $\ker(B)$. When A is symmetric positive semidefinite, we have the following result (see, e.g., the discussion of quadratic programming in [251] or [339, page 424]). Although this is a well-known result, we include a proof to make our treatment more self-contained.

THEOREM 3.2. *Assume that A is symmetric positive semidefinite, $B_1 = B_2 = B$ has full rank, and $C = O$. Then a necessary and sufficient condition for the saddle point matrix $\mathcal{A}$ to be nonsingular is that* $\ker(A) \cap \ker(B) = \{0\}$.

Proof. Let $u = \left[\begin{array}{c} x \\ y \end{array}\right]$ be such that $\mathcal{A}u = 0$. Hence, $Ax + B^T y = 0$ and $Bx = 0$. It follows that $x^T Ax = -x^T B^T y = (Bx)^T y = 0$. Since A is symmetric positive semidefinite, $x^T Ax = 0$ implies $Ax = 0$ (see [273, page 400]), and therefore $x \in \ker(A) \cap \ker(B)$, thus $x = 0$. Also, $y = 0$ since $B^T y = 0$ and B^T has full column rank. Therefore $u = 0$, and $\mathcal{A}$ is nonsingular. This proves the sufficiency of the condition.

Assume now that $\ker(A) \cap \ker(B) \neq \{0\}$. Taking $x \in \ker(A) \cap \ker(B)$, $x \neq 0$ and letting $u = \left[\begin{array}{c} x \\ 0 \end{array}\right]$ we have $\mathcal{A}u = 0$, implying that $\mathcal{A}$ is singular. Hence, the condition is also necessary. □

REMARK 3.1. *It is clear from the proof of this theorem that the requirement that A be positive semidefinite can be somewhat relaxed: it suffices that A be definite on* $\ker(B)$. *In fact, all we need is that $x^T Ax \neq 0$ for $x \in \ker(B)$, $x \neq 0$. This implies that A is either positive definite or negative definite on* $\ker(B)$. *In any case, the rank of A must be at least $n - m$ for $\mathcal{A}$ to be nonsingular.*

How restrictive is the assumption that B has full rank? A rank deficient B signifies that some of the constraints are redundant. It is generally easy to eliminate this redundancy. For instance, in the Stokes and Oseen case, where B^T represents a discrete gradient, a one-dimensional subspace (containing all the constant vectors) is often present. Hence $\mathcal{A}$ has one zero eigenvalue, corresponding to the so-called *hydrostatic pressure mode*, due to the fact that the pressure is defined up to a constant. A similar situation occurs with electric networks, where y, the vector of nodal potentials, is also defined up to an additive constant. The rank deficiency in B can easily be removed by "grounding" one of the nodes, that is, by specifying the value of the potential (or of the pressure) at one point. One problem with this approach is that the resulting linear system may be rather ill-conditioned; see [63]. Fortunately, since the system $\mathcal{A}u = b$ is consistent by construction, it may not be necessary to remove the singularity of $\mathcal{A}$. Iterative methods like GMRES [424] are largely unaffected by the presence of a single eigenvalue exactly equal to zero, at least when using a zero initial guess, $u_0 = 0$. The reader is referred to [170, Section 8.2.2] for a detailed discussion of this issue in the context of fluid flow problems.

General case. When $C = O$, a *necessary* condition for invertibility is provided by the following theorem, a slight generalization of a similar result for the case $B_1 = B_2$, see [202].

THEOREM 3.3. *If the matrix $\mathcal{A} = \left[\begin{array}{cc} A & B_1^T \\ B_2 & O \end{array}\right]$ is nonsingular, then* $\operatorname{rank}(B_1) = m$ *and*

$\operatorname{rank}\begin{pmatrix} A \\ B_2 \end{pmatrix} = n$.

Proof. If $\operatorname{rank}(B_1) < m$ then there exists a nonzero vector $y \in \mathbb{R}^m$ with $B_1^T y = 0$. Therefore, letting $u = \begin{bmatrix} 0 \\ y \end{bmatrix}$, we get $\mathcal{A}u = 0$, a contradiction.

If $\operatorname{rank}\begin{pmatrix} A \\ B_2 \end{pmatrix} < n$ then there exists a nonzero vector $x \in \mathbb{R}^n$ such that $\begin{pmatrix} A \\ B_2 \end{pmatrix} x = 0$. Letting $u = \begin{bmatrix} x \\ 0 \end{bmatrix}$ we get $\mathcal{A}u = 0$, a contradiction. ∎

It is easy to show that these conditions are not sufficient to ensure the invertibility of $\mathcal{A}$. Some additional conditions are needed. Recall that for any matrix $A \in \mathbb{R}^{n\times n}$ we can write $A = H + K$ where $H = \frac{1}{2}(A + A^T)$ and $K = \frac{1}{2}(A - A^T)$ are the symmetric and skew-symmetric part of A, respectively. The following result provides a necessary and a sufficient condition for $\mathcal{A}$ to be invertible when $B_1 = B_2$.

THEOREM 3.4. *Assume that H, the symmetric part of A, is positive semidefinite, $B_1 = B_2 = B$ have full rank, and C is symmetric positive semidefinite (possibly zero). Then*

(i) $\ker(H) \cap \ker(B) = \{0\} \Rightarrow \mathcal{A}$ *invertible.*
(ii) $\mathcal{A}$ *invertible* $\Rightarrow \ker(A) \cap \ker(B) = \{0\}$.

The converses of (i)-(ii) do not hold in general.

Proof. The proof of part (i) is similar to the sufficiency proof in Theorem 3.1, and can be found in [46, Lemma 1.1]. The proof of part (ii) is exactly as the necessity proof in Theorem 3.1.

Now we show that the converses of (i) and (ii) are false, in general. To see that the converse of (i) is not true in general, consider the matrix

$$\mathcal{A} = \left[\begin{array}{ccc|c} 1 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 \end{array}\right] = \begin{bmatrix} A & B^T \\ B & O \end{bmatrix}.$$

Here we have $\ker(H) \cap \ker(B) = \operatorname{span}([0\,1\,0]^T) \neq \{0\}$ and yet $\mathcal{A}$ is invertible.

To see that the converse of (ii) is not generally true, consider the matrix

$$\mathcal{A} = \left[\begin{array}{cc|c} 0 & -1 & 0 \\ 1 & 1 & 1 \\ \hline 0 & 1 & 0 \end{array}\right] = \begin{bmatrix} A & B^T \\ B & O \end{bmatrix}.$$

This matrix is manifestly singular, yet A is nonsingular and thus $\ker(A) \cap \ker(B) = \{0\}$. ∎

We note that when H is positive semidefinite, the inclusion $\ker(A) \subset \ker(H)$ holds. For if $x \in \ker(A)$, then $Ax = Hx + Kx = 0$ and thus $x^T Ax = x^T Hx = 0$. But since H is symmetric positive semidefinite the last equality implies $Hx = 0$, and therefore $x \in \ker(H)$. That this is a proper inclusion can be seen from the simple example $A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$.

3.3. The inverse of a saddle point matrix. If A is nonsingular, then we know that $\mathcal{A}$ is invertible if and only if $S = -(C + B_2 A^{-1} B_1^T)$ is nonsingular and we have the following explicit

expression for the inverse:

$$\mathcal{A}^{-1} = \begin{bmatrix} A & B_1^T \\ B_2 & -C \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + A^{-1}B_1^T S^{-1} B_2 A^{-1} & -A^{-1}B_1^T S^{-1} \\ -S^{-1}B_2 A^{-1} & S^{-1} \end{bmatrix}. \tag{3.4}$$

If A is singular but C is nonsingular, an analogous expression can be given if we assume that the matrix $A + B_1^T C^{-1} B_2$, the Schur complement of C in $\mathcal{A}$, is nonsingular. However, such expression is of limited interest in the numerical solution of saddle point problems. See [337] for additional expressions for the inverse of a block 2×2 matrix.

An interesting special case arises when A is symmetric positive definite, $B_1 = B_2 = B$, $C = O$, $S = -BA^{-1}B^T$ is nonsingular, and $g = 0$. Then the explicit expression for $\mathcal{A}^{-1}$ shows that the solution (x_*, y_*) of (1.3) is given by

$$\begin{bmatrix} x_* \\ y_* \end{bmatrix} = \begin{bmatrix} (I + A^{-1}B^T S^{-1} B)\, A^{-1} f \\ S^{-1} B A^{-1} f \end{bmatrix}. \tag{3.5}$$

It is easy to see that the matrix $\Pi \equiv -A^{-1}B^T S^{-1} B = A^{-1}B^T (BA^{-1}B^T)^{-1} B$ satisfies $\Pi^2 = \Pi$, i.e., Π is a projector. Moreover, the relations

$$\Pi v \;\in\; \mathrm{ran}(A^{-1}B^T) \quad \text{and} \quad v - \Pi v \;\perp\; \mathrm{ran}(B^T), \quad \text{for all } \; v \in \mathbb{R}^n,$$

show that Π represents an (oblique) projector onto range$(A^{-1}B^T)$ and orthogonal to range(B^T). The first component in (3.5) then can be written as

$$x_* = (I - \Pi)\, \hat{x},$$

where $\hat{x} \equiv A^{-1} f$ is the solution of the unconstrained problem (1.4). Hence x_* is orthogonal to range(B^T). Furthermore, $\hat{x} = \Pi\hat{x} + x_*$, which means that the solution of the unconstrained problem is decomposed into a part that is in range(AB^T) and a part that is orthogonal to range(B^T). By the nature of Π, this decomposition generally is oblique, and only is orthogonal when range$(AB^T) =$ range(B^T) (the latter being true particularly for $A = I$). Next note that since $f - B^T y_* = A x_*$, and $B x_* = 0$,

$$0 \;=\; B x_* \;=\; (BA^{-1})(A x_*) \;=\; (A^{-1}B^T)^T\, (f - B^T y_*). \tag{3.6}$$

By assumption, A^{-1} is symmetric positive definite, so that the function $\langle v, w \rangle_{A^{-1}} \equiv w^T A^{-1} v$ is an inner product. Then (3.6) shows that the vector $f - B^T y_* \in f +$ range(B^T) is orthogonal with respect to the A^{-1}-inner product (A^{-1}-orthogonal) to the space range(B^T). But this means that y_* is the solution of the (generalized) least squares problem $B^T u \approx f$ with respect to the A^{-1}-norm, $\|v\|_{A^{-1}} \equiv (\langle v, v \rangle_{A^{-1}})^{1/2}$, i.e.,

$$\|f - B^T y_*\|_{A^{-1}} \;=\; \min_u \|f - B^T u\|_{A^{-1}}. \tag{3.7}$$

The relation (3.7) is also derived in [41, Section 2]. There it is used to compute the solution of the standard saddle point problem (1.3) with $g = 0$ by first solving the generalized least squares problem with the matrix B^T and the right hand side f (giving y_*), and then computing $x_* = A^{-1}(f - B^T y_*)$. To solve the generalized least squares problem (3.7) numerically, a generalized version of the LSQR method of Paige and Saunders [379] is developed in [41]. The numerical experiments in [41] show that this approach to compute y_* is often superior to applying the MINRES method [378] (see section 9 below for details about this method) to the (symmetric) Schur complement system $S\, y = BA^{-1} f$, which is the second component in (3.5).

Another important special case arises when A and C are both symmetric positive definite and $B_1 = B_2 = B$. Then the corresponding (symmetric) saddle point matrix $\mathcal{A}$ is said to be *quasidefinite*, regardless of what the rank of B may be [477]. These matrices result from interior point methods in constrained optimization; see the discussion in section 2.3. Properties of symmetric quasidefinite matrices have been studied, e.g., in [206, 211, 477]. One of the basic properties is that if $\mathcal{A}$ is quasidefinite, so is $\mathcal{A}^{-1}$ (under the same 2×2 block partitioning), as one can immediately see from (3.4). In [205], this property has been extended to the class of invertible *weakly quasidefinite* matrices, in which A and C are only assumed to be symmetric positive semidefinite.

An alternative expression of the inverse that does not require A to be invertible is the following. Assume that $B_1 = B_2 = B$ has full rank and $C = O$. Denote by $Z \in \mathbb{R}^{n\times(n-m)}$ any matrix whose columns form a basis for $\ker(B)$. If H, the symmetric part of A, is positive semidefinite, then it is easy to see that condition (i) in Theorem 3.4 implies that the $(n-m)\times(n-m)$ matrix Z^TAZ is invertible; indeed, its symmetric part Z^THZ is positive definite. Letting $W = Z(Z^TAZ)^{-1}Z^T$, we have the following expression for the inverse of $\mathcal{A}$:

$$\mathcal{A}^{-1} = \begin{bmatrix} A & B^T \\ B & O \end{bmatrix}^{-1} = \begin{bmatrix} W & (I-WA)B^T(BB^T)^{-1} \\ (BB^T)^{-1}B(I-AW) & -(BB^T)^{-1}B(A-AWA)B^T(BB^T)^{-1} \end{bmatrix}, \tag{3.8}$$

which can be easily proved keeping in mind that $B^T(BB^T)^{-1}B = I - ZZ^T$; see [202].

These explicit expressions for $\mathcal{A}^{-1}$ are of limited practical use, and their interest is primarily theoretical. See, however, [401] for a situation where the inverse of $\mathcal{A}$ is explicitly needed, and for a careful discussion of the problem of updating the inverse of $\mathcal{A}$ when a few of its rows and corresponding columns are modified.

It must be mentioned that in the finite element context the mere nonsingularity of $\mathcal{A}$ is not sufficient to ensure meaningful computed solutions. In order for the discrete problem to be well-posed it is essential that the saddle point matrix remains *uniformly invertible* as h, the meshsize parameter, goes to zero. This means that an appropriate (generalized) condition number of $\mathcal{A}$ remains bounded as $h \to 0$. Sufficient conditions for this to happen include the already mentioned discrete LBB (or inf-sup) conditions; see [79, 111, 370]. Some discussion of conditioning issues from a linear algebra standpoint can be found in section 3.5.

Finally, we point out that *singular* saddle point systems have been studied in [202] and [508]; another relevant reference is [495]. Explicit forms for various generalized inverses of saddle point matrices arising in constrained optimization can be found in [90, Section 3.5]; see also [205].

3.4. Spectral properties of saddle point matrices. In this section we collect a few facts on the spectral properties of saddle point matrices which are relevant when solving the equations by iterative methods. We also introduce an alternative formulation of the saddle point equations leading to a (nonsymmetric) positive definite coefficient matrix.

Eigenvalues: The symmetric case. Assume that A is symmetric positive definite, $B_1 = B_2 = B$ has full rank, and C is symmetric positive semidefinite (possibly zero). Then from (3.1) we obtain

$$\begin{bmatrix} I & O \\ -BA^{-1} & I \end{bmatrix} \begin{bmatrix} A & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} I & -A^{-1}B^T \\ O & I \end{bmatrix} = \begin{bmatrix} A & O \\ O & S \end{bmatrix} \tag{3.9}$$

where $S = -(C + BA^{-1}B^T)$ is symmetric negative definite. Hence $\mathcal{A}$ is congruent to the block diagonal matrix $\begin{bmatrix} A & O \\ O & S \end{bmatrix}$. It follows from Sylvester's Law of Inertia (see [273, page 224]) that

$\mathcal{A}$ is indefinite, with n positive and m negative eigenvalues. The same of course is true if B is rank deficient, as long as S remains negative definite. Clearly, in case S is rank deficient, say $\operatorname{rank}(S) = m - r$, $\mathcal{A}$ has n positive, $m - r$ negative and r zero eigenvalues. A simple limiting argument shows that these result remains true if A is only assumed to be positive semidefinite, provided that the usual condition $\ker(A) \cap \ker(B) = \{0\}$ is satisfied. We refer the reader to [186, 188, 209, 227] for additional results on the inertia of symmetric saddle point matrices under various assumptions on A, B and C. Generally speaking, unless m is very small (which is seldom the case in practice), the matrix $\mathcal{A}$ is *highly indefinite*, in the sense that it has many eigenvalues of both signs.

The following result from [418] establishes eigenvalue bounds for an important class of saddle point matrices.

THEOREM 3.5. *Assume A symmetric positive definite, $B_1 = B_2 = B$ has full rank, and $C = O$. Let μ_1 and μ_n denote the largest and smallest eigenvalue of A (resp.), and let σ_1 and σ_m denote the largest and smallest singular value of B (resp.). Let $\sigma(\mathcal{A})$ denote the spectrum of $\mathcal{A}$. Then*

$$\sigma(\mathcal{A}) \subset I^- \cup I^+,$$

where

$$I^- = \Big[\frac{1}{2}\Big(\mu_n - \sqrt{\mu_n^2 + 4\sigma_1^2}\Big),\, \frac{1}{2}\Big(\mu_1 - \sqrt{\mu_1^2 + 4\sigma_m^2}\Big)\Big]$$

and

$$I^+ = \Big[\mu_n,\, \frac{1}{2}\Big(\mu_1 + \sqrt{\mu_1^2 + 4\sigma_1^2}\Big)\Big].$$

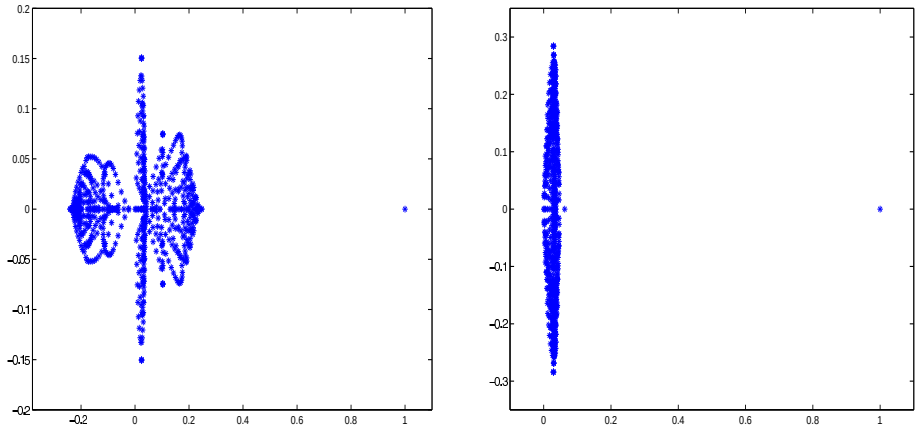
An extension of this result to the case where $C \neq O$ (with C symmetric and positive semidefinite) can be found in [439]. These bounds can be used to obtain estimates for the condition number of $\mathcal{A}$ in specific cases. In turns, these estimates can be used to predict the rate of convergence of iterative methods like MINRES [378]; see [178, 352, 492] and Section 9 below. Eigenvalue bounds are also important when assessing the (inf-sup) stability of mixed finite element discretizations; see, e.g., [58, 346].

Eigenvalues: The general case. In the general case there is not much that can be said about the eigenvalues of $\mathcal{A}$. However, in most cases of interest the convex hull of the eigenvalues of $\mathcal{A}$ contains the origin. If we consider for example the case where $A \neq A^T$, $B_1 = B_2 = B$, and $C = C^T$ (as in the Oseen problem, for example) then we have that the symmetric part of $\mathcal{A}$ is

$$\frac{1}{2}(\mathcal{A} + \mathcal{A}^T) = \begin{bmatrix} H & B^T \\ B & -C \end{bmatrix},$$

where H is the symmetric part of A. If H is positive definite and C positive semidefinite (as in the Oseen problem), then the symmetric part of $\mathcal{A}$ is indefinite and therefore $\mathcal{A}$ has eigenvalues on both sides of the imaginary axis. Fig. 3.1(a) displays the eigenvalues of the discrete Oseen operator obtained from a Q1-P0 finite element approximation of problem (2.7)–(2.9) with $\nu = 0.01$ and $\Omega = [0,1] \times [0,1]$. The matrix was generated using the IFISS software package [170].

Algorithms for solving both standard and generalized eigenvalue problems for saddle point matrices have been studied in the literature, particularly for investigating the stability of incompressible flows and in electromagnetism. See [8, 112, 230, 319, 357].



(a) Indefinite form of $\mathcal{A}$ (b) Positive definite form of $\mathcal{A}$

FIG. 3.1. *Eigenvalues for two-dimensional Oseen problem (leaky lid driven cavity) using Q1-P0 discretization.*

An alternative formulation. Eigenvalue distributions such as that shown in Fig. 3.1(a) are generally considered unfavorable for solution by Krylov subspace methods and indeed, without preconditioning, Krylov subspace methods tend to converge poorly when applied to the corresponding linear system. It has been observed by several authors (e.g., [46, 178, 396, 436], in addition to [213, page 20] and [407, page 304]) that a simple transformation can be used to obtain an equivalent linear system with a coefficient matrix whose spectrum is entirely contained in the half-plane $\Re(z) > 0$. (Here we use $\Re(z)$ and $\Im(z)$ to denote the real and imaginary part of $z \in \mathbb{C}$.) Indeed, assuming that $B_1 = B_2 = B$, we can rewrite the saddle point system in the equivalent form

$$\begin{bmatrix} A & B^T \\ -B & C \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ -g \end{bmatrix}, \quad \text{or} \quad \hat{\mathcal{A}}u = \hat{b}\,. \tag{3.10}$$

Note that $\hat{\mathcal{A}} = \mathcal{J}\mathcal{A}$ where

$$\mathcal{J} = \begin{bmatrix} I_n & O \\ O & -I_m \end{bmatrix} \tag{3.11}$$

and therefore $\hat{\mathcal{A}}$ is nonsingular if $\mathcal{A}$ is. Moreover, we have the following result.

THEOREM 3.6. *Let $\hat{\mathcal{A}}$ be the coefficient matrix in (3.10). Assume $H = \frac{1}{2}(A + A^T)$ is positive semidefinite, $B_1 = B_2 = B$ has full rank, $C = C^T$ is positive semidefinite, and* $\ker(H) \cap \ker(B) = \{0\}$. *Let $\sigma(\hat{\mathcal{A}})$ denote the spectrum of $\hat{\mathcal{A}}$. Then*

(i) $\hat{\mathcal{A}}$ is positive semidefinite, in the sense that $v^T\hat{\mathcal{A}}v \geq 0$ for all $v \in \mathbb{R}^{n+m}$.
(ii) $\hat{\mathcal{A}}$ is positive semistable*, that is, the eigenvalues of $\hat{\mathcal{A}}$ have nonnegative real part: $\Re(\lambda) \geq 0$ for all $\lambda \in \sigma(\hat{\mathcal{A}})$.*
(iii) If in addition $H = \frac{1}{2}(A + A^T)$ is positive definite, then $\hat{\mathcal{A}}$ is positive stable*: $\Re(\lambda) > 0$ for all $\lambda \in \sigma(\hat{\mathcal{A}})$.*

Proof. To prove (i) we observe that for any $v \in \mathbb{R}^{n+m}$ we have $v^T\hat{\mathcal{A}}v = v^T\mathcal{H}v$, where

$$\mathcal{H} \equiv \tfrac{1}{2}(\hat{\mathcal{A}} + \hat{\mathcal{A}}^T) = \begin{bmatrix} H & O \\ O & C \end{bmatrix}$$

is the symmetric part of $\hat{\mathcal{A}}$. Clearly $\mathcal{H}$ is positive semidefinite, hence $v^T\hat{\mathcal{A}}v \geq 0$.

To prove (ii), let (λ, v) be an eigenpair of $\mathcal{A}$, with $||v||_2 = 1$. Then $v^*\hat{\mathcal{A}}v = \lambda$ and $(v^*\hat{\mathcal{A}}v)^* = v^*\hat{\mathcal{A}}^T v = \bar{\lambda}$. Therefore $\frac{1}{2}v^*(\hat{\mathcal{A}} + \hat{\mathcal{A}}^T)v = \frac{\lambda+\bar{\lambda}}{2} = \Re(\lambda)$. To conclude the proof, observe that

$$v^*(\hat{\mathcal{A}} + \hat{\mathcal{A}}^T)v = \Re(v)^T(\hat{\mathcal{A}} + \hat{\mathcal{A}}^T)\Re(v) + \Im(v)^T(\hat{\mathcal{A}} + \hat{\mathcal{A}}^T)\Im(v),$$

a real nonnegative quantity.

To prove (iii), assume (λ, v) is an eigenpair of $\hat{\mathcal{A}}$ with $v = \begin{bmatrix} x \\ y \end{bmatrix}$, then

$$\Re(\lambda) = x^*Hx + y^*Cy = \Re(x)^T H\Re(x) + \Im(x)^T H\Im(x) + \Re(y)^T C\Re(y) + \Im(y)^T C\Im(y).$$

This quantity is nonnegative, and it can be zero only if $x = 0$ (since H is assumed to be positive definite) and $Cy = 0$. But if $x = 0$ then from the first equation in the system $\hat{\mathcal{A}}v = \lambda v$ we get $B^T p = 0$, hence $p = 0$ since B^T has full column rank. Hence $v = 0$, a contradiction. □

Fig. 3.1(b) displays the eigenvalues of the matrix $\hat{\mathcal{A}}$ corresponding to the same Oseen problem as before. As it can be seen, all the eigenvalues lie in the right half-plane. However, such distribution of eigenvalues is not necessarily a more favorable one for Krylov subspace methods, and in fact Krylov subspace methods without preconditioning perform just as poorly as they do on the original problem. We will revisit this topic repeatedly in the course of this survey.

When $A = A^T$ and $C = C^T$ the matrix $\mathcal{A}$ is symmetric indefinite whereas $\hat{\mathcal{A}}$ is nonsymmetric positive (semi-)definite. Moreover, $\hat{\mathcal{A}}$ satisfies

$$\mathcal{J}\,\hat{\mathcal{A}} = \hat{\mathcal{A}}^T\,\mathcal{J}\,, \tag{3.12}$$

with $\mathcal{J}$ defined as in (3.11); that is, $\hat{\mathcal{A}}$ is $\mathcal{J}$-*symmetric* (or *pseudosymmetric*, see [343]). In other words, $\hat{\mathcal{A}}$ is symmetric with respect to the indefinite inner product defined on $\mathbb{R}^{n+m}$ by $[v, w] \equiv w^T\mathcal{J}v$. Conversely, any $\mathcal{J}$-symmetric matrix is of the form $\begin{bmatrix} A & B^T \\ -B & C \end{bmatrix}$ for some $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{m\times n}$, and $C \in \mathbb{R}^{m\times m}$ with $A = A^T$ and $C = C^T$. Note that the set of all $\mathcal{J}$-symmetric matrices

$$\mathbb{J} = \left\{ \begin{bmatrix} A & B^T \\ -B & C \end{bmatrix} \,\middle|\, A = A^T \in \mathbb{R}^{n\times n}\,, B \in \mathbb{R}^{m\times n}\,, C = C^T \in \mathbb{R}^{m\times m} \right\}$$

is closed under matrix addition and under the so-called *Jordan product*, defined as

$$\mathcal{F} \cdot \mathcal{G} \equiv \frac{1}{2}(\mathcal{F}\,\mathcal{G} + \mathcal{G}\,\mathcal{F})\,.$$

The triple $(\mathbb{J}\,, +\,, \cdot)$ is a non-associative, commutative algebra over the reals. It is known as the *Jordan algebra* associated with the real Lie group $O(n, m, \mathbb{R})$ of $\mathcal{J}$-orthogonal (or *pseudo-orthogonal*) matrices, i.e., the group of all matrices $\mathcal{Q} \in \mathbb{R}^{n+m}$ that satisfy the condition $\mathcal{Q}^T\mathcal{J}\,\mathcal{Q} = \mathcal{J}$; see [7] and [343]. The spectral theory of these matrices has been investigated by several authors. A Schur-like decomposition for matrices in $\mathcal{J}$ has been given in [7, Theorem 8], and properties of invariant subspaces of $\mathcal{J}$-symmetric matrices have been studied in [217].

Besides being mathematically appealing, these algebraic properties have implications from the point of view of iterative methods: see for example [197, page 80], where it is shown how $\mathcal{J}$-symmetry can be exploited to develop transpose-free variants of basic Krylov methods. This is of course not enough to justify using the nonsymmetric form $\hat{\mathcal{A}}$ when $\mathcal{A}$ is symmetric, since in this case one may as well use a symmetric Krylov solver on the original (symmetric) formulation; see [177, 178] and

section 9 below. Nevertheless, there are some advantages in using the transformed linear system $\hat{\mathcal{A}}v = \hat{b}$ instead of the original one, especially when certain preconditioners are used; see [46, 436] and section 10.3 below. It can be shown that when A and C are symmetric, at most $2m$ of the $n+m$ eigenvalues of $\hat{\mathcal{A}}$ can have a nonzero imaginary part [441]; furthermore, in some important special cases it turns out that the eigenvalues of $\hat{\mathcal{A}}$ are all real and positive. This implies the existence of a nonstandard inner product on $\mathbb{R}^{n+m}$ with respect to which $\hat{\mathcal{A}}$ is symmetric positive definite, a desirable property from the point of view of iterative methods. The following result gives a sufficient condition (easily checked in many cases) for the eigenvalues of $\hat{\mathcal{A}}$ to be real.

THEOREM 3.7. *Assume that A is symmetric positive definite, $B_1 = B_2 = B$ has full rank, and $C = O$. Let $S = BA^{-1}B^T$, and let μ_n denote the smallest eigenvalue of A. If $\mu_n \geq 4||S||_2$, then all the eigenvalues of the matrix $\hat{\mathcal{A}}$ in (3.10) are real and positive.*

Proof. See [442]. □

We note that the conditions expressed in Theorem 3.7 are satisfied, for instance, for the stationary Stokes problem under a variety of finite differences and finite element discretization schemes.

A more detailed analysis is available in the special situation that $A = \eta I$, where $\eta > 0$ is a positive scaling parameter, $B_1 = B_2 = B$, and $C = O$. Denote the resulting usual saddle point matrix by $\mathcal{A}_\eta^+$, and the alternative formulation with negative (2,1) block by $\mathcal{A}_\eta^-$, i.e.,

$$\mathcal{A}_\eta^\pm = \begin{bmatrix} \eta I & B^T \\ \pm B & O \end{bmatrix}. \tag{3.13}$$

The following theorem characterizes the influence of the choice of $+$ or $-$ as well as η on the eigenvalues of the matrices $\mathcal{A}_\eta^\pm$.

THEOREM 3.8. *Suppose that the matrix B has rank $m - r$, and denote the nonzero singular values of B by $\sigma_1 \leq \cdots \leq \sigma_{m-r}$.*

1. *The $n + m$ eigenvalues of $\mathcal{A}_\eta^+$ in (3.13) are given by*
   - *(i) zero with multiplicity r,*
   - *(ii) η with multiplicity $n - m + r$,*
   - *(iii) $\frac{1}{2}\left(\eta \pm \sqrt{4\sigma_k^2 + \eta^2}\right)$ for $k = 1, \ldots, m - r$.*
2. *Furthermore, if $\sigma_1 \leq \cdots \leq \sigma_t \leq \frac{\eta}{2} < \sigma_{t+1} \leq \cdots \leq \sigma_{m-r}$, then the $n + m$ eigenvalues of $\mathcal{A}_\eta^-$ in (3.13) are given by*
   - *(i) zero with multiplicity r,*
   - *(ii) η with multiplicity $n - m + r$,*
   - *(iii) $\frac{1}{2}\left(\eta \pm \sqrt{\eta^2 - 4\sigma_k^2}\right)$ for $k = 1, \ldots, t$,*
   - *(iv) $\frac{1}{2}\left(\eta \pm i\sqrt{4\sigma_k^2 - \eta^2}\right)$ for $k = t + 1, \ldots, m - r$.*

Proof. See [178, Section 2]. □

This result shows that the eigenvalues of the symmetric indefinite matrix $\mathcal{A}_\eta^+$ (except for the multiple eigenvalues zero and η) always lie in two intervals symmetric about the point $\eta/2$. Changing η only leads to a scaling of these two intervals. On the other hand, the choice of η has a significant effect on the eigenvalues of the nonsymmetric matrix $\mathcal{A}_\eta^-$. For example, if $\eta > 2\sigma_{m-r}$, then all eigenvalues of $\mathcal{A}_\eta^-$ are real, while for $\eta < 2\sigma_1$ all eigenvalues (except zero and η) are purely imaginary. For intermediate values of η, the eigenvalues (except zero and η) form a cross in the complex plane with midpoint $\eta/2$. One is immediately tempted to determine what eigenvalue distribution is the most favorable for the solution by Krylov subspace methods, see [436]. We will discuss this topic in

section 10.1 where the matrices $\mathcal{A}_\eta^\pm$ arise naturally as a result of block diagonal preconditioning of a symmetric saddle point matrix $\mathcal{A}$ with A positive definite and $C = O$.

3.5. Conditioning issues. Saddle point systems that arise in practice can be very poorly conditioned, and care must be taken when developing and applying solution algorithms. It turns out that in some cases the special structure of the saddle point matrix $\mathcal{A}$ can be exploited to avoid or mitigate the effect of ill-conditioning. Moreover, the structure of the right hand side b in (1.1) also plays a role. Indeed, it is frequently the case that either f or g in (1.1) are zero. For instance, $f = 0$ in structural analysis (in the absence of dilation) and in mixed formulations of Poisson's equation, while $g = 0$ in incompressible flow problems and weighted least-squares. So if g (say) is zero, the $(1,2)$ and $(2,2)$ blocks in $\mathcal{A}^{-1}$ (see (3.4) and (3.8)) have no influence on the solution $u = \mathcal{A}^{-1}b$. In particular, any ill-conditioning that may be present in these blocks will not affect the solution, an important fact that should be taken into account when developing robust solution algorithms; see [144, 202, 482].

Let us consider, for the sake of simplicity, a standard saddle point problem where $A = A^T$ is positive definite, $B_1 = B_2 = B$ has full rank, and $C = O$. In this case $\mathcal{A}$ is symmetric and its spectral condition number is given by

$$\kappa(\mathcal{A}) = \frac{\max |\lambda(\mathcal{A})|}{\min |\lambda(\mathcal{A})|}.$$

From Theorem 3.5 one can see that the condition number of $\mathcal{A}$ grows unboundedly as either $\mu_1 = \lambda_{\min}(A)$ or $\sigma_1 = \sigma_{\min}(B)$ goes to zero (assuming that $\lambda_{\max}(A)$ and $\sigma_{\max}(B)$ are kept constant). For mixed finite element formulations of elliptic PDEs, both μ_1 and σ_1 go to zero as h, the mesh size parameter, goes to zero, and the condition number of $\mathcal{A}$ grows like $O(h^{-p})$ for some positive value of p; see, e.g., [352, 492]. This growth of the condition number of $\mathcal{A}$ means that the rate of convergence of most iterative solvers (like Krylov subspace methods) deteriorates as the problem size increases. As discussed in section 10, preconditioning may be used to reduce or even eliminate this dependency on h in many cases. Similar considerations apply to nonsymmetric saddle point problems.

A different type of ill-conditioning is encountered when solving saddle point systems from interior point methods. Consider for instance the case of linear programming, where the $(1,1)$ block A is diagonal. As the iterates generated by the interior point algorithm approach the solution, many of the entries of A tend to zero, and thus $\mathcal{A}$ becomes very ill-conditioned (the constraint matrix B remains constant throughout the iteration process). In particular, the norm of the inverse Schur complement $S^{-1} = (BA^{-1}B^T)^{-1}$ goes to infinity. However, Stewart [447] and Todd [461] (see also [185]) have shown that the norm of the matrices $X = S^{-1}BA^{-1}$ (a *weighted pseudoinverse* of B^T) and B^TX (the associated oblique projector onto the column space of B^T) are bounded by numbers that are independent of A. This important observation has been exploited in a series of paper by Vavasis and collaborators [62, 274, 482, 483] to develop stable algorithms for certain saddle point problems with a severely ill-conditioned A block.

When using direct methods based on triangular factorization, Björck [59, Sections 2.5.3 and 4.4.2] has noted the importance of scaling the $(1,1)$ block A by a positive scalar quantity. Suitable tuning of this scaling factor can be interpreted as a form of preconditioning and has a dramatic impact on the accuracy attainable by sparse direct solvers [11, 144]. On the other hand, such scaling seems to have little or no effect on the convergence behavior of Krylov subspace methods [178].

Another possible approach for dealing with an ill-conditioned or even singular $(1,1)$ block A is the augmented Lagrangian method; see [191, 215, 265, 400] and the more general treatment in [219, 237]. Here we assume that $A = A^T$ (possibly singular), $B_1 = B_2 = B$ has full rank, and

$C = O$. The idea is to replace the saddle point system (1.3) with the equivalent one

$$\begin{bmatrix} A + B^T W B & B^T \\ B & O \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f + B^T W g \\ g \end{bmatrix}. \tag{3.14}$$

The $m \times m$ matrix W, to be suitably determined, is symmetric positive semidefinite. The simplest choice is to take $W = \gamma I$ ($\gamma > 0$). In this case the $(1,1)$ block in (3.14) is nonsingular, and indeed positive definite, provided that A is positive definite on $\ker(B)$. The goal is to choose W so that system (3.14) is easier to solve than the original one, particularly when using iterative methods. When $W = \gamma I$ is used, the choice $\gamma = ||A||_2/||B||_2^2$ has been found to perform well in practice, in the sense that the condition number of both the $(1,1)$ block and of the whole coefficient matrix in (3.14) are approximately minimized. This choice also results in rapid convergence of classical iterative schemes like the method of multipliers; see [219] and section 8.2 below.

The conditioning of equality constrained and weighted least squares problems has been studied in depth by several authors; see [242, 496] and the references therein.

Conditioning properties of quasidefinite matrices and of saddle-point matrices arising from interior-point methods in linear programming have also been investigated in [204, 310]. Finally, we mention that a numerical validation method for verifying the accuracy of approximate solutions of symmetric saddle point problems has been presented in [99].

4. Overview of solution algorithms. Besides the usual (and somewhat simplistic) distinction between direct and iterative methods, solution algorithms for generalized saddle point problems can be subdivided into two broad categories, which we will call *segregated* and *coupled* (or "all at once") methods. Segregated methods compute the two unknown vectors, x and y, separately; in some cases it is x to be computed first, in others it is y. This approach involves the solution of two linear systems of size smaller than $n + m$ (called *reduced systems*), one for each of x and y; in some cases a reduced system for an intermediate quantity is solved. Segregated methods can be either direct or iterative, or involve a combination of the two; for example, one of the reduced systems could be solved by a direct method and the other iteratively. The main representatives of the segregated approach are the Schur complement reduction method, which is based on a block LU factorization of $\mathcal{A}$, and the null space method, which relies on a basis for the null space for the constraints.

Coupled methods, on the other hand, deal with the system (1.1) as a whole, computing x and y (or approximations to them) simultaneously and without making explicit use of reduced systems. These methods include both direct solvers based on triangular factorizations of the global matrix $\mathcal{A}$, and iterative algorithms like Krylov subspace methods applied to the entire system (1.1), typically with some form of preconditioning. As we shall see, preconditioning tends to blur the distinction between direct and iterative solvers, and also that between segregated and coupled schemes. This is because direct solvers may be used to construct preconditioners, and also because preconditioners for coupled iterative schemes are frequently based on segregated methods.

In the next sections we review a number of solution methods, starting with direct solvers and continuing with stationary iterative methods, Krylov subspace solvers, and preconditioners. We also include a brief discussion of multilevel methods, including multigrid and Schwarz-type algorithms. Within each group, we discuss segregated as well as coupled schemes and the interplay between them. It is simply not possible to cover every method that has been described in the literature; instead, we have strived to include, besides all of the "classical" algorithms, those among the more recent methods that appear to be the most widely applicable and effective.

5. Schur complement reduction. Consider the saddle point system (1.1), or

$$Ax + B_1^T y = f, \quad B_2 x - Cy = g.$$

We assume that both A and $\mathcal{A}$ are nonsingular; by (3.1) this implies that $S = -(C + B_2 A^{-1} B_1^T)$ is also nonsingular. Pre-multiplying both sides of the first equation by $B_2 A^{-1}$, we obtain

$$B_2 x + B_2 A^{-1} B_1^T y = B_2 A^{-1} f.$$

Using $B_2 x = g + Cy$ and rearranging, we find

$$(B_2 A^{-1} B_1^T + C)\, y = B_2 A^{-1} f - g\,, \tag{5.1}$$

a reduced system of order m for y involving the (negative) Schur complement $-S = B_2 A^{-1} B_1^T + C$. Note that unless $f = 0$, forming the right-hand side of (5.1) requires solving a linear system with coefficient matrix A.

Once y_* has been computed from (5.1), x_* can be obtained by solving

$$Ax = f - B_1^T y_*\,, \tag{5.2}$$

a reduced system of order n for x involving the (1,1) block, A. Note that this is just block Gaussian elimination applied to (1.1). Indeed, using the block LU factorization (3.3) we get the transformed system

$$\begin{bmatrix} I & O \\ -B_2 A^{-1} & I \end{bmatrix} \begin{bmatrix} A & B_1^T \\ B_2 & -C \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} I & O \\ -B_2 A^{-1} & I \end{bmatrix} \begin{bmatrix} f \\ g \end{bmatrix},$$

that is,

$$\begin{bmatrix} A & B_1^T \\ O & S \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g - B_2 A^{-1} f \end{bmatrix}.$$

Solving this block upper triangular system by block backsubstitution leads to the two reduced systems (5.1) and (5.2) for y and x. These systems can be solved either directly or iteratively. In the important special case where A and $-S$ are symmetric positive definite, highly reliable methods such as Cholesky factorization or the conjugate gradient algorithm can be applied.

