Iterative Schemes Induced by Block Splittings for Solving Absolute Value Equations

Nafiseh Nasseri Shams[a], Alireza Fakharzadeh Jahromi[a], Fatemeh Panjeh Ali Beik[b]

[a]Faculty of Mathematics, Shiraz University of Technology, Shiraz, Iran
[b]Department of Mathematics, Vali-e-Asr University of Rafsanjan, PO Box 518, Rafsanjan, Iran

Abstract. In this paper, we develop the idea of constructing iterative methods based on block splittings (BBS) to solve absolute value equations. The class of BBS methods incorporates the well–known Picard iterative method as a special case. Convergence properties of mentioned schemes are proved under some sufficient conditions. Numerical experiments are examined to compare the performance of the iterative schemes of BBS-type with some of existing approaches in the literature such as generalized Newton and Picard(-HSS) iterative methods.

1. Introduction

The generic absolute value equation (AVE) is defined as follows:

$$Ax - |x| = b, \tag{1}$$

where $A \in \mathbb{R}^{n \times n}$ and the right-hand side $b \in \mathbb{R}^n$ are known and $x = (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^n$ is the unknown vector to be determined. Here, $|x|$ is a vector in $\mathbb{R}^n$ whose i-th component is the absolute value of x_i for $i = 1, 2, \ldots, n$. The absolute value equations of the form (1) may appear in linear programming, convex quadratic programing, bimatrix games and other areas of optimization, scientific computing and engineering, the readers are advised to see [1, 8, 9, 13, 14, 16, 20, 21] and the references therein for further details.

For $W \in \mathbb{C}^{n \times n}$, the notation $\|W\|$ denotes the spectral norm defined by

$$\|W\| = \max\{\|Wx\| : x \in \mathbb{C}^n, \|x\| = 1\}$$

where $\|x\|^2 = x^H x$.

We comment here that the condition $\|A^{-1}\| < 1$ ensures the existence of a unique solution for the AVE (1) by the following proposition; see [10] for the proof.

Proposition 1.1. *Assume that $A \in \mathbb{R}^{n \times n}$ is invertible. If $\|A^{-1}\| < 1$, then the AVE (1) has a unique solution for any $b \in \mathbb{R}^n$.*

Before stating the main contribution of this paper, we give a brief overview on the literature in the context of applying iterative techniques for solving (1); for more details see [3, 5, 8, 9, 11, 12, 14–17, 19, 21, 23] and the references therein. For instance, Mangasarian [11] discussed that the AVE (1) is equivalent to minimization of a piecewise linear concave function on a polyhedral set and proposed a finite successive linearizion algorithm. In [12], Mangasarian exploited the nonsmooth Newton method for solving AVEs. The proposed method is given by,

$$x^{(k+1)} = (A - D(x^{(k)}))^{-1}b, \quad k = 0, 1, 2, \dots, n, \tag{2}$$

where $D(x) = \text{diag}(\text{sgn}(x))$ for $x \in \mathbb{R}^n$. Two sufficient conditions were established for the linear convergence of the above iterative scheme, for more details, we recall these results as follows.

Proposition 1.2. *Let $\|(A - D)^{-1}\| < \frac{1}{3}$ for any diagonal matrix D with diagonal elements of ± 1 or 0, the generalized Newton iteration* (2) *converges linearly from any starting point to a solution x^* for any solvable AVE* (1).

Proposition 1.3. *Let $\|A^{-1}\| < \frac{1}{4}$ and $\|D(x^{(k)})\| \neq 0$. Then, the AVE* (1) *has a unique solution and the generalized Newton iteration* (2) *is well defined and converges linearly to the unique solution of AVE from any starting point.*

Basically, at each step of (2), one need to solve a linear system of equations with coefficient matrix $A - D(x^{(k)})$ which depends on the index k; consequently, the iterative method (2) is computationally expensive. The Picard iteration for solving the AVE is given by

$$x^{(k+1)} = A^{-1}(|x^{(k)}| + b), \quad k = 0, 1, 2, \dots, n, \tag{3}$$

see [15, 17] for further details. As seen, a linear system of equations with a constant coefficient matrix should be solved at each iteration of (3). This fact that the coefficient matrix in all of iterations remains fixed, can be seen as an advantage of the Picard iteration over the generalized Newton method (2). When the coefficient matrix A is ill-conditioned, it is more advisable to use an iterative scheme instead of a direct solver as the act of A^{-1}. Bai and Yang [3] proposed the Picard-HSS iteration method to solve a class of nonlinear systems. The Picard-HSS iteration method consists of using Hermitian and skew-Hermitian splitting (HSS) method [2] at each step of the Picard iteration for solving the linear system with the coefficient matrix A. The Picard-HSS iteration method was revisited for solving AVEs in [19] and the following iterative scheme was proposed.

The Picard-HSS iteration method: (Residual-Updating Variant) Let $A \in \mathbb{R}^{n \times n}$ be a positive definite matrix. Consider the Hermitian and skew-Hermitian splitting $A = H + S$ with $H = \frac{1}{2}(A + A^T)$ and $S = \frac{1}{2}(A - A^T)$. Given an initial guess $x^{(0)} \in \mathbb{R}^n$ and a sequence $\{l_k\}_{k=0}^{\infty}$ of positive integers, compute $x^{(k+1)}$ by the following iterative scheme for $k = 0, 1, 2, \dots$, until the convergence:

- Set $s^{(k,0)} = 0$ and $b^{(k)} = |x^{(k)}| + b - Ax^{(k)}$;

- For $l = 0, 1, 2, \dots, l_k - 1$, find $s^{(k,l+1)}$ by solving the following linear systems of equations:

$$\begin{aligned}
(\alpha I + H)s^{(k,l+\frac{1}{2})} &= (\alpha I - S)s^{(k,l)} + b^{(k)}, \\
(\alpha I + S)s^{(k,l+1)} &= (\alpha I - H)s^{(k,l+\frac{1}{2})} + b^{(k)},
\end{aligned} \tag{4}$$

where α is a given positive constant.

- Set $x^{(k+1)} = x^{(k)} + s^{(k,l_k)}$.

Here we comment that the convergence of the Picard-HSS iteration is proved under some sufficient conditions such as the positive definiteness of A in (1), $\nu = \|A^{-1}\| < 1$ and $l = \liminf_{k \to \infty} l_k \geq N$, where N is a natural number satisfying

$$\|T(\alpha)^s\| < \frac{1 - \nu}{1 + \nu}; \qquad \forall s \geq N$$

in which

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

However, from the reported numerical experiments in [19], it is seen that the method is applicable for the case that one or all of these conditions are not satisfied.

Recently, Ke [9] proposed the following iterative scheme to solve the AVE (1). For $k = 0, 1, 2, \ldots$, the method produces sequence of approximation $\{[x^{(k)}; y^{(k)}]\}_{k=0}^{\infty}$ by the following iterative scheme in which τ is a prescribed positive constant,

$$\begin{cases} x^{(k+1)} = A^{-1}(y^{(k)} + b), \\ y^{(k+1)} = (1 - \tau)y^{(k)} + \tau \left| x^{(k+1)} \right|; \end{cases} \tag{5}$$

here the initial guess $x^{(0)}$ is given and $y^{(0)} = |x^{(0)}|$.

It is worth to recall the succeeding lemma from [4] which is useful to analyze the convergence of iteration (5). We also use the lemma for deriving some of our main results.

Lemma 1.4. *For any vectors $x, y \in \mathbb{R}^n$, the following statements hold*

- $\||x| - |y|\| \le \|x - y\|$;

- *if $0 \le x \le y$, then $\|x\| \le \|y\|$;*

- *if $x \le y$ and P is a nonnegative matrix then $Px \le Py$.*

The following theorem provides sufficient conditions under which the above iterative method converges to the unique solution of (1); see [9] for the proof.

Theorem 1.5. *Let $A \in \mathbb{R}^{n \times n}$ be a nonsingular matrix. Denote*

$$\nu = \|A^{-1}\| \quad and \quad \mathcal{E}_k = \left[\begin{array}{c} e(k, x) \\ e(k, y) \end{array} \right],$$

with $e(k, x) = x^* - x^{(k)}$ and $e(k, y) = y^* - y^{(k)}$ where $x^{(k)}$ is the k-th approximate solution obtained by (5), x^* is the unique solution of AVE (1) and $y^* = |x^*|$. If

$$0 < \nu < \frac{\sqrt{2}}{2} \quad and \quad \frac{1 - \sqrt{1 - \nu^2}}{1 - \nu} < \tau < \frac{1 + \sqrt{1 - \nu^2}}{1 + \nu},$$

then $\|\mathcal{E}_{k+1}\| < \|\mathcal{E}_k\|$ for $k = 0, 1, 2, \ldots$, which ensures that iteration (5) is convergent.

