Improved Inexact Alternating Direction Methods for a Class of Nonlinear Complementarity Problems

Jiewen He[1,*], Hua Zheng[2] and Seakweng Vong[1]

[1]*Department of Mathematics, University of Macau, Taipa, Macao 999078, China.*
[2]*School of Mathematics and Statistics, Shaoguan University, Shaoguan 512000, China.*

Abstract. Three types of improved inexact alternating direction methods for solving nonlinear complementarity problems with positive definite matrices and nonlinear terms are proposed. The convergence of the methods is proven. Numerical examples confirm the theoretical analysis and show that the methods have advantages over similar existing methods, especially in large size problems.

AMS subject classifications: 90C33, 65F10, 65F50, 65G40

Key words: Inexact alternating direction method, nonlinear complementarity problem, successive overrelaxation, iterative method, symmetric positive definite.

1. Introduction

Complementarity problems (CPs), first introduced by Cottle, are widely used in computing the equilibrium points of bimatrix games [11, 28]. Such problems arise in scientific computing, engineering and economy when considering numerous problems in elasticity, traffic network design, optimal control, free boundary problems of fluid dynamics, asset pricing and image processing [14, 19, 20, 23, 37, 60].

Over the past several decades, various numerical algorithms have been proposed to solve nonlinear complementarity problems (NCPs). On the other hand, Mangasarian [34] showed the equivalence of CPs to systems of nonlinear equations and Mangasarian and Solodov [35] reformulated NCPs as unconstrained and constrained minimisation problems, Noor [38] employed several fixed point methods to complementarity problems. Other approaches include projection-type methods [47], merit functions [16], smooth and nonsmooth Newton methods [10, 13, 27, 40, 41], interior-point methods [50], linearisation methods [18], domain decomposition methods [1, 26]. Existence and uniqueness results for complementarity problems are summarised in surveys and books [17–19, 23]. Using the solvers based on matrix splitting and multisplitting, Bai [2, 3] and Bai and Wang [9]

---

*Corresponding author. *Email addresses:* `yc07443@umac.mo` (J.W. He), `hzheng@sgu.edu.cn` (H. Zheng), `swvong@umac.mo` (S.W. Vong)

developed parallel nonlinear monotonically convergent multisplitting relaxation methods. Asynchronous parallel nonlinear multisplitting relaxation methods for large sparse NCPs have been studied in [5, 6]. These methods are very efficient on high-speed MIMD multi-processor systems since they allow to avoid the synchronous delays between corresponding processors. Matrix splitting methods are special cases of these multisplitting methods.

Here, we focus on a class of weakly NCPs. Recall that Bai [4] first considered the systems of weakly nonlinear equations, also called the mildly nonlinear systems [8]. Linear complementarity problems (LCPs) represent a special case of NCPs and in recent years, a number of LCP solvers — viz. modulus-based matrix splitting (MMS) iteration methods, attracted considerable attention [7, 31, 39], since very often they perform better than projected relaxation methods [12] and modified modulus methods [15]. In particular, Mezzadri [36] and Li [32] used MMSs to solve horizontal linear complementarity problems and second-order cone LCPs, respectively. MMS iteration methods have been also applied to a class of weakly NCPs — cf. [25, 45]. It turns out that MMSs are more efficient than the fixed point method [38] and the Fischer-Burmeister semismooth Newton method [13]. Zheng [53] improved the convergence of MMS iteration method for NCPs with H-matrices. The convergence results have been then extended from H-compatible splitting to H-splitting of NCPs with H_+-matrices [30]. Extensions of MMS methods, such as modified MMS method [33], accelerated MMS method [29], relaxed MMS method [58], two-step MMS method [46, 55], preconditioned MMS method [54, 59] and modulus-based matrix multisplitting methods [43, 44, 48, 49], developed in recent years improved the original one. Besides, a modulus-based nonsmooth Newton method and sign-based methods are studied in [57] and [56], respectively.

In recent years, alternating direction methods (ADMs) have been applied to LCPs of the free boundary problems of mathematical physics [21, 22, 42]. Zhang *et al.* [51, 52] developed inexact ADMs of multipliers for solving a kind of LCPs. These methods were applied to weakly NCPs in He *et al.* [24]. The main contribution of our work is as follows:

- Three new numerical methods for solving weakly NCPs are proposed and their convergence is analysed.

- The methods have their own advantages and are much more efficient than the existing methods.

In practical applications, the system matrix A of weakly NCPs is usually large, sparse, symmetric and positive definite. This motivates the current study on different inexact ADMs. In Section 2, a direct inexact ADM, a symmetric successive overrelaxation (SSOR) based inexact ADM and a modified SSOR-based inexact ADM are introduced. Convergence of these methods is studied in Section 3. The numerical examples presented in Section 4, illustrate the theoretical results. Section 5 contains concluding remarks.

2. Improved Inexact ADMs

A nonlinear complementarity problem with respect to a vector-valued function $F : D \rightarrow \mathbb{R}^n$, where D is a subset of $\mathbb{R}^n$ containing at least the nonnegative cone, consists in finding

a vector $u \in \mathbb{R}^n$, which satisfies the following conditions:

$$F(u) \geq 0, \quad u \geq 0, \quad u^T F(u) = 0.$$

In this paper, we consider a class of NCP derived from the following problem in a region $D \subset \mathbb{R}^2$:

$$
\begin{aligned}
&-\mathscr{L}v(x,y) + \psi\big(v(x,y)\big) + f(x,y) \geq 0, \\
&v(x,y) \geq 0, \\
&v(x,y)\big[-\mathscr{L}v(x,y) + \psi(v(x,y)) + f(x,y)\big] = 0,
\end{aligned}
$$

and such that $v(x,y) = g(x,y)$ on the boundary D. The function f, difference operator $\mathscr{L}$, nonlinear function ψ and the boundary condition g are given. In numerical solution of the problem, the unknown function $v(x,y)$ is approximated by a vector u, which is defined on a grid and satisfies the conditions

$$
\begin{aligned}
Au + \phi(u) &\geq 0, \\
u &\geq 0, \\
u^T\big[Au + \phi(u)\big] &= 0,
\end{aligned}
\tag{2.1}
$$

where $A \in \mathbb{R}^{n \times n}$ is a symmetric positive definite matrix. The nonlinear term $\phi(u)$ can be written as

$$\phi(u) = \big(\psi_1(u_1), \psi_2(u_2), \ldots, \psi_n(u_n)\big)^T + b$$

with a vector b generated by the boundary condition and the function $f(x,y)$. The coefficient matrix A in (2.1) usually has a special structure.

In this section, we propose various improvements of inexact alternating direction methods for finding the solution of a class of NCPs. It is worth noting that the condition

$$\frac{\mathrm{d}\psi_i}{\mathrm{d}u_i} \geq 0, \quad i = 1, 2, \ldots, n \tag{2.2}$$

is assumed throughout the paper. If A is a symmetric positive definite matrix and $\mathrm{d}\psi_i/\mathrm{d}u_i$, $i = 1, 2, \ldots, n$ are nonnegative, then according to [23], NCP has a unique solution u^*. Recall that u^* is a solution of (2.1) if and only if u^* solves the constrained programming problem

$$
\begin{aligned}
&\min S(u) = \frac{1}{2}u^T Au + \Phi(u), \\
&\text{subject to: } u \geq 0,
\end{aligned}
\tag{2.3}
$$

where

$$\Phi(u) = \sum_{i=1}^{n} \int_{-\infty}^{u_i} \big(\psi_i(u_i) + b_i\big)\mathrm{d}u_i$$

is convex on u because of the condition (2.2). Therefore we can consider the problem (2.3) instead of (2.1).

