Filomat 35:2 (2021), 459–476
https://doi.org/10.2298/FIL2102459J

Published by Faculty of Sciences and Mathematics, University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

An Optimized AOR Iterative Method for Solving Absolute Value Equations

Alireza Fakharzadeh Jahromi[a,b], Nafiseh Naseri Shams[a]

[a]*Department of OR, Faculty of Maths, Shiraz University of Technology, Shiraz, Iran*
[b]*Fars Elite Foundation, Shiraz, Iran, P.O. Box 71966-98893*

Abstract. In recent years, the AOR iterative method has been proposed for solving absolute value equations. This method has two parameters γ and ω. In this paper, we intend to find the optimal parameters of this method to improve convergence rate by suitable optimization techniques. Meanwhile, the convergence of the optimized AOR iterative method is discussed. It is both theoretically and experimentally demonstrated efficiency of the optimized AOR iterative method in contrast with the AOR and SOR methods.

1. Introduction

We consider the following system of the absolute value equations (AVEs):

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

where $A \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$ are given, and $|\,.\,|$ denotes the absolute value. In the special case $B = I$, where I denotes the identity matrix, (1) reduces to the form

$$Ax - |x| = b. \tag{2}$$

The AVE (1) requires that B is a nonzero matrix. Based on this reality, (1) is nonlinear and nondifferentiable. This equation was first introduced in [24] and has been studied by Mangasarian [14–16], Mangasarian and Meyer [17], Prokopyev [22], and Rohn [25, 27]. In all these papers, the authors were interested in finding some solutions of (1). In [17] a bilinear program was prescribed for the special case when the singular values of A are not less than one. As was shown in [17], the general NP-hard linear complementarity problem (LCP) [5–7], which subsumes many mathematical programming problems, can be formulated as AVE (2). This implies that (2) is NP-hard. The significance of the AVE (2) arises from the fact that linear programs, quadratic programs, bimatrix games and other problems can all be reduced to an LCP [6, 7], which in turn is equivalent to the AVE (2). Recently in [18], it has also been shown that the general AVE (1) is equivalent to a horizontal linear complementarity problem (HLCP). If B is a zero matrix, then the absolute value equation (1) reduces to a system of linear equations $Ax = b$, which has several applications in scientific computations, see [20].

2010 *Mathematics Subject Classification*. Primary 49xx; Secondary 49Mxx, 65K10.
Keywords. Absolute value equation, Optimized AOR method, Convergence, L-matrix, Symmetric positive definite.
Received: 14 January 2019; Revised: 02 August 2020; Accepted: 20 May 2021
Communicated by Marko Petković
Email addresses: a_fakharzadeh@sutech.ac.ir (Alireza Fakharzadeh Jahromi), n.naseri@sutech.ac.ir (Nafiseh Naseri Shams)

Lately, to solve (1) effectively, some numerical methods have been expanded, such as the Newton-type method [4, 10, 12, 16, 35], the sign accord (SA) method [25] and the AOR method [13]. For other numerical methods, one can look at the works of Noor et al. [20, 21]. In fact, little heed has been paid to iterative methods for solving (1). In recent years, a lot of effort has been made in expanding iterative methods for solving (1). For example, Rohn et al. [28] presented a general preconditioned Richardson iteration to solve (1). Based on Hermitian and skew-Hermitian splitting (HSS in Bai et al. [2]) of the coefficient matrix A in (1), the Picard-HSS iterative method for AVE (2) has been established by Salkuyeh [30]. Clearly, the Picard-HSS method belongs to the category of stationary matrix splitting iterative methods. Whereas, based on our knowledge, the classical matrix-splitting iterative methods (such as Gauss-Seidel (GS), Jacobi, Successive Over Relaxation (SOR) and Accelerated Over Relaxation (AOR) methods) for solving (1) are granted little attention.

In the current article, we first briefly review the AOR iterative method and its properties for solving (1). Thereafter, we apply Interior point, SQP, Random search and PSO techniques which are popular approaches in the literature, to find the optimal parameters of the AOR iteration. Then we present a modified method called the optimal AOR (OAOR) method, which is more stable and effective than the AOR method for solving the absolute value equations. We will show that if the AOR method is convergent for solving AVE (1), the OAOR method can be converged faster. In this regard, the article is organized as follows:
In section 2, we recall some beneficial definitions and theorems which are used in the next sections. In section 3, the AOR method and its properties are reviewed for solving (1). Afterwards in section 4 we calculate the optimal parameters of AOR iteration and call the modified method OAOR iterative method. We claim that the OAOR method works far better than the AOR method. In order to verify the validity of our claim, some numerical results are presented in section 5. Finally, the paper is terminated with a summary and conclusion in section 6.

2. Preliminaries

For an arbitrary matrix $A \in \mathbb{R}^{n\times m}$, we say that $A \geq 0$ ($A > 0$) when all entries of A are nonnegative (positive). For two matrices A and B in $\mathbb{R}^{n\times m}$, $A \geq B$ ($A > B$) that is $A - B \geq 0$ ($A - B > 0$). For a square matrix A, we denote the spectral radius of A by $\rho(A)$.
For an arbitrary matrix $A \in \mathbb{R}^{n\times n}$, the decomposition $A = M - N$ is named a splitting if $M, N \in \mathbb{R}^{n\times n}$ and M is nonsingular. In the following, we remind some definitions and results which are utilized throughout this paper.