Experimental results in [9] illustrate the feasibility of iterative scheme (5) which is induced by a block splitting. For future work, there is also a potential to develop alternative iterative methods when A is not well-conditioned, i.e., one may use few steps of an iterative method instead of implementing a direct solver for the act of A^{-1} at each step. Here, we are inspired to consider another possible block splitting and construct an alternative iterative method assuming that A in (1) is not ill-conditioned. Basically, it is aimed to develop a method whose convergence can be established for larger area without any further restriction on $\nu = \|A^{-1}\|$ except $\nu < 1$.

The remainder of this paper is organized as follows: In Section 2, we present a new iterative method for solving (1) and analyze its convergence properties in details. Furthermore, we briefly discuss on suitable choice for the parameter of proposed method. In Section 3, some numerical results are disclosed to illustrate the feasibility and applicability of the proposed method and to compare its performance with some of existing methods in the literature. Concluding remarks can be found in Section 4.

2. Iterative methods based on block splitting

This section deals with developing a new iterative method for solving (1). We first start by reviewing some basic concepts and relevant works in the literature. Then, we propose a new iterative method which relies on a fixed parameter and analyze its convergence properties in details. Brief discussions are included on estimating the optimal parameter of the new method in a separate part.

Evidently, iterative method (5) reduces to the Picard iteration for $\tau = 1$ and $y^{(0)} = |x^{(0)}|$ whereas the experimentally obtained optimum values for τ are not necessarily equal to one, see [9, Table 1]. Generally, a BBS-type iterative scheme is extracted from a splitting of (preconditioned) block matrix. In fact, setting $y = |x|$, one may rewrite $Ax - |x| = b$ as follows:

$$\begin{cases} Ax - y = b \\ -|x| + y = 0 \end{cases},$$

or equivalently,

$$\mathcal{A}\mathbf{z} = \begin{bmatrix} A & -I \\ -D(x) & I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix} := \mathbf{b} \tag{6}$$

with $D(x) := \text{diag}(\text{sgn}(x))$ for $x \in \mathbb{R}^n$.

The iterative scheme (5) can be seen as a method for solving $\mathcal{P}_\tau \mathcal{A}\mathbf{z} = \mathcal{P}_\tau \mathbf{b}$ by using the following block splitting:

$$\mathcal{P}_\tau \mathcal{A} = \begin{bmatrix} A & 0 \\ -\tau D(x) & I \end{bmatrix} - \begin{bmatrix} 0 & I \\ 0 & (1-\tau)I \end{bmatrix},$$

with

$$\mathcal{P}_\tau = \begin{bmatrix} I & 0 \\ 0 & \tau I \end{bmatrix}.$$

In [7], the SOR-like method is extended for solving (1). More precisely, the following iterative scheme is developed

$$\begin{cases} x^{(k+1)} = (1-\omega)x^{(k)} + \omega A^{-1}\left(y^{(k)} + b\right) \\ y^{(k+1)} = (1-\omega)y^{(k)} + \omega \left|x^{(k+1)}\right| \end{cases} \tag{7}$$

which can be regarded as an iterative scheme corresponding to the following SOR-like block splitting:

$$\mathcal{A} = \frac{1}{\omega} \begin{bmatrix} A & 0 \\ -\omega D(x) & I \end{bmatrix} - \frac{1}{\omega} \begin{bmatrix} (1-\omega)A & \omega I \\ 0 & (1-\omega)I \end{bmatrix}.$$

The established convergence results for the above method rely on the spectrum of $D(x^{(k+1)})A^{-1}$. In fact, it is proved that if the eigenvalues of $D(x^{(k+1)})A^{-1}$ are real then iterative scheme (7) is convergent for $0 < \omega \leq 1$. In the case that all eigenvalues of $D(x^{(k+1)})A^{-1}$ are positive then the method is convergent for $0 < \omega < 2$. In practice, it is not easy to check the validity of required assumptions on spectrum of $D(x^{(k+1)})A^{-1}$ due to the fact that $D(x^{(k+1)})A^{-1}$ may change at each iteration. For the choice of optimum parameter, the following theorem is established.

Theorem 2.1. [7, Theorem 3.2] *Let $A \in \mathbb{R}^{n \times n}$, $\|A^{-1}\| < 1$ and $\rho = \rho(D(x^{(k+1)})A^{-1})$. Suppose that all eigenvalues of $D(x^{(k+1)})A^{-1}$ are positive. Then the optimal parameter ω_o is given by*

$$\omega_o = \frac{2}{1 + \sqrt{1 - \rho}}. \tag{8}$$

Our goal is to examine an alternative BBS-type iterative method using a different block splitting for $\mathcal{A}$.

2.1. Proposed method and its convergence analysis

In this part, we construct a new iterative method for solving (1). To this end, we first need to split the coefficient matrix $\mathcal{A}$ in (6). For a prescribed positive constant τ, let us consider the following block splitting for $\mathcal{A}$:

$$\mathcal{A} = \begin{bmatrix} A & 0 \\ -\tau D(x) & I \end{bmatrix} - \begin{bmatrix} 0 & I \\ (1-\tau)D(x) & 0 \end{bmatrix}.$$

Using the above decomposition, we rewrite the AVE as a fixed point problem, i.e.,

$$\begin{bmatrix} A & 0 \\ -\tau D(x) & I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 & I \\ (1-\tau)D(x) & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} b \\ 0 \end{bmatrix}. \tag{9}$$

Considering the preceding equation, we are motived to examine the following iterative scheme

$$\begin{cases} x^{(k+1)} = A^{-1}\left(y^{(k)} + b\right) \\ y^{(k+1)} = (1-\tau)\left|x^{(k)}\right| + \tau\left|x^{(k+1)}\right| \end{cases} \tag{10}$$

for $k = 0, 1, 2, \ldots$, where the initial guess $x^{(0)}$ is given and $y^{(0)} = \left|x^{(0)}\right|$. In the sequel, we establish three propositions which are useful for analyzing the convergence of iteration (10).

Proposition 2.2. *Let $0 < v < 1$. If $0 < \tau < \frac{1+\sqrt{2-v^4}}{v^2+1}$ then $(1-\tau)^2 + v^2(1+\tau^2) < 2$.*

Proof. Let $f(\tau) = 2 - (1-\tau)^2 - v^2(1+\tau^2)$. By some algebraic computations, we observe that

$$f(\tau) = -(1+v^2)\tau^2 + 2\tau - v^2 + 1.$$

The roots of the above quadratic equations are given by

$$\tau_1 = \frac{1+\sqrt{2-v^4}}{v^2+1} \quad \text{and} \quad \tau_2 = \frac{1-\sqrt{2-v^4}}{v^2+1}.$$

Notice that $\tau_2 \leq 0$ and it is immediate to see that $f(\tau) > 0$ for $0 < \tau < \tau_1$ which completes the proof. $\square$

Proposition 2.3. *Let v be a positive constant. Then*

$$(1-\tau)^2 + v^2(\tau^2+1) \geq 2v\left|1-\tau\right|.$$

where τ is a prescribed real parameter.