The solution strategy outlined above is commonly used in structural mechanics, where it is known as the *displacement* method, since the vector of nodal displacements y is computed first; the reduction to the Schur complement system (5.1) is known as *static condensation*, and the Schur complement itself is called the *assembled stiffness matrix*. In electrical engineering it is known as the *nodal analysis* method, and in optimization as the *range-space* method [482]. In all these applications A is symmetric positive (semi)definite, $B_1 = B_2$, and $C = O$.

This approach is attractive if the order m of the reduced system (5.1) is small and if linear systems with coefficient matrix A can be solved efficiently. The main disadvantages are the need for A to be nonsingular, and the fact that the Schur complement $S = -(BA^{-1}B^T + C)$ may be completely full and too expensive to compute and to factor. Numerical instabilities may also be a concern when forming S, especially when A is ill-conditioned [482]. Dense Schur complements occur in the case of Stokes and Oseen problems, where A corresponds to a (vector) differential operator. Other examples include problems from optimization when B contains one or more dense columns. Note, however, that when B contains no dense columns and A is such that A^{-1} is sparse (e.g., diagonal or block diagonal with small blocks), then S is usually quite sparse. In this case efficient

(graph-based) algorithms can be used to form S, and it is sometimes possible to apply the Schur complement reduction recursively and in a way that preserves sparsity through several levels, in the sense that the number of nonzeros to be stored remains nearly constant throughout the successive reduction steps; see [353] for an example arising from the solution of groundwater flow problems.

In cases where A is positive semidefinite and singular, Schur complement reduction methods may still be applied by making use of augmented Lagrangian techniques (3.14), which replace the original saddle point system with an original one with a nonsingular (1,1) block. If S is too expensive to form or factor, Schur complement reduction can still be applied by solving (5.1) by iterative methods that do not need access to individual entries of S, but only need S in the form of matrix-vector products

$$p = -Sy = (B_2 A^{-1} B^T + C)y\,.$$

The action of S on y can be computed by means of matrix-vector products with B_1^T, B_2 and C and by solving a linear system with coefficient matrix A. If the latter can be performed efficiently and the iteration converges sufficiently fast, this is a viable option. The Schur complement system (5.1), however, may be rather ill-conditioned, in which case preconditioning will be required. Preconditioning the system (5.1) is non-trivial when S is not explicitly available. Some options are discussed in section 10.1 below, in the context of block preconditioners.

6. Null space methods. In this section we assume that $B_1 = B_2 = B$ has full rank and $C = O$. Furthermore, we assume that $\ker(H) \cap \ker(B) = \{0\}$, where H is the symmetric part of A. The saddle point system is then

$$Ax + B^T y = f, \quad Bx = g.$$

The null space method assumes that the following are available:

1. A particular solution $\hat{x}$ of $B\,x = g$;
2. A matrix $Z \in \mathbb{R}^{n\times(n-m)}$ such that $BZ = O$; that is, $\mathrm{range}\,(Z) = \ker(B)$ (the columns of Z span the null space of B).

Then the solution set of $Bx = g$ is described by $x = Zv + \hat{x}$ as v ranges in $\mathbb{R}^{n-m}$. Substituting $x = Zv + \hat{x}$ in

$$Ax + B^T y = f\,,$$

we obtain

$$A(Zv + \hat{x}) = f - B^T y.$$

Pre-multiplying by the full-rank matrix Z^T and using $Z^T B^T = O$ we get

$$Z^T A Z\, v = Z^T (f - A\hat{x})\,, \tag{6.1}$$

a reduced system of order $n - m$ for the auxiliary unknown v (the *dual variable*). This system is nonsingular under our assumptions. Once the solution v_* has been obtained, we set $x_* = Zv_* + \hat{x}$; finally, y_* can be obtained by solving

$$BB^T y = B(f - Ax_*)\,, \tag{6.2}$$

a reduced system of order m with a symmetric positive definite coefficient matrix BB^T. Of course, (6.2) is just the normal equations for the overdetermined system $B^T y = f - Ax_*$, or

$$\min_y ||(f - Ax_*) - B^T y||_2\,.$$

Just as the Schur complement reduction method can be related to expression (3.4) for $\mathcal{A}^{-1}$, the null space method is related to the alternative expression (3.8). It is interesting to observe that when A is invertible, the null space method is just the Schur complement reduction method applied to the *dual* saddle point problem

$$\begin{bmatrix} A^{-1} & Z \\ Z^T & O \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} = \begin{bmatrix} -\hat{x} \\ -Z^T f \end{bmatrix}.$$

This strategy subsumes a whole family of *null space methods*, which differ primarily in the way the matrix Z (often called a *null basis*) is computed; see the discussion below. Null space methods are quite popular in optimization, where they are usually referred to as *reduced* (or *projected*) *Hessian* methods; see [114, 183, 210, 371, 505]. In this setting the matrix A is the $(n \times n)$ Hessian of the function to be minimized subject to the constraint $Bx = g$, and $Z^T AZ$ is the reduced $((n-m) \times (n-m))$ Hessian, obtained by elimination of the constraints. The reduced system (6.2) can also be seen as a projection of the problem onto the constraint set. The null space approach has been extensively used in structural mechanics where it is known under the name of *force* method, since x, the vector of internal forces, is computed first; see, e.g., [53, 260, 289, 290, 395, 414]. Other application areas where the null space approach is used include fluid mechanics (under the name of *dual variable* method, see [6, 12, 13, 14, 245, 253, 429]) and electrical engineering (under the name of *loop analysis*, see [109, 449, 467, 482]).

The null space method has the advantage of not requiring A^{-1}. In fact, the method is applicable even when A is singular, as long as the condition $\ker(H) \cap \ker(B) = \{0\}$ is satisfied. The null space method is often used in applications that require the solution of a sequence of saddle point systems of the type

$$\begin{bmatrix} A_k & B^T \\ B & O \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f_k \\ g_k \end{bmatrix}, \quad k = 1, 2, \ldots,$$

where the A_k submatrix changes with k while B remains fixed. This situations arises, for instance, in the solution of unsteady fluid flow problems, and in the *reanalysis* of structures in computational mechanics; see, e.g., [35, 253, 395]. Another example is the analysis of resistive networks with a fixed connectivity and different values of the resistances. In all these cases the null basis matrix Z needs to be computed only once.

Null space methods are especially attractive when $n-m$ is small. If A is symmetric and positive semidefinite, then $Z^T AZ$ is symmetric positive definite and efficient solvers can be used to solve the reduced system (6.1). If Z is sparse then it may be possible to form and factor $Z^T AZ$ explicitly, otherwise iterative methods must be used, such as conjugate gradients or others.

The method is less attractive if $n - m$ is large, and cannot be applied if $C \neq O$. The main difficulty, however, is represented by the need for a null basis Z for B. We note that computing a particular solution for $Bx = g$ is usually not a difficult problem, and it can be obtained as a byproduct of the computations necessary to obtain Z. In the case where $g = 0$ (arising for instance from the divergence-free condition in incompressible flow problems), the trivial solution $\hat{x} = 0$ will do. Hence, the main issue is the computation of a null basis Z. There are a number of methods that one can use, at least in principle, to this end. In the large and sparse case, graph-based methods invariably play a major role.

Let P denote a permutation matrix chosen so that $BP = \begin{bmatrix} B_b & B_n \end{bmatrix}$, where B_b is $m \times m$ and nonsingular (this is always possible, since B is of rank m). Then it is straightforward to verify that the matrix

$$Z = P \begin{bmatrix} -B_b^{-1} B_n \\ I \end{bmatrix}, \tag{6.3}$$

where I denotes the identity matrix of order $n-m$, is a null basis for B. This approach goes back to Wolfe [505]; a basis of the form (6.3) is called a *fundamental basis*. Quite often, the matrix $B_b^{-1}B_n$ is not formed explicitly; rather, an LU factorization of B_b is computed and used to perform operations involving B_b^{-1}. For instance, if an iterative method like conjugate gradients is used to solve (6.1), then matrix-vector products with Z^TAZ can be performed by means of forward and backsubstitutions with the triangular factors of B_b, in addition to matrix-vector products with B_n, B_n^T, and A.

Since there are in principle many candidate submatrices B_b, (i.e., many permutation matrices P) it is natural to ask whether one can find a matrix B_b with certain desirable properties. Ideally, one would like B_b to be easy to factor, well-conditioned, and to satisfy certain sparsity requirements (either in B_b, or in its factors, or in $B_b^{-1}B_n$.) Another desirable property could be some kind of diagonal dominance. In the literature, this is known as the *nice basis* problem. This is a very difficult problem in general. Consider first the sparsity requirement. Unfortunately, not all sparse matrices admit a sparse null basis. To see this, consider the matrix $B = \begin{bmatrix} I & e \end{bmatrix}$, where e is the column vector all of whose components are equal to 1; clearly, there is no explicit sparse representation for its one-dimensional null space [207]. Moreover, even if a sparse null basis exists, the problem of computing a null basis Z (fundamental or not) with minimal number of nonzero entries has been shown to be NP-hard [115, 397]. In spite of this, there are important situations where a sparse null basis exists and can be explicitly obtained. As we shall see, there may be no need to explicitly factor or invert any submatrix of B.

An example of this is Kirchhoff's classical (1847) method for finding the currents in a resistive electrical network [299]. Our discussion closely follows [449]. For this problem B is just the node-edge incidence matrix of the network, or directed graph, describing the connectivity of the network. More precisely, if the network consists of $m+1$ nodes and n edges, let B_0 be the $(m+1)\times n$ matrix with entries b_{ij} given by -1 if there is a directed edge from node i to node j, and by $+1$ if there is a directed edge from node j to node i; of course, $b_{ij}=0$ if there is no edge connecting nodes i and j. Hence, B_0 behaves as a discrete divergence operator on the network: each column contains precisely two nonzero entries, one equal to $+1$ and the other equal to -1. Matrix B_0 can be shown to be of rank m; note that $B_0^Te=0$, where e denotes the vector with components all equal to 1. A full-rank matrix B can be obtained by dropping the last row of B_0; that is, by "grounding" the last node in the network, see [449, page 112].

A null space for B can be found using Kirchhoff's Voltage Law, which implies that the sum of the voltage drops around each closed loop in the network must be zero. In other words, for current flowing around a loop there is no buildup of charge. In matrix terms, each loop current is a solution to $By=0$. Since B has full rank, there are exactly $n-m$ independent loop currents, denoted by $z_1, z_2, \ldots z_{n-m}$. The loop currents can be determined by a procedure due to Kirchhoff which consists of the following steps:

1. Find a *spanning tree* for the network (graph); this is just a connected subgraph consisting of the m nodes and just $m-1$ edges, so that between any two nodes there is precisely one path, and there are no loops. As shown by Kirchhoff, there are exactly $t=\det BB^T$ spanning trees in the network (which is assumed to be be connected).
2. Once a spanning tree has been picked, the remaining $n-m$ edges can be used to construct the $n-m$ loop currents by noticing that adding any of these edges to the spanning tree will create a loop. For each of these *fundamental loops* we construct the corresponding column z_i of Z by setting the jth entry equal to ± 1 if edge j belongs to the loop, and equal to 0 otherwise; the choice of sign specifies the orientation of the edge.

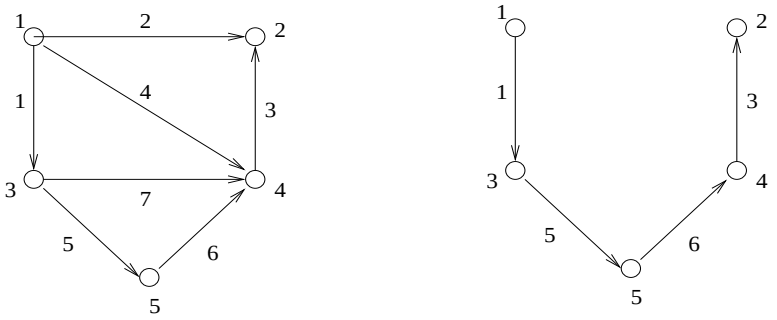


FIG. 6.1. *A directed graph with one of its spanning trees.*

The resulting $Z = [z_1, z_2, \ldots, z_{n-m}]$ is then called an edge-loop matrix, and is a basis for the null space of B. As a simple example, consider the directed graph of Fig. 6.1, with the spanning tree on the right. In this example, $m = 4$ and $n = 7$.

The node-edge incidence matrix B_0 for this graph is

$$B_0 = \begin{bmatrix} -1 & -1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & -1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 \end{bmatrix}.$$

Note that $\text{rank}\,(B_0) = 4$; the matrix B obtained from B_0 by grounding node 5 (i.e., by dropping the last row of B_0) has full row rank, equal to 4. Consider now the spanning tree on the right of Fig. 6.1. By adding the remaining edges (numbered 2, 4 and 7) to the tree in turn we obtain, respectively, the following loops:

1. $1 \to 3 \to 5 \to 4 \to 2 \to 1$, corresponding to the edge sequence (1, 5, 6, 3, -2);
2. $1 \to 3 \to 5 \to 4 \to 1$, corresponding to the edge sequence (1, 5, 6, -4);
3. $1 \to 3 \to 5 \to 4 \to 3$, corresponding to the edge sequence (1, 5, 6, -7).

Note that an edge was given the negative sign whenever its orientation required it. It follows that the edge-loop matrix for the network under consideration is

$$Z = \begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & -1 \end{bmatrix}.$$

and it is straightforward to check that $BZ = O$.

It turns out that this elegant method is fairly general and can be applied to other problems besides the analysis of resistive networks. An important example is fluid dynamics, in particular

the Darcy, Stokes and Oseen problems, where B represents a discrete divergence operator. In this case the null basis Z is called a *solenoidal basis*, since the columns of Z span the subspace of all discrete solenoidal (i.e., divergence-free) functions. In other words, C can be regarded as a discrete curl operator [97]. In the case of finite differences on a regular grid, B is just the incidence matrix of a directed graph associated with the grid, and the cycles (loops) in this graph can be used to construct a sparse Z; see [6, 87, 97, 253, 427, 429]. Note that in this context, solving the system (6.2) for y amounts to solving a Poisson equation. This methodology is not restricted to simple finite difference discretizations or to structured grids; see [4, 12, 13, 14, 254, 428, 430] for applications to a variety of discretization methods on possibly unstructured grids.

We note that no floating point arithmetic is needed to form Z. Furthermore, the sparsity pattern of the matrix Z^TAZ can be easily determined and the matrix Z^TAZ assembled rather cheaply, as long as it is sufficiently sparse. The sparsity will depend on the particular spanning tree used to form Z. Finding a tree that minimizes the number of nonzeros in Z is equivalent to finding the tree for which the sum of all the lengths of the fundamental loops is minimal, which is a NP-hard problem. Nevertheless, many efficient heuristics have been developed; see [458] for the fundamental concepts and algorithms. The relative size of n, m and $n-m$ depends on the discretization scheme used, and on whether the underlying fluid flow problem is posed in 2D or 3D. For lowest order discretizations in 2D, $n-m$ and m are comparable, whereas $n-m$ is much larger than m in 3D or for certain mixed finite element discretizations. If sparse direct solvers are used to solve for the dual variable in (6.2), this makes the null space approach not viable in 3D. In this case iterative solvers must be used, and the spectral properties of Z^TAZ determine the convergence rate. When the matrix Z^TAZ is not formed explicitly, finding appropriate preconditioners for it requires some cleverness. Some work in this direction can be found in [117] and in [367] for constrained optimization problems. See also [425] for closely related work in the context of constrained finite element analyses, and [33, 283, 284] for earlier work on the use of preconditioned conjugate gradients in the context of *implicit null space algorithms*—i.e., null space algorithms in which the matrix Z is not formed explicitly.

For many mixed finite element formulations of second order elliptic problems, A is symmetric positive definite and has condition number bounded independently of the discretization parameter h. In this case fast conjugate gradient convergence can be obtained by using incomplete factorization preconditioners based on Z^TZ, see [4]. Point and block Jacobi preconditioners constructed without explicitly forming Z^TAZ have been tested in the finite element solution of the potential fluid flow problem (2.18)–(2.20) in [13].

Null space methods have been used for a long time in structural optimization. Some relevant references in this area include [94, 262, 294, 398]. In this case B is an equilibrium matrix associated with Newton's Third Law (or Principle of action and reaction). Now B is no longer an incidence matrix, but many of the above described concepts can be extended to this more general situation. Algorithms to find a null basis for B developed by Coleman and Pothen [116] consists of two phases, a combinatorial one and a numerical one. In the first phase, a maximum matching in the bipartite graph of B is used to locate the nonzero entries in the null basis. In the second phase (not needed when B is an incidence matrix), the numerical values of the nonzero entries in the basis are computed by solving certain systems of equations.

When additional structure, such as bandedness, is present in B, it is usually possible to exploit it so as to develop more efficient algorithms. Banded equilibrium matrices often arise in structural engineering. The so-called *turnback algorithm* [465] can be used to compute a banded Z; see also [289], where an interpretation of the turnback algorithm in terms of matrix factorizations is given, and [207] for additional methods motivated by the turnback algorithm. We also mention [395] for approaches based on different graph-theoretic concepts with a focus on parallel implementation

aspects, and [106] for another example of how structure in the constraints can be exploited to find a null basis resulting in a sparse reduced matrix Z^TAZ.

One problem that may occur in the null space method is that a computed null basis matrix Z may be very ill-conditioned, even numerically rank deficient. A way to avoid this problem is to compute a Z with orthogonal columns, which would be optimally conditioned. An orthogonal null basis for B can be computed by means of the QR factorization as follows. Let

$$B^T = Q \begin{bmatrix} R \\ O \end{bmatrix}$$

where Q is $n \times n$ orthogonal and R is $m \times m$, upper triangular and nonsingular. Then the first m columns of Q form an orthonormal basis for range(B^T) and the remaining $n - m$ columns form an orthonormal basis for range$(B^T)^\perp = \ker(B)$. Therefore, if q_i denotes the ith column of Q, then

$$Z = \begin{bmatrix} q_{m+1} & q_{m+2} & \cdots & q_n \end{bmatrix}$$

is the desired matrix. Of course, in the sparse case special ordering techniques must be utilized in order to maintain sparsity in Z [5, 355]. The fact that the columns of Z are orthogonal is advantageous not only from the point of view of conditioning, but also for other reasons. For example, in the computation of thin-plate splines it is often required to solve saddle point systems of the form

$$\begin{bmatrix} A + \rho I & B^T \\ B & O \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}, \tag{6.4}$$

where A is symmetric, $\rho > 0$ is a smoothing parameter, and B has full row rank; see [435]. Usually, problem (6.4) needs to be solved for several values of ρ. If Z is a null basis matrix with orthogonal columns, the coefficient matrix in the reduced system (6.2) is $Z^T(A + \rho I)Z = Z^TAZ + \rho I$. If $n - m$ is so small that a spectral decomposition $Z^TAZ = U\Lambda U^T$ of Z^TAZ can be computed, then for any ρ we have $Z^T(A + \rho I)Z = U(\Lambda + \rho I)U^T$ and reduced linear systems can be solved efficiently.

One further advantage of having an orthogonal null basis is that the reduced system (6.2) is guaranteed to be well-conditioned if A is. For example, if A is symmetric positive definite and has condition number bounded independently of mesh size, the same is true of the Z^TAZ and therefore the conjugate gradient method applied to (6.2) converges in a number of iterations independent of mesh size, even without preconditioning; see [12] for an example from groundwater flow computations. This property fails to hold if Z does not have orthogonal columns, generally speaking; see [13, 391].

Sparse orthogonal schemes have been developed in [52, 207, 260, 290] in the context of structural optimization, and in [9, 12, 14] in the context of mixed-hybrid finite element formulations of the potential fluid flow problem (2.18)–(2.20). A parallel orthogonal null space scheme has been presented in [402] for trajectory optimization problems in quadratic dynamic programming. One limitation of the QR factorization approach is that the null bases obtained by this method are often rather dense compared to those obtained by other sparse schemes; indeed, the sparsest orthogonal null basis may be considerably less sparse than an arbitrary null basis, see [116, 207]. Hence, there is a trade-off between good conditioning properties and sparsity.

Error analyses of various null space methods have been carried out in [121] for dense problems and in [10] for the sparse case. See further [31, 32, 60, 122, 184, 241, 243, 274, 454, 482] for stable implementations and other numerical stability aspects of algorithms for saddle point problems, in particular for equality constrained and weighted least squares problems. Finally, appropriate stopping criteria for the conjugate gradient method applied to the reduced system (6.2) in a finite element context have been given in [12].

7. Coupled direct solvers. In this section we give a brief overview of direct methods based on triangular factorizations of $\mathcal{A}$. Our discussion is limited to the symmetric case ($A = A^T$, $B_1 = B_2$ and $C = C^T$, possibly zero). As far as we know, no specialized direct solver exists for nonsymmetric saddle point problems. Although such problems are often *structurally* symmetric, in the sense that the nonzero pattern of $\mathcal{A}$ is symmetric, some form of numerical pivoting is going to be almost certainly needed for stability reasons; such pivoting pivoting would in turn destroy symmetry. See [145] for a treatment of direct methods for general sparse matrices.

There are several ways to perform Gaussian elimination on a symmetric, possibly indefinite matrix in a way that exploits (and preserves) symmetry. A factorization of the form

$$\mathcal{A} = \mathcal{Q}^T \mathcal{L} \mathcal{D} \mathcal{L}^T \mathcal{Q}\,, \tag{7.1}$$

where $\mathcal{Q}$ is a permutation matrix, $\mathcal{L}$ is unit lower triangular, and $\mathcal{D}$ a block diagonal matrix with blocks of dimension 1 and 2 is usually referred to as an LDL^T factorization. The need for pivot blocks of size 2 is made clear by the following simple example:

$$\mathcal{A} = \left[\begin{array}{cc|c} 0 & 1 & 1 \\ 1 & 0 & 1 \\ \hline 1 & 1 & 0 \end{array}\right],$$

for which selecting pivots from the main diagonal is impossible. Diagonal pivoting may also fail on matrices with a zero-free diagonal due to instabilities. The use of 2×2 pivot blocks dates back to Lagrange (1759). In 1965, W. Kahan (in correspondence with R. de Meersman and L. Schotsmans) suggested that Lagrange's method could be used to devise stable factorizations for symmetric indefinite matrices. The idea was developed by Bunch and Parlett in [85], resulting in a stable algorithm for factoring symmetric indefinite matrices at a cost comparable to that of a Cholesky factorization for positive definite ones. The Bunch-Parlett pivoting strategy is akin to complete pivoting; in subsequent papers [83, 84], alternative pivoting strategies requiring only $O(n^2)$ comparisons for a dense $n \times n$ matrix have been developed; see also [182]. The Bunch-Kaufman pivoting strategy [84] is widely accepted as the algorithm of choice for factoring general symmetric indefinite matrices; several sparse implementations are available, see [146, 150, 151, 332]. In this case the permutation matrix $\mathcal{Q}$ is the result of symmetric row and column interchanges aimed at preserving sparsity in the factors as well as numerical stability. While the Bunch-Kaufman algorithm is normwise backward stable, the resulting factors can have unusual scaling, which may result in a degradation of the accuracy of computed solutions. As reported in [21], such difficulties have been observed in the solution of saddle point systems arising in sparse nonlinear opitimization codes. We refer the reader to [21] for a thorough discussion of such accuracy issues and ways to address these problems; see also [482].

We note that when A is positive definite and B has full rank, the saddle point matrix $\mathcal{A}$ admits an LDL^T factorization with $\mathcal{D}$ diagonal and $\mathcal{Q} = I$ (i.e., no pivoting is needed). Indeed, since A is positive definite it can be decomposed as $A = L_A D_A L_A^T$ with L_A unit lower triangular and D_A diagonal (and positive definite); furthermore the Schur complement $S = -(C + BA^{-1}B^T)$ is negative definite and therefore it can be decomposed as $S = -L_S D_S L_S^T$. Hence, we can write

$$\mathcal{A} = \begin{bmatrix} A & B^T \\ B & -C \end{bmatrix} = \begin{bmatrix} L_A & O \\ L_B & L_S \end{bmatrix} \begin{bmatrix} D_A & O \\ O & -D_S \end{bmatrix} \begin{bmatrix} L_A^T & L_B^T \\ O & L_S^T \end{bmatrix} = \mathcal{L}\,\mathcal{D}\,\mathcal{L}^T\,, \tag{7.2}$$

where $L_B = BL_A^{-T}D_A^{-1}$; note that $BA^{-1}B^T = L_B D_A L_B^T$. In practice, however, the factors will be rather dense with the original ordering, and symmetric permutations have to be used in order to preserve sparsity. Note that L_S and L_B will be completely full if the Schur complement is. However,

not all sparsity-preserving permutations are acceptable. It can be shown that $\mathcal{Q}\mathcal{A}\mathcal{Q}^T$ does not have an LDL^T factorization with $\mathcal{D}$ diagonal for all symmetric permutations $\mathcal{Q}$. Furthermore, some permutations may lead to numerical instability problems.

For many symmetric indefinite codes the factorization consists of two phases, a symbolic and a numeric one. In the symbolic phase, an initial fill-reducing ordering is computed based on the structure of $\mathcal{A}$ only. This is often some variant of minimum degree or nested dissection. In the numeric phase, the actual factorization is computed. Frequently in the course of this phase, the pivot order from the symbolic phase may have to be altered for numerical stability reasons. There are, however, a few exceptions to this rule. An important one is the quasidefinite case discussed in section 3.3, i.e., when C (as well as A) is symmetric positive definite. In this case $\mathcal{Q}\mathcal{A}\mathcal{Q}^T$ always has an LDL^T factorization with $\mathcal{D}$ diagonal, regardless of the choice of $\mathcal{Q}$; see [477]. This is an important result: it suggests that the fill-reducing ordering computed in the symbolic phase of the factorization will not need to be altered in the course of the numeric phase because of stability concerns. Since no pivoting is used in the numeric phase, it is possible exploit all the features of modern supernodal sparse Cholesky factorization codes [368]. The resulting algorithm is more efficient than performing a Bunch-Parlett or Bunch-Kaufman factorization. Numerical stability considerations in [477] suggest that the resulting factorization is usually sufficiently accurate. A stability analysis was given in [211], where the close relationship between $\mathcal{A}$ and its nonsymmetric positive definite form 3.10 was used—together with results in [221]—to derive stability conditions.

A further exception (with $C = O$) has been identified by Tůma in [457]. For a large class of saddle point matrices arising from mixed and hybrid finite element discretizations it is possible to prove the existence of static, fill-reducing pre-orderings $\mathcal{Q}$ such that the permuted matrix $\mathcal{Q}\mathcal{A}\mathcal{Q}^T$ has the LDL^T factorization with $\mathcal{D}$ diagonal. Such pre-orderings are characterized in terms of conditions on the resulting elimination tree. The factorization can be carried out in three phases: a first, symbolic phase in which an initial fill-reducing ordering is computed and the corresponding elimination tree is built; a second phase, also symbolic, where the initial ordering is modified so that the permuted matrix satisfies the conditions that guarantee the existence of the factorization; and a final, numeric phase where the LDL^T factorization itself is computed. The numerical experiments in [457] show that this is an effective approach. As in the quasidefinite case, no numerical stability problems have appeared in practice; however, a formal error analysis has not yet been carried out. We mention that examples of saddle point systems that cause difficulties for symmetric indefinite factorization algorithms have been pointed out in [482].

For the general case, sophisticated strategies for computing sparse LDL^T factorizations with 1×1 and 2×2 pivot blocks have been developed over many years by Duff and Reid together with several collaborators; see [144, 145, 146, 149, 150, 151, 152]. This work has led to a series of widely used codes which are part of the HSL library; see section 12 for information on how to access these codes. The first is MA27, developed in the early 1980s; the second is MA47, a code geared towards symmetric indefinite systems in saddle point form (with $C = O$); later came MA57 and, recently, the MA67 code. All these codes, except for MA67, are multifrontal codes. The need for codes specifically designed for saddle point systems (with $C = O$) is clear when one considers the presence of the zero block in position (2,2). Clearly, any form of symmetric pivoting must be restricted so that pivots are not chosen from the zero block. Failing to do so during the symbolic phase leads to a very large number of pivot order alterations during the numeric factorization phase, dramatically slowing done the computation. Furthermore, the structure of the matrix during the subsequent factorization steps must also be taken into account in order to avoid excessive fill-in. The code MA47 has been designed with both of these goals in mind; see [144, 152] for details. The code MA67, also geared towards saddle point systems, is based on design concepts that are quite different from those of the previous codes; as already mentioned it is not a multifrontal code, and furthermore it does not have

separate symbolic and numeric phases. Instead, the numerical values of the entries are taken into account during the selection of the pivots. A Markowitz-type strategy is used to balance sparsity and numerical stability needs. Unfortunately, the extensive comparison of HSL codes performed in [229] indicates that MA67 is generally inferior to its predecessors.

Other sparse direct solvers for symmetric indefinite systems, based on different design primciples, exist; see for instance the recent reports [359, 431], and section 12 below. While these codes have not been developed specifically for saddle point matrices, they may work quite well on such problems. For instance, the authors of [431] report that their code factors a saddle point matrix from optimization of order approximately 2 millions in less than a minute on a 2.4 GHz Intel 32-bit processor, producing a factor with about 1.4×10^8 nonzeros.

Although fairly reliable in practice, sparse LDL^T factorization methods are not entirely foolproof. Besides the examples given in [482], a few failures to compute acceptably accurate solutions have been reported in [229], even with the use of iterative refinement; see also [431]. Nevertheless, sparse LDL^T methods are the solvers of choice in various sparse optimization codes, where they are often preferred to methods based on Schur complement reduction ("normal equations methods") based both on stability and sparsity considerations. Sparse direct solvers have been less popular in the numerical solution of PDE problems due to their intrinsic storage and computational limitations, although these solvers can be quite competitive for 2D problems; see, e.g., [392]. For saddle point systems arising from PDE problems on 3D meshes, it is necessary to turn to iterative methods.

8. Stationary iterations. We begin our discussion of iterative algorithms with stationary schemes. These methods have been popular for years as "standalone" solvers, but nowadays they are most often used as preconditioners for Krylov subspace methods (equivalently, the convergence of these stationary iterations can be accelerated by Krylov subspace methods.) Another common use for stationary iterations is as smoothers for multigrid methods; we shall return on this in section 11.

8.1. The Arrow-Hurwicz and Uzawa methods. The first iterative schemes for the solution of saddle point problems of a rather general type were the ones developed by the mathematical economists Arrow, Hurwicz and Uzawa [18]. The original papers addressed the case of inequality constraints; see Polyak [396] for an early study of these methods in the context of the equality-constrained problem (1.4)–(1.5).

The Arrow-Hurwicz and Uzawa methods are stationary schemes consisting of simultaneous iterations for both x and y, and can be expressed in terms of splittings of the coefficient matrix $\mathcal{A}$. By elimination of one of the unknown vectors, they can also be interpreted as iterations for the reduced (Schur complement) system. Hence, these algorithms may be regarded both as coupled and as segregated solvers.

We start with Uzawa's method [471], which enjoys considerable popularity in fluid dynamics, especially for solving the (steady) Stokes problem [191, 213, 214, 459, 469]. For simplicity, we assume A is invertible and we describe the algorithm in the case $B_1 = B_2 = B$ and $C = O$. Generalization to problems with $B_1 \neq B_2$ or $C \neq O$ is straightforward. Starting with initial guesses x_0 and y_0, Uzawa's method consists of the following coupled iteration:

$$\begin{cases} x_{k+1} = A^{-1}(f - B^T y_k)\,, \\ y_{k+1} = y_k + \omega(B x_{k+1} - g)\,, \end{cases} \tag{8.1}$$

where $\omega > 0$ is a relaxation parameter. As noted in [220, page 591], (see also [423, page 258]), this

iteration can be written in terms of a matrix splitting $\mathcal{A} = \mathcal{P} - \mathcal{Q}$, i.e., as the fixed-point iteration

$$u_{k+1} = \mathcal{P}^{-1}\mathcal{Q}u_k + \mathcal{P}^{-1}b\,,$$

where

$$\mathcal{P} = \begin{bmatrix} A & O \\ B & -\frac{1}{\omega}I \end{bmatrix}, \quad \mathcal{Q} = \begin{bmatrix} O & -B^T \\ O & -\frac{1}{\omega}I \end{bmatrix}, \quad \text{and} \quad u_k = \begin{bmatrix} x_k \\ y_k \end{bmatrix}. \tag{8.2}$$

Note that the iteration matrix is

$$\mathcal{T} = \mathcal{P}^{-1}\mathcal{Q} = \begin{bmatrix} O & -A^{-1}B^T \\ O & I - \omega BA^{-1}B^T \end{bmatrix},$$

and therefore the eigenvalues of $\mathcal{T}$ are all real (and at least n of them are exactly zero).

On the other hand, if we use the first equation in (8.1) to eliminate x_{k+1} from the second one we obtain

$$y_{k+1} = y_k + \omega\left(BA^{-1}f - g - BA^{-1}B^T y_k\right), \tag{8.3}$$

showing that Uzawa's method is equivalent to a stationary Richardson iteration applied to the Schur complement system

$$BA^{-1}B^T y = BA^{-1}f - g\,. \tag{8.4}$$

If A is symmetric and positive definite, so is $BA^{-1}B^T$. Denoting the smallest and largest eigenvalues of $BA^{-1}B^T$ by $\lambda_{\min}$ and $\lambda_{\max}$, respectively, it is well known that Richardson's iteration (8.3) converges for all ω such that

$$0 < \omega < \frac{2}{\lambda_{\max}}\,,$$

see, e.g., [423, page 114]. Furthermore, the spectral radius of the iteration matrix $I - \omega BA^{-1}B^T$ of (8.3) is minimized by taking

$$\omega_* = \frac{2}{\lambda_{\min} + \lambda_{\max}}\,.$$

In some special cases, the optimal value of ω can be estimated analytically [317]. An important example is the discrete steady-state Stokes system, for which the Schur complement is *spectrally equivalent* to the identity [485]. This means that the eigenvalues of $BA^{-1}A^T$ are bounded below and above by positive constants, i.e., by numbers that do not depend on the mesh size h. As a result, Uzawa's iteration converges at a rate independent of h. We note that this is not the case for the so-called *generalized Stokes* problem, that arises from the solution of the unsteady Stokes problems using implicit methods; see [89]. The convergence of Uzawa's method in this case is rather slow, as it is for most other problems, particularly for the Oseen problem at high Reynolds numbers (small ν) [190]. Improved convergence can be achieved by suitably preconditioning the Uzawa iteration, see [89, 163]. Uzawa-type algorithms for the stabilized case ($C \neq O$) have been first studied in [487]. An Uzawa-type method with variable relaxation parameters has been proposed in [276]. Uzawa's method is still being actively developed by many researchers: recent papers discussing various extensions and improvements of Uzawa's classical algorithm include [54, 75, 93, 98, 125, 277, 333, 344, 372, 512]. Not all applications of Uzawa's method are to fluid flow problems: see [282] for a recent application to image restoration.

The bulk of the computational effort in Uzawa's method is spent in the solution of linear systems with coefficient matrix A. This system can be solved by direct methods or, more often, by an inner iterative scheme. For instance, in the case of the Stokes problem A is a direct sum of discrete Laplace operators, and multigrid methods are a natural choice. The case of inexact inner solves has been studied in detail in [74, 98, 101, 102, 124, 163, 413, 514].

The Arrow-Hurwicz [17] method may be regarded as an inexpensive alternative to Uzawa's method, useful when solution of linear systems with coefficient matrix A is too expensive. Here we follow the derivation given in [423, section 8.4]. By noting that iterate x_{k+1} given by the first of (8.1) is the minimizer of the objective function

$$\phi(x) = \frac{1}{2} x^T A x - x^T (f - B^T y_k)\,,$$

we can derive a less expensive method by taking one step in the direction of the (negative) gradient of $\phi(x)$, with fixed step length α. The resulting method is the Arrow–Hurwicz iteration:

$$\begin{cases} x_{k+1} = x_{k+1} + \alpha(f - Ax_k - B^T y_k)\,, \\ y_{k+1} = y_k + \omega(Bx_{k+1} - g)\,. \end{cases} \tag{8.5}$$

As in the case of Uzawa's method, the Arrow-Hurwicz method can be cast as a fixed-point iteration induced by the splitting

$$\mathcal{A} = \mathcal{P} - \mathcal{Q} \quad \text{where} \quad \mathcal{P} = \begin{bmatrix} \frac{1}{\alpha} I & O \\ B & -\frac{1}{\omega} I \end{bmatrix}, \quad \mathcal{Q} = \begin{bmatrix} \frac{1}{\alpha} I - A & -B^T \\ O & -\frac{1}{\omega} I \end{bmatrix}. \tag{8.6}$$

The convergence of this algorithm depends on the two relaxation parameters, α and ω. Convergence conditions and theoretical estimates for the optimal choice of parameters have been given in [191, 396, 409] and, more recently, in [22, 88]. Because the convergence of the Arrow-Hurwicz method is usually rather slow, various improvements have been proposed, including preconditioned variants of the form

$$\begin{cases} Q_A x_{k+1} = x_{k+1} + \alpha(f - Ax_k - B^T y_k)\,, \\ Q_B y_{k+1} = y_k + \omega(Bx_{k+1} - g)\,, \end{cases} \tag{8.7}$$

where Q_A and Q_B are appropriately chosen "preconditioning matrices," see [22, 409, 413]. Obviously, the line between preconditioned versions of the Arrow-Hurwicz algorithm and inexact/preconditioned variants of Uzawa's method is blurred. Additional hybrids can be obtained by combining the Arrow-Hurwicz and conjugate gradient algorithms: see, e.g., [1, 448].

8.2. Penalty and multiplier methods. Here we assume that $A = A^T$ is positive semidefinite, $B_1 = B_2 = B$ is of full rank, and $C = O$. We further assume that $\ker(A) \cap \ker(B) = \{0\}$, so that the saddle point system has a unique solution. As we have noted in section 1, the saddle point problem is then equivalent to the constrained minimization problem

$$\min \; J(x) = \frac{1}{2} x^T A x - f^T x \tag{8.8}$$

$$\text{subject to} \;\; Bx = g\,. \tag{8.9}$$

A very old method for finding the solution x_* of (8.8)–(8.9) is based on the observation that

$$x_* = \lim_{\gamma \to \infty} x(\gamma)\,,$$

where $x(\gamma)$ is the unique solution of the unconstrained minimization problem

$$\min \ \hat{J}(x) \equiv J(x) + \frac{\gamma}{2} ||Bx - g||_2^2 .$$

In mechanical terms, rigid constraints can be thought of as limiting cases of very large restoring forces, i.e., in this case, forces with potential energy of the form $U(x) = \frac{\gamma}{2}(Bx-g)^T(Bx-g)$; see [120]. The minimizer $x(\gamma)$ can be found by setting the gradient of $J(x) + \frac{\gamma}{2}(Bx-g)^T(Bx-g)$ to zero, leading to the linear system

$$(A + \gamma B^T B)\, x = f + \gamma B^T g \,. \tag{8.10}$$

If we let $y(\gamma) = \gamma^{-1}(Bx(\gamma) - g)$, where $x = x(\gamma)$ is the solution of (8.10), then it is possible to prove that

$$||x_* - x(\gamma)||_2 = O(\gamma^{-1}) \quad \text{and} \quad ||y_* - y(\gamma)||_2 = O(\gamma^{-1}) \quad \text{for} \quad \gamma \to \infty \,,$$

see [213, pages 21-22]. Therefore, provided that γ is taken large enough, $x(\gamma)$ and $y(\gamma)$ are approximate solutions of the original saddle point problem. The penalty method can be thought of as an *approximate* direct method. Since a monotonically increasing sequence $\gamma_1, \gamma_2, \ldots$ of values of γ may be used to compute better and better approximations to x_*, it may also be regarded as a stationary iterative method. In some cases, the choice of γ may be made on the basis of physical considerations: see, e.g., [134, 373].

Since the coefficient matrix $A+\gamma B^T B$ in (8.10) is symmetric and positive definite for $\gamma > 0$, one can in principle use a Cholesky factorization or the conjugate gradient algorithm to compute $x(\gamma)$. Unfortunately, such an approach cannot be recommended, since the condition number of $A+\gamma B^T B$ grows like a (possibly large) multiple of γ; see [213, pages 22-23], and [476] for an analysis of the penalty method applied to equality constrained least squares problems. In practice, for large values of γ the coefficient matrix $A+\gamma B^T B$ is dominated by the (highly singular) term $\gamma B^T B$, and accurate solutions of (8.10) are difficult to obtain.

This drawback of the penalty method can be overcome in two ways. One way is to observe that $(x(\gamma), y(\gamma))$ is the unique solution of the *regularized* saddle point system

$$\begin{bmatrix} A & B^T \\ B & -\varepsilon I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}, \quad \varepsilon = \gamma^{-1} \,. \tag{8.11}$$

Stable solution of this linear system is now possible, for instance using a sparse direct solver; this approach is popular in optimization. However, using a direct solver is not always feasible.

