SIAM J. NUMER. ANAL.
Vol. 53, No. 4, pp. 1716–1737

ON THE NUMERICAL BEHAVIOR OF MATRIX SPLITTING ITERATION METHODS FOR SOLVING LINEAR SYSTEMS*

ZHONG-ZHI BAI† AND MIROSLAV ROZLOŽNÍK‡

Abstract. We study the numerical behavior of stationary one-step or two-step matrix splitting iteration methods for solving large sparse systems of linear equations. We show that inexact solutions of inner linear systems associated with the matrix splittings may considerably influence the accuracy of the approximate solutions computed in finite precision arithmetic. For a general stationary matrix splitting iteration method, we analyze two mathematically equivalent implementations and discuss the conditions when they are componentwise or normwise forward or backward stable. We distinguish two different forms of matrix splitting iteration methods and prove that one of them is significantly more accurate than the other when employing inexact inner solves. The theoretical results are illustrated by numerical experiments with an inexact one-step and an inexact two-step splitting iteration method.

Key words. matrix splitting, stationary iteration method, backward error, rounding error analysis

AMS subject classifications. 65F10, 65F35, 65G30, 65G50, CR: G1.3

DOI. 10.1137/140987936

1. Introduction.

We consider the iterative solution of a large sparse system of linear equations

$$Ax = b, \quad A \in \mathbb{C}^{n,n}, \quad \text{and} \quad b \in \mathbb{C}^n, \tag{1.1}$$

where A is a nonsingular and, in general, a non-Hermitian matrix, and b is the right-hand side vector. Many iterative methods for solving the linear system (1.1) are based on efficient splittings of the coefficient matrix A in the form $A = M - N$, where M is a nonsingular matrix such that a linear system with the coefficient matrix M is easily solvable. The classical examples are the *Jacobi*, the *Gauss–Seidel*, and the *successive overrelaxation* (SOR) iteration methods [26, 14, 13, 15], in which the matrix A is split into its diagonal, off-diagonal, and triangular parts, giving rise to diagonal and lower/upper triangular matrices M, respectively; see [27, 28] and the references therein. Recent examples are the *Hermitian and skew-Hermitian splitting* (HSS) iteration method [7] and its variants such as PMHSS (*preconditioned and modified Hermitian and skew-Hermitian splitting*) [5], in which the matrix A is split into its Hermitian and skew-Hermitian parts, giving rise to shifted Hermitian and shifted skew-Hermitian matrices M; see also [8, 4] and the references therein. In general,

*Received by the editors September 19, 2014; accepted for publication (in revised form) April 27, 2015; published electronically July 14, 2015.
http://www.siam.org/journals/sinum/53-4/98793.html

†State Key Laboratory of Scientific/Engineering Computing, Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, People's Republic of China (bzz@lsec.cc.ac.cn). The research of this author was supported by The National Basic Research Program (2011CB309703), The National Natural Science Foundation (91118001), The National Natural Science Foundation for Creative Research Groups (11321061), and The Defense Industrial Technology Development Program (B1520133016), People's Republic of China.

‡Institute of Computer Science, Czech Academy of Sciences, CZ-182 07 Prague 8, Czech Republic (miro@cs.cas.cz). The research of this author was supported by the Grant Agency of the Czech Republic under project 13-06684S.

the HSS iteration method belongs to the framework of two-step matrix splitting iteration methods [9, 3], which, given two splittings $A = M_1 - N_1$ and $A = M_2 - N_2$ with M_1 and M_2 being nonsingular, iterate alternately between these two splittings in an analogous fashion to the classical *alternating direction implicit* (ADI) iteration method for solving partial differential equations [22, 11].

In some cases, computing the solution of a linear system with the coefficient matrix M (or M_1 or M_2) can be expensive and impractical in actual implementations. To further improve the computational efficiency, we usually solve this linear system, called the inner linear system, by another iteration scheme to some prescribed accuracy, resulting in an inexact or an inner/outer iteration method; see, e.g., [21, 20, 12, 2, 10, 7, 8, 4] and the references therein. The inexact solution of the inner linear system causes two important effects on the numerical behavior of the overall matrix splitting iteration process, i.e., a certain convergence delay of the iteration sequence and a possible accuracy limit on the computed approximate solution. By a componentwise or a normwise backward error analysis [15], we will prescribe the fixed tolerance τ (or the tolerances τ_1 and τ_2) for the inner iteration method, with respect to the splitting matrix M (or the splitting matrices M_1 and M_2), in a one-step (or a two-step) iteration process, which equivalently determines the number of the inner iteration steps. In other words, we interpret each computed approximate solution of an inner linear system as an exact solution of a perturbed linear system, where the relative perturbation of the coefficient matrix of the inner linear system, measured either by the size of its components or by its norm, is bounded by the parameter τ (or the parameters τ_1 and τ_2), being of the ideal order $\tau = \mathcal{O}(u)$ (or τ_1, $\tau_2 = \mathcal{O}(u)$) for a backward stable method, but being much larger than the roundoff unit u in practical implementations. Here, u denotes the unit roundoff and the term $\mathcal{O}(u)$ represents a low-degree polynomial in the problem dimension n multiplied by the unit roundoff u. It is independent of the system parameters but is dependent on details of the computer arithmetic. We emphasize that in this work we do not use variable tolerance $\tau^{(k)}$ (or tolerances $\tau_1^{(k)}$ and $\tau_2^{(k)}$) at the kth iterate as they were considered in [7, 25].

In this paper, we derive bounds on the best accuracy we can obtain from such inexact schemes when implemented in finite precision arithmetic. The fact that the inner solution tolerance strongly influences the accuracy of the computed iterates is known and was studied in several contexts [7, 24, 25, 8, 18, 19]. Stationary iterative methods with the inner linear systems solved to working accuracy have been analyzed in [16, 10]. However, significantly less is known for inexact iteration methods with splitting matrices that are not diagonal or triangular. We estimate the maximum attainable accuracy of inexact two-step splitting iteration methods in terms of τ (or τ_1 and τ_2) and in terms of spectral properties of corresponding splitting matrices. In this sense, we extend the work achieved in [16] and give results similar to [18, 19]. In our work, we analyze two mathematically equivalent implementations and point out the conditions when they are componentwise or normwise forward or backward stable.

The organization of the paper is as follows. In section 2, we derive the main results on the convergence delay and maximum attainable accuracy for stationary (one-step) matrix splitting iteration methods. Section 3 is devoted to the analysis of stationary two-step matrix splitting iteration methods. In section 4, we review the HSS and the PMHSS iteration methods [7, 5], describe two numerical examples, state the computing settings, and report the experimental results. Finally, in section 5, we give conclusions.

Throughout the paper, we adopt the following notation and concepts. For a given vector x and matrix X, $|x|$ and $|X|$ stand for their absolute values. Given a set of n-

dimensional vectors $x_0, x_1, \ldots, x_k$, by $\max_{0\le j\le k} x_j$ we mean the n-dimensional vector with entries equal to maximums in each component throughout this set of vectors. The term I denotes the identity matrix of suitable dimension, and the symbol $\|\cdot\|$ indicates the Euclidean norm of either a vector or a matrix. Given two vectors x and y of the same dimension, we use $x \lessapprox y$ to stand for $x \le \omega y$ with ω being a generic positive constant. When X is a square and nonsingular matrix, we use the quantity $\kappa(X) = \|X\|\|X^{-1}\|$ to represent its Euclidean-norm condition number. For a square matrix X, we denote by $\rho(X)$ its spectral radius. For distinction with their exact arithmetic counterparts, we denote quantities computed in finite precision arithmetic by using an extra upper-hat. In addition, we assume the standard model for floating-point computations, e.g., the IEEE standard 764 model; see [15, Chapter 2]. For simplicity, we do not evaluate the terms proportional to higher powers of u and also occasionally skip the technical details. Given a computed approximate solution $\hat{x}$ to the linear system (1.1), an iteration method is called componentwise forward stable if the error $\hat{x} - x$ satisfies the bound $|\hat{x} - x| \le \mathcal{O}(u)\,|A^{-1}||A||x|$, and it is called normwise forward stable if the Euclidean norm of the error satisfies the bound $\|\hat{x}-x\| \le \mathcal{O}(u)\,\|A^{-1}\|\|A\|\|x\|$. Similarly, an iteration method is called componentwise backward stable if the residual $b - A\hat{x}$ satisfies $|b - A\hat{x}| \le \mathcal{O}(u)\,(|A||\hat{x}| + |b|)$, and it is called normwise backward stable if the Euclidean norm of the residual satisfies $\|b - A\hat{x}\| \le \mathcal{O}(u)\,(\|A\|\|\hat{x}\| + \|b\|)$.

2. Stationary matrix splitting iteration methods. Assume that $A = M-N$ is a splitting of the coefficient matrix A of the linear system (1.1), with M being nonsingular. Starting from an arbitrary initial vector x_0, a stationary (one-step) matrix splitting iteration method for solving the linear system (1.1) produces a sequence of approximate solutions x_{k+1}, $k = 0, 1, 2, \ldots$, with

$$x_{k+1} = M^{-1}(Nx_k + b) \tag{2.1}$$

or

$$x_{k+1} = x_k + M^{-1}(b - Ax_k). \tag{2.2}$$

Note that the iteration schemes (2.1) and (2.2) are mathematically equivalent, but as we will see later they are numerically different in actual implementations. In particular, (2.1) is called a direct-splitting scheme and (2.2) a residual-updating scheme. From (2.1) and (2.2), we see that the error of the approximate solution $x_{k+1} - x$ and the associated residual $b - Ax_{k+1}$ satisfy, respectively, the recurrences

$$x_{k+1} - x = (I - M^{-1}A)(x_k - x) = G(x_k - x), \tag{2.3}$$

$$b - Ax_{k+1} = (I - AM^{-1})(b - Ax_k) = F(b - Ax_k) \tag{2.4}$$

with $G = I - M^{-1}A$ and $F = I - AM^{-1}$. Note that the matrices G and F have the equivalent expressions $G = M^{-1}N$ and $F = NM^{-1}$. We also introduce the notation $H = M^{-1}$.