Proof. The assertion clearly holds for $\tau = 1$. In the rest of proof, we may assume that $\tau \neq 1$ without loss of generality. Let $h(\tau) = (1-\tau)^2 + v^2(\tau^2+1) - 2v\left|1-\tau\right|$. Obviously, one may see that $h(\tau)$ reduces to the quadratic equations

$$h_1(\tau) = \tau^2(v^2+1) + \tau(2v-2) + (v-1)^2,$$

and

$$h_2(\tau) = \tau^2(v^2+1) - \tau(2v+2) + (v+1)^2,$$

for $\tau < 1$ and $\tau > 1$, respectively. The sign of discriminates corresponding to $h_1(\tau)$ and $h_2(\tau)$ are negative for all $v > 0$. This ensures that the values of $h_1(\tau)$ and $h_1(\tau)$ are positive for all $v > 0$. $\square$

Remark 2.4. *We comment that Proposition 2.3 shows that if $v > 0$ then*

$$\left((1-\tau)^2 + v^2(1+\tau^2)\right)^2 - 4v^2(1-\tau)^2 > 0.$$

Proposition 2.5. *Let $0 < v < 1$. If*

$$0 < \tau < \min\left\{\frac{1 + \sqrt{2 - v^4}}{v^2 + 1}, 2(1 - v^2)\right\},$$

then,

$$\sqrt{\left((1 - \tau)^2 + v^2(1 + \tau^2)\right)^2 - 4v^2(1 - \tau)^2} < 2 - (1 - \tau)^2 - v^2(1 + \tau^2).$$

Proof. For the notational simplicity, we set $\xi = (1-\tau)^2 + v^2(1+\tau^2)$. Evidently the function $g(\tau) = \tau(\tau - 2(1 - v^2))$ is negative for $0 < \tau < 2(1 - v^2)$. It can be seen that

$$g(\tau) = \tau^2 + \tau(2v^2 - 2) = -v^2(1 - \tau)^2 - 1 + (1 - \tau)^2 + v^2(1 + \tau^2).$$

Consequently, $g(\tau) < 0$ implies that $-v^2(1 - \tau)^2 < 1 - \xi$ which is equivalent to say that

$$\xi^2 - 4v^2(1 - \tau)^2 \quad < \quad \xi^2 + 4 - 4\xi.$$

Therefore, we conclude

$$\xi^2 - 4v^2(1 - \tau)^2 < (2 - \xi)^2.$$

Notice that $2 - \xi > 0$ by Proposition 2.2. The results follows after taking square root from the both sides of the above inequality. $\square$

Now we present the following theorem which establishes the convergence of proposed iterative method under some sufficient conditions. For proving the theorem, we applied the same strategy used in the proof of Theorem 1.5. Nevertheless, we do not need to set further restriction on v.

Theorem 2.6. *Let A be a nonsingular matrix such that $v = \|A^{-1}\| < 1$. The iterative scheme* (10) *converges to the unique solution of* (1) *for any initial guess $x^{(0)}$, if*

$$0 < \tau < \min\left\{\frac{1 + \sqrt{2 - v^4}}{v^2 + 1}, 2(1 - v^2)\right\}. \tag{11}$$

Proof. Let x^* be the exact solution of (1) and $y^* = |x^*|$. Suppose that the vector $[x^{(k)}; y^{(k)}]$ stands for the k-th approximate solution obtained by iterative method (10). For simplicity, we define

$$e(k, x) = x^* - x^{(k)} \quad \text{and} \quad e(k, y) = y^* - y^{(k)},$$

recalling that $y^* = |x^*|$. Note that $x^* = A^{-1}y^* + A^{-1}b$, therefore from (10), it is not difficult to verify that

$$e(k + 1, x) = A^{-1}e(k, y).$$

Straightforward computations show that

$$e(k + 1, y) = \tau(|x^*| - |x^{(k+1)}|) + (1 - \tau)(|x^*| - |x^{(k)}|). \tag{12}$$

As a result, we obtain

$$\|e(k + 1, x)\| \le \|A^{-1}\|\|e(k, y)\| = v\|e(k, y)\|. \tag{13}$$

Using Lemma 1.4, the above inequality and (12), we derive

$$\begin{aligned}\|e(k + 1, y)\| &\le& \tau\|e(k + 1, x)\| + |1 - \tau|\|e(k, x)\| \\ &\le& \tau v\|e(k, y)\| + |1 - \tau|\|e(k, x)\|. \end{aligned} \tag{14}$$

Defining $\mathcal{E}_k := [\|e(k, x)\|; \|e(k, y)\|]$, from (13) and (14), we get

$$\mathcal{E}_{k+1} \leq \mathcal{W}\mathcal{E}_k \tag{15}$$

where

$$\mathcal{W} = \begin{bmatrix} 0 & \nu \\ |1 - \tau| & \tau\nu \end{bmatrix}.$$

Notice that Lemma 1.4 implies that

$$\|\mathcal{E}_{k+1}\| \leq \|\mathcal{W}\|\|\mathcal{E}_k\|. \tag{16}$$

In the rest of the proof, we show that if τ satisfies in (11) then $\|\mathcal{W}\| < 1$ which concludes the convergence of iteration (10). To this end, we show that the eigenvalues of the symmetric positive semidefinite matrix $\mathcal{W}^T\mathcal{W}$ belong to interval $[0, 1)$. Evidently,

$$\mathcal{W}^T\mathcal{W} = \begin{bmatrix} (1 - \tau)^2 & \tau\nu|1 - \tau| \\ \tau\nu|1 - \tau| & (1 + \tau^2)\nu^2 \end{bmatrix}.$$

Let λ be an arbitrary eigenvalue of $\mathcal{W}^T\mathcal{W}$; as a result, we have $\det(\mathcal{W}^T\mathcal{W} - \lambda I) = 0$. Therefore, it is immediate to see that λ satisfies in the following quadratic equation

$$\lambda^2 - \gamma\lambda + \eta = 0, \tag{17}$$

with $\gamma = (1 - \tau)^2 + \nu^2 + \tau^2\nu^2$ and $\eta = \nu^2(1 - \tau)^2$. The roots of (17) are given by

$$\lambda_1 = \frac{\gamma + \sqrt{\gamma^2 - 4\eta}}{2} \quad \text{and} \quad \lambda_2 = \frac{\gamma - \sqrt{\gamma^2 - 4\eta}}{2}.$$

Now, from Proposition 2.5, we have

$$\sqrt{\left((1 - \tau)^2 + \nu^2(1 + \tau^2)\right)^2 - 4\nu^2(1 - \tau)^2} < 2 - (1 - \tau)^2 - \nu^2(1 + \tau^2),$$

which is equivalent to say that $\lambda_1 < 1$ and therefore the result follows immediately. $\square$

Notice that in Theorem 1.5 the convergence of iterative scheme (5) is proved assuming that $0 < \nu < \frac{\sqrt{2}}{2}$. Also, we comment that when ν becomes very close to 1 then the following value in the statement of Theorem 2.6,

$$\min\left\{\frac{1 + \sqrt{2 - \nu^4}}{\nu^2 + 1}, 2(1 - \nu^2)\right\},$$

is strictly less than one. Note that the above value gives an upper bound for the convergence interval of iterative method (10). Therefore, we use another strategy to derive wider convergence intervals for τ in iterations (5) and (10). To this end, we first recall the following lemma.

Lemma 2.7. *[22] Consider the quadratic equation $x^2 - bx + c = 0$, where b and c are real numbers. Both roots of the equation are less than one in modulus if and only if $|c| < 1$ and $|b| < 1 + c$.*

From the following theorem, one may conclude that it is not advisable to choose τ outside interval $(0, \frac{1+\nu}{2\nu})$.

Theorem 2.8. *Let A be a nonsingular matrix such that $\nu = \|A^{-1}\| < 1$. The iterative scheme (10) converges to the unique solution of (1) for any initial guess $x^{(0)}$, if*

$$0 < \tau < \frac{1 + \nu}{2\nu}. \tag{18}$$

Proof. Let us follow the same notations used in the proof of Theorem 2.6. From (15), it is observed that

$$\mathcal{E}_{k+1} \leq \mathcal{W}\mathcal{E}_k.$$

In view of Lemma 1.4, this implies that

$$\mathcal{E}_\ell \leq \mathcal{W}^\ell \mathcal{E}_0, \quad \ell > 0.$$

It is well-known that $\lim_{\ell \to \infty} \mathcal{W}^\ell = 0$ iff $\rho(\mathcal{W}) < 1$; hence, it can be deduced that $\lim_{\ell \to \infty} \mathcal{E}_\ell = 0$, if $\rho(\mathcal{W}) < 1$.

Now let λ be an arbitrary nonzero eigenvalue of $\mathcal{W}$ with the corresponding eigenvector $[z_1; z_2]$. Consequently, we have

$$\begin{bmatrix} 0 & \nu \\ |1 - \tau| & \tau\nu \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \lambda \begin{bmatrix} z_1 \\ z_2 \end{bmatrix},$$

which is equivalent to say that

$$\nu z_2 = \lambda z_1, \tag{19}$$
$$|1 - \tau| z_1 + \tau\nu z_2 = \lambda z_2. \tag{20}$$

Notice that for $\lambda \neq 0$, the vector z_1 is nonzero. Otherwise, $z_2 = 0$ by (19) which is a contraction with the assumption that $[z_1; z_2]$ is an eigenvector. As a result, without loss of generality, we may assume that $\|z_1\| = 1$. By multiplying both sides of (20) on νz_1^H and substituting νz_2 from (19) into (20), it is deduced that λ satisfies the following quadratic equation

$$\lambda^2 - \tau\nu\lambda - |1 - \tau|\nu = 0.$$

Evidently, the assumption (18) concludes

$$0 < \tau < \frac{1 + \nu}{\nu}.$$

Now it is not difficult to observe that $|1 - \tau|\nu < 1$ and $\tau\nu < 1 - |1 - \tau|\nu$ which ensure that the roots of above equation are strictly lower than 1 in modulus by Lemma 2.7. $\quad\square$

The following remark reveals that iterative scheme (5) converges under weaker conditions than those assumed in [9, Theorem 2.1].