The other option is to modify the penalty method so as to avoid ill-conditioning. This leads to the *method of multipliers*, developed independently by Arrow and Solow [19, page 172], Hestenes [265], and Powell [400]. A further advantage of this method, which combines the use of penalty with Lagrange multipliers, is that it produces the *exact* solution (x_*, y_*) rather than an approximate one. The method of multipliers can be described as follows. Select $\gamma > 0$ and consider the *augmented Lagrangian* function

$$\mathcal{L}(x, y) = J(x) + (Bx - g)^T y + \frac{\gamma}{2} ||Bx - g||_2^2 \,. \tag{8.12}$$

Given an approximation y_k for the Lagrange multiplier vector y, we compute the minimum x_{k+1} of the function $\psi(x) \equiv \mathcal{L}(x, y_k)$. This requires solving the linear system

$$(A + \gamma B^T B)x = f - B^T y_k + \gamma B^T g \,. \tag{8.13}$$

Now we use the computed solution x_{k+1} to obtain the new Lagrange multiplier approximation y_{k+1} according to

$$y_{k+1} = y_k + \gamma\,(Bx_{k+1} - g)\,,$$

and so on. Clearly, the method of multipliers is precisely Uzawa's iteration applied to the saddle point system

$$\begin{bmatrix} A+\gamma B^TB & B^T \\ B & O \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f+\gamma B^Tg \\ g \end{bmatrix}, \tag{8.14}$$

which has exactly the same solution (x_*, y_*) as the original one. Note that the parameter γ does double duty here, in that it appears both in the definition of the augmented Lagrangian and as the relaxation parameter for the Uzawa iteration. As we know from our discussion of Uzawa's method, the iteration converges for $\gamma \in (0, 2/\rho)$ where ρ denotes the largest eigenvalue of the Schur complement $B(A+\gamma B^TB)^{-1}B^T$. This interval becomes unbounded, and the rate of convergence arbitrarily large, as $\gamma \to \infty$. Again, taking too large a value of γ results in extreme ill-conditioning of the coefficient matrix in (8.13). It is necessary to strike a balance between the rate of convergence of the method and the conditioning properties of (8.13). The choice of γ and many other aspects of the multiplier method have been discussed by a number of authors, including [55, 100, 191, 237, 266, 339, 511]. See further [23] for a study of the nonsymmetric case. Another possibility is to combine the augmented Lagrangian method with a (preconditioned) Arrow-Hurwicz scheme; see [191, page 26], and [311] for an application of this idea to problems in structural mechanics.

8.3. Other stationary iterations. In addition to the foregoing algorithms, a number of other stationary iterations based on matrix splittings $\mathcal{A} = \mathcal{P} - \mathcal{Q}$ can be found in the literature. In particular, SOR and block-SOR type schemes have been proposed in [451] for the Stokes problem, in [33, 42, 394] for structural analysis computations, and in [98, 225, 321] for general saddle point systems. Some of these schemes can be interpreted as preconditioned or inexact variants of the classical Uzawa algorithm. Alternating-direction iterative methods for saddle point problems have been studied in [80, 140, 141]. Other stationary iterative methods for saddle point problems have been studied in [30, 46, 156, 224, 464]; since these methods are most often used as preconditioners for Krylov subspace methods, we defer their description to section 10.

9. Krylov subspace methods. In this section we will discuss Krylov subspace methods for solving (preconditioned) saddle point problems. Our goal is not to survey all existing methods and implementations (more complete surveys can be found, e.g., in the monographs [232, 423, 475] or in the papers [157, 197]), but to describe the main properties of the most commonly used methods. We discuss the general theory, the main convergence results, and implementational details. For simplicity, we describe the basics of Krylov subspace methods for the unpreconditioned and nonsingular system (1.1)–(1.2). The following sections will describe the general ideas of preconditioning (section 10), and different preconditioning techniques specifically constructed for (generalized) saddle point systems (subsections 10.1– 10.4).

General Theory. Suppose that u_0 is an initial guess for the solution u of (1.1)–(1.2), and define the initial residual $r_0 = b - \mathcal{A}u_0$. Krylov subspace methods are iterative methods whose kth iterate u_k satisfies

$$u_k \;\in\; u_0 + \mathcal{K}_k(\mathcal{A}, r_0)\,, \quad k = 1, 2, \ldots\,, \tag{9.1}$$

where

$$\mathcal{K}_k(\mathcal{A}, r_0) \;\equiv\; \text{span}\,\{r_0, \mathcal{A}r_0, \ldots, \mathcal{A}^{k-1}r_0\} \tag{9.2}$$

denotes the kth Krylov subspace generated by $\mathcal{A}$ and r_0. It is well known that the Krylov subspaces form a nested sequence that ends with dimension $d \equiv \dim \mathcal{K}_{n+m}(\mathcal{A}, r_0) \leq n+m$, i.e.,

$$\mathcal{K}_1(\mathcal{A}, r_0) \subset \cdots \subset \mathcal{K}_d(\mathcal{A}, r_0) = \cdots = \mathcal{K}_{n+m}(\mathcal{A}, r_0)\,.$$

In particular, for each $k \leq d$, the Krylov subspace $\mathcal{K}_k(\mathcal{A}, r_0)$ has dimension k. Because of the k degrees of freedom in the choice of the the iterate u_k, k constraints are required to make u_k unique. In Krylov subspace methods this is achieved by requiring that the kth residual $r_k = b - \mathcal{A}u_k$ is orthogonal to a k-dimensional space $\mathcal{C}_k$, called the constraints space,

$$r_k \;=\; b - \mathcal{A}u_k \;\in\; r_0 + \mathcal{A}\mathcal{K}_k(\mathcal{A}, r_0)\,, \qquad r_k \;\perp\; \mathcal{C}_k\,. \tag{9.3}$$

Orthogonality here is meant with respect to the Euclidean inner product. The relations (9.1)–(9.3) show that Krylov subspace methods are based on a general type of *projection process* that can be found in many areas of mathematics. For example, in the language of the finite element method, we may consider $\mathcal{K}_k(\mathcal{A}, r_0)$ the test and $\mathcal{C}_k$ the trial space for constructing the approximate solution u_k. In this sense the projection process (9.1)–(9.3) corresponds to the Petrov-Galerkin framework; see, e.g., [407, Chapter 5]. The interpretation of Krylov subspace methods as projection processes has been popularized by Saad in a series of papers in the early 1980s [420, 421]. A survey of his approach can be found in his book [423]. For additional analyses of Krylov subspace methods in terms of projections see, e.g., [34, 157].

Knowing the properties of the system matrix $\mathcal{A}$ it is possible to determine constraint spaces $\mathcal{C}_k$ that lead to uniquely defined iterates u_k, $k = 1, 2, \ldots$, in (9.1)–(9.3). Examples for such spaces are given in the following theorem.

THEOREM 9.1. *Suppose that the Krylov subspace $\mathcal{K}_k(\mathcal{A}, r_0)$ has dimension k. If*

(C) *$\mathcal{A}$ is symmetric positive definite and $\mathcal{C}_k = \mathcal{K}_k(\mathcal{A}, r_0)$, or*
(M) *$\mathcal{A}$ is nonsingular and $\mathcal{C}_k = \mathcal{A}\mathcal{K}_k(\mathcal{A}, r_0)$,*

then there exists a uniquely defined iterate u_k of the form (9.1), for which the residual $r_k = b - \mathcal{A}u_k$ satisfies (9.3).

Proof. See [423, Proposition 5.1]. □

The items (C) and (M) in Theorem 9.1 represent mathematical characterizations of the projection properties of well known Krylov subspace methods. The item (C) characterizes the conjugate gradient (CG) method of Hestenes and Stiefel for symmetric positive definite matrices [267]. Note that in case $\mathcal{A}$ is *not* symmetric positive definite, an approximate solution u_k satisfying both (9.1)–(9.2) and (9.3) with $\mathcal{C}_k = \mathcal{K}_k(\mathcal{A}, r_0)$ may not exist, cf. e.g., [81, 126]. Nevertheless, there are several implementations of this projection process, in particular the full orthogonalization method (FOM) of Saad [420]. Implementations of the projection process characterized by the item (M) are the minimal residual (MINRES) method of Paige and Saunders for nonsingular symmetric (possibly indefinite) matrices [378], and the generalized minimal residual (GMRES) method of Saad and Schultz for general nonsingular matrices [424]. Further mathematically equivalent implementations are discussed in [325].

Numerous other choices of constraint spaces for constructing Krylov subspace methods exist. For example, in case of a nonsymmetric matrix $\mathcal{A}$ one may choose $\mathcal{C}_k = \mathcal{K}_k(\mathcal{A}^T, r_0)$, which represents a generalization of the projection process characterized in the item (C). Specific implementations based on this choice include the method of Lanczos [316] and the biconjugate gradient (BCG) method of Fletcher [181]. However, for a general nonsymmetric matrix $\mathcal{A}$ the process based on $\mathcal{C}_k = \mathcal{K}_k(\mathcal{A}^T, r_0)$ is not well defined, because it may happen that no iterate u_k satisfying both (9.1)–(9.2) and (9.3)

exists. In an actual implementation such as BCG this will lead to a breakdown. In practice such breakdowns are unlikely to occur, but near breakdowns may cause irregular convergence and serious build-up of rounding errors, see [232, Chapter 5] for further discussion. Such instabilities are often overcome by the stabilized BCG (BCGStab) method of van der Vorst [474], which combines the BCG projection principle with an additional minimization step in order to "stabilize" the convergence behavior. It is important to note that although BCGStab is based on BCG, it avoids using the transposed matrix $\mathcal{A}^T$. Closely related is the transpose-free quasi minimal residual method (TFQMR) for general nonsymmetric matrices developed by Freund [195]. While both BCGStab and TFQMR typically produce smoother convergence curves than BCG, none of these methods is guaranteed to be free of breakdowns. A related BCG-like method, which overcomes some of the numerical instabilities of BCG, is the quasi minimal residual (QMR) method for general nonsymmetric matrices of Freund and Nachtigal [199]. We point out that despite the naming similarities, QMR and TFQMR are not mathematically equivalent, and hence may produce completely different approximate solutions even in exact arithmetic. In case of a symmetric (possibly indefinite) matrix $\mathcal{A}$ the Lanczos process underlying the QMR method can be simplified, leading to a mathematically equivalent but numerically more efficient implementation. This is exploited in the symmetric QMR (SQMR) method of Freund and Nachtigal [200].

Additional Krylov subspace methods can be construced by using different search spaces in (9.1). For example, the projection process

$$u_k \in u_0 + \mathcal{A}\mathcal{K}_k(\mathcal{A}, r_0)\,, \qquad r_k \in r_0 + \mathcal{A}\mathcal{K}_k(\mathcal{A}, \mathcal{A}r_0) \perp \mathcal{K}_k(\mathcal{A}, r_0)\,, \tag{9.4}$$

yields a uniquely defined iterate u_k whenever $\mathcal{A}$ is nonsingular and symmetric. An implementation of this mathematical principle is the SYMMLQ method of Paige and Saunders for nonsingular symmetric (possibly indefinite) matrices [378].

Convergence Analysis. In exact arithmetic, the methods that are mathematically described by the items (C) and (M) in Theorem 9.1 as well as by (9.4) for symmetric nonsingular $\mathcal{A}$ terminate with the exact solution in step $d \equiv \dim \mathcal{K}_{n+m}(\mathcal{A}, r_0)$, i.e. they yield $u_d = u$. This feature is called the *finite termination property*. In practice, however, one is typically not interested in computing the exact solution. For example, when the linear system represents a discretized partial differential equation, then an *approximate solution* u_k of the linear system with an error norm on the level of the discretization error is often sufficient. Once this error level is reached, the iterative method can be stopped. In this way iterative methods such as Krylov subspace methods may significantly outperform any direct method which cannot be stopped prematurely, since their intermediate results do not represent approximate solutions. The question is how fast a given Krylov subspace method can reach a given accuracy level.

To analyze this question we use the fact that the geometric *orthogonality condition* expressed in (9.3) is often equivalent to an algebraic *optimality condition* for a certain norm of either the error $e_k = u - u_k$ or the residual $r_k = b - \mathcal{A}u_k$. This optimality condition can be derived using the following well known theorem for best approximations in Hilbert spaces, see e.g. [119, Chapter 1].

THEOREM 9.2. *Suppose that $\mathcal{H}$ is a Hilbert space with inner product $\langle\cdot,\cdot\rangle$ and associated norm $\|\cdot\|$. If $\mathcal{M} \subset \mathcal{H}$ is a closed linear subspace, then for each fixed $h \in \mathcal{H}$ there exists a unique element $m_0 \in \mathcal{M}$ with $\|h - m_0\| = \inf_{m\in\mathcal{M}} \|h - m\|$. Moreover, $h - m_0$ is orthogonal to $\mathcal{M}$ with respect to $\langle\cdot,\cdot\rangle$. Conversely, if $m_0 \in \mathcal{M}$ is such that $h - m_0$ is orthogonal to $\mathcal{M}$ with respect to $\langle\cdot,\cdot\rangle$, then $\|h - m_0\| = \inf_{m\in\mathcal{M}} \|h - m\|$.*

To apply this result in our context, consider a symmetric positive definite matrix $\mathcal{A}$ and the projection process (9.1)–(9.3) with $\mathcal{C}_k = \mathcal{K}_k(\mathcal{A}, r_0)$. According to the item (C) in Theorem 9.1, this process leads to a uniquely defined iterate $u_k \in u_0 + \mathcal{K}_k(\mathcal{A}, r_0)$. The residual $r_k = \mathcal{A}e_k$ is

orthogonal to $\mathcal{K}_k(\mathcal{A}, r_0)$ with respect to the Euclidean inner product, or, equivalently, the error $e_k = u - u_k = (u - u_0) - (u_k - u_0)$ is orthogonal to $\mathcal{K}_k(\mathcal{A}, r_0)$ with respect to the $\mathcal{A}$-inner product defined by $\langle v, w\rangle_{\mathcal{A}} \equiv w^T \mathcal{A} v$. With $\mathbb{R}^{n+m}$, $\mathcal{K}_k(\mathcal{A}, r_0)$, $u - u_0 \in \mathbb{R}^{n+m}$, and $u_k - u_0 \in \mathcal{K}_k(\mathcal{A}, r_0)$ taking the roles of $\mathcal{H}$, $\mathcal{M}$, h, and m_0 in Theorem 9.2, respectively, we see that the orthogonality condition (9.3) is equivalent to

$$\|e_k\|_{\mathcal{A}} \;=\; \min_{z \in u_0 + \mathcal{K}_k(\mathcal{A}, r_0)} \|u - z\|_{\mathcal{A}}\,,$$

where $\|\cdot\|_{\mathcal{A}}$ denotes the $\mathcal{A}$-norm (sometimes called energy norm) associated with the $\mathcal{A}$-inner product. As mentioned above, the item (C) represents a mathematical characterization of the CG method; what we have just derived is its well known optimality property (error minimization in the energy norm). Analogously, we may show that the methods characterized by the item (M) in Theorem 9.1 minimize the Euclidean norm of the residual $r_k = b - \mathcal{A}u_k$ over the affine subspace $u_0 + \mathcal{K}_k(\mathcal{A}, r_0)$. These results are summarized in the following theorem.

THEOREM 9.3. *Suppose that the Krylov subspace $\mathcal{K}_k(\mathcal{A}, r_0)$ in the projection process (9.1)–(9.3) has dimension k. Then the iterate u_k satisfies the following optimality properties:*

(C) *If $\mathcal{A}$ is symmetric positive definite and $\mathcal{C}_k = \mathcal{K}_k(\mathcal{A}, r_0)$, then*

$$\|u - u_k\|_{\mathcal{A}} \;=\; \|e_k\|_{\mathcal{A}} \;=\; \min_{z \in u_0 + \mathcal{K}_k(\mathcal{A}, r_0)} \|u - z\|_{\mathcal{A}} \;=\; \min_{p \in \Pi_k} \|p(\mathcal{A}) e_0\|_{\mathcal{A}}\,. \tag{9.5}$$

(M) *If $\mathcal{A}$ is nonsingular and $\mathcal{C}_k = A\mathcal{K}_k(\mathcal{A}, r_0)$, then*

$$\|b - \mathcal{A}u_k\|_2 \;=\; \|r_k\|_2 \;=\; \min_{z \in u_0 + \mathcal{K}_k(\mathcal{A}, r_0)} \|b - \mathcal{A}z\|_2 \;=\; \min_{p \in \Pi_k} \|p(\mathcal{A}) r_0\|_2\,. \tag{9.6}$$

Here $\|\cdot\|_{\mathcal{A}}$ and $\|\cdot\|_2$ denote the $\mathcal{A}$-norm and the Euclidean norm, respectively, and Π_k denotes the set of polynomials of degree at most k with value one at the origin.

For reasons apparent from (9.6) we will refer in the following to the methods characterized by the item (M) as *minimal residual methods*. Note that the Euclidean residual norm in these methods gives a lower bound for the residual norm of all Krylov subspace methods based on (9.1)–(9.3). Therefore significant research efforts have been made to understand the convergence behavior of these methods. The interpretation of the kth error and residual in (9.5) and (9.6) in terms of the initial error and residual multiplied by a certain polynomial in the matrix $\mathcal{A}$, respectively, is the typical starting point for the convergence analysis of the Krylov subspace methods characterized by the items (C) and (M).

First consider the item (M) (as implemented by the CG method). Since $\mathcal{A}$ is assumed to be symmetric positive definite, it is unitarily diagonalizable, $\mathcal{A} = \mathcal{V}\mathcal{D}\mathcal{V}^T$, with $\mathcal{V}^T\mathcal{V} = I$ and $\mathcal{D} = \mathrm{diag}(\lambda_j)$. The kth $\mathcal{A}$-norm of the error satisfies

$$\begin{aligned}
\|e_k\|_{\mathcal{A}} &= \min_{p \in \Pi_k} \|p(\mathcal{A}) e_0\|_{\mathcal{A}} \;=\; \min_{p \in \Pi_k} \|\mathcal{A}^{1/2} p(\mathcal{A}) e_0\|_2 \;=\; \min_{p \in \Pi_k} \|p(\mathcal{A}) \mathcal{A}^{1/2} e_0\|_2 \\
&\le \min_{p \in \Pi_k} \|p(\mathcal{A})\|_2 \, \|\mathcal{A}^{1/2} e_0\|_2 \\
&= \|e_0\|_{\mathcal{A}} \min_{p \in \Pi_k} \|p(\mathcal{D})\|_2 \\
&= \|e_0\|_{\mathcal{A}} \min_{p \in \Pi_k} \max_{\lambda_j} |p(\lambda_j)|\,. \qquad (9.7)
\end{aligned}$$

Hence $\|e_k\|_{\mathcal{A}} / \|e_0\|_{\mathcal{A}}$, the kth relative $\mathcal{A}$-norm of the error, is bounded by the value of a polynomial approximation problem on the eigenvalues of $\mathcal{A}$. This bound is sharp in the sense that for each

(symmetric positive definite) matrix $\mathcal{A}$ and each iteration step k there exists an initial error $e_0^{(k)}$ for which equality holds, see [231]. Consequently, the *worst-case* behavior of the CG method is completely determined by the eigenvalues of the coefficient matrix. An immediate question is how small a kth degree polynomial with value one at the origin can become on a given set of matrix eigenvalues. While the polynomial that solves the min-max problem is explicitly known, see [231] and [329] for different derivations, no simple expression for the min-max value itself exists. The sharp bound (9.7), however, provides some intuition of how the eigenvalue distribution influences the worst-case convergence behavior. For example, if all eigenvalues are tightly clustered around a *single* point that is far away from the origin, one may expect fast convergence. Widely spread eigenvalues, on the other hand, will potentially lead to slow convergence. The standard approach for estimating the right hand side of (9.7) is to replace the min-max problem on the discrete set of eigenvalues by a min-max approximation problem on its convex hull (i.e., on an interval from the smallest eigenvalue $\lambda_{\min}$ to the largest eigenvalue $\lambda_{\max}$ of $\mathcal{A}$). The latter is solved by scaled and shifted Chebyshev polynomials of the first kind, giving the well known bound

$$\min_{p\in\Pi_k} \max_{\lambda_j} |p(\lambda_j)| \;\le\; 2\left(\frac{\sqrt{\kappa(\mathcal{A})}-1}{\sqrt{\kappa(\mathcal{A})}+1}\right)^k, \qquad \text{where } \kappa(\mathcal{A}) \;=\; \frac{\lambda_{\max}}{\lambda_{\min}}\,, \tag{9.8}$$

see, e.g., [232, Theorem 3.1.1]. The bounds (9.7)–(9.8) show that a small condition number of $\mathcal{A}$ is *sufficient* (but not necessary) for a fast decrease of the relative $\mathcal{A}$-norm of the error. This fact motivates the classical goal of *preconditioning*, which is to modify the given linear system in order to reduce the condition number of the system matrix. However, while (9.7) is sharp, the right hand side of (9.8) often overestimates the left hand side and thus the worst-case relative $\mathcal{A}$-norm of the error. Moreover, the actual behavior for a specific right hand side vector b depends not only on the eigenvalue distribution, but also on the coefficients of b in the eigenvectors of $\mathcal{A}$. Several case studies on model problems have recently been performed, see, e.g., [39, 40, 328, 366] for more details.

Next consider the minimal residual methods. The resulting projection process (9.1)–(9.3) is well defined for each nonsingular matrix $\mathcal{A}$. For simplicity, suppose that $\mathcal{A}$ is diagonalizable, $\mathcal{A} = \mathcal{X}\mathcal{D}\mathcal{X}^{-1}$, with $\mathcal{D} = \mathrm{diag}(\lambda_j)$. Then the kth Euclidean norm of the residual satisfies

$$\begin{aligned}
\|r_k\|_2 &= \min_{p\in\Pi_k} \|p(\mathcal{A})r_0\|_2 \\
&\le \min_{p\in\Pi_k} \|\mathcal{X}p(\mathcal{D})\mathcal{X}^{-1}\|_2\, \|r_0\|_2 \\
&\le \|\mathcal{X}\|_2 \min_{p\in\Pi_k} \|p(\mathcal{D})\|_2\, \|\mathcal{X}^{-1}\|_2\, \|r_0\|_2 \\
&= \|r_0\|_2\, \kappa(\mathcal{X}) \min_{p\in\Pi_k} \max_{\lambda_j} |p(\lambda_j)|\,. \qquad (9.9)
\end{aligned}$$

Apparently, $\|r_k\|_2/\|r_0\|_2$, the kth relative Euclidean residual norm, is bounded by the value of the same type of polynomial approximation problem as in (9.7), multiplied by the condition number of the eigenvector matrix $\mathcal{X}$ of $\mathcal{A}$.

If $\mathcal{A}$ is *normal*, then $\kappa(\mathcal{X}) = 1$, and it can be shown that the bound (9.9) is sharp in the same sense as the bound (9.7), see [234, 286]. In this case the same intuition as described above for the worst-case behavior of CG also applies to the worst-case behavior of minimal residual methods. In particular, a single eigenvalue cluster far away from the origin implies fast convergence (here measured by the relative Euclidean residual norm). Additionally, (9.8) can be used if $\mathcal{A}$ is symmetric positive definite, which shows that in this case a small condition number of $\mathcal{A}$ is *sufficient* (but not necessary) for fast convergence. As in the case of CG, reducing the condition number of a symmetric positive definite system matrix $\mathcal{A}$ also represents a reasonable goal of preconditioning for minimal residual methods.

In case of a nonsingular symmetric indefinite matrix $\mathcal{A}$, the min-max approximation problem on the matrix eigenvalues in (9.9) cannot be replaced by the min-max problem on their convex hull, as eigenvalues lie on both sides of the origin. Here one may replace the discrete set of eigenvalues by the union of two intervals containing all of them and excluding the origin, say $I^- \cup I^+ \equiv [\lambda_{\min}, \lambda_s] \cup [\lambda_{s+1}, \lambda_{\max}]$ with $\lambda_{\min} \leq \lambda_s < 0 < \lambda_{s+1} \leq \lambda_{\max}$.

When both intervals are of the same length, i.e. $\lambda_{\max} - \lambda_{s+1} = \lambda_s - \lambda_{\min}$, the solution of the corresponding min-max approximation problem

$$\min_{p \in \Pi_k} \max_{\lambda \in I^- \cup I^+} |p(\lambda)|\,, \tag{9.10}$$

is characterized by a result of de Boor and Rice [130]. This leads to the bound

$$\min_{p \in \Pi_k} \max_{\lambda_j} |p(\lambda_j)| \;\leq\; 2 \left(\frac{\sqrt{|\lambda_{\min}\lambda_{\max}|} - \sqrt{|\lambda_s \lambda_{s+1}|}}{\sqrt{|\lambda_{\min}\lambda_{\max}|} + \sqrt{|\lambda_s \lambda_{s+1}|}} \right)^{[k/2]}, \tag{9.11}$$

where $[k/2]$ denotes the integer part of $k/2$, see [232, Chapter 3]. For an illustration of this bound suppose that $|\lambda_{\min}| = \lambda_{\max} = 1$ and $|\lambda_s| = \lambda_{s+1}$. Then $\kappa(\mathcal{A}) = \lambda_{s+1}^{-1}$, and the right hand side of (9.11) reduces to

$$2 \left(\frac{1/\lambda_{s+1} - 1}{1/\lambda_{s+1} + 1} \right)^{[k/2]}. \tag{9.12}$$

Note that (9.12) corresponds to the value of right hand side of (9.8) at step $[k/2]$ for a symmetric positive definite matrix having all its eigenvalues in the interval $[\lambda_{s+1}^2, 1]$, and thus a condition number of λ_{s+1}^{-2}. Hence the convergence bound for an indefinite matrix with condition number κ needs twice as many steps to decrease to the value of the bound for a definite matrix with condition number $\sqrt{\kappa}$. Although neither of the two bounds is sharp, this clearly indicates that solving indefinite problems represents a significant challenge.

In the general case when the two intervals I^- and I^+ are not of the same length, the explicit solution of (9.10) becomes quite complicated, see, e.g., [176, Chapter 3], and no simple and explicit bound on the min-max value is known. One may of course extend the smaller interval to match the length of the larger one, and still apply (9.11). But this usually results in a significantly weaker convergence bound, which fails to give relevant information about the actual convergence behavior. An alternative is to consider the asymptotic behavior of the min-max value (9.10), and in particular the *asymptotic convergence factor*

$$\rho(I^- \cup I^+) \;\equiv\; \lim_{k \to \infty} \left(\min_{p \in \Pi_k} \max_{\lambda \in I^- \cup I^+} |p(\lambda)| \right)^{1/k}. \tag{9.13}$$

Obviously, $\rho(I^- \cup I^+)$ may be estimated even if the value (9.10) for each step k is unknown. Asymptotic convergence results are common in the theory of semiiterative methods [158, 159] and of classical iterative methods such as SOR [479]. Because of the finite termination property, asymptotic convergence results for Krylov subspace methods have to be put into proper perspective. They certainly can be useful in the convergence analysis of minimal residual methods for sequences of linear systems of growing dimension, e.g. when studying the dependence of the convergence behavior on the mesh size in a discretized differential equation. An example will be discussed in section 10.1 below.

If $\mathcal{A}$ is *nonnormal*, then $\kappa(\mathcal{X})$ may be very large, and (9.9) may be a very large overestimate even of the worst-case relative residual norm of minimal residual methods. In particular, the matrix

eigenvalues may in the nonnormal case give misleading information about the convergence behavior. In fact, it has been shown that any nonincreasing convergence curve of relative residual norms can be obtained by a (nonnormal) matrix $\mathcal{A}$ having any prescribed set of eigenvalues [235]. While the examples constructed by the theory in [235] may be artificial, misleading eigenvalue information was also demonstrated and analyzed in the more practical context of discretized convection-diffusion problems, see [171, 327] for further details. Several other sets associated with the matrix $\mathcal{A}$ have been used in the convergence analysis for nonnormal problems, among them the field of values [171, 446], pseudospectra [365], and the polynomial numerical hull [233]. In the context of saddle point problems, the field of values seems to be quite useful. In particular, rate of convergence estimates obtained from this set for preconditioned saddle point systems arising from mixed finite element discretizations of PDEs are sometimes optimal in the sense that they are independent of the mesh size parameter; see, e.g., [305, 336] for details. However, the first inequality in (9.9) has been shown to be strict for some nonnormal matrices $\mathcal{A}$, see [173, 462]. Hence no convergence analysis based solely on the matrix $\mathcal{A}$ can in the nonnormal case give a sharp bound on the worst-case residual norm, so that the initial residual should be included in the convergence analysis whenever possible. This is sometimes only possible in the context of the specific application, see [327] for an example. An important and often overlooked fact is that the convergence of minimal residual methods is *not* slower for nonnormal than for normal matrices. In particular, it has been shown that for each nonnormal matrix there exists a normal matrix for which the same convergence behavior can be observed (for the same initial residual), see [236]. Yet, as described above, the convergence behavior in the nonnormal case is significantly more difficult to analyze than in the normal case. Sharp convergence results usually can only be obtained by considering the specific properties of $\mathcal{A}$ (e.g., its eigenvalue-eigenvector structure) in relation to the given initial residual.

Implementational Details. When implementing Krylov subspace methods one needs to generate (at least implicitly) a basis of the Krylov subspace $\mathcal{K}_k(\mathcal{A}, r_0)$. For reasons of numerical stability this basis should preferably be *orthogonal*, and for reasons of computational efficiency, the basis should be generated by a *short-term recurrence*.

In case of a *symmetric* matrix $\mathcal{A}$, these two goals are achieved simultaneously by the symmetric Lanczos method [316], which generates an orthogonal basis $v_1, \ldots, v_k$ of $\mathcal{K}_k(\mathcal{A}, r_0)$ using a three-term recurrence, meaning that in step k only the vectors v_{k-1} and v_k are required to compute v_{k+1}. This highly efficient method is the basic ingredient of many Krylov subspace methods for symmetric matrices, among them CG, MINRES and SYMMLQ. Symmetry of the system matrix (even positive definiteness in case of CG) turns out to be a strong restriction in the case of generalized saddle point systems (1.1)–(1.2). First of all, $\mathcal{A}$ needs to be symmetric, which requires $A = A^T$, $C = C^T$, and $B_1 = B_2$. These conditions are satisfied in many applications, see Section 2. However, in case the system is preconditioned, the preconditioned system matrix needs to be symmetric (positive definite) as well. This in general requires that the preconditioner $\mathcal{P}$ be symmetric positive definite. For example, consider left preconditioning by $\mathcal{P}$, i.e., the preconditioned system

$$\mathcal{P}^{-1}\,\mathcal{A}\,u \;=\; \mathcal{P}^{-1}\,b\,.$$

If $\mathcal{P}$ is symmetric positive definite, its Cholesky decomposition exists, $\mathcal{P} = \mathcal{L}\,\mathcal{L}^T$, and hence the preconditioned system is equivalent to

$$(\mathcal{L}^{-1}\,\mathcal{A}\,\mathcal{L}^{-T})\,(\mathcal{L}^T u) \;=\; \mathcal{L}^{-1}\,b\,.$$

The system matrix $\mathcal{L}^{-1}\mathcal{A}\mathcal{L}^{-T}$ is again symmetric (positive definite), and one can apply the same method to solve the unpreconditioned as well as the preconditioned system. If $\mathcal{P}$ is symmetric but *not* positive definite, then the preconditioned system matrix $\mathcal{P}^{-1}\mathcal{A}$ is in general nonsymmetric, regardless of $\mathcal{A}$ being symmetric or not (unless $\mathcal{P}$ and $\mathcal{A}$ commute, e.g., when $\mathcal{P}$ is a polynomial

Method	Required $\mathcal{A}$	Type	Recurrence	Required $\mathcal{P}$
CG	symm. def.	optimal	three-term	symm. def.
MINRES/SYMMLQ	symm. indef.	optimal	three-term	symm. def.
SQMR	symm. indef.	nonoptimal	three-term	symm.
GMRES	general	optimal	full	general
QMR/BCGStab/TFQMR	general	nonoptimal	three-term	general

TABLE 9.1
Summary of Krylov subspace methods discussed in Section 9.

in $\mathcal{A}$). In case that no good symmetric positive definite preconditioner is known, or if a very good nonsymmetric preconditioner is available, the possible advantage of $\mathcal{A}$ being symmetric is lost, and one usually has to use a solution method for nonsymmetric matrices. See [440] and Section 10 below.

In case of a general, *nonsymmetric* matrix $\mathcal{A}$, the two goals mentioned above cannot be achieved at once (with few exceptions possessing a particular structure, see [174, 326]). Here one has to choose between a full recurrence and an orthogonal Krylov subspace basis, or a short-term recurrence and a nonorthogonal basis. The former approach, implemented by the Arnoldi method [16], is used in the GMRES algorithm. The latter approach is implemented in the nonsymmetric (or two-sided) Lanczos method [316], which forms a main ingredient of methods like BCG, BCGStab, and QMR. Methods based on orthogonal bases are sometimes called optimal methods, while the other class is referred to as nonoptimal methods. The nonoptimal methods also include truncated and restarted versions of the optimal methods. Recent research shows that when an optimal method such as GMRES converges quickly, the related nonoptimal methods based on (9.1)–(9.2) often also converge quickly [445]. The nonsymmetric Lanczos method is also used in SQMR, which represents a nonoptimal method for symmetric indefinite matrices. However, unlike the optimal methods for this class of problems discussed above (in particular MINRES and SYMMLQ), the SQMR method does not require a symmetric positive definite preconditioner, since its underlying Lanczos method works for nonsymmetric matrices as well. A summary of the methods discussed in this section is given in Table 9.1.

10. Preconditioners. The use of preconditioning has been already referred to several times in previous sections. As we saw, preconditioning may be used in the iterative solution of reduced systems arising from Schur complement reduction or from the application of various null space methods. In this section we discuss preconditioners in greater detail, and provide a description of some of the most widely used or promising techniques. In view of the fact that preconditioning has been and remains a most active area of research, accounting for the vast majority of papers on the numerical solution of saddle point problems (more generally, on linear solvers) in the last several years, a completely exhaustive survey is impossible. In some cases we limit ourselves to very brief notes and to pointers to the literature.

As is well known, the term *preconditioning* refers to transforming the linear system $\mathcal{A}u = b$ into another system with more favorable properties for iterative solution. A *preconditioner* is a matrix $\mathcal{P}$ (or $\mathcal{P}^{-1}$) that effects such a transformation. Generally speaking, preconditioning attempts to improve the spectral properties of the coefficient matrix. For symmetric problems, the rate of convergence of Krylov subspace method like CG or MINRES depends on the distribution of the eigenvalues of $\mathcal{A}$. Hopefully, the preconditioned matrix $\mathcal{M} = \mathcal{P}^{-1}\mathcal{A}$ (or $\mathcal{M} = \mathcal{A}\mathcal{P}^{-1}$) will have a smaller spectral condition number, and/or eigenvalues clustered around 1. Another favorable situation is when the preconditioned matrix has a minimum polynomial of small degree.

For nonsymmetric (nonnormal) problems the situation is more complicated, and the eigenvalues

may not describe the convergence of nonsymmetric matrix iterations like GMRES; see the discussion in section 9. Nevertheless, a clustered spectrum (away from 0) often results in rapid convergence, especially if the departure from normality of the preconditioned matrix is not too high.

Generally speaking, there are two approaches to constructing preconditioners. One is based on purely algebraic techniques, like incomplete factorizations, sparse approximate inverses, and algebraic multilevel methods. These preconditioners require little knowledge of the problem at hand besides the entries of the coefficient matrix, and can be applied—at least in principle—in a more or less black-box fashion. This type of preconditioning has proven quite effective in the solution of linear systems arising from the discretization of *scalar* partial differential equations of elliptic type, and is widely used in many areas of computational science and engineering; see [43, 360, 423, 475] for recent treatements. When applied to saddle point systems, on the other hand, standard algebraic preconditioners are often found to perform poorly. Because of the indefiniteness and lack of diagonal dominance, these preconditioners are often unstable. Even when the computation of the preconditioner does not suffer from some type of breakdown (e.g., zero pivots in an incomplete factorization), the quality of the resulting preconditioner is often not very satisfactory, and slow convergence is observed. Also, because of the absence of decay in $\mathcal{A}^{-1}$, it is difficult to construct good sparse approximate inverse preconditioners for saddle point matrices.

The second approach consists in developing preconditioners that are tailored to the particular application at hand. This approach requires knowledge of the origin of the problem, including (for PDEs) details about the discretization used, the underlying geometry, properties of the coefficients, and so forth. Of course, the more information one can use, the better the quality of the resulting preconditioner. The drawback of this approach is that the range of problems that can be treated with a particular preconditioner will be necessarily narrow, but this may not be a problem from the user's viewpoint.

For saddle point problems, the construction of high-quality preconditioners necessitates exploiting the block structure of the problem, together with detailed knowledge about the origin and structure of the various blocks. Because the latter varies greatly from application to application, there is no such thing as the "best" preconditioner for saddle point problems. The choice of a preconditioner is strongly problem-dependent. For instance, techniques that give excellent results for the time-dependent Stokes problem may be completely inadequate to deal with the steady-state case, or with the Oseen equations. Preconditioners that have been successfully used in optimization may be useless in fluid dynamics, and conversely. The good news is that powerful preconditioning techniques have been developed for many problems of practical interest.

We review several such techniques below. We begin with block preconditioners, especially popular in fluid dynamics, and continue with constraint preconditioners, especially popular in optimization. As it turns out, the first class of preconditioners has close ties with Schur complement reduction, while the second is related to the null space approach. Next we describe a more recent, but promising class of methods based on the Hermitian and skew-Hermitian splitting of $\hat{\mathcal{A}}$ in (3.10). This approach is rather general and has already been applied to a fairly wide range of problems, from fluid dynamics to weighted least squares. We conclude this section with a brief discussion of recent attempts to develop reliable incomplete factorization techniques for symmetric indefinite systems. It is also worth noting that both the stationary iterative methods described in section 8 and the multilevel methods described in section 11 can be accelerated by (used as preconditioners for) Krylov subspace methods.

The survey of preconditioners in this section is by no means exhaustive; see [26, 514] for other overviews of preconditioning techniques for saddle point problems.

10.1. Block preconditioners. In this section we consider block diagonal and block triangular preconditioners for Krylov subspace methods applied to the coupled system $\mathcal{A}\,u = b$. As we shall see, the performance of such preconditioners depends on whether fast, approximate solvers for linear systems involving A and the Schur complement S are available. Therefore, the preconditioners considered in this section are related to the "segregated" solvers considered in section 5.

10.1.1. Block diagonal preconditioners. Here we consider block diagonal preconditioning for the case of an invertible but possibly nonsymmetric saddle point matrix $\mathcal{A}$ with $C = O$. Then the basic block diagonal preconditioner is given by

$$\mathcal{P}_d = \begin{bmatrix} A & O \\ O & -S \end{bmatrix}, \tag{10.1}$$

where $S = -B_2 A^{-1} B_1^T$ is the Schur complement. Left preconditioning of $\mathcal{A}$ with $\mathcal{P}_d$ results in the matrix

$$\mathcal{M} = \mathcal{P}_d^{-1}\mathcal{A} = \begin{bmatrix} I & A^{-1}B_1^T \\ -S^{-1}B_2 & O \end{bmatrix}. \tag{10.2}$$

The matrix $\mathcal{M}$ is nonsingular by assumption and, as pointed out in [314, 364], it satisfies

$$(\mathcal{M} - I)\left(\mathcal{M} + \frac{1}{2}(1+\sqrt{5})I\right)\left(\mathcal{M} - \frac{1}{2}(1+\sqrt{5})I\right) = O.$$

Hence $\mathcal{M}$ is diagonalizable and has only three distinct eigenvalues, namely 1, $\frac{1}{2}(1+\sqrt{5})$, and $\frac{1}{2}(1-\sqrt{5})$ (see [131] for the complete form of the eigendecomposition of $\mathcal{M}$). Hence for each inititial residual r_0, $\dim \mathcal{K}_{n+m}(\mathcal{M}, r_0) \le 3$, which means that the GMRES method applied to the preconditioned system with coefficient matrix $\mathcal{M}$ will terminate after at most 3 steps. The same can be shown for right preconditioning with $\mathcal{P}_d$, or any centered preconditioning of the form $\mathcal{P}_1^{-1}\mathcal{A}\mathcal{P}_2^{-1}$ with $\mathcal{P}_1\mathcal{P}_2 = \mathcal{P}_d$. Furthermore, these results generalize to the case of a nonzero (2,2) block C of the matrix $\mathcal{A}$ [281].

At first sight this looks promising. However, a simple calculation using the formula (3.4) for the inverse of $\mathcal{A}$ shows that

$$\mathcal{A}^{-1} = \left(\mathcal{M} + \begin{bmatrix} A^{-1}B_1^T S^{-1} B_2 & 0 \\ 0 & -I \end{bmatrix}\right)\mathcal{P}_d^{-1}.$$

We see that forming the preconditioned system $\mathcal{M}u = \mathcal{P}_d^{-1}b$ out of the given saddle point system $\mathcal{A}u = b$ using the block diagonal preconditioner (10.1) is essentially as expensive as computing the inverse of $\mathcal{A}$ directly using (3.4). In practice, the exact preconditioner (10.1) needs to be replaced by an approximation,

$$\hat{\mathcal{P}}_d = \begin{bmatrix} \hat{A} & O \\ O & -\hat{S} \end{bmatrix}, \tag{10.3}$$

where both $\hat{A}$ and $\hat{S}$ are approximations of A and S, respectively.