In practical situations, the inner linear systems, induced by the iteration schemes (2.1) and (2.2), with respect to the coefficient matrix M, cannot be solved exactly. Instead, we will assume that every computed solution of a linear system with the coefficient matrix M and with some right-hand side vector can be interpreted as an exact solution of a linear system with a perturbed coefficient matrix $M + \Delta M$. Note that we will not consider perturbations of the right-hand side here. Indeed, under the

reasonable assumption that $M + \Delta M$ is invertible, its inverse can be written in the form

$$(M + \Delta M)^{-1} = (I + \Delta H)H = H(I + \Delta E),$$
$$\Delta H = -(M + \Delta M)^{-1}\Delta M, \quad \Delta E = -\Delta M(M + \Delta M)^{-1}.$$

If M^{-1} in the iteration matrices G and F appearing in (2.3) and (2.4) is straightforwardly replaced by $(M+\Delta M)^{-1}$, then we could obtain the recurrences with the iteration matrices $G+\Delta G$ and $F+\Delta F$, where $\Delta G = \Delta H(G-I)$ and $\Delta F = (F-I)\Delta E$. Hence, inexact solutions of the inner linear systems with respect to the coefficient matrix M affect the convergence rate of the corresponding overall iteration scheme. Roughly speaking, a potential delay in the convergence is determined by the sizes of ΔH and ΔE. For stationary iteration methods, this phenomenon has been analyzed by several authors; see, e.g., [23, 16, 15, 10].

The accuracy of the approximate solutions computed by two equivalent iteration schemes (2.1) and (2.2) can be estimated by the standard tools of rounding error analysis [15]. The iteration scheme (2.1) has been analyzed by Higham and Knight in [16], where they discussed the recurrence for the computed approximate solutions $\hat{x}_{k+1}$, $k = 0, 1, 2, \ldots$, in the form

$$(M + \Delta M_k)\hat{x}_{k+1} = N\hat{x}_k + b + \Delta s_k, \tag{2.5}$$
$$|\Delta M_k| \le \mathcal{O}(u)\,|M|, \quad |\Delta s_k| \le \mathcal{O}(u)\,(|N||\hat{x}_k| + |b|), \tag{2.6}$$

where Δs_k comes from the computation of the right-hand side vector in (2.1); see also [15, Chapter 17]. The bound on $|\Delta M_k|$ is valid if the matrix M is triangular, which is the case for stationary relaxation iteration methods such as Jacobi, Gauss–Seidel, and SOR [13, 26]. These classical matrix splitting iteration methods can be shown to be forward stable in a componentwise sense and backward stable in a normwise sense. The inner linear systems with respect to the coefficient matrix M are, in general, not easily solvable, so they are solved iteratively in practical implementations. As a result, we cannot expect that all these inner linear systems will be solved in a backward stable way. Instead, we assume that the relative componentwise backward error associated with $\hat{x}_{k+1}$ is bounded by the parameter τ ($\tau \le 1$), i.e., we use the stopping criterion based on the backward error and terminate the inner iteration process once $|\Delta M_k| \le \tau|M|$ is satisfied. As a matter of fact, with the assumption $\tau\cdot\kappa(M) < 1$ we can guarantee the existence and uniform boundedness of the inverse of the matrix $M + \Delta M_k$. Consequently, some accuracy could be achieved in computing the approximate solutions for all inner linear systems, and this is sufficient for showing our results.

2.1. Bounds for the direct-splitting scheme. We analyze the maximum attainable accuracy of the computed approximate solutions caused by the inexact solutions of the inner linear systems in (2.1). More specifically, we are going to show how the level of inexactness given by the tolerance τ affects the maximum attainable accuracy of the computed approximate solution $\hat{x}_{k+1}$ satisfying (2.5), together with

$$|\Delta M_k| \le \tau|M| \quad \text{and} \quad |\Delta s_k| \le \mathcal{O}(u)\,(|N||\hat{x}_k| + |b|). \tag{2.7}$$

THEOREM 2.1. *For the inexact direct-splitting iteration scheme* (2.1) *satisfying the assumptions in* (2.7), *the error and the residual are componentwise bounded as*

$$|\hat{x}_{k+1} - x| \lessapprox \tau \left(\sum_{i=0}^{k} |G^i| \right) |M^{-1}|(|M| + |N|) \left(\max_{0 \le j \le k+1} |\hat{x}_j| + |x| \right), \tag{2.8}$$

$$|b - A\hat{x}_{k+1}| \lessapprox \tau \left(\sum_{i=0}^{k} |F^i| \right) |AM^{-1}| \left((|M| + |N|) \max_{0 \le j \le k+1} |\hat{x}_j| + |b| \right). \tag{2.9}$$

Proof. Given an initial guess $\hat{x}_0$, the computed approximate solution $\hat{x}_{k+1}$ for $k = 0, 1, 2, \ldots$, is thus the exact solution of (2.5), which can be reformulated as

$$\hat{x}_{k+1} = G\hat{x}_k + Hb + \Delta y_k = G^{k+1}\hat{x}_0 + \sum_{i=0}^{k} G^i(Hb + \Delta y_{k-i}), \tag{2.10}$$

where

$$\Delta y_{k-i} = H\left(\Delta s_{k-i} - \Delta M_{k-i}\hat{x}_{k-i+1}\right), \quad i = 0, 1, \ldots, k. \tag{2.11}$$

For the residual vectors corresponding to the solution $\hat{x}_{k+1}$, by making use of the identities

$$AG = AM^{-1}N = NM^{-1}A = FA \quad \text{and} \quad I - AM^{-1} = NM^{-1} = F$$

we can derive the recurrence in the form

$$b - A\hat{x}_{k+1} = F(b - A\hat{x}_k) - A\,\Delta y_k = F^{k+1}(b - A\hat{x}_0) + \sum_{i=0}^{k} F^i A\,\Delta y_{k-i}. \tag{2.12}$$

Using the identities

$$x = Gx + Hb = G^{k+1}x + \sum_{i=0}^{k} G^i Hb,$$

together with (2.10), we then obtain the formula for the error $\hat{x}_{k+1} - x$ of the $(k+1)$th approximate solution $\hat{x}_{k+1}$ computed by the scheme (2.1) as follows:

$$\hat{x}_{k+1} - x = G^{k+1}(\hat{x}_0 - x) + \sum_{i=0}^{k} G^i\,\Delta y_{k-i}.$$

Therefore, the componentwise bound for the error $\hat{x}_{k+1} - x$ is given by

$$|\hat{x}_{k+1} - x| \le |G^{k+1}(\hat{x}_0 - x)| + \left(\sum_{i=0}^{k} \left|G^i\right| \right) \max_{0 \le j \le k} |\Delta y_j|. \tag{2.13}$$

Analogously, using (2.12), we can obtain the componentwise bound for the residual $b - A\hat{x}_{k+1}$ as follows:

$$|b - A\hat{x}_{k+1}| \le |F^{k+1}(b - A\hat{x}_0)| + \left(\sum_{i=0}^{k} |F^i| \right) \max_{0 \le j \le k} |A\,\Delta y_j|. \tag{2.14}$$

If the spectral radius of the iteration matrix G is less than 1, i.e., $\rho(G) < 1$, then the term $|G^{k+1}(\hat{x}_0 - x)|$ converges to the zero vector, and hence for a large k the bound for the maximum attainable accuracy of the computed approximate solution (measured in terms of its error) is given by the supremum of the second term in (2.13). If $\rho(G) < 1$, then, equivalently, $\rho(F) < 1$ and the term $|F^{k+1}(b - A\hat{x}_0)|$ converges to the zero vector, too. As a result, for a large k, the bound for the maximum attainable accuracy of the computed approximate solution (measured in terms of the residual) is given by the supremum of the second term in (2.14). Indeed, then the series $\sum_{i=0}^{\infty} G^i$ and $\sum_{i=0}^{\infty} F^i$ converge and, with $|\Delta M_i| \le \tau |M|$,

$$|\Delta y_i| \le |M^{-1}| \left(|\Delta M_i||\hat{x}_{i+1}| + |\Delta s_i| \right) \qquad \text{and} \quad |A\,\Delta y_i| \le |AM^{-1}| \left(|\Delta M_i||\hat{x}_{i+1}| + |\Delta s_i| \right),$$

corresponding to the recurrence (2.5), we obtain the bounds

(2.15)
$$|\hat{x}_{k+1} - x| \lessapprox \left(\sum_{i=0}^{k} \left| G^i \right| \right) \left(\left[\tau |M^{-1}||M| + \mathcal{O}(u)\,|M^{-1}||N| \right] \max_{0 \le j \le k+1} |\hat{x}_j| + \mathcal{O}(u)\,|M^{-1}||b| \right),$$

(2.16)
$$|b - A\hat{x}_{k+1}| \lessapprox \left(\sum_{i=0}^{k} \left| F^i \right| \right) |I - F| \left(\left[\tau |M| + \mathcal{O}(u)\,|N| \right] \max_{0 \le j \le k+1} |\hat{x}_j| + \mathcal{O}(u)\,|b| \right).$$

Using $\tau \ge \mathcal{O}(u)$ and $|b| \le |A||x| \le (|M| + |N|)|x|$, we can rewrite (2.15) and (2.16) as (2.8) and (2.9), respectively. ∎

Provided that the entries of $\sum_{i=0}^{\infty} |G^i|$ or $\sum_{i=0}^{\infty} |F^i|$ are not too large, in the case of backward stable solutions of all inner linear systems with $\tau = \mathcal{O}(u)$, the estimates in (2.8) and (2.9) guarantee small forward and backward errors in the componentwise sense, respectively. These bounds also contain the factor $\max_{0 \le j \le k+1} |\hat{x}_j|$ that can significantly depend on the convergence behavior of our stationary iteration method. Provided that this factor is not too large, i.e., $\max_{0 \le j \le k+1} |\hat{x}_j| \approx |x|$, the componentwise forward and backward stability is then ensured if $|M^{-1}| \approx |A^{-1}|$ and $|M| + |N| \approx |A|$. However, in practice we have $\tau \gg u$, and therefore the maximum attainable accuracy in general does depend on the parameter τ.

The normwise approach is similar. Replacing the componentwise bounds in (2.6) by the normwise ones,

$$\|\Delta M_k\| \le \tau \|M\| \quad \text{and} \quad \|\Delta s_k\| \le \mathcal{O}(u) \left(\|N\| \|\hat{x}_k\| + \|b\| \right), \tag{2.17}$$

we can obtain normwise bounds on the error and the residual for the direct-splitting iteration scheme (2.1). To this end, similarly to [15], we define the normwise growth factor

$$\theta_{k+1} = \sup_{0 \le i \le k+1} \left\{ \frac{\|\hat{x}_i\|}{\|x\|} \right\}, \tag{2.18}$$

so that $\|\hat{x}_i\| \le \theta_{k+1} \|x\|$, $i = 0, 1, \ldots, k+1$.