Let

$$\mathbb{R}_+^n := \{u \in \mathbb{R}^n : u_i \geq 0, \ i = 1, 2, \ldots, n\}$$

and $G(u)$ be the indicator function of $\mathbb{R}_+^n$, i.e.

$$G(u) = \begin{cases} 0, & \text{if} \quad u \in \mathbb{R}_+^n, \\ +\infty, & \text{otherwise.} \end{cases}$$

The arguments similar to [52, Lemma 1] show that the problem (2.3) can be equivalently written as

$$\begin{aligned} &\min \big\{S(u) + G(w)\big\}, \\ &\text{subject to: } \Lambda(w - u) = 0, \quad w \in \mathbb{R}_+^n, \end{aligned} \tag{2.4}$$

where Λ is a given diagonal matrix with positive diagonal entries. For simplicity, here we set $\Lambda = \mu I$, $\mu > 0$.

The augmented Lagrangian function of (2.4) has the form

$$\mathrm{L}(u, w, \lambda) = S(u) + G(w) + \lambda^T \Lambda(w - u) + \frac{\beta}{2} \|\Lambda(w - u)\|^2,$$

where $\lambda \in \mathbb{R}^n$ is the Lagrangian multiplier and $\beta > 0$ is the penalty parameter. The classical alternating direction method of multipliers for the problem (2.4) or, equivalently, for the problem (2.3) is

$$\begin{aligned} u^{(k+1)} &= \operatorname*{argmin}_{u \in \mathbb{R}^n} \mathrm{L}\big(u, w^{(k)}, \lambda^{(k)}\big), \\ w^{(k+1)} &= \operatorname*{argmin}_{w \in \mathbb{R}_+^n} \mathrm{L}\big(u^{(k+1)}, w, \lambda^{(k)}\big), \\ \lambda^{(k+1)} &= \lambda^{(k)} + \beta \mu \big(w^{(k+1)} - u^{(k+1)}\big). \end{aligned} \tag{2.5}$$

Considering the minimisation problems in (2.5), we have

$$\big(A + \beta \Lambda^2\big) u^{(k+1)} + \phi\big(u^{(k+1)}\big) = \Lambda \lambda^{(k)} + \beta \Lambda^2 w^{(k)}. \tag{2.6}$$

For the second minimisation problem in (2.5), we have

$$w^{(k+1)} = \left(u^{(k+1)} - \frac{\Lambda^{-1} \lambda^{(k)}}{\beta}\right)_+, \tag{2.7}$$

where $(.)_+$ denotes the projection on the set $\mathbb{R}_+^n$.

We note that it is difficult to determine $u^{(k+1)}$ directly from (2.6). Therefore, we will use an inexact approach — viz. we use the system of linear equations

$$\big(A + \beta \Lambda^2\big) u^{(k+1)} \approx \Lambda \lambda^{(k)} + \beta \Lambda^2 w^{(k)} - \phi(u^{(k)}), \tag{2.8}$$

where $u^{(k)}$, $w^{(k)}$ and $\lambda^{(k)}$ are given. The relations (2.7), (2.8) lead to the following algorithm.

Algorithm 2.1 Direct inexact ADM (DADM)

Input: A, ϕ, $\Lambda = \mu I$, $\mu > 0$, $\alpha > 0$ and $\beta > 0$; give initial points $u^{(0)}$, $w^{(0)}$ and $\lambda^{(0)}$.

While not converged, **do**

    Generate $u^{(k+1)}$ according to (2.8);

    Generate $w^{(k+1)}$ according to (2.7);

    Update $\lambda^{(k)}$ according to $\lambda^{(k+1)} = \lambda^{(k)} + \beta \Lambda (w^{(k+1)} - u^{(k+1)})$;

End While

It will be proven in the next section that $(u^{(k+1)}, w^{(k+1)})$ generated by (2.7), (2.8) converges to the solution (u^*, w^*) of the problem (2.4).

At each iteration in Algorithm 2.1, we can find the solution of the system of linear equations (2.8) by linear solvers. However, it is not always possible to solve it directly, especially for large sparse problems. This motivates us to exploit more implementable iterative methods, thus leading to another inexact alternating direction method. Here, we use SSOR iterations to solve (2.8) approximately, and propose the SSOR-based inexact ADM for the symmetric positive definite NCPs.

Noting that A is symmetric positive definite, we consider the splitting

$$A = D - L - U = D - U^T - L^T,$$

where D is a diagonal matrix, $-L$ and $-U$ are strictly lower-triangular and upper-triangular matrices of A, respectively. After that we apply a SSOR method to (2.8). More exactly, given $u^{(k)}$, $w^{(k)}$ and $\lambda^{(k)}$, we determine $u^{(k+1)}$ from the equations

$$
\begin{aligned}
&\left(D - \alpha L + \alpha \beta \Lambda^2\right) u^{(k+1/2)} \\
&= \left((1-\alpha)D + \alpha U\right) u^{(k)} + \alpha \left(\Lambda \lambda^{(k)} + \beta \Lambda^2 w^{(k)} - \phi\left(u^{(k)}\right)\right), \\
&\left(D - \alpha U + \alpha \beta \Lambda^2\right) u^{(k+1)} \\
&= \left((1-\alpha)D + \alpha L\right) u^{(k+1/2)} + \alpha \left(\Lambda \lambda^{(k)} + \beta \Lambda^2 w^{(k)} - \phi\left(u^{(k)}\right)\right).
\end{aligned}
\tag{2.9}
$$

Taking into account (2.7) and (2.9), we arrive at another algorithm.

Algorithm 2.2 SSOR-based inexact ADM (SADM)

Input: D, L, U with $A = D - L - U$, ϕ, $\Lambda = \mu I$, $\mu > 0$, $\alpha > 0$ and $\beta > 0$; give initial points $u^{(0)}$, $w^{(0)}$ and $\lambda^{(0)}$.

While not converged, **do**

    Generate $u^{(k+1)}$ according to (2.9);

    Generate $w^{(k+1)}$ according to (2.7);

    Update $\lambda^{(k)}$ according to $\lambda^{(k+1)} = \lambda^{(k)} + \beta \Lambda (w^{(k+1)} - u^{(k+1)})$;

End While

We propose another type of efficient SSOR-based inexact ADM by using a different SSOR iteration for solving $u^{(k+1)}$ in (2.8). It is called the modified SSOR-based inexact ADM for

solving symmetric positive definite NCPs. Its effectiveness and comparison to SADM is tested by numerical experiments in Section 4.