Definition 2.1. *[3] $A \in \mathbb{R}^{n\times n}$ is named a $Z-$matrix if $a_{ij} \leq 0$ for $i, j = 1, 2, 3, ..., n$ ($i \neq j$). A $Z-$matrix with positive diagonal elements is called an L-matrix.*

Definition 2.2. *[3] Assume that A is a $Z-$matrix with positive diagonal elements. Then the matrix A is called an $M-$matrix if A is nonsingular and $A^{-1} \geq 0$.*

Lemma 2.3. *Suppose that A is a $Z-$matrix. Then, A is an $M-$matrix if and only if there is a positive vector x such that $Ax > 0$.*

Proof. See [33]. □

Definition 2.4. *A splitting $A = M - N$ is called an $M-$splitting of A if M is an $M-$matrix and $N \geq 0$.*

Definition 2.5. *[31] A matrix A is told to be reducible if there exists a permutation matrix P such that PAP^T is a block upper triangular matrix; otherwise, it is irreducible.*

Theorem 2.6. *[29] Let $A \in \mathbb{R}^{n\times n}$. Then for every natural norm $\|\,.\,\|$ we have $\rho(A) \leq \| A \|$.*

Theorem 2.7. *[29] Suppose A and B are two square matrices such that $0 \leq A \leq B$. Then $\rho(A) \leq \rho(B)$.*

3. AOR method and its convergence analysis

For solving (1), Li [13] proposed the following splitting with two parameters for the coefficient matrix A as $A = \frac{1}{\omega}(M_{\gamma,\omega} - N_{\gamma,\omega})$ with

$$M_{\gamma,\omega} = A_D - \gamma A_L \quad \textit{and} \quad N_{\gamma,\omega} = (1-\omega)A_D + (\omega - \gamma)A_L + \omega A_U, \tag{3}$$

where $\gamma, \omega \neq 0$ are parameters, A_D is the diagonal part of A, and $-A_L$ and $-A_U$ are strictly lower and strictly upper triangular parts of A, respectively. The iterative format of AOR method for solving the absolute value equation (1) is

$$\begin{aligned}\frac{1}{\omega}(A_D - \gamma A_L)x^{(i+1)} =& \frac{1}{\omega}[(1-\omega)A_D + (\omega-\gamma)A_L + \omega A_U]x^{(i)} \\ &+ B|x^{(i)}| + b, \quad i = 0, 1, 2, ...,\end{aligned} \tag{4}$$

which is equivalent to

$$\begin{aligned}x^{(i+1)} =& (A_D - \gamma A_L)^{-1}[(1-\omega)A_D + (\omega-\gamma)A_L + \omega A_U]x^{(i)} \\ &+ \omega(A_D - \gamma A_L)^{-1}B|x^{(i)}| + \omega(A_D - \gamma A_L)^{-1}b, \quad i = 0, 1, 2, \dots .\end{aligned} \tag{5}$$

This method is non-stationary and there is no iteration matrix. If $\omega = \gamma$, then AOR method reduce to the SOR method.

Lemma 3.1. *[13] Suppose that $A = M - N$ with $det(M) \neq 0$ and $x^{(0)} \in \mathbb{R}^n$ be an arbitrary initial guess. Then for $\rho(|M^{-1}N| + |M^{-1}B|) < 1$, the iterative sequence $x^{(i)}$ given by*

$$x^{(i+1)} = M^{-1}Nx^{(i)} + M^{-1}B|x^{(i)}| + M^{-1}b, \qquad i = 1, 2, 3, ..., \tag{6}$$

converges to the unique solution x^* of (1).

Proof. Suppose that x^* is a solution of (1). *Then*

$$x^* = M^{-1}Nx^* + M^{-1}B|x^*| + M^{-1}b. \tag{7}$$

After subtracting (7) *from* (6), *we have*

$$x^{(i+1)} - x^* = M^{-1}N(x^{(i)} - x^*) + M^{-1}B(|x^{(i)}| - |x^*|). \tag{8}$$

From (8) *we obtain*

$$\begin{aligned}|x^{(i+1)} - x^*| &= |M^{-1}N(x^{(i)} - x^*) + M^{-1}B(|x^{(i)}| - |x^*|)| \\ &\leq |M^{-1}N(x^{(i)} - x^*)| + |M^{-1}B(|x^{(i)}| - |x^*|)| \\ &\leq |M^{-1}N| \cdot |(x^{(i)} - x^*)| + |M^{-1}B| \cdot |(|x^{(i)}| - |x^*|)| \\ &\leq |M^{-1}N| \cdot |x^{(i)} - x^*| + |M^{-1}B| \cdot |x^{(i)} - x^*| \\ &= (|M^{-1}N| + |M^{-1}B|)|x^{(i)} - x^*|.\end{aligned}$$

By Theorem 4.1 from [29], this shows that $\lim_{i\to\infty} x^{(i)} = x^*$ *when* $\rho(|M^{-1}N| + |M^{-1}B|) < 1$.
Now, we prove that if $\rho(|M^{-1}N| + |M^{-1}B|) < 1$ *then, the iterative form* (6) *converges to the unique solution x^* of the AVE* (1) . *Let* (1) *has another solution $\widehat{x}$; then*

$$\widehat{x} = M^{-1}N\widehat{x} + M^{-1}B|\widehat{x}| + M^{-1}b. \tag{9}$$

Subtracting (9) *from* (7), *we obtain*

$$|x^* - \widehat{x}| \leq (|M^{-1}N| + |M^{-1}B|)|x^* - \widehat{x}|.$$

Since $\rho((|M^{-1}N| + |M^{-1}B|)) < 1$, *then* $x^* = \widehat{x}$. □

Corollary 3.2. *Let $A = M - N$ with $det(M) \neq 0$ and $x^{(0)} \in \mathbb{R}^n$ be an arbitrary initial guess. If $\| |M^{-1}N| \| + \| |M^{-1}B| \| < 1$, where $\| . \|$ is an consistent matrix norm, then the given iterative sequence $x^{(i)}$ in* (6) *converges to the unique solution x^* of* (1).