THEOREM 2.2. *For the inexact direct-splitting iteration scheme* (2.1) *satisfying the assumptions in* (2.17)*, the error and the residual are normwise bounded as*

$$\|\hat{x}_{k+1} - x\| \lessapprox \frac{1+\theta_{k+1}}{1-\|G\|} \left(\tau\,\kappa(M) + \mathcal{O}(u)\,\|M^{-1}\|\|N\|\right)\|x\|, \tag{2.19}$$

$$\|b - A\hat{x}_{k+1}\| \lessapprox \frac{\|I-F\|}{1-\|F\|}\left[\theta_{k+1}(\tau\|M\| + \mathcal{O}(u)\|N\|)\|x\| + \mathcal{O}(u)\,\|b\|\right]. \tag{2.20}$$

Proof. From (2.11), we can obtain the estimates

$$\begin{aligned}\|\Delta y_i\| &\le \|M^{-1}\|\left(\|\Delta M_i\|\|\hat{x}_{i+1}\| + \|\Delta s_i\|\right)\\ &\le \tau\kappa(M)\|\hat{x}_{i+1}\| + \mathcal{O}(u)\,\|M^{-1}\|\left(\|N\|\|\hat{x}_i\| + \|b\|\right).\end{aligned} \tag{2.21}$$

Analogously to (2.13) and (2.14), we now have the normwise bounds

$$\|\hat{x}_{k+1} - x\| \le \|G^{k+1}(\hat{x}_0 - x)\| + \sum_{i=0}^{k}\|G^i\| \max_{0\le j\le k}\|\Delta y_j\|, \tag{2.22}$$

$$\|b - A\hat{x}_{k+1}\| \le \|F^{k+1}(b - A\hat{x}_0)\| + \sum_{i=0}^{k}\|F^i\| \max_{0\le j\le k}\|A\,\Delta y_j\|. \tag{2.23}$$

Provided that $\|G\| < 1$ and $\|F\| < 1$, it holds that

$$\sum_{i=0}^{k}\|G^i\| \le \frac{1}{1-\|G\|} \quad \text{and} \quad \sum_{i=0}^{k}\|F^i\| \le \frac{1}{1-\|F\|}.$$

By making use of (2.21) and $\|b\| \le (\|M\| + \|N\|)\|x\|$, we have for $i = 0, 1, \ldots, k$ that

$$\|\Delta y_i\| \le \theta_{k+1}\|M^{-1}\|(\tau\|M\| + \mathcal{O}(u)\,\|N\|)\|x\| + \mathcal{O}(u)\,\|M^{-1}\|\|b\|$$

and

$$\begin{aligned}\|\Delta y_i\| &\le \theta_{k+1}(\tau\kappa(M) + \mathcal{O}(u)\,\|M^{-1}\|\|N\|)\|x\| + \mathcal{O}(u)\,\|M^{-1}\|(\|M\| + \|N\|)\|x\|\\ &\le (1+\theta_{k+1})(\tau\kappa(M) + \mathcal{O}(u)\,\|M^{-1}\|\|N\|)\|x\|,\end{aligned}$$

where we have used the fact $\tau \ge \mathcal{O}(u)$. Substituting these estimates into (2.22) and (2.23), we then obtain the bounds

$$\begin{aligned}\|\hat{x}_{k+1} - x\| &\le \|G^{k+1}(\hat{x}_0 - x)\| + \frac{1+\theta_{k+1}}{1-\|G\|}\left(\tau\,\kappa(M) + \mathcal{O}(u)\,\|M^{-1}\|\|N\|\right)\|x\|,\\ \|b - A\hat{x}_{k+1}\| &\le \|F^{k+1}(b - A\hat{x}_0)\|\\ &\quad + \frac{\|I-F\|}{1-\|F\|}\left[\theta_{k+1}(\tau\|M\| + \mathcal{O}(u)\|N\|)\|x\| + \mathcal{O}(u)\,\|b\|\right].\end{aligned}$$

Similarly, if $\|G\| < 1$ and $\|F\| < 1$, then the terms $\|G^{k+1}(\hat{x}_0 - x)\|$ and $\|F^{k+1}(b - A\hat{x}_0)\|$ converge to zero. Hence, for a sufficiently large k, these bounds can be further simplified to the normwise bounds in (2.19) and (2.20), respectively. ∎

In practical situations, when $\tau \gg u$, the relative error of the computed approximate solution will be proportional to the parameter τ. Provided that $\|G\|$ and $\|F\|$

are not too close to 1, and θ_{k+1} is not too large, neglecting the terms with $\mathcal{O}(u)$ in (2.19) and (2.20) we see that the normwise relative error and the normwise residual satisfy

$$\frac{\|\hat{x}_{k+1}-x\|}{\|x\|} \lessapprox \tau\,\frac{1+\theta_{k+1}}{1-\|G\|}\,\kappa(M) \quad \text{and} \quad \|b-A\hat{x}_{k+1}\| \lessapprox \tau\,\frac{\theta_{k+1}\|I-F\|}{1-\|F\|}\|M\|\|x\|,$$

respectively. In the case of backward stable solutions of all inner linear systems with $\tau=\mathcal{O}(u)$, the bounds (2.19) and (2.20) reduce to the bounds (17.11) and (17.19) in [15]. This guarantees a small normwise forward error if $\kappa(M)\approx\kappa(A)$ and a small normwise backward error if $\|M\|\approx\|A\|$ under the abovementioned conditions.

2.2. Bounds for the residual-updating scheme. As also noted in [17], if higher computing accuracy is required, it is better to work with the residual-updating iteration scheme (2.2). This iteration scheme is similar to the iterative refinement in fixed precision, which is a popular technique for improving the accuracy of linear solvers; see [13]. We will show that under mild conditions, this iteration scheme will deliver an approximate solution with the accuracy proportional to the roundoff unit u but independent of the parameter τ. This indicates a significant difference to the direct-splitting iteration scheme (2.1).

Given an initial guess $\hat{x}_0$, at the $(k+1)$th step of the residual-updating iteration scheme (2.2), we first compute the residual of the previously computed approximate solution $\hat{x}_k$ as follows:

$$\hat{r}_k = b-A\hat{x}_k+\Delta r_k \quad \text{with} \quad |\Delta r_k|\le\mathcal{O}(u)\,(|b|+|A||\hat{x}_k|). \tag{2.24}$$

Then we solve approximately the correction equation with the matrix M so that the computed correction vector $\hat{z}_k$ satisfies

$$(M+\Delta M_k)\hat{z}_k=\hat{r}_k \quad \text{with} \quad |\Delta M_k|\le\tau|M|, \tag{2.25}$$

where the stopping criterion in the inner iteration is again based on a backward error smaller than the parameter τ. Finally, we obtain an approximate solution $\hat{x}_{k+1}$ that satisfies

$$\hat{x}_{k+1}=\hat{x}_k+\hat{z}_k+\Delta x_k \quad \text{with} \quad |\Delta x_k|\le u(|\hat{x}_k|+|\hat{z}_k|). \tag{2.26}$$

This scheme is well defined if the matrix $M+\Delta M_k$ is nonsingular, which is guaranteed under relatively mild conditions on the accuracy in the inner iterations (measured by the parameter τ), e.g., if $\sigma_{\min}(M)>\|\Delta M_k\|$, $k=0,1,\ldots$, where $\sigma_{\min}(M)$ is the smallest singular value of the matrix M. We show that for the residual-updating iteration scheme (2.2) the maximum attainable accuracy is proportional to the roundoff unit u.

THEOREM 2.3. *For the inexact residual-updating iteration scheme* (2.2) *satisfying the assumptions in* (2.24)–(2.26), *the error and the residual are componentwise*

bounded as

$$|\hat{x}_{k+1} - x| \lessapprox \left(\sum_{i=0}^{k} |G + \Delta G|^i\right) \Big[\mathcal{O}(u)\ \left(I - \tau|M^{-1}||M|\right)^{-1} |M^{-1}||A| \left(|x| + \max_{0\le j\le k} |\hat{x}_j|\right) + u \max_{0\le j\le k} |\hat{x}_j|\Big], \tag{2.27}$$

$$|b - A\hat{x}_{k+1}| \lessapprox \left(\sum_{i=0}^{k} |F + \Delta F|^i\right) \Big[\mathcal{O}(u)\, |AM^{-1}| \left(I - \tau|M^{-1}||M|\right)^{-1} \left(|b| + |A| \max_{0\le j\le k} |\hat{x}_j|\right) + u|A| \max_{0\le j\le k} |\hat{x}_j|\Big]. \tag{2.28}$$

Proof. By using (2.26), we can derive the following recurrences for the error $\hat{x}_{k+1} - x$ and the residual $b - A\hat{x}_{k+1}$ corresponding to the computed approximate solution $\hat{x}_{k+1}$:

$$\hat{x}_{k+1} - x = \left[I - (M + \Delta M_k)^{-1} A\right] (\hat{x}_k - x) + (M + \Delta M_k)^{-1} \Delta r_k + \Delta x_k, \tag{2.29}$$

$$b - A\hat{x}_{k+1} = \left[I - A(M + \Delta M_k)^{-1}\right] (b - A\hat{x}_k) - A(M + \Delta M_k)^{-1} \Delta r_k - A\, \Delta x_k. \tag{2.30}$$

From the definition of the update $\hat{z}_k$, we have

$$\hat{z}_k = (M + \Delta M_k)^{-1} \left[(b - A\hat{x}_k) + \Delta r_k\right] = (M + \Delta M_k)^{-1} \left[A(x - \hat{x}_k) + \Delta r_k\right].$$

Therefore,

$$\begin{aligned} |\hat{z}_k| &\le |(M + \Delta M_k)^{-1}| \left(|b - A\hat{x}_k| + |\Delta r_k|\right) \\ &\le |(M + \Delta M_k)^{-1}| \left[|b - A\hat{x}_k| + \mathcal{O}(u) \left(|b| + |A||\hat{x}_k|\right)\right], \\ |\hat{z}_k| &\le |(M + \Delta M_k)^{-1}| \left(|A||x - \hat{x}_k| + |\Delta r_k|\right) \\ &\le (1 + \mathcal{O}(u))\, |(M + \Delta M_k)^{-1}||A|(|x| + |\hat{x}_k|). \end{aligned}$$

