U.P.B. Sci. Bull., Series A, Vol. 78, Iss. 1, 2016 ISSN 1223-7027

A FAST CONVERGENT TWO-STEP ITERATIVE METHOD TO SOLVE THE ABSOLUTE VALUE EQUATION

Hamid Esmaeili[1], Mahdi Mirzapour[2], Ebrahim Mahmoodabadi[3]

In this paper, we propose a fast convergent two-step iterative algorithm to solve the NP-hard absolute value equation when the solution is unique. Our method is based on fixed point method in first step and modification of the generalized Newton method introduced by Mangasarian in second step. It is proved that the proposed algorithm has all of the properties of the generalized Newton method while converges faster than it. Especially, our wide numerical experiments showed that our algorithm can solve much more problems with an accuracy of 10^{-11}*, whereas the generalized Newton method may fail.*

Keywords: Absolute value equation, Generalized Newton method, Iterative method, Two-step iterative algorithm.

MSC2010: 90C05, 90C30, 15A06, 15A15, 49M15.

1. Introduction

In this article, we consider the absolute value equation (AVE):

$$Ax - |x| - b = 0, \tag{1}$$

in which $A \in R^{n\times n}$ and $b \in R^n$ are given, and $|\cdot|$ denotes absolute value. The significance of the absolute value equation (1) arises from the fact that the absolute value equation (1) is equivalent to the linear complementarity problem [7,8]. This equivalence formulation has been used by Mangasarian [7,8] to solve the absolute value equation using the linear complementarity problems. As we know, linear programs, quadratic programs, bimatrix games and other problems can all be reduced to a linear complementarity problem [3,4]. A more general form of the AVE, $Ax + B|x| = b$, was introduced in [13] and investigated in a more general context in [5]. Also, the AVE (1) was investigated in detail theoretically in [8] and a bilinear program was prescribed there for the especial case when the singular values of A are not less than one. As was shown in [8], the general NP-hard linear complementarity problem [2,3,4] can be formulated as an AVE (1). This implies that (1) is NP-hard in its general form. In order to solve the AVE (1), Mangasarian [7] proposed the generalized Newton method that is convergent when the singular values of A exceed 1. But, this method may fail to convergence when the accuracy increases. In Sect. 2 of the present work, we introduce a two-step iterative algorithm and prove that it has all of the properties of the generalized Newton method. Effectiveness of proposed algorithm, with respect to the generalized Newton method, is demonstrated in Sect. 3 by solving 800 random solvable AVEs of the size $n = 100, 200, 500, 1000$. Each AVE is solved to an accuracy of 10^{-11}. Numerical

[1,2,3]Department of Mathematics, Bu-Ali Sina University, Hamedan, Iran.
E-mail: {[1]esmaeili,[2]m.mirzapour,[3]mahmoodabadi}@basu.ac.ir
[2]Corresponding Author: E-mail: mahdimirzapour67@gmail.com

results showed that the our two-step iterative algorithm solves much more problems successfully, whereas the generalized Newton method may fail in many cases, especially when n is large. Moreover, the average time taken by computer for each problem in the two-step iterative algorithm is less than that of the generalized Newton method. Also we compared our algorithm with residual iterative method [9] and Yong method [15] and we saw that our algorithm is faster than mentioned methods.

2. **The two-step iterative algorithm**

By defining the function $f(x)$ as follows:

$$f(x) = Ax - |x| - b, \tag{2}$$

we can write the AVE problem (1) in the following form:

$$f(x) = 0. \tag{3}$$

Notice that $|x|$ is not differentiable. A generalized Jacobian $\partial|x|$ of $|x|$ based on a subgradient [11,12] of its components is given by the diagonal matrix $D(x)$:

$$\partial|x| = D(x) = diag(sign(x)), \tag{4}$$

in which $sign(x)$ denotes a vector with components equal to 1, 0 or -1, depending on whether the corresponding component of x is positive, zero or negative. To solve the equation (3), Mangasarian [7] used the Newton method with a generalized Jacobian $\partial f(x)$ of $f(x)$ defined by:

$$\partial f(x) = A - D(x). \tag{5}$$

The generalized Newton method for finding a zero of the equation $f(x) = 0$ consists of the following iteration:

$$f(x^k) + \partial f(x^k)(x^{k+1} - x^k) = 0, \tag{6}$$

or

$$x^{k+1} = x^k - (A - D(x^k))^{-1} f(x^k). \tag{7}$$

Noting that $D(x^k)x^k = |x^k|$, we can simplify the generalized Newton iteration (7) to solve the AVE (1) in the following simple form:

$$x^{k+1} = (A - D(x^k))^{-1} b. \tag{8}$$

It is shown [7] that if the singular values of A exceed 1, then the generalized Newton method (8) is well defined and bounded. Consequently, there exists an accumulation point $\bar{x}$ such that $(A - D(\bar{x}))\bar{x} = b$, meaning that the $\bar{x}$ is a solution of the AVE (1). Furthermore, under the assumption that $\|A^{-1}\| < 1/4$ and $D(x^k) \neq 0$, for all k, the generalized Newton iteration (8) converges linearly to the unique solution x^* of the AVE (1) from any starting point. However, under stopping criterion $\|x^{k+1} - x^k\| < \varepsilon$, the generalized Newton method may fail when $\varepsilon < 10^{-6}$, say 10^{-11}, and $n \geq 100$. Now we use the following two-step iterative algorithm to solving AVE (1).

Algorithm 2.1. **Two-Step Iterative Algorithm(TSIA)**
Initializing:

choose a stopping criterion $\varepsilon > 0$, $x^0 \in R^n$, $\varepsilon < \alpha$ *and set* $k = 0$

First Step :

$$while\ \|f(x^k)\| > \alpha$$
$$x^{k+1} = x^k - A^{-1}f(x^k)$$

Second Step :

$$while\ \|f(x^k)\| > \varepsilon$$
$$y^k = x^k - A^{-1}f(x^k)$$
$$D(y^k) = diag(sign(y^k))$$
$$x^{k+1} = x^k - (A - D(y^k))^{-1}f(x^k)$$

In the following we studied the main property of the Algorithm 2.1, and the convergence of two steps.
First, we give the following lemma.

Lemma 2.1. *[7,1]* Let x and y be points in R^n. Then:

$$\||x| - |y|\| \leq 2\|x - y\| \tag{9}$$

Proposition 2.1. *Under the assumption that $\|A^{-1}\| < \frac{1}{2}$ the first step iteration of Algorithm 2.1 converges linearly from any starting point to a solution x^* for any solvable AVE (1).*

Proof. By using $f(x^k) = Ax^k - |x^k| - b$ in first step iteration of Algorithm 2.1 we have:

$$x^{k+1} = A^{-1}(|x^k| + b)$$

On the other hand, $Ax^* - |x^*| = b$. So,

$$x^{k+1} - x^* = A^{-1}(|x^k| - |x^*|)$$

Now, using the Lemma 2.1 we have:

$$\|x^{k+1} - x^*\| \leq \|A^{-1}\|(2\|x^k - x^*\|) \Longrightarrow \frac{\|x^{k+1} - x^*\|}{\|x^k - x^*\|} < 1$$

Hence the sequence $\{\|x^k - x^*\|\}$ converges linearly to zero and x^k converges linearly to x^*. □

Also Rohn and et al. in [14,Theorem 3] showed that the first step iteration of Algorithm 2.1 by assumption $x^0 = A^{-1}b$ and $\|A^{-1}\| < 1$ converges linearly to the unique solution x^* of AVE (1).
Now we are ready to prove main property and convergence of second step iteration of Algorithm 2.1.
In second step we use the following modified generalized Newton method to solve the AVE (1):

$$\begin{aligned} y^k &= x^k - A^{-1}f(x^k), \\ x^{k+1} &= x^k - (A - D(y^k))^{-1}f(x^k). \end{aligned} \tag{10}$$