Several different approximations have been considered in the literature. Examples from a few specific applications are given in section 10.1.3 below. Here we describe a fairly general framework developed in [131]. There the preconditioner (10.3) is obtained by considering a splitting of A into

$$A = D - E,$$

where D is invertible. Then $\hat{A} = D$ and $\hat{S} = -B_2 D^{-1} B_1^T$ are chosen, and bounds for the eigenvalues of the resulting preconditioned matrix

$$\hat{\mathcal{M}} = \hat{\mathcal{P}}_d^{-1}\mathcal{A} = \begin{bmatrix} I - D^{-1}E & D^{-1}B_1^T \\ -\hat{S}^{-1}B_2 & O \end{bmatrix} = \begin{bmatrix} I & D^{-1}B_1^T \\ -\hat{S}^{-1}B_2 & O \end{bmatrix} - \begin{bmatrix} D^{-1}E & O \\ O & O \end{bmatrix} \tag{10.4}$$

are given in terms of the eigenvalues of $\mathcal{M} = \mathcal{P}_d^{-1}\mathcal{A}$. These bounds show that the distance of the eigenvalues of $\hat{\mathcal{M}}$ to the three distinct eigenvalues of $\mathcal{M}$ depends on several factors, including the norm of the matrix $D^{-1}E$. In particular, when A is diagonally dominant, and $D = \mathrm{diag}(A)$, then a good clustering of the eigenvalues of $\hat{\mathcal{M}}$ around 1, $\frac{1}{2}(1+\sqrt{5})$, and $\frac{1}{2}(1-\sqrt{5})$ can be expected. However, the preconditioned matrix $\hat{\mathcal{M}}$ still has (now potentially $n+m$ distinct) eigenvalues that may be on both sides of the imaginary axis.

To overcome this drawback the authors of [131] propose to combine the approximate block diagonal preconditioner (10.3) with an Uzawa-type fixed point iteration (see section 8.1 above) that is based on a splitting of $\hat{\mathcal{M}}$ as in the rightmost expression in (10.4) into $\hat{\mathcal{M}} = \mathcal{P}_1 - \mathcal{P}_2$. The inverse of $\mathcal{P}_1$ is known explicitly, and if D is simple enough (e.g., diagonal), it can be efficiently computed. Then the preconditioned system $\hat{\mathcal{M}}u = \hat{\mathcal{P}}_d^{-1}b = \hat{b}$ can be written as

$$(\mathcal{P}_1 - \mathcal{P}_2)u = \hat{b} \quad \Leftrightarrow \quad u = \mathcal{P}_1^{-1}\mathcal{P}_2 u + \mathcal{P}_1^{-1}\hat{b},$$

which yields the fixed point iteration

$$u_{k+1} = \mathcal{P}_1^{-1}\mathcal{P}_2 u_k + \mathcal{P}_1^{-1}\hat{b}, \quad \text{where} \quad \mathcal{P}_1^{-1}\mathcal{P}_2 = \begin{bmatrix} (I + D^{-1}B_1^T\hat{S}^{-1}B_2)D^{-1}E & O \\ -\hat{S}^{-1}B_2 D^{-1}E & O \end{bmatrix}. \tag{10.5}$$

Because of the form of the iteration matrix $\mathcal{P}_1^{-1}\mathcal{P}_2$, it is easy to see that this fixed point iteration depends only on the first n components of u, i.e. it is sufficient to consider the iteration

$$x_{k+1} = (I + D^{-1}B_1^T\hat{S}^{-1}B_2)D^{-1}E\,x_k + [\mathcal{P}_1^{-1}\hat{b}]_{1:n}\,,$$

where $[\mathcal{P}_1^{-1}\hat{b}]_{1:n}$ denotes the first n components of the vector $\mathcal{P}_1^{-1}\hat{b}$. Hence this approach can be considered a segregated solution method, in which (an approximation to) x_* is computed first, which is then used to compute (an approximation to) y_*. The *reduced system* of order n here is given by

$$\Big(I - (I + D^{-1}B_1^T\hat{S}^{-1}B_2)D^{-1}E\Big)\,x = [\mathcal{P}_1^{-1}\hat{b}]_{1:n}. \tag{10.6}$$

Obviously, this solution technique reduces the number of unknowns compared to the preconditioned system with the matrix $\hat{\mathcal{M}}$ from $n+m$ to n. This might result in a significant saving of computational resources especially when m is relatively large. Moreover, and possibly even more importantly, the eigenvalues of the system matrix in (10.6) tend to be clustered around the point one, with tighter clustering corresponding to a smaller norm of the matrix $D^{-1}E$. In particular, the spectrum has only one instead of the three clusters of $\hat{\mathcal{M}}$, see [131]. Because of this, the performance of a Krylov subspace method such as GMRES applied to (10.6) typically outperforms the performance for the block diagonally preconditioned system with the matrix $\hat{\mathcal{M}}$. It is important to note that the approach through (10.6) is closely related to (left) *constraint preconditioning* as described in section 10.2 below. For simplicity, consider the case with $B_1 = B_2 = B$, and the constraint preconditioner $\mathcal{P}_c$ in (10.15) with $G = D$ resulting from the splitting $A = D - E$. Then a straightforward calculation shows that the system matrix in (10.6) is exactly the same as the (1,1) block of the preconditioned matrix $\mathcal{M}$ in (10.17). Also note that the block diagonal preconditioner $\hat{\mathcal{P}}_d$ in this case is equal to the block diagonal matrix in the factorization of the constraint preconditioner $\mathcal{P}_c$ in (10.15). Further details about the relation between these two approaches can be found in [131].

Next, consider the case of a symmetric saddle point matrix $\mathcal{A}$ with A positive definite and $C = O$. Then the authors of [178] consider block diagonal preconditioners of the form

$$\hat{\mathcal{P}}_d^{\pm} = \begin{bmatrix} \eta^{-1}A & O \\ O & \pm\hat{S} \end{bmatrix}, \tag{10.7}$$

where $\eta > 0$ is a scaling parameter, and $\hat{S}$ is a symmetric positive definite approximation of the Schur complement. Note that $\hat{\mathcal{P}}_d^{+}$ is positive definite, while $\hat{\mathcal{P}}_d^{-}$ is indefinite. Since both A and $\hat{S}$ are symmetric positive definite, their Cholesky decompositions $A = LL^T$ and $\hat{S} = MM^T$ can be used to transform the system $\mathcal{A}u = b$ into the equivalent system

$$\begin{bmatrix} \eta I & (M^{-1}BL^{-T})^T \\ \pm M^{-1}BL^{-T} & O \end{bmatrix} \begin{bmatrix} \eta^{-1}L^T x \\ M^T y \end{bmatrix} = \begin{bmatrix} L^{-1}f \\ (\eta M)^{-1}g \end{bmatrix}. \tag{10.8}$$

Apparently, the system matrix in (10.8) is of the same form as $\mathcal{A}_\eta^{\pm}$ in (3.13), and hence its eigenvalues depending on the choice of $+$ or $-$ and of η are characterized in Theorem 3.8. There it is shown that the eigenvalue distributions vary greatly from each other. In addition, the positive definite preconditioner $\hat{\mathcal{P}}_d^{+}$ yields a symmetric system matrix $\mathcal{A}_\eta^{+}$ in (10.8), while the indefinite preconditioner $\hat{\mathcal{P}}_d^{-}$ leads to the nonsymmetric matrix $\mathcal{A}_\eta^{-}$. For a fixed η it is therefore interesting to understand which choice yields the better performance of a Krylov subspace method applied to the preconditioned system (10.8). Curiously, there may be no difference at all. In fact, assuming that $g = 0$ in the saddle point system, the authors of [178] show that the preconditioned matrices $\mathcal{A}_\eta^{\pm}$ in (10.8) both generate the same Krylov subspace iterates u_k when special initial residuals are used. In this case the residual norms of MINRES applied to the symmetric system coincide with the residual norms of GMRES applied to the nonsymmetric system in every step of the iteration. This exact equivalence is lost when random initial residuals are used, but as shown numerically in [178], the convergence curves are very close in such cases as well. Since the MINRES method is based on three-term recurrences due to symmetry of the system matrix (cf. section 9), the positive definite preconditioner clearly represents the superior strategy.

Block diagonal preconditioners for saddle point problems arising from (stabilized) discretized Stokes problems (2.10)–(2.12) have been studied, for example, in [439, 493]. The system matrix of the discrete problem is a generalized saddle point matrix of the form

$$\mathcal{A} = \begin{bmatrix} A & B^T \\ B & -\beta C \end{bmatrix}, \tag{10.9}$$

where A is block diagonal and symmetric positive definite, C is symmetric positive semidefinite, and $\beta > 0$ is a stabilization parameter. Silvester and Wathen provide in [439, 493] several results on the eigenvalue distribution of the preconditioned matrix $\hat{\mathcal{P}}_d^{-1}\mathcal{A}$ for different symmetric positive definite preconditioners $\hat{\mathcal{P}}_d$ of the form (10.3).

For example, consider the case $\hat{A} = \operatorname{diag}(A)$, and $-\hat{S} = \beta \operatorname{diag}(C)$ if $C \neq O$ or $-\hat{S} = \beta h^d I$ if $C = O$ (here d is the spatial dimension of the Stokes problem, and h is the mesh size parameter), i.e. a positive definite diagonal preconditioner $\hat{\mathcal{P}}_d$. Then it can be shown that the eigenvalues of the symmetric indefinite preconditioned matrix $\hat{\mathcal{P}}_d^{-1}\mathcal{A}$ are contained in the union of two real intervals of the form

$$(-a, -bh) \cup (ch^2, d), \tag{10.10}$$

where a, b, c, d are positive constants that are independent of the mesh size h [493, Theorems 1 and 2]. As discussed in section 9, the (worst-case) convergence of the MINRES method applied to the

preconditioned system depends on the eigenvalue distribution of $\hat{\mathcal{P}}_d^{-1}\mathcal{A}$, and an upper convergence bound can be found by using (the smallest) two intervals containing the eigenvalues. If these intervals are of the same length, then such a convergence bound is given by (9.9)–(9.11). But in the case of (10.10), the two intervals are in fact not of the same length, since the negative eigenvalues spread out less rapidly under mesh refinement than the positive eigenvalues. Extending the two intervals in (10.10) to make their lengths equal, say by replacing (10.10) with $(-d, -ch^2) \cup (ch^2, d)$ (where we assume, for simplicity, $d > a$ and $b > c$), would predict an asymptotic convergence rate of $O(1 - h^2 c/d)$ for the MINRES method. However, with some careful analysis it can be shown that the asymptotic convergence factor of (10.10) is indeed of order $O(1 - h^{3/2}\sqrt{bc/ad})$; see [492] for details. More sophisticated positive definite block diagonal preconditioners in this context leading to even better asymptotic convergence rates are studied in [439]; see also section 10.1.3 below.

Apart from the work in [439, 493], block diagonal preconditioners have been studied in the context of numerous additional applications. The upshot is that for suitably chosen approximations to A and S, block diagonal preconditioners resulting in mesh-independent rates of convergence exist for mixed finite element formulations of many problems. For example, an analysis of the eigenvalue distribution of block diagonally preconditioned saddle point systems arising from mixed finite element discretizations of magnetostatics problems is given in [391, Section 4]. Other analyses of block diagonal preconditioners in specific applications can be found in [36, 103, 104, 301, 312, 314, 315, 340, 342, 347, 383, 384, 399, 463, 480].

10.1.2. Block triangular preconditioners. Block triangular preconditioners of the form

$$\mathcal{P}_t = \begin{bmatrix} A & B^T \\ O & S \end{bmatrix}$$

have been first considered in [72] and extensively developed in the last few years. Note that Uzawa's method may also be regarded as a block (lower) triangular preconditioner, cf. (8.2). This class of preconditioners includes some of the most efffective solvers for saddle point problems, both symmetric and nonsymmetric. Note that the first block row of $\mathcal{P}_t$ coincides with the first block row of $\mathcal{A}$; hence, for the initial guess $u_0 = 0$, the solution of the initial preconditioning equation $\mathcal{P}_t\, z_0 = r_0 = b$ must in particular satisfy the first of the two equations in the saddle point system (1.6). In section 10.2 below we will examine preconditioning strategies that require satisfying the *second* equation in the saddle point system (constraint preconditioning). In the setting of fluid flow problems, the first approach requires the preconditioner to respect the conservation of momentum, whereas the second one imposes a mass balance condition.

Recalling the block factorization (3.3), we immediately see that the spectrum of $\mathcal{P}_t^{-1}\mathcal{A}$ is $\sigma(\mathcal{P}_t^{-1}\mathcal{A}) = \{1\}$ and moreover, the preconditioned matrix has minimum polynomial of degree 2, so that a method like GMRES would converge in at most two steps. In practice, of course, approximations $\hat{A}$ and $\hat{S}$ to A and S have to be used. The computation of $z_k = \mathcal{P}_t^{-1} r_k$ at each step of a nonsymmetric Krylov subspace method can be implemented on the basis of the factorization

$$\mathcal{P}_t^{-1} = \begin{bmatrix} \hat{A}^{-1} & O \\ O & I \end{bmatrix} \begin{bmatrix} I & B^T \\ O & -I \end{bmatrix} \begin{bmatrix} I & O \\ O & -\hat{S}^{-1} \end{bmatrix}. \tag{10.11}$$

Note that the cost of applying the preconditioner is only slightly higher than in the block diagonal case: besides the solves with $\hat{A}$ and $\hat{S}$, there is an additional multiplication by B^T. Once again, the choice of the approximations $\hat{A}$ and $\hat{S}$ is problem-dependent, see section 10.1.3 below. Generally speaking, the better the approximations, the faster the convergence. However, since the preconditioned matrix $\mathcal{P}_t^{-1}\mathcal{A}$ (or $\mathcal{A}\mathcal{P}_t^{-1}$) is nonnormal, convergence estimates are not easy to obtain. Field of values analysis can been used in some instances to obtain eigenvalue bounds and convergence

estimates for GMRES. These analyses and numerical experiments indicate that h-independent convergence rates can be achieved for some important problems, including the Oseen equations; see [305, 336] and the discussion in the next section. Other analyses of block triangular preconditioners for specific applications can be found in [36, 92, 168, 291, 302, 312, 384]; see also [72, 440, 514] for analyses of the inexact case. An apparent disadvantage in the symmetric case is that block triangular preconditioning destroys symmetry. A symmetrized version of block triangular preconditioning has been proposed in [72]; see further [26, 155, 514]. However, symmetrization is seldom necessary in practice: if good approximations to A and S are available, a method like GMRES with block triangular preconditioning will converge quickly and the overhead incurred from the use of a nonsymmetric solver will be negligible.

We further mention preconditioners based on incomplete block triangular factorizations of the form

$$\mathcal{P} = \begin{bmatrix} \hat{A} & O \\ B & \hat{S} \end{bmatrix} \begin{bmatrix} I & \hat{A}^{-1}B^T \\ O & I \end{bmatrix} \approx \mathcal{A}\,,$$

where $\hat{A}$ and $\hat{S}$ are easily invertible approximations of A and of the Schur complement S. Note that application of this preconditioner requires two solves with $\hat{A}$ rather than one, in addition to the solve with $\hat{S}$. Multiplying out the factors, we obtain

$$\mathcal{P} = \begin{bmatrix} \hat{A} & B^T \\ B & -\hat{C} \end{bmatrix}, \quad \hat{C} = -(B\hat{A}^{-1}B^T + \hat{S})\,.$$

Hence, generally speaking, $\mathcal{P}$ is an indefinite preconditioner of the type considered in section 10.2 below. Note that $\mathcal{P}$ is symmetric if $\hat{A}$ and $\hat{S}$ are. Again, the key issue here is the choice of the approximations $\hat{A} \approx A$ and $\hat{S} \approx S$. Such incomplete block LU factorization preconditioners correspond to classical solution methods known in computational fluid dynamics as SIMPLE schemes (for "Semi-Implicit Method for Pressure-Linked Equations"), see [380, 382]. In the original SIMPLE scheme $\hat{A} = D_A$ and $\hat{S} = -BD_A^{-1}B^T$, respectively, where D_A denotes the main diagonal of A. Different choices of the approximations involved lead to different preconditioners. Variants of SIMPLE are also often used as smoothers for multigrid; see the discussion in section 11 below. A spectral analysis of the SIMPLE preconditioner applied to the Oseen problem can be found in [322]. See further [48] for numerical experiments and comparisons with other preconditioners. These experiments confirm the intuition that SIMPLE preconditioning is effective when A is strongly diagonally dominant, as is the case for generalized Stokes and Oseen problems with sufficiently small time steps. On the other hand, SIMPLE is not competitive when the time steps are large or "infinite" (steady case), for in this case the approximations to A and especially to S are poor. It should be mentioned that for unsteady problems classical (Chorin-style [105]) pressure-correction schemes are also very efficient, see [499, page 303]. Further references on incomplete block factorizations for the treatment of time-dependent Navier-Stokes equations include [203, 389, 406, 426, 484].

10.1.3. Approximating A and S. As we have seen, many algorithms for solving saddle point systems depend on the availability of good approximations for the (1,1) block A and for the Schur complement S. Such approximations are needed in segregated approaches and in preconditioned Uzawa schemes where the Schur complement system (5.1) is solved by a preconditioned iteration, and in the construction of block diagonal and block triangular preconditioners for coupled Krylov iterations applied to $\mathcal{A}u = b$.

The construction of such approximations is a strongly problem-dependent process, and a large body of literature exists on this topic. The problem has been especially well-studied for mixed finite element formulations of elliptic PDEs, for the Stokes and Oseen problems, and for linear elasticity

problems. Furthemore, some work has been done towards the construction of preconditioners for the Schur complement ("normal equations") systems arising from interior point methods in optimization.

In the case of mixed finite element approximations of elliptic boundary value problems and of Stokes and Oseen-type equations, the submatrix A usually corresponds either to a zeroth-order differential operator (multiplication by a function or tensor K, as in (2.18)), or to a direct sum of second order differential operators of diffusion or convection-diffusion type. When A represents multiplication by a function (mass matrix), it can be well approximated by either a (scaled) identity matrix or by $D_A = \mathrm{diag}\,(A)$. If the coefficients in K are sufficiently smooth, the resulting approximation $\hat{A}$ is then spectrally equivalent to A, in the sense that the eigenvalues of $\hat{A}^{-1}A$ are positive and contained in an interval whose endpoints do not depend on the mesh size; see [418, 419]. When A represent a discrete (vector) diffusion or convection-diffusion operator, a spectrally equivalent approximation $\hat{A}$ can be obtained in many cases by performing one or more multigrid sweeps on the linear system $A\,x = v$. Thus, the approximation $\hat{A}$ of A is not constructed explicitly, but is defined implicitly by the action $\hat{A}^{-1}v$ on a given vector v. We refer the reader to [170] for details. Here we note that for the generalized (unsteady) Stokes and Oseen problems the matrix A is usually well-conditioned due to the presence of an additional term inversely proportional to the time step Δt. This makes the (1,1) block strongly diagonally dominant, and a single multigrid V-cycle with an appropriate Gauss-Seidel smoother is often sufficient, see [347, 469].

When A does not arise from the discretization of a differential operator, approximate solves, if necessary, may be obtained by either incomplete factorizations of by a few iterations of a Krylov subspace method, possibly with an appropriate preconditioner.

The approximation $\hat{S}$ of the Schur complement matrix S is usually more delicate. In the context of saddle point systems arising from the discretization of PDEs, good approximations to the Schur complement are critical to the performance of block preconditioners. In many cases, excellent approximations can be constructed by thinking in terms of the underlying differential operators. We recommend [335] for a highly readable discussion of this problem in terms of pseudodifferential operators.

For mixed finite element discretizations of elliptic problems of the type (2.18)–(2.20), the Schur complement $S = -BA^{-1}B^T$ can be interpreted as a discretization of the second order diffusion operator $\nabla K^{-1} \cdot \nabla$ acting on the unknown function corresponding to the y variable (e.g., on the pressure for incompressible flow problems). In this case, a number of approximations $\hat{S}$ are possible. Provided that K is sufficiently well-behaved, the action of S^{-1} can be approximated by an iteration of multigrid. For higher order methods, a low order discretization of the same diffusion operator may also expected to give a very good approximation. The literature in this area is vast; see, e.g., [291, 314, 315, 335, 350, 363, 391, 399, 417, 418, 419, 491].

For LBB-stable discretizations of the linear elasticity and steady-state Stokes problem, the Schur complement is spectrally equivalent to a mass matrix [485]. This is not surprising if one thinks of S as a discrete counterpart of a pseudodifferential operator of the form $\mathrm{div}\,\Delta^{-1}\,\mathrm{grad}$ and keeping in mind the identity $\Delta = \mathrm{div}\,\mathrm{grad}$. Hence, under appropriate discretization the Schur complement has condition number bounded independently of mesh size and can be approximated effectively and cheaply in a number of ways; see the previous discussion on approximating zeroth order differential operators, as well as the references [160, 170, 301, 302, 335, 336, 383, 439, 493, 509].

For mixed finite element discretizations of the generalized Stokes problem that arises from the implicit treatment of the time-dependent Stokes problem, on the other hand, the Schur complement is of the form $S = -B(A + \beta I)^{-1}B^T$ where $\beta > 0$ is inversely proportional to the time step. In the

2D case A is the block diagonal matrix

$$A = \begin{bmatrix} \nu H & O \\ O & \nu H \end{bmatrix},$$

where H is a discrete Laplace operator and $\nu > 0$ denotes the viscosity. In the 3D case, there will be three blocks along the diagonal instead of two. Clearly, for very large time steps (β small) the matrix $S = -B(A + \beta I)^{-1}B^T$ is close to the Schur complement of the steady Stokes problem and is well-conditioned independent of mesh size. In this case a scaled pressure mass matrix will be a good approximation. On the other hand, for small step sizes (β large) this operator has a condition number that grows as $h \to 0$. Note that in the limit as $\beta \to \infty$, the Schur complement behaves essentially as a discrete diffusion operator. The same is true for the case of small viscosity ν (high Reynolds numbers Re), regardless of the size of the time step, for in this case the matrix A, which contains terms proportional to $\nu = Re^{-1}$, will be negligible. In these cases the SIMPLE approximation for the Schur complement can be quite good for sufficiently small β. Robust preconditioners that work well for small as well as large time steps and over a wide range of Reynolds numbers have been first proposed and analyzed in [89]. The idea is to build preconditioners that are discrete counterparts of the pseudodifferential operator L implicitly defined by

$$L^{-1} = \nu\, I - \beta\, \Delta^{-1},$$

where I denotes the identity operator and Δ is a pressure Laplacian. Here Δ^{-1} stands for the data-to-solution mapping for a pressure-Poisson equation with suitable boundary conditions. Thus, the action of the approximate Schur complement $\hat{S}$ on a vector v consists of

$$\hat{S}^{-1}v = \nu\, M_p\, v - \beta\, (B M_u^{-1} B^T)^{-1} v,$$

where M_p is a pressure mass matrix and M_u a velocity mass matrix. Note that $BM_u^{-1}B^T$ is nothing but the so-called *compatible discretization* of the Laplacian [89]. Again, inexact solves can be used to approximate $(BM_u^{-1}B^T)^{-1}v$. See further [469] and the recent paper [347].

The situation is more complicated for the nonsymmetric saddle point problems arising from discretizations of the (steady and unsteady) Oseen equations. For steady problems with large viscosity (small Reynolds numbers), using a pressure mass matrix to approximate the Schur complement is usually sufficient and results in rates of convergence independent of mesh size when block diagonal or block triangular preconditioners are used. However, the rate of convergence tends to quickly deteriorate as the viscosity gets smaller. The problem is that as $\nu \to 0$, the exact Schur complement becomes increasingly nonsymmetric and cannot be well approximated by a (necessarily symmetric) mass matrix; see, e.g, [92, 167, 312]. Much progress has been made in the last few years towards developing robust preconditioners of the block triangular type for the Oseen equations, see [161, 162, 164, 165, 168, 169, 170, 297, 334, 438]. Out of this effort, two especially interesting techniques have emerged. The first idea is to approximate the inverse of the Schur complement matrix $S = BA^{-1}B^T$ with

$$\hat{S}^{-1} = (BB^T)^{-1}BAB^T(BB^T)^{-1} \tag{10.12}$$

(here and in the remainder of this section we dispense with the negative sign for ease of notation). This approximation, originally proposed by Elman [161] (see also [426]), has been given various justifications. A simple one is to observe that if B were square and invertible, then the inverse of $BA^{-1}B^T$ would be $B^{-T}AB^{-1}$. However, B is rectangular. Therefore it makes sense to replace the inverse of B with the Moore-Penrose pseudoinverse $B^\dagger$, see [90]. If we assume B to have full row rank, the pseudoinverse of B is given by $B^\dagger = B^T(BB^T)^{-1}$, thus justifying the approximation

(10.12). Now the approximate Schur complement is (strongly) nonsymmetric if A is. Note that evaluating the action of $\hat{S}^{-1}$ on a given vector v requires the solution of two pressure-Poisson type problems, plus matrix-vector multiplies involving B^T, A and B. It follows from (10.11) that the block triangular preconditioner can be applied by performing a matrix-vector products involving B^T, A and B, and a few iterations of multigrid. For 3D problems, multigrid is needed to approximately solve three convection-diffusion problems for the three components of the velocity field (approximate inversion of A), and to approximately solve the two Poisson-type problems on the pressure space. Although this preconditioner is quite robust with respect to mesh refinement, as it stands it still suffers from some deterioration as the viscosity goes to zero. Furthermore, it was found to perform poorly (occasionally failing to converge) on some difficult problems, see [473]. As noted in [164], for finite element problems the performance of this preconditioner can be greatly improved by the use of appropriate scalings. Denoting by M_1 and M_2 suitable diagonal matrices, the authors of [164] derive the more accurate approximation

$$\hat{S}^{-1} = (BM_2^{-2}B^T)^{-1}BM_2^{-2}AM_1^{-1}B^T(BM_1^{-2}B^T)\,. \tag{10.13}$$

Tha matrix M_1 is taken to be the diagonal of the velocity mass matrix M_u; taking $M_2 = M_1^{1/2}$ makes the two variable-coefficient discrete diffusion operators in (10.13) identical. Although the cost of the two variants (10.12) and (10.13) is almost the same, the improvement in performance obtained by the use of scaling can be quite dramatic, see the numerical experiments in [164].

Another effective approximation of the Schur complement for the Oseen problem has been introduced in [297] and analyzed in [169]. Again, several justifications have been given for this approach, the original one being based on the Green's tensor of the Oseen operator on the continuous level. Another elegant derivation in terms of pseudodifferential operators can be found in [335]. A simpler, heuristic argument is to first consider the case where A and B^T are both square and commute: $AB^T = B^TA$. If A is nonsingular, its inverse must also commute with B^T, hence $A^{-1}B^T = B^TA^{-1}$. If we think of A as representing a second order differential oprerator and B a first order one then, apart for possible problems near the boundary, assuming commutativity is reasonable. It follows from these assumptions that $S^{-1} = (BA^{-1}B^T)^{-1} = A(BB^T)^{-1}$. In practice, however, A and B^T are rectangular and represent operators acting on different function spaces. Note that BB^T is, again, a (scalar) pressure-Poisson type operator; on the other hand, A is a (vector) convection-diffusion operator acting on the velocities. This suggests introducing a discrete convection-diffusion operator A_p acting on the pressures. Including the necessary mass matrix scalings, the resulting approximation to the inverse of the Schur complement is thus

$$\hat{S}^{-1} = M_p^{-1}A_p(BM_u^{-1}B^T)^{-1}\,. \tag{10.14}$$

The block triangular preconditioner based on this approximate Schur complement has been extensively tested on a variety of steady and unsteady model problems, in both 2D and 3D; see, e.g., [297, 165, 168, 169, 261, 334]. This method, which is referred to as the *pressure convection-diffusion* preconditioner in [170], appears to be very robust with respect to mesh refinement, and fairly robust with respect to the viscosity, exhibiting only a mild deterioration as $\nu \to 0$ on a wide range of problems and discretizations. Note that this approximation is much less expensive than the one based on (10.11), since computing the action of $\hat{S}^{-1}$ only requires solving a single pressure-Poisson problem and a matrix-vector product with A_p.

One drawback of this apprach is the need to construct an additional operator, A_p, which is not required by the formulation of the Oseen problem but is needed only to form the preconditioner. This discrete convection-diffusion operator is not usually available in standard fluid dynamics codes. This has motivated new, fully algebraic approaches to compute approximations to A_p from the existing blocks A, B and B^T based on the notion of *sparse approximate commutators* (SPAC); see

[164]. The approximate A_p is the matrix that minimizes the cost functional

$$F(X) = ||AB^T - BX||_F$$

(where $||\cdot||_F$ denotes the Frobenius norm) over a space of matrices with prescribed sparsity pattern. Note that the solution of the *unconstrained* minimization problem is precisely Elman's approximation $\hat{S}^{-1} = (BB^T)^{-1}BAB^T(BB^T)^{-1}$.

The block triangular preconditioner based on the pressure convection-diffusion operator has been also used with good results as a smoother for multigrid applied to the Oseen problem in [455]. It has also been used as a preconditioner for the linear systems arising from Newton's method applied to the Navier-Stokes equations in [166]. In this case the method was again found to result in h-independent convergence, but some growth in the iteration number was observed for small ν.

The pressure convection-diffusion operator, on the other hand, does not seem to be applicable to the rotation form (2.15)–(2.17) of the Oseen equations. Different approximations need to be used in this case; see [375] and section 10.3 below.

Approximations for the Schur complement matrices arising in interior point methods for constrained optimization have been studied by several authors. Here we give a brief discussion based on [193]. As already pointed out in section 3.5, such linear systems can be extremely ill-conditioned. An interesting technique is the *tree preconditioner* described in [412], based on ideas first put forth in [472]. The approximation to $S = BA^{-1}B^T$ is of the type $\hat{S} = \hat{B}M\hat{B}^T$ where $\hat{B}$ and M are now both square matrices of order m. The actual construction of these matrices depend on the application. In network flow problems (the motivating application in [412]), the constraint matrix B is the node-edge incidence matrix of a directed graph G, and $\hat{B}$ is defined as the node-edge incidence matrix of a spanning tree of G. Furthermore, A is diagonal and M is obtained by restricting A^{-1} to the arcs that comprise the tree. (Note that when $A = I$, the matrix $S = BB^T$ is closely related to the Laplacian matrix of the undirected version of G.) More precisely, $\hat{B}$ is chosen as the node-edge incidence matrix of an (approximate) maximum weight spanning tree, where the weight of each edge (i, j) is given by the corresponding entry of M. Such spanning trees can be constructed in $O(m)$ work using a variant of the classical Kruskal algorithm. The resulting linear systems with coefficient matrix $\hat{S}$ can be solved in $O(n)$ work by solving linear systems with matrices $\hat{B}^T$, M and $\hat{B}$. These linear systems can be solved by visiting the spanning tree, without fill-in. This preconditioner can be expected to be especially effective in the final stages of the interior point iteration, since in this case the weights m_{ij} tend to zero on all arcs, except on those corresponding to the basic optimal solution that form a spanning tree. The preconditioner is less effective in the initial stages of the interior point algorithm. This observation has motivated the use of hybrid strategies where a diagonal preconditioner is used initially and the tree preconditioner in later stages of the algorithm, see [412] for details. Improvements of the tree preconditioner have been proposed in [358]. A different strategy, based on an incomplete QR factorization, has been proposed in [287], and found to be superior to the tree preconditioner for the special case of transportation problems.

New preconditioners based on combinatorial principles have been recently introduced in [193]. Motivated by the tree preconditioner idea, these authors construct approximations of S of the form $\hat{S} = \hat{B}M\hat{B}^T$ where now $\hat{B}$ corresponds to a *subgraph* of the graph G of B; this subgraph, which may contain (strictly) a spanning tree of G, should be such that linear systems associated with the matrix $\hat{S}$ can be easily solved. For example, if the chosen subgraph happens to be *chordal*, then there is an ordering of $\hat{S}$ such that no fill-in occurs when factoring $\hat{S}$; see [415]. Fast algorithms exist to compute approximate maximum weight chordal subgraphs. We refer the reader to [193] for further details and for the result of numerical experiments showing that in many cases the new subgraph-based preconditioners improve on existing techniques both in terms of iteration counts and in CPU time.

An interesting open question is to what extent these ideas can be exploited to construct approximate Schur complements for saddle point problems arising from mixed finite element discretizations of PDE problems, where the matrix B is a discrete divergence operator.

We conclude this section with a brief discussion of purely algebraic methods that have been used to construct approximations $\hat{S}$ or $\hat{S}^{-1}$ to the Schur complement matrix or to its inverse. See also the general treatment in [25, Chapter 9] and the discussion in [437, section 4]. For simplicity, we confine ourselves to the case where $B_1 = B_2$ and $C = O$. Extension to cases with $B_1 \neq B_2$ is usually straightforward; the case $C \neq O$, on the other hand, may be less obvious, especially when seeking explicit approximations to S^{-1}.

The simplest approximation is perhaps to set $\hat{S} = BD_A^{-1}B^T$ where D_A is either diagonal or possibly block diagonal with blocks of small size. Note that in this case $\hat{S}$ will be sparse if B is, except for special situations (e.g., when B contains one or more dense columns). The diagonal approximation $D_A = \mathrm{diag}\,(A)$ is used for instance in the already mentioned SIMPLE-type schemes popular in computational fluid dynamics. Better approximations can be obtained by means of incomplete factorizations of the form $A \approx LU \equiv \hat{A}$. Then $\hat{S} = XY$ where $X = BU^{-1}$ and $Y = L^{-1}B^T$. Matrices X and Y may be computed by solving two lower triangular linear systems with coefficient matrices U^T and L and with multiple right-hand-sides given by the columns of B^T. A posteriori dropping of small entries may be necessary to keep X and Y sparse. Alternatively, one can use (explicit) sparse approximate inverse techniques directly applied to A. For example, the approximate inverse M_A may be computed as the minimizer of the matrix functional

$$F(X) = ||I - AX||_F$$

over a prescribed set of sparse matrices. Many other approximate inverse techniques have been developed; see [43] for an overview. Whatever the technique employed, the resulting sparse approximate inverse $M_A \approx A^{-1}$ can then be used to form an approximation $\hat{S} = BM_AB^T$. See [107, 330, 331] for implementation details and for experimental comparisons of various strategies. These techniques may be expected to work well when the entries in A^{-1} decay rapidly away from the main diagonal. This can be expected to hold, for instance, for implicit time discretizations of time-dependent Stokes and Oseen problems. Provided that sufficiently small time steps are taken, the diagonal entries of A will be large compared to the off-diagonal and will result in fast decay in A^{-1}, see [132].

Other approximations that have been used in the literature for specific applications can be used, at least in principle, in more general settings. In [288], the simple approximation $(BA^{-1}B^T)^{-1} \approx BAB^T$ is proposed as an approximate inverse of the Schur complement for saddle point matrices arising from the use of Lagrange multipliers in domain decomposition schemes. In this case B is often a very simple matrix representing a restriction operator, and the approximation is not unreasonable. Although this approximation is not very accurate, it has the advantage of having zero construction cost; furthermore, computing the action of $\hat{S}^{-1}$ on a vector only requires matrix vector products with sparse matrices. See further [342, 470] for more sophisticated approximations to the Schur complements arising in similar contexts.

Another possible approximation is the already mentioned Elman preconditioner of the form

$$\hat{S}^{-1} = (BB^T)^{-1}BAB^T(BB^T)^{-1}\,.$$

Although developed specifically for fluid flow problems, it should be possible, at least in principle, to use this approximation whenever the solution of linear systems with coefficient matrix BB^T (or of the equivalent least squares problems) can be computed efficiently. Note that accommodating a nonzero C in this and the previous case is not easy.

Yet another possibility is to approximate the action of S^{-1} on a vector v by performing a few step of an iterative method on the Schur complement system $Sz = v$. Krylov subspace methods only require S in the form of matrix-vector products. These in turn necessitate multiplications with B and its transpose and solves with A. The latter operation may in turn be performed by an (inner) iterative process. If a variable number of inner iterations is performed, then a generalized conjugate gradient method [27] or a flexible solver like FGMRES [422] should be used for the outer iteration. These inner solves need not be performed to high accuracy. Even if an accurate approximation $z \approx S^{-1}v$ is sought, the accuracy of the inner solves with A can be relaxed in the course of the outer iteration, see [444]. We further mention that matrix-vector products involving S and selected vectors are used in the *probing* technique to compute banded approximations to S; see, e.g., [25, Chapter 8.5.1]. Finally, in [65] a parallel hybrid direct-iterative approximate Schur complement algorithm has been proposed for linear systems arising in circuit simulation.

10.2. Constraint and indefinite preconditioning. Constraint preconditioning is based on the principle that the preconditioning matrix should have the same 2×2 block structure as the original saddle point matrix. In other words, the saddle point matrix is preconditioned by another, "easier to invert" saddle point matrix. Constraint preconditioning has been extensively used in the solution of saddle point systems arising from mixed finite element formulations of elliptic partial differential equations [26, 30, 172, 361, 391, 416, 464]. The method is also popular in the solution of saddle point ("KKT") systems in optimization [50, 156, 201, 228, 298, 340, 434]. In [309], the preconditioner is applied to problems arising in the finite element analysis of structures.

We describe constraint preconditioning in the case where $B_1 = B_2 = B$ has full row rank and $C = O$. The preconditioning matrix is then

$$\mathcal{P}_c = \begin{bmatrix} G & B^T \\ B & O \end{bmatrix}, \tag{10.15}$$

where G is some approximation of A, $G \neq A$. Of course, G should be chosen so that $\mathcal{P}_c$ is invertible and solving linear systems with coefficient matrix $\mathcal{P}_c$ is significantly easier than solving the original linear system $\mathcal{A}\,u = b$. The name *constrained preconditioning* comes from the fact that $\mathcal{P}_c$ is the coefficient matrix for a saddle point problem with the same constraints as the original problem. Moreover, the constraint preconditioner projects the problem onto the null space of the constraint matrix B, as noted in [340, 392]. With an appropriate choice of the initial guess u_0, the component x_k of every iterate u_k generated by a Krylov method preconditioned with $\mathcal{P}_c$ satisfies the constraint $Bx_k = g$; see, e.g., [416].

When A has positive diagonal entries G is often taken to be the diagonal part of A, and if $A = (a_{ij})$ has been scaled so that $a_{ii} = 1$ for $1 \leq i \leq n$, then $G = I$. In this case the preconditioning matrix is just

$$\mathcal{P}_c = \begin{bmatrix} I & B^T \\ B & O \end{bmatrix},$$

i.e., the augmented matrix for a least squares problem of the form

$$\min_z \|h - B^T z\|_2 ,$$

and application of the preconditioner within a Krylov subspace method involves repeated solution of such problem, either explicitly or implicitly. More generally, solution of linear systems with coefficient matrix $\mathcal{P}_c$ given by (10.15) can be accomplished based on the identity

$$\begin{bmatrix} G & B^T \\ B & O \end{bmatrix}^{-1} = \begin{bmatrix} I & -G^{-1}B^T \\ O & I \end{bmatrix} \begin{bmatrix} G^{-1} & O \\ O & -(BG^{-1}B^T)^{-1} \end{bmatrix} \begin{bmatrix} I & O \\ -BG^{-1} & I \end{bmatrix} \tag{10.16}$$

If G is sufficiently simple (e.g., diagonal) and linear systems with coefficient matrix $BG^{-1}B^T$ can be solved efficiently, then the action of the preconditioner on a vector can be computed at a reasonable cost. In many cases, however, the "exact" solution of linear systems with coefficient matrix $BG^{-1}B^T$ is too expensive, and approximations must be used. For instance, incomplete factorizations or sparse approximate inverse techniques can be used to approximately form and invert $BG^{-1}B^T$; see [107, 258, 391, 463]. These strategies are often more efficient than methods based on a direct LDL^T factorization of $\mathcal{P}_c$. However, a sparse factorization of $\mathcal{P}_c$ would be advantageous when B contains a dense column, as it sometimes happens in optimization problems [50], since in this case $BG^{-1}B^T$ is completely dense. Recently, an interesting new factorization (alternative to (10.16)) has been introduced by Schilders; see [138] and section 10.4 below. This can also be used to efficiently compute the action of $\mathcal{P}_c^{-1}$.

For a symmetric positive definite G, the (left) preconditioned matrix is given by

$$\mathcal{M} = \begin{bmatrix} G & B^T \\ B & O \end{bmatrix}^{-1} \begin{bmatrix} A & B^T \\ B & O \end{bmatrix} = \begin{bmatrix} (I-\Pi)\,G^{-1}A + \Pi & O \\ X & I \end{bmatrix} \tag{10.17}$$

where $\Pi = G^{-1}B^T(BG^{-1}B^T)^{-1}B$ is the G-orthogonal projector onto range$\,(G^{-1}B^T)$ and $X = (BG^{-1}B^T)^{-1}B(G^{-1}A - I)$. Note that in the special case $m = n$, it is $\Pi = I$ and all the eigenvalues of the preconditioned matrix are equal to one; furthermore,

$$\mathcal{M} - I = \begin{bmatrix} O & O \\ X & O \end{bmatrix} \Rightarrow (\mathcal{M} - I)^2 = O\,.$$

Therefore the minimal polynomial of the preconditioned matrix has degree 2, and GMRES applied to the preconditioned system delivers the solution in at most 2 steps, independently of A and G. Of course, if $m = n$ and B is nonsingular then the solution of the saddle point system is simply given by $x_* = B^{-1}g$ and $y_* = B^{-T}(f - Ax_*)$, and the augmented system formulation is neither necessary nor recommended.

In the following, we deal with the more interesting case $m < n$. The eigenvalues and eigenvectors of the preconditioned matrix $\mathcal{M}$ in this case have been studied in several publications, e.g. [153, 340, 391, 416]. The following result was given in [298, Theorems 2.1 and 2.3].