It follows from these estimates, the bounds (2.24) and (2.26), and the identities (2.29) and (2.30) that

$$\begin{aligned} |\hat{x}_{k+1} - x| &\le |I - (M + \Delta M_k)^{-1} A||\hat{x}_k - x| \\ &\quad + \mathcal{O}(u)\, |(M + \Delta M_k)^{-1}||A|(|x| + |\hat{x}_k|) + u|\hat{x}_k|, \\ |b - A\hat{x}_{k+1}| &\le \left[|I - A(M + \Delta M_k)^{-1}| + u|(M + \Delta M_k)^{-1}||A|\right] |b - A\hat{x}_k| \\ &\quad + \mathcal{O}(u)\, |A(M + \Delta M_k)^{-1}|(|b| + |A||\hat{x}_k|) + u|A||\hat{x}_k|. \end{aligned}$$

If $\rho(\tau|M^{-1}||M|) < 1$, then from $|\Delta M_k| \le \tau|M|$ we have

$$|(M + \Delta M_k)^{-1}| \le \sum_{i=0}^{\infty} \left(\tau|M^{-1}||M|\right)^i |M^{-1}| = \left(I - \tau|M^{-1}||M|\right)^{-1} |M^{-1}|,$$

$$|A(M + \Delta M_k)^{-1}| \le |I - F| \sum_{i=0}^{\infty} \left(\tau|M^{-1}||M|\right)^i = |I - F| \left(I - \tau|M^{-1}||M|\right)^{-1}.$$

Note that for $\tau = \mathcal{O}(u)$, the assumption $\rho(\tau|M^{-1}||M|) < 1$ represents the assumption on numerical nonsingularity of the matrix M. If $\rho(\tau|M^{-1}||M|) \approx 1$, then the fact that the matrix M has the conditioning (in the normwise Bauer–Skeel sense) larger than $1/\tau$ and the perturbation $|\Delta M| \approx \tau|M|$ could imply a singular matrix $M + \Delta M$. This can lead in the worst case to a complete loss of accuracy in a solution of the corresponding inner linear system with the matrix M, meaning essentially that the prescribed tolerance was chosen inappropriately. Moreover, we claim that there exist matrices ΔG and ΔF such that

$$|I - (M + \Delta M_k)^{-1}A| \le |G + \Delta G| \quad \text{and} \quad |I - A(M + \Delta M_k)^{-1}| + u|(M + \Delta M_k)^{-1}||A| \le |F + \Delta F|.$$

Indeed, such matrices ΔG and ΔF do exist and they can be bounded as

$$\begin{aligned}
|\Delta G| &\le \tau \sum_{i=0}^{\infty} (\tau|M^{-1}||M|)^i |M^{-1}||M||M^{-1}A| \\
&= \tau \left(I - \tau|M^{-1}||M|\right)^{-1} |M^{-1}||M||M^{-1}A|, \\
|\Delta F| &\le \left(\tau|AM^{-1}||M| + u|A|\right) |M^{-1}| \sum_{i=0}^{\infty} (\tau|M^{-1}||M|)^i \\
&= \left(\tau|AM^{-1}||M| + u|A|\right) |M^{-1}| \left(I - \tau|M^{-1}||M|\right)^{-1}.
\end{aligned}$$

As a result, we obtain the following bounds for $|\hat{x}_{k+1} - x|$ and $|b - A\hat{x}_{k+1}|$:

(2.31)

$$\begin{aligned}
|\hat{x}_{k+1} - x| &\le |G + \Delta G||\hat{x}_k - x| \\
&\quad + \mathcal{O}(u) \left(I - \tau|M^{-1}||M|\right)^{-1} |M^{-1}||A|(|x| + |\hat{x}_k|) + u|\hat{x}_k| \\
&\le |G + \Delta G|^{k+1}|\hat{x}_0 - x| + \sum_{i=0}^{k} |G + \Delta G|^i \\
&\quad \cdot \left[\mathcal{O}(u) \left(I - \tau|M^{-1}||M|\right)^{-1} |M^{-1}||A| \left(|x| + \max_{0 \le j \le k} |\hat{x}_j|\right) + u \max_{0 \le j \le k} |\hat{x}_j|\right],
\end{aligned}$$

(2.32)

$$\begin{aligned}
|b - A\hat{x}_{k+1}| &\le |F + \Delta F||b - A\hat{x}_k| \\
&\quad + \mathcal{O}(u)\,|I - F| \left(I - \tau|M^{-1}||M|\right)^{-1} (|b| + |A||\hat{x}_k|) + u|A||\hat{x}_k| \\
&\le |F + \Delta F|^{k+1}|b - A\hat{x}_0| + \sum_{i=0}^{k} |F + \Delta F|^i \\
&\quad \cdot \left[\mathcal{O}(u)\,|I - F| \left(I - \tau|M^{-1}||M|\right)^{-1} \left(|b| + |A| \max_{0 \le j \le k} |\hat{x}_j|\right) \right. \\
&\qquad \left. + u|A| \max_{0 \le j \le k} |\hat{x}_j|\right].
\end{aligned}$$

Provided that the spectral radii $\rho(|G + \Delta G|)$ and $\rho(|F + \Delta F|)$ are less than 1, the first terms in (2.31) and (2.32) will be small after a sufficiently large number of iteration steps. Then the error $\hat{x}_{k+1} - x$ and the residual $b - A\hat{x}_{k+1}$ will be proportional to the roundoff unit u as shown in (2.27) and (2.28), respectively. ∎

These bounds are significantly better than the bounds we have obtained for the direct-splitting iteration scheme (2.1). Although in practical situations it is $\tau \gg u$ that is used in the residual-updating iteration scheme (2.2), we will obtain very accurate approximate solutions after sufficiently many iteration steps.

For the normwise approach, the componentwise bounds in (2.24), (2.25), and (2.26) are, respectively, replaced by the normwise ones

$$\hat{r}_k = b - A\hat{x}_k + \Delta r_k \quad \text{with} \quad \|\Delta r_k\| \le \mathcal{O}(u)\,(\|b\| + \|A\|\|\hat{x}_k\|), \tag{2.33}$$

$$(M + \Delta M_k)\hat{z}_k = \hat{r}_k \quad \text{with} \quad \|\Delta M_k\| \le \tau\|M\|, \tag{2.34}$$

$$\hat{x}_{k+1} = \hat{x}_k + \hat{z}_k + \Delta x_k \quad \text{with} \quad \|\Delta x_k\| \le u(\|\hat{x}_k\| + \|\hat{z}_k\|). \tag{2.35}$$

Based on the identities (2.29) and (2.30), using an analogous approach we can derive normwise bounds on the error $\hat{x}_{k+1} - x$ and the residual $b - A\hat{x}_{k+1}$ for the residual-updating iteration scheme (2.2).

THEOREM 2.4. *For the inexact residual-updating iteration scheme* (2.2) *satisfying the assumptions in* (2.33)–(2.35), *the error and the residual are normwise bounded as*

$$\frac{\|\hat{x}_{k+1} - x\|}{\|x\|} \lessapprox \mathcal{O}(u)\frac{(1+\theta_k)\left(\|M^{-1}\|\|A\| + 1 - \tau\,\kappa(M)\right)}{1 - \|G\| - \tau\,\kappa(M)\,(1 - \|G\| + \|I - G\|)},$$

$$\|b - A\hat{x}_{k+1}\| \lessapprox \mathcal{O}(u)\frac{\|I - F\|}{1 - \|F\| - \tau\,\kappa(M)\,(1 - \|F\| + \|I - F\|) - u\|M^{-1}\|\|A\|} \cdot \left[\|b\| + \theta_k\|A\|\|x\|\left(1 + \frac{1 - \tau\,\kappa(M)}{\|I - F\|}\right)\right]$$

under the assumptions

$$\|G + \Delta G\| \le \|G\| + \tau\,\kappa(M)\,(1 - \tau\,\kappa(M))^{-1}\|I - G\| < 1, \tag{2.36}$$

$$\|F + \Delta F\| \le \|F\| + (1 - \tau\,\kappa(M))^{-1}\left(\tau\,\kappa(M)\,\|I - F\| + u\|M^{-1}\|\|A\|\right) < 1, \tag{2.37}$$

where θ_k *is the normwise growth factor defined in* (2.18).

Recall that the normwise growth factor θ_k depends on all preceding computed iterates $\{\hat{x}_i\}_{i=0}^{k}$. Again, these bounds guarantee small normwise forward and backward errors, respectively, under mild conditions as stated in (2.36) and (2.37).

In summary, if the iteration schemes (2.1) and (2.2) are either componentwise or normwise forward or backward stable, and if the splitting matrix N is as sparse and structured as the coefficient matrix A, then, at the kth iteration step of these two schemes, computing the vector $Nx_k + b$ should be as costly as computing the residual $b - Ax_k$. So the residual-updating iteration scheme (2.2) costs about the same workloads as the direct-splitting iteration scheme (2.1) at each iteration step. Roughly speaking, provided that the inner linear systems having the same coefficient matrix M are solved inexactly with accuracies controlled by the same tolerance τ, the residual-updating iteration scheme (2.2) can always achieve higher computational efficiency than the direct-splitting iteration scheme (2.1).

3. Stationary two-step matrix splitting iteration methods. In this section, we study the numerical behavior of the stationary two-step matrix splitting iteration methods [22, 11, 3, 9] and give results similar to the ones obtained for the stationary matrix splitting iteration methods in section 2.