Using $f(x^k) = Ax^k - |x^k| - b$, we can simplify the iteration (10) as follows:

$$\begin{aligned} y^k &= A^{-1}\left(|x^k| + b\right), \\ x^{k+1} &= (A - D(y^k))^{-1}\left(|x^k| - D(y^k)x^k + b\right). \end{aligned} \tag{11}$$

To establish convergence of the iteration (10), we need a few theoretical results from the literature.

Lemma 2.2. *[7] If the singular values of $A \in R^{n\times n}$ exceed 1, then $(A-D)^{-1}$ exists for any diagonal matrix D whose diagonal elements equal ± 1 or 0.*

As mentioned before, our aim is to prove that the modified generalized Newton method (10) in second step of Algorithm 2.1 has all of the good properties of the generalized Newton method (8) discussed in [7]. Therefore, we now establish boundedness of the iterates of (10) and hence the existence of an accumulation point for them.

Proposition 2.2. *Let the singular values of A exceed 1. Then, the iterates (10) of the modified generalized Newton method in second step of Algorithm 2.1 are well defined and bounded. Consequently, there exists an accumulation point $\bar{x}$ such that $(A - \overline{D})\bar{x} = b$ for some diagonal matrix $\overline{D}$ with diagonal elements of ± 1 or 0.*

Proof. Using Lemma 2.2, we notice that the matrices $(A - D(x^k))^{-1}$ and $(A - D(y^k))^{-1}$ exist and the modified generalized Newton iteration (10) is well defined. Suppose now that the sequence $\{x^k\}$ is unbounded. From the finite number of possible configurations for $D(x^k)$ in the sequence $\{D(x^k)\}$, there exists a subsequence $\{x^{k_i}\}$, $\|x^{k_i}\| \to \infty$, such that $D(x^{k_i}) = \tilde{D}$ is a fixed diagonal matrix with diagonal elements equal to ± 1 or 0. This means that the bounded subsequence $\{x^{k_i}/\|x^{k_i}\|\}$ converges to a point $\tilde{x}$ such that $\|\tilde{x}\| = 1$ and $D(\tilde{x}) = \tilde{D}$. Hence, using (11), we have

$$\lim_{i\to\infty} \frac{y^{k_i}}{\|x^{k_i}\|} = \frac{1}{2}\lim_{i\to\infty} \frac{x^{k_i} + (A-\tilde{D})^{-1}b}{\|x^{k_i}\|} = \frac{\tilde{x}}{2},$$

so $D(y^{k_i}) = \tilde{D}$, too. Again, using (11), we obtain

$$\begin{aligned} \lim_{i\to\infty}(A-\tilde{D})\frac{x^{k_i+1}}{\|x^{k_i+1}\|} &= (A-\tilde{D})\lim_{i\to\infty}\frac{(A-D(y^{k_i}))^{-1}\left(|x^{k_i}| - D(y^{k_i})x^{k_i} + b\right)}{\|x^{k_i+1}\|} \\ &= \lim_{i\to\infty}\frac{b}{\|x^{k_i+1}\|} = 0. \end{aligned}$$

In summary, there exists a $\tilde{x} \neq 0$ such that $(A-\tilde{D})\tilde{x} = 0$ which is a contradiction to Lemma 2.2. Consequently, the sequence $\{x^k\}$ is bounded and there exists an accumulation point $\bar{x}$ of $\{x^k\}$ such that $f(\bar{x}) = 0$. □

Proposition 2.3. *Let the singular values of A exceed 1. Consider the sequence $\{x^k\}$ generated by the modified generalized Newton iteration (10) in second step of Algorithm 2.1. If $x^{k+1} = x^k$ for some k, then x^{k+1} solves the AVE (1).*