Remark 2.9. *Let $\mathcal{E}_k$ be defined as in Theorem 1.5. In [9, Theorem 1.2], it is seen that*

$$\mathcal{E}_\ell \leq \tilde{\mathcal{W}}\mathcal{E}_0, \quad \ell = 0, 1, 2, \ldots,$$

where,

$$\tilde{\mathcal{W}} = \begin{bmatrix} 0 & \nu \\ 0 & \tau\nu + |1 - \tau| \end{bmatrix},$$

Note that $\rho(\tilde{\mathcal{W}}) = \tau(\nu - 1) + 1$ for $0 < \tau \leq 1$ and $\rho(\tilde{\mathcal{W}}) = \tau(\nu + 1) - 1$ for $1 \leq \tau$. Therefore, similar to the proof of Theorem 2.8, it can be verified that the iterative scheme (5) converges for

$$0 < \tau < \frac{2}{1 + \nu}.$$

Clearly the above convergence interval is larger then the one provided in Theorem 1.5. Here, we comment that $\frac{2}{1+\nu} \leq \frac{1+\nu}{2\nu}$ which shows that the convergence interval for iterative method (5) is smaller than (10).

2.2. On the optimal choice of the parameter

In the previous part, it has been proved that the iterative scheme (10) converges to the unique solution of (1) under certain conditions. Finding the optimum value of the parameter for the presented BBS-type iterative method is not easy in general cases. We comment that finding an expression for the optimum value of the parameter in either Picard-HSS method or iterative method (5) is still an open problem. In what follows, we give an approximation for the optimum value of the parameter in proposed method which works well numerically.

Considering Eq. (9), the iteration matrix is different at each step of the proposed method. Having the k-th approximation computed, we face to the following eigenvalue problem for determining the spectrum of new iteration matrix

$$\lambda \begin{bmatrix} A & 0 \\ -\tau D(x^{(k+1)}) & I \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0 & I \\ (1-\tau)D(x^{(k)}) & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}.$$

When the iterative method (10) is convergent, there exists an integer number k such that $D(x^{(\ell)}) = D(x^{(k)})$ for $\ell \geq k$. Indeed, after few steps, the iteration matrix remains fixed, if the method is convergent. This helps us to derive an explicit formula for τ which provides a good approximation for optimum value. In the sequel, we consider the following eigenvalue problem:

$$\lambda \begin{bmatrix} A & 0 \\ -\tau D & I \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0 & I \\ (1-\tau)D & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

and briefly discuss on the choice of parameter τ which minimizes $|\lambda|$. Notice that D is a diagonal matrix and $D = D^{-1}$. Without loss of generality, we may assume that $z_2^H z_2 = 1$. It is not difficult to verify that λ satisfies the following relation

$$\lambda^2 A D^{-1} z_2 = ((1-\tau) + \lambda\tau)z_2. \tag{21}$$

Multiplying both sides of the above relation by z_2^H from left, we derive

$$(z_2^H A D^{-1} z_2)\lambda^2 = (1-\tau) + \lambda\tau. \tag{22}$$

For notational simplicity, we set $\zeta = z_2^H A D^{-1} z_2$. The above relation is equivalent to

$$\zeta\lambda^2 - \lambda\tau + (\tau - 1) = 0. \tag{23}$$

The roots of above quadratic equations satisfy

$$\lambda = \frac{\tau \pm \sqrt{\tau^2 - 4\zeta(\tau - 1)}}{2\zeta}, \tag{24}$$

assuming that $\zeta \neq 0$. For constant value of τ, in view of (22), it is not difficult to verify that $|\lambda|$ increases as the value of $|\zeta|$ gets smaller. We assume that $\tau \neq 1$ which implies that $\lambda \neq 0$. Now, from Eq. (21), it can be seen that $A D^{-1} z_2$ is a multiple of z_2, i.e., $A D^{-1} z_2 = k z_2$ where k is a nonzero scalar. Therefore, since $z_2^H z_2 = 1$, we can conclude that

$$|z_2^H A D^{-1} z_2| = \|A D^{-1} z_2\|.$$

For notational simplicity, we set $|\zeta_m| := \min_{\|z\|=1} \|A D^{-1} z\|$. It is immediate to see that[1]

$$|\zeta_m| = \frac{1}{\|D A^{-1}\|} = \frac{1}{\|A^{-1}\|},$$

---

[1] Here we used the fact that for any nonsingular matrix A, we have $\|A^{-1}\| = \frac{1}{\min_{\|x\|=1} \|Ax\|}$.

where the last equality follows from the fact that D is an orthogonal matrix. Notice that

$$
\begin{aligned}
|\zeta_m| &\leq \min_{\|z_2\|=1} \left\{ \|AD^{-1}z_2\| \mid AD^{-1}z_2 = kz_2 \quad \text{for some } k \neq 0 \right\} \\
&= \min_{\|z_2\|=1} \left\{ |\zeta| \mid \zeta = z_2^H AD^{-1}z_2 \text{ and } AD^{-1}z_2 = kz_2 \quad \text{for some } k \neq 0 \right\} := \tilde{\zeta}.
\end{aligned}
$$

In order to approximate the optimal value of the parameter τ, we work with the lower bound for the minimum of $|\zeta|$. Setting $\zeta = |\zeta_m|$, basically, quadratic equation (23) reduces to

$$
\lambda^2 - \nu\tau\lambda + \nu(\tau - 1) = 0 \tag{25}
$$

where $\nu = \|A^{-1}\|$. We comment that one could derive the following results with respect to $\tilde{\zeta}$ instead of $|\zeta_m|$. In this case, the value of ν is replaced by $\tilde{\nu} = \frac{1}{\tilde{\zeta}}$. Although this could provide a better approximation for the optimum parameter in theoretical point of view, it is not easy to numerically approximate the value of $\tilde{\nu}$.

Using Lemma 2.7, one can verify that, if

$$
0 < \tau < \frac{1+\nu}{\nu}, \tag{26}
$$

then the roots of (25) are strictly lower than one in modulus. The roots of (25) are given by

$$
\lambda = \frac{\nu\tau \pm \sqrt{\nu^2\tau^2 - 4\nu(\tau - 1)}}{2}, \tag{27}
$$

To approximate the optimal value of τ which minimizes $|\lambda|$, we set

$$
\nu^2\tau^2 - 4\nu(\tau - 1) = 0. \tag{28}
$$

The roots of above quadratic equation are given by

$$
\tau = \frac{2 \pm 2\sqrt{1-\nu}}{\nu}.
$$

Note that $\tau = \frac{2+2\sqrt{1-\nu}}{\nu} > 2$ when $\nu < 1$ and the iterative method (10) is not convergent for $\tau > 2$. The optimal value of τ can be approximated by the other root of (28), i.e.,

$$
\tau^* := \frac{2 - 2\sqrt{1-\nu}}{\nu} = \frac{2}{1 + \sqrt{1-\nu}}. \tag{29}
$$

Our numerical experiments reveal that τ^* can provide a good approximation for experimentally found optimum value of τ.

The following remark justifies that the parameter τ^* given by (29) lies in interval (26).

Remark 2.10. *Evidently, for $-2 < \eta < 1$, we have*

$$
\eta^2 + \eta - 2 < 0.
$$

Let $\nu < 1$. By substituting $\eta = \sqrt{1-\nu}$ in the above inequality, we get

$$
2 - \sqrt{1-\nu} - (1 - \nu) > 0.
$$

Multiplying the both side of above inequality by $\sqrt{1-\nu}$, by some straightforward algebraic computations, it is seen that

$$
\frac{2}{1 + \sqrt{1-\nu}} < 1 + \nu.
$$

It is immediate to see that $1 + \nu < \frac{1+\nu}{\nu}$ for $\nu < 1$. Therefore, for the parameter τ^* given by (29), we have $0 < \tau^* < \frac{1+\nu}{\nu}$.