THEOREM 10.1. *Assume that $A \in \mathbb{R}^{n\times n}$ is symmetric, $B \in \mathbb{R}^{m\times n}$ is of full rank ($= m < n$), and that $\mathcal{A}$ is nonsingular. Denote by $Z \in \mathbb{R}^{n\times(n-m)}$ a matrix whose columns form a basis for* $\ker(B)$. *Furthermore, let $G \in \mathbb{R}^{n\times n}$ with $G = G^T \neq A$ be such that $\mathcal{P}_c$ in (10.15) is nonsingular. Then the preconditioned matrix $\mathcal{M}$ in (10.17) has*

1. *the eigenvalue one with multiplicity $2m$;*
2. *$n - m$ eigenvalues that are defined by the generalized eigenproblem $Z^TAZx_z = \lambda Z^TGZx_z$.*

Assume, in addition, that Z^TGZ is positive definite. Then $\mathcal{M}$ has the following $m + i + j$ linearly independent eigenvectors:

1. *m eigenvectors of the form $[0^T, y^T]^T$ corresponding to the eigenvalue one of $\mathcal{M}$;*
2. *i ($0 \le i \le n$) eigenvectors of the form $[w^T, y^T]^T$ corresponding to the eigenvalue one of $\mathcal{M}$, where the components w arise from the generalized eigenproblem $Aw = Gw$;*
3. *j ($0 \le j \le n - m$) eigenvectors of the form $[x_z^T, 0^T, y^T]^T$ corresponding to the eigenvalues of $\mathcal{M}$ not equal to one, where the components x_z arise from the generalized eigenproblem $Z^TAZx_z = \lambda Z^TGZx_z$ with $\lambda \neq 1$.*

Proof. See [298]. □

As pointed out in [298], if either Z^TAZ or Z^TGZ are positive definite, then the indefinite preconditioner $\mathcal{P}_c$ (10.15) applied to the indefinite saddle point matrix $\mathcal{A}$ yields a preconditioned matrix $\mathcal{M}$ (10.17) which has real eigenvalues. Clearly, the better G approximates A, the more clustered near one will be the eigenvalues not already equal to one. Spectral properties of the preconditioned matrices in the presence of inexact solves with G and $BG^{-1}B^T$ have been studied in [391, 463]. We further mention the approach proposed in [117], where the blocks B and B^T in the constraint preconditioner $\mathcal{P}_c$ are replaced by approximations $\bar{B}$, $\bar{B}^T$ in order to simplify solves involving the preconditioning matrix. In [117], $\bar{B}$ is obtained from B by sparsification; in this case, care must be taken in order to avoid a (nearly) rank deficient approximation.

If, in the notation of Theorem 10.1, Z^TGZ is positive definite, then for each initial residual r_0, $\dim \mathcal{K}_{n+m}(\mathcal{M}, r_0) \leq n-m+2$, see, e.g., [340, Theorem 3.4], [298, Theorem 3.5]. Hence the GMRES method applied to the preconditioned system with matrix $\mathcal{M}$ will terminate after at most $n-m+2$ steps. This is a very attractive feature of constraint preconditioning in case $n-m$ is small.

In many applications $n-m+2$ is quite large, so that a total number of $n-m+2$ iterations may be prohibitive. In addition, one typically is interested only in an approximate rather than the exact solution, which means that the iteration can be stopped after a prescribed tolerance is achieved. Then a convergence analysis of the type described in section 9 is required. Unfortunately, the matrix $\mathcal{M}$ is not diagonalizable, so that standard convergence estimates like (9.9) are not applicable. In some cases, however, it is still possible to derive convergence bounds similar to (9.9) for the GMRES method applied to the preconditioned system. For example, assume that $A = A^T$ is positive definite, B has full rank, and $G = I$. Then for a special initial residual of the form $r_0 = [s_0^T, 0^T]^T$, it can be shown that the convergence of GMRES for the *right* constraint preconditioned system only depends on the (1,1) block of the preconditioned matrix $\mathcal{A}\mathcal{P}_c^{-1}$. In our notation this (1,1) block is given by the (diagonalizable) matrix $A(I-\Pi)+\Pi$; see [416, Section 6] for details.

It should be pointed out that application of the GMRES method might not be the best approach to solve the constraint preconditioned linear system. The fact that constraint preconditioning incorporates a projection onto $\ker(B)$ suggests that this preconditioning technique is closely related to the null space method discussed in section 7. The two methods are, in fact, mathematically equivalent [228, 340, 392]. Exploitation of this equivalence allows the construction of a conjugate gradient method for a constraint preconditioned positive definite reduced system [228]. Unlike the GMRES method (when applied to a system with the nonsymmetric matrix $\mathcal{M}$), this conjugate gradient method is based on efficient three-term recurrences. In the numerical experiments in [298], this preconditioned conjugate gradient method represents a viable alternative to a direct solver (here MA27 from the Harwell subroutine library [150]), even for systems of relatively small order, and it typically outperforms the GMRES method applied to $\mathcal{M}$.

All numerical experiments in [298] are performed using test problems arising in linear and nonlinear optimization taken from the *constrained and unconstrained testing environment (CUTE)* [66]. Another situation where constraint preconditioning performs well is when the saddle point system arises from a mixed finite element formulation of a second order, elliptic PDE. In this case it is often possible to find an easily invertible (e.g., diagonal) G which is spectrally equivalent to A. In many cases of practical interest, $G = I$ will do. The resulting constraint preconditioner (10.15) leads to asymptotic rates of convergence for the *right* preconditioned system that are independent of h, the mesh discretization parameter, when the initial guess $u_0 = [0^T, g^T]^T$ (meaning that r_0 is of the form $r_0 = [s_0^T, 0^T]^T$) is used. Similarly to the analysis in [416] mentioned above, the convergence of GMRES then only depends on the (diagonalizable) matrix $A(I-\Pi)+\Pi$. The eigenvalues of this matrix are either equal to one, or contained in an interval of the form $[\alpha_0, \alpha_1]$, $0 < \alpha_0 \leq \alpha_1 < 1$, where both α_0 and α_1 are independent of h [391, Theorem 1] (also cf. [172]). Furthermore, the

condition number of the eigenvector matrix of $A(I-\Pi)+\Pi$ is bounded independently of h [391, Proposition 6]. The asymptotic optimality of these methods, however, does not always translate in computational efficiency. As shown in [391] it may be much faster, in terms of CPU times, to replace the exact (but costly) preconditioner solves with inexact ones, even if this introduces a dependency on h in the number of preconditioned iterations. This is especially true of saddle point systems that arise from the solution of three-dimensional PDE problems. See also [361] for a discussion of inexact constrained preconditioning to mixed formulations of biharmonic problems in 2D.

Some of these results have been extended to the case where both A and G are allowed to be nonsymmetric, see [91]. The case where $B_1 \neq B_2$ has been investigated in [280, 497]. Dyn and Ferguson [156] study the case where A is symmetric but G may be nonsymmetric, and $B_1 = B_2 = B$. Assuming that A is symmetric positive semidefinite with $\ker(A) \cap \ker(B) = \{0\}$ and that G is nonsingular and such that $G + G^T - A$ is positive definite, they show that the matrix $\mathcal{P}_c$ in (10.15) is nonsingular, and that the stationary iteration

$$\begin{bmatrix} G & B^T \\ B & O \end{bmatrix} \begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} G-A & O \\ O & O \end{bmatrix} \begin{bmatrix} x_k \\ y_k \end{bmatrix} + \begin{bmatrix} f \\ g \end{bmatrix}, \quad k = 0, 1, \ldots \tag{10.18}$$

is convergent for any choice of the initial guess. Therefore, the spectrum of $\mathcal{P}^{-1}\mathcal{A}$ is contained in the disk $D(1,1) = \{z \in \mathbb{C}\,;\, |z-1| < 1\}$. A matrix splitting of the form $A = G - (G - A)$ such that $G + G^T - A$ is positive definite is called a *P-regular splitting*, see [377, page 122]. Examples of such splittings when A is symmetric positive semidefinite with a nonzero diagonal include the damped Jacobi, SOR, and SSOR splittings [156]. We mention here that variants of the coupled iteration (10.18) have been used as smoothers for multigrid methods; see [71, 318, 488, 513] and section 11 below.

The important case where A is nonsymmetric but has positive definite symmetric part has been studied in [224], motivated by the Oseen (linearized Navier–Stokes) equations, and more recently in [28], motivated by particulate flow problems. In both of these papers, the idea is to use $G = H \equiv \frac{1}{2}(A + A^T)$, the symmetric part of A. For the nonsymmetric linear system arising from the discretization of the Oseen equations (2.7)–(2.9), the preconditioning matrix becomes the (symmetric) matrix of the corresponding Stokes system (2.10)–(2.12). When the kinematic viscosity coefficient ν in (2.7) is not too small, A is not too far from being symmetric and this preconditioning strategy is very effective, yielding convergence rates which are independent of the mesh size. In practice the exact solution of the preconditioning equation

$$\mathcal{P}_c z = r, \quad \mathcal{P}_c = \begin{bmatrix} H & B^T \\ B & O \end{bmatrix} \tag{10.19}$$

may be too expensive, and inexact solves may be required. For example, (10.19) may be solved approximately by an iterative method, leading to a nested (or inner-outer) iterative scheme. As shown in [28], it is often possible in this way to retain the excellent convergence properties of the outer scheme, while at the same time reducing the work per iteration significantly. On the other hand, this approach does not work well when A is far from symmetric (case of large Reynolds number, or small ν in (2.7)); see the numerical experiments reported in [48, 68]. In this case, better results can be obtained with preconditioners of the form

$$\mathcal{P}_c = \begin{bmatrix} T & B^T \\ B & O \end{bmatrix}$$

where the submatrix T is nonsymmetric positive definite with a "large" skew-symmetric part which incorporates information from K, the skew-symmetric part of A; see [68].

The case where $A = A^T$, $B_1 = B_2 = B$ and $C = C^T \neq O$ has been considered in [26, 50, 391, 463, 514]. Here A and C are positive semidefinite and G is chosen so that the preconditioning matrix

$$\mathcal{P}_c = \begin{bmatrix} G & B^T \\ B & -C \end{bmatrix} \tag{10.20}$$

is nonsingular. Furthermore, linear systems with coefficient matrices G and $C + BG^{-1}B^T$ should be easy to (approximately) solve. The spectral properties of this preconditioner have been studied in [26, 391, 463]. Here we mention the following results from [26].

LEMMA 10.1. *Let B, C, E be real matrices of order $m \times n$, $m \times m$ and $n \times n$, respectively, where B has full rank ($= m$), $C = C^T$ is positive semidefinite and $E = E^T$. Then the eigenvalues of the generalized eigenproblem*

$$\gamma \begin{bmatrix} I & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} E & O \\ O & O \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}, \quad ||x|| + ||y|| \neq 0\,, \tag{10.21}$$

where $x \in \mathbb{C}^n$ and $y \in \mathbb{C}^m$, satisfy

(i) $\gamma = \frac{x^H E x}{x^H (I + B^T C^{-1} B) x} \neq 0$ *if $Ex \neq 0$ and C is positive definite;*
(ii) $\gamma = 0$ *if and only if $Ex = 0$, $y \neq 0$ and the dimension of the eigenspace corresponding to the zero eigenvalue is $m + q$, where $q = \dim \ker(E)$;*
(iii) the nonzero eigenvalues are contained in the interval $\min\{0, \lambda_{\min}(E)\} \leq \gamma \leq \lambda_{\max}(E)$.

Consider now a symmetric saddle point matrix $\mathcal{A}$ and the preconditioner $\mathcal{P}_c$ given by (10.20), with G symmetric and positive definite. The eigenvalues of the preconditioned matrix $\mathcal{P}_c^{-1}\mathcal{A}$ are those of the generalized eigenproblem $\mathcal{A}u = \lambda \mathcal{P}_c u$; but these are of the form $\lambda = \gamma + 1$ where γ are the eigenvalues of the generalized eigenproblem

$$\gamma \begin{bmatrix} G & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} A - G & O \\ O & O \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$

The latter generalized eigenproblem is equivalent to

$$\gamma \begin{bmatrix} I & \hat{B}^T \\ \hat{B} & -C \end{bmatrix} \begin{bmatrix} \hat{x} \\ y \end{bmatrix} = \begin{bmatrix} \hat{E} & O \\ O & O \end{bmatrix} \begin{bmatrix} \hat{x} \\ y \end{bmatrix}, \tag{10.22}$$

where $\hat{B} = BG^{-1/2}$, $\hat{E} = G^{-1/2} A G^{-1/2} - I$ and $\hat{x} = G^{-1/2} x$. This generalized eigenproblem is of the form (10.21) and therefore Lemma 10.1 can be applied. In particular, the eigenvalues of $\mathcal{P}_c^{-1}\mathcal{A}$ are real. Clearly, the better G approximates A, the more clustered around 1 the eigenvalues of $\mathcal{P}_c^{-1}\mathcal{A}$.

Another form of indefinite preconditioning is based on the observation that "stabilized" saddle point systems (i.e., systems of the type (1.6) with $C \neq O$) are generally easier to solve than standard saddle point systems (i.e., systems with $C = O$), in the sense that iterative methods tend to converge faster when $C \neq O$. It is also easier to construct effective block preconditioners when $C \neq O$, since the (negative) Schur complement $C + BA^{-1}B^T$ is typically better conditioned than $BA^{-1}B^T$. This observation justifies the use of preconditioners of the form (10.20), with a suitably chosen C, even the $(2,2)$ block is zero in the original problem. This approach, referred to as *regularized preconditioning*, was first considered in [24] in the special case $G = A$ and $C = \varepsilon I$ where $\varepsilon > 0$. The preconditioning equation itself, $\mathcal{P}_c z = r$, is solved by an iterative method, leading again to an inner-outer process.

The case where $G = A$ is symmetric positive definite, $B_1 = B_2 = B$ has full rank, and C is symmetric positive semidefinite is easily analyzed. In this case we have

$$\mathcal{A} - \mathcal{P}_c = \begin{bmatrix} O & O \\ O & C \end{bmatrix} \;\Rightarrow\; I - \mathcal{A}^{-1}\mathcal{P}_c = \mathcal{A}^{-1}(\mathcal{A} - \mathcal{P}_c) = \begin{bmatrix} O & Y \\ O & S^{-1}C \end{bmatrix},$$

where $S = -BA^{-1}B^T$ and Y is a certain nonzero matrix. It follows that

$$\mathcal{P}_c^{-1}\mathcal{A} = \begin{bmatrix} I & \hat{Y} \\ O & (I - S^{-1}C)^{-1} \end{bmatrix},$$

where $\hat{Y}$ is again a nonzero matrix. Therefore the spectrum of the preconditioned matrix $\mathcal{P}_c^{-1}\mathcal{A}$ is real and contained in the interval $(\beta, 1]$ where

$$\beta = \frac{1}{1 + \lambda_{\max}(-S^{-1}C)} \geq \frac{1}{1 + ||S^{-1}||\,||C||}.$$

Also, $\mathcal{P}_c^{-1}\mathcal{A}$ has the eigenvalue 1 with multiplicity $n + p$, where $p = \dim\ker(C)$. This approach is especially efficient for the Stokes problem, for which S is spectrally equivalent to the identity. Then $||S^{-1}|| \approx 1$ and as long as $||C||$ is bounded independently of h, the spectrum of $\mathcal{P}_c^{-1}\mathcal{A}$ also remains bounded as $h \to 0$ and fast convergence of the preconditioned iteration can be expected, independent of mesh size. In practice there is no need to invert $\mathcal{P}_c^{-1}$ exactly: an approximate solution will suffice and will be more efficient. Spectral bounds for the inexact case can be found in [137]. It should be mentioned that regularization of indefinite (constraint) preconditioners has also been used with good results in optimization problems, see [50] as well as [153, 154]; the latter papers also include results for a parallel implementation.

Concerning the choice of a Krylov subspace method to use with constrained preconditioning, we distinguish between the symmetric case ($A = A^T$, $B_1 = B_2$, $C = C^T$) and the nonsymmetric case. In the nonsymmetric case it is generally necessary to use a nonsymmetric Krylov subspace method, such as GMRES or Bi-CGSTAB, regardless of whether the constraint preconditioner used is symmetric or not. In the case of a symmetric $\mathcal{A}$ with a symmetric constraint preconditioner $\mathcal{P}_c$, on the other hand, there are alternatives to the use of nonsymmetric Krylov subspace methods. Although the preconditioned system matrix $\mathcal{M}$ itself is nonsymmetric, the SQMR method can be used; see the discussion at the end of section 9, in particular Table 9.1. Furthermore, as described above, in some cases a combination of the null space method and the conjugate gradient method may be very effective. As was noted in [72, 340], in the symmetric case it is sometimes even possible to successfully use the conjugate gradient method for the preconditioned system with $\mathcal{M}$. However, the indefiniteness of both $\mathcal{P}_c$ and $\mathcal{A}$ does in general not lead to a robust algorithm, so that safeguard strategies have to be applied to overcome potential breakdowns. For the case of right constraint preconditioning and assuming $g = 0$ in (1.3), such strategies are discussed in detail in [340, 416]. When such strategies are applied, the preconditioned conjugate gradient method can be competitive with direct solution methods, particularly for problems from non-linear equality constrained optimization [340, 341]. Finally, use of the conjugate gradient method with an inexact form of regularized constraint preconditioning has been rigorously justified in [137].

10.3. Hermitian/Skew-Hermitian preconditioning. The Hermitian and Skew-Hermitian Splitting (HSS) has been introduced as a stationary iterative method in [29], where it was shown to converge for non-Hermitian positive definite systems, i.e., linear systems $Ax = b$ with $A + A^*$ positive definite. In the real case (which is the only one we consider here), such systems are said to be *positive real*. Problems of this type arise, for instance, in the numerical solution of convection-diffusion equations. The use of HSS as a preconditioner for rather general saddle point problems has been studied in [45, 46, 443].

The HSS preconditioner is based on the nonsymmetric formulation (3.10). Here we are under the assumptions of Theorem 3.6. In particular, $B_1 = B_2 = B$ and $C = C^T$. Letting $H \equiv \frac{1}{2}(A + A^T)$ and $K \equiv \frac{1}{2}(A - A^T)$ we have the following splitting of $\hat{\mathcal{A}}$ into its symmetric and skew-symmetric

part:

$$\hat{\mathcal{A}} = \begin{bmatrix} A & B^T \\ -B & C \end{bmatrix} = \begin{bmatrix} H & O \\ O & C \end{bmatrix} + \begin{bmatrix} K & B^T \\ -B & O \end{bmatrix} = \mathcal{H} + \mathcal{K}\,. \tag{10.23}$$

Note that $\mathcal{H}$, the symmetric part of $\hat{\mathcal{A}}$, is symmetric positive semidefinite since both H and C are. Let $\alpha > 0$ be a parameter. Similar in spirit to the classical ADI (Alternating-Direction Implicit) method [479], we consider the following two splittings of $\hat{\mathcal{A}}$:

$$\hat{\mathcal{A}} = (\mathcal{H} + \alpha\mathcal{I}) - (\alpha\mathcal{I} - \mathcal{K}) \qquad \text{and} \qquad \hat{\mathcal{A}} = (\mathcal{K} + \alpha\mathcal{I}) - (\alpha\mathcal{I} - \mathcal{H})\,.$$

Here $\mathcal{I}$ denotes the identity matrix of order $n + m$. The stationary HSS iteration is obtained by alternating between these two splittings. Given an initial guess u_0, the HSS iteration computes a sequence $\{u_k\}$ as follows:

$$\begin{cases} (\mathcal{H} + \alpha\mathcal{I})\, u_{k+\frac{1}{2}} = (\alpha\mathcal{I} - \mathcal{K})\, u_k + \hat{b}, \\ (\mathcal{K} + \alpha\mathcal{I})\, u_{k+1} = (\alpha\mathcal{I} - \mathcal{H})\, u_{k+\frac{1}{2}} + \hat{b}\,. \end{cases} \tag{10.24}$$

Note that both $\mathcal{H} + \alpha\mathcal{I}$ and $\mathcal{K} + \alpha\mathcal{I}$ are invertible. The first matrix is symmetric positive definite while second one is a shifted skew-symmetric matrix with eigenvalues of the form $\alpha + i\nu_i$, where $\nu_i \in \mathbb{R}$ for all $i = 1, \ldots, n + m$.

The two-step process (10.24) can be written as a fixed point iteration by eliminating the intermediate vector $u_{k+\frac{1}{2}}$, yielding

$$u_{k+1} = \mathcal{T}_\alpha u_k + c\,. \tag{10.25}$$

Here

$$\mathcal{T}_\alpha := (\mathcal{K} + \alpha\mathcal{I})^{-1}(\alpha\mathcal{I} - \mathcal{H})(\mathcal{H} + \alpha\mathcal{I})^{-1}(\alpha\mathcal{I} - \mathcal{K})$$

is the iteration matrix of the method, and $c := (\mathcal{K} + \alpha\mathcal{I})^{-1}[\mathcal{I} + (\alpha\mathcal{I} - \mathcal{H})(\mathcal{H} + \alpha\mathcal{I})^{-1}]\hat{b}$. The fixed point iteration (10.25) converges for arbitrary initial guesses u_0 and right-hand sides $\hat{b}$ to the solution $u = \hat{\mathcal{A}}^{-1}\hat{b}$ if and only if $\rho(\mathcal{T}_\alpha) < 1$, where $\rho(\mathcal{T}_\alpha)$ denotes the spectral radius of $\mathcal{T}_\alpha$. It follows from the results in [29] that when $\mathcal{H}$ is positive definite, the stationary iteration (10.24) converges for all $\alpha > 0$ to the solution of $\hat{\mathcal{A}}\, u = \hat{b}$. For saddle point problems, $\mathcal{H}$ is positive definite if and only if H and C are. These conditions are rather restrictive in practice, especially the one on C. However, it was shown in [46] that the HSS iteration converges for all $\alpha > 0$ if H is positive definite, B is of full rank and C is positive semidefinite (possibly zero). The method can be made to converge even when H is positive semidefinite and singular, provided that $\ker(H) \cap \ker(B) = \{0\}$, see [46].

It was shown in [29] that when $\mathcal{H}$ is positive definite, the choice

$$\alpha = \sqrt{\lambda_{\min}(\mathcal{H})\lambda_{\max}(\mathcal{H})}$$

minimizes an upper bound on the spectral radius of the iteration matrix $\mathcal{T}_\alpha$. Unfortunately, in most saddle point problems it is $\lambda_{\min}(\mathcal{H}) = 0$, therefore the result does not apply. Furthermore, the rate of convergence of the HSS iteration is rather slow, even with the "optimal" choice of α. For these reasons it was proposed in [46] that GMRES or other Krylov subspace methods should be used to accelerate the convergence of the HSS method. In other words, the HSS method is best used as a preconditioner for (say) GMRES rather than as a stationary iterative method. As observed in [46], there is a unique splitting $\hat{\mathcal{A}} = \mathcal{P} - \mathcal{Q}$ with $\mathcal{P}$ nonsingular such that the iteration matrix $\mathcal{T}_\alpha$ is the

matrix induced by that splitting, i.e., $\mathcal{T}_\alpha = \mathcal{P}^{-1}\mathcal{Q} = \mathcal{I} - \mathcal{P}^{-1}\mathcal{A}$. An easy calculation shows that the HSS iteration (10.24) can be written in *correction form* as

$$u_{k+1} = u_k + \mathcal{P}_\alpha^{-1} r_k, \quad r_k = \hat{b} - \hat{\mathcal{A}} u_k \,,$$

where the preconditioner $\mathcal{P}$ is given by

$$\mathcal{P} \equiv \mathcal{P}_\alpha = \tfrac{1}{2\alpha}(\mathcal{H} + \alpha\mathcal{I})(\mathcal{K} + \alpha\mathcal{I}) \,. \tag{10.26}$$

Note that as a preconditioner we can use $\mathcal{P}_\alpha = (\mathcal{H} + \alpha\mathcal{I})(\mathcal{K} + \alpha\mathcal{I})$ instead of the expression given in (10.26), since the factor $\frac{1}{2\alpha}$ has no effect on the preconditioned system. It is just a normalization factor that allows us to conclude that the eigenvalues of the preconditioned matrix $\mathcal{P}_\alpha^{-1}\hat{\mathcal{A}}$ (or $\hat{\mathcal{A}}\,\mathcal{P}_\alpha^{-1}$, which has the same spectrum) are all contained in the disk $D(1,1) = \{z \in \mathbb{C}\,;\, |z-1| < 1\}$. In particular, the spectrum of the preconditioned matrix, like that of $\hat{\mathcal{A}}$, lies entirely in the right-half plane: the preconditioned matrix is positive stable.

The rate of convergence of nonsymmetric Krylov iterations, such as GMRES, preconditioned by $\mathcal{P}_\alpha$ depends on the particular choice of α. Finding the value of α that optimizes the rate of convergence is a very difficult problem in general. Note that the value of α that minimizes the number of GMRES iterations may be quite different from the one that minimizes the spectral radius of $\mathcal{T}_\alpha$, see [45]. Numerical experiments show that with an appropriate scaling of the system (such that the nonzero diagonal entries of $\hat{\mathcal{A}}$ are equal to 1), there is a unique value α_* of α for which the number of preconditioned iterations is minimized, and this α_* is usually a small number, between 0 and 1. In some cases, but not always, the optimal α can be determined by trial-and-error on a small example (e.g., a coarse discretization of the continuous problem to be solved) and then used with good results on larger problems corresponding to finer discretizations; see [49], where HSS preconditioning was used to solve weighted least squares problems arising in image processing.

Spectral properties of the preconditioned matrix as a function of α have been studied, under different sets of assumptions, in [45, 49, 443]. A Fourier analysis of HSS preconditioning for saddle point formulations of Poisson's equation (including the anisotropic case) was given in [45]. The analysis showed that using a sufficiently small value of α results in h-independent convergence. Furthermore, as $\alpha \to 0$ the eigenvalues of the preconditioned matrix are all real and fall in two small intervals $(0, \varepsilon_1)$ and $(2-\varepsilon_2, 2)$, with $\varepsilon_1, \varepsilon_2 > 0$ and $\varepsilon_1, \varepsilon_2 \to 0$ as $\alpha \to 0$. This clustering result was generalized in [443] to general saddle point systems with $A = A^T$ positive definite and $C = O$, using purely algebraic arguments. However, h-independent convergence is not always guaranteed, for example it does not occur for the Stokes problem. Nevertheless, good results have been obtained in the solution of the Oseen problem, where HSS preconditioning appears to be competitive with other preconditioners for large Reynolds number (small ν). Numerical experiments indicate that the optimal value of α for steady 2D problems is largely independent of ν and is approximately given by $h^{1/2}$, where h denotes the mesh size. Furthermore, rapid convergence independently of h and ν is observed for unsteady problems; see the numerical experiments in [44, 48].

Application of the HSS preconditioner within GMRES requires solving a linear system of the form $\mathcal{P}_\alpha z_k = r_k$ at each iteration. This is done by first solving the system

$$(\mathcal{H} + \alpha\mathcal{I})\, w_k = r_k \tag{10.27}$$

for w_k, followed by

$$(\mathcal{K} + \alpha\mathcal{I})\, z_k = w_k \,. \tag{10.28}$$

Recalling the form of $\mathcal{H}$, see (10.23), equation (10.27) consists of two decoupled systems with coefficient matrices $H + \alpha I$ and $C + \alpha I$, respectively. Both matrices are symmetric positive definite,

and a number of efficient methods can be applied, including Cholesky factorization, preconditioned conjugate gradient (PCG) schemes, or multigrid, either geometric or algebraic. Multigrid methods are the solver of choice for a number of problems arising from the discretization of partial differential equations, particularly in the case of the Oseen problem (2.10)–(2.11) in $\Omega \subset \mathbb{R}^3$, where H is a direct sum of three discrete Laplace operators. The solution of the system with matrix $C + \alpha I$ is often much easier and reduces to a scaling by α when $C = O$. When solving regularized weighted least squares problems, both H and C are diagonal and the cost of solving (10.27) is negligible, see [49].

Note that the addition of a positive term α to the main diagonal of H (and C) improves the condition number. This, in turn, tends to improve the rate of convergence of iterative methods applied to (10.27). More precisely, if H is normalized so that its largest eigenvalue is equal to 1, then for the spectral condition number of $H + \alpha I$ we have

$$\kappa\,(H + \alpha I) = \frac{1+\alpha}{\lambda_{\min}(H) + \alpha} \leq 1 + \frac{1}{\alpha},$$

independent of the size of the problem. Note that even a fairly small value of α, such as $\alpha = 0.1$, yields a small condition number ($\kappa\,(H + \alpha I) \leq 11$). For most problems, both multigrid and the conjugate gradient method applied to (10.27) can be expected to converge rapidly, independent of the number n of unknowns.

Solving (10.28) is usually more involved, especially when $K \neq O$. It requires the solution of two coupled linear systems of the form

$$\begin{cases} (\alpha I + K)\, x_{k+1} + B^T y_{k+1} = f_k, \\ -B x_{k+1} + \alpha y_{k+1} = g_k. \end{cases} \tag{10.29}$$

This system can be solved in several ways. One approach is to eliminate x_{k+1} from the second equation using the first one (Schur complement reduction), leading to a smaller (order m) linear system of the form

$$[B(I + \alpha^{-1}K)^{-1}B^T + \alpha^2 I]y_{k+1} = B(I + \alpha^{-1}K)^{-1}f_k + \alpha g_k. \tag{10.30}$$

Once the solution y_{k+1} to (10.30) has been computed, the vector x_{k+1} is given by

$$x_{k+1} = (\alpha I + K)^{-1}(f_k - B^T y_{k+1})\,.$$

When $K = O$, system (10.30) simplifies to

$$(BB^T + \alpha^2 I)\, y_{k+1} = B f_k + \alpha g_k, \tag{10.31}$$

and $x_{k+1} = \alpha^{-1}(f_k - B^T y_{k+1})$. If BB^T is sufficiently sparse, system (10.30) could be formed explicitly and solved by a sparse Cholesky factorization. Otherwise, a PCG iteration with a simple preconditioner not requiring access to all the entries of the coefficient matrix $BB^T + \alpha^2 I$ could be used. If B represents a discrete divergence operator, then $BB^T + \alpha^2 I$ is just a shifted discrete Laplace operator, and many fast solvers for system (10.31) are available. When $K \neq O$ the coefficient matrix in (10.30) is generally dense. An important exception is the Oseen problem in rotation form (2.15)–(2.17). In the 2D case the coefficient matrix has the form

$$\hat{\mathcal{A}} = \begin{bmatrix} A_1 & D & B_1^T \\ -D & A_2 & B_2^T \\ -B_1 & -B_2 & C \end{bmatrix} = \begin{bmatrix} A_1 & O & O \\ O & A_2 & O \\ O & O & C \end{bmatrix} + \begin{bmatrix} O & D & B_1^T \\ -D & O & B_2^T \\ -B_1 & -B_2 & O \end{bmatrix} = \mathcal{H} + \mathcal{K}\,. \tag{10.32}$$

Here A_1, A_2 are discrete Laplace operators with appropriate boundary conditions, $B = \begin{bmatrix} B_1 & B_2 \end{bmatrix}$, $C = C^T$ is a stabilization term that can be assumed to be zero if the discretization used is already

stable, and D is a matrix that corresponds to multiplication by $w = \nabla \times \mathbf{v}$; see section 2.1 for details. For finite difference schemes (like, for instance, MAC [256]) D is a diagonal matrix; for finite elements, it will be a scaled mass matrix. When D is diagonal, the Schur complement $B(I+\alpha^{-1}K)^{-1}B^T+\alpha^2 I$ is a sparse matrix and can be formed explicitly. This follows from the fact that

$$(I+\alpha^{-1}K)^{-1} = \begin{bmatrix} I & \alpha^{-1}D \\ -\alpha^{-1}D & I \end{bmatrix}^{-1} = \begin{bmatrix} E_1 & -E_2 \\ E_2 & E_3 \end{bmatrix},$$

where E_1, E_2 and E_3 are diagonal matrices given by

$$E_1 = I + \alpha^{-2}D(I+\alpha^{-2}D^2)^{-1}D\,, \quad E_2 = \alpha^{-1}D(I+\alpha^{-2}D^2)^{-1}\,, \quad \text{and} \quad E_3 = (I+\alpha^{-2}D^2)^{-1}\,.$$

When D is not diagonal, we can replace it with a spectrally equivalent diagonal approximation and still have a sparse Schur complement; since we are constructing a preconditioner, the action of $(\mathcal{K}+\alpha\mathcal{I})^{-1}$ need not be computed exactly. Hence, the Schur complement $B(I+\alpha^{-1}K)^{-1}B^T+\alpha^2 I$ (or the approximation of it obtained by replacing D with a diagonal matrix) is sparse, and system (10.28) can be efficiently solved via (10.30) using sparse matrix techniques. It is also possible to use multigrid, since the Schur complement can be interpreted as a discretization of a second order, elliptic operator with variable coefficients. In alternative, an ILU-preconditioned GMRES can be used. While we have focused here on the 2D case, the 3D case can be treated along the same lines; see [48].

In alternative to the Schur complement reduction, there are other approaches that can be used to solve linear systems with matrix $\mathcal{K}+\alpha\mathcal{I}$. Note that this is a normal matrix of the form "identity-plus-skew-symmetric." There are various Lanczos-type methods that can be applied to systems of this kind; see [118, 500] and, more generally, [279]. Other approaches to the solution of shifted skew-symmetric systems are studied in [223]. Yet another possibility is to regard (10.29) as a general nonsymmetric system and to use preconditioned GMRES (say). Many of these schemes can benefit from the fact that for even moderate values of $\alpha > 0$, the condition number of $\mathcal{K}+\alpha\mathcal{I}$ is often rather small.

It is important to stress that the linear systems in (10.24) need not be solved exactly. The use of inexact solves was considered in [29] for the positive real case. The upshot is that inexact solves can be used to greatly reduce the cost of each iteration, at the expense of somewhat slower convergence. Typically, in practical implementations, inexact solves result in a much more competitive algorithm. Here we observe that when the alternating scheme is used as a preconditioner for a Krylov method, inexact solves are a natural choice, and there is no theoretical restriction on the accuracy of the inner solves. Inexact solutions are often obtained by iterative methods, leading to an inner-outer scheme; in this case, a flexible method like FGMRES [422] should be used for the outer iteration. Inexact solves may also be performed by means of matrix splittings or incomplete factorizations, see [46]. In this case, standard GMRES can be used for the outer iteration.

Other iterative solvers for the Navier-Stokes equations in rotation form have been introduced and studied in [375, 376]. Although the rotation form does not appear to be widely used in practice, it has some advantages over the (standard) convective form (2.7)–(2.9). As mentioned in [376, page 1685], the numerical solution of the Navier-Stokes equations in rotation form is a topic that deserves further study.

10.4. Approximate and incomplete factorization methods. We conclude this section on preconditioners with a brief discussion of recent attempts to develop effective and robust approximate and incomplete factorizations methods for saddle point matrices.

Approximate factorizations for symmetric saddle point systems have been used in [208] in the context of interior point methods for constrained optimization. Here we assume that $A = A^T$ is positive definite, $B_1 = B_2 = B$ is of full rank, and $C = C^T$ is positive semidefinite (possibly zero). The approach taken in [208] is to define the preconditioner in terms of an *exact* LDL^T factorization of an approximation $\mathcal{P} \approx \mathcal{A}$. More precisely, the idea is to compute a sparse Bunch-Parlett factorization

$$\mathcal{P} = \mathcal{Q}^T \mathcal{L} \mathcal{D} \mathcal{L}^T \mathcal{Q} ,$$

where $\mathcal{P}$ is symmetric indefinite, $\mathcal{Q}$ is a permutation matrix, $\mathcal{L}$ is unit lower triangular, and $\mathcal{D}$ a block diagonal matrix with blocks of dimension 1 and 2. The resulting factorization can be used as a preconditioner with SQMR or with a nonsymmetric Krylov subspace solver. The authors of [208] modify $\mathcal{P}$ to guarantee it is positive definite; the resulting preconditioner can then be used with a symmetric Krylov subspace solver, such as SYMMLQ or MINRES. In order to do so, it is sufficient to compute the eigendecomposition of $\mathcal{D}$ (which can be done very easily), and then to change the sign of the negative eigenvalues of $\mathcal{D}$. If $\bar{\mathcal{D}}$ denotes the resulting block diagonal matrix, the symmetric positive definite preconditioner is

$$\bar{\mathcal{P}} = \mathcal{Q}^T \mathcal{L} \bar{\mathcal{D}} \mathcal{L}^T \mathcal{Q} .$$

The main issue is the choice of the approximation $\mathcal{P} \approx \mathcal{A}$. As usual, a trade-off is involved. On one hand, $\mathcal{P}$ must be sufficiently close to $\mathcal{A}$ so that convergence of the preconditioned iteration will be rapid; on the other, $\mathcal{P}$ must be such that an LDL^T factorization can be computed fast and without too much fill-in. The choice of such $\mathcal{P}$ is clearly problem-dependent. One possibility is to introduce a re-partitioning of $\mathcal{A}$ in the form

$$\mathcal{A} = \begin{bmatrix} A_{11} & A_{12}^T \\ A_{12} & A_{22} \end{bmatrix} , \tag{10.33}$$

where the order of A_{11} does not exceed $n - m$. This condition is necessary in order to have a nonsingular block A_{22}. If A_{11} is strongly diagonally dominant, then a reasonable approximation could be

$$\mathcal{P} = \begin{bmatrix} D_{11} & O \\ O & A_{22} \end{bmatrix} ,$$

where D_{11} denotes the main diagonal of A_{11}. For linear systems arising in interior point methods, it is often possible to find a permutation of the (1,1) block A of $\mathcal{A}$ so that $\mathcal{A}$ can be cast in the form (10.33) with A_{11} having very large entries on the main diagonal. Indeed, when a variable x_i is approaching the boundary of the feasible set, the ith diagonal entry of the Hessian A becomes very large; by numbering these variables first, the saddle point matrix $\mathcal{A}$ can be given the desired form. The block A_{22} is now a smaller saddle point-type matrix. If $m \ll n$, the submatrix A_{22} may be much smaller than $\mathcal{A}$, and a sparse Bunch-Parlett factorization of A_{22} can be computed efficiently. In the context of interior point methods, this preconditioner can be expected to be especially effective in later stages of Newton's iteration, when many of the diagonal entries in A_{11} are large. See [208] for additional discussion and alternatives approaches.

A different approximate factorization method has been presented in [138] in the context of constraint preconditioning. Consider a constrained preconditioner (10.15) partitioned as

$$\mathcal{P}_c = \begin{bmatrix} G_{11} & G_{12} & B_1^T \\ G_{21} & G_{22} & B_2^T \\ B_1 & B_2 & O \end{bmatrix} ,$$

where the block B_1 is nonsingular. Assume we choose matrices $L_1 \in \mathbb{R}^{m\times m}$ and $L_2 \in \mathbb{R}^{(n-m)\times(n-m)}$, with L_2 nonsingular; for instance, $L_2 = I$. Then $\mathcal{P}_c$ can be factorized as follows:

$$\mathcal{P}_c = \begin{bmatrix} B_1^T & O & L_1 \\ B_2^T & L_2 & E \\ O & O & I \end{bmatrix} \begin{bmatrix} D_1 & O & I \\ O & D_2 & O \\ I & O & O \end{bmatrix} \begin{bmatrix} B_1 & B_2 & O \\ O & L_2^T & O \\ L_1^T & E^T & I \end{bmatrix}, \tag{10.34}$$

where $D_1 = B_1^{-T} G_{11} B_1^{-1} - L_1^T B_1^{-1} - B_1^{-T} L_1$, $D_2 = L_2^{-1}(G_{22} - B_2^T D_1 B_2 - E B_2 - B_2^T E^T) L_2^{-T}$ and $E = G_{21} B_1^{-1} - B_2^T D_1 - B_2^T L_1^T B_1^{-1}$. A decomposition of the form (10.34) is known as a *Schilders factorization*. In [138] it is shown how such a factorization can be used to efficiently solve linear systems of the form $\mathcal{P}_c z_k = r_k$ at each iteration of an iterative method. Since (10.34) is an exact factorization of an approximation to $\mathcal{A}$, we can think of it as a rather special approximate factorization of the original coefficient matrix $\mathcal{A}$.

A more standard approach is to compute an *incomplete* factorization of the *exact* matrix $\mathcal{A}$:

$$\mathcal{A} = \mathcal{Q}^T \mathcal{L}\mathcal{D}\mathcal{L}^T \mathcal{Q} + \mathcal{R} \equiv \mathcal{P} + \mathcal{R},$$

where $\mathcal{R}$ represents a remainder matrix that contains terms that have been discarded in the course of the incomplete factorization. The size of $\mathcal{R}$ depends on the permutation matrix $\mathcal{Q}$ and on the dropping strategy used: although levels-of-fill can be used, threshold-based dropping is likely to be far more effective for indefinite systems. Unfortunately, the development of reliable and effective incomplete LDL^T factorizations for highly indefinite systems has turned out to be a very difficult problem, and until recently not much progress has been made in this area. This is in contrast with the symmetric positive definite case and even with the general, nonsymmetric case, for which many successful techniques exist [25, 43, 360, 423, 475].