The stationary two-step matrix splitting iteration framework has been studied extensively by several authors from several perspectives; see, e.g., [7, 6, 4] and the

references therein. We consider two splittings of the matrix A in the form $A = M_1 - N_1$ and $A = M_2 - N_2$. Given an initial vector x_0, we define the stationary two-step matrix splitting iteration method by the following two successive recurrences:

$$M_1 x_{k+1/2} = N_1 x_k + b, \qquad M_2 x_{k+1} = N_2 x_{k+1/2} + b. \tag{3.1}$$

Denote by $G_1 = M_1^{-1} N_1 = I - H_1 A$ and $G_2 = M_2^{-1} N_2 = I - H_2 A$ with $H_1 = M_1^{-1}$ and $H_2 = M_2^{-1}$. Then (3.1) can be rewritten as

$$x_{k+1/2} = G_1 x_k + H_1 b, \quad x_{k+1} = G_2 x_{k+1/2} + H_2 b.$$

These give rise to the alternative recurrence

$$x_{k+1/2} = x_k + H_1(b - A x_k), \quad x_{k+1} = x_{k+1/2} + H_2(b - A x_{k+1/2}). \tag{3.2}$$

At each iteration step, the recurrence (3.2) involves the computations of two residuals $b - Ax_k$ and $b - Ax_{k+1/2}$, which require two matrix-vector multiplications with respect to the matrix A. According to Lemma 2.1 in [7], this can be avoided by the substitution of $x_{k+1/2}$ into x_{k+1}, leading to

$$\begin{aligned} x_{k+1} &= x_k + H_1(b - A x_k) + H_2\left[b - A\left(x_k + H_1(b - A x_k)\right)\right] \\ &= (I - H_2 A)(I - H_1 A) x_k + [(I - H_2 A) H_1 + H_2] b \\ &= G x_k + H b, \end{aligned}$$

where

$$G = G_2 G_1 \quad \text{and} \quad H = G_2 H_1 + H_2. \tag{3.3}$$

We remark that the matrix H admits the equivalent expressions

$$H = H_1 + H_2 G_1 = H_1 + H_2 - H_2 A H_1 = H_2(M_1 + M_2 - A) H_1, \tag{3.4}$$

and the matrices G and H satisfy the identity $G = I - HA$. Thus, instead of (3.2), we can use only one single recurrence

$$x_{k+1} = x_k + H(b - A x_k). \tag{3.5}$$

The detailed convergence analysis about the alternating splitting iteration method can be found in [9, 3] and the references therein.

In practical situations, the inner linear systems, induced by the iteration scheme (3.1) or (3.2) with respect to the coefficient matrices M_1 and M_2, cannot be solved exactly, and they are often solved inexactly by some other iteration schemes; see [7, 8] and the references therein. It follows that inexact solutions of the inner linear systems with respect to the coefficient matrices M_1 and M_2 affect the convergence rate of the corresponding overall iteration scheme.

3.1. Bounds for the direct-splitting scheme. We estimate the maximum attainable accuracy for the approximate solution computed with the direct-splitting iteration scheme (3.1). Using the same approach as for the stationary matrix splitting iteration method defined by (2.1) in section 2, we can write

$$\hat{x}_{k+1/2} = M_1^{-1}(N_1 \hat{x}_k + b + \Delta s_{k+1/2}), \tag{3.6}$$

$$\hat{x}_{k+1} = M_2^{-1}(N_2 \hat{x}_{k+1/2} + b + \Delta s_{k+1}), \tag{3.7}$$

where

$$|\Delta s_{k+1/2}| \le \tau_1 |M_1||\hat{x}_{k+1/2}| + \mathcal{O}(u)\,(|N_1||\hat{x}_k| + |b|), \tag{3.8}$$

$$|\Delta s_{k+1}| \le \tau_2 |M_2||\hat{x}_{k+1}| + \mathcal{O}(u)\,(|N_2||\hat{x}_{k+1/2}| + |b|). \tag{3.9}$$

The quantities τ_1 and τ_2 are the tolerances employed to describe the accuracies in solving the inner linear systems with respect to the matrices M_1 and M_2, respectively. For convenience of the statements, in what follows we will use the notation $F_1 = N_1 M_1^{-1}$ and $F_2 = N_2 M_2^{-1}$.

THEOREM 3.1. *For the inexact direct-splitting iteration scheme* (3.1) *satisfying the assumptions in* (3.6)–(3.9), *the error and the residual are componentwise bounded as*

(3.10)

$$\begin{aligned} &|\hat{x}_{k+1} - x| \\ &\quad \lessapprox \left(\sum_{i=0}^{k} |G^i|\right) \left[|G_2 M_1^{-1}|(\tau_1|M_1| + \mathcal{O}(u)\,|N_1|) + |M_2^{-1}|(\tau_2|M_2| + \mathcal{O}(u)\,|N_2|)\right] \\ &\qquad \cdot \left(\max_{1\le j\le 2k+3} |\hat{x}_{(j-1)/2}| + |x|\right), \end{aligned}$$

(3.11)

$$\begin{aligned} &|b - A\hat{x}_{k+1}| \\ &\quad \lessapprox \left(\sum_{i=0}^{k} |F^i|\right) \left[|F_2(I - F_1)| \left((\tau_1|M_1| + \mathcal{O}(u)\,|N_1|) \max_{1\le j\le 2k+3} |\hat{x}_{(j-1)/2}| + |b|\right)\right. \\ &\qquad\qquad \left. + |I - F_2| \left((\tau_2|M_2| + \mathcal{O}(u)\,|N_2|) \max_{1\le j\le 2k+3} |\hat{x}_{(j-1)/2}| + |b|\right)\right]. \end{aligned}$$

Proof. Substituting $\hat{x}_{k+1/2}$ into the formula for $\hat{x}_{k+1}$, we obtain the expression

$$\hat{x}_{k+1} = G\hat{x}_k + Hb + \Delta y_k \quad \text{with} \quad \Delta y_k = G_2 H_1\,\Delta s_{k+1/2} + H_2\,\Delta s_{k+1}.$$

Then it follows from direct manipulation that

$$A\,\Delta y_k = F_2(I - F_1)\,\Delta s_{k+1/2} + (I - F_2)\,\Delta s_{k+1},$$

where we have used the commutative property of the matrices A and $M_2^{-1}N_2$, i.e., $AM_2^{-1}N_2 = M_2^{-1}N_2 A$. Analogously to the proof of Theorem 2.1, we can derive the bounds in (3.10) and (3.11). ∎

Provided that the entries of the vector $\max_{1\le j\le 2k+3} |\hat{x}_{(j-1)/2}|$ and the entries in the matrix $\sum_{i=0}^{k} |G^i|$ or $\sum_{i=0}^{k} |F^i|$ are not too large, in the case of backward stable solutions of all inner linear systems with $\tau_1 = \mathcal{O}(u)$ and $\tau_2 = \mathcal{O}(u)$, these estimates then guarantee small forward and backward errors in the componentwise sense, respectively, if $|M_1^{-1}| \approx |M_2^{-1}| \approx |A^{-1}|$ and $|M_1| + |N_1| \approx |M_2| + |N_2| \approx |A|$. However, in practice we have $\tau_1 \gg u$ and $\tau_2 \gg u$. Therefore, the maximum attainable accuracy in general does depend on the parameters τ_1 and τ_2.

The normwise approach can be conducted in a similar fashion. Replacing the componentwise bounds by the normwise ones and introducing the normwise growth factor $\theta_{k+1} = \sup_{1\le i\le 2k+3}\{\|\hat{x}_{(i-1)/2}\|\}/\|x\|$, we can formulate the following theorem.

THEOREM 3.2. *For the inexact direct-splitting iteration scheme* (3.1), *the error and the residual are normwise bounded as*

$$\frac{\|\hat{x}_{k+1}-x\|}{\|x\|} \lessapprox \frac{1+\theta_{k+1}}{1-\|G\|}(\tau_1\|G_2\|\,\kappa(M_1)+\tau_2\,\kappa(M_2)),$$

$$\|b-A\hat{x}_{k+1}\| \lessapprox \frac{\theta_{k+1}}{1-\|F\|}(\tau_1\|F_2(I-F_1)\|\|M_1\|+\tau_2\|I-F_2\|\|M_2\|)\|x\| + \frac{\|b\|}{1-\|F\|}(\|F_2(I-F_1)\|+\|I-F_2\|).$$

Roughly speaking, the limiting accuracy level measured in terms of the error is given by the quantity $\tau_1\|G_2\|\,\kappa(M_1)+\tau_2\,\kappa(M_2)$, so the τ_1-term is damped by the quantity $\|G_2\|$. In actual implementations, we should balance the choices of the tolerances τ_1 and τ_2 in such a way that a desired overall accuracy of the error is achieved. For example, we may set $\tau_1=\frac{\tau}{\|G_2\|}$ and $\tau_2=\tau$, where τ is a prescribed tolerance. Consequently, it holds that

$$\frac{\|\hat{x}_{k+1}-x\|}{\|x\|} \lessapprox \frac{1+\theta_{k+1}}{1-\|G\|}\,\tau(\kappa(M_1)+\kappa(M_2)).$$

The quantity $\tau_1\|F_2(I-F_1)\|\|M_1\|+\tau_2\|I-F_2\|\|M_2\|$ plays an analogous role in the result for the norm of the residual, and the tolerances τ_1 and τ_2 can be chosen in a similar fashion to the above, e.g., through a prescribed common tolerance $\tau_1=\tau_2=\tau$. Definitely, the maximum attainable accuracies depend on the levels of inexactness (measured in terms of τ) in solving the inner linear systems either with the matrix M_1 or with the matrix M_2.

3.2. Bounds for the residual-updating scheme. In some applications, one needs the maximum attainable accuracy proportional to the machine precision u. Hence, it makes sense that we discuss the residual-updating iteration scheme (3.2) or, equivalently, the recurrence (3.5) by applying the theory for the recurrence (2.2) established in section 2. In this manner, we can derive the componentwise and the normwise bounds for the error $\hat{x}_{k+1}-x$ and the residual $b-A\hat{x}_{k+1}$.

To this end, we recall that the matrix H, defined in (3.3) and reformulated in (3.4), adopts the equivalent expression $H=M_2^{-1}(M_1+M_2-A)M_1^{-1}$. Similar to the computing assumptions described in (2.24), (2.25), and (2.26), at the $(k+1)$th step of the iteration scheme (3.5) we assume that the computed solution $\hat{x}_{k+1}$ is obtained by

$$\hat{r}_k=b-A\hat{x}_k+\Delta r_k,\quad \hat{z}_k=(H+\Delta H^{(k)})\hat{r}_k,\quad \text{and}\quad \hat{x}_{k+1}=\hat{x}_k+\hat{z}_k+\Delta x_k \tag{3.12}$$

with a given initial guess $\hat{x}_0$, where

$$|\Delta r_k|\le\mathcal{O}(u)\,(|b|+|A||\hat{x}_k|),\quad |\Delta x_k|\le u(|\hat{x}_k|+|\hat{z}_k|), \tag{3.13}$$

and $\Delta H^{(k)}$ is a perturbation to the matrix H, which is defined implicitly by

$$H+\Delta H^{(k)}=(M_2+\Delta M_2^{(k)})^{-1}(M_1+M_2-A+\Delta M^{(k)})(M_1+\Delta M_1^{(k)})^{-1} \tag{3.14}$$

with $\Delta M_1^{(k)}$, $\Delta M_2^{(k)}$, and $\Delta M^{(k)}$ being imposed to satisfy

$$|\Delta M_1^{(k)}|\le\tau_1|M_1|,\quad |\Delta M_2^{(k)}|\le\tau_2|M_2| \quad\text{and}\quad |\Delta M^{(k)}|\le\mathcal{O}(u)\,(|M_1|+|M_2|+|A|). \tag{3.15}$$

The quantities τ_1 and τ_2 are two prescribed tolerances used to measure the accu-

racies in solving the inner linear systems with respect to the matrices M_1 and M_2, respectively. We can formulate the following theorem.