We finish this part by a remark to comment on asymptotic convergence rate of iterative method (10); see [18, Chapter 4] for more details about the concept of asymptotic convergence rate.

Remark 2.11. *In view of Eqs.* (27) *and* (29)*, the asymptotic convergence rate of iterative method* (10) *can be estimated by*

$$R_\infty := -\mathrm{Ln}\left(\frac{\nu}{1 + \sqrt{1 - \nu}}\right).$$

In Figure 1, we depict the values of R_∞ with respect to ν for more clarification. In fact, one expect the faster convergence speed for smaller values of ν. We recall that ν refers to the spectral norm of A^{-1} dealing with (1).
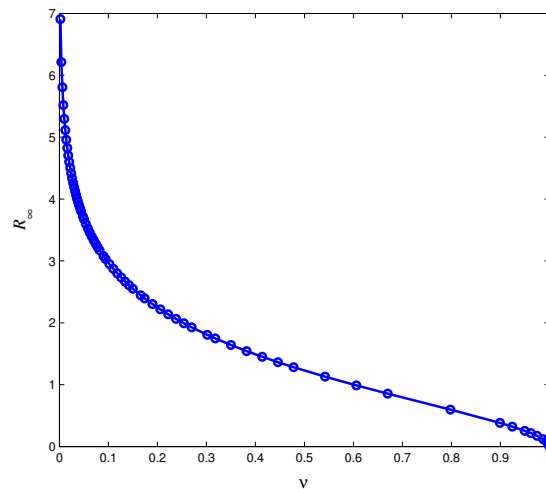

Figure 1: Estimated asymptotic convergence rate of iterative method (10) versus $\nu \in (0, 1)$.

3. Numerical experiments

In this section, we report some numerical results to compare the performance of proposed method with iterative schemes (2), (3), (4), (5) and (7). All computations were carried out on a computer with an Intel Core i7-4770K CPU @ 3.50GHz processor and 24GB RAM using MATLAB R2018b.

In all of the following experiments, the initial guess is taken to be zero vector and the iterations are stopped either

$$\delta_k = \frac{\|b + |x^{(k)}| - Ax^{(k)}\|}{\|b\|} \leq 10^{-8}, \tag{30}$$

or once the number of 1000 iterations is reached, here $\|.\|$ stands for the well–known Euclidean norm and $x^{(k)}$ refers to kth approximate solution. Notation "†" in tables means that the corresponding iterative method was stopped after 1000 iterations while the computed approximate solution was not satisfied in (30).

We comment that the condition $\nu = \|A^{-1}\| < 1$ is a sufficient condition in the convergence analyses of mentioned iterative schemes. Therefore, we also report the performance of iterations for the case $\nu \geq 1$ in Examples 3.1 and 3.2.

In the implementation of the Picard-HSS iteration (4), the inner iterations were terminated once $\|b^{(k)} - As^{(k,l)}\| \leq 10^{-2}\|b^{(k)}\|$ or $l_k = 10$ for $k = 0, 1, 2, \ldots$, which is similar to the stopping criterion exploited in [19] for inner loops. In [19] only the number of outer iterations was reported, here, we disclose the sum of inner and outer iterations steps for more details.

Similar to [9, 19], the optimal value of parameters in iterations (4) and (5) were found experimentally for which the examined iterative schemes have their best convergence speed. For the proposed method, we report the results for τ^* (cf. (29)) when $\nu < 1$. In Examples 3.1 and 3.2, results for the experimentally found optimum value of τ are also reported to demonstrate that τ^* can provide a good approximation for the optimum value of τ when $\nu < 1$.

The experimentally obtained optimal value of parameter τ for iterative methods (5) and (10) are respectively reported under $\tau_{exp}^{(1)}$ and $\tau_{exp}^{(2)}$.

For the act of A^{-1} in the implementation of BBS-type iterative schemes, in Examples 3.1, 3.2 and 3.3, we used the LU factorization. For the generalized Newton iteration (2), we disclose the results for two different variants of implementation:

- **Version 1:** We used Matlab backslash "\" in all iterations for applying the inverse of $A - D(x^{(k)})$ ($k = 0, 1, 2, \ldots$);

- **Version 2:** At each iterate, we computed the LU factorization of $A - D(x^{(k)})$ and employed its inverse accordingly.

Our observations demonstrate the first version outperforms the second one significantly for the case that matrix A is sparse.

Example 3.1. [19] Consider the two-dimensional convection diffusion equation

$$-(u_{xx} + u_{yy}) + q(u_x + u_y) + pu = f(x, y) \qquad (x, y) \in \Omega,$$
$$u(x, y) = 0 \qquad u(x, y) \in \partial\Omega,$$

where $\Omega = (0, 1) \times (0, 1)$, $\partial\Omega$ denotes boundary of Ω, q is a nonnegative constant and p is a real number. By using the five-point finite difference scheme to the diffusive terms and the central difference scheme to the convective terms with the equidistant step size $h = 1/(m + 1)$, we reach to the system of linear equations $Bx = d$, where B is matrix of order $n = m^2$ of the form

$$B = T_x \otimes I_m + I_m \otimes T_y + pI_n,$$

where $T_x = \text{tridiag}\{-1 - r, 4, -1 + r\}$ and $T_y = \text{tridiag}\{-1 - r, 0, -1 + r\}$, the constant $r = (qh)/2$ is called the mesh Reynolds number. Here we set $q = 0$ and report the results for solving (1) corresponding to the following two cases:

- Case I. Let $p = 0$ and $A = B + 5 * (B_L - B_L^T)$ where B_L is the strictly lower triangular part of B.

- Case II. Let $p = 1, 4$ and $A = B$. In this case, for $p = 4$, the problem reduces to the test example in [9, Example 3.1].

The right-hand side of (1) is constructed such that $x^* = (1, 2, \ldots, n)^T$ satisfies $Ax^* - |x^*| = b$.

All of the mentioned iterative schemes are convergent for Case I whereas $\nu \geq 1$. The reported results in Table 1 show that BBS-type iterations are superior to other examined iterative methods.

In Tables 2 and 3, we receptively report the results for $p = 1$ and $p = 4$ where $\nu < 1$. It should be pointed out that for $p = 1$ the value of ν becomes very close to one. When $p = 4$, all of examined iterative schemes are convergent and iterative scheme (5) outperforms other approaches. As seen, for both τ^* and $\tau_{exp}^{(2)}$, the iterative method (10) has a very close convergence speed to (5) for $p = 4$. Note that the iterations (3), (4) and (5) do not converge with respect to stopping criterion (30) when $p = 1$. For further details, we plot the convergence histories of iterations in Figure 2.

Iterative scheme (IS)	$m = 50$ ($\nu = 3.0026$) $(\alpha, \tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (4.72, 1.13, 1.1)$		$m = 70$ ($\nu = 4.1842$) $(\alpha, \tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (5.26, 1.25, 1.1)$		$m = 100$ ($\nu = 5.9701$) $(\alpha, \tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (4.51, 1.25, 1.1)$	
	Iter (CPU)	Err	Iter (CPU)	Err	Iter (CPU)	Err
IS (2) [12] (Version 1)	7(0.0532)	7.960e-16	9(0.1335)	8.588e-16	13(0.4216)	9.668e-16
IS (2) [12] (Version 2)	7(0.1105)	3.859e-15	9(0.4798)	4.991e-15	13(2.4785)	5.801e-15
IS (3) [17]	23(0.0314)	9.995e-09	29(0.1223)	4.480e-09	37(0.4637)	4.895e-09
IS (4) [19]	144(0.2237)	2.725e-09	198(0.2600)	4.367e-09	284(3.5866)	4.888e-09
IS (5) [9]	17(0.0246)	7.299e-09	21(0.0904)	7.955e-09	26(0.3299)	8.582e-09
Proposed method (with $\tau_{exp}^{(2)}$)	20(0.0283)	4.269e-09	24(0.1013)	5.440e-09	30(0.3776)	7.650e-09

Table 1: Numerical results for Example 3.1 in the case that $(q, p) = (0, 0)$.

