Technical Report

TR-2008-007

A Generalization of the Hermitian and Skew-Hermitian Splitting Iteration

by

Michele Benzi

Mathematics and Computer Science

EMORY UNIVERSITY

A GENERALIZATION OF THE HERMITIAN AND SKEW-HERMITIAN SPLITTING ITERATION

MICHELE BENZI*

To Gene Golub, in memoriam

Abstract. This paper is concerned with a generalization of the Hermitian and skew-Hermitian splitting iteration for solving positive definite, non-Hermitian linear systems. It is shown that the new scheme has some advantages over the standard HSS method, and can be used as an effective preconditioner for certain linear systems in saddle point form. Numerical experiments using discretizations of incompressible flow problems demonstrate the effectiveness of the generalized HSS preconditioner.

Key words. matrix splittings, HSS iteration, preconditioning, saddle point systems, generalized Stokes and Oseen problems, Navier–Stokes equations

AMS subject classifications. Primary 65F10, 65N22, 65F50. Secondary 15A06.

1. Introduction. The Hermitian and skew-Hermitian Splitting (HSS) iteration was first introduced by Bai, Golub and Ng in [4] for the solution of a broad class of non-Hermitian linear systems $Ax = b$. The basic iterative scheme works as follows. Let $\alpha > 0$ be fixed. Denoting by $H = (A + A^*)/2$ the Hermitian part of A and by $S = (A - A^*)/2$ the skew-Hermitian part, the HSS method is the alternating iteration

$$\begin{cases} (H + \alpha I)\, x_{k+\frac{1}{2}} = (\alpha I - S)\, x_k + b, \\ (S + \alpha I)\, x_{k+1} = (\alpha I - H)\, x_{k+\frac{1}{2}} + b \end{cases} \tag{1.1}$$

$(k = 0, 1, \ldots)$, where x_0 is an arbitrary initial guess. It was shown in [4] that when H is positive definite the HSS iteration is unconditionally convergent, i.e., the sequence $\{x_k\}$ converges to the solution $x_* = A^{-1}b$ as $k \to \infty$ for all $\alpha > 0$ and for any choice of x_0. Moreover, it was shown in the same paper that choosing $\alpha = \sqrt{ab}$, where $a = \lambda_{\min}(H)$ and $b = \lambda_{\max}(H)$ are the extreme eigenvalues of H, minimizes an upper bound on the spectral radius of the iteration matrix associated with the stationary scheme (1.1).

Due to its promising performance and elegant mathematical properties, the HSS scheme immediately attracted considerable attention, resulting in numerous papers devoted to various aspects of the new algorithm.[1] In one direction, the method was extended to the solution of saddle point problems in [7, 8]; see also [1, 5, 12, 21]. Notice that in these problems the Hermitian part of A is generally singular, requiring a different convergence analysis. Another natural development, first considered in [7, 8], was to use the HSS scheme not as a stationary iterative solver, but as a preconditioner for Krylov subspace methods [20], resulting in a far more efficient and robust class of solvers. Other significant developments include "preconditioned" variants of the HSS iteration, the extension to certain singular systems, studies on the optimal selection of iteration parameters, and application of the HSS preconditioner to specific

*Department of Mathematics and Computer Science, Emory University, Atlanta, Georgia 30322, USA (benzi@mathcs.emory.edu). The work of this author was supported by the National Science Foundation grant DMS-0511336.

[1] It is worth mentioning that the original HSS paper [4] was one of just 22 included in the volume [13] of selected works of Gene Golub.

problems like convection-diffusion equations, incompressible flow problems, and real formulations of complex linear systems, often with excellent results; see [2, 3, 6, 9, 11] and the references therein.

A potential difficulty with the HSS approach is the need to solve two linear systems at each iteration (or at each application of the preconditioner), one with matrix $H + \alpha I$ and one with matrix $S + \alpha I$, cf. (1.1). The first of these two systems is Hermitian positive definite and often well-conditioned (at least for α not too small) and its solution does not present great difficulties. The shifted skew-Hermitian system, however, can be much more problematic; in some cases its solution is as difficult as that of the original linear system $Ax = b$. There are situations where the matrix S is structured in such a way as to make systems involving $S+\alpha I$ easy to solve; see [10] for an example arising in image processing, and [9] for one arising in fluid mechanics, as well as section 3 below. In general, however, this will not be the case. It is therefore not surprising that beginning with the first HSS paper [7] considerable attention has been devoted to the use of *inexact* solves. In a typical situation, the two linear systems in (1.1) are solved to relatively low accuracy, usually by inner iterative schemes which could be some preconditioned Krylov iteration or a multigrid method. There is now considerable evidence that the good convergence properties are preserved even when the inner solves are performed to rather low accuracy, resulting in considerable savings, especially for very large problems; see, e.g., [9].