THEOREM 3.3. *For the inexact residual-updating iteration scheme* (3.5) *satisfying the assumptions in* (3.12)–(3.15), *the error and the residual are componentwise bounded as*

$$|\hat{x}_{k+1} - x| \lessapprox \left(\sum_{i=0}^{k} |G + \Delta G|^i\right) \left[\mathcal{O}(u)\,|I - G - \Delta G||A^{-1}||A| \left(|x| + \max_{0\le j\le k} |\hat{x}_j|\right) + u \max_{0\le j\le k} |\hat{x}_j|\right],$$

$$|b - A\hat{x}_{k+1}| \lessapprox \left(\sum_{i=0}^{k} |F + \Delta F|^i\right) \left[\mathcal{O}(u)\,|I - F - \Delta F| \left(|b| + |A| \max_{0\le j\le k} |\hat{x}_j|\right) + u|A| \max_{0\le j\le k} |\hat{x}_j|\right].$$

These bounds are significantly better than the bounds we have obtained for the direct-splitting iteration scheme (3.1). Although in practical situations it is $\tau_1 \gg u$ and $\tau_2 \gg u$ that are used in the residual-updating iteration scheme (3.5), we will obtain very accurate approximate solutions after sufficiently many iteration steps.

For the normwise approach, analogously the componentwise bounds (3.13) and (3.15) are replaced by the normwise ones. Provided that $\|G + \Delta G\| < 1$ and $\|F + \Delta F\| < 1$ and assuming that the normwise growth factor $\theta_k = \sup_{0\le i\le k}\{\|\hat{x}_i\|\}/\|x\|$ is not too large, we can give the following normwise bounds.

THEOREM 3.4. *For the inexact residual-updating iteration scheme* (3.5), *the error and the residual are normwise bounded as*

$$\frac{\|\hat{x}_{k+1} - x\|}{\|x\|} \lessapprox \mathcal{O}(u)\,\frac{1+\theta_k}{1 - \|G + \Delta G\|}\left(\|I - (G + \Delta G)\|\,\kappa(A) + \frac{\theta_k}{1+\theta_k}\right),$$

$$\|b - A\hat{x}_{k+1}\| \lessapprox \mathcal{O}(u)\,\frac{\|I - (F + \Delta F)\|}{1 - \|F + \Delta F\|}\left[\|b\| + \left(1 + \frac{1}{\|I - (F + \Delta F)\|}\right)\theta_k \|A\|\|x\|\right].$$

These bounds guarantee small normwise forward and backward errors, respectively, under mild conditions on the coefficient matrix A as well as the splitting matrices M_1, N_1 and M_2, N_2.

In summary, if the iteration schemes (3.1) and (3.2) are either componentwise or normwise forward or backward stable and if the splitting matrices N_1 and N_2 are as sparse and structured as the coefficient matrix A, then at the kth iteration step of these two schemes computing the vectors $N_1 x_k + b$ and $N_2 x_{k+1/2} + b$ should be as costly as computing the residuals $b - Ax_k$ and $b - Ax_{k+1/2}$, respectively. So the direct-splitting iteration scheme (3.1) costs about the same workloads as the residual-updating iteration scheme (3.2) at each iteration step. Roughly speaking, provided that the inner linear systems having the same coefficient matrices M_1 and M_2 are solved inexactly in accuracies controlled by the same tolerances τ_1 and τ_2, respectively, the residual-updating iteration scheme (3.2) will always achieve higher accuracy than the direct-splitting iteration scheme (3.1).

4. Numerical experiments. In this section, we recall the implementations of the PMHSS and the HSS iteration methods [7, 5]. We remark that PMHSS and HSS are the typical examples of the stationary one-step and two-step matrix splitting iteration methods for solving the large sparse linear system (1.1), respectively; see

also [8, 4] and the references therein. We also describe two numerical examples with complex symmetric and nonsymmetric positive-definite linear systems. Finally, we report and analyze the experimental results.

4.1. Solution methods. The PMHSS iteration method is used to solve the linear system (1.1), with its coefficient matrix $A \in \mathbb{C}^{n,n}$ being complex symmetric and given by $A = W + \mathrm{i}T$, where $W, T \in \mathbb{R}^{n,n}$ are real, symmetric, and positive semidefinite matrices with at least one of them, say, W, being positive definite. Here and in what follows, we use $\mathrm{i} = \sqrt{-1}$ to denote the imaginary unit. A specific form of this iteration method is given by setting the iteration parameter α to be 1 and choosing the preconditioning matrix to be W, which has the following algorithmic description.

METHOD 4.1 (the PMHSS iteration method [5]). *Let $x_0 \in \mathbb{C}^n$ be an arbitrary initial guess. For $k = 0, 1, 2, \ldots$ until the sequence of iterates $\{x_k\}_{k=0}^{\infty} \subset \mathbb{C}^n$ converges, compute the next iterate x_{k+1} according to either the direct-splitting scheme*

$$(W + T)x_{k+1} = \frac{1+\mathrm{i}}{2}(W - \mathrm{i}\,T)x_k + \frac{1-\mathrm{i}}{2}b \tag{4.1}$$

or the residual-updating scheme

$$x_{k+1} = x_k + \frac{1-\mathrm{i}}{2}(W + T)^{-1}(b - Ax_k). \tag{4.2}$$

In fact, the PMHSS iteration schemes (4.1) and (4.2) are stationary one-step matrix splitting iteration methods induced by the matrix splitting $A = M - N$ with $M = (1 + \mathrm{i})(W + T)$ and $N = \mathrm{i}(W - \mathrm{i}\,T)$. They converge unconditionally to the unique solution of the complex symmetric linear system (1.1) for any initial guess if $\mathrm{null}(W) \cap \mathrm{null}(T) = \{0\}$, where $\mathrm{null}(\cdot)$ denotes the null space of the corresponding matrix. For distinction, we call (4.1) and (4.2), respectively, the PMHSS iteration schemes I and II, or for short PMHSS-I and PMHSS-II, in the subsequent discussion. In actual computations, we solve the linear subsystems with respect to the coefficient matrix $W + T$ iteratively by the *preconditioned conjugate gradient* (PCG) method with the incomplete Cholesky factorization [13] preconditioner (MATLAB code `ichol(sparse(·))`).

The HSS iteration method is used to solve the linear system (1.1) with its coefficient matrix $A \in \mathbb{C}^{n,n}$ being non-Hermitian and positive definite, i.e., its Hermitian part $\mathcal{H}(A) = \frac{1}{2}(A + A^*)$ is positive definite, where A^* is the conjugate transpose of the matrix A; see [7]. Denote by $\mathcal{S}(A) = \frac{1}{2}(A - A^*)$ the skew-Hermitian part of the matrix A. Then it holds that $A = \mathcal{H}(A) + \mathcal{S}(A)$, and the HSS iteration method can be algorithmically described as follows.

METHOD 4.2 (the HSS iteration method [7]). *Let $x_0 \in \mathbb{C}^n$ be an arbitrary initial guess. For $k = 0, 1, 2, \ldots$ until the sequence of iterates $\{x_k\}_{k=0}^{\infty} \subset \mathbb{C}^n$ converges, compute the next iterate x_{k+1} according to either the direct-splitting scheme*

$$\begin{cases} (\alpha I + \mathcal{H}(A))x_{k+1/2} = (\alpha I - \mathcal{S}(A))x_k + b, \\ (\alpha I + \mathcal{S}(A))x_{k+1} = (\alpha I - \mathcal{H}(A))x_{k+1/2} + b \end{cases} \tag{4.3}$$

or the residual-updating scheme

$$\begin{cases} x_{k+1/2} = x_k + (\alpha I + \mathcal{H}(A))^{-1}(b - Ax_k), \\ x_{k+1} = x_{k+1/2} + (\alpha I + \mathcal{S}(A))^{-1}(b - Ax_{k+1/2}), \end{cases} \tag{4.4}$$

where α is a given positive constant.

In fact, the HSS iteration schemes (4.3) and (4.4) are stationary two-step matrix splitting iteration methods induced by the matrix splittings

$$M_1 = \alpha I + \mathcal{H}(A),\ N_1 = \alpha I - \mathcal{S}(A) \quad \text{and} \quad M_2 = \alpha I + \mathcal{S}(A),\ N_2 = \alpha I - \mathcal{H}(A).$$

Alternatively, they can be also induced by the matrix splitting $A = M(\alpha) - N(\alpha)$ with

$$M(\alpha) = \frac{1}{2\alpha}(\alpha I + \mathcal{H}(A))(\alpha I + \mathcal{S}(A)) \quad \text{and} \quad N(\alpha) = \frac{1}{2\alpha}(\alpha I - \mathcal{H}(A))(\alpha I - \mathcal{S}(A)).$$

These two schemes converge unconditionally to the unique solution of the non-Hermitian positive definite linear system (1.1) for any initial guess. For distinction, we call (4.3) and (4.4), respectively, the HSS iteration schemes I and II, or for short HSS-I and HSS-II, in the subsequent discussion. In actual computations, the iteration parameter α is chosen to be the experimentally optimal one that minimizes the number of iteration steps of the HSS iteration method. We solve the linear subsystems with respect to the coefficient matrices $\alpha I + \mathcal{H}(A)$ and $\alpha I + \mathcal{S}(A)$ iteratively by the PCG and the *preconditioned conjugate gradient for normal equation* (PCGNE) methods with the incomplete Cholesky (MATLAB code `ichol(sparse(·))`) or the incomplete LU (MATLAB code `ilu(sparse(·))`) factorization preconditioners [13].