A possible explanation for the difficulties encountered in developing reliable incomplete factorizations for indefinite systems has been offered by van der Vorst [139, pages 198–199]. When the coefficient matrix $\mathcal{A}$ is highly indefinite, it has many eigenvalues on both sides of the imaginary axis. The eigenvalues of the preconditioned matrix $\mathcal{P}^{-1}\mathcal{A}$ depend continuously on the entries of $\mathcal{P}$. For $\mathcal{P} = I$, they coincide with the eigenvalues of $\mathcal{A}$; for $\mathcal{P} = \mathcal{A}$, they are all equal to 1. As the preconditioner $\mathcal{P}$ approaches $\mathcal{A}$, for instance by allowing more and more fill-in in the incomplete factors, the eigenvalues of $\mathcal{P}^{-1}\mathcal{A}$ approach the value 1; in particular, the negative eigenvalues, *en route* to 1, have to cross the imaginary axis. There is a high risk that some (perhaps many) of these eigenvalues will come arbitrarily close to the origin. Hence, "improving" the preconditioner (by allowing additional fill-in) may actually cause the preconditioned matrix to become very close to singular, which in turn may cause the preconditioned iteration to converge more slowly or even fail. This non-monotonic behavior of incomplete factorization preconditioners with respect to fill-in has been often observed in numerical experiments with indefinite matrices. Moreover, serious difficulties are often met in the course of computing an incomplete factorization, owing to various types of instabilities originating from the highly indefinite and non-diagonally dominant nature of saddle point matrices [108].

It is plausible that using the alternative, nonsymmetric positive definite form (3.10) of the saddle point system may alleviate this problem, particularly when $C \neq O$; however, we we are not aware of any experimental studies in this direction. Another possibility would be to ignore the structure or symmetry of $\mathcal{A}$ altogether and apply one of the numerous, time-tested incomplete LU (ILU) factorization algorithms that have been developed for general sparse matrices, combined with some form of pivoting to promote stability. However, since fill-in tends to be very heavy with the original ordering of $\mathcal{A}$, large numbers of fill-ins have to be discarded, often resulting in preconditioners of low quality. Band-reducing and sparsity-preserving symmetric orderings (such as reverse Cuthill-McKee, minimum degree or nested dissection, see [145]) are of limited use here and often produce unstable pivot sequences.

Some degree of success has been achieved through the use of nonsymmetric permutations and scalings aimed at increasing the diagonal dominance of the coefficient matrix; see [47, 258, 259], where the HSL MC64 preprocessing subroutines [147, 148] have been used in combination with various ILU and sparse approximate inverse techniques. When applied to saddle point matrices with $C = O$, these permutations produce a matrix with a zero-free diagonal; moreover, the zero (or small) diagonal entries are replaced by (relatively) large nonzeros. The net effect of this preprocessing is that stable ILU-type factorizations can now be computed in most cases. Unfortunately, this preprocessing destroys the symmetry and other structural properties of $\mathcal{A}$, and may lead to unnecessarily high storage demands in some cases.

Very recently, some new approaches have been introduced for constructing sparse incomplete LDL^T factorizations of symmetric indefinite matrices [180, 196, 252, 323, 403, 404, 405]. While not necessarily targeted to indefinite matrices in saddle point form, these techniques may be quite effective when applied to such systems; see the results reported in [252, 323]. In particular, maximum weighted matching techniques have been used in [252] to minimize the need for dynamic (Bunch-type [83]) pivoting in the course of the factorization [149]. Such matching techniques can be regardes as symmetric analogues of the nonsymmetric permutations used in [147, 148] to permute large entries to the diagonal. Here the purpose is to preprocess the matrix in a symmetry-preserving fashion with the goal of creating "large" 2×2 diagonal blocks, thus enabling a more stable (incomplete) LDL^T factorization. In [323], stability is promoted instead by a different kind of preprocessing, referred to as *Bunch equilibration*, see [82]. These new preconditioners appear to be reasonably efficient and robust. In particular, one of the approaches presented in [252] appears to be especially well-suited for saddle point matrices. Since no comparison between these general purpose methods and more specialized preconditioners for saddle point problems has been carried out yet, it is not yet clear how competitive this class of preconditioners really is. Nevertheless, there is certainly a strong interest in the engineering community for general-purpose techniques (and software) that can be applied with little or no modifications to a wide range of problems, even though such preconditioners may not be the best for any particular problem.

Some work has been done towards developing incomplete factorization preconditioners specifically designed for saddle point matrices. Unlike general-purpose methods, these approaches take into account the block structure of $\mathcal{A}$. One approach is based on the following observation [510, 411]. If $A = A^T$ is positive definite, $B_1 = B_2 = B$ is of full rank and $C = C^T$ is positive semidefinite (possibly zero), then

$$\mathcal{A} = \begin{bmatrix} A & B^T \\ B & -C \end{bmatrix} = \begin{bmatrix} L_{11} & O \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} L_{11}^T & L_{21}^T \\ O & -L_{22}^T \end{bmatrix} = \mathcal{L}\mathcal{U}\,, \tag{10.35}$$

where $A = L_{11}L_{11}^T$ is the Cholesky factorization of A, $L_{21} = BL_{11}^{-T}$, and $-S = C + L_{21}L_{21}^T = L_{22}L_{22}^T$ is the Cholesky factorization of the (negative) Schur complement. Note that (10.35) is a triangular, Cholesky-like factorization of $\mathcal{A}$; such a factorization always exists, without the need for pivoting (cf. section 7. If we could use $\mathcal{P} = \mathcal{L}\,\mathcal{L}^T$ as a (split) preconditioner, it is easily verified that

$$\mathcal{L}^{-1}\mathcal{A}\mathcal{L}^{-T} = \begin{bmatrix} I & O \\ O & -I \end{bmatrix},$$

and a symmetric Krylov subspace algorithm like MINRES or SYMMLQ would deliver the solution in at most two iterations. In practice, the exact factor $\mathcal{L}$ is replaced by an incomplete one, as follows. First, an incomplete Cholesky factorization $A \approx \bar{L}_{11}\bar{L}_{11}^T$ is computed; several stable and efficient algorithms exist for this task. Next, we compute a sparse approximation $\bar{L}_{21}$ to L_{21} by solving a matrix equation of the form $\bar{L}_{11}^T X = B^T$ using backsubstitutions; some form of dropping may be needed to preserve sparsity in $\bar{L}_{21}$. Finally, an incomplete Cholesky factorization $C + \bar{L}_{21}\bar{L}_{21}^T \approx \bar{L}_2\bar{L}_2^T$

is computed. The resulting incomplete factor

$$\bar{\mathcal{L}} = \begin{bmatrix} \bar{L}_{11} & O \\ \bar{L}_{21} & \bar{L}_{22} \end{bmatrix}$$

and its transpose can be used to define a positive definite, factorized preconditioner $\mathcal{P} = \bar{\mathcal{L}}\,\bar{\mathcal{L}}^T$ for a method like SYMMLQ or MINRES. Numerical experiments in [411] indicate that this can be an effective preconditioning strategy. Moreover, the resulting incomplete Cholesky factorization of the Schur complement ($-S \approx \bar{L}_{22}\bar{L}_{22}^T$) can be used as a preconditioner for Uzawa's method, resulting in a significant acceleration. Clearly, this approach will only work well when the Schur complement S is sparse or can be well approximated by a sparse matrix; see the discussion in section 10.1.3. This approach may be extended to nonsymmetric saddle point systems; in this case, however, the existence of the corresponding incomplete ILU factorizations may not be guaranteed.

Finally, we mention that incomplete factorizations for special classes of nonsymmetric saddle point problems have been developed by Wille and collaborators for solving the linear systems arising from Newton's method applied to mixed finite element discretizations of the steady Navier–Stokes equations; see [128, 501, 502]. Here A is nonsymmetric (but positive definite), B_1 may or may not be equal to B_2, and $C = O$. In this work, fill-in is allowed only in predetermined locations within the factors, corresponding to the pressure block—i.e., in the (2,2) block of the factors. More precisely, the guiding principle is that fill-in is accepted at locations in the global matrix where the nodes belong to the same finite element. The resulting incomplete factorizations appear to be numerically stable, and satisfactory convergence rates are observed for problems in both 2D and 3D. In [502], parallel implementation aspects are also discussed.

11. Multilevel methods. A survey on the numerical solution of saddle point systems would not be complete without some discussion of multilevel solvers. In this section we give a very brief overview of multigrid and domain decomposition methods for saddle point systems arising from PDE problems. The literature on this topic is vast and highly technical, the construction of multilevel solvers usually being tied to a particular discretization and to analytical properties of the coefficients at the continuous (infinite-dimensional) level. Since our main emphasis is on the linear algebra aspects of finite-dimensional saddle point problems, we refrain from going into details and mostly restrict ourselves to providing pointers to the literature.

11.1. Multigrid methods. As already mentioned repeatedly in the course of this survey, multigrid and, more generally, multilevel methods can be used in segregated approaches to solve reduced systems, as in Uzawa-type or block preconditioning schemes. In the incompressible flow setting these subproblems correspond to discretizations of scalar elliptic PDEs for the velocities and/or the pressure field, such as diffusion or convection-diffusion-type problems. Multigrid methods are ideally suited for such problems, often achieving optimal computational complexity, in the sense that the number of operations scales linearly with the number of unknowns. In the context of preconditioning inexact solves are usually sufficient, and it may be enough to perform just one iteration (V-, W- or F-cycle). The choice of multigrid components (smoothers, coarse grid operators and intergrid transfer operators) is well-understood for this type of PDEs, and a plethora of algorithmic and software tools are available to deal with a wide range of discretizations, properties of coefficients, and problem geometries. Standard treatments of multigrid methods can be found in [249, 468, 498]. For detailed discussions of the use of multigrid components in the solution of saddle point problems from finite element discretizations, the reader is referred to [170, 469].

More challenging is the construction of *coupled* multilevel methods, i.e., multilevel methods that are applied to the entire system $\mathcal{A}\,u = b$. This topic has seen extensive development (especially in

the last 15 years) and there are now a number of coupled multilevel methods for incompressible flow problems [71, 77, 160, 388, 478, 486, 498, 499, 503, 504], mixed formulations of second and fourth order elliptic equations and of Maxwell's equations [15, 70, 268, 269, 270, 271, 468, 481], optimization [142], and parameter estimation problems [20]. These multigrid schemes may be used alone or as preconditioners for Krylov subspace methods; see [468, pages 287–288] for a discussion of the *solver vs. preconditioner* question for multigrid.

For simplicity we consider here the two-grid case only. Our discussion, although fairly general, is geared towards Stokes and Oseen-type problems and is based on [133, Section 4] and [499, Chapter 7.6]. Introducing subscripts h and H for the fine and the coarse grid discretization parameters (e.g., $H = 2h$), we rewrite $\mathcal{A}\,u = b$ as $\mathcal{A}_h\,u_h = b_h$, and we use the H subscripts for the coarse grid problem. The restriction and prolongation operators are denoted $\mathcal{I}_H^h$ and $\mathcal{I}_h^H$, respectively. The so-called Full Approximation Storage (FAS) two-grid method consists of the following steps:

(i) Perform ν_1 pre-smoothing steps on $\mathcal{A}_h\,u_h = b_h$;
(ii) Compute the fine grid residual $r_h = b_h - \mathcal{A}_h\,u_h$;
(iii) Apply the restriction to the fine grid residual, $r_H = \mathcal{I}_H^h r_h$;
(iv) Solve the coarse grid problem $\mathcal{A}_H\,u_H = r_H$;
(v) Add the correction $u_h = u_h + \alpha\,\mathcal{I}_h^H u_H$;
(vi) Perform ν_2 post-smoothing steps.

At step (v), the correction control parameter α is either held fix (possibly $\alpha = 1$) or is chosen adaptively so as to minimize an appropriate measure of the error. A truly multilevel algorithm (V-cycle) is obtained by recursively applying the above procedure in step (iv). There are many variants of the basic scheme just described.

The choice of the restriction and prolongation operators has been discussed at length in the above mentioned monographs (e.g., in [469, Chapter 3.4.2]) and will not be pursued here. More critical are the construction of the coarse grid operator $\mathcal{A}_H$ and the choice of the coarse grid solver in step (iv), and the choice of the smoother in steps (i) and (vi). The matrix $\mathcal{A}_H$ can be constructed in at least two ways. One possibility is to simply re-discretize the problem on the coarse grid: this strategy, which is referred to as DCGA (for Discretization Coarse Grid Approximation), may be applied also to nonlinear problems and is frequently used for the Navier-Stokes equations. An alternative is to compute $\mathcal{A}_H$ as $A_H = \mathcal{I}_H^h \mathcal{A}_h \mathcal{I}_h^H$. This choice, which is referred to as GCGA (for Galerkin Coarse Grid Approximation) is restricted to linear problems, but it has the advantage that it can be computed without knowledge of the underlying discretization. Hence, it is the preferred choice for Algebraic Multigrid (AMG) algorithms, which only exploit information contained in the coefficient matrix [452, 468]. If the prolongation operator $\mathcal{I}_h^H$ is taken to be the transpose of the restriction operator $\mathcal{I}_H^h$, then the coarse grid operator $\mathcal{A}_H$ is symmetric if the fine grid one is. Preserving symmetry may be important if the multigrid iteration is to be accelerated by a symmetric Krylov subspace method. In this case the pre- and post-smoothing steps must also be performed in such a way as to preserve symmetry. The restriction operator is usually of the form

$$\mathcal{I}_H^h = \begin{bmatrix} I_H^h & O \\ O & \hat{I}_H^h \end{bmatrix},$$

where I_H^h and $\hat{I}_H^h$ are restriction operators for the x and y unknowns, respectively. If we assume for simplicity that $B_1 = B_2 = B_h$ and $C_h = O$ and take $\mathcal{I}_h^H = (\mathcal{I}_H^h)^T$, the coarse grid matrix has the same form as the fine one:

$$\mathcal{A}_H = \begin{bmatrix} I_H^h & O \\ O & \hat{I}_H^h \end{bmatrix} \begin{bmatrix} A_h & B_h^T \\ B_h & O \end{bmatrix} \begin{bmatrix} I_h^H & O \\ O & \hat{I}_h^H \end{bmatrix} = \begin{bmatrix} A_H & B_H^T \\ B_H & O \end{bmatrix},$$

where $A_H = I_H^h A_h I_h^H$ and $B_H = \hat{I}_H^h B_h I_h^H$. Note that A_H is symmetric if A_h is.

The choice of the solver for the coarse grid problem is not as simple as it may appear at first. Only seldom can the grid be refined to the point that a direct method can be used to solve the coarse problem. This is because grid geometries can be highly irregular (thus allowing only a modest amount of coarsening), and also because a very coarse grid cannot be expected to resolve fine features of the solution (such as internal or boundary layers), thus making the coarse grid corrections almost useless [388]. It is therefore necessary to resort to iterative methods and inexact coarse grid solves. In this case the multigrid method should be used as a preconditioner for an (outer) flexible Krylov subspace iteration in order to guarantee convergence. Alternatively, a hybrid nonlinear multigrid-inexact Newton method like the one proposed in [388] can be used.

We now come to the choice of the smoother. To appreciate the importance of this component, it should be kept in mind that the smoothing steps is where most of the computational effort is usually spent, at least for geometric multigrid. Moreover, using the wrong smoother will destroy the efficiency of the entire multigrid algorithm. The definition of appropriate smoothing operations for saddle point problems is highly problem-dependent, and far from obvious. The smoothers used for scalar elliptic PDEs, such as diffusion or convection-diffusion-type problems, are typically Gauss-Seidel or damped Jacobi relaxation, possibly combined with an appropriate renumbering of the grid points. These smoothers are not appropriate for saddle point problems, and they are not even defined when $C = O$. One approach is to use Richardson or Kaczmarz-type iterations, which is equivalent to using Jacobi or Gauss-Seidel smoothing on the normal equations (or squared system for symmetric problems), see [250, 486]. A better approach is to use some of the stationary iterations (or preconditioners) for saddle point problems discussed in sections 8–10, but even this is not entirely straightforward.

The so-called *distributive relaxation schemes* (or *transforming iterations*) provide a unified framework for the description of smoothers for saddle point problems, see [77, 503, 504]. The idea is to transform the original problem $\mathcal{A}\,u = b$ by right preconditioning (often referred to as *postconditioning* in the multigrid literature) so that the transformed system

$$\mathcal{A}\,\mathcal{B}\,v = b, \quad u = \mathcal{B}\,v$$

is more easily solved. A splitting

$$\mathcal{A}\,\mathcal{B} = \mathcal{S} - \mathcal{T}$$

of the transformed matrix induces the following splitting of the original matrix $\mathcal{A}$:

$$\mathcal{A} = \mathcal{S}\,\mathcal{B}^{-1} - \mathcal{T}\,\mathcal{B}^{-1}\,.$$

This splitting defines a stationary iterative method for the original system:

$$\mathcal{S}\,\mathcal{B}^{-1}u_{k+1} = \mathcal{T}\,\mathcal{B}^{-1}u_k + b$$

or, in correction form,

$$u_{k+1} = u_k + \mathcal{B}\,\mathcal{S}^{-1}(b - \mathcal{A}\,u_k)\,. \tag{11.1}$$

As long as $\mathcal{B}$ and $\mathcal{S}$ are nonsingular, the iteration (11.1) is consistent with $\mathcal{A}\,u = b$, in the sense that if it converges, it converges to the solution $u_* = \mathcal{A}^{-1}b$. The scheme (11.1) is called a *distributive relaxation* in the multigrid literature. The reason is that the correction $\mathcal{S}^{-1}(b - \mathcal{A}\,u_k)$ corresponding to non-distributive ($\mathcal{B} = I$) iterations is distributed over the entries of u_{k+1}; see [77] and [499, page 295].

A number of different distributive iterations can be obtained by special choices of $\mathcal{B}$ and $\mathcal{S}$. One possibility is to pick $\mathcal{B}$ such that $\mathcal{A}\mathcal{B}$ is block lower triangular. For instance, the block factorization (3.2) suggests the choice

$$\mathcal{B} = \begin{bmatrix} I & -A^{-1}B^T \\ O & I \end{bmatrix} \quad \Rightarrow \quad \mathcal{A}\mathcal{B} = \begin{bmatrix} A & O \\ B & S \end{bmatrix}.$$

This leads to a decoupling (segregation) of the x and y variables, and various choices of $\mathcal{S}$ lead essentially to preconditioned versions of Uzawa's method. A slightly more general form of $\mathcal{B}$ is

$$\mathcal{B} = \begin{bmatrix} I & -A^{-1}B^T Q \\ O & Q \end{bmatrix},$$

which leads to the transformed matrix

$$\mathcal{A}\mathcal{B} = \begin{bmatrix} A & O \\ B & M \end{bmatrix}, \quad M = -BA^{-1}B^T Q\,.$$

A variety of methods result from the choice of Q and $\mathcal{S}$. For example, taking $Q = I$ and

$$\mathcal{S} = \begin{bmatrix} \hat{A} & O \\ B & \hat{S} \end{bmatrix},$$

where $\hat{A}$ and $\hat{S}$ are easily invertible approximations of A and of the Schur complement S, results in the SIMPLE scheme [382, 380] already described in the context of block preconditioning in section 10.1.2 above. In the original SIMPLE scheme $\hat{A} = D_A$ and $\hat{S} = -BD_A^{-1}B^T$, where D_A denotes the main diagonal of A. The matrix $\mathcal{B}$ used for the distribution step is further approximated by

$$\hat{\mathcal{B}} = \begin{bmatrix} I & -\hat{A}^{-1}B^T \\ O & I \end{bmatrix}.$$

Different choices of the approximations involved lead to variations on the original SIMPLE scheme [380, 381, 433]. Other options include the distributive Gauss-Seidel iteration and distributive ILU methods; see [77, 503, 504] and the nice description in [499, Chapter 7.6]. For the Stokes problem, good results have been reported in [71] using as a smoother an iteration of the form (10.18), i.e., a constraint-type preconditioner. This paper also discusses several variants of the SIMPLE scheme and shows that not all variants result in good smoothers. See also [513] for a generalization to the case$C \neq O$. Smoothers of the form (10.18) have also been used in connection with AMG methods for the Oseen problem in [488], and in [318] in the context of meshless discretizations. Recently, block triangular preconditioners of the kind discussed in section 10.1 (so-called pressure convection-diffusion preconditioners) have been used with good results as smoothers for multigrid methods applied to the Oseen problem in [455]. For a discussion of the barriers that remain to be overcome before *textbook multigrid efficiency* (i.e., solution of the governing equations in a number of operations not exceeding a small multiple of the nonzeros in the discretized equations) can be achieved in realistic computational fluid dynamics simulations, see [76]. See also [460] for recent progress in this direction.

All the smoothers discussed so far can be implemented in a segregated fashion, with each relaxation involving solution of decoupled linear systems for x and y. A strongly coupled smoothing procedure, not of the distributive type, has been introduced by Vanka in [478]. The main principle of Vanka smoothing as applied to fluid flow problems is to visit each cell in some order and to apply relaxation simultaneously to each of the variables associated with that cell. This is equivalent to a reordering of the unknowns so that all the velocities and the pressures associated with each

cell are numbered consecutively. Both multiplicative (Gauss-Seidel-type) and additive (Jacobi-type) variants have been developed. Vanka-type smoothing requires solving, at each step, a number of tiny saddle point problems, one for each cell. Various approximations and suitable damping (underrelaxation) are used to improve efficiency and to ensure the smoothing property. We refer to the original paper [478] as well as to [499, pages 300–301] for additional details. We note that this type of smoothing can also be interpreted as a form of domain decomposition, each subdomain reducing to a single grid cell.

Problems involving unstructured meshes on complicated geometries have motivated the development of Algebraic Multigrid methods. AMG algorithms for scalar problems have been around for almost 20 years and have been used for some time in the solution of subproblems from segregated aproaches; see, e.g., [240, 452, 489]. Coupled AMG methods for saddle point systems are barely 10 years old and are currently under development. We mention [410, 488, 494] for applications to incompressible flow problems, [2] for the development of AMG solvers in solid mechanics, and [318] for AMG methods for saddle point systems arising from meshfree discretizations. The reported results on difficult problems are promising. However, many open questions concerning coarsening strategies, the choice of appropriate smoothers, and parallel implementation issues remain to be addressed before coupled AMG solvers for saddle point problems can be considered fully mature.

Multilevel preconditioners for discontinuous Galerkin mixed finite element discretizations of radiation-diffusion problems on unstructured meshes have been developed in [491]. Here the main idea is to use a continuous finite element approximation of the same problem to precondition the discontinuous one. Although only one mesh needs to be generated, suitable restriction and prolongation operators are needed to transfer information between the two discretizations. The continuous approximation can be solved efficiently by standard algorithms like AMG or preconditioned conjugate gradients. This approach, which results in nearly optimal rates of convergence, is akin to an approximate Schur complement method. It is worth noting that in both [318] and [491] the nonsymmetric formulation (3.10) was found to be advantageous, even though the original problems were symmetric.

Finally we mention some recent work on wavelet-based, *multiscale* methods. Originally developed for positive definite elliptic operators, these techniques have recently been extended to symmetic indefinite saddle point problems; see [129, 272, 313]. These are discretization schemes that produce saddle point systems that are well-conditioned, and can be efficiently solved by either Uzawa-type schemes or by diagonally preconditioned Krylov methods, resulting in convergence rates independent of discretization parameters.

11.2. Domain decomposition methods. Just as in the case of multigrid, domain decomposition methods can be applied straightforwardly to the (elliptic, scalar) subproblems arising from segregated approaches; see, e.g., [179] for a recent example. Coupled algorithms have been recently developed by several authors, in particular for fluid flow and linear elasticity problems. Such methods are seldom (if ever) used as solvers: rather, they are used either as preconditioners for Krylov subspace methods, or as smoothers for multigrid, often motivated by parallel processing. We refer the reader to [95, 408] for general treatments of domain decomposition methods, and to [466, Chapter 9] for an excellent discussion of methods designed specifically for saddle point problems. Because an up-to-date overview is already available and also due to space limitations, we do not go into details and we largely limit ourselves to providing pointers to the literature.

Substructuring and additive Schwarz-type preconditioners for the symmetric saddle point problems arising from mixed finite element discretizations of second order elliptic PDEs have been proposed and analyzed in [417, 419]. Different boundary conditions (Dirichlet or Neumann) on the

interior interfaces are considered. The preconditioned systems are symmetrizable (so that MINRES can be used), and have condition numbers that are independent of mesh size.

An algebraic additive Schwarz domain decomposition preconditioner for solving saddle point problems arising from mixed finite element simulations of stochastic PDEs modelling flow in heterogeneous media has been decribed in [113]. In this paper the preconditioner is not applied to to the coupled saddle point problem, but rather to a reduced (SPD) system of the type (6.1) where Z is a suitably constructed solenoidal basis. The resulting solver exhibits good robustness with respect to problem parameters and almost optimal levels of parallel efficiency.

Several substructuring and domain decomposition preconditioners exist for linear elasticity and Stokes problems. Overlapping Schwarz preconditioners for saddle point systems containing a nonzero (2,2) block which depends on a penalty parameter (as in (10.9)) have been proposed in [303]. These preconditioners require the solution of local saddle point problems on overlapping subdomains, plus the solution of a coarse-level saddle point problem. Numerical experiments show that these preconditioners result in convergence rates that are independent of the mesh size, the number of subdomains, and the penalty parameter. The additive preconditioners are scalable also in the sense of parallel efficiency. In [304], the additive Schwarz preconditioner proposed in [303] is compared experimentally with a block diagonal and with a block triangular preconditioner (both with inexacts solves). The results indicate that the additive Schwarz preconditioner (used with GMRES) is superior to the other methods in terms of iteration counts and very often in terms of operation counts, with the block triangular preconditioner being occasionally slightly better in terms of operation count. A convergence analysis for the additive Schwarz preconditioner, however, is still lacking. The preconditioned matrices have complex eigenvalues and an indefinite symmetric part. On the other hand, they appear to be positive stable with eigenvalues contained in a rectangular region of the right-half plane bounded away from zero, which explains in part the good convergence properties of GMRES. In [385] the overlapping Schwarz preconditioner is applied to mixed spectral or finite element discretizations of time-dependent Stokes problems, again with excellent results.

A different approach, based on iterative substructuring, has been introduced and analyzed in [386] in the context of spectral element mixed approximations. A related balancing Neumann-Neumann approach for the Stokes problem is proposed in [387], and extended to heterogeneous problems from linear elasticity in [218]. This is a non-overlapping hybrid domain decomposition method in which the local problems are treated additively while the coarse problem is treated multiplicatively. Theoretical and numerical evidence show that these methods have excellent convergence and scalability properties.

Another domain decomposition method for linear elasticity problems has been introduced in [306]. This preconditioner uses inexact subdomain solves and Lagrange multipliers and is based on a reformulation of the popular FETI method as a saddle point problem with both primal and dual variables as unknowns. It is shown in [306] that the condition number of the preconditioned matrices is bounded independently of the number of subdomains and grows only polylogarithmically with the number of unknowns in each subdomain. See further [307], and [308] for recent developments. We also mention the overlapping additive and multiplicative two-level Schwarz methods proposed in [490] for plane elasticity problems, which exhibit convergence rates independent of mesh size.

In [3], p and h-p finite element discretizations of the Stokes problem are considered. Both segregated (block diagonal) and coupled (indefinite) preconditioners of the additive Schwarz type are studied. For each type of scheme, eigenvalue bounds for the preconditioned matrices are derived in terms of p (the polynomial degree), the fine and coarse mesh sizes, and the inf-sup constant for the method. These estimates and the results of actual computations on representative problems show that the rate of convergence for both methods does not deteriorate significantly as the mesh

is refined and the polynomial degree is increased.

We note that while there is a rich literature on domain decomposition methods for symmetric saddle point problems and in particular for mixed discretizations of the Stokes equations, there seems to be very few papers concerned with coupled domain decomposition methods for nonsymmetric saddle point systems and in particular for the Oseen equations. One such paper is [473], where domain decomposition methods are implemented and experimentally compared with (and found to be superior to) block triangular preconditioners for the Oseen problem. Theoretical understanding, however, is still largely lacking.

Substructuring preconditioners for saddle point problems arising from edge element discretizations of Maxwell's equation in 3D have been introduced and analyzed in [278]. These preconditioners are shown to result in nearly optimal convergence rates, with the condition number of the preconditioned matrices growing as the logarithm of the ratio between the subdomain diameter and the finite element mesh size.

Domain decomposition methods for saddle point systems arising from PDE-constrained optimal control problems have been studied in [369]. Numerical experiments indicate that the performance of these (overlapping and non-overlapping) preconditioners with respect to mesh size and number of subdomains is close to that of the corresponding domain decomposition preconditioners for scalar PDEs. Furthermore, the preconditioners proposed in [369] appear to be rather insensitive to control regularization parameters.

Finally, the application of additive Schwarz-type iterations as smoothers in coupled multigrid methods for saddle point problems has been studied in [432]. These are additive variants of the already mentioned (multiplicative) Vanka smoother [478]. The additive smoothers are shown to be related to an inexact version of Uzawa's method. An analysis of the smoothing property in the symmetric case is also given. Numerical experiments indicate that the rates of convergence for the multiplicative Vanka-type smoothers are significantly better than for the additive smoothers, as one would expect; however, a theoretical analysis of the convergence and smoothing properties in the multiplicative case is still missing.

12. Available software. In spite of vigorous algorithmic and theoretical developments, the production of high-quality, widely accessible software for solving linear systems in saddle point form has been somewhat lagging. This may be in part a reflection of the fact that many saddle point systems arising in the applications require a customized approach—for example, a particular Schur complement approximation. Another explanation is that when segregated approaches are used, standard sparse linear equations packages are often sufficient to solve the reduced linear systems. For example, many of the existing algorithms for saddle point problems from incompressible flow calculations require the availability of efficient solvers for diffusion or convection-diffusion-type problems, and can utilize a number of "standard" multigrid codes. In other situations, sparse direct solvers may be used to solve the subproblems arising from segregated approaches. Because many general-purpose linear solver packages exist and are readily accessible, there hasn't been great demand for software specifically designed to solve linear systems in saddle point form. An exception is represented by the field of optimization, where the interior point revolution has sparked a strong need for reliable and efficient solvers for augmented systems.

As usual, the software situation tends to be better for direct solvers than for iterative ones. Professional implementations of sparse direct solvers for symmetric indefinite systems include the MA27/MA47/MA57 suite of codes from the Harwell Subroutine Library (HSL) and the PARDISO package from the University of Basel [431]. Of the HSL codes, the one that is best suited for saddle point matrices is perhaps MA47, although MA57 appears to be the most efficient overall of

the symmetric solvers among the HSL codes [229]. For further information on the HSL symmetric indefinite solvers, including licensing information, the reader is referred to the web page

http://www.cse.clrc.ac.uk/nag/hsl/contents.shtml

For information about the PARDISO package, see

http://www.computational.unibas.ch/cs/scicomp/software/pardiso/

Although not specifically designed with saddle point systems in mind, the out-of-core sparse symmetric indefinite code developed as part of the TAUCS package at Tel-Aviv university is also of interest; see

http://www.tau.ac.il/∼stoledo/taucs/

and the accompanying paper [359].

Software tools for incorporating block preconditioners geared towards incompressible flow problems into Krylov subspace methods are now available in the Meros package, which is part of the Trilinos project out of Sandia National Laboratories [264]; see

http://software.sandia.gov/trilinos/

Many of the solvers within Trilinos may also be used as part of segregated approches; for instance, multilevel solvers are part of the ML package which is part of Trilinos.

Algebraic multigrid codes for incompressible flow problems, including Markus Wabro's AMuSE (Algebraic Multigrid for Stokes-type Equations) package, can be accessed from the Institute of Computational Mathematics at the Johannes Kepler University (Linz, Austria) web site

http://www.numa.uni-linz.ac.at/Research/Projects/P14953.html

A number of other freely available codes implementing a host of multigrid and domain decomposition methods can be downloaded from the MGNET web site; see

http://www.mgnet.org/mgnet-codes.html

We conclude this brief—and by no means complete—overview of available software with the IFISS ("Incompressible

http://www.ma.umist.ac.uk/djs/software.html

See also the book [170]. This is a finite element package that can be used to generate a variety of incompressible flow problems on both structured and unstructured 2D meshes. The code, written in Matlab [354], allows the user to experiment with several preconditioner and solver options.

13. Concluding remarks. In this paper we have surveyed a large number of solution methods for solving linear systems in saddle point form, with a focus on iterative methods suitable for large and sparse problems. We have discussed classical algorithms based on Schur complement reduction, null space methods, triangular factorization and stationary iterations like the Arrow-Hurwicz and Uzawa schemes and their variants. We have further discussed the use of Krylov subspace methods and described a number of preconditioning techniques including block diagonal, block triangular, constrained and incomplete factorization preconditioners. We have reviewed a promising new approach based on the Hermitian and skew-Hermitian splitting which appears to be especially well-suited for the Oseen problem in rotation form. We saw that many preconditioning techniques rely

(either explicitly or implicitly) on the availability of good and inexpensive approximations of a Schur complement matrix; similar comments apply to other solvers, such as multigrid or domain decomposition methods. We have seen that very effective solvers exist for some classes of problems, including mixed finite element formulations of elliptic PDEs and for the Stokes problems. Furthermore, great strides have been made in recent years towards the development of effective preconditioners for strongly nonsymmetric problems, such as the Oseen equations with low viscosity.

In spite of this, many challenges remain. Effective preconditioners are yet to be developed for large classes of linear systems arising from interior point methods in constrained optimization. Some degree of success has been achieved in this area with various types of approximate Schur complement and constraint preconditioners, but there is plenty of room for improvements. For many saddle point systems arising from optimal control problems, the (1,1) block A is often symmetric indefinite. Although these systems may sometimes be transformed to reduced saddle point systems with a definite (1,1) block, this reduction is not always easy to accomplish and it may be necessary to deal with the unreduced system. Hence, there is a need for effective solvers that are able to handle saddle point matrices with an indefinite (1,1) block. While some of the techniques described in this paper may be adapted to this situation, there is a need for new approaches.

Space and time limitations did not allow us to give very detailed treatments of many interesting techniques. In particular, important subjects like direct solvers and multilevel methods have been touched upon only very briefly. Sparse direct solvers have been around for a long time and are widely used in some areas (such as optimization), but there are still some difficulties to overcome before these methods can be considered fully mature for solving saddle point systems: see for instance [229], where it is mentioned that the symmetric HSL sparse direct solvers could not solve four out of sixty-one of the test problems considered there. The same holds true, of course, (even more so!) of iterative solvers. The ever increasing complexity and size of the linear systems to be solved is already making the use of iterative methods absolutely mandatory for many applications, thus requiring continually improving solvers and preconditioners, with scalability more than ever a central concern. In line with current trends, it is quite likely that the future will see many new and improved multilevel algorithms, both geometric and algebraic, for solving large-scale saddle point problems from PDEs and PDE-constrained optimization.

Acknowledgments. During the preparation of this paper we have benefited from the help and advice of numerous colleagues and friends. We are especially thankful to Mario Arioli, Howard Elman, Chen Greif, Eldad Haber, Amnon Meir, Wil Schilders, Valeria Simoncini, Daniel Szyld and Andy Wathen for informative discussions and pointers to the literature.

REFERENCES

[1] R. Aboulaich and M. Fortin, *Iterative methods for the solution of Stokes equations*, Comput. Methods Appl. Mech. Engrg., 75 (1989), pp. 317–324.
[2] M. F. Adams, *Algebraic multigrid methods for constrained linear systems with applications to contact problems in solid mechanics*, Numer. Linear Algebra Appl., 3 (2004), pp. 275–300.
[3] M. Ainsworth and S. Sherwin, *Domain decomposition preconditioners for p and hp finite element approximation of Stokes equations*, Comput. Methods Appl. Mech. Engrg., 175 (1999), pp. 243–266.
[4] P. Alotto and I. Perugia, *Mixed finite element methods and tree-cotree implicit condensation*, Calcolo, 36 (1999), pp. 233–248.
[5] P. Amestoy, I. S. Duff, and C. Puglisi, *Multifrontal QR factorization in a multiprocessor environment*, Numer. Linear Algebra Appl., 3 (1996), pp. 275–300.
[6] R. Amit, C. A. Hall, and T. A. Porsching, *An application of network theory to the solution of implicit Navier-Stokes difference equations*, J. Comput. Phys., 40 (1981), pp. 183–201.
[7] G. Ammar, C. Mehl, and V. Mehrmann, *Schur-like forms for matrix Lie groups, Lie algebras and Jordan algebras*, Linear Algebra Appl., 287 (1999), pp. 11–39.