Proof. According to Lemma 2.1, the modified generalized Newton iteration (10) is well defined. If $x^{k+1} = x^k$, then from (11), we have

$$\begin{aligned} &Ax^{k+1} - D(y^k)x^{k+1} = |x^k| - D(y^k)x^k + b \qquad \Longrightarrow \\ &Ax^{k+1} - D(y^k)x^k = |x^k| - D(y^k)x^k + b \qquad \Longrightarrow \\ &Ax^{k+1} = |x^k| + b \qquad \Longrightarrow \\ &Ax^{k+1} - |x^{k+1}| - b = 0. \end{aligned}$$

This shows that x^{k+1} solves the AVE (1). $\square$

Lemma 2.3. *Under the assumption that $\|(A-D)^{-1}\| < 1/3$ for any diagonal matrix D with diagonal elements of ± 1 or 0, the modified generalized Newton iteration (10) converges linearly from any starting point to a solution x^* for any solvable AVE (1).*

Proof. According to (10), we have

$$(A - D(y^k))x^{k+1} = |x^k| - D(y^k)x^k + b.$$

On the other hand, $Ax^* - |x^*| - b = 0$. So,

$$(A - D(y^k))(x^{k+1} - x^*) = |x^k| - |x^*| - D(y^k)(x^k - x^*),$$

that results in

$$x^{k+1} - x^* = (A - D(y^k))^{-1}\left(|x^k| - |x^*| - D(y^k)(x^k - x^*)\right).$$

Now, using Lemma 2.1 and noticing that $\|D(y^k)\| = 1$, we can write

$$\|x^{k+1} - x^*\| \leq 3\|(A - D(y^k))^{-1}\|\,\|x^k - x^*\| < \|x^k - x^*\|.$$

Therefore, the sequence $\{\|x^k - x^*\|\}$ converges linearly to zero and $\{x^k\}$ converges linearly to x^*. $\square$

We are now ready to prove our final result.

Proposition 2.4. *Let $\|A^{-1}\| < 1/4$ and $D(x^k) \neq 0$, for all k. Then, the AVE (1) is uniquely solvable for any b and the modified generalized Newton iteration (10) in second step of Algorithm 2.1 is well defined and converges linearly to the unique solution of the AVE (1) from any starting point x^0.*

Proof. The unique solvability of the AVE (1) for any b is resulted from [8, Proposition 4] which requires that $\|A^{-1}\| < 1$. By the Banach perturbation lemma [10, p. 45], $\|(A - D(x^k))^{-1}\|$ exists for any x^k since A^{-1} exists and $\|A^{-1}\|\,\|D(x^k)\| < 1$. The same lemma also suggests that:

$$\|(A - D(x^k))^{-1}\| \leq \frac{\|A^{-1}\|\,\|D(x^k)\|}{1 - \|A^{-1}\|\,\|D(x^k)\|} < \frac{1}{3}.$$

Hence, Lemma 2.3 denotes that the sequence $\{x^k\}$ converges linearly to the unique solution of the AVE (1) from any starting point x^0. $\square$

Remark 2.1. *There is a similar result to Proposition 2.4 in [7] that indicates the generalized Newton method (8) is well defined and converges linearly to the unique solution of the AVE (1) from any starting point x^0, when $\|A^{-1}\| < 1/4$. Moreover, our proof is the same as in [7].*

3. **Computational results**

To illustrate the implementation and efficiency of the proposed (TSIA), we consider the following examples. All the experiments are performed with Intel(R) Core(TM)2 Due CPU 1.8 GHz, 1GB RAM, and the codes are written in Matlab 7.1.