Noting that $\tilde{A} = A + \beta\Lambda^2$ is also a symmetric positive definite matrix, we consider the splitting

$$\tilde{A} = \tilde{D} - L - U = \tilde{D} - U^T - L^T,$$

where $\tilde{D}$ is a diagonal matrix, and $-L$, $-U$ are strictly lower-triangular and upper-triangular matrices of A, respectively. After that, we employ a modified SSOR method to solve (2.8). More exactly, given $u^{(k)}$, $w^{(k)}$ and $\lambda^{(k)}$, we find $u^{(k+1)}$ from the equations

$$
\begin{aligned}
(\tilde{D} - \alpha L)u^{(k+1/2)} &= \big((1-\alpha)\tilde{D} + \alpha U\big)u^{(k)} + \alpha\big(\Lambda\lambda^{(k)} + \beta\Lambda^2 w^{(k)} - \phi\big(u^{(k)}\big)\big), \\
(\tilde{D} - \alpha U)u^{(k+1)} &= \big((1-\alpha)\tilde{D} + \alpha L\big)u^{(k+1/2)} + \alpha\big(\Lambda\lambda^{(k)} + \beta\Lambda^2 w^{(k)} - \phi\big(u^{(k)}\big)\big).
\end{aligned}
\tag{2.10}
$$

Exploiting (2.7) and (2.10), we finally get Algorithm 2.3.

Algorithm 2.3 Modified SSOR-based inexact ADM (MSADM)

Input: $\tilde{D}$, L, U with $(A + \beta\Lambda^2) = \tilde{D} - L - U$, ϕ, $\Lambda = \mu I$, $\mu > 0$, $\alpha > 0$ and $\beta > 0$; give initial
  points $u^{(0)}$, $w^{(0)}$ and $\lambda^{(0)}$.
  While not converged, **do**
    Generate $u^{(k+1)}$ according to (2.10);
    Generate $w^{(k+1)}$ according to (2.7);
    Update $\lambda^{(k)}$ according to $\lambda^{(k+1)} = \lambda^{(k)} + \beta\Lambda(w^{(k+1)} - u^{(k+1)})$;
  End While

---

3. Convergence

In this section, we show that the sequence $(u^{(k)}, w^{(k)})$ generated by Algorithms 2.1-2.3 converges to the solution (u^*, w^*) of the problem (2.4). Recall that, according to the discussion in Section 2 the solution u^* of (2.4) is also a solution of (2.1). Note that the solving of (2.4) is equivalent to the finding of $(u^*, w^*, \lambda^*) \in \mathbb{R}^n \times \mathbb{R}^n_+ \times \mathbb{R}^n$, which satisfies the relations

$$
\begin{aligned}
\big(Au^* + \phi(u^*)\big) - \Lambda\lambda^* &= 0, \\
(w - w^*)^T \Lambda\lambda^* &\geq 0, \quad \forall w \in \mathbb{R}^n_+, \\
\Lambda(u^* - w^*) &= 0.
\end{aligned}
\tag{3.1}
$$

Let $e_u^{(k)} = u^{(k)} - u^*$, $e_w^{(k)} = w^{(k)} - w^*$, $e_\lambda^{(k)} = \lambda^{(k)} - \lambda^*$ and

$$
\begin{aligned}
D_\phi^{(k)} &= \operatorname{diag}\left(\frac{\phi_1\big(u_1^{(k)}\big) - \phi_1(u_1^*)}{u_1^{(k)} - u_1^*}, \ldots, \frac{\phi_n\big(u_n^{(k)}\big) - \phi_n(u_n^*)}{u_n^{(k)} - u_n^*}\right) \\
&= \operatorname{diag}\big(\phi_1'\big(\xi_1^{(k)}\big), \ldots, \phi_n'\big(\xi_n^{(k)}\big)\big),
\end{aligned}
$$

where $\xi_i^{(k)}$ is located between $u_i^{(k)}$ and u_i^*, $i = 1, \ldots, n$.

Theorem 3.1. *Let $\phi_i'(x) \geq 0$, $i = 1, 2, \ldots, n$ for any $x \in \mathbb{R}$ and*

$$d_{\dot{\phi}} := \max_{i=1,\ldots,n} \left\{ \max_{x \in \mathbb{R}} \left\{ \phi_i'(x) \right\} \right\}.$$

If $S := A - d_{\dot{\phi}} I$ is a symmetric positive definite matrix, then the sequence $(u^{(k)}, w^{(k)})$ generated by Algorithm 2.1 converges to the solution (u^*, w^*) of the problem (2.4).

Proof. It follows from (2.8) that

$$\left(u^* - u^{(k+1)} \right)^T \left[Au^{(k+1)} + \beta \Lambda^2 \left(u^{(k+1)} - w^{(k)} \right) - \Lambda \lambda^{(k)} + \phi\left(u^{(k)} \right) \right] = 0,$$

and (3.1) yields

$$\left(u^{(k+1)} - u^* \right)^T \left[Au^* + \beta \Lambda^2 (u^* - w^*) - \Lambda \lambda^* + \phi(u^*) \right] = 0.$$

Consequently, we obtain

$$\left(u^* - u^{(k+1)} \right)^T \left[Ae_u^{(k+1)} + \beta \Lambda^2 \left(e_u^{(k+1)} - e_w^{(k)} \right) - \Lambda e_\lambda^{(k)} + \phi\left(u^{(k)} \right) - \phi(u^*) \right] = 0,$$

or

$$2\beta \left\langle e_u^{(k+1)}, Ae_u^{(k+1)} \right\rangle + 2\beta^2 \Lambda^2 \left\langle e_u^{(k+1)}, e_u^{(k+1)} \right\rangle - 2\beta^2 \Lambda^2 \left\langle e_u^{(k+1)}, e_w^{(k)} \right\rangle$$
$$- 2\beta \left\langle e_u^{(k+1)}, \Lambda e_\lambda^{(k)} \right\rangle + 2\beta \left\langle e_u^{(k+1)}, \phi(u^{(k)}) - \phi(u^*) \right\rangle = 0, \tag{3.2}$$

where

$$\langle x, y \rangle := x^T y = \sum_{i=1}^{n} x_i y_i, \quad x, y \in \mathbb{R}^n$$

is the inner product in $\mathbb{R}^n$. For compactness, the notation $\langle x, x \rangle = \|x\|^2$ is used whenever possible, and $\| \cdot \|$ denotes the Euclidean norm on $\mathbb{R}^n$.

The equation

$$\lambda^{(k+1)} = \lambda^{(k)} + \beta \Lambda \left(w^{(k+1)} - u^{(k+1)} \right)$$

and the third equation in (3.1) give

$$e_\lambda^{(k+1)} = e_\lambda^{(k)} + \beta \Lambda \left(e_w^{(k+1)} - e_u^{(k+1)} \right).$$

Therefore,

$$\left\| e_\lambda^{(k)} \right\|^2 - \left\| e_\lambda^{(k+1)} \right\|^2 = -2\beta \left\langle e_\lambda^{(k)}, \Lambda \left(e_w^{(k+1)} - e_u^{(k+1)} \right) \right\rangle - \beta^2 \left\| \Lambda \left(e_w^{(k+1)} - e_u^{(k+1)} \right) \right\|^2.$$

Following [52], we have

$$2\beta^2 \left\| \Lambda e_w^{(k+1)} \right\|^2 - 2\beta^2 \left\langle \Lambda e_w^{(k+1)}, \Lambda e_u^{(k+1)} \right\rangle + 2\beta \left\langle \Lambda e_w^{(k+1)}, e_\lambda^{(k)} \right\rangle \leq 0,$$

which implies

$$\left\| e_\lambda^{(k)} \right\|^2 - \left\| e_\lambda^{(k+1)} \right\|^2 \geq 2\beta \left\langle e_\lambda^{(k)}, \Lambda e_u^{(k+1)} \right\rangle + \beta^2 \left\| \Lambda e_w^{(k+1)} \right\|^2 - \beta^2 \left\| \Lambda e_u^{(k+1)} \right\|^2. \tag{3.3}$$