In this paper a generalization of the original HSS scheme is presented and some of its basic properties are studied. The idea is to split H into the sum of two Hermitian positive semidefinite matrices: $H = G + K$, where K is is of simple form (e.g., diagonal), and to associate K to the skew-Hermitian portion S of A, so that $A = G + (S + K)$. The generalized HSS scheme is based on the splittings

$$A = (G + \alpha I) - (\alpha I - S - K) \quad \text{and} \quad A = (S + K + \alpha I) - (\alpha I - G)\,.$$

When either G or K is positive definite the resulting scheme, which reduces to the classical HSS method when $K = 0$, is shown to be convergent for all $\alpha > 0$; see the next section. Its rate of convergence is often faster than that of the standard HSS scheme. However, the main interest of the generalized scheme consists in the fact that solution of systems with coefficient matrix $S + K + \alpha I$ by inner iterations is made easier, since this matrix is more "diagonally dominant" (loosely speaking) and typically better conditioned than $S + \alpha I$. The new variant (to be used primarily as a preconditioner for a Krylov subspace method) was motivated by the type of matrices that arise in the solution of time-dependent incompressible flow problems, but it is not restricted to them and can be described in a completely algebraic manner.

The remainder of the paper is organized as follows: in section 2 the generalized HSS scheme is described and some of its properties are discussed. Types of problems for which the generalized HSS method appears to be well-suited are described in section 3 and a few numerical tests are discussed in section 4. Concluding remarks are given in section 5.

2. The generalized HSS scheme. First, a brief review of the classical HSS iteration is needed. This method is based on the splitting $A = H + S$, which in turn induces the two splittings

$$A = (H + \alpha I) - (\alpha I - S) \qquad \text{and} \qquad A = (S + \alpha I) - (\alpha I - H)\,,$$

with $\alpha > 0$. Alternating between these two splittings yields the HSS iteration (1.1). Elimination of $x_{k+\frac{1}{2}}$ from the second of (1.1) yields

$$x_{k+1} = T_\alpha x_k + P_\alpha^{-1} b$$

$(k = 0, 1, \ldots)$, where

$$T_\alpha = I - P_\alpha^{-1} A, \quad P_\alpha = \frac{1}{2\alpha}(H + \alpha I)(S + \alpha I)\,.$$

Here $A = P_\alpha - (P_\alpha - A)$ is the splitting induced by the HSS iteration, and P_α is the HSS preconditioner. The iteration matrix T_α is given by

$$T_\alpha = (S + \alpha I)^{-1}(\alpha I - H)(H + \alpha I)^{-1}(\alpha I - S)\,,$$

which is similar to

$$\hat{T}_\alpha = (\alpha I - H)(H + \alpha I)^{-1}(\alpha I - S)(S + \alpha I)^{-1}\,.$$

Hence, the following bound for the spectral radius of T_α holds:

$$\rho(T_\alpha) = \rho(\hat{T}_\alpha) \le \|(\alpha I - H)(H + \alpha I)^{-1}\|\,\|(\alpha I - S)(S + \alpha I)^{-1}\|\,,$$

where $\|\cdot\|$ denotes the spectral norm. Since $(\alpha I - S)(S + \alpha I)^{-1}$ is the Cayley transform of a skew-Hermitian matrix, it is unitary (see, e.g., [16]) and therefore it has norm 1. Denoting by $\sigma(H)$ the spectrum of H it follows that

$$\rho(T_\alpha) \le \max_{\lambda \in \sigma(H)} \frac{|\lambda - \alpha|}{|\lambda + \alpha|} < 1\,,$$

showing that the method converges for all $\alpha > 0$. As a consequence, the spectrum of the preconditioned matrix $P_\alpha^{-1} A$ lies inside the circle of center $(1, 0)$ and radius 1 in the complex plane.

If the Hermitian part H is only positive semidefinite (i.e., at least one of the eigenvalues of H is zero), then the foregoing upper bound on the spectral radius becomes $\rho(T_\alpha) \le 1$ and the HSS method may not converge, in general. However, in the special case of (generalized) saddle point problems the method can be shown to be unconditionally convergent; see [8, Theorem 3.1].