Proof. *Using triangle inequality, we have:*

$$\| |M^{-1}N| + |M^{-1}B| \| \leq \| |M^{-1}N| \| + \| |M^{-1}B| \| < 1 \tag{10}$$

From (10) *and Theorem 2.6, we have*

$$\rho(|M^{-1}N| + |M^{-1}B|) \leq \| |M^{-1}N| + |M^{-1}B| \| < 1 \tag{11}$$

Now by (11) *and Lemma 3.1, the result is obtained.* □

Corollary 3.3. *Suppose that $A = M - N$ be an $M-$splitting and $x^{(0)} \in \mathbb{R}^n$ be an arbitrary initial guess. Then for $\rho(M^{-1}N + M^{-1}|B|) < 1$, the iterative sequence $x^{(i)}$ given by* (6), *converges to the unique solution of* (1).

Proof. *Since A is an $M-$splitting, $M^{-1} \geq 0$ and $N \geq 0$. Hence $M^{-1}N \geq 0$ and $M^{-1}N + M^{-1}|B| \geq 0$. Besides*

$$0 \leq |M^{-1}N| + |M^{-1}B| \leq M^{-1}N + M^{-1}|B|. \tag{12}$$

From (12) *and Theorem 2.7, we have*

$$\rho(|M^{-1}N| + |M^{-1}B|) \leq \rho(M^{-1}N + M^{-1}|B|) < 1. \tag{13}$$

Therefore using (13) *and Lemma 3.1, the result is obtained.* □

Suppose that

$$T_{AOR} = (A_D - \gamma A_L)^{-1}[(1-\omega)A_D + (\omega - \gamma)A_L + \omega A_U] + \omega(A_D - \gamma A_L)^{-1}|B|; \tag{14}$$

then we have the following Theorem. Note that for SOR and OAOR iterative methods, T_{AOR} changes to T_{SOR} and T_{OAOR} in the following respectively.

$$T_{SOR} = (A_D - \omega A_L)^{-1}[(1-\omega)A_D + \omega A_U] + \omega(A_D - \omega A_L)^{-1}|B|;$$

$$T_{OAOR} = (A_D - \gamma^* A_L)^{-1}[(1-\omega^*)A_D + (\omega^* - \gamma^*)A_L + \omega^* A_U] + \omega^*(A_D - \gamma^* A_L)^{-1}|B|;$$

where γ^* and ω^* are the optimal parameters obtained using one of the Interior point, SQP, Random search or PSO optimization methods.

Theorem 3.4. *Let $A = (a_{ij}) \in \mathbb{R}^{n\times n}$ be an $L-$matrix. If $\rho(T_{AOR}) < 1$, then the iterative sequence $\{x_i\}$ given by AOR method* (5) *converges to the unique solution of* (1) *for an arbitrary initial guess $x_0 \in \mathbb{R}^n$.*

Proof. *See [13]* □

4. The OAOR method

Selecting the parameters for applying (5) could be randomly; but to have more suitable results, it is better to choose them optimally. In this regard, and also to have a better approximation in (5) and also fast rate of convergence to reach the best solution of (1) and also least possible time-consuming, specially for large scale problems, we prefer fo find the parameters optimally. The convergence analysis of an iterative method is based on the spectral radius of the iteration matrix. For large number of iterations, the corresponding error remarkably decreases using the spectral radius factor of the iteration matrix; that is, when the spectral radius is smaller, the convergence is faster. In fact, the AOR iterative method for solving AVEs is an non-stationary method, therefore the actual iteration matrix can change at every iteration, depending on the sign of x and can not consider a iteration matrix for it. Therefore we can not calculate the optimal parameters using minimizing the spectral radius of iteration matrix. In such a case, it is reasonable that we calculate the optimal parameters of the method by minimizing an upper boundary of the spectral radius of the iteration matrix. Besides TAOR is not exactly the iteration matrix, but it is a bound that is used in convergence theorem. Therefore, it is logical to optimize the parameters by minimizing of $\rho(T_{AOR})$. In the following, we present several optimization techniques to find the optimal parameters of the AOR method, which just need to minimize $\rho(T_{AOR})$ (under the assumption that the matrix A is a irreducible L-matrix), or need to minimize 2-norm of the residual vector (provided that A is a general nonsymmetric matrix or a symmetric positive definite matrix).

A family of optimization methods that have been studied are the family of interior point methods which are remarkably efficient in solving optimization methods [19]. Interior-point (or barrier) methods have proved to be as successful for nonlinear optimization as for linear programming and they are currently considered as the most powerful algorithms for large-scale nonlinear programming. But if one uses some interior point standard forms (like `fmincon` from MATLAB) by regarding necessary preparations, the CPU times may be not suitably small. Therefore, we advised to use some other classical optimization based methods or some suitable meta-heuristic techniques, specially when the dimension is high. In this cases, we preferred to use the sequential quadratic programming (SQP) algorithm, the PSO or Random search techniques.