4.2. Numerical examples. In the following, we describe two numerical examples used in our experiments.

Example 4.1. The linear system (1.1) is of the form

$$\left[\left(K + \frac{3-\sqrt{3}}{\eta} I\right) + \mathrm{i}\left(K + \frac{3+\sqrt{3}}{\eta} I\right)\right] x = b,$$

where η is the time step-size and K is the five-point centered difference matrix approximating the negative Laplacian operator $L = -(u_{xx} + u_{yy} + u_{zz})$ with homogeneous Dirichlet boundary conditions on a uniform mesh in the unit cube $\Omega = [0,1] \times [0,1] \times [0,1]$ with the mesh-size $h = \frac{1}{m+1}$. The matrix $K \in \mathbb{R}^{n,n}$ possesses the tensor-product form $K = B_m \otimes I \otimes I + I \otimes B_m \otimes I + I \otimes I \otimes B_m$ with $B_m = h^{-2} \cdot \mathrm{tridiag}(-1, 2, -1) \in \mathbb{R}^{m,m}$. Hence, K is an $n \times n$ block-pentadiagonal matrix with $n = m^3$. We take $W = K + \frac{3-\sqrt{3}}{\eta} I$ and $T = K + \frac{3+\sqrt{3}}{\eta} I$, and the right-hand side vector b with its jth entry $[b]_j$ being given by $[b]_j = \frac{(1-\mathrm{i})j}{\eta(j+1)^2}$, $j = 1, 2, \ldots, n$. Furthermore, we normalize the coefficient matrix and right-hand side by multiplying both by h^2. See [1, 4].

In our tests, we take $\eta = h$. For more details about the practical background of this class of problems, we refer to [1, 4] and the references therein.

Example 4.2. Consider the linear system (1.1), for which $A \in \mathbb{R}^{n,n}$ is the upwind difference matrix of the three-dimensional convection-diffusion equation

$$-(u_{xx} + u_{yy} + u_{zz}) + \frac{q \cdot \exp(x+y+z)}{x+y+z}(x u_x + y u_y + z u_z) = f(x, y, z)$$

on the unit cube $\Omega = [0,1] \times [0,1] \times [0,1]$ with the homogeneous Dirichlet boundary conditions. The step-sizes along all x, y, and z directions are the same, i.e., $h = \frac{1}{m+1}$, and the right-hand side vector b is taken to be $b = Ae$ with $e \in \mathbb{R}^n$ being the vector of all entries equal to 1. We denote by $\mathrm{Re} = qh$ the mesh Reynolds number. See [7, 6].

TABLE 1
Numerical results of PMHSS iteration schemes for Example 4.1 at IT = 50.

m	Method	Index	τ				
			10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}
32	PMHSS-I	CPU	2.69	1.73	3.02	4.69	6.94
		BERR	1.06E-04	1.72E-06	1.34E-08	1.17E-10	1.49E-12
	PMHSS-II	CPU	10.35	10.88	11.57	12.76	15.92
		BERR	5.47E-16	5.45E-16	5.48E-16	5.45E-16	5.47E-16
64	PMHSS-I	CPU	11.01	24.38	48.67	72.25	106.50
		BERR	1.08E-04	1.83E-06	1.41E-08	1.32E-10	1.08E-12
	PMHSS-II	CPU	205.12	214.01	236.21	256.48	286.18
		BERR	5.77E-16	5.64E-16	5.62E-16	5.63E-16	5.61E-16

All outer iteration processes are started from zero and terminated once the Euclidean norms of the current residuals are reduced by a factor of 10^8 from those of the initial residuals. In addition, all codes are run in MATLAB (version R2013a) in double precision, and all experiments are performed on a personal computer with a 2.66GHz central processing unit (Intel(R) Core(TM)2 Duo CPU E6750), 2 GB of memory, and the Windows operating system.

4.3. Experimental results. By implementing two equivalent schemes of the PMHSS iteration method used to solve Example 4.1 and those of the HSS iteration method used to solve Example 4.2, we show that the residual-updating schemes, i.e., PMHSS-II and HSS-II, are always significantly more accurate than the direct-splitting schemes, i.e., PMHSS-I and HSS-I, for large spectrums of the stopping tolerance(s) τ (or τ_1 and τ_2) of the inner iteration method(s). To this end, we report numerical results with respect to the number of iteration steps (IT), the computing time in seconds (CPU), and the normwise backward error (BERR) for these iteration schemes. Here, BERR is defined as $\text{BERR} = \frac{\|b-Ax_k\|}{\|b\|+\|A\|\|x_k\|}$ with k being the iterate index.

At IT = 50, in Table 1 we list CPU and BERR for PMHSS-I and PMHSS-II when they are used to solve Example 4.1 with respect to different problem sizes and various stopping tolerances. We observe that for each fixed m, the CPU time for each scheme increases significantly when the tolerance τ becomes smaller; and for fixed m and τ, PMHSS-I always costs much less CPU than PMHSS-II. The price paid for PMHSS-II to achieve a higher accuracy than PMHSS-I is more CPU time. In fact, the inner iteration of PMHSS-II starts from zero initial guess, so at each outer iteration step PMHSS-II requires more inner iteration steps than PMHSS-I to satisfy its stopping criterion. In Figure 1, we depict the curves of BERR versus IT when $m = 64$ with respect to various stopping tolerances for PMHSS-I and PMHSS-II when they are used to solve Example 4.1. From Table 1 and Figure 1, we observe that for fixed m, the normwise backward error of PMHSS-I is of the same order of magnitude as τ, but that of PMHSS-II is always of the order $\mathcal{O}(u)$ no matter whether τ is large or small. Hence, in actual computations, PMHSS-II is always backward stable independent of the tolerance τ, but PMHSS-I may be backward stable only for those τ of about the order $\mathcal{O}(u)$ of magnitude.

At IT = 250, in Table 2 we list CPU and BERR for HSS-I and HSS-II when they are used to solve Example 4.2 with respect to $m = 64$, Re = 10, and various stopping tolerances. We observe that the CPU time for each scheme increases significantly when either of the tolerances τ_1 and τ_2 becomes smaller; and for fixed τ_1 and τ_2, HSS-I always costs much less CPU than HSS-II. Analogously, the price paid for HSS-II to

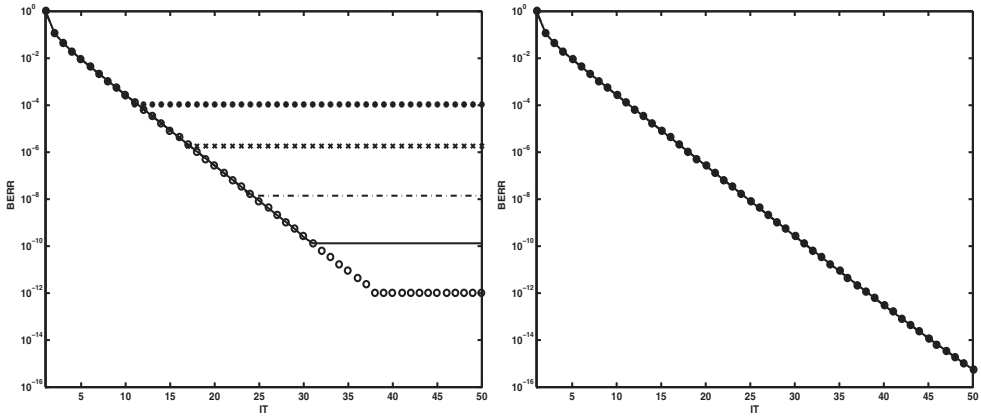

FIG. 1. *Pictures of BERR versus IT for PMHSS when $m = 64$ with PMHSS-I (left) and PMHSS-II (right). $\tau = 10^{-4}$: the star line "$*\,*\,*$", $\tau = 10^{-6}$: the cross line "$\times\,\times\,\times$", $\tau = 10^{-8}$: the dash-dotted line "$-\cdot-\cdot-\cdot$", $\tau = 10^{-10}$: the solid line "$---$", and $\tau = 10^{-12}$: the circle line "$\circ\,\circ\,\circ$".*

TABLE 2
Numerical results of HSS for Example 4.2 *with $m = 64$ and $Re = 10$ at $IT = 250$.*

τ_1	Method	Index	τ_2					
			10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}	10^{-14}
10^{-4}	HSS-I	CPU	250.17	339.56	423.30	510.85	580.82	583.76
		BERR	3.01E-04	8.66E-05	1.23E-04	1.21E-04	1.21E-04	1.21E-04
	HSS-II	CPU	389.48	431.18	495.28	539.88	621.17	703.32
		BERR	7.33E-17	7.46E-17	7.34E-17	7.34E-17	7.48E-17	7.41E-17
10^{-6}	HSS-I	CPU	253.96	345.61	426.30	519.16	584.64	585.40
		BERR	2.52E-05	1.56E-06	2.51E-06	1.09E-06	1.09E-06	1.09E-06
	HSS-II	CPU	402.32	442.74	502.64	559.40	626.10	714.77
		BERR	7.44E-17	7.42E-17	7.46E-17	7.44E-17	7.44E-17	7.39E-17
10^{-8}	HSS-I	CPU	272.11	352.52	449.93	524.44	597.51	622.80
		BERR	2.50E-05	1.54E-07	9.74E-09	7.98E-09	1.30E-08	1.30E-08
	HSS-II	CPU	455.02	480.71	513.60	574.09	641.13	731.26
		BERR	7.47E-17	7.47E-17	7.34E-17	7.41E-17	7.46E-17	7.35E-17
10^{-10}	HSS-I	CPU	304.95	391.21	466.61	578.35	697.57	941.22
		BERR	2.50E-05	1.48E-07	1.94E-09	3.64E-10	2.20E-10	2.20E-10
	HSS-II	CPU	437.73	482.88	535.62	585.11	655.26	739.58
		BERR	7.46E-17	7.52E-17	7.53E-17	7.48E-17	7.34E-17	7.45E-17
10^{-12}	HSS-I	CPU	508.51	614.75	735.35	824.93	903.22	985.34
		BERR	2.50E-05	1.48E-07	1.90E-09	2.48E-11	3.78E-12	3.78E-12
	HSS-II	CPU	459.49	532.42	579.38	609.75	683.28	768.80
		BERR	7.61E-17	7.44E-17	7.50E-17	7.47E-17	7.49E-17	7.19E-17
10^{-14}	HSS-I	CPU	520.55	668.16	764.35	756.78	624.93	628.16
		BERR	2.50E-05	1.48E-07	1.90E-09	2.48E-11	3.78E-12	3.78E-12
	HSS-II	CPU	481.66	521.08	589.94	642.01	713.43	806.06
		BERR	7.49E-17	7.51E-17	7.40E-17	7.46E-17	7.42E-17	7.45E-17

achieve a higher accuracy than HSS-I is more CPU time. In fact, the inner iterations of HSS-II start from zero initial guess, so at each outer iteration step HSS-II requires more inner iteration steps than HSS-I to satisfy its stopping criterion. Moreover, the normwise backward error of HSS-I is of an order of magnitude like $\mathcal{O}(\max\{\tau_1, \tau_2\})$, but that of HSS-II is always of the order $\mathcal{O}(u)$ no matter whether τ_1 or τ_2 is large or small; see Figure 2, in which $\tau \equiv \tau_1 = \tau_2$. Hence, in actual computations, HSS-II is