Numerical experiments show that the HSS iteration is especially effective if either H or S "dominates," i.e., if either part of A is much larger than the other. For instance, assume that $A = H + S = \varepsilon L + S$ where $\varepsilon > 0$ is a (small) parameter, $L = L^*$ is positive definite, and $S = -S^*$. Assume further that L and S have been scaled so as to have approximately the same norm. Then

$$P_\alpha = \frac{1}{2\alpha}(\varepsilon L + \alpha I)(S + \alpha I) = \frac{1}{2} A + \frac{\varepsilon}{2\alpha} LS + \frac{\alpha}{2} I\,.$$

This shows that up to a numerical factor, the HSS preconditioner P_α can be regarded as an approximate factorization of the coefficient matrix A; for fixed α and for $\varepsilon \to 0$, the matrix $2P_\alpha$ approaches $A + \alpha I$, which is a good approximation of A when α is small. Taking $\alpha = \sqrt{\varepsilon}$ yields a preconditioner that (apart from the factor $1/2$) approaches A as $\varepsilon \to 0$. This heuristic argument helps explain the good performance of the method on linear systems with dominant skew-Hermitian part; obviously, a similar heuristic argument applies in case the Hermitian part dominates.

Consider now the situation where the Hermitian part H can be split as $H = \varepsilon L + K$ ($\varepsilon > 0$ small) where $L = L^*$ is positive definite and $K = K^*$ is positive semidefinite, possibly positive definite. (For example, K could be a matrix of the form $K = \sigma M$ where σ is related to a time step, e.g., $\sigma = O(1/\Delta t)$, and M is a mass matrix, or possibly a scaled identity.) In this case it is easy to see that P_α does not yield a good approximation of A as $\varepsilon \to 0$ (for any α), and indeed the performance of HSS on this type of problems is sometimes poor. The following extension of the HSS method may be used instead. Write $G = \varepsilon L$ and consider the splittings

$$A = (G + \alpha I) - (\alpha I - S - K) \quad \text{and} \quad A = (S + K + \alpha I) - (\alpha I - G),$$

together with the corresponding alternating iterative scheme

$$\begin{cases} (G + \alpha I)\, x_{k+\frac{1}{2}} = (\alpha I - S - K)\, x_k + b, \\ (S + K + \alpha I)\, x_{k+1} = (\alpha I - G)\, x_{k+\frac{1}{2}} + b\,, \end{cases} \tag{2.1}$$

($k = 0, 1, \ldots$). The following convergence result holds.

THEOREM 2.1. *Let $A = (G + K) + S = H + S$ where G and K are Hermitian positive semidefinite and S is skew-Hermitian. If either G or K is positive definite, the alternating iteration (2.1) converges unconditionally to the unique solution of $Ax = b$.*

Proof. Elimination of $x_{k+\frac{1}{2}}$ from (2.1) leads to the iterative scheme $x_{k+1} = T_\alpha x_k + P_\alpha^{-1} b$ where

$$T_\alpha = (S + K + \alpha I)^{-1}(\alpha I - G)(G + \alpha I)^{-1}(\alpha I - S - K) = I - P_\alpha^{-1} A\,,$$

with $P_\alpha = \frac{1}{2\alpha}(G + \alpha I)(S + K + \alpha I)$. The iteration matrix T_α is similar to

$$\hat{T}_\alpha = (\alpha I - G)(G + \alpha I)^{-1}(\alpha I - S - K)(S + K + \alpha I)^{-1}\,.$$

Clearly,

$$\|(\alpha I - G)(G + \alpha I)^{-1}\| = \rho((\alpha I - G)(G + \alpha I)^{-1}) = \max_{\lambda \in \sigma(G)} \frac{|\alpha - \lambda|}{|\alpha + \lambda|} \le 1\,,$$

with the inequality being strict if G is positive definite. Moreover, by Kellogg's Lemma (see, e.g., [18]) it holds that

$$\|(\alpha I - S - K)(S + K + \alpha I)^{-1}\| \le 1\,,$$

the inequality again being strict if K is positive definite. Therefore if either G or K is positive definite it holds

$$\rho(T_\alpha) = \rho(\hat{T}_\alpha) \le \|(\alpha I - G)(G + \alpha I)^{-1}\|\, \|(\alpha I - S - K)(S + K + \alpha I)^{-1}\| < 1\,,$$

showing the unconditional convergence of the iteration (2.1). ∎

Some remarks are in order. First of all, it is clear that the iterative scheme (2.1) is a generalization of the standard HSS scheme (1.1), to which it reduces whenever $K = 0$.

Secondly, if G and K are both positive definite then the iteration matrix T_α is similar to a matrix $\hat{T}_\alpha$ which is the product of two factors, $(\alpha I - G)(G + \alpha I)^{-1}$ and $(\alpha I - S - K)(S + K + \alpha I)^{-1}$, both of which are contractions with respect to the spectral norm. In contrast, in the original HSS matrix T_α is similar to a matrix $\hat{T}_\alpha$ which is

the product of a contraction times an isometry. This is not enough to conclude that the new scheme converges (asymptotically) faster than the standard HSS iteration, since it can be easily seen by examples that the spectral radius of $(G+\alpha I)^{-1}(G-\alpha I)$ is often larger than that of $(H+\alpha I)^{-1}(H-\alpha I)$. This increase, however, may be offset by the fact that the norm of $(S+K+\alpha I)^{-1}(S+K-\alpha I)$ is strictly less than 1 when K is positive definite, and numerical experiments show that the the spectral radius is often smaller for the new method.