The SQP approach is appropriate for small or large problems. SQP methods show their strength when solving problems with significant nonlinearities in the constraints [19]. For optimization through Interior point and SQP techniques, we use the fmincon in MATLAB with Interior point and SQP options, respectively. The convergence of these methods has been proved in [19]. The PSO technique solves a problem by having a population of candidate solutions, and moving these particles around in the search space according to simple mathematical formulae over the particle's position and velocity. Each particle's movement is influenced by its local best known position, but is also guided toward the best known positions in the search space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions. PSO is a metaheuristic as it makes few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions. Also, PSO does not use the gradient of the problem being optimized, which means PSO does not require that the optimization problem be differentiable as is required by classic optimization methods. Moreover, the convergence of this method has been investigated in [23]. Random search is a family of numerical optimization methods that do not require the gradient of the problem to be optimized, and it can hence be used on functions that are not continuous or differentiable. Such optimization methods are also known as direct-search or derivative-free. Random Search works by iteratively moving to better positions in the search space, which are sampled from a hypersphere surrounding the current position. Plus, the convergence of this method has been shown in [34]. Also the experience of authors in [8, 9] shows that PSO (as a meta-heuristic algorithm) and Random search (as a classical algorithm) are the two most appropriate once in applications.

4.1. **Proposed Algorithms**

Let r_k denote the residual vector of the AOR method at the kth iterative step; that is:

$$r_k = b - Ax_k + B|x_k|.$$

Suppose that the matrix A is an irreducible $L-$matrix. In this case, first by minimizing $\rho(T_{AOR})$ on

$$F_1 = \{(\gamma, \omega)|\ \| (A_D - \gamma A_L)^{-1}[(1-\omega)A_D + (\omega-\gamma)A_L + \omega A_U] + \omega(A_D - \gamma A_L)^{-1}|B|\ \|< 1\}$$

the optimal parameters γ^* and ω^* are obtained. Then, substituting them in (6) gives

$$\begin{aligned} x_{k+1} =& (A_D - \gamma^* A_L)^{-1}[(1-\omega^*)A_D + (\omega^* - \gamma^*)A_L + \omega^* A_U]x_k \\ &+ \omega^*(A_D - \gamma^* A_L)^{-1}B|x_k| + \omega^*(A_D - \gamma^* A_L)^{-1}b. \end{aligned} \tag{15}$$

In fact, the optimal parameters of the AOR method, in case that A is an irreducible $L-$matrix, can be obtained by solving a constrained optimization problem; more precisely

$$(\gamma^*, \omega^*) = \underset{F_1}{\operatorname{argmin}}\ \rho(T_{AOR}.)$$

According to this, we can present the following algorithm for solving (1) in this case.

Algoritm 4.1.1
Step0. Given an initial vector $x_0 \in R^n$, a precision ϵ_1 and Set $k = 0$.
Step1. Solve $\min\limits_{F_1} \rho(T_{AOR})$ by an appropriate optimization method to obtain γ^* and ω^*.
Step2. Compute x_{k+1} from (15).
Step3. If the stopping condition is satisfied, stop and x_{k+1} is the solution. If not, set $k \leftarrow k+1$ and go to Step2.

Regarding the definition of F_1 and Theorem 2.6, Theorem 3.4 indicates that the obtained sequence by (15) is converged to the unique solution of (1).
Now assume that A is a symmetric positive definite matrix; in this case, we minimize the residual vector on

$$F_2 = \{(\gamma, \omega) | \, det(A_D - \gamma A_L) \neq 0 \, and \, \| \, |(A_D - \gamma A_L)^{-1}((1-\omega)A_D + (\omega - \gamma)A_L + \omega A_U)| \, \| + \| \, |\omega(A_D - \gamma A_L)^{-1} B| \, \| < 1\}$$

and determine the optimal parameters γ^* and ω^*, i.e.

$$(\gamma^*, \omega^*) = \underset{F_2}{\operatorname{argmin}}\ \|r^{k+1}\|_2.$$

Then, one can construct the related sequence (15). Regarding the definition F_2, the Corollary 3.2 shows that this sequence converges to a unique solution of (1). So, we have the following algorithm for the case that A in (1) is a symmetric positive definite matrix.

Algoritm 4.1.2
Step0. Given an initial vector $x_0 \in R^n$ and a precision ϵ_2 and set $k = 0$.
Step1. Solve the problem $\min\limits_{F_2} \|r^{k+1}\|_2$ by an appropriate method and obtain γ^* and ω^*.
Step2. Compute x_{k+1} from (15).
Step3. If the stopping criteria is holded, stop and x_{k+1} is the solution. If not, set $k \leftarrow k + 1$ and go to Step2.

In the third case, suppose A is a general nonsymmetric matrix; here, we compute the optimal parameters and their related sequence as the previous case and use Algorithm 4.1.2 to solve (1).

We remind that the stopping condition can be a fixed number of iterations, a residual threshold and etc. We consider $\epsilon_1, \epsilon_2 > 0$ as a residual threshold in the above algorithms.
Note. In each iteration of our proposed method, the parameters are fixed, since they are firstly determined optimally, and then the obtained optimal values are used in each iteration.

Remark 4.1. *We know that in the AOR method, parameters γ and ω are randomly selected. However, in the above algorithms, parameters γ and ω are optimally calculated and then we will have AOR iteration with γ^* and ω^* optimal parameters. In other words, OAOR method is AOR iterative method by applying optimal parameters of γ^* and ω^*. Since the convergence of the AOR method with the randomly selected parameters is proved in Theorem 3.4, we conclude that the optimized AOR iterative method, in which we apply the AOR method with the optimal parameters, under the assumptions of Theorem 3.4 is convergent and its convergence rate is much faster than AOR method.*

5. Numerical results

To demonstrate the performance of the proposed method for solving the AVE (1), here, we report some numerical experiments. In our numerical computations, the initial guess is taken to be

$$x^{(0)} = (1, 0, 1, 0, ..., 1, 0, ...)^T \in \mathbb{R}^n$$

and all the iterations are terminated as soon as

$$\frac{\| Ax^{(k)} - B \mid x^{(k)} \mid -b \|_2}{\| b \|_2} \leq 10^{-6},$$

where $x^{(k)}$ denotes the kth approximate solution [13]. the All numerical procedures have been done by MATLAB R2014a.