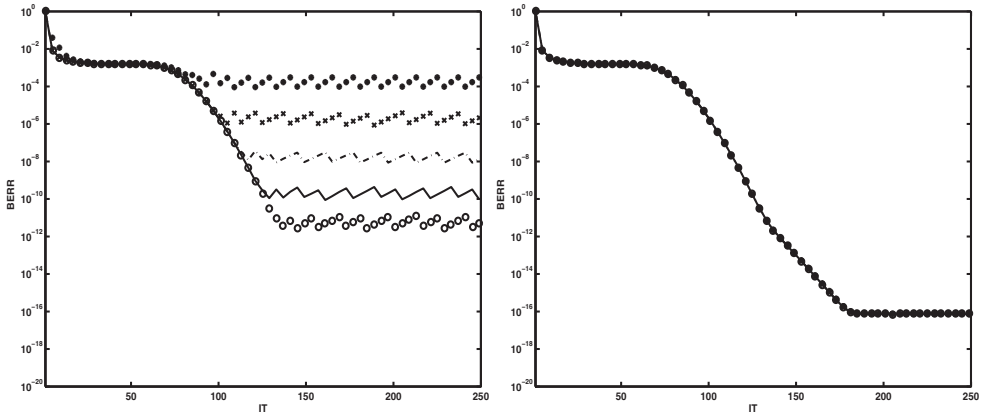

FIG. 2. *Pictures of BERR versus IT for HSS when* $m = 64$ *and* $\tau_1 = \tau_2 \equiv \tau$ *with HSS-I (left) and HSS-II (right).* $\tau = 10^{-4}$*: the star line "*$*\,*\,*$*", *$\tau = 10^{-6}$*: the cross line "*$\times\,\times\,\times$*", *$\tau = 10^{-8}$*: the dash-dotted line "*$-\cdot-\cdot-\cdot-$*", *$\tau = 10^{-10}$*: the solid line "*$---$*", and *$\tau = 10^{-12}$*: the circle line "*$\circ\,\circ\,\circ$*".*

always backward stable independent of the tolerances τ_1 and τ_2, but HSS-I may be backward stable only for those τ_1 and τ_2 of about the order $\mathcal{O}(u)$ of magnitude.

With regard to Tables 1 and 2, the reason for the CPU times of PMHSS-I and HSS-I being much less than the CPU times of PMHSS-II and HSS-II is that the stopping criteria of the inner iteration methods adopted in PMHSS-I and HSS-I are much more easily achievable than those adopted in PMHSS-II and HSS-II, respectively, especially when the iterates are approaching the exact solution of the system of linear equations (1.1). Admittedly, as the inexactly computed solutions have very different accuracy, the CPUs here do not reflect the computing efficiency of both iteration schemes, and they only show the overall (or the average) computational costs of the inner iterations or, in other words, the average numbers of inner iteration steps.

5. Concluding remarks. Stationary matrix splitting iteration methods for solving large sparse systems of linear equations typically have two equivalent formulations: a residual-updating scheme and a direct-splitting scheme. For both formulations, inexact solutions of inner linear systems associated with the matrix splittings may considerably influence the convergence and the accuracy of the approximate solutions computed in finite precision arithmetic. Both theoretical analyses and numerical experiments have shown that the former is always significantly more accurate than the latter for a large spectrum of the stopping tolerance in the inner iteration method. However, the price paid for the residual-updating schemes to achieve a higher accuracy than the direct-splitting schemes is more CPU time. These conclusions hold equally true for both one-step and two-step matrix splitting iteration methods.

We point out that the errors and the residuals in the matrix splitting iteration methods could be overestimated due to the standard assumptions on the worst-case behavior in rounding error analysis. However, it is clear that for the direct-splitting schemes, the consistency of the linear system (1.1) is essentially violated after each inexact application of the inverse in the formula (2.1) or (3.1). Therefore, the error and the residual bounds depend on τ (or τ_1 and τ_2) and only mildly on u, causing the direct-splitting iteration schemes to be less competitive than the residual-updating iteration schemes. However, it implies that if an iteration sequence has to achieve

the working accuracy by solving the inner linear systems inexactly, the only way to attain this object is to make a kind of iterative refinement. In fact, the inner iterations of the residual-updating schemes start from zero initial guess, so at each outer iteration step the residual-updating schemes requires more inner iteration steps than the direct-splitting schemes to satisfy their stopping criterions. In addition, using variable tolerance $\tau^{(k)}$ (or tolerances $\tau_1^{(k)}$ and $\tau_2^{(k)}$) at the kth iterate may improve the convergence property and numerical behavior of both direct-splitting and residual-updating iteration schemes of the one-step and the two-step types [7, 25], but this needs further theoretical verification and numerical validation.

REFERENCES

[1] O. Axelsson and A. Kucherov, *Real valued iterative methods for solving complex symmetric linear systems*, Numer. Linear Algebra Appl., 7 (2000), pp. 197–218.

[2] Z.-Z. Bai, *A class of two-stage iterative methods for systems of weakly nonlinear equations*, Numer. Algorithms, 14 (1997), pp. 295–319.

[3] Z.-Z. Bai, *On the convergence of additive and multiplicative splitting iterations for systems of linear equations*, J. Comput. Appl. Math., 154 (2003), pp. 195–214.

[4] Z.-Z. Bai, M. Benzi, and F. Chen, *Modified HSS iteration methods for a class of complex symmetric linear systems*, Computing, 87 (2010), pp. 93–111.

[5] Z.-Z. Bai, M. Benzi, and F. Chen, *On preconditioned MHSS iteration methods for complex symmetric linear systems*, Numer. Algorithms, 56 (2011), pp. 297–317.

[6] Z.-Z. Bai, G.H. Golub, L.-Z. Lu, and J.-F. Yin, *Block triangular and skew-Hermitian splitting methods for positive-definite linear systems*, SIAM J. Sci. Comput., 26 (2005), pp. 844–863.

[7] 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.

[8] Z.-Z. Bai, G.H. Golub, and M.K. Ng, *On inexact Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*, Linear Algebra Appl., 428 (2008), pp. 413–440.

[9] M. Benzi and D.B. Szyld, *Existence and uniqueness of splittings for stationary iterative methods with applications to alternating methods*, Numer. Math., 76 (1997), pp. 309–321.

[10] Z.-H. Cao, *Rounding error analysis of two-stage iterative methods for large linear systems*, Appl. Math. Comput., 139 (2003), pp. 371–381.

[11] J. Douglas and H.H. Rachford, *On the numerical solution of heat conduction problems in two and three space variables*, Trans. Amer. Math. Soc., 82 (1956), pp. 421–439.

[12] A. Frommer and D.B. Szyld, *H-splittings and two-stage iterative methods*, Numer. Math., 63 (1992), pp. 345–356.

[13] G.H. Golub and C.F. Van Loan, *Matrix Computations*, 3rd ed., The Johns Hopkins University Press, Baltimore, MD, 1996.

[14] L.A. Hageman and D.M. Young, *Applied Iterative Methods*, Academic Press, New York, 1981.

[15] N.J. Higham, *Accuracy and Stability of Numerical Algorithms*, 2nd ed., SIAM, Philadelphia, 2002.

[16] N.J. Higham and P.A. Knight, *Componentwise error analysis for stationary iterative methods*, in Linear Algebra, Markov Chains, and Queueing Models, C.D. Meyer and R.J. Plemmons, eds., IMA Vol. Math. Appl., 48 (1993), pp. 29–46.

[17] E. Isaacson and H.B. Keller, *Analysis of Numerical Methods*, Dover, New York, 1966.

[18] P. Jiránek and M. Rozložník, *Maximum attainable accuracy of inexact saddle point solvers*, SIAM J. Matrix Anal. Appl., 29 (2008), pp. 1297–1321.

[19] P. Jiránek and M. Rozložník, *Limiting accuracy of segregated solution methods for nonsymmetric saddle point problems*, J. Comput. Appl. Math., 215 (2008), pp. 28–37.

[20] R.J. Lanzkron, D.J. Rose, and D.B. Szyld, *Convergence of nested classical iterative methods for linear systems*, Numer. Math., 58 (1991), pp. 685–702.

[21] N.K. Nichols, *On the convergence of two-stage iterative processes for solving linear equations*, SIAM J. Numer. Anal., 10 (1973), pp. 460–469.

[22] D.W. Peaceman and H.H. Rachford, *The numerical solution of parabolic and elliptic differential equations*, J. Soc. Indust. Appl. Math., 3 (1955), pp. 28–41.

[23] H.H. Rachford, *Rounding errors in alternating direction methods for parabolic problems*, SIAM J. Numer. Anal., 5 (1968), pp. 407–421.
[24] 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.
[25] J. Van den Eshof and G.L.G. Sleijpen, *Inexact Krylov subspace methods for linear systems*, SIAM J. Matrix Anal. Appl., 26 (2004), pp. 125–153.
[26] R.S. Varga, *Matrix Iterative Analysis*, Springer-Verlag, Berlin, 2000.
[27] Q. Xiang and S.-P. Wu, *A modified alternating direction method for positive definite systems*, in Information Engineering and Applications, R.-B. Zhu and Y. Ma, eds., Lect. Notes Electr. Eng. 154, 2012, pp. 437–444.
[28] Q. Xiang, S.-P. Wu, and Y. Xu, *Alternating lower-upper splitting iterative method and its application in CFD*, J. Beijing Univ. Aeron. Astron., 38 (2012), pp. 953–956 (in Chinese).