As a simple example, let $n = 100$ and let $A = G + K + S$ where $G = \varepsilon L$ with $\varepsilon = 10^{-3}$ and

$$L = \begin{bmatrix} 2 & -1 & & 0 \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ 0 & & -1 & 2 \end{bmatrix}, \quad S = \begin{bmatrix} 0 & 0.1 & & 0 \\ -0.1 & \ddots & \ddots & \\ & \ddots & \ddots & 0.1 \\ 0 & & -0.1 & 0 \end{bmatrix}, \quad K = 10^{-1} I_n \,.$$

Then for $\alpha = 0.1$ the spectral radius of the iteration matrix is $\rho = 0.8999$ using the standard HSS splitting and $\rho = 0.7999$ using the generalized HSS splitting.

It is instructive to interpret the corresponding preconditioner

$$P_\alpha = \frac{1}{2\alpha}(\varepsilon L + \alpha I)(S + K + \alpha I)$$

as an approximate factorization of A. Indeed, expanding the product yields

$$P_\alpha = \frac{1}{2}A + \frac{\varepsilon}{2\alpha}L(S+K) + \frac{\alpha}{2}I \,.$$

Again, this shows that up to a numerical factor, the new preconditioner P_α can be regarded as an approximate factorization of the coefficient matrix A; for fixed α and for $\varepsilon \to 0$, the matrix $2P_\alpha$ approaches $A + \alpha I$, which is a good approximation of A when α is small. Again, taking $\alpha = \sqrt{\varepsilon}$ yields a preconditioner that (apart from the factor $1/2$) approaches A as $\varepsilon \to 0$.

A few comments are in order concerning the "optimal" choice of the parameter α. In the literature, a great deal of effort has been put in determining the value of α which minimizes the spectral radius of the iteration matrix T_α, or some upper bound on it. Such estimates on α can be given for the generalized HSS method using for example the approach detailed in [18, Chapter 4]. However, the practical usefulness of such estimates is questionable. First of all, the estimated value of α usually depends on spectral information that may not be accessible, such as the extreme eigenvalues of H (or, in the case of the generalized HSS method, of G). Second, minimizing the spectral radius or an upper bound on it does not always result in the best choice of α when the stationary iteration is accelerated by a Krylov method; see, e.g., the analysis and examples in [7]. Furthermore, when these techniques are used as preconditioners the exact value of α is not as important as when the algorithms are used as iterative solvers. Since Krylov acceleration dramatically improves the rate of convergence of these methods (with a largely negligible increase in cost per iteration), there is little reason to try to estimate the "optimal" value of α. Experience suggests that in most applications, and for an appropriate scaling of the problem, a "small" value of α (usually between 0.01 and 0.5) gives good results.

As already mentioned, the main potential advantage of the generalized HSS scheme over the classical one is the fact that the solution of linear sytems with coefficient matrix $S + K + \alpha I$, where $K \neq 0$ is positive semidefinite and of simple form

(e.g., diagonal or block diagonal with blocks of small size, or perhaps banded) can be expected to be less expensive than in the classical HSS scheme ($K = 0$) in the common situation where inner iterative solvers are used. For example, when K is a nonnegative diagonal matrix, the corresponding system will have a "heavier" diagonal and be better conditioned. Of course, the solution of systems involving the matrix $H+\alpha I$ is now replaced by the solution of systems with matrix $G+\alpha I$, which has now a somewhat "weaker" diagonal and could be somewhat less well-conditioned. However, linear systems with matrix $G + \alpha I$ will typically be still fairly easy to solve since this matrix is Hermitian positive definite and efficient solvers exist for problems of this kind. Some examples are considered in the next section.

3. Some applications of the new scheme. As already noted, the HSS preconditioner has been successfully tested on problems from incompressible fluid dynamics; see in particular [9, 17]. An important example is given by the *unsteady Navier–Stokes equations*:

$$\frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega \times [0, T] \tag{3.1}$$

$$\operatorname{div} \mathbf{u} = 0 \quad \text{in } \Omega \times [0, T] \tag{3.2}$$

where $\nu > 0$ is the kinematic viscosity, $\Omega \subset \mathbb{R}^3$ an open bounded region with sufficiently smooth boundary $\partial\Omega$, $[0, T]$ a time interval, $\mathbf{u}$ the unknown velocity field, and p the pressure. The unknown functions $u = u(\mathbf{x}, t)$ and $p = p(\mathbf{x}, t)$ are subject to suitable boundary and initial conditions. When implicit methods are used to integrate (3.1)-(3.2) in time and a simple linearization is applied, a sequence of semidiscrete, linear boundary value problems of the form