As pointed out in [17], if 1 is not an eigenvalue of M, then the linear complementarity problem (LCP)

$$Mz + q \geq 0, \qquad z \geq 0 \quad \text{and} \quad z^T(Mz + q) = 0,$$

with $M \in \mathbb{R}^{n\times n}$ and $q \in \mathbb{R}^n$, can reduced to the AVE

$$(M - I)^{-1}(M + I)x - |x| = (M - I)^{-1}q,$$

where

$$x = \frac{1}{2}((M - I)z + q).$$

Also recently the equivalence between HLCPs and (1) has been noted in [18]. More precisely, it is shown that if x is a solution of the general AVE (1), then the vectors $z = \max(0, x)$ and $w = \max(0, -x)$ solve the HLCP

$$(A - B)z - (A + B)w = b; \quad z \geq 0; \quad w \geq 0; \quad z^T w = 0.$$

Conversely, if (z, w) solve the HLCP

$$Cz - Dw = b; \quad z \geq 0; \quad w \geq 0; \quad z^T w = 0,$$

with $C, D \in \mathbb{R}^{n\times n}$ and $z, w, b \in \mathbb{R}^n$, then $x = z - w$ solves the general AVE (1) with

$$A = \frac{1}{2}(C + D) \qquad B = \frac{1}{2}(D - C).$$

It is necessary to mention we randomly selected the parameters of the not optimized AOR and SOR methods from the interval [0,1]. In this way, each of AOR and SOR methods has been run ten times with random parameters, and then provided a mean of computational times, number of iterations, etc.

We remind that Rohn showed in [26] that if $\sigma_{max}(|B|) < \sigma_{min}(A)$, where σ_{max} and σ_{min} denote the maximal and minimal singular values respectively, then for each right-hand side $b \in \mathbb{R}^n$, the absolute value equation $Ax + B|x| = b$ has a unique solution. Accordingly, the absolute value equations given in the following examples have a unique solution.

Example 5.1. *[11] Consider the AVE* (1) *with* $B = I$,

$$A = tridiag(-1, 4, -1) \in \mathbb{R}^{n\times n} \quad \text{and} \quad b = Az - |z|,$$

where $z = (-1, 1, -1, 1, ..., -1, 1)^T \in \mathbb{R}^n$.

Obviously, A is an irreducible $L-$matrix. In Tables 1-4, the optimal parameters of the AOR iterative method have been calculated using Interior point, SQP, Random search and PSO techniques. Also, the required time (denoted as CPU) to calculate the optimal parameters has been reported. In Tables 5-6, we have also compared $\rho(T_{OAOR})$ with $\rho(T_{AOR})$ and $\rho(T_{SOR})$. Moreover the OAOR, AOR and SOR methods are compared respect to the number of iterations and CPU times. The numerical results demonstrate that the optimized AOR method has a faster asymptotic rate of convergence compared with the AOR and SOR methods. Moreover, as the tables show, the number of iterations of the OAOR method is remarkably less than that of AOR and SOR.

Table 1: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.1 by Interior point technique.

n	γ^*	ω^*	CPU
25	1	1	1.9361
100	1	1	2.7075
400	0.7897	0.9586	15.4577
900	0.4052	0.8746	110.0264
1600	0.6777	0.7644	363.5897

Table 2: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.1 by SQP technique.

n	γ^*	ω^*	CPU
25	1	1	1.6665
100	1	1	2.2016
400	1	1	5.9072
900	1	1	21.9294
1600	0.3063	1	732.4196

Table 3: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.1 by Random search technique.

n	γ^*	ω^*	CPU
25	0.9749	0.9945	0.0650
100	0.9732	0.9854	0.8842
400	0.9476	0.9991	6.5027
900	0.9419	0.9786	43.4342
1600	0.9774	0.9828	153.6666

Table 4: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.1 by PSO technique.

n	γ^*	ω^*	CPU
25	1	1	0.1472
100	1	1	0.9130
400	1	1	2.5802
900	1	1	6.4703
1600	0.9943	0.9992	23.5622

Table 5: Comparison results between the number of iterations (CPU-time in seconds) for Example 5.1.

n	Algorithm	SOR	AOR	OAOR
25	Interior point	112 (0.0164)	65 (0.0128)	14 (0.0033)
	SQP			14 (0.0032)
	Random search			14 (0.0033)
	PSO			14 (0.0032)
100	Interior point	117 (0.0196)	287 (0.0499)	14 (0.0035)
	SQP			14 (0.0034)
	Random search			14 (0.0034)
	PSO			14 (0.0033)
400	Interior point	98 (0.0258)	661 (0.1557)	16 (0.0051)
	SQP			14 (0.0043)
	Random search			14 (0.0047)
	PSO			14 (0.0044)
900	Interior point	295 (0.1186)	76 (0.0301)	21 (0.0092)
	SQP			14 (0.0059)
	Random search			15 (0.0069)
	PSO			14 (0.0059)
1600	Interior point	45 (0.0242)	66 (0.0430)	22 (0.0136)
	SQP			19 (0.0131)
	Random search			14 (0.0104)
	PSO			14 (0.0093)