[8] P. Arbenz and R. Geus, *Multilevel preconditioned iterative eigensolvers for Maxwell eigenvalue problems*, Appl. Numer. Math., (2004). To appear.
[9] M. Arioli, *The use of QR factorization in sparse quadratic programming*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 829–839.
[10] M. Arioli and L. Baldini, *A backward error analysis of a null space algorithm in sparse quadratic programming*, SIAM J. Matrix Anal. Appl., 23 (2001), pp. 425–442.
[11] M. Arioli, I. S. Duff, and P. P. M. De Rijk, *On the augmented system approach to sparse least-squares problems*, Numer. Math., 55 (1989), pp. 667–684.
[12] M. Arioli and G. Manzini, *A null space algorithm for mixed finite-element approximations of Darcy's equation*, Comm. Numer. Meth. Engng., 18 (2002), pp. 645–657.
[13] ———, *Null space algorithm and spanning trees in solving Darcy's equation*, BIT, 43 (2003), pp. 839–848.
[14] M. Arioli, J. Maryška, M. Rozložník, and Tůma, *Dual variable methods for mixed-hybrid finite element approximation of the potential fluid flow problem in porous media*, Technical Report RAL-TR-2001-023, Rutherford Appleton Laboratory, RAL-CLRC, Oxfordshire, UK, 2001.
[15] D. N. Arnold, R. S. Falk, and R. Winther, *Preconditioning in H(div) and applications*, Math. Comp., 66 (1997), pp. 957–984.
[16] W. E. Arnoldi, *The principle of minimized iteration in the solution of the matrix eigenvalue problem*, Quart. Appl. Math., 9 (1951), pp. 17–29.
[17] K. J. Arrow and L. Hurwicz, *Gradient method for concave programming. I, Local results*, in Studies in Linear and Nonlinear Programming, K. J. Arrow, L. Hurwicz, and H. Uzawa, eds., Stanford University Press, Stanford, CA, 1958, pp. 117–126.
[18] K. J. Arrow, L. Hurwicz, and H. Uzawa, *Studies in Linear and Nonlinear Programming*, Stanford University Press, Stanford, CA, 1958.
[19] K. J. Arrow and R. M. Solow, *Gradient methods for constrained maxima, with weakened assumptions*, in Studies in Linear and Nonlinear Programming, K. J. Arrow, L. Hurwicz, and H. Uzawa, eds., Stanford University Press, Stanford, CA, 1958, pp. 166–176.
[20] U. M. Ascher and E. Haber, *A multigrid method for distributed parameter estimation problems*, Electron. Trans. Numer. Anal., 15 (2003), pp. 1–17.
[21] C. Ashcraft, R. Grimes, and J. G. Lewis, *Accurate symmetric indefinite linear equation solvers*, SIAM J. Matrix Anal. Appl., 20 (1998), pp. 513–561.
[22] G. P. Astrakhantsev, *Analysis of algorithms of the Arrow–Hurwicz type*, Comput. Math. Math. Phys., 41 (2001), pp. 15–26.
[23] G. M. Awanou and M. J. Lai, *On convergence rate of the augmented Lagrangian algorithm for nonsymmetric saddle point problems*, Appl. Numer. Math., (2005). To appear.
[24] O. Axelsson, *Preconditioning of indefinite problems by regularization*, SIAM J. Numer. Anal., 16 (1979), pp. 58–69.
[25] O. Axelsson, *Iterative Solution Methods*, Cambridge University Press, Cambridge, UK, 1994.
[26] O. Axelsson and M. Neytcheva, *Preconditioning methods for linear systems arising in constrained optimization problems*, Numer. Linear Algebra Appl., 10 (2003), pp. 3–31.
[27] O. Axelsson and P. S. Vassilevski, *A black-box generalized conjugate gradient solver with inner iterations and variable-step preconditioning*, SIAM J. Matrix Anal. Appl., 12 (1991), pp. 625–644.
[28] A. Baggag and A. Sameh, *A nested iterative scheme for indefinite linear systems in particulate flows*, Comput. Methods Appl. Mech. Engrg., 193 (2004), pp. 1923–1957.
[29] Z.-Z. Bai, G. H. Golub, and M. K. Ng, *Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*, SIAM J. Matrix Anal. Appl., 24 (2003), pp. 603–626.
[30] R. E. Bank, B. D. Welfert, and H. Yserentant, *A class of iterative methods for solving saddle point problems*, Numer. Math., 56 (1990), pp. 645–666.
[31] J. L. Barlow, *Error analysis and implementation aspects of deferred correction for equality-constrained least squares problems*, SIAM J. Numer. Anal., 25 (1988), pp. 1340–1358.
[32] J. L. Barlow and S. L. Handy, *The direct solution of weighted and equality constrained least squares problems*, SIAM J. Sci. Statist. Comput., 9 (1988), pp. 704–716.
[33] J. L. Barlow, N. K. Nichols, and R. J. Plemmons, *Iterative methods for equality-constrained least squares problems*, SIAM J. Sci. Statist. Comput., 9 (1988), pp. 892–906.
[34] T. Barth and T. Manteuffel, *Variable metric conjugate gradient methods*, in Advances in Numerical Methods for Large Sparse Sets of Linear Equations, Number 10, Matrix Analysis and Parallel Computing, PCG 94, M. Natori and T. Nodera, eds., 1994, pp. 165–188.
[35] J. R. Batt and S. Gellin, *Rapid reanalysis by the force method*, Comput. Methods Appl. Mech. Engrg., 53 (1985), pp. 105–117.
[36] A. Battermann and M. Heinkenschloss, *Preconditioners for Karush-Kuhn-Tucker systems arising in the optimal control of distributed systems*, in Optimal Control of Partial Differential Equations, W. Desch, F. Kappel, and K. Kunisch, eds., Birkhäuser Verlag, 1998, pp. 15–32.
[37] A. Battermann and E. W. Sachs, *Block preconditioners for KKT systems arising in PDE-governed optimal control problems*, in Fast Solution of Discretized Optimization Problems (Berlin, 2000), K.-H. Hoffmann,

R. H. W. Hoppe, and W. Schulz, eds., Birkhäuser Verlag, 2001, pp. 1–18.
[38] J. Bear, *Dynamics of Fluids in Porous Media*, American Elsevier, New York, NY, 1972.
[39] B. Beckermann and A. B. J. Kuijlaars, *Superlinear convergence of conjugate gradients*, SIAM J. Numer. Anal., 39 (2001), pp. 300–329.
[40] ———, *Superlinear CG convergence for special right-hand sides*, Electron. Trans. Numer. Anal., 14 (2002), pp. 1–19.
[41] S. J. Benbow, *Solving generalized least-squares problems with LSQR*, SIAM J. Matrix Anal. Appl., 21 (1999), pp. 166–177.
[42] M. Benzi, *Solution of equality-constrained quadratic programming problems by a projection iterative method*, Rend. Mat. Appl. (7), 13 (1993), pp. 275–296.
[43] ———, *Preconditioning techniques for large linear systems: a survey*, J. Comput. Phys, 182 (2002), pp. 418–477.
[44] ———, *HSS preconditioning for the Oseen problem*, in Proceedings of the European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2004), Jyväskylä, Finland, P. Neittaanmäki, T. Rossi, S. Korotov, E. Oñate, J. Périaux, and D. Knörzer, eds., CD-ROM, 2004.
[45] M. Benzi, M. J. Gander, and G. H. Golub, *Optimization of the Hermitian and skew-Hermitian splitting iteration for saddle-point problems*, BIT, 43 (2003), pp. 881–900.
[46] M. Benzi and G. H. Golub, *A preconditioner for generalized saddle point problems*, SIAM J. Matrix Anal. Appl., 26 (2004), pp. 20–41.
[47] M. Benzi, J. C. Haws, and M. Tůma, *Preconditioning highly indefinite and nonsymmetric matrices*, SIAM J. Sci. Comput., 22 (2000), pp. 1333–1353.
[48] M. Benzi and J. Liu, *An efficient solver for the incompressible Navier–Stokes equations in rotation form*, J. Comput. Phys., (2005). To be submitted.
[49] M. Benzi and M. K. Ng, *Preconditioned iterative methods for weighted Toeplitz least squares*, SIAM J. Matrix Anal. Appl., (2004). Submitted.
[50] L. Bergamaschi, J. Gondzio, and G. Zilli, *Preconditioning indefinite systems in interior point methods for optimization*, Comput. Optim. Appl., 28 (2004), pp. 149–171.
[51] A. R. Bergen, *Power Systems Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1986.
[52] M. W. Berry, M. T. Heath, I. Kaneko, M. Lawo, R. J. Plemmons, and R. C. Ward, *An algorithm for computing a sparse basis of the null space*, Numer. Math., 47 (1985), pp. 483–504.
[53] M. W. Berry and R. J. Plemmons, *Algorithms and experiments for structural mechanics on high-performance architectures*, Comput. Methods Appl. Mech. Engrg., 64 (1987), pp. 487–507.
[54] F. Bertrand and P. A. Tanguy, *Krylov-based Uzawa algorithms for the solution of the Stokes equations using discontinuous-pressure tetrahedral finite elements*, J. Comput. Phys., 181 (2002), pp. 617–638.
[55] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.
[56] J. T. Betts, *Practical Methods for Optimal Control Using Nonlinear Programming*, Advances in Design and Control, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2001.
[57] G. Biros and O. Ghattas, *Parallel preconditioners for KKT systems arising in optimal control of viscous incompressible flows*, in Proceedings of Parallel CFD '99, Williamsburg, VA, May 23-26, 1999, D. Keyes, A. Ecer, J. Periaux, and N. Satofuka, eds., North-Holland, 2000, pp. 1–8.
[58] L. Bitar and C. Vincent, *Eigenvalue upper bounds for the discretized Stokes operator*, Commun. Numer. Meth. Engng., 16 (2000), pp. 449–457.
[59] Å. Björck, *Numerical Methods for Least Squares Problems*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1996.
[60] Å. Björck and C. C. Paige, *Solution of augmented linear systems using orthogonal factorizations*, BIT, 34 (1994), pp. 1–24.
[61] E. Y. Bobrovnikova and S. A. Vavasis, *A norm bound for projections with complex weights*, Linear Algebra Appl., 307 (2000), pp. 69–75.
[62] ———, *Accurate solution of weighted least squares by iterative methods*, SIAM J. Matrix Anal. Appl., 22 (2001), pp. 1153–1174.
[63] P. Bochev and R. B. Lehoucq, *On finite element solution of the pure Neumann problem*, SIAM Rev., (To appear).
[64] A. Bojanczyk, N. J. Higham, and H. Patel, *Solving the indefinite least squares problem by hyperbolic QR factorization*, SIAM J. Matrix Anal. Appl., 24 (2003), pp. 914–931.
[65] C. W. Bomhof and H. A. van der Vorst, *A parallel linear system solver for circuit-simulation problems*, Numer. Linear Algebra Appl., 7 (2000), pp. 649–665.
[66] I. Bongartz, A. R. Conn, N. I. M. Gould, and P. L. Toint, *CUTE: Constrained and unconstrained testing environment*, ACM Trans. Math. Software, 21 (1995), pp. 123–160.
[67] A. Bossavit, *'Mixed' systems of algebraic equations in computational electromagnetics*, COMPEL, 17 (1998), pp. 59–63.
[68] M. A. Botchev and G. H. Golub, *A class of nonsymmetric preconditioners for saddle point problems*, Technical Report SCCM-04-14, Scientific Computing and Computational Mathematics, Stanford University,

Stanford, California, 2004.
[69] D. Braess, *Finite Elements. Theory, Fast Solvers, and Applications in Solid Mechanics*, Cambridge University Press, Cambridge, UK, second ed., 2001.
[70] D. Braess, W. Dahmen, and C. Wieners, *A multigrid algorithm for the mortar finite element method*, SIAM J. Numer. Anal., 37 (1999), pp. 48–69.
[71] D. Braess and R. Sarazin, *An efficient smoother for the Stokes problem*, Appl. Numer. Math., 23 (1997), pp. 3–19.
[72] J. H. Bramble and J. E. Pasciak, *A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems*, Math. Comp., 50 (1988), pp. 1–17.
[73] ———, *Iterative techniques for time-dependent Stokes problems*, Comput. Math. Appl., 33 (1997), pp. 13–30.
[74] J. H. Bramble, J. E. Pasciak, and A. T. Vassilev, *Analysis of the inexact Uzawa algorithm for saddle point problems*, SIAM J. Numer. Anal., 34 (1997), pp. 1072–1092.
[75] ———, *Uzawa type algorithms for nonsymmetric saddle point problems*, Math. Comp., 69 (2000), pp. 667–689.
[76] A. Brandt, *Barriers to achieving textbook multigrid efficiency (TME) in CFD*, Gauss Center Technical Report WI-GC 10, Weizmann Institute of Science, Rehovot, Israel, 1998.
[77] A. Brandt and D. Dinar, *Multigrid solutions to elliptic flow problems*, in Numerical Methods for Partial Differential Equations, S. Parter, ed., Academic Press, New York, 1979, pp. 53–147.
[78] F. Brezzi, *On the existence, uniqueness, and approximation of saddle-point problems arising from Lagrangian multipliers*, RAIRO Anal. Numér., 8 (1974), pp. 129–151.
[79] F. Brezzi and M. Fortin, *Mixed and Hybrid Finite Element Methods*, vol. 15 of Springer Series in Computational Mathematics, Springer-Verlag, New York, 1991.
[80] D. C. Brown, *Alternating-direction iterative schemes for mixed finite element methods for second order elliptic problems*, PhD thesis, Department of Mathematics, University of Chicago, Chicago, IL, 1982.
[81] P. N. Brown, *A theoretical comparison of the Arnoldi and GMRES algorithms*, SIAM J. Sci. Statist. Comput., 12 (1991), pp. 58–78.
[82] J. R. Bunch, *Equilibration of symmetric matrices in the Max-norm*, J. of the ACM, 18 (1971), pp. 566–572.
[83] ———, *Partial pivoting strategies for symmetric matrices*, SIAM J. Numer. Anal., 11 (1974), pp. 521–528.
[84] J. R. Bunch and L. Kaufman, *Some stable methods for calculating inertia and solving symmetric linear systems*, Math. Comp., 31 (1977), pp. 163–179.
[85] J. R. Bunch and B. N. Parlett, *Direct methods for solving symmetric indefinite systems of linear equations*, SIAM J. Numer. Anal., 8 (1971), pp. 639–655.
[86] M. Burger and W. Mühlhuber, *Iterative regularization of parameter identification problems by sequential quadratic programming methods*, Inverse Problems, 18 (2002), pp. 943–969.
[87] J. Burkardt, C. Hall, and T. Porsching, *The dual variable method for the solution of compressible fluid flow problems*, SIAM J. Alg. Disc. Meth., 7 (1986), pp. 476–483.
[88] Y. V. Bychenkov, *Optimization of one class of nonsymmetrizable algorithms for saddle point problems*, Russian J. Numer. Anal. Math. Modelling, 17 (2002), pp. 521–546.
[89] J. Cahouet and J. P. Chabard, *Some fast 3-D finite element solvers for the generalized Stokes problem*, Int. J. Numer. Meth. Fluids, 8 (1988), pp. 869–895.
[90] S. L. Campbell and C. D. Meyer Jr., *Generalized Inverses of Linear Transformations*, Pitman Publishing Ltd., London, 1979. Reprinted by Dover Publications, Inc., New York, 1991.
[91] Z.-H. Cao, *A note on constraint preconditioning for nonsymmetric indefinite matrices*, SIAM J. Matrix Anal. Appl., 24 (2002), pp. 121–125.
[92] ———, *Fast iterative solution of stabilized Navier–Stokes systems*, Appl. Math. Comp., 157 (2004), pp. 219–241.
[93] ———, *Fast Uzawa algorithms for solving nonsymmetric stabilized saddle point problems*, Numer. Linear Algebra Appl., 11 (2004), pp. 1–24.
[94] A. Cassell, J. C. Henderson, and A. Kaveh, *Cycle bases for the flexibility analysis of structures*, Int. J. Numer. Methods Engng., 8 (1974), pp. 521–528.
[95] T. Chan and T. P. Mathew, *Domain decomposition algorithms*, Acta Numerica 1994, pp. 61–143.
[96] S. Chandrasekaran, M. Gu, and A. H. Sayed, *A stable and efficient algorithm for the indefinite linear least-squares problem*, SIAM J. Matrix Anal. Appl., 20 (1998), pp. 476–483.
[97] W. Chang, F. Giraldo, and J. B. Perot, *Analysis of an exact fractional step method*, J. Comput. Phys., 180 (2002), pp. 183–199.
[98] X. Chen, *On preconditioned Uzawa methods and SOR methods for saddle-point problems*, J. Comput. Appl. Math., 100 (1998), pp. 207–224.
[99] X. Chen and K. Hashimoto, *Numerical validation of solutions of saddle point matrix equations*, Numer. Linear Algebra Appl., 10 (2003), pp. 661–672.
[100] Z. M. Chen and J. Zou, *An augmented Lagrangian method for identifying discontinuous parameters in elliptic systems*, SIAM J. Contr. Optim., 37 (1999), pp. 892–910.
[101] X. L. Cheng, *On the nonlinear inexact Uzawa algorithm for saddle-point problems*, SIAM J. Numer. Anal., 37 (2000), pp. 1930–1934.
[102] X. L. Cheng and J. Zou, *An inexact Uzawa-type iterative method for solving saddle point problems*, Int. J. Computer Math., 80 (2003), pp. 55–64.

[103] E. V. Chizhonkov, *On solving an algebraic system of Stokes type under block diagonal preconditioning*, Comput. Math. Math. Phys., 41 (2001), pp. 514–521.
[104] ———, *Improving the convergence of the Lanczos method in solving algebraic saddle point problems*, Comput. Math. Math. Phys., 42 (2002), pp. 483–491.
[105] A. J. Chorin, *Numerical solution of the Navier-Stokes equations*, Math. Comp., 22 (1968), pp. 745–762.
[106] E. Chow, T. Manteuffel, C. Tong, and B. K. Wallin, *Algebraic elimination of slide surface constraints in implicit structural analysis*, Int. J. Numer. Methods Engng., 57 (2003), pp. 1129–1144.
[107] E. Chow and Y. Saad, *Approximate inverse techniques for block-partitioned matrices*, SIAM J. Sci. Comput., 18 (1997), pp. 1657–1675.
[108] ———, *Experimental study of ILU preconditioners for indefinite matrices*, J. Comput. Appl. Math., 86 (1997), pp. 387–414.
[109] L. O. Chua, C. A. Desoer, and E. S. Kuh, *Linear and Nonlinear Circuits*, McGraw-Hill, New York, 1987.
[110] P. Ciarlet, *Mathematical Elasticity. Volume I: Three Dimensional Elasticity*, vol. 20 of Studies in Mathematics and its Applications, North-Holland Publishing Co., Amsterdam, 1988.
[111] P. Ciarlet Jr., J. Huang, and J. Zou, *Some observations on generalized saddle-point problems*, SIAM J. Matrix Anal. Appl., 25 (2003), pp. 224–236.
[112] K. A. Cliffe, T. J. Garratt, and A. Spence, *Eigenvalues of block matrices arising from problems in fluid mechanics*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 1310–1318.
[113] K. A. Cliffe, I. G. Graham, R. Scheichl, and L. Stals, *Parallel computation of flow in heterogeneous media modelled by mixed finite elements*, J. Comput. Phys., 164 (2000), pp. 258–282.
[114] T. F. Coleman, *Large Sparse Numerical Optimization*, Springer-Verlag, New York, 1984.
[115] T. F. Coleman and A. Pothen, *The null space problem I: complexity*, SIAM J. Alg. Disc. Meth., 7 (1986), pp. 527–537.
[116] ———, *The null space problem II: algorithms*, SIAM J. Alg. Disc. Meth., 8 (1987), pp. 544–561.
[117] T. F. Coleman and A. Verma, *A preconditioned conjugate gradient approach to linear equality constrained minimization*, Comput. Optim. Appl., 20 (2001), pp. 61–72.
[118] P. Concus and G. H. Golub, *A generalized conjugate gradient method for nonsymmetric systems of linear equations*, in Computing Methods in Applied Sciences and Engineering, R. Glowinski and J. L. Lions, eds., Springer-Verlag, 1976, pp. 56–65.
[119] J. B. Conway, *A Course in Functional Analysis*, vol. 96 of Graduate Texts in Mathematics, Springer-Verlag, New York, second ed., 1990.
[120] R. Courant, *Variational methods for the solution of problems of equilibrium and vibration*, Bull. Amer. Math. Soc., 69 (1943), pp. 1–23.
[121] A. J. Cox and N. J. Higham, *Accuracy and stability of the null space method for solving the equality constrained least squares problem*, BIT, 39 (1999), pp. 34–50.
[122] ———, *Row-wise backward stable elimination methods for the equality constrained least squares problem*, SIAM J. Matrix Anal. Appl., 21 (1999), pp. 313–326.
[123] E. J. Craig, *The N-step iteration procedures*, J. Math. Phys., 34 (1955), pp. 64–73.
[124] M.-R. Cui, *A sufficient condition for the convergence of the inexact Uzawa algorithm for saddle point problems*, J. Comput. Appl. Math., 139 (2002), pp. 189–196.
[125] ———, *Analysis of iterative algorithms of Uzawa type for saddle point problems*, Appl. Numer. Math., 50 (2004), pp. 133–146.
[126] J. Cullum and A. Greenbaum, *Relations between Galerkin and norm-minimizing iterative methods for solving linear systems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 223–247.
[127] J. Czyzyk, R. Fourer, and S. Mehrotra, *Using a massively parallel processor to solve large sparse linear programs by an interior-point method*, SIAM J. Sci. Comput., 19 (1998), pp. 553–565.
[128] O. Dahl and S. Ø. Wille, *An ILU preconditioner with coupled node fill-in for iterative solution of the mixed finite element formulation of the 2D and 3D Navier-Stokes equations*, Int. J. Numer. Meth. Fluids, 15 (1992), pp. 525–544.
[129] S. Dahlke, R. Hochmuth, and K. Urban, *Adaptive wavelet methods for saddle point problems*, Math. Model. Numer. Anal., 34 (2000), pp. 1003–1022.
[130] C. de Boor and J. R. Rice, *Extremal polynomials with application to Richardson iteration for indefinite linear systems*, SIAM J. Sci. Statist. Comput., 3 (1982), pp. 47–57.
[131] E. de Sturler and J. Liesen, *Block-diagonal and constraint preconditioners for nonsymmetric indefinite linear systems. Part I: Theory*, SIAM J. Sci. Comput., (2005). To appear.
[132] S. Demko, W. F. Moss, and P. W. Smith, *Decay rates for inverses of band matrices*, Math. Comp., 44 (1984), pp. 491–499.
[133] G. B. Deng, J. Piquet, P. Queutey, and M. Visonneau, *Navier-Stokes equations for incompressible flows: finite-difference and finite volume methods*, in Handbook of Computational Fluid Mechanics, R. Peyret, ed., Academic Press, London, UK, 1996, pp. 25–97.
[134] P. Destuynder and T. Nevers, *Some numerical aspects of mixed finite elements for bending plates*, Comput. Methods Appl. Mech. Engrg., 78 (1990), pp. 73–87.
[135] M. Discacciati, E. Miglio, and A. Quarteroni, *Mathematical and numerical models for coupling surface*

and groundwater flows, Appl. Numer. Math., 43 (2002), pp. 57–74.

[136] M. Discacciati and A. Quarteroni, *Convergence analysis of a subdomain iterative method for the finite element approximation of the coupling of Stokes and Darcy equations*, Comput. Visual. Sci., 6 (2004), pp. 93–103.

[137] C. Dohrmann and R. B. Lehoucq, *A primal based penalty preconditioner for elliptic saddle point systems*, Research Report SAND-2004-59644, Sandia National Laboratories, Albuquerque, New Mexico, 2004.

[138] H. S. Dollar and A. J. Wathen, *Incomplete factorization constraint preconditioners for saddle-point matrices*, Research Report NA-04/01, Numerical Analysis Group, Oxford University, 2004.

[139] J. J. Dongarra, I. S. Duff, D. C. Sorensen, and H. A. van der Vorst, *Numerical Linear Algebra for High-Performance Computers*, vol. 11 of Software, Environments and Tools, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1998.

[140] J. Douglas Jr., R. Durán, and P. Pietra, *Alternating-direction iteration for mixed finite element methods*, in Computing Methods in Applied Sciences and Engineering VII, Versailles, France, 1985), Springer-Verlag, New York, 1986, pp. 181–196.

[141] ———, *Formulation of alternating-direction iteration for mixed methods in three-space*, in Numerical Approximation of Partial Differential Equations, vol. 133 of Studies inMathematics, North-Holland, Amsterdam, 1987, pp. 21–30.

[142] T. Dreyer, B. Maar, and V. Schulz, *Multigrid optimization in applications*, J. Comput. Appl. Math., 120 (2000), pp. 67–84.

[143] F. Duchin and D. B. Szyld, *Application of sparse matrix techniques to inter-regional input-output analysis*, Economics of Planning, 15 (1979), pp. 142–167.

[144] I. S. Duff, *The solution of augmented systems*, in Numerical Analysis 1993 (Proc. 15th Dundee Conf., Dundee University, June-July 1993), Pitman Research Notes in Mathematics Series, Vol. 303, Longman Scientific & Technical, Harlow, England, 1994, pp. 40–55.

[145] I. S. Duff, A. M. Erisman, and R. J. K., *Direct Methods for Sparse Matrices*, Monographs on Numerical Analysis, Oxford University Press, Oxford, UK, 1986.

[146] I. S. Duff, N. I. M. Gould, J. K. Reid, J. A. Scott, and K. Turner, *The factorization of sparse symmetric indefinite matrices*, IMA J. Numer. Anal., 11 (1991), pp. 181–204.

[147] I. S. Duff and J. Koster, *The design and use of algorithms for permuting large entries to the diagonal of sparse matrices*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 889–901.

[148] ———, *An algorithm for permuting large entries to the diagonal of a sparse matrix*, SIAM J. Matrix Anal. Appl., 22 (2001), pp. 973–996.

[149] I. S. Duff and S. Pralet, *Strategies for scaling and pivoting for sparse symmetric indefinite problems*, Research Report TR/PA/04/59, CERFACS, Toulouse, France, 2004.

[150] I. S. Duff and J. K. Reid, *The multifrontal solution of indefinite sparse symmetric linear systems*, ACM Trans. Math. Software, 9 (1983), pp. 302–325.

[151] ———, *MA47, a Fortran code for direct solution of indefinite sparse symmetric linear systems*, Research Report RAL-95-001, Rutherford Appleton Laboratory, 1995.

[152] ———, *Exploiting zeros on the diagonal in the direct solution of indefinite sparse symmetric linear systems*, ACM Trans. Math. Software, 22 (1996), pp. 227–257.

[153] C. Durazzi and V. Ruggiero, *Indefinitely preconditioned conjugate gradient method for large sparse equality and inequality constrained quadratic problems*, Numer. Linear Algebra Appl., 10 (2003), pp. 673–688.

[154] ———, *Numerical solution of special linear and quadratic programs via a parallel interior-point method*, Parallel Comput., 29 (2003), pp. 485–503.

[155] E. D'Yakonov, *On iterative methods with saddle operators*, Soviet Math. Dokl., 35 (1987), pp. 166–170.

[156] N. Dyn and W. E. Ferguson, Jr., *The numerical solution of equality constrained quadratic programming problems*, Math. Comp., 41 (1983), pp. 165–170.

[157] M. Eiermann and O. G. Ernst, *Geometric aspects of the theory of Krylov subspace methods*, Acta Numerica 2001, pp. 251–312.

[158] M. Eiermann and W. Niethammer, *On the construction of semi-iterative methods*, SIAM J. Numer. Anal., 20 (1983), pp. 1153–1160.

[159] M. Eiermann, W. Niethammer, and R. S. Varga, *A study of semi-iterative methods for nonsymmetric systems of linear equations*, Numer. Math., 47 (1985), pp. 505–533.

[160] H. C. Elman, *Multigrid and Krylov subspace methods for the discrete Stokes equations*, Int. J. Numer. Meth. Fluids, 22 (1996), pp. 755–770.

[161] ———, *Preconditioning for the steady-state Navier-Stokes equations with low viscosity*, SIAM J. Sci. Comput., 20 (1999), pp. 1299–1316.

[162] ———, *Preconditioners for saddle point problems arising in computational fluid dynamics*, Appl. Numer. Math., 43 (2002), pp. 75–89.

[163] H. C. Elman and G. H. Golub, *Inexact and preconditioned Uzawa algorithms for saddle point problems*, SIAM J. Numer. Anal., 31 (1994), pp. 1645–1661.

[164] H. C. Elman, V. E. Howle, J. Shadid, R. Shuttleworth, and R. Tuminaro, *Block preconditioners based on approximate commutators*, SIAM J. Sci. Comput., (2005). To appear.

[165] H. C. Elman, V. E. Howle, J. Shadid, and R. Tuminaro, *A parallel block multi-level preconditioner for the 3D incompressible Navier-Stokes equations*, J. Comput. Phys., 187 (2003), pp. 505–523.
[166] H. C. Elman, D. Loghin, and A. J. Wathen, *Preconditioning techniques for Newton's method for the incompressible Navier-Stokes equations*, BIT, 43 (2003), pp. 961–974.
[167] H. C. Elman and D. J. Silvester, *Fast nonsymmetric iterations and preconditioning for Navier-Stokes equations*, SIAM J. Sci. Comput., 17 (1996), pp. 33–46.
[168] H. C. Elman, D. J. Silvester, and A. J. Wathen, *Block preconditioners for the discrete incompressible Navier-Stokes equations*, Internat. J. Numer. Methods Fluids, 40 (2002), pp. 333–344.
[169] ———, *Performance and analysis of saddle point preconditioners for the discrete steady-state Navier-Stokes equations*, Numer. Math., 90 (2002), pp. 665–688.
[170] ———, *Finite Elements and Fast Iterative Solvers*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, UK, 2005.
[171] O. G. Ernst, *Residual-minimizing Krylov subspace methods for stabilized discretizations of convection-diffusion equations*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1079–1101.
[172] R. E. Ewing, R. D. Lazarov, P. Lu, and P. S. Vassilevski, *Preconditioning indefinite systems arising from mixed finite element discretization of second-order elliptic problems*, in Preconditioned Conjugate Gradient Methods (Nijmegen, 1989), vol. 1457 of Lecture Notes in Math., Springer, Berlin, 1990, pp. 28–43.
[173] V. Faber, W. Joubert, E. Knill, and T. Manteuffel, *Minimal residual method stronger than polynomial preconditioning*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 707–729.
[174] V. Faber and T. Manteuffel, *Necessary and sufficient conditions for the existence of a conjugate gradient method*, SIAM J. Numer. Anal., 21 (1984), pp. 352–362.
[175] C. Farhat and F.-X. Roux, *A method of finite element tearing and interconnecting and its parallel solution algorithm*, Int. J. Numer. Methods Engng., 32 (1991), pp. 1205–1227.
[176] B. Fischer, *Polynomial Based Iteration Methods for Symmetric Linear Systems*, Wiley-Teubner Series Advances in Numerical Mathematics, John Wiley & Sons Ltd., Chichester,UK, 1996.
[177] B. Fischer and F. Peherstorfer, *Chebyshev approximation via polynomial mappings and the convergence behaviour of Krylov subspace methods*, Electr. Trans. Numer. Anal., 12 (2001), pp. 205–215.
[178] B. Fischer, A. Ramage, D. J. Silvester, and A. J. Wathen, *Minimum residual methods for augmented systems*, BIT, 38 (1998), pp. 527–543.
[179] P. F. Fischer and J. W. Lottes, *Hybrid Schwarz-multigrid methods for the spectral element method: Extensions to Navier-Stokes*, in Domain Decomposition Methods in Science and Engineering, R. Kornhuber, R. H. W. Hoppe, J. Périaux, O. Pironneau, O. B. Widlund, and J. Xu, eds., Lecture notes in Computer Science and Engineering, vol. 40, Springer, 2004, pp. 35–49.
[180] J. Fish and Y. Qu, *Global basis two-level method for indefinite systems. Part 1: Convergence studies*, Int. J. Numer. Methods Engng., 49 (2000), pp. 439–460.
[181] R. Fletcher, *Conjugate gradient methods for indefinite systems*, in Numerical Analysis 1975 (Proc 6th Biennial Dundee Conf., Univ. Dundee, Dundee, 1975), Springer, Berlin, 1976, pp. 73–89. Lecture Notes in Math., Vol. 506.
[182] ———, *Factorizing symmetric indefinite matrices*, Linear Algebra Appl., 14 (1976), pp. 257–272.
[183] ———, *Practical Methods of Optimization*, John Wiley & Sons Ltd., Chichester,UK, second ed., 1987.
[184] R. Fletcher and T. Johnson, *On the stability of null-space methods for KKT systems*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 938–958.
[185] A. Forsgren, *On linear least-squares problems with diagonally dominant weight matrices*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 763–788.
[186] ———, *Inertia-controlling factorizations for optimization algorithms*, Appl. Numer. Math., 43 (2002), pp. 91–107.
[187] A. Forsgren, P. E. Gill, and J. R. Shinnerl, *Stability of symmetric ill-conditioned systems arising in interior methods for constrained optimization*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 187–211.
[188] A. Forsgren and W. Murray, *Newton methods for large-scale linear equality-constrained minimization*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 560–587.
[189] M. Fortin, *Finite element solution of the Navier-Stokes equations*, Acta Numerica 1993, pp. 239–284.
[190] M. Fortin and A. Fortin, *A generalization of Uzawa's algorithm for the solution of the Navier-Stokes equations*, Comm. Appl. Numer. Meth., 1 (1985), pp. 205–208.
[191] M. Fortin and R. Glowinski, *Augmented Lagrangian Methods. Applications to the Numerical Solution of Boundary Value Problems*, vol. 15 of Studies in Mathematics and its Applications, North-Holland Publishing Co., Amsterdam, 1983. Translated from the French by B. Hunt and D. C. Spicer.
[192] R. Fourer and S. Mehrotra, *Solving symmetric indefinite systems in an interior-point method for linear programming*, Math. Programming, 62 (1993), pp. 15–39.
[193] A. Frangioni and C. Gentile, *New preconditioners for KKT systems of network flow problems*, SIAM J. Optim., 14 (2004), pp. 894–913.
[194] R. W. Freund, *Model reduction methods based on Krylov subspaces*, Acta Numerica 2003, pp. 267–319.
[195] ———, *A transpose-free quasi-minimal residual algorithm for non-Hermitian linear systems*, SIAM J. Sci. Comput., 14 (1993), pp. 470–482.

[196] ———, *Preconditioning of symmetric, but highly indefinite linear systems*, in Proceedings of the 15th IMACS World Congress on Scientific Computation, Modeling and Applied Mathematics, A. Sydow, ed., Wissenschaft und Technik Verlag, 1997, pp. 551–556.
[197] R. W. Freund, G. H. Golub, and N. M. Nachtigal, *Iterative solution of linear systems*, Acta Numerica 1992, pp. 57–100.
[198] R. W. Freund and F. Jarre, *A QMR-based interior-point algorithm for solving linear programs*, Math. Programming, 76 (1996), pp. 183–210.
[199] R. W. Freund and N. M. Nachtigal, *QMR: a quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math., 60 (1991), pp. 315–339.
[200] R. W. Freund and N. M. Nachtigal, *A new Krylov-subspace method for symmetric indefinite linear systems*, in Proceedings of the 14th IMACS World Congress on Computational and Applied Mathematics, W. F. Ames, ed., IMACS, 1994, pp. 1253–1256.
[201] R. W. Freund and N. M. Nachtigal, *Software for simplified Lanczos and QMR algorithms*, Appl. Numer. Math., 19 (1995), pp. 319–341.
[202] W. N. Gansterer, J. Schneid, and C. W. Ueberhuber, *Mathematical properties of equilibrium systems*, Technical Report AURORA TR2003-13, University of Vienna and Vienna University of Technology, 2003.
[203] A. Gauthier, F. Saleri, and A. Quarteroni, *A fast preconditioner for the incompressible Navier-Stokes equations*, Comput. Visual. Sci., 6 (2004), pp. 105–112.
[204] A. George and K. Ikramov, *On the condition of symmetric quasi-definite matrices*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 970–977.
[205] ———, *The closedness of certain classes of matrices with respect to pseudoinversion*, Comput. Math. Math. Phys., 42 (2002), pp. 1242–1246.
[206] A. George, K. Ikramov, and B. Kucherov, *Some properties of symmetric quasi-definite matrices*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1318–1323.
[207] J. Gilbert and M. T. Heath, *Computing a sparse basis for the null space*, SIAM J. Alg. Disc. Meth., 8 (1987), pp. 446–459.
[208] P. E. Gill, W. Murray, D. B. Ponceleón, and M. A. Saunders, *Preconditioners for indefinite systems arising in optimization*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 292–311.
[209] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, *Inertia-controlling methods for general quadratic programming*, SIAM Rev., 33 (1991), pp. 1–36.
[210] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press Inc. [Harcourt Brace Jovanovich Publishers], London, 1981.
[211] P. E. Gill, M. A. Saunders, and J. R. Shinnerl, *On the stability of Cholesky factorization for symmetric quasidefinite systems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 35–46.
[212] V. Girault and P. A. Raviart, *Finite Element Approximation of the Navier-Stokes Equations*, vol. 749 of Lecture Notes in Mathematics, Springer-Verlag, New York, 1986.
[213] R. Glowinski, *Numerical Methods for Nonlinear Variational Problems*, Springer Series in Computational Physics, Springer-Verlag, New York, 1984.
[214] ———, *Finite Element Methods for Incompressible Viscous Flow*, vol. IX of Handbook of Numerical Analysis. Numerical Methods for Fluids (Part 3), North-Holland Publishing Co., Amsterdam, 2003.
[215] R. Glowinski and P. Le Tallec, *Augmented Lagrangian and Operator Splitting Methods in Nonlinear Mechanics*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1989.
[216] R. Glowinski and O. Pironneau, *Numerical methods for the first biharmonic equation and for the two-dimensional Stokes problem*, SIAM Rev., 21 (1979), pp. 167–212.
[217] I. Gohberg, P. Lancaster, and L. Rodman, *Matrices and Indefinite Scalar Products*, vol. 8 of Operator Theory: Advances and Applications, Birkhäuser, Basel, 1983.
[218] P. Goldfeld, L. F. Pavarino, and O. B. Widlund, *Balancing Neumann-Neumann preconditioners for mixed approximations of heterogeneous problems in linear elasticity*, Numer. Math., 95 (2003), pp. 283–324.
[219] G. H. Golub and C. Greif, *On solving block-structured indefinite linear systems*, SIAM J. Sci. Comput., 24 (2003), pp. 2076–2092.
[220] G. H. Golub and M. L. Overton, *The convergence of inexact Chebyshev and Richardson iterative methods for solving linear systems*, Numer. Math., 53 (1988), pp. 571–493.
[221] G. H. Golub and C. F. Van Loan, *Unsymmetric positive definite linear systems*, Lin. Algebra Appl., 28 (1979), pp. 85–98.
[222] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, Baltimore, MD, third ed., 1996.
[223] G. H. Golub and D. Vanderstraeten, *On the preconditioning of matrices with skew-symmetric splittings*, Numer. Algorithms, 25 (2000), pp. 223–239.
[224] G. H. Golub and A. J. Wathen, *An iteration for indefinite systems and its application to the Navier-Stokes equations*, SIAM J. Sci. Comput., 19 (1998), pp. 530–539.
[225] G. H. Golub, X. Wu, and J.-Y. Yuan, *SOR-like methods for augmented systems*, BIT, 41 (2001), pp. 71–85.
[226] R. Gonzales and R. Woods, *Digital Image Processing*, Addison-Wesley Publishing Co., New York, NY, 1992.
[227] N. I. M. Gould, *On practical conditions for the existence and uniqueness of solutions to the general equality*

quadratic programming problem, Math. Programming, 32 (1985), pp. 90–99.
[228] N. I. M. Gould, M. E. Hribar, and J. Nocedal, *On the solution of equality constrained quadratic programming problems arising in optimization*, SIAM J. Sci. Comput., 23 (2001), pp. 1376–1395.
[229] N. I. M. Gould and J. A. Scott, *A numerical evaluation of HSL packages for the direct solution of large sparse, symmetric linear systems of equations*, ACM Trans. Math. Software, 30 (2004), pp. 300–325.
[230] I. G. Graham, A. Spence, and E. Vainikko, *Parallel iterative methods for Navier-Stokes equations and application to eigenvalue calculation*, Concurrency Computat.: Pract. Exper., 15 (2003), pp. 1151–1168.
[231] A. Greenbaum, *Comparison of splittings used with the conjugate gradient algorithm*, Numer. Math., 33 (1979), pp. 181–193.
[232] ———, *Iterative Methods for Solving Linear Systems*, vol. 17 of Frontiers in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.
[233] ———, *Generalizations of the field of values useful in the study of polynomial functions of a matrix*, Linear Algebra Appl., 347 (2002), pp. 233–249.
[234] A. Greenbaum and L. Gurvits, *Max-min properties of matrix factor norms*, SIAM J. Sci. Comput., 15 (1994), pp. 348–358.
[235] A. Greenbaum, V. Pták, and Z. Strakoš, *Any nonincreasing convergence curve is possible for GMRES*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 465–469.
[236] A. Greenbaum and Z. Strakoš, *Matrices that generate the same Krylov residual spaces*, in Recent Advances in Iterative Methods, vol. 60 of IMA Vol. Math. Appl., Springer, New York, 1994, pp. 95–118.
[237] C. Greif, G. H. Golub, and J. M. Varah, *Augmented Lagrangian techniques for solving saddle point linear systems*, SIAM J. Matrix Anal. Appl., (2004). To appear.
[238] P. M. Gresho, *Incompressible fluid dynamics: some fundamental formulation issues*, Annu. Rev. Fluid Mech., 23 (1991), pp. 413–453.
[239] P. M. Gresho and R. L. Sani, *Incompressible Flow and the Finite Element Method. Advection-Diffusion and Isothermal Laminar Flow*, John Wiley & Sons Ltd., Chichester, UK, 1998.
[240] M. Griebel, T. Neunhoeffer, and H. Regler, *Algebraic multigrid methods for the solution of the Navier-Stokes equations in complicated geometries*, Int. J. Numer. Meth. Fluids, 26 (1998), pp. 281–301.
[241] M. Gulliksson, *Iterative refinement for constrained and weighted linear least squares*, BIT, 34 (1996), pp. 239–253.
[242] M. Gulliksson, X.-Q. Jin, and Y. Wei, *Perturbation bounds for constrained and weighted least squares problems*, Linear Algebra Appl., 349 (2002), pp. 221–232.
[243] M. Gulliksson and P.-A. Wedin, *Modifying the QR-decomposition to constrained and weighted linear least squares*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 1298–1313.
[244] M. Gunzburger, *Finite Element Methods for Viscous Incompressible Flows*, Academic Press, San Diego, 1989.
[245] K. Gustafsson and R. Hartmann, *Divergence-free bases for finite element schemes in hydrodynamics*, SIAM J. Numer Anal., 20 (1983), pp. 697–721.
[246] E. Haber and U. M. Ascher, *Preconditioned all-at-once methods for large, sparse parameter estimation problems*, Inverse Problems, 17 (2001), pp. 1847–1864.
[247] E. Haber, U. M. Ascher, and D. Oldenburg, *On optimization techniques for solving nonlinear inverse problems*, Inverse Problems, 16 (2000), pp. 1263–1280.
[248] E. Haber and J. Modersitzki, *Numerical methods for volume-preserving image registration*, Inverse Problems, 20 (2004), pp. 1621–1638.
[249] W. Hackbusch, *Multi-grid Methods and Applications*, Springer Verlag, Berlin, 1985.
[250] ———, *Iterative Solution of Large Sparse Systems of Equations*, Springer Verlag, Berlin, 1994.
[251] G. Hadley, *Nonlinear and Dynamic Programming*, Addison-Wesley Publishing Co., Reading, MA, 1964.
[252] M. Hagemann and O. Schenk, *Weighted matchings for the preconditioning of symmetric indefinite linear systems*, Technical Report CS-2004-005, Department of Computer Science, University of Basel, 2004.
[253] C. A. Hall, *Numerical solution of Navier-Stokes problems by the dual variable method*, SIAM J. Alg. Disc. Meth., 6 (1985), pp. 220–236.
[254] C. A. Hall, J. C. Cavendish, and W. H. Frey, *The dual variable method for solving fluid flow difference equations on Delaunay triamgulations*, Comput. Fluids, 20 (1991), pp. 145–164.
[255] E. L. Hall, *Computer Image Processing and Recognition*, Academic Press, New York, NY, 1979.
[256] F. H. Harlow and J. E. Welch, *Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface*, Phys. Fluids, 8 (1965), pp. 2182–2189.
[257] B. Hassibi, A. H. Sayed, and T. Kailath, *Linear estimation in Krein spaces—Part I: Theory*, IEEE Trans. Automat. Control, 41 (1996), pp. 18–33.
[258] J. C. Haws, *Preconditioning KKT Systems*, PhD thesis, Department of Mathematics, North Carolina State University, Raleigh, NC, 2002.
[259] J. C. Haws and C. D. Meyer, *Preconditioning KKT systems*, Technical Report M&CT-Tech-01-021, The Boeing Company, 2003.
[260] M. T. Heath, R. J. Plemmons, and R. C. Ward, *Sparse orthogonal schemes for structural optimization using the force method*, SIAM J. Sci. Stat. Comp., 5 (1984), pp. 514–532.
[261] L. Hemmingsson-Frändén and A. Wathen, *A nearly optimal preconditioner for the Navier-Stokes equations*,

Numer. Linear Algebra Appl., 8 (2001), pp. 229–243.
[262] J. C. Henderson and E. A. W. Maunder, *A problem in applied topology: on the selection of cycles for the flexibility analysis of skeletal structures*, J. Inst. Maths. Applics., 5 (1969), pp. 254–269.
[263] P. Heres and W. Schilders, *Practical issues of model order reduction with Krylov-subspace methods*, in Progress in Electromagnetic Research Symposium (PIERS 2004), 2004. To appear.
[264] M. A. Heroux, R. A. Bartlett, V. E. Howle, R. J. Hoekstra, J. J. Hu, T. G. Kolda, R. B. Lehoucq, K. R. Long, R. P. Pawlowski, E. T. Phipps, A. G. Salinger, H. K. Thornquist, R. S. Tuminaro, J. S. Willenbring, A. Williams, and K. S. Stanley, *An overview of the Trilinos project*, ACM Trans. Math. Software, (2004). To appear.
[265] M. R. Hestenes, *Multiplier and gradient methods*, J. Optim. Theory Appl., 4 (1969), pp. 303–320.
[266] ———, *Optimization Theory. The Finite Dimensional Case*, John Wiley & Sons, New York, NY, 1975.
[267] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Research Nat. Bur. Standards, 49 (1952), pp. 409–436.
[268] R. Hiptmair, *Multilevel Preconditioning for Mixed Problems in Three Dimensions*, vol. 8 of Augsburger Mathematisch-Naturwissenschaftliche Schriften, Wissner, Augsburg, Germany, 1996.
[269] ———, *Multigrid method for H(div) in three dimensions*, Electron. Trans. Numer. Anal., 6 (1997), pp. 133–152.
[270] R. Hiptmair and R. H. W. Hoppe, *Multilevel methods for mixed finite elements in three dimensions*, Numer. Math., 82 (1999), pp. 253–279.
[271] R. Hiptmair, T. Shiekofer, and B. Wohlmuth, *Multilevel preconditioned augmented Lagrangian techniques for 2nd order mixed problems*, Comput., 57 (1996), pp. 25–48.
[272] R. Hochmuth, *Stable multiscale discretization for saddle point problems and preconditioning*, Numer. Funct. Anal. Optim., 19 (1998), pp. 789–806.
[273] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1985.
[274] P. D. Hough and S. D. Vavasis, *Complete orthogonal decomposition for weighted least squares*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 369–392.
[275] Q. Hu, Z. Shi, and D. Yu, *Efficient solvers for saddle-point problems arising from domain decompositions with Lagrange multipliers*, SIAM J. Numer. Anal., 42 (2004), pp. 905–933.
[276] Q. Hu and J. Zou, *An iterative method with variable relaxation parameters for saddle-point problems*, SIAM J. Matrix Anal. Appl., 23 (2001), pp. 317–338.
[277] ———, *Two new variants of nonlinear inexact Uzawa algorithms for saddle-point problems*, Numer. Math., 93 (2002), pp. 333–359.
[278] ———, *Substructuring preconditioners for saddle-point problems arising from Maxwell's equations in three dimensions*, Math. Comp., 73 (2003), pp. 35–61.
[279] M. Huhtanen, *A Hermitian Lanczos method for normal matrices*, SIAM J. Matrix Anal. Appl., 23 (2002), pp. 1092–1108.
[280] K. D. Ikramov, *On a certain type of preconditioning for KKT-matrices*, Moscow Univ. Comput. Math. Cybernet., 2 (2003), pp. 1–3.
[281] I. C. F. Ipsen, *A note on preconditioning nonsymmetric matrices*, SIAM J. Sci. Comput., 23 (2001), pp. 1050–1051.
[282] K. Ito and K. Kunisch, *An active set strategy based on the augmented Lagrangian formulation for image restoration*, Math. Model. Numer. Anal., 33 (1999), pp. 1–21.
[283] D. James, *Implicit null space iterative methods for constrained least squares problems*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 962–978.
[284] D. James and R. J. Plemmons, *An iterative substructuring algorithm for equilibrium equations*, Numer. Math., 57 (1990), pp. 625–633.
[285] X.-Q. Jin, *A preconditioner for constrained and weighted least squares problems with Toeplitz structure*, BIT, 36 (1996), pp. 101–109.
[286] W. Joubert, *A robust GMRES-based adaptive polynomial preconditioning algorithm for nonsymmetric linear systems*, SIAM J. Sci. Comput., 15 (1994), pp. 427–439.
[287] J. Júdice, J. Patricio, L. Portugal, M. Resende, and G. Veiga, *A study of preconditioners for network interior point methods*, Comput. Optim. Appl., 24 (2003), pp. 5–35.
[288] X. Juvigny, *Solution of Large Linear Systems on Massively Parallel Machines*, PhD thesis, Applied Mathematics, University of Paris VI, Paris, France, 1997. In French.
[289] I. Kaneko, M. Lawo, and G. Thierauf, *On computational procedures for the force method*, Int. J. Numer. Methods Engng., 18 (1982), pp. 1469–1495.
[290] I. Kaneko and R. J. Plemmons, *Minimum norm solutions to linear elastic analysis problems*, Int. J. Numer. Methods Engng., 20 (1984), pp. 983–998.
[291] G. Kanschat, *Preconditioning methods for local discontinuous Galerkin discretizations*, SIAM J. Sci. Comput., 25 (2003), pp. 815–831.
[292] O. A. Karakashian, *On a Galerkin-Lagrange multiplier method for the stationary Navier-Stokes equations*, SIAM J. Numer. Anal., 19 (1982), pp. 909–923.
[293] A. Kaveh, *Application of Topology and Matroid Theory to the Analysis of Structures*, PhD thesis, Imperial College of Science and Technology, London, UK, 1974.