$$\sigma \mathbf{u} - \nu \Delta \mathbf{u} + \mathbf{w} \cdot \nabla \mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega \tag{3.3}$$

$$\operatorname{div} \mathbf{u} = 0 \quad \text{in } \Omega \tag{3.4}$$

$$\mathbf{u} = \mathbf{g} \quad \text{on } \partial\Omega \tag{3.5}$$

is obtained. This problem is often referred to as the *generalized Oseen problem*. Here $\mathbf{w}$ is a given divergence-free vector field (the "wind"), and $\sigma > 0$ is related to the time step Δt, for example, $\sigma = O(1/\Delta t)$. In the 3D case, this is a system of four partial differential equations, to be solved for the unknowns $\mathbf{u} = (u, v, w)$ and p. For $\sigma = 0$, the steady-state case is obtained. For $\mathbf{w} = \mathbf{0}$ problem (3.3)-(3.5) reduces to the *generalized Stokes problem*, which is also of considerable importance in the numerical solution of the unsteady Navier–Stokes equations; see [15] or [19] for details.

Space discretization of the generalized Oseen problem (3.3)-(3.5) by finite difference or finite element schemes leads to a large, sparse linear system in *saddle point form*:

$$\begin{bmatrix} A & B^T \\ -B & C \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ 0 \end{bmatrix}. \tag{3.6}$$

Here $A = \sigma M + \nu L + N$ is the discrete counterpart of the operator $\sigma - \nu\Delta + \mathbf{w} \cdot \nabla$. The matrix L is symmetric positive definite (SPD) and consists of a direct sum of discrete Laplace operators. The skew-symmetric matrix N contains the first-order (convective) terms; note that $N = 0$ for the generalized Stokes problem. Also, M is a mass matrix, possibly a scaled identity. The (rectangular) matrix B^T is the discrete gradient and $-B$ the discrete divergence operator. Finally, C is a symmetric positive

semidefinite pressure stabilization matrix; usually $C = 0$ for div-stable discretizations [15]. The (1,1) block A has dimensions $n \times n$ while the (2,2) block C is $m \times m$, with $n > m$. Rather than writing the saddle point system in the equivalent indefinite formulation (with B instead of $-B$ in the (2,1) block and $-C$ instead of C in the (2,2) block), the nonsymmetric positive semidefinite formulation (3.6) is used here in order to easily form the HSS and generalized HSS splittings. The standard HSS method is based on the splittings

$$\begin{bmatrix} \sigma M + \nu L + N & B^T \\ -B & C \end{bmatrix} = \begin{bmatrix} \sigma M + \nu L + \alpha I & 0 \\ 0 & C + \alpha I \end{bmatrix} - \begin{bmatrix} \alpha I - N & -B^T \\ B & \alpha I \end{bmatrix}$$

and

$$\begin{bmatrix} \sigma M + \nu L + N & B^T \\ -B & C \end{bmatrix} = \begin{bmatrix} N + \alpha I & B^T \\ -B & \alpha I \end{bmatrix} - \begin{bmatrix} \alpha I - \sigma M - \nu L & 0 \\ 0 & \alpha I - C \end{bmatrix}.$$

In contrast, the generalized HSS method is based on the splittings

$$\begin{bmatrix} \sigma M + \nu L + N & B^T \\ -B & C \end{bmatrix} = \begin{bmatrix} \nu L + \alpha I & 0 \\ 0 & C + \alpha I \end{bmatrix} - \begin{bmatrix} \alpha I - \sigma M - N & -B^T \\ B & \alpha I \end{bmatrix}$$

and

$$\begin{bmatrix} \sigma M + \nu L + N & B^T \\ -B & C \end{bmatrix} = \begin{bmatrix} \sigma M + N + \alpha I & B^T \\ -B & C + \alpha I \end{bmatrix} - \begin{bmatrix} \alpha I - \nu L & 0 \\ 0 & \alpha I - C \end{bmatrix}.$$

Therefore, the difference between HSS and generalized HSS preconditioning amounts to the structure of the systems to be solved at each application of the preconditioner. HSS requires solving first two decoupled SPD linear systems, one with coefficient matrix $\sigma M + \nu L + \alpha I$ and the other with matrix $C + \alpha I$, followed by a shifted skew-symmetric linear system of the form

$$\begin{bmatrix} N + \alpha I & B^T \\ -B & \alpha I \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{c} \\ d \end{bmatrix}. \tag{3.7}$$

For 3D problems, the first SPD system decouples into four independent linear systems that are strongly diagonally dominant and rather well-conditioned; in an inexact implementation, just one or two iterations of multigrid or preconditioned Conjugate Gradients are usually enough to achieve a sufficiently accurate solution, in the sense that the accuracy is enough to preserve the rate of convergence of the outer iteration.