Combining it with (3.2), we arrive at the estimate

$$
\begin{aligned}
\left\| e_\lambda^{(k)} \right\|^2 - \left\| e_\lambda^{(k+1)} \right\|^2 &\geq \beta^2 \left\| \Lambda e_w^{(k+1)} \right\|^2 + \beta^2 \left\| \Lambda e_u^{(k+1)} \right\|^2 - 2\beta^2 \Lambda^2 \left\langle e_u^{(k+1)}, e_w^{(k)} \right\rangle \\
&\quad + 2\beta \left\langle e_u^{(k+1)}, A e_u^{(k+1)} \right\rangle + 2\beta \left\langle e_u^{(k+1)}, \phi(u^{(k)}) - \phi(u^*) \right\rangle.
\end{aligned}
$$

Consequently,

$$
\begin{aligned}
&\left\| e_\lambda^{(k)} \right\|^2 - \left\| e_\lambda^{(k+1)} \right\|^2 + \beta^2 \left(\left\| \Lambda e_w^{(k)} \right\|^2 - \left\| \Lambda e_w^{(k+1)} \right\|^2 \right) \\
&\geq \beta^2 \left\| \Lambda \big(e_w^{(k)} - e_u^{(k+1)} \big) \right\|^2 + 2\beta \left\langle e_u^{(k+1)}, A e_u^{(k+1)} \right\rangle + 2\beta \left\langle e_u^{(k+1)}, D_{\dot\phi}^{(k)} e_u^{(k)} \right\rangle.
\end{aligned}
$$

Let $\phi_i'(x) \geq 0$, $i = 1, 2, \ldots, n$ for any $x \in \mathbb{R}$. The Cauchy-Schwartz inequality yields

$$
\begin{aligned}
2\beta \left\langle e_u^{(k+1)}, D_{\dot\phi}^{(k)} e_u^{(k)} \right\rangle &\geq -2\beta \left\| \left(D_{\dot\phi}^{(k)} \right)^{\frac{1}{2}} e_u^{(k+1)} \right\| \left\| \left(D_{\dot\phi}^{(k)} \right)^{\frac{1}{2}} e_u^{(k)} \right\| \\
&\geq -\beta \left\| \left(D_{\dot\phi}^{(k)} \right)^{\frac{1}{2}} e_u^{(k+1)} \right\|^2 - \beta \left\| \left(D_{\dot\phi}^{(k)} \right)^{\frac{1}{2}} e_u^{(k)} \right\|^2 \\
&\geq -\beta d_{\dot\phi} \left\| e_u^{(k+1)} \right\|^2 - \beta d_{\dot\phi} \left\| e_u^{(k)} \right\|^2,
\end{aligned}
$$

and it follows that

$$
\begin{aligned}
&\left\| e_\lambda^{(k)} \right\|^2 - \left\| e_\lambda^{(k+1)} \right\|^2 + \beta^2 \left(\left\| \Lambda e_w^{(k)} \right\|^2 - \left\| \Lambda e_w^{(k+1)} \right\|^2 \right) \\
&\geq \beta^2 \left\| \Lambda \big(e_w^{(k)} - e_u^{(k+1)} \big) \right\|^2 + 2\beta \left\langle e_u^{(k+1)}, A e_u^{(k+1)} \right\rangle + 2\beta \left\langle e_u^{(k+1)}, D_{\dot\phi}^{(k)} e_u^{(k)} \right\rangle \\
&\geq \beta^2 \left\| \Lambda \big(e_w^{(k)} - e_u^{(k+1)} \big) \right\|^2 + 2\beta \left\langle e_u^{(k+1)}, A e_u^{(k+1)} \right\rangle - \beta d_{\dot\phi} \left\| e_u^{(k+1)} \right\|^2 - \beta d_{\dot\phi} \left\| e_u^{(k)} \right\|^2.
\end{aligned}
$$

This inequality can be written in the form

$$
\begin{aligned}
&\left\| e_\lambda^{(k)} \right\|^2 - \left\| e_\lambda^{(k+1)} \right\|^2 + \beta^2 \left(\left\| \Lambda e_w^{(k)} \right\|^2 - \left\| \Lambda e_w^{(k+1)} \right\|^2 \right) + \beta d_{\dot\phi} \left(\left\| e_u^{(k)} \right\|^2 - \left\| e_u^{(k+1)} \right\|^2 \right) \\
&\geq \beta^2 \left\| \Lambda \big(e_w^{(k)} - e_u^{(k+1)} \big) \right\|^2 + 2\beta \left\langle e_u^{(k+1)}, (A - d_{\dot\phi} I) e_u^{(k+1)} \right\rangle. \tag{3.4}
\end{aligned}
$$

Setting

$$
E^{(k)} := \left\| e_\lambda^{(k)} \right\|^2 + \beta^2 \left\| \Lambda e_w^{(k)} \right\|^2 + \beta d_{\dot\phi} \left\| e_u^{(k)} \right\|^2,
$$

we write (3.4) as

$$
E^{(k)} - E^{(k+1)} \geq \beta^2 \left\| \Lambda \big(e_w^{(k)} - e_u^{(k+1)} \big) \right\|^2 + 2\beta \left\langle e_u^{(k+1)}, (A - d_{\dot\phi} I) e_u^{(k+1)} \right\rangle. \tag{3.5}
$$

Summing the inequalities (3.5) in k from 0 to n gives

$$
\begin{aligned}
E^{(0)} - E^{(n+1)} &= \left\| e_\lambda^{(0)} \right\|^2 - \left\| e_\lambda^{(n+1)} \right\|^2 + \beta^2 \left(\left\| \Lambda e_w^{(0)} \right\|^2 - \left\| \Lambda e_w^{(n+1)} \right\|^2 \right) \\
&\quad + \beta d_{\dot\phi} \left(\left\| e_u^{(0)} \right\|^2 - \left\| e_u^{(n+1)} \right\|^2 \right) \\
&\geq \sum_{k=0}^{n} \beta^2 \left\| \Lambda \big(e_w^{(k)} - e_u^{(k+1)} \big) \right\|^2 + \sum_{k=0}^{n} 2\beta \left\langle e_u^{(k+1)}, (A - d_{\dot\phi} I) e_u^{(k+1)} \right\rangle.
\end{aligned}
$$