Example 3.1. *In this example we compare (TSIA) with the generalized Newton method (8) (GNM) on a random solvable AVE with fully dense matrix* $A \in R^{n\times n}$, $n = 100, 200, 500, 1000$. *For any dimension n, we generated 100 random As from a uniform distribution on* $[-10, 10]$. *We ensured that the singular values of each A exceeded 1 by actually computing the minimum singular value and rescaling A by dividing it by the minimum singular value multiplied by a random number in the interval* $[0, 1]$. *Then, we chose a random x from a uniform distribution on* $[-1, 1]$ *and computed* $b = Ax - |x|$. *also we chose* $\alpha = 10^{-2}$. *Our stopping criterion for the (TSIA) and (GNM) is* $\|Ax - |x| - b\| < \varepsilon$ *in which* $\varepsilon = 10^{-6}$ *or* 10^{-11}. *Therefore, overall 800 solvable AVEs is solved.*
We divide our numerical experiments in two categories. One of those is corresponding to the precision $\varepsilon = 10^{-6}$ *as summarized in Table 1. The other one is corresponding to the precision* $\varepsilon = 10^{-11}$ *as summarized in Table 2. In these tables, NS, AV, INV and TOC denote the number of solved problems, the average number of iterations for solved problems, the number of computed inverse of matrix and also total solved AVEs time taken by CPU, respectively.*
As Table 1 shows, Algorithm 2.1 and the generalized Newton method (8) have almost similar behavior in the case of $\varepsilon = 10^{-6}$ *and both of them solved all of AVEs but our Algorithm 2.1 compute approximately half inverse of matrices compare to GNM (8), So the computed time for the Algorithm 2.1 is less than half the computation time for the GNM (8).*
The difference will be clearer in Table 2 for any $n \geq 100$. *We notice that the generalized Newton method can not solve* 24% *problems of dimension* $n = 500$ *and* 70% *problems of dimension n = 1000 Whereas proposed Algorithm 2.1 can solve all of the problems. Furthermore, for* $n = 100, 200, 500, 1000$ *same as table 1 our Algorithm 2.1 can solve all of problems with a small number of computations and hence converges faster than GNM (8) to uniqe solution of AVE.*
furthermore if we apply the generalized Newton method (8) in second step of proposed Algorithm 2.1 and solve above 800 random solvable AVE, and compare with GNM (8) in number of solved AVEs, we see they have same result.

Table 1. *Numerical results for precision* $\varepsilon = 10^{-6}$ *and* $\alpha = 0.01$

	100		200		500		1000	
	GNM	*TSIA*	*GNM*	*TSIA*	*GNM*	*TSIA*	*GNM*	*TSIA*
NS	100	100	100	100	100	100	100	100
AV	4.27	8.03	4.49	8.19	4.97	8.90	5.12	9.01
INV	427	209	449	206	497	210	512	218
TOC	0.473	0.249	2.465	1.150	34.26	14.91	249.4	107.4

Table 2. *Numerical results for precision* $\varepsilon = 10^{-11}$ *and* $\alpha = 0.01$

	100		200		500		1000	
	GNM	*TSIA*	*GNM*	*TSIA*	*GNM*	*TSIA*	*GNM*	*TSIA*
NS	98	100	98	100	76	100	30	100
AV	4.14	7.89	4.60	8.25	4.93	8.90	5.1	9.06
INV	406	202	451	207	375	210	153	201
TOC	0.459	0.240	2.214	1.125	25.28	15.03	74.51	101.5

Example 3.2. *(see [9,15]).Consider random matrix A and b in Matlab code as*

$$n = input(dimension\ of\ matrix\ A\ =);$$

$$rand(state, 0);$$

$$R = rand(n, n);$$

$$b = rand(n, 1);$$

$$A = R' * R + n * eye(n);$$

with random initial guess. The comparison between Algorithm 2.1, the residual iterative method [9] and the Yong method [15] is presented in Table 3.
In Table 3 *TOC denotes time taken by CPU. For any order of n we solve* 100 *randomly generated AVEs based on obove codes and solve them using Algorithm 2.1. The average times taken by CPU for every order of n are presented in Table* 3*. Note that for any size of dimension n Algorithm 2.1 converges faster than both the residual iterative method [9] and Yong method [15].*