The second system (3.7) is more complicated. For the generalized Stokes problem ($N = 0$), it can be reduced to the solution of a much smaller ($m \times m$) system involving the Schur complement $\alpha I + \frac{1}{\alpha} BB^T$. This system is also SPD, and is essentially a discrete analogue of a (shifted) Poisson-type equation; it can be solved efficiently by Conjugate Gradients or by multigrid. When $N \neq 0$, however, such reduction is not possible, since the corresponding Schur complement matrix is now dense.

With the generalized HSS method, the first system to be solved also decouples into a set of independent systems with SPD matrices that are well-conditioned and diagonally dominant. The second system can also be reduced, in the case $N = 0$, to a much smaller one involving the Schur complement $\alpha I + B(\alpha I + \sigma M)^{-1} B^T$. If M is a diagonal matrix (such as a scaled identity), this matrix can be explicitly formed. If M is not diagonal, for the purpose of constructing a preconditioner it can be approximated by a diagonal matrix, either by lumping or simply by replacing M with

its diagonal. The resulting system is very easy to solve; in an inexact implementation, one or two steps of an inner preconditioned Conjugate Gradients iteration usually suffice.

As before, when $N \neq 0$ no explicit Schur complement reduction is available, and a coupled system of the form

$$\begin{bmatrix} \sigma M + N + \alpha I & B^T \\ -B & \alpha I \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{c} \\ d \end{bmatrix} \tag{3.8}$$

must be solved at each iteration. Fortunately, the presence of the matrix σM in the (1,1) block tends to make this system much easier to solve than the one arising from the standard HSS method (cf. (3.7)), where the term σM is not present (since it is included in the symmetric part). A few numerical experiments in support of this claim are presented in the following section.

In concluding this section, it is worth mentioning that the (exact) generalized HSS iteration can be shown to be unconditionally convergent for saddle-point problems of the type (3.6), thus generalizing Theorem 3.1 in [8]. The formal statement is as follows.

THEOREM 3.1. *Assume that $A \in \mathbb{R}^{n\times n}$ has positive definite symmetric part $H = (A + A^T)/2$, $B \in \mathbb{R}^{m\times n}$ has full row rank, and that $C = C^T \in \mathbb{R}^{m\times m}$ is positive semidefinite. Assume further that H is split as $H = G + K$ with G symmetric positive definite and K positive semidefinite. Then the generalized HSS iteration converges unconditionally to the unique solution of problem (3.6).*

The proof is a straightforward modification of the one given in [8, Theorem 3.1], and is omitted. See also [17, pages 61–65] for a special case.

4. Numerical experiments. In this section, a few numerical experiments are discussed. Their purpose is to demonstrate the performance of the generalized HSS preconditioner on sparse linear systems arising from the numerical solution of problem (3.3)-(3.5). Some of the results also show that for $\sigma > 0$, the generalized HSS scheme can greatly improve over the standard HSS preconditioner.

The first set of experiments concerns the generalized Stokes problem, i.e., problem (3.3)-(3.5) with $\mathbf{w} = \mathbf{0}$. In Table 4.1 numerical results are reported for Flexible GMRES [20] with an inexact variant of the generalized HSS preconditioner. The discrete saddle point problems were generated in this case by the Marker-and-Cell (MAC) finite difference discretization on a $40 \times 40 \times 40$ grid for different values of σ and ν. This discretization is known to be div-stable, so that $C = 0$ in (3.6). Homogeneous Dirichlet boundary conditions were imposed on the velocities. Here $\Omega = [0, 1] \times [0, 1] \times [0, 1]$; the discrete problem has over 250,000 unknowns. The parameter α was set to 0.5, and a zero initial guess was used. The outer iteration was stopped when a reduction of the initial residual by six orders of magnitude was reached. For the inexact inner solves the Conjugate Gradient algorithm with incomplete Cholesky preconditioning was employed throughout; the incomplete factorization used a drop tolerance $\tau = 0.001$. The inner iterations were stopped as soon as a reduction of the initial residual by one order of magnitude was attained. This only required 1-2 PCG iterations per inner linear solve. The iteration counts, which can be shown to be largely independent of the grid size, improve for increasing σ and decreasing ν.

Table 4.2 shows timings (in seconds) for a generalized Stokes problem with $\nu = 0.001$ for different grids. Here $\sigma = h^{-1}$ where h denotes the grid size. Generalized HSS preconditioning with $\alpha = 0.5$ is used. In the table, the dimensions n and m and the total number of FGMRES iterations are also indicated. The test runs were done

TABLE 4.1
Iteration count for 3D generalized Stokes problem, inexact variant.