Hence, we have

$$
\begin{aligned}
& \left\|e_\lambda^{(0)}\right\|^2 + \beta^2 \left\|\Lambda e_w^{(0)}\right\|^2 + \beta d_{\dot\phi} \left\|e_u^{(0)}\right\|^2 \\
& \geq \left\|e_\lambda^{(0)}\right\|^2 - \left\|e_\lambda^{(n+1)}\right\|^2 + \beta^2 \left(\left\|\Lambda e_w^{(0)}\right\|^2 - \left\|\Lambda e_w^{(n+1)}\right\|^2\right) \\
& \qquad + \beta d_{\dot\phi} \left(\left\|e_u^{(0)}\right\|^2 - \left\|e_u^{(n+1)}\right\|^2\right) - \sum_{k=0}^{n} \beta^2 \left\|\Lambda\left(e_w^{(k)} - e_u^{(k+1)}\right)\right\|^2 \\
& \geq \sum_{k=0}^{n} 2\beta \left\langle e_u^{(k+1)}, (A - d_{\dot\phi} I) e_u^{(k+1)}\right\rangle,
\end{aligned}
$$

where the assumption that $S = A - d_{\dot\phi} I$ is a symmetric positive definite matrix is used. Therefore, the series $\sum_{k=0}^{\infty} 2\beta \left\langle e_u^{(k+1)}, (A - d_{\dot\phi} I) e_u^{(k+1)}\right\rangle$ converges. Consequently, $2\beta \left\langle e_u^{(k+1)}, (A - d_{\dot\phi} I) e_u^{(k+1)}\right\rangle$ tends to 0, and Algorithm 2.1 converges. □

Theorem 3.2. *Under the conditions of Theorem* 3.1 *and* $0 < \alpha < 2$, *the sequence* $(u^{(k)}, w^{(k)})$ *generated by Algorithm* 2.2 *converges to the solution* (u^*, w^*) *of the problem* (2.4).

Proof. Combining the first equation in (3.1) and (2.9) gives

$$
\begin{aligned}
& \left(D - \alpha L + \alpha\beta\Lambda^2\right) e_u^{(k+1/2)} \\
& = \left((1 - \alpha)D + \alpha U\right) e_u^{(k)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)}\right) + \phi(u^*)\right), \\
& \left(D - \alpha U + \alpha\beta\Lambda^2\right) e_u^{(k+1)} \\
& = \left((1 - \alpha)D + \alpha L\right) e_u^{(k+1/2)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)}\right) + \phi(u^*)\right). \quad (3.6)
\end{aligned}
$$

Noting that $U = L^T$, we write

$$
\begin{aligned}
D - \alpha L + \alpha\beta\Lambda^2 &= \left((2 - \alpha)D + \alpha\beta\Lambda^2\right) - \left((1 - \alpha)D + \alpha L\right), \\
D - \alpha L^T + \alpha\beta\Lambda^2 &= \left((2 - \alpha)D + \alpha\beta\Lambda^2\right) - \left((1 - \alpha)D + \alpha L^T\right).
\end{aligned}
$$

Therefore, the Eq. (3.6) takes form

$$
\begin{aligned}
& \left[\left((2 - \alpha)D + \alpha\beta\Lambda^2\right) - \left((1 - \alpha)D + \alpha L\right)\right] e_u^{(k+1/2)} \\
& = \left((1 - \alpha)D + \alpha U\right) e_u^{(k)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)}\right) + \phi(u^*)\right), \\
& \left[\left((2 - \alpha)D + \alpha\beta\Lambda^2\right) - \left((1 - \alpha)D + \alpha L^T\right)\right] e_u^{(k+1)} \\
& = \left((1 - \alpha)D + \alpha L\right) e_u^{(k+1/2)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)}\right) + \phi(u^*)\right). \quad (3.7)
\end{aligned}
$$

Since $0 < \alpha < 2$, the matrix $(2 - \alpha)D + \alpha\beta\Lambda^2$ is invertible. Multiplying (3.7) by the corresponding inverse matrix gives

$$
\left[I - \left((2 - \alpha)D + \alpha\beta\Lambda^2\right)^{-1}\left((1 - \alpha)D + \alpha L\right)\right] e_u^{(k+1/2)}
$$

$$= \left((2-\alpha)D + \alpha\beta\Lambda^2 \right)^{-1} \left[\left((1-\alpha)D + \alpha U \right) e_u^{(k)} + \alpha \left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)} \right) + \phi(u^*) \right) \right],$$

$$\left[I - \left((2-\alpha)D + \alpha\beta\Lambda^2 \right)^{-1} \left((1-\alpha)D + \alpha L^T \right) \right] e_u^{(k+1)}$$
$$= \left((2-\alpha)D + \alpha\beta\Lambda^2 \right)^{-1} \left[\left((1-\alpha)D + \alpha L \right) e_u^{(k+1/2)} + \alpha \left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)} \right) + \phi(u^*) \right) \right].$$

Subtracting the first equality from the second one gives

$$\left[I - \left((2-\alpha)D + \alpha\beta\Lambda^2 \right)^{-1} \left((1-\alpha)D + \alpha L^T \right) \right] e_u^{(k+1)}$$
$$= e_u^{(k+1/2)} - \left((2-\alpha)D + \alpha\beta\Lambda^2 \right)^{-1} \left((1-\alpha)D + \alpha U \right) e_u^{(k)},$$

or equivalently,

$$\left((1-\alpha)D + \alpha L \right) \left[I - \left((2-\alpha)D + \alpha\beta\Lambda^2 \right)^{-1} \left((1-\alpha)D + \alpha L^T \right) \right] e_u^{(k+1)}$$
$$= \left((1-\alpha)D + \alpha L \right) \left[e_u^{(k+1/2)} - \left((2-\alpha)D + \alpha\beta\Lambda^2 \right)^{-1} \left((1-\alpha)D + \alpha U \right) e_u^{(k)} \right].$$

Using the notation

$$W = \left((1-\alpha)D + \alpha L \right) \left((2-\alpha)D + \alpha\beta\Lambda^2 \right)^{-1} \left((1-\alpha)D + \alpha L^T \right),$$

we write

$$\left[\left((1-\alpha)D + \alpha L \right) - W \right] e_u^{(k+1)} = \left((1-\alpha)D + \alpha L \right) e_u^{(k+1/2)} - W e_u^{(k)}.$$

This and the second equation in (3.6) give

$$\left[\left((1-\alpha)D + \alpha L \right) - W \right] e_u^{(k+1)}$$
$$= \left(D - \alpha U + \alpha\beta\Lambda^2 \right) e_u^{(k+1)} - \alpha \left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)} \right) + \phi(u^*) \right) - W e_u^{(k)}$$

or

$$\left(\alpha\beta\Lambda^2 + \alpha A + W \right) e_u^{(k+1)} = W e_u^{(k)} + \alpha \left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)} \right) + \phi(u^*) \right).$$

Multiplying the last equation by $e_u^{(k+1)}$ yields

$$\frac{2\beta}{\alpha} \left\langle e_u^{(k+1)}, \left(\alpha\beta\Lambda^2 + \alpha A + W \right) e_u^{(k+1)} \right\rangle$$
$$= \frac{2\beta}{\alpha} \left\langle e_u^{(k+1)}, W e_u^{(k)} + \alpha \left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)} \right) + \phi(u^*) \right) \right\rangle. \qquad (3.8)$$

Combining (3.8) and (3.3), we have

$$\left\| e_\lambda^{(k)} \right\|^2 - \left\| e_\lambda^{(k+1)} \right\|^2 \geq \beta^2 \left(\left\| \Lambda e_w^{(k+1)} \right\|^2 - 2 \left\langle e_u^{(k+1)}, \Lambda^2 e_w^{(k)} \right\rangle \right) - \beta^2 \left\| \Lambda e_u^{(k+1)} \right\|^2$$
$$+ \frac{2\beta}{\alpha} \left\langle e_u^{(k+1)}, \left(\alpha\beta\Lambda^2 + \alpha A + W \right) e_u^{(k+1)} \right\rangle$$
$$- \frac{2\beta}{\alpha} \left\langle e_u^{(k+1)}, W e_u^{(k)} + \alpha \left[- \phi\left(u^{(k)} \right) + \phi(u^*) \right] \right\rangle.$$