Example 5.2. *[1] Assume that m be a prescribed positive integer and $n = m^2$. Consider the LCP(q,M), in which $M \in \mathbb{R}^{n\times n}$ is given by $M = \hat{M} + 2I$ and $q \in \mathbb{R}^n$ is given by $q = -Mz^*$, where*

$$\hat{\mathbf{M}} = \begin{pmatrix} S & -I & 0 & \dots & 0 & 0 \\ -I & S & -I & \dots & 0 & 0 \\ 0 & -I & S & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & S & -I \\ 0 & 0 & 0 & \dots & -I & S \end{pmatrix} \in \mathbb{R}^{n\times n},$$

with

$$\mathbf{S} = \begin{pmatrix} 4 & -1 & 0 & \dots & 0 & 0 \\ -1 & 4 & -1 & \dots & 0 & 0 \\ 0 & -1 & 4 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 4 & -1 \\ 0 & 0 & 0 & \dots & -1 & 4 \end{pmatrix} \in \mathbb{R}^{m\times m},$$

and

$$z^* = (1, 2, 1, 2, ..., 1, 2, ...)^T \in \mathbb{R}^n.$$

It is clear that, A is a symmetric positive definite matrix. In Tables 7-10, the optimal parameters of the AOR iterative method have been calculated by Interior point, SQP, Random search and PSO techniques. Furthermore, the required

Table 6: Comparison results between spectral radius of T_{SOR}, T_{AOR} and T_{OAOR} for Example 5.1.

n	Algorithm	SOR	AOR	OAOR
25	Interior point	0.8527	0.8392	0.6503
	SQP			0.6503
	Random search			0.6560
	PSO			0.6503
100	Interior point	0.8724	0.9193	0.6542
	SQP			0.6542
	Random search			0.6633
	PSO			0.6542
400	Interior point	0.8867	0.9157	0.6960
	SQP			0.6548
	Random search			0.6628
	PSO			0.6548
900	Interior point	0.8911	0.8632	0.7559
	SQP			0.6558
	Random search			0.6710
	PSO			0.6558
1600	Interior point	0.8472	0.8836	0.7672
	SQP			0.7288
	Random search			0.6663
	PSO			0.6571

CPU-time (in second) to calculate the optimal parameters are presented. Comparison results between OAOR, AOR and SOR iterative methods from point of view the 2-norm of the residual vector, the number of iterations and CPU time have been presented for Example 5.2 in Tables 11-12.

Table 7: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.2 by interior point technique.

n	γ^*	ω^*	CPU
25	1.0121e-6	0.5555	2.0475
100	6.0634e-7	0.5030	2.4564
400	2.36587e-7	0.4830	9.9263
900	1.3112e-7	0.47679	73.6330
1600	8.8824e-8	0.4740	402.6275

Table 8: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.2 by SQP technique.

n	γ^*	ω^*	CPU
25	0	0.5555	2.0701
100	0	0.5030	2.0874
400	0	0.4830	8.3794
900	0	0.4768	82.7319
1600	0	0.4740	372.6798

Table 9: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.2 by Random search technique.

n	γ^*	ω^*	CPU
25	0.6869	0.7341	0.4839
100	0.7772	0.7779	4.6180
400	0.1207	0.5211	128.9289
900	0.5892	0.8080	591.9871
1600	0.4865	0.5692	1988.6268

Table 10: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.2 by PSO technique.

n	γ^*	ω^*	CPU
25	0.001	0.5615	0.3063
100	0.0047	0.5358	2.2256
400	0.0043	0.5000	25.1021
900	0.0095	0.5284	149.0745
1600	0.0098	0.5802	672.3024

Table 11: Comparison results between the number of iterations (CPU-time in seconds) for Example 5.2.

n	Algorithm	SOR	AOR	OAOR
25	Interior point	77 (0.0026)	55 (0.0024)	9 (0.0003)
	SQP			9 (0.0004)
	Random search			11 (0.0004)
	PSO			9 (0.0003)
100	Interior point	41 (0.0112)	53 (0.0153)	9 (0.0027)
	SQP			9 (0.0026)
	Random search			15 (0.0044)
	PSO			11 (0.0033)
400	Interior point	26 (0.1196)	35 (0.1699)	10 (0.0513)
	SQP			10 (0.0514)
	Random search			10 (0.0493)
	PSO			10 (0.0480)
900	Interior point	56 (1.1813)	78 (1.6272)	10 (0.2767)
	SQP			10 (0.2754)
	Random search			31 (0.7821)
	PSO			14 (0.3302)
1600	Interior point	41 (3.0222)	36 (2.2962)	10 (0.6766)
	SQP			10 (0.6715)
	Random search			8 (0.5372)
	PSO			21 (1.3737)

Example 5.3. *[1] Assume that m be a prescribed positive integer and $n = m^2$. Consider the LCP(q,M), in which $M \in \mathbb{R}^{n\times n}$ is given by $M = \hat{M} + 2I$ and $q \in \mathbb{R}^n$ is given by $q = -Mz^*$, where*

$$\hat{\mathbf{M}} = \begin{pmatrix} S & -0.5I & 0 & \dots & 0 & 0 \\ -1.5I & S & -0.5I & \dots & 0 & 0 \\ 0 & -1.5I & S & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & S & -0.5I \\ 0 & 0 & 0 & \dots & -1.5I & S \end{pmatrix} \in \mathbb{R}^{n\times n},$$

with

$$\mathbf{S} = \begin{pmatrix} 4 & -0.5 & 0 & \dots & 0 & 0 \\ -1.5 & 4 & -0.5 & \dots & 0 & 0 \\ 0 & -1.5 & 4 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 4 & -0.5 \\ 0 & 0 & 0 & \dots & -1.5 & 4 \end{pmatrix} \in \mathbb{R}^{m\times m},$$

and

$$z^* = (1, 2, 1, 2, ..., 1, 2, ...)^T \in \mathbb{R}^n.$$

Table 12: Comparison results between 2-norm of the residual vector for Example 5.2.