Iterative scheme (IS)	$m = 50$ ($\nu = 0.9925$) $(\alpha, \tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (1, 2.11, 1.85)$		$m = 70$ ($\nu = 0.9961$) $(\alpha, \tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (1, 2.11, 1.89)$		$m = 100$ ($\nu = 0.9981$) $(\alpha, \tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (1, 2.1, 1.923)$	
	Iter (CPU)	Err	Iter (CPU)	Err	Iter (CPU)	Err
IS (2) [12] (Version 1)	6(0.0205)	2.143e-15	7(0.0592)	2.102e-15	7(0.1336)	2.477e-15
IS (2) [12] (Version 2)	6(0.0879)	4.150e-15	7(0.3483)	5.959e-15	7(1.2561)	8.174e-15
IS (3) [17]	†		†		†	
IS (4) [19]	†		†		†	
IS (5) [9]	895(0.2417)	9.931e-09	†		†	
Proposed method (with $\tau_{exp}^{(2)}$)	174(0.0474)	8.960e-09	237(0.2199)	9.733e-09	320(1.0915)	9.826e-09
Proposed method (with τ^*)	187(0.0535)	9.999e-09	259(0.2434)	9.611e-09	342(1.1655)	9.972e-09

Table 2: Numerical results for Example 3.1 in the case that $(q, p) = (0, 1)$.
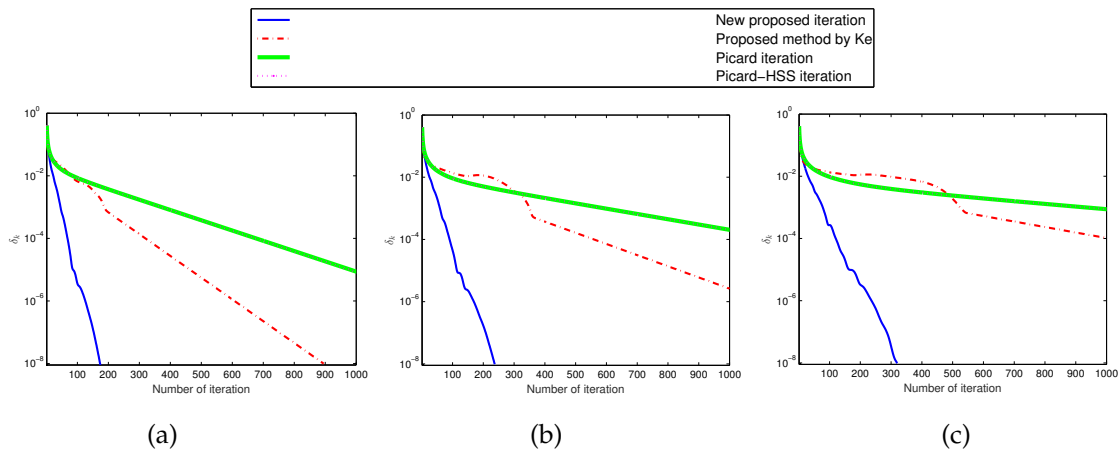

Figure 2: Example 3.1 Case II with $p = 1$: (a) $m = 50$, (b) $m = 70$, (c) $m = 100$.

In Example 3.2, we consider the case that A is full. To this end, we work with random matrices. The case, in which A is a random matrix, has been already examined in the literature, see [21] for instance. Algorithms were applied for several runs. Here the results corresponding to two runs are reported. We comment that the Picard iteration fails to converge and the Picard-HSS iteration does not work well which is expected as the matrix is not positive definite, hence we do not report the results for these iterative methods in Tables 4 and 5.

Example 3.2. In this example, we work with a randomly generated AVE problem in the following two cases where $Ax^* - |x^*| = b$ for $x^* = (1, 2, \ldots, n)^T$,

- Case I. Let $A =$ `rand(n,n)'*rand(n,n)+0.75*rand(n,n)`.

Iterative scheme (IS)	$m = 50$ ($\nu = 0.2495$) $(\alpha, \tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (4.03, 1.22, 1.11)$		$m = 70$ ($\nu = 0.2498$) $(\alpha, \tau_{exp}^{(1)}, {}_{exp}^{(2)}) = (3.99, 1.26, 1.08)$		$m = 100$ ($\nu = 0.2499$) $(\alpha, \tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (4, 1.26, 1.08)$	
	Iter (CPU)	Err	Iter (CPU)	Err	Iter (CPU)	Err
IS (2) [12] (Version 1)	3(0.0123)	6.109e-16	3(0.0319)	5.478e-16	3(0.0778)	5.525e-16
IS (2) [12] (Version 2)	3(0.0455)	6.494e-16	3(0.1558)	6.744e-16	3(0.5529)	6.907e-16
IS (3) [17]	14(0.0054)	3.173e-09	14(0.0147)	3.324e-09	14(0.0488)	3.441e-09
IS (4) [19]	24(0.0090)	3.220e-09	24(0.0138)	3.170e-09	23(0.0289)	3.375e-09
IS (5) [9]	8(0.0040)	6.400e-09	7(0.0089)	9.325e-09	7(0.0263)	8.439e-09
Proposed method (with $\tau_{exp}^{(2)}$)	10(0.0045)	6.508e-09	9(0.0107)	8.942e-09	9(0.0326)	7.837e-09
Proposed method (with τ^*)	11(0.0051)	2.135e-09	11(0.0161)	2.245e-09	11(0.0382)	2.350e-09

Table 3: Numerical results for Example 3.1 in the case that $(q, p) = (0, 4)$.

- Case II. Let $A =$`rand(n,n)'*rand(n,n)`$-0.75*$`rand(n,n)`.

We set $n = 2500$. The performance of mentioned iterative schemes are reported in Tables 4 and 5 for Cases I and II of Example 3.2. As seen, iterations (5) and (10) surpass other approaches. As pointed out earlier, the Picard iteration do not converge with respect to stopping criterion (30). For more details, we display the convergence histories of iterative schemes (2), (5) and (10) in Figures 3 and 4.

Iterative scheme (IS)	First run ($\nu = 100.4757$) $(\tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (0.64, 0.7)$			Second run ($\nu = 85.6176$) $(\tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (0.62, 0.74)$		
	Iter	CPU	Err	Iter	CPU	Err
IS (2) [12] (Version 1)	9	1.6424	2.164e-09	4	0.7114	2.493e-09
IS (2) [12] (Version 2)	9	1.8408	2.164e-09	4	0.7863	2.493e-09
IS (5) [9]	23	0.8902	7.801e-09	8	0.3255	9.673e-09
Proposed method (with $\tau_{exp}^{(2)}$)	19	0.7184	8.913e-09	9	0.3487	8.0157e-09

Table 4: Numerical results for two runs of Case I in Example 3.2.

Iterative scheme (IS)	First run ($\nu = 74.5402$) $(\tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (0.64, 0.65)$			Second run ($\nu = 41.4888$) $(\tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (0.54, 0.65)$		
	Iter	CPU	Err	Iter	CPU	Err
IS (2) [12] (Version 1)	30	5.1694	8.949e-10	7	1.2119	1.051e-10
IS (2) [12] (Version 2)	30	5.5358	8.949e-10	7	1.3867	1.051e-10
IS (5) [9]	12	0.4650	9.359e-09	8	0.3087	9.278e-09
Proposed method (with $\tau_{exp}^{(2)}$)	11	0.4242	9.831e-09	7	0.2747	7.037e-09

Table 5: Numerical results for two runs of Case II in Example 3.2.

As numerically observed, the value of τ^* can provide a good approximation of optimal parameter for the proposed iterative method when $\nu < 1$. Hence, we only report the performance of proposed method with τ^* for the following two examples.