σ	$\nu = 0.1$	$\nu = 0.01$	$\nu = 0.001$	$\nu = 10^{-6}$
1	45	27	16	13
10	32	19	15	12
20	30	18	14	11
50	28	15	13	11
100	25	14	12	10

TABLE 4.2
Results for 3D generalized Stokes problem, $\nu = 0.001$, $\sigma = h^{-1}$.

grid	n	m	iterations	cpu time
$10 \times 10 \times 10$	2700	1000	12	0.42
$20 \times 20 \times 20$	22800	8000	12	4.66
$30 \times 30 \times 30$	78300	27000	12	20.97
$40 \times 40 \times 40$	187200	64000	13	66.02

on one processor of a SunFire V880 workstation with 8 CPUs and 16 GB of memory. These results show the good performance of the generalized HSS preconditioner.

On this problem, the standard HSS preconditioner gives similar results. The last experiments show that for more difficult problems, such as the Oseen problem, the generalized HSS approach can drastically outperform the standard HSS preconditioner. The underlying application is the classical leaky-lid driven cavity problem. The matrices were generated using the IFISS Matlab toolbox [14]. The discretization was obtained using quadrilateral Q1-P0 elements on the unit square (uniform grid). Pressure stabilization was used, hence $C \neq 0$. For a 32×32 mesh, the matrix dimensions were $n = 2178$ and $m = 1024$, for a total of 3202 unknowns. The default values $\nu = 0.01$ and $\beta = 0.25$ were used for the viscosity and stabilization parameter, respectively. IFISS scales the resulting matrices so that the symmetric part of the saddle point matrix in (3.6) has norm 1 (using the infinity norm) while the skew-symmetric part has norm 0.25. To simulate a generalized Oseen problem, a matrix of the form σM was added to the (1,1) block, where $\sigma \approx 10h^{-1}$ (h is the mesh size) and M was the mass matrix for the velocity space. The matrix σM has norm 0.1. In the preconditioners used below, M was replaced by a scaled identity.

Table 4.3 contains results for the HSS and generalized HSS (indicated by GHSS in the table) preconditioners. The outer iteration was (full) GMRES, with a convergence tolerance of 10^{-6}. The symmetric and nonsymmetric linear systems arising at each application of the preconditioner were solved by preconditioned Conjugate Gradients (PCG) and by GMRES, respectively, with a rather tight convergence tolerance (also set to 10^{-6}). The inner PCG iteration was preconditioned by an incomplete Cholesky factorization, and the inner GMRES iteration was preconditioned by an incomplete LU factorization; in both cases the drop tolerance was set to 0.01. The table reports the number of outer preconditioned GMRES iterations, the average number of inner PCG and ILU-GMRES iterations, and the total number of inner iterations. It is obvious that the generalized HSS preconditioner is much better, both in terms of rate of convergence and in terms of work per iteration. For these experiments, the value $\alpha = 0.01$ was used in both HSS and GHSS: no effort was spent in finding the optimal value of α for each method.

TABLE 4.3
Results for 2D generalized Oseen problem, $\nu = 0.01$.

Prec.	Outer its.	PCG/outer	GMRES/outer	Total inner
HSS	67	2	12	938
GHSS	24	2	5	182

The results in Table 4.3 correspond to what are essentially exact variants of the two preconditioners. Substantial savings in overall work can be obtained by solving the subproblems arising at each iteration inexactly rather than exactly. Numerical experiments indicate that an inner convergence tolerance of 10^{-2} is generally enough to preserve the rate of convergence of the outer iteration. Of course GMRES must now be replaced by Flexible GMRES. With the inexact variants, the number of outer iterations is essentially unchanged from those reported in Table 4.3, while the average number of inner PCG and ILU-GMRES is reduced to about six (= 2 PCG +4 ILU-GMRES) for the HSS method and to about three (= 2 PCG +1 ILU-GMRES) for GHSS. The total number of inner iterations is about 400 for inexact HSS preconditioning, and about 75 for inexact GHSS preconditioning.

So even in the inexact case, the generalized scheme is much better than the standard one. Moreover, the total cost of the incomplete Cholesky and LU factorizations is much smaller for the generalized scheme. This is shown in Table 4.4, where the number of nonzeros in the incomplete Cholesky and ILU factors for the two methods is shown. The ILU factors of the shifted skew-symmetric matrix arising in the HSS method suffered enormous fill-in with the original ordering; the number reported in the table is for the reverse Cuthill–McKee ordering. It is clear that adding a positive diagonal matrix to the (1,1) block of the shifted skew-symmetric matrix (3.7) has a beneficial effect on the cost of the associated ILU factorization. In contrast, the increase in fill-in in the incomplete Cholesky factor for the generalized HSS scheme is quite modest. Note that the system matrix itself contains 35,266 nonzero entries.