n	Algorithm	SOR	AOR	OAOR
25	Interior point	9.407e-05	8.250e-05	3.185e-06
	SQP			3.185e-06
	Random search			6.297e-06
	PSO			2.514e-06
100	Interior point	2.104e-04	2.067e-04	2.352e-05
	SQP			2.352e-05
	Random search			1.401e-05
	PSO			2.337e-05
400	Interior point	3.927e-04	4.547e-04	3.505e-05
	SQP			3.505e-05
	Random search			2.039e-05
	PSO			5.206e-05
900	Interior point	6.491e-04	7.408e-04	7.599e-05
	SQP			7.595e-05
	Random search			5.927e-05
	PSO			4.619e-05
1600	Interior point	9.006e-04	8.849e-04	1.188e-04
	SQP			1.188e-04
	Random search			1.023e-04
	PSO			8.378e-05

Evidently, A is a nonsymmetric matrix. The optimal parameters of the AOR iterative method and the required time to calculate them have been reported in Tables 13-16. In Tables 17-18, some iterative results are presented in order to illustrate the convergence behavior of the SOR, AOR and OAOR methods for Example 5.3. As the numerical results show, the OAOR method requires less iteration steps than SOR and AOR methods. According to the numerical results in Tables 1-18, the OAOR method is more powerful and efficient than the SOR and AOR methods.

Table 13: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.3 by interior point technique.

n	γ^*	ω^*	CPU
25	1.6435e-7	0.5671	1.6444
100	5.1024e-6	0.4988	1.6665
400	8.5831e-8	0.4740	9.3217
900	1.4216e-5	0.4666	62.8804
1600	3.9918e-9	0.4633	333.3669

Table 14: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.3 by SQP technique.

n	γ^*	ω^*	CPU
25	0	0.5671	1.7301
100	0	0.4988	1.9328
400	0	0.4740	6.3138
900	0	0.4665	46.2533
1600	0	0.4633	305.4300

Table 15: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.3 by Random search technique.

n	γ^*	ω^*	CPU
25	0.00340	0.4340	0.4523
100	0.0029	0.4759	3.1995
400	0.0038	0.5283	72.5528
900	0.0046	0.5028	299.8855
1600	0.0027	0.4892	455.0418

Table 16: Calculating the optimal parameters of AOR method and the required time to calculate them for Example 5.3 by PSO technique.

n	γ^*	ω^*	CPU
25	8.59e-6	0.5834	0.3997
100	2e-6	0.5231	2.6300
400	0	0.5029	10.3744
900	6.53e-6	0.5317	57.8127
1600	6.79e-6	0.5540	401.9900

Table 17: Comparison results between the number of iterations (CPU-time in seconds) for Example 5.3.

n	Algorithm	SOR	AOR	OAOR
25	Interior point	44 (0.0015)	35 (0.0012)	9 (0.0004)
	SQP			9 (0.0003)
	Random search			13 (0.0005)
	PSO			9 (0.0003)
100	Interior point	1559 (0.3771)	29 (0.0068)	9 (0.0020)
	SQP			9 (0.0020)
	Random search			10 (0.0022)
	PSO			9 (0.0020)
400	Interior point	65 (0.3680)	27 (0.1417)	10 (0.0487)
	SQP			10 (0.0486)
	Random search			11 (0.0547)
	PSO			9 (0.0432)
900	Interior point	41 (0.9875)	42 (1.0550)	10 (0.2684)
	SQP			10 (0.2462)
	Random search			11 (0.2565)
	PSO			14 (0.3717)
1600	Interior point	34 (2.1661)	105 (7.7605)	11 (0.8246)
	SQP			11 (0.8343)
	Random search			11 (0.8235)
	PSO			18 (1.4669)

Table 18: Comparison results between 2-norm of the residual vector for Example 5.3.

n	Algorithm	SOR	AOR	OAOR
25	Interior point	8.847e-05	9.488e-05	3.289e-06
	SQP			3.289e-06
	Random search			1.192e-05
	PSO			8.015e-06
100	Interior point	1.570e-04	1.961e-04	2.182e-05
	SQP			2.182e-05
	Random search			1.874e-05
	PSO			1.645e-05
400	Interior point	4.016e-04	3.693e-04	3.592e-05
	SQP			3.592e-05
	Random search			4.376e-05
	PSO			5.298e-05
900	Interior point	5.984e-04	5.671e-04	7.805e-05
	SQP			7.850e-05
	Random search			4.148e-05
	PSO			5.092e-05
1600	Interior point	8.457e-05	8.853e-05	2.999e-05
	SQP			2.997e-05
	Random search			3.846e-05
	PSO			7.086e-05

6. Conclusion

In this paper, we presented the way of finding the optimal parameters of AOR method to improve convergence rate by suitable optimization techniques and the optimized method called OAOR method. We shown that how this new iterative method works and moreover, its convergence is also guaranteed. Besides, the effectiveness of OAOR method have been demonstrated through different cases of numerical experiments come out from literatures. As it is seen from the obtained results, the proposed method has been succeeded in improving the convergence rate compared with the classical SOR and AOR methods. Also, the number of iterations and the spectral radius (or 2-norm residual vector) of the optimized AOR method for different size (dimension) are remarkably better. Moreover, one can conclude that among the mentioned optimization algorithms, generally, the SQP and PSO algorithms have better performance for calculating the optimal parameters.