The matrix A in Example 3.3 is a special case of B in Example 3.1. Indeed, in [7, 9] a reformulation of the matrix B is used for which $r = 0$. Our goals for presenting this test problem are working with larger dimensions than those examined in the first example and presenting comparison results between BBS-type iterative methods.
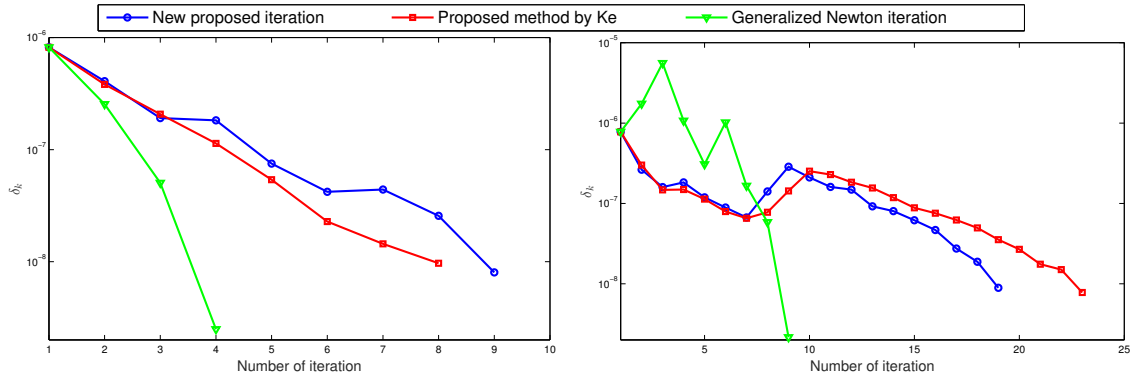
Figure 3: Case I in Example 3.2 with $n = 2500$; $(\tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (0.62, 0.74)$ (Left) and $(\tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (0.64, 0.7)$ (Right).
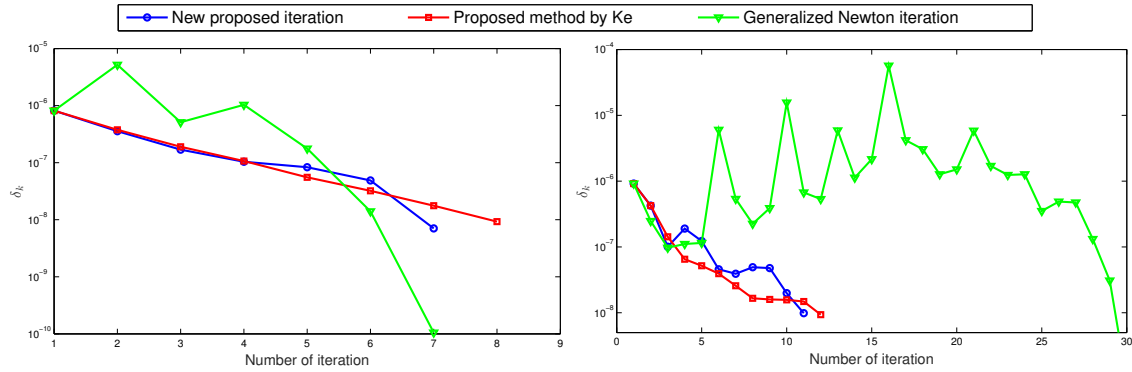

Figure 4: Case II in Example 3.2 with $n = 2500$; $(\tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (0.54, 0.65)$ (Left) and $(\tau_{exp}^{(1)}, \tau_{exp}^{(2)}) = (0.64, 0.65)$ (Right).

Example 3.3. [7, 9] Let m be a specified positive integer and $n = m^2$. Consider the AVE (1) with $A = M + \mu I_n \in \mathbb{R}^{n \times n}$ where $M = S \otimes I_m + I_m \otimes S \in \mathbb{R}^{n \times n}$ and $S = \text{tridiag}\{-1, 4, -1\} \in \mathbb{R}^{m \times m}$.

The numerical results for Example 3.3 are disclosed in Tables 6, 7 and 8. The results in Table 6 are reported to illustrate the behavior of SOR-like method in comparison with iterative scheme (10). To this end, we only work with the dimensions used in [7, Example 2] and the right-hand side b is constructed such that $x^* = (-1, 1, -1, 1, \ldots, -1, 1)^T$ satisfies in Eq. (1) for Table 6. Results in Tables 7 and 8 are associated with the case $Ax^* - |x^*| = b$ for $x^* = (1, 2, \ldots, n)^T$.

In the reported numerical results, Guo et al. [7] approximated ρ in the expression for ω_o (cf. (8)) by $\rho(A^{-1})$. Therefore, the values of ω_o in [7, Table 2] are exactly equal to τ^*. In fact, $\rho(A^{-1}) = \|A^{-1}\|$ since the matrix A is symmetric.

Although the iterative method (5) outperforms other examined methods in Example 3.3, the optimum value of its parameter was found experimentally. For more details on the effect of parameters on the convergence speed of iterations, we depict the performance of iterative methods (4) and (5) in terms of both required CPU-time and number of iterations for the convergence in Figure 5.

The following example was originally used in [1]. Basically, if we set $p = 0$ then the test problem is reduced to the one examined in [1, Subsection 6.1]. For the following example, the Matlab backslash "\" was exploited in all iterations for applying the inverse of A.

Example 3.4. [1] Consider the ordinary differential equation

$$\ddot{x}(t) - |x(t)| + px(t) = 0, \qquad x(0) = x_0, \qquad \dot{x}(0) = \gamma,$$

Iterative scheme	$n = 64$ ($\nu = 0.2358$) $\tau^{(1)}_{exp} = 0.97$		$n = 256$ ($\nu = 0.2458$) $\tau^{(1)}_{exp} = 0.97$		$n = 1024$ ($\nu = 0.2489$) $\tau^{(1)}_{exp} = 0.97$		$n = 4096$ ($\nu = 0.2497$) $\tau^{(1)}_{exp} = 0.97$	
	Iter (CPU)	Err	Iter (CPU)	Err	Iter (CPU)	Err	Iter (CPU)	Err
IS (2) (Version 1)	2(0.0019)	2.729e-16	2(0.0025)	2.854e-16	2(0.0043)	3.076e-16	2(0.0186)	3.269e-16
IS (2) (Version 2)	2(0.0022)	2.218e-16	2(0.0030)	3.515e-16	2(0.0089)	3.256e-16	2(0.0808)	3.386e-16
IS (5) [9]	11(0.0018)	6.806e-09	11(0.0020)	8.216e-09	11(0.0028)	8.917e-09	11(0.0094)	9.267e-09
SOR-like (with ω_o)	16(0.0021)	4.941e-09	16(0.0023)	8.512e-09	17(0.0035)	3.384e-09	17(0.0132)	3.649e-09
Proposed method (with τ^*)	13(0.0018)	3.020e-09	13(0.0020)	4.291e-09	13(0.0028)	4.901e-09	13(0.0103)	5.174e-09

Table 6: Numerical results for Example 3.3 with $\mu = 4$.

Iterative scheme (IS)	$n = 40000$ ($\nu = 0.2500$) $(\alpha, \tau^{(1)}_{exp}) = (3.99, 1.26)$		$n = 90000$ ($\nu = 0.2500$) $(\alpha, \tau^{(1)}_{exp}) = (3.98, 1.26)$		$n = 160000$ ($\nu = 0.2500$) $(\alpha, \tau^{(1)}_{exp}) = (3.98, 1.26)$	
	Iter (CPU)	Err	Iter (CPU)	Err	Iter (CPU)	Err
IS (2) [12] (Version 1)	3(0.8049)	5.743e-16	3(3.6506)	5.746e-16	3(11.087)	5.910e-16
IS (2) [12] (Version 2)	3(7.2022)	6.978e-16	3(43.903)	6.842e-16	3(170.85)	7.042e-16
IS (3) [17]	14(0.3722)	3.581e-09	14(1.1577)	3.628e-09	14(2.6237)	3.652e-09
IS (4) [19]	20(0.1092)	3.425e-09	19(0.2742)	3.355e-09	18(0.6322)	3.358e-09
IS (5) [9]	7(0.1941)	7.559e-09	7(0.5924)	7.314e-09	7(1.2844)	7.203e-09
Proposed method (with τ^*)	11(0.2890)	2.477e-09	11(0.9022)	2.525e-09	11(2.0301)	2.550e-09

Table 7: Numerical results for Example 3.3 with $\mu = 4$.

Iterative scheme (IS)	$n = 40000$ ($\nu = 0.1250$) $(\alpha, \tau^{(1)}_{exp}) = (7.97, 1.13)$		$n = 90000$ ($\nu = 0.1250$) $(\alpha, \tau^{(1)}_{exp}) = (7.97, 1.13)$		$n = 160000$ ($\nu = 0.1250$) $(\alpha, \tau^{(1)}_{exp}) = (7.93, 1.13)$	
	Iter (CPU)	Err	Iter (CPU)	Err	Iter (CPU)	Err
IS (2) [12] (Version 1)	3(0.8050)	4.543e-16	3(3.6786)	4.491e-16	3(11.118)	4.582e-16
IS (2) [12] (Version 2)	3(8.4123)	7.919e-16	3(56.025)	7.822e-16	3(223.94)	8.044e-16
IS (3) [17]	9(0.2480)	7.318e-09	9(0.7003)	7.362e-09	9(2.1313)	7.384e-09
IS (4) [19]	11(0.0634)	6.537e-09	9(0.1294)	6.435e-09	9(0.3412)	5.510e-09
IS (5) [9]	5(0.1421)	3.514e-09	5(0.3980)	3.069e-09	5(1.2014)	2.830e-09
Proposed method (with τ^*)	8(0.2078)	2.480e-09	8(0.6317)	2.503e-09	8(1.9364)	2.514e-09

Table 8: Numerical results for Example 3.3 with $\mu = 8$.

on domain $t \in [0, 4]$. Using a finite difference scheme for discretizing the above equation lead to an AVE in the form (1). Basically, one can apply the following second–order backward difference to approximate the second derivative

$$\frac{x_{i-2} - 2x_{i-1} + x_i}{h^2} - |x_i| + px_i = 0,$$

where $x_i = ih$, $i = 1, 2, \ldots, n$. Here, h denotes the step-size and $x_1, x_2, \ldots, x_n$ are equispaced points. Neumann boundary conditions were approximated applying a centered finite difference scheme

$$\frac{x_{-1} - x_1}{2h} = \gamma.$$

Similar to [1], the initial conditions are $x_0 = -1$ and $\gamma = 1$.