Therefore,

$$
\left\|e_\lambda^{(k)}\right\|^2 - \left\|e_\lambda^{(k+1)}\right\|^2 + \beta^2 \left(\left\|\Lambda e_w^{(k)}\right\|^2 - \left\|\Lambda e_w^{(k+1)}\right\|^2\right)
$$
$$
\geq \beta^2 \left\|\Lambda\left(e_w^{(k)} - e_u^{(k+1)}\right)\right\|^2 + \frac{2\beta}{\alpha}\left\langle e_u^{(k+1)}, (\alpha A + W)e_u^{(k+1)}\right\rangle
$$
$$
- \frac{2\beta}{\alpha}\left\langle e_u^{(k+1)}, We_u^{(k)} + \alpha\left[-D_{\dot\phi}^{(k)}\right]e_u^{(k)}\right\rangle.
$$

Set

$$
V := \left((2-\alpha)D + \alpha\beta\Lambda^2\right)^{-1/2}\left((1-\alpha)D + \alpha L^T\right).
$$

Since $0 < \alpha < 2$ and $\phi_i'(x) \geq 0$, $i = 1, 2, \ldots, n$ for all $x \in \mathbb{R}$, the Cauchy-Schwartz inequality implies

$$
-\frac{2\beta}{\alpha}\left\langle e_u^{(k+1)}, We_u^{(k)}\right\rangle \geq -\frac{2\beta}{\alpha}\left\|Ve_u^{(k+1)}\right\|\left\|Ve_u^{(k)}\right\|
$$
$$
\geq -\frac{\beta}{\alpha}\left\|Ve_u^{(k+1)}\right\|^2 - \frac{\beta}{\alpha}\left\|Ve_u^{(k)}\right\|^2 = -\frac{\beta}{\alpha}\left\langle e_u^{(k+1)}, We_u^{(k+1)}\right\rangle - \frac{\beta}{\alpha}\left\|Ve_u^{(k)}\right\|^2.
$$

Finally we get

$$
\left\|e_\lambda^{(k)}\right\|^2 - \left\|e_\lambda^{(k+1)}\right\|^2 + \beta^2\left(\left\|\Lambda e_w^{(k)}\right\|^2 - \left\|\Lambda e_w^{(k+1)}\right\|^2\right)
$$
$$
+ \frac{\beta}{\alpha}\left\|Ve_u^{(k)}\right\|^2 - \frac{\beta}{\alpha}\left\|Ve_u^{(k+1)}\right\|^2 + \beta d_{\dot\phi}\left(\left\|e_u^{(k)}\right\|^2 - \left\|e_u^{(k+1)}\right\|^2\right)
$$
$$
\geq \beta^2\left\|\Lambda\left(e_w^{(k)} - e_u^{(k+1)}\right)\right\|^2 + 2\beta\left\langle e_u^{(k+1)}, (A - d_{\dot\phi}I)e_u^{(k+1)}\right\rangle.
$$

Similar to the considerations in Theorem 3.1, we obtain that Algorithm 2.2 converges. $\quad\square$

Theorem 3.3. *Under the conditions of Theorem 3.1 and $0 < \alpha < 2$, the sequence $(u^{(k)}, w^{(k)})$ generated by Algorithm 2.3 converges to the solution (u^*, w^*) of the problem (2.4).*

Proof. Combining the first equation in (3.1) and (2.10) yields

$$
(\tilde{D} - \alpha L)e_u^{(k+1/2)} = \left((1-\alpha)\tilde{D} + \alpha U\right)e_u^{(k)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi(u^{(k)}) + \phi(u^*)\right),
$$
$$
(\tilde{D} - \alpha U)e_u^{(k+1)} = \left((1-\alpha)\tilde{D} + \alpha L\right)e_u^{(k+1/2)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi(u^{(k)}) + \phi(u^*)\right). \tag{3.9}
$$

Since $U = L^T$, we have

$$
\tilde{D} - \alpha L = (2-\alpha)\tilde{D} - \left((1-\alpha)\tilde{D} + \alpha L\right),
$$
$$
\tilde{D} - \alpha L^T = (2-\alpha)\tilde{D} - \left((1-\alpha)\tilde{D} + \alpha L^T\right).
$$

Thus, the Eq. (3.9) takes form

$$
\left[(2-\alpha)\tilde{D} - \left((1-\alpha)\tilde{D} + \alpha L\right)\right]e_u^{(k+1/2)}
$$
$$
= \left((1-\alpha)\tilde{D} + \alpha U\right)e_u^{(k)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi(u^{(k)}) + \phi(u^*)\right),
$$
$$
\left[(2-\alpha)\tilde{D} - \left((1-\alpha)\tilde{D} + \alpha L^T\right)\right]e_u^{(k+1)}
$$
$$
= \left((1-\alpha)\tilde{D} + \alpha L\right)e_u^{(k+1/2)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi(u^{(k)}) + \phi(u^*)\right).
$$

Since $0 < \alpha < 2$, the matrix $(2-\alpha)\tilde{D}$ is invertible, and we can write

$$\left[I - \left((2-\alpha)\tilde{D}\right)^{-1}\left((1-\alpha)\tilde{D} + \alpha L\right)\right]e_u^{(k+1/2)}$$
$$= \left((2-\alpha)\tilde{D}\right)^{-1}\left[\left((1-\alpha)\tilde{D} + \alpha U\right)e_u^{(k)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)}\right) + \phi(u^*)\right)\right],$$
$$\left[I - \left((2-\alpha)\tilde{D}\right)^{-1}\left((1-\alpha)\tilde{D} + \alpha L^T\right)\right]e_u^{(k+1)}$$
$$= \left((2-\alpha)\tilde{D}\right)^{-1}\left[\left((1-\alpha)\tilde{D} + \alpha L\right)e_u^{(k+1/2)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)}\right) + \phi(u^*)\right)\right].$$

Subtracting the first equation from the second one gives

$$\left[I - \left((2-\alpha)\tilde{D}\right)^{-1}\left((1-\alpha)\tilde{D} + \alpha L^T\right)\right]e_u^{(k+1)}$$
$$= e_u^{(k+1/2)} - \left((2-\alpha)\tilde{D}\right)^{-1}\left((1-\alpha)\tilde{D} + \alpha U\right)e_u^{(k)}$$

or

$$\left((1-\alpha)\tilde{D} + \alpha L\right)\left[I - \left((2-\alpha)\tilde{D}\right)^{-1}\left((1-\alpha)\tilde{D} + \alpha L^T\right)\right]e_u^{(k+1)}$$
$$= \left((1-\alpha)\tilde{D} + \alpha L\right)\left[e_u^{(k+1/2)} - \left((2-\alpha)\tilde{D}\right)^{-1}\left((1-\alpha)\tilde{D} + \alpha U\right)e_u^{(k)}\right].$$