Acknowledgments

The authors would like to express their sincere thanks to anonymous referees for their useful comments which improved the quality of paper. The authors are also grateful to Professor Marko Petkovic for managing the review process.

References

[1] Z. Z. Bai, Modulus-based matrix splitting iteration methods for linear complementarity problems, Numer Linear Algebr Appl 17 (2010) 917–933.
[2] Z. Z. Bai, G. H. Golub, MK. Ng, Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems, SIAM J Matrix Anal Appl 24 (2003) 603–626.
[3] A. Berman, R. J. Plemons, Nonnegative matrices in the mathematics sciences, Philadelphia, SIAM.
[4] L. Caccetta, B. Qu, G. L. Zhou, A globally and quadratically convergent method for absolute value equations, Comput Optim Appl 48 (2011) 45–58.
[5] S. J. Chung, NP-completeness of the linear complementarity problem, J Optim Theory Appl 60 (1989) 393–399.
[6] R. W. Cottle, G. B. Dantzig, Complementary pivot theory of mathematical programming, Linear Algebra Appl 1 (1968) 103–125.
[7] R. W. Cottle, J. S. Pang, R. E. Stone, The linear complementarity problem, Academic Press, New York, 1992.
[8] A. Fakharzadeh J., Z. Rafiei, Best minimizing algorithm for shape-measure method, The Journal of Mathematics and Computer Science 5 no. 3 (2012) 176–184.
[9] A. Fakharzadeh J., M. Goodarzi, M. K. Jahromi, On efficiency of some meta-huristic optimization algorithm in shapemeasure method, The First National Conference on Meta-Heuristic Algorithms and Their Applications in Engineering and Science; 7-8 August 2014; Pardisan High Education Institute, Feridonkenar, Iran.
[10] S. L. Hu, Z. H. Huang, Q. Zhang, A generalized Newton method for absolute value equations associated with second order cones, J Comput Appl Math 235 (2011) 1490–1501.
[11] Y. F. Ke, C. F. Ma, SOR-like iteration method for solving absolute value equations, Applied Mathematics and Computation 311 (2017) 195–202.
[12] S. Ketabchi, H. Moosaei, An efficient method for optimal correcting of absolute value equations by minimal changes in the right hand side, J Comput Appl Math 64 (2012) 1882–1885.
[13] C. X. Li, A preconditioned AOR iterative method for the absolute value equations, Int J Comput Methods 14 (2016).
[14] O. L. Mangasarian, Absolute value equations via concave minimization, Optim Lett 1 (2007) 1–8.
[15] O. L. Mangasarian, Absolute value programming, Comput Optim Appl 36 (2007) 43–53.
[16] O. L. Mangasarian, A generalized Newton method for absolute value equations, Optim Lett 3 (2009) 101–108.
[17] O. L. Mangasarian, R. R. Meyer, Absolute value equations, Linear Algebr Appl 419 (2006) 359–367.
[18] F. Mezzadri, On the solution of general absolute value equations, Applied Mathematics Letters 107 (2020).
[19] J. Nocedal, S. J. Wright, Numerical Optimization, Springer, 1999.
[20] M. A. Noor, J. Iqbal, K. I. Noor, E. A1-Said, On an iterative method for solving absolute value equations, Optim Lett 6 (2012) 1027–1033.
[21] M. A. Noor, J. Iqbal, E. A1-Said, Residual iterative method for solving absolute value equations, Abstr Appl Anal 2012.
[22] O. Prokopyev, On equivalent reformulations for absolute value equations, Comput Optim Appl 44 no. 3 (2009) 363–372.
[23] Z. Qingqing,H. Xingshi and S. Na, Convergence Analysis and Parameter Select on PSO, Second International Symposium on Information Science and Engineering, (2009).
[24] J. Rohn, Systems of linear interval equations, Linear Algebra Appl 126 (1989) 39–78.
[25] J. Rohn, An algorithm for solving the absolute value equations, Electron J Linear Algebra 18 (2009) 589–599.
[26] J. Rohn, On unique solvability of the absolute value equation, Electron Optim Lett 3 (2009) 603–609.
[27] J. Rohn, An algorithm for solving the absolute value equations, An improvement, Technical Report 1063, Institute of Computer Science, Academy of Sciences of the Czech Republic, Prague 2010.

[28] J. Rohn, V. Hooshyarbakhsh, R. Farhadsefat, An iterative method for solving absolute value equations and sufficient conditions for unique solvability, Optim Lett 8 (2014) 35–44.
[29] Y. Saad, Iterative methods for sparse linear systems, SIAM, Philadelphia.
[30] D. K. Salkuyeh, The Picard-HSS iteration method for absolute value equations, Optim Lett 8 (2014) 2191–2202.
[31] R. S. Varga, Matrix Iterative Analysis, Springer, Berlin, 2000.
[32] Z. I. Woznicki, Basic comparison theorems for weak and weaker matrix splitting, Electron J Linear Algebra 8 (2003) 53–59.
[33] M. Wu, L. Wang, Y. Song, Preconditioned AOR iterative method for linear systems, Appl Numer Math 57 (2007) 672–685.
[34] Z. B. Zabinsky, Random Search algorithms, Wiley Encyclopedia of Operations Research and Management Science, (2011).
[35] C. Zhang, Q. J. Wei, Global and finite convergence of a generalized Newton method for absolute value equations, J Optim Theory Appl 143 (2009) 391–403.