The results in Table 9 show the effectiveness of BBS-type iterative methods. Notice that the proposed method (with τ^*) is compatible with iterative method (5) while an experimentally obtained optimum parameter is used for (5).
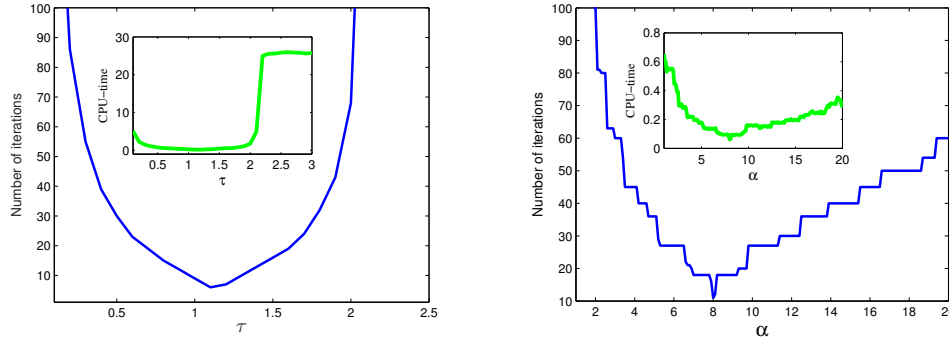
Figure 5: Example 3.3 with $\mu = 8$ and $n = 40000$ for iterative method (5) (left) and Picard-HSS (right); Outside figure: number of iterations versus the parameter; Inside figure: CPU-time versus the parameter.

Iterative scheme (IS)	$n = 40000$ ($v = 0.4373$) $\tau_{exp}^{(1)} = 0.84$		$n = 90000$ ($v = 0.4375$) $\tau_{exp}^{(1)} = 0.85$		$n = 160000$ ($v = 0.4376$) $\tau_{exp}^{(1)} = 0.7$	
	Iter (CPU)	Err	Iter (CPU)	Err	Iter (CPU)	Err
IS (2) [12]	3(0.6792)	4.719e-15	2(3.2719)	4.803e-09	2(10.3661)	2.026e-09
IS (5) [9]	4(0.0056)	9.615e-09	3(0.0084)	9.425e-09	3(0.0164)	9.766e-09
Proposed method (with τ^*)	4(0.0051)	8.447e-09	3(0.0077)	9.354e-09	3(0.0164)	3.947e-09

Table 9: Numerical results for Example 3.4.

4. Conclusions

The absolute value equation $Ax - |x| = b$ was reformulated by $\mathcal{A}\mathbf{z} = \mathbf{b}$ where

$$\mathcal{A} = \begin{bmatrix} A & -I \\ -D(x) & I \end{bmatrix}.$$

The performance of iterative methods, extracted from block splittings of $\mathcal{A}$ and its preconditioned form, were studied. The methods rely on a fixed parameter τ and convergence intervals were derived for the parameter. Numerical experiments were reported for some available test problems in the literature to show the effectiveness of iterative schemes induced by block splittings in comparison to generalized Newton and Picrd(-HSS) iterative methods.

Alternative possible block splittings can be mentioned for $\mathcal{A}$ and its preconditioned forms which result new iterative schemes. Also inexact solvers can be implemented instead of direct ones for the act of A^{-1} in iterative methods (5) and (10) which consists of using different splittings of A and using few steps of a stationary iterative method like the idea exploited for constructing the Picard-HSS iteration. Following the idea used in [6], one can also try using principle of hierarchical identification and the HSS splitting to replace an iterative scheme for approximating A^{-1} in iterative schemes extracted from block splittings.

Acknowledgments

References

[1] L. Abdallah, M. Haddou and T. Migot, Solving absolute value equation using complementarity and smoothing functions, *Journal of Computational and Applied Mathematics*, 327 (2018) 196–207.
[2] Z. Z. Bai, G.H. Golub and M.K. Ng, Hermitian and skew-Hermitian splitting methods for non–Hermitian positive definite linear systems, *SIAM Journal on Matrix Analysis and Applications*, 24 (2003) 603–626.
[3] Z. Z. Bai and X. Yang, On HSS–based iteration methods for weakly nonlinear systems, *Applied Numerical Mathematics*, 59 (2009) 2923–2936.
[4] A. Berman and R. Plemmons, Nonnegative Matrices in the Mathematical Sciences, Academic Press, New York, 1979.
[5] L. Caccetta, B. Qu and G. Zhou, A globally and quadratically convergent method for absolute value equations, *Computational Optimization and Applications*, 48 (2011) 45–58.
[6] M. Dehghan and H. Hajarian, SSHI methods for solving generallinear matrix equations, *Engineering Computations*, 28 (2018) 1028–1043.
[7] P. Guo, S. L. Wu and C. X. Li, On the SOR–like iteration method for solving absolute value equations, *Applied Mathematics Letters*, 97 (2019) 107–113.
[8] Y. F. Ke and C. F. Ma, On the convergence analysis of two-step modulus-based matrix splitting iteration method for linear complementarity problems, *Applied Mathematics and Computation*, 243 (2014) 413–418.
[9] Y. F. Ke, The new iteration algorithm for absolute value equation, *Applied Mathematics Letters*, 99 (2020), 105990.
[10] O. L. Mangasarian and R. R. Meyer, Absolute value equations, *Linear Algebra and its Applications*, 419 (2006) 359–367.
[11] O. L. Mangasarian, Absolute value equation solution via concave minimization. *Optimization Letter*, 1 (2007) 3–8.
[12] O. L. Mangasarian, A generalized Newton method for absolute value equations, *Optimization Letter*, 3 (2009) 101–108.
[13] A. Mansoori and M. Erfanian, A dynamic model to solve the absolute value equations, *Journal of Computational and Applied Mathematics*, 333 (2018) 28–35.
[14] M. A. Noor, J. Iqbal, K. I. Noor and E. Al-Said, On an iterative method for solving absolute value equations, *Optimization Letter*, 6 (2012) 1027–1033.
[15] J. M. Ortega, W. C. Rheinbolt, Iterative solution of nonlinear equations in several variables, Academic Press, New York, 1970.
[16] H. Ren, X. Wang, X.-B. Tang and T. Wang, The general two-sweep modulus-based matrix splitting iteration method for solving linear complementarity problems, *Computers & Mathematics with Applications*, 77 (2019) 1071–1081.
[17] J. Rohn, V. Hooshyarbakhsh and R. Farhadsefat, An iterative method for solving absolute value equations and sufficient conditions for unique solvability, *Optimization Letter*, 8 (2014) 35–44.
[18] Y. Saad, Iterative methods for sparse linear systems, Second Edition. Society for Industrial and Applied Mathematics, Philadelphia, 2003.
[19] D. K. Salkuyeh, The Picard-HSS iteration method for absolute value equations, *Optimization Letter*, 8 (2014) 2191-2202.
[20] X. Wang, X. Li, L. H. Zhang and R. C. Li, An efficient numerical method for the symmetric positive definite second-order cone linear complementarity problem, *Journal of Scientific Computing*, 79 (2019) 1608–1629.
[21] L. Yong, Iterative method for absolute value equation and applications in two-point boundary value problem of linear differentiation equation, *Journal of Interdisciplinary Mathematics*, 18 (2015) 355–374.
[22] D. M. Young, Iterative Solution for Large Linear Systems, Academic Press, New York, 1971.
[23] C. Zhang and Q. J. Wei, Global and finite convergence of a generalized Newton method for absolute value equations, *Journal of Optimization Theory and Applications*, 143 (2009) 391–403.