TABLE 4.4
Number of nonzeros in the incomplete factors.

Prec.	Incomplete Cholesky	Incomplete LU
HSS	11,546	179,437
GHSS	14,682	77,808

5. Conclusions. In this paper, a generalization of the Hermitian and skew-Hermitian Splitting method of Bai, Golub, and Ng has been described. The exact version of the new scheme has been shown to be unconditionally convergent. Examples indicate that the generalized scheme can be in some cases asymptotically faster than the classical HSS iteration. In practice, however, the new method is most useful as a preconditioner for a (flexible) Krylov subspace method with inexact inner solves. Besides providing much faster convergence than the standard HSS preconditioner for some important problems, the new scheme has the considerable advantage of replacing the solution of the shifted skew-symmetric system in the HSS method by the solution of an easier (i.e., more diagonally dominant, better conditioned) system. Numerical experiments have been presented showing the effectiveness of the new scheme.

Acknowledgement. I would like to thank Jia Liu for writing the Matlab code used to perform the numerical experiments for the generalized Stokes problem.

REFERENCES

[1] Z. Z. Bai and G. H. Golub, *Accelerated Hermitian and skew-Hermitian splitting iteration methods for saddle-point problems*, IMA J. Numer. Anal., 27 (2007), pp. 1–23.

[2] Z. Z. Bai, G. H. Golub, and C.-K. Li, *Optimal parameter in Hermitian and skew-Hermitian splitting method for certain two-by-two block matrices*, SIAM J. Sci. Comput., 28 (2006), pp. 583–603.

[3] Z. Z. Bai, G. H. Golub, and C.-K. Li, *Convergence properties of preconditioned Hermitian and skew-Hermitian splitting methods for non-Hermitian positive demidefinite matrices*, Math. Comp., 76 (2007), pp. 287–298.

[4] 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.

[5] Z. Z. Bai, G. H. Golub, and J. Y. Pan, *Preconditioned Hermitian and skew-Hermitian splitting methods for non-Hermitian positive semidefinite linear systems*, Numer. Math., 98 (2004), pp. 1–32.

[6] M. Benzi and D. Bertaccini, *Block preconditioning of real-valued iterative algorithms for complex linear systems*, IMA J. Numer. Anal., to appear.

[7] 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.

[8] M. Benzi and G. H. Golub, *A preconditioner for generalized saddle point problems*, SIAM J. Matrix Anal. Appl., 26 (2004), pp. 20–41.

[9] M. Benzi and J. Liu, *An efficient solver for the Navier–Stokes equations in rotation form*, SIAM J. Sci. Comput., 29 (2007), pp. 1959–1981.

[10] M. Benzi and M. K. Ng, *Preconditioned iterative methods for weighted Toeplitz least squares problems*, SIAM J. Matrix Anal. Appl., 27 (2006), pp. 1106–1124.

[11] D. Bertaccini, G. H. Golub, S. Serra-Capizzano, and C. Tablino-Possio, *Preconditioned HSS methods for the solution of non-Hermitian positive definite linear systems and applications to the discrete convection-diffusion equation*, Numer. Math., 99 (2005), pp. 441–484.

[12] L. C. Chan, M. K. Ng, and N. K. Tsing, *Spectral analysis for HSS preconditioners*, Numer. Math. Theor. Meth. Appl., 15 (2006), pp. 1–18.

[13] R. H. Chan, C. Greif, and D. P. O'Leary (Eds.), *Milestones in Matrix Computations. The Selected Works of Gene H. Golub With Commentaries*, Oxford Science Publications, Oxford University Press, Oxford, UK, 2007.

[14] H. C. Elman, A. Ramage, and D. J. Silvester, *Algorithm 866: IFISS, a Matlab toolbox for modelling incompressible flow*, ACM Trans. Math. Software, 33(2), 2007, Article 14.

[15] H. C. Elman, D. J. Silvester and A. J. Wathen, *Finite Elements and Fast Iterative Solvers*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, UK, 2005.

[16] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1991.

[17] J. Liu, *Preconditioned Krylov Subspace Methods for Incompressible Flow Problems*, Ph.D. thesis, Department of Mathematics and Computer Science, Emory University, 2006; available online at http://www.uwf.edu/jliu/research.htm.

[18] G. I. Marchuk, *Methods of Numerical Mathematics*, Springer-Verlag, New York, 1984.

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

[20] Y. Saad, *Iterative Methods for Sparse Linear Systems, Second Edition*, SIAM, Philadelphia, PA, 2003.

[21] V. Simoncini and M. Benzi, *Spectral properties of the Hermitian and skew-Hermitian splitting preconditioner for saddle point problems*, SIAM J. Matrix Anal. Appl., 26 (2004), pp. 377–389.