Setting

$$W = \left((1-\alpha)\tilde{D} + \alpha L\right)\left((2-\alpha)\tilde{D}\right)^{-1}\left((1-\alpha)\tilde{D} + \alpha L^T\right),$$

we can write

$$\left[\left((1-\alpha)\tilde{D} + \alpha L\right) - W\right]e_u^{(k+1)} = \left((1-\alpha)\tilde{D} + \alpha L\right)e_u^{(k+1/2)} - We_u^{(k)}.$$

Combining this with the second equation in (3.9), we have

$$\left[\left((1-\alpha)\tilde{D} + \alpha L\right) - W\right]e_u^{(k+1)}$$
$$= \left(\tilde{D} - \alpha U\right)e_u^{(k+1)} - \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)}\right) + \phi(u^*)\right) - We_u^{(k)},$$

or

$$\left(\alpha\beta\Lambda^2 + \alpha A + W\right)e_u^{(k+1)} = We_u^{(k)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)}\right) + \phi(u^*)\right).$$

Multiplying it by $e_u^{(k+1)}$ gives

$$\frac{2\beta}{\alpha}\left\langle e_u^{(k+1)}, \left(\alpha\beta\Lambda^2 + \alpha A + W\right)e_u^{(k+1)}\right\rangle$$
$$= \frac{2\beta}{\alpha}\left\langle e_u^{(k+1)}, We_u^{(k)} + \alpha\left(\Lambda e_\lambda^{(k)} + \beta\Lambda^2 e_w^{(k)} - \phi\left(u^{(k)}\right) + \phi(u^*)\right)\right\rangle. \qquad (3.10)$$

Combining (3.10) and (3.3) yields

$$\left\|e_\lambda^{(k)}\right\|^2 - \left\|e_\lambda^{(k+1)}\right\|^2 \geq \beta^2\left(\left\|\Lambda e_w^{(k+1)}\right\|^2 - 2\left\langle e_u^{(k+1)}, \Lambda^2 e_w^{(k)}\right\rangle\right) - \beta^2\left\|\Lambda e_u^{(k+1)}\right\|^2$$
$$+ \frac{2\beta}{\alpha}\left\langle e_u^{(k+1)}, \left(\alpha\beta\Lambda^2 + \alpha A + W\right)e_u^{(k+1)}\right\rangle$$
$$- \frac{2\beta}{\alpha}\left\langle e_u^{(k+1)}, We_u^{(k)} + \alpha\left[-\phi\left(u^{(k)}\right) + \phi(u^*)\right]\right\rangle.$$

Therefore,

$$\left\|e_\lambda^{(k)}\right\|^2 - \left\|e_\lambda^{(k+1)}\right\|^2 + \beta^2 \left(\left\|\Lambda e_w^{(k)}\right\|^2 - \left\|\Lambda e_w^{(k+1)}\right\|^2\right)$$

$$\geq \beta^2 \left\|\Lambda\left(e_w^{(k)} - e_u^{(k+1)}\right)\right\|^2 + \frac{2\beta}{\alpha}\left\langle e_u^{(k+1)}, (\alpha A + W)e_u^{(k+1)}\right\rangle$$

$$- \frac{2\beta}{\alpha}\left\langle e_u^{(k+1)}, We_u^{(k)} + \alpha\left[-D_{\dot\phi}^{(k)}\right]e_u^{(k)}\right\rangle.$$

Setting

$$V := \left((2-\alpha)\tilde{D}\right)^{-1/2}\left((1-\alpha)\tilde{D} + \alpha L^T\right)$$

and using the Cauchy-Schwartz inequality gives

$$-\frac{2\beta}{\alpha}\left\langle e_u^{(k+1)}, We_u^{(k)}\right\rangle \geq -\frac{2\beta}{\alpha}\left\|Ve_u^{(k+1)}\right\|\left\|Ve_u^{(k)}\right\|$$

$$\geq -\frac{\beta}{\alpha}\left\|Ve_u^{(k+1)}\right\|^2 - \frac{\beta}{\alpha}\left\|Ve_u^{(k)}\right\|^2 = -\frac{\beta}{\alpha}\left\langle e_u^{(k+1)}, We_u^{(k+1)}\right\rangle - \frac{\beta}{\alpha}\left\|Ve_u^{(k)}\right\|^2.$$

Finally we get

$$\left\|e_\lambda^{(k)}\right\|^2 - \left\|e_\lambda^{(k+1)}\right\|^2 + \beta^2 \left(\left\|\Lambda e_w^{(k)}\right\|^2 - \left\|\Lambda e_w^{(k+1)}\right\|^2\right)$$

$$+ \frac{\beta}{\alpha}\left\|Ve_u^{(k)}\right\|^2 - \frac{\beta}{\alpha}\left\|Ve_u^{(k+1)}\right\|^2 + \beta d_{\dot\phi}\left(\left\|e_u^{(k)}\right\|^2 - \left\|e_u^{(k+1)}\right\|^2\right)$$

$$\geq \beta^2 \left\|\Lambda\left(e_w^{(k)} - e_u^{(k+1)}\right)\right\|^2 + 2\beta\left\langle e_u^{(k+1)}, (A - d_{\dot\phi}I)e_u^{(k+1)}\right\rangle.$$

Considerations similar to the ones in Theorem 3.1 show that Algorithm 2.3 converges. □

4. Numerical Experiments

As is shown in [7,25,30,31,45,53], the modulus-based matrix splitting iteration methods perform better than certain popular iteration methods such as projected type methods [12] and modified modulus methods [15] applied to LCPs, and they are more efficient than the Noor fixed point approach [38] and the Fischer-Burmeister semismooth Newton method [13] applied to NCPs.

We provide a brief review of the modulus-based Jacobi method (MJ), modulus-based Gauss-Seidel method (MGS), modulus-based successive overrelaxation method (MSOR), modulus-based accelerated overrelaxation method (MAOR) [25,45] and the inexact alternating direction method of multipliers (IADM) [24] for solving NCPs and compare them with the methods studied in this paper. Abbreviations of all testing methods are listed in Table 1.

We note that the modulus-based matrix splitting iteration methods are based on a splitting $A = F - G$ and the iterations used are

$$(\Omega + F)x^{(k+1)} = Gx^{(k)} + (\Omega - A)\left|x^{(k)}\right| - \gamma\left[b + \phi\left(\frac{1}{\gamma}\left(\left|x^{(k)}\right| + x^{(k)}\right)\right)\right]$$

Table 1: Method abbreviations.

Abbreviation	Description
DADM	Proposed Algorithm 1
SADM	Proposed Algorithm 2
MSADM	Proposed Algorithm 3
IADM	The inexact alternating direction method of multipliers
MAOR	The modulus-based accelerated overrelaxation method
MSOR	The modulus-based successive overrelaxation method
MGS	The modulus-based Gauss-Seidel method
MJ	The modulus-based Jacobi method

with

$$u^{(k+1)} = \frac{1}{\gamma}\left(\left|x^{(k+1)}\right| + x^{(k+1)}\right).$$

We also note that:

(a) If $F = D$, $G = L + U$, we obtain the modulus-based Jacobi method.

(b) If $F = D - L$, $G = U$, we obtain the modulus-based Gauss-Seidel method.

(c) If $F = (1/\alpha)D - L$, $G = (1/\alpha - 1)D + U$, we obtain the modulus-based successive overrelaxation method.