[294] ———, *A combinatorial optimization problem: optimal generalized cycle bases*, Comput. Methods Appl. Mech. Engrg., 20 (1979), pp. 39–51.
[295] ———, *Recent developments in the force method of structural analysis*, Appl. Mech. Rev., 45 (1992), pp. 401–418.
[296] ———, *Structural Mechanics: Graph and Matrix Methods*, Research Studies Press, UK, third ed., 2004.
[297] D. Kay, D. Loghin, and A. J. Wathen, *A preconditioner for the steady-state Navier-Stokes equations*, SIAM J. Sci. Comput., 24 (2002), pp. 237–256.
[298] C. Keller, N. I. M. Gould, and A. J. Wathen, *Constraint preconditioning for indefinite linear systems*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1300–1317.
[299] G. R. Kirchhoff, *Über die Auflösung der Gleichungen, auf welche man bei der Untersuchung der linearen Verteilung galvanischer Ströme geführt wird*, Ann. Phys. Chem., 72 (1847), pp. 497–508.
[300] T. H. Kjeldsen, *A contextualized historical analysis of the Kuhn-Tucker theorem in nonlinear programming: the impact of World War II*, Historia Mathematica, 27 (2000), pp. 331–361.
[301] A. Klawonn, *Block-triangular preconditioners for saddle point problems with a penalty term*, SIAM J. Sci. Comput., 19 (1998), pp. 172–184.
[302] ———, *An optimal preconditioner for a class of saddle point problems with a penalty term*, SIAM J. Sci. Comput., 19 (1998), pp. 540–552.
[303] A. Klawonn and L. F. Pavarino, *Overlapping Schwarz methods for mixed linear elasticity and Stokes problems*, Comput. Methods Appl. Mech. Engrg., 165 (1998), pp. 233–245.
[304] ———, *A comparison of overlapping Schwarz methods and block preconditioners for saddle point problems*, Numer. Linear Algebra Appl., 7 (2000), pp. 1–25.
[305] A. Klawonn and G. Starke, *Block triangular preconditioners for nonsymmetric saddle point problems: field-of-values analysis*, Numer. Math., 81 (1999), pp. 577–594.
[306] A. Klawonn and O. B. Widlund, *A domain decomposition method with Lagrange multipliers and inexact solves for linear elasticity*, SIAM J. Sci. Comput., 22 (2000), pp. 1199–1219.
[307] ———, *FETI and Neumann-Neumann iterative substructuring methods: Connections and new results*, Comm. Pure Applied Math., 54 (2001), pp. 57–90.
[308] A. Klawonn and O. B. Widlund, *Dual-Primal FETI methods for linear elasticity*, Technical Report TR2004-855, Courant Institute, New York University, 2004.
[309] L. Komszik, P. Poschmann, and I. Sharapov, *A preconditioning technique for indefinite linear systems*, Finite Elements Anal. Des., 26 (1997), pp. 253–258.
[310] J. Korzak, *Eigenvalue relations and conditions of matrices arising in linear programming*, Comput., 62 (1999), pp. 45–54.
[311] R. Kouhia and C. M. Menken, *On the solution of second-order post-buckling fields*, Commun. Numer. Meth. Engng., 11 (1995), pp. 443–453.
[312] P. Krzyżanowski, *On block preconditioners for nonsymmetric saddle point problems*, SIAM J. Sci. Comput., 23 (2001), pp. 157–169.
[313] A. Kunoth, *Fast iterative solution of saddle point problems in optimal control based on wavelets*, Comput. Optim. Appl., 22 (2002), pp. 225–259.
[314] Y. A. Kuznetsov, *Efficient iterative solvers for elliptic finite element problems on nonmatching grids*, Russian J. Numer. Anal. Math. Modelling, 10 (1995), pp. 187–211.
[315] ———, *Efficient preconditioner for mixed finite element methods on nonmatching meshes*, Russian J. Numer. Anal. Math. Modelling, 19 (2004), pp. 163–172.
[316] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Research Nat. Bur. Standards, 45 (1950), pp. 255–282.
[317] U. Langer and W. Queck, *On the convergence factor of Uzawa's algorithm*, J. Comput. Appl. Math., 15 (1986), pp. 191–202.
[318] K. H. Leem, S. Oliveira, and D. E. Stewart, *Algebraic multigrid AMG for saddle point systems from meshfree discretizations*, Numer. Linear Algebra Appl., 11 (2004), pp. 293–308.
[319] R. B. Lehoucq and A. G. Salinger, *Large-scale eigenvalue calculations for stability analysis of steady flows on massively parallel computers*, Int. J. Numer. Meth. Fluids, 36 (2001), pp. 309–327.
[320] W. Leontief, F. Duchin, and D. B. Szyld, *New approaches in economic analysis*, Science, 228 (1985), pp. 419–422.
[321] C. Li, Z. Li, D. J. Evans, and T. Zhang, *A note on an SOR-like method for augmented systems*, IMA J. Numer. Anal., 23 (2003), pp. 581–592.
[322] C. Li and C. Vuik, *Eigenvalue analysis of the SIMPLE preconditioning for incompressible flow*, Numer. Linear Algebra Appl., 11 (2004), pp. 511–523.
[323] N. Li and Y. Saad, *Crout versions of ILU factorization with pivoting for sparse symmetric matrices*, Technical Report UMSI-2004-044, Minnesota Supercomputer Institute, 2004.
[324] J. Liesen, E. de Sturler, A. Sheffer, Y. Aydin, and C. Siefert, *Preconditioners for indefinite linear systems arising in surface parameterization*, in Proceedings of the 10th International Meshing Round Table, 2001, pp. 71–81.
[325] J. Liesen, M. Rozložník, and Z. Strakoš, *Least squares residuals and minimal residual methods*, SIAM J.

Sci. Comput., 23 (2002), pp. 1503–1525.
[326] J. Liesen and P. E. Saylor, *Orthogonal Hessenberg reduction and orthogonal Krylov subspace bases*, SIAM J. Numer. Anal., (To appear).
[327] J. Liesen and Z. Strakoš, *GMRES convergence analysis for a convection-diffusion model problem*, SIAM J. Sci. Comput., (To appear).
[328] J. Liesen and P. Tichý, *Behavior of CG and MINRES for symmetric tridiagonal Toeplitz matrices*, Technical Report 34-2004, Technical University of Berlin, Department of Mathematics, 2004.
[329] ———, *The worst-case GMRES for normal matrices*, BIT, 44 (2004), pp. 79–98.
[330] L. Little and Y. Saad, *Block preconditioners for saddle point problems*, Numer. Algorithms, 33 (2003), pp. 367–379.
[331] L. Little, Y. Saad, and L. Smoch, *Block LU preconditioners for symmetric and nonsymmetric saddle point problems*, SIAM J. Sci. Comput., 25 (2003), pp. 729–748.
[332] J. W.-H. Liu, *A partial pivoting strategy for sparse symmetric matrix decomposition*, ACM Trans. Math. Softw., 13 (1987), pp. 173–182.
[333] W. M. Liu and S. Xu, *A new improved Uzawa method for finite element solution of Stokes' problem*, Comput. Mech., 27 (2001), pp. 305–310.
[334] D. Loghin and A. J. Wathen, *Schur complement preconditioners for the Navier-Stokes equations*, Internat. J. Numer. Methods Fluids, 40 (2002), pp. 403–412.
[335] ———, *Schur complement preconditioning for elliptic systems of partial differential equations*, Numer. Linear Algebra Appl., 10 (2003), pp. 423–443.
[336] ———, *Analysis of preconditioners for saddle-point problems*, SIAM J. Sci. Comput., 25 (2004), pp. 2029–2049.
[337] T.-T. Lu and S.-H. Shiou, *Inverses of* 2×2 *block matrices*, Comput. Math. Appl., 43 (2002), pp. 119–129.
[338] G. Lube and M. A. Olshanskii, *Stable finite element calculation of incompressible flows using the rotation form of convection*, IMA J. Numer. Anal., 22 (2002), pp. 437–461.
[339] G. Luenberger, *Linear and Nonlinear Programming*, Addison-Wesley Publishing Co., Reading, MA, 1984.
[340] L. Lukšan and J. Vlček, *Indefinitely preconditioned inexact Newton method for large sparse equality constrained non-linear programming problems*, Numer. Linear Algebra Appl., 5 (1998), pp. 219–247.
[341] ———, *Numerical experience with iterative methods for equality constrained non-linear programming problems*, Optim. Methods Softw., 16 (2001), pp. 257–287.
[342] T. Lyche, T. K. Nilssen, and R. Winther, *Preconditioned iterative methods for scattered data interpolation*, Adv. Comput. Math., 17 (2002), pp. 237–256.
[343] D. S. Mackey, N. Mackey, and F. Tisseur, *Structured tools for structured matrices*, Electr. J. Linear Algebra, 10 (2003), pp. 106–145.
[344] Y. Maday, D. Meiron, A. T. Patera, and E. M. Ronquist, *Analysis of iterative methods for the steady and unsteady Stokes problem. Application to spectral element discretizations*, SIAM J. Sci. Comput., 14 (1993), pp. 310–337.
[345] H. Mahawar and V. Sarin, *Parallel iterative methods for dense linear systems in inductance extraction*, Parallel Comput., 29 (2003), pp. 1219–1235.
[346] D. S. Malkus, *Eigenproblems associated with the discrete LBB condition for incompressible finite elements*, Int. J. Engng. Sci., 19 (1981), pp. 1299–1310.
[347] K.-A. Mardal and R. Winther, *Uniform preconditioners for the time dependent Stokes problem*, Numer. Math., 98 (2004), pp. 305–327.
[348] H. M. Markowitz, *Portfolio Selection: Efficient Diversification of Investments*, John Wiley, New York, NY, 1959.
[349] H. M. Markowitz and A. F. Perold, *Sparsity and piecewise linearity in large portfolio optimization problems*, in Sparse Matrices and Their Uses, I. S. Duff, ed., London, 1981, Academic Press, pp. 89–108.
[350] J. Martikainen, *Numerical study of two sparse AMG methods*, Math. Model. Numer. Anal., 37 (2003), pp. 133–142.
[351] J. Maryška, M. Rozložník, and M. Tůma, *Mixed-hybrid finite element approximation of the potential fluid flow problem*, J. Comput. Appl. Math., 63 (1995), pp. 383–392.
[352] ———, *The potential fluid flow problem and the convergence rate of the minimal residual method*, Numer. Linear Algebra Applic., 3 (1996), pp. 525–542.
[353] ———, *Schur complement systems in the mixed-hybrid finite element approximation of the potential fluid flow problem*, SIAM J. Sci. Comput., 22 (2000), pp. 704–723.
[354] MATLAB, *The MathWorks Company, Natick, MA.* http://www.mathworks.com.
[355] P. Matstoms, *Sparse QR Factorization with Applications to Linear Least Squares Problems*, PhD thesis, Linköping University, Linköping, Sweden, 1994.
[356] W. McGuire and R. H. Gallagher, *Matrix Structural Analysis*, John Wiley, New York, NY, 1979.
[357] K. Meerbergen and A. Spence, *Implicitly restarted Arnoldi with purification for the shift-invert transformation*, Math. Comp., 66 (1997), pp. 667–689.
[358] S. Mehrotra and J. Wang, *Conjugate gradient based implementation of interior point methods for network flow problems*, in Linear and Nonlinear Conjugate Gradient-Related Methods, L. Adams and L. Nazareth, eds., Society for Industrial and Applied Mathematics, Philadelphia, 1996, pp. 124–142.

[359] O. Meshar and S. Toledo, *An out-of-core sparse symmetric indefinite factorization method*, ACM Trans. Math. Softw., (2004). Submitted.
[360] G. Meurant, *Computer Solution of Large Linear Systems*, vol. 28 of Studies in Mathematics and its Applications, North-Holland Publishing Co., Amsterdam, 1999.
[361] M. D. Mihajlović and D. J. Silvester, *Efficient parallel solvers for the biharmonic equation*, Parallel Comput., 30 (2004), pp. 35–55.
[362] J. Modersitzki, *Numerical Methods for Image Registration*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, UK, 2003.
[363] J. D. Moulton, J. E. Morel, and U. M. Ascher, *Approximate Schur complement preconditioning of the lowest-order nodal discretization*, SIAM J. Sci. Comput., 19 (1998), pp. 185–205.
[364] M. F. Murphy, G. H. Golub, and A. J. Wathen, *A note on preconditioning for indefinite linear systems*, SIAM J. Sci. Comput., 21 (2000), pp. 1969–1972.
[365] N. M. Nachtigal, S. C. Reddy, and L. N. Trefethen, *How fast are nonsymmetric matrix iterations?*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 778–795.
[366] A. E. Naiman, I. M. Babuška, and H. C. Elman, *A note on conjugate gradient convergence*, Numer. Math., 76 (1997), pp. 209–230.
[367] S. G. Nash and A. Sofer, *Preconditioning reduced matrices*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 47–68.
[368] E. G. Ng and B. W. Peyton, *Block sparse Cholesky algorithms on advanced uniprocessor computers*, SIAM J. Sci. Comput., 14 (1993), pp. 1034–1056.
[369] H. Q. Nguyen, *Domain Decomposition Methods for Linear-Quadratic Elliptic Optimal Control Problems*, PhD thesis, Department of Computational and Applied Mathematics, Rice University, Houston, TX, 2004.
[370] R. A. Nicolaides, *Existence, uniqueness and approximation for generalized saddle point problems*, SIAM J. Numer. Anal., 19 (1982), pp. 349–357.
[371] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer Series in Operations Research, Springer-Verlag, Berlin, 1999.
[372] R. H. Nochetto and J.-H. Pyo, *Optimal relaxation parameter for the Uzawa method*, Numer. Math., 98 (2004), pp. 695–702.
[373] B. Nour-Omid and P. Wriggers, *A two-level iteration method for solution of contact problems*, Comput. Methods Appl. Mech. Engrg., 54 (1986), pp. 131–144.
[374] A. R. L. Oliveira and D. C. Sorensen, *A new class of preconditioners for large-scale linear systems for interior point methods for linear programming*, Linear Algebra Appl., 394 (2004), pp. 1–24.
[375] M. A. Olshanskii, *An iterative solver for the Oseen problem and numerical solution of incompressible Navier-Stokes equations*, Numer. Linear Algebra Appl., 6 (1999), pp. 353–378.
[376] M. A. Olshanskii and A. Reusken, *Navier-Stokes equations in rotation form: a robust multigrid solver for the velocity problem*, SIAM J. Sci. Comput., 23 (2002), pp. 1683–1706.
[377] J. M. Ortega, *Numerical Analysis: A Second Course*, Computer Science and Applied Mathematics, Academic Press, New York and London, 1972.
[378] C. C. Paige and M. A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.
[379] ———, *LSQR: An algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Software, 8 (1982), pp. 43–71.
[380] S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, McGraw-Hill, New York, 1980.
[381] ———, *A calculation procedure for tw-dimensional elliptic situations*, Numer. Heat Transfer, 4 (1981), pp. 409–425.
[382] S. V. Patankar and D. B. Spalding, *A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows*, Int. J. Heat Mass Transfer, 15 (1972), pp. 1787–1806.
[383] L. F. Pavarino, *Preconditioned conjugate residual methods for mixed spectral discretizations of elasticity and Stokes problems*, Comput. Methods Appl. Mech. Engrg., 146 (1997), pp. 19–30.
[384] ———, *Preconditioned mixed spectral element methods for elasticity and Stokes problems*, SIAM J. Sci. Comput., 19 (1998), pp. 1941–1957.
[385] ———, *Indefinite overlapping Schwarz methods for time-dependent Stokes problems*, Comput. Methods Appl. Mech. Engrg., 187 (2000), pp. 35–51.
[386] L. F. Pavarino and O. B. Widlund, *Iterative substructuring methods for spectral element discretizations of elliptic systems. II: Mixed methods for linear elasticity and Stokes flow*, SIAM J. Numer. Anal., 37 (2000), pp. 375–402.
[387] ———, *Balancing Neumann-Neumann methods for incompressible Stokes equations*, Comm. Pure Applied Math., 55 (2002), pp. 302–335.
[388] M. Pernice, *A hybrid multigrid method for the steady-state incompressible Navier-Stokes equations*, Electr. Trans. Numer. Anal., 10 (2000), pp. 74–91.
[389] J. B. Perot, *An analysis of the fractional step method*, J. Comput. Phys., 108 (1993), pp. 51–58.
[390] I. Perugia, *A field-based mixed formulation for the 2-d magnetostatic problem*, SIAM J. Numer. Anal., 34 (1997), pp. 2382–2391.

[391] I. Perugia and V. Simoncini, *Block-diagonal and indefinite symmetric preconditioners for mixed finite element formulations*, Numer. Linear Algebra Appl., 7 (2000), pp. 585–616.
[392] I. Perugia, V. Simoncini, and M. Arioli, *Linear algebra methods in a mixed approximation of magnetostatic problems*, SIAM J. Sci. Comput., 21 (1999), pp. 1085–1101.
[393] R. Peyret and T. D. Taylor, *Computational Methods for Fluid Flow*, Springer Series in Computational Physics, Springer-Verlag, Berlin, 1983.
[394] R. J. Plemmons, *A parallel block iterative scheme applied to computations in structural analysis*, SIAM J. Algebraic Discrete Methods, 7 (1986), pp. 337–347.
[395] R. J. Plemmons and R. E. White, *Substructuring methods for computing the nullspace of equilibrium matrices*, SIAM J. Matrix Anal. Appl., 11 (1990), pp. 1–22.
[396] B. T. Polyak, *Iterative methods using Lagrange multipliers for solving extremal problems with constraints of the equation type*, USSR Comp. Math. Math. Phys., 10 (1970), pp. 42–52.
[397] A. Pothen, *Sparse Null Bases and Marriage Theorems*, PhD thesis, Department of Computer Science, Cornell University, Ithaca, NY, 1984.
[398] ———, *Sparse null basis computations in structural optimization*, Numer. Math., 55 (1989), pp. 501–519.
[399] C. E. Powell and D. Silvester, *Optimal preconditioning for Raviart-Thomas mixed formulation of second-order elliptic problems*, SIAM J. Matrix Anal. Appl., 25 (2004), pp. 718–738.
[400] M. J. D. Powell, *A method for nonlinear constraints in minimization problems*, in Optimization, R. Fletcher, ed., Academic Press, London, 1969, pp. 283–298.
[401] ———, *On updating the inverse of a KKT matrix*, Technical Report DAMTP2004/NA01, Department of Applied Mathematics and Theoretical Physics, Cambridge University, 2004.
[402] M. L. Psiaki and K. Park, *Parallel orthogonal factorization null-space method for dynamic-quadratic programming*, J. Optim. Theory Appl., 85 (1995), pp. 409–434.
[403] Y. Qu, *Multilevel Methods for Indefinite Systems*, PhD thesis, Department of Civil Engineering, Renssealer Polytechnic Institute, Troy, NY, 2000.
[404] Y. Qu and J. Fish, *Global basis two-level method for indefinite systems. Part 2: Computational issues*, Int. J. Numer. Methods Engng., 49 (2000), pp. 461–478.
[405] ———, *Multifrontal incomplete factorization for indefinite and complex symmetric systems*, Int. J. Numer. Methods Engng., 53 (2002), pp. 1433–1459.
[406] A. Quarteroni, F. Saleri, and A. Veneziani, *Factorization methods for the numerical approximation of Navier-Stokes equations*, Comput. Methods Appl. Mech. Engrg., 188 (2000), pp. 505–526.
[407] A. Quarteroni and A. Valli, *Numerical Approximation of Partial Differential Equations*, vol. 23 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, 1994.
[408] ———, *Domain Decomposition Methods for Partial Differential Equations*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, UK, 1999.
[409] W. Queck, *The convergence factor of preconditioned algorithms of the Arrow–Hurwicz type*, SIAM J. Numer. Anal., 26 (1989), pp. 1016–1030.
[410] M. J. Raw, *A coupled algebraic multigrid method for the 3D Navier-Stokes equations*, in Fast Solvers for Flow Problems, Proc. 10th GAMM-Seminar, W. Hackbusch and G. Wittum, eds., Notes on Numerical Fluid Mechanics, Wiesbaden, Germany, 1995, Vieweg, pp. 204–215.
[411] W.-Q. Ren and J.-X. Zhao, *Iterative methods with preconditioners for indefinite systems*, J. Comput. Math., 17 (1999), pp. 89–96.
[412] M. Resende and G. Veiga, *An implementation of the dual affine scaling algorithm for minimum-cost flow on bipartite uncapacitated networks*, SIAM J. Optim., 3 (1993), pp. 516–537.
[413] M. P. Robichaud, P. A. Tanguy, and M. Fortin, *An iterative implementation of the Uzawa algorithm for 3-D fluid flow problems*, Int. J. Numer. Meth. Fluids, 10 (1990), pp. 429–442.
[414] J. Robinson, *Integrated Theory of Finite Element Methods*, John Wiley, New York, NY, 1973.
[415] D. Rose, *Triangulated graphs and the elimination process*, J. Math. Anal. Appl., 32 (1970), pp. 597–609.
[416] M. Rozložník and V. Simoncini, *Krylov subspace methods for saddle point problems with indefinite preconditioning*, SIAM J. Matrix Anal. Appl., 24 (2002), pp. 368–391.
[417] T. Rusten, P. S. Vassilevski, and R. Winther, *Interior penalty preconditioners for mixed finite element approximations of elliptic problems*, Math. Comp., 65 (1996), pp. 4473–466.
[418] T. Rusten and R. Winther, *A preconditioned iterative method for saddlepoint problems*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 887–904.
[419] ———, *Substructure preconditioners for elliptic saddle-point problems*, Math. Comp., 60 (1993), pp. 23–48.
[420] Y. Saad, *Krylov subspace methods for solving large unsymmetric linear systems*, Math. Comp., 37 (1981), pp. 105–126.
[421] ———, *The Lanczos biorthogonalization algorithm and other oblique projection methods for solving large unsymmetric systems*, SIAM J. Numer. Anal., 19 (1982), pp. 485–506.
[422] ———, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM J. Sci. Comput., 14 (1993), pp. 856–869.
[423] ———, *Iterative Methods for Sparse Linear Systems*, Society for Industrial and Applied Mathematics, Philadelphia, PA, second ed., 2003.
[424] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric*

linear systems, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 856–869.
[425] P. Saint-Georges, Y. Notay, and G. Warzée, *Efficient iterative solution of constrained finite element analyses*, Comput. Methods Appl. Mech. Engrg., 160 (1998), pp. 101–114.
[426] F. Saleri and A. Veneziani, *Pressure-correction algebraic splitting methods for the incompressible Navier-Stokes equations*, SIAM J. Numer. Anal., (2005). To appear.
[427] A. Sameh and V. Sarin, *Parallel algorithms for indefinite linear systems*, Parallel Comput., 28 (2002), pp. 285–299.
[428] V. Sarin, *Efficient Iterative Methods for Saddle Point Problems*, PhD thesis, Department of Computer Science, University of Illinois, Urbana-Champaign, IL, 1997.
[429] V. Sarin and A. Sameh, *An efficient iterative method for the generalized Stokes problem*, SIAM J. Sci. Comput., 19 (1998), pp. 206–226.
[430] ———, *Hierarchical divergence-free bases and their application to particulate flows*, Trans. ASME, 70 (2003), pp. 44–49.
[431] O. Schenk and K. Gärtner, *On sparse factorization pivoting methods for sparse symmetric indefinite systems*, Technical Report CS-2004-004, Department of Computer Science, University of Basel, 2004.
[432] J. Schöberl and W. Zulehner, *On Schwarz-type smoothers for saddle point problems*, Numer. Math., 95 (2003), pp. 377–399.
[433] G. J. Shaw and S. Sivaloganathan, *On the smoothing of the SIMPLE pressure correction algorithm*, Int. J. Numer. Meth. Fluids, 8 (1988), pp. 441–462.
[434] Y. Shi, *Solving linear systems involved in constrained optimization*, Lin. Algebra Appl., 229 (1995), pp. 175–189.
[435] R. Sibson and G. Stone, *Computation of thin-plate splines*, SIAM J. Sci. Stat. Comput., 12 (1991), pp. 1304–1313.
[436] A. Sidi, *A zero-cost preconditioning for a class of indefinite linear systems*, WSEAS Trans. Math., 2 (2003), pp. 142–150.
[437] C. Siefert and E. de Sturler, *Preconditioners for generalized saddle-point problems*, Technical Report UIUCDCS-R-2004-2448, Department of Computer Science, University of Illinois, 2004.
[438] D. Silvester, H. Elman, D. Kay, and A. Wathen, *Efficient preconditioning of the linearized Navier-Stokes equations for incompressible flow*, J. Comput. Appl. Math., 128 (2001), pp. 261–279.
[439] D. J. Silvester and A. J. Wathen, *Fast iterative solution of stabilised Stokes systems. II. Using general block preconditioners*, SIAM J. Numer. Anal., 31 (1994), pp. 1352–1367.
[440] V. Simoncini, *Block triangular preconditioners for symmetric saddle point problems*, Appl. Numer. Math., 49 (2004), pp. 63–80.
[441] ———, *Personal communication*, (August 2004).
[442] V. Simoncini and M. Benzi, *A note on the eigenvalues of a class of saddle point matrices*, (2004). In preparation.
[443] ———, *Spectral properties of the Hermitian and skew-Hermitian splitting preconditioner for saddle point problems*, SIAM J. Matrix Anal. Appl., 26 (2004), pp. 377–389.
[444] V. Simoncini and D. B. Szyld, *Theory of inexact Krylov subspace methods and applications to scientific computing*, SIAM J. Sci. Comput., 25 (2003), pp. 454–477.
[445] ———, *The effect of non-optimal bases on the convergence of Krylov subspace methods*, (submitted).
[446] G. Starke, *Field-of-values analysis of preconditioned iterative methods for nonsymmetric elliptic problems*, Numer. Math., 78 (1997), pp. 103–117.
[447] G. W. Stewart, *On scaled projections and pseudoinverses*, Linear Algebra Appl., 112 (1989), pp. 189–193.
[448] G. Stoyan, *Iterative Stokes solvers in the harmonic Velte subspace*, Computing, 67 (2001), pp. 13–33.
[449] G. Strang, *Introduction to Applied Mathematics*, Wellesley-Cambridge Press, Wellesley, MA, 1986.
[450] ———, *A framework for equilibrium equations*, SIAM Rev., 30 (1988), pp. 283–297.
[451] J. C. Strikwerda, *An iterative method for solving finite difference approximations to the Stokes equations*, SIAM J. Numer. Anal., 21 (1984), pp. 447–458.
[452] K. Stüben, *A review of algebraic multigrid*, J. Comput. Appl. Math., 128 (2001), pp. 281–309.
[453] T. Stykel, *Balanced truncation model reduction for semidiscretized Stokes equations*, Linear Algebra Appl., (2004). To appear.
[454] J. G. Sun, *Structured backward errors for KKT systems*, Linear Algebra Appl., 288 (1999), pp. 75–88.
[455] Syamsudhuha and D. J. Silvester, *Efficient solution of the steady-state Navier-Stokes equations by a multigrid preconditioned Newton-Krylov method*, Int. J. Numer. Meth. Fluids, 43 (2003), pp. 1407–1427.
[456] D. B. Szyld, *Using sparse matrix techniques to solve a model of the world economy*, in Sparse Matrices and Their Uses, I. S. Duff, ed., Academic Press, 1981, pp. 357–365.
[457] M. Tůma, *A note on the LDL^T decomposition of matrices from saddle-point problems*, SIAM J. Matrix Anal. Appl., 23 (2002), pp. 903–915.
[458] R. E. Tarjan, *Data Structures and Network Algorithms*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1983.
[459] R. Temam, *Navier-Stokes Equations. Theory and Numerical Analysis*, vol. 2 of Studies in Mathematics and its Applications, North-Holland Publishing Co., Amsterdam, third ed., 1984.

[460] J. L. Thomas, B. Diskin, and A. Brandt, *Textbook multigrid efficiency for the incompressible Navier-Stokes equations: high Reynolds number wakes and boundary layers*, Computers & Fluids, 30 (2001), pp. 853–874.
[461] M. J. Todd, *A Dantzig-Wolfe-like variant of Karmarkar's interior point linear programming algorithm*, Oper. Res., 38 (1990), pp. 1006–1018.
[462] K.-C. Toh, *GMRES vs. ideal GMRES*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 30–36.
[463] K.-C. Toh, K.-K. Phoon, and S.-H. Chan, *Block preconditioners for symmetric indefinite linear systems*, Int. J. Numer. Methods Engng., 60 (2004), pp. 1361–1381.
[464] Z. Tong and A. Sameh, *On an iterative method for saddle point problems*, Numer. Math., 79 (1998), pp. 643–646.
[465] A. Topcu, *A Contribution to the Systematic Analysis of Finite Element Structures Using the Force Method*, PhD thesis, University of Essen, Germany, 1979. In German.
[466] A. Toselli and O. B. Widlund, *Domain Decomposition Methods - Algorithms and Theory*, Springer Series in Computational Mathematics, vol. 34, Springer Verlag, 2004.
[467] A. M. Tropper, *Matrix Theory for Electrical Engineers*, vol. 1 of International Series in the Engineering Sciences, Addison-Wesley Publishing Co., New York, NY, 1962.
[468] U. Trottenberg, C. W. Oosterlee, and A. Schüller, *Multigrid*, Academic Press, London, UK, 2001.
[469] S. Turek, *Efficient Solvers for Incompressible Flow Problems. An Algorithmic and Computational Approach*, vol. 6 of Lecture Notes in Computational Science and Engineering, Springer-Verlag, Berlin, 1999.
[470] E. Tyrtyshnikov and Y. Vassilevski, *A mosaic preconditioner for a dual Schur complement*, in Numerical Mathematics and Advanced Applications, F. Brezzi, A. Buffa, S. Corsaro, and A. Murli, eds., Springer Verlag, 2003, pp. 867–877.
[471] H. Uzawa, *Iterative methods for concave programming*, in Studies in Linear and Nonlinear Programming, K. J. Arrow, L. Hurwicz, and H. Uzawa, eds., Stanford University Press, Stanford, CA, 1958, pp. 154–165.
[472] P. Vaidya, *Solving linear equations with diagonally dominant matrices by constructing good preconditioners*, technical report, Department of Computer Science, University of Illinois, 1990.
[473] E. Vainikko and I. G. Graham, *A parallel solver for PDE systems and application to the incompressible Navier-Stokes equations*, Appl. Numer. Math., 49 (2004), pp. 97–116.
[474] H. A. van der Vorst, *Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 631–644.
[475] ———, *Iterative Krylov Methods for Large Linear Systems*, vol. 13 of Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, Cambridge, 2003.
[476] C. F. Van Loan, *On the method of weighting for equality constrained least squares problems*, SIAM J. Numer. Anal., 23 (1985), pp. 851–864.
[477] R. J. Vanderbei, *Symmetric quasidefinite matrices*, SIAM J. Optim., 5 (1995), pp. 100–113.
[478] S. P. Vanka, *Block-implicit multigrid solution of Navier-Stokes equations in primitive variables*, J. Comput. Phys., 65 (1986), pp. 138–158.
[479] R. S. Varga, *Matrix Iterative Analysis*, Prentice-Hall Inc., Englewood Cliffs, N.J., 1962.
[480] P. S. Vassilevski and R. D. Lazarov, *Preconditioning mixed finite element saddle-point elliptic probems*, Numer. Linear Algebra Appl., 3 (1996), pp. 1–20.
[481] P. S. Vassilevski and J. Wang, *Multilevel iterative methods for mixed mixed finite element discretizations of elliptic probems*, Numer. Math., 63 (1992), pp. 513–520.
[482] S. A. Vavasis, *Stable numerical algorithms for equilibrium systems*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 1108–1131.
[483] ———, *Stable finite elements for problems with wild coefficients*, SIAM J. Numer. Anal., 33 (1996), pp. 890–916.
[484] A. Veneziani, *Block factorized preconditioners for high-order accurate in time approximation of the Navier-Stokes equations*, Numer. Methods Partial Differential Eq., 19 (2003), pp. 487–510.
[485] R. Verfürth, *A combined conjugate gradient-multigrid algorithm for the numerical solution of the Stokes problem*, IMA J. Numer Anal., 4 (1984), pp. 441–455.
[486] ———, *A multilevel algorithm for mixed problems*, SIAM J. Numer Anal., 21 (1984), pp. 264–271.
[487] C. Vincent and R. Boyer, *A preconditioned conjugate gradient Uzawa-type method for the solution of the Stokes problem by mixed Q_1-P_0 stabilized finite elements*, Int. J. Numer. Meth. Fluids, 14 (1992), pp. 289–298.
[488] M. Wabro, *Coupled algebraic multigrid methods for the Oseen problem*, Comput. Visual. Sci., 7 (2004), pp. 141–151.
[489] C. Wagner, W. Kinzelbach, and G. Wittum, *Schur-complement multigrid: a robust method for groundwater flow and transport problems*, Numer. Math., 75 (1997), pp. 523–545.
[490] Y. Wang, *Overlapping Schwarz preconditioner for the mixed formulation of plane elasticity*, Appl. Numer. Math., (2005). To appear.
[491] J. S. Warsa, M. Benzi, T. A. Wareing, and J. E. Morel, *Preconditioning a mixed discontinuous finite element method for radiation diffusion*, Numer. Linear Algebra Appl., 11 (2004), pp. 795–811.
[492] A. Wathen, B. Fischer, and D. Silvester, *The convergence rate of the minimal residual method for the Stokes problem*, Numer. Math., 71 (1995), pp. 121–134.
[493] A. Wathen and D. Silvester, *Fast iterative solution of stabilised Stokes systems. I. Using simple diagonal*

preconditioners, SIAM J. Numer. Anal., 30 (1993), pp. 630–649.
[494] R. Webster, *An algebraic multigrid solver for Navier-Stokes problems*, Int. J. Numer. Meth. Fluids, 18 (1996), pp. 761–780.
[495] M. Wei, *Algebraic properties of the rank-deficient equality constrained and weighted least squares problems*, Linear Algebra Appl., 161 (1992), pp. 27–44.
[496] ———, *Perturbation theory for the rank-deficient equality constrained least squares problem*, SIAM J. Numer Anal., 29 (1992), pp. 1462–1481.
[497] Y. Wei and N. Zhang, *Further note on constraint preconditioning for nonsymmetric indefinite matrices*, Appl. Math. Comp., 152 (2004), pp. 43–46.
[498] P. Wesseling, *An Introduction to Multigrid Methods*, John Wiley & Sons Ltd., Chichester, UK, 1992.
[499] ———, *Principles of Computational Fluid Dynamics*, vol. 29 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, 2001.
[500] O. B. Widlund, *A Lanczos method for a class of nonsymmetric systems of linear equations*, SIAM J. Numer. Anal., 15 (1978), pp. 801–812.
[501] S. Ø. Wille and A. F. D. Loula, A priori *pivoting in solving the Navier-Stokes equations*, Commun. Numer. Meth. Engng., 18 (2002), pp. 691–698.
[502] S. Ø. Wille, O. Staff, and A. F. D. Loula, *Parallel ILU preconditioning,* a priori *pivoting and segregation of variables for iterative solution of the mixed finite element formulation of the Navier-Stokes equations*, Int. J. Numer. Meth. Fluids, 41 (2003), pp. 977–996.
[503] G. Wittum, *Multi-grid methods for Stokes and Navier-Stokes equations with transforming smoothers: Algorithms and numerical results*, Numer. Math., 54 (1989), pp. 543–563.
[504] ———, *On the convergence of multi-grid methods with transforming smoothers*, Numer. Math., 57 (1990), pp. 15–38.
[505] P. Wolfe, *The reduced gradient method*, technical report, The RAND Corporation, Santa Monica, CA, 1962. Unpublished.
[506] M. H. Wright, *Interior methods for constrained optimization*, Acta Numerica 1992, pp. 341–407.
[507] S. J. Wright, *Primal-Dual Interior Point Methods*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997.
[508] X. Wu, B. P. B. Silva, and J. Y. Yuan, *Conjugate gradient method for rank deficient saddle point problems*, Numer. Algorithms, 35 (2004), pp. 139–154.
[509] D. Yang, *Iterative schemes for mixed finite element methods with applications to elasticity and compressible flow problems*, Numer. Math., 83 (2002), pp. 177–200.
[510] J.-X. Zhao, *The generalized Cholesky factorization method for saddle point problems*, Appl. Math. Comp., 92 (1998), pp. 49–58.
[511] O. C. Zienkiewicz, J. P. Vilotte, S. Toyoshima, and S. Nakazawa, *Iterative method for constrained and mixed approximation. An inexpensive improvement of F.E.M. performance*, Comput. Methods Appl. Mech. Engrg., 51 (1985), pp. 3–29.
[512] A. Zsaki, D. Rixen, and M. Paraschivoiu, *A substructure-based iterative inner solver coupled with Uzawa's algorithm for the Stokes problem*, Int. J. Numer. Meth. Fluids, 43 (2003), pp. 215–230.
[513] W. Zulehner, *A class of smoothers for saddle point problems*, Comput., 65 (2000), pp. 227–246.
[514] ———, *Analysis of iterative methods for saddle point problems: a unified approach*, Math. Comp., 71 (2002), pp. 479–505.