Table 3. *Numerical results for Example 3.2.*

order	*Residual Iterative Method[9]*		*Yong method[15]*		*Algorithm 2.1*	
	No. of iter.	*TOC*	*No. of iter.*	*TOC*	*No. of iter.*	*TOC*
4	*2*	*0.006*	*2*	*2.230*	*3*	*0.00050*
8	*2*	*0.022*	*2*	*3.340*	*3*	*0.00065*
16	*2*	*0.025*	*3*	*3.790*	*3*	*0.00078*
32	*2*	*0.053*	*2*	*4.120*	*3*	*0.00101*
64	*2*	*0.075*	*3*	*6.690*	*3*	*0.01224*
128	*2*	*0.142*	*3*	*12.450*	*3*	*0.04209*
256	*2*	*0.201*	*3*	*34.670*	*3*	*0.06714*
512	*3*	*1.436*	*5*	*79.570*	*3*	*0.32896*
1024	*3*	*6.604*	*5*	*157.12*	*3*	*0.83559*

4. **Conclusion**

We have proposed a fast linearly convergent algorithm for solving the NP-hard absolute value equation $Ax - |x| = b$ under certain assumptions on A. We proved that our method has all of the properties of the generalized Newton method. Numerical experiments showed that, in terms of the number of successfully solved problems and time of successfully solved problems, proposed Algorithm 2.1 works better than the generalized Newton method (8), Residual Iterative Method [9] and Young method [15]. Especially when high accuracy is needed and n is large.

REFERENCES

[1] *D.P. Bertsekas*, Nonlinear Programming, Athena Scientific, Belmont, 1995.
[2] *S.J. Chung*, NP-completeness of the linear complementarity problem, J. Optim. Theory Appl., **60**(1989), No. 3, 393-399.
[3] *R.W. Cottle and G. Dantzig*, Complementary pivot theory of mathematical programming, Linear Algebra Appl., **1**(1986), No. 1, 103-125.
[4] *R.W. Cottle, J.S. Pang and R.E. Stone*, The Linear Complementarity Problem, Academic Press, New York, 1992.
[5] *O.L. Mangasarian*, Absolute value programming, Comput. Optim. Appl., **36**(2007), No. 1, 43-53.
[6] *O.L. Mangasarian*, Absolute value equation solution via concave minimization, Optim. Lett., **1**(2007), No. 1, 3-8.

[7] *O.L. Mangasarian*, A generalized Newton method for absolute value equations, Optim. Lett., **3**(2009), No. 1, 101-108.

[8] *O.L. Mangasarian and R.R. Meyer*, Absolute value equations, Linear Algebra Appl. **419**(2006), No. 2, 359-367.

[9] *M.A. Noor, J. Iqbal and E. Al-Said*, Residual Iterative Method for Solving Absolute Value Equations, Abstr. Appl. Anal., **2012**(2012), No. 1, 1-9.

[10] *J.M. Ortega and W.C. Rheinboldt*, Iterative Solution of Nonlinear Equations in Several Variables, Academic Press, New York, 1970.

[11] *B.T. Polyak*, Introduction to Optimization, Optimization Software Inc, Publications Division, New York, 1987.

[12] *R.T. Rockafellar*, New applications of duality in convex programming, In Proceedings Fourth Conference on Probability, Brasov, (1971). 73-81.

[13] *J. Rohn*, A theorem of the alternatives for the equation $Ax+B|x|=b$, Linear. Multilinear. A., **52**(2004), No. 6, 421-426.

[14] *J. Rohn, V. Hooshyarbakhsh and R. Farhadsefat*, An Iterative Method for Solving Absolute Value Equations and Sufficient Conditions for Unique Solvability, Optim. Lett., **8**(2014), No. 1, 35-44.

[15] *L. Yong*, Particle Swarm Optimization for absolute value equations, J. Comput. Inform. Syst., **6**(2010), No. 7, 2359-2366.