(d) If $F = (1/\alpha)(D - \beta L)$, $G = (1/\alpha)[(1-\alpha)D + (\alpha - \beta)L + \alpha U]$, we obtain the modulus-based accelerated overrelaxation method.

In some practical applications, the discretisation matrix can be represented in the form $A = H + V$, where H and V correspond to the discretisation matrices of the differential operator in each spatial direction [24, 52]. Furthermore, both H and V are symmetric positive definite, large and sparse. The iteration of IADM is given as follows:

$$\begin{aligned}
\left(H + \beta\mu^2 I\right)u^{(k+1/2)} &= -Vu^{(k)} + \left(\mu\lambda^{(k)} + \beta\mu^2 w^{(k)} - \phi\left(u^{(k)}\right)\right), \\
\left(V + \beta\mu^2 I\right)u^{(k+1)} &= -Hu^{(k+1/2)} + \left(\mu\lambda^{(k)} + \beta\mu^2 w^{(k)} - \phi\left(u^{(k)}\right)\right).
\end{aligned} \tag{4.1}$$

Algorithm 4.1 Inexact alternating direction method of multipliers (IADM)

Input: H, V with $A = H + V$, ϕ, μ and $\beta > 0$; give initial points $u^{(0)}$, $w^{(0)}$ and $\lambda^{(0)}$.
  While not converged, **do**
    Generate $u^{(k+1)}$ according to (4.1);
    Generate $w^{(k+1)}$ according to $w^{(k+1)} = (u^{(k+1)} - \mu^{-1}\lambda^{(k)}/\beta)_+$;
    Update $\lambda^{(k)}$ according to $\lambda^{(k+1)} = \lambda^{(k)} + \beta\mu(w^{(k+1)} - u^{(k+1)})$;
  End While

We next compare our methods with MJ, MGS, MSOR, MAOR and IADM. Let

$$\text{RES}\left(u^{(k)}\right) := \left\|\min\left(Au^{(k)} + \phi\left(u^{(k)}\right), u^{(k)}\right)\right\|,$$

where the minimum is taken componentwise. We set $\mu = 1$ in DADM, SADM and MSADM. All other parameters are obtained experimentally by minimising the corresponding iteration steps. Moreover, $\gamma = 2$ and all initial vectors are $(0, 0, \ldots, 0)^T \in \mathbb{R}^n$. Let IT denote the number of iterations and CPU the elapsed CPU time in seconds. Finally, for all methods the stopping criteria are RES $\leq 10^{-6}$ or IT > 10000.

Example 4.1 (cf. Refs. [25, 33, 45, 54, 56, 59]). Consider $A = \hat{A} + \sigma I \in \mathbb{R}^{n \times n}$ and

$$b = -Az - \left(\psi_1(z_1), \psi_2(z_2), \ldots, \psi_n(z_n)\right)^T \in \mathbb{R}^n,$$

where the matrix $\hat{A} \in \mathbb{R}^{n \times n}$ has the form

$$\hat{A} = \begin{pmatrix} S & -I & 0 & \cdots & 0 & 0 \\ -I & S & -I & \cdots & 0 & 0 \\ 0 & -I & S & \ddots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -I & S & -I \\ 0 & 0 & \cdots & 0 & -I & S \end{pmatrix}$$

with the identity matrix $I \in \mathbb{R}^{m \times m}$, the tri-diagonal $m \times m$ matrix

$$S = \begin{pmatrix} 4 & -1 & 0 & \cdots & 0 & 0 \\ -1 & 4 & -1 & \cdots & 0 & 0 \\ 0 & -1 & 4 & \ddots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 4 & -1 \\ 0 & 0 & \cdots & 0 & -1 & 4 \end{pmatrix},$$

and with

$$z = [1, 2, 1, 2, \ldots, 1, 2, \ldots]^T \in \mathbb{R}^n.$$

For experiments, we choose two options $\sigma = 0$, $\psi(z) = \arctan(z)$ and $\sigma = 4$, $\psi(z) = \ln(1 + e^z)$.

Tables 2 and 3 show the corresponding numerical results. We note that the new DADM requires the least number of iterations. However, MSADM is the fastest at all sizes in Example 4.1 and it can be almost 2 times faster than MAOR. On the other hand, although SADM is slightly slower than MSADM, it is still faster and takes fewer iterations than the existing methods at all time. If $\sigma = 0$, $\psi(z) = \arctan(z)$, the matrix $S := A - d_{\dot{\phi}} I$ is not positive definite. Nevertheless, the methods proposed still work.

Table 2: Example 4.1. Numerical results, $\sigma = 0$, $\psi(z) = \arctan(z)$.

m	300			500			700		
	CPU	IT	RES	CPU	IT	RES	CPU	IT	RES
DADM	1.67	**11**	3.47e-07	4.90	**11**	5.92e-07	10.78	**11**	8.36e-07
SADM	**0.09**	17	4.79e-07	**0.38**	17	6.09e-07	0.77	17	7.63e-07
MSADM	**0.09**	17	5.35e-07	**0.38**	17	6.13e-07	**0.76**	17	9.33e-07
IADM	2.37	42	6.53e-07	9.88	43	6.82e-07	19.51	43	9.41e-07
MAOR	0.17	39	9.55e-07	0.75	40	7.43e-07	1.46	40	9.93e-07
MSOR	0.48	121	9.00e-07	2.15	124	9.48e-07	4.44	126	9.97e-07
MGS	0.49	121	9.96e-07	2.15	125	9.22e-07	4.47	127	9.61e-07
MJ	0.80	219	9.81e-07	3.46	226	9.51e-07	7.16	230	9.47e-07

Table 3: Example 4.1. Numerical results, $\sigma = 4$, $\psi(z) = \ln(1 + e^z)$.

m	300			500			700		
	CPU	IT	RES	CPU	IT	RES	CPU	IT	RES
DADM	0.87	**6**	1.56e-08	3.38	**6**	2.60e-08	7.88	**6**	2.06e-07
SADM	**0.04**	**6**	6.76e-07	**0.16**	**6**	4.29e-07	0.33	**6**	7.30e-07
MSADM	**0.04**	**6**	3.99e-07	**0.16**	**6**	3.79e-07	**0.32**	**6**	5.18e-07
IADM	1.46	26	8.41e-07	5.72	27	8.02e-07	12.65	27	7.03e-07
MAOR	0.07	13	7.78e-07	0.29	13	8.29e-07	0.59	13	4.00e-07
MSOR	0.10	19	9.98e-07	0.42	20	5.42e-07	0.90	20	6.83e-07
MGS	0.10	19	9.85e-07	0.45	20	6.95e-07	0.89	20	6.89e-07
MJ	0.12	26	5.41e-07	0.54	26	8.91e-07	1.13	27	5.34e-07

Example 4.2 (cf. Harker & Pang [24])**.** Let M be a given positive integer, $m = 2^M - 1$, $n = m^2$, $h = 1/m + 1$, and $A = H + V \in \mathbb{R}^{n \times n}$, where $H = I \otimes V_1$, $V = V_1 \otimes I$,

$$V_1 = \frac{1}{h^2} \times \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 & 0 \\ -1 & 2 & -1 & \cdots & 0 & 0 \\ 0 & -1 & 2 & \ddots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 2 & -1 \\ 0 & 0 & \cdots & 0 & -1 & 2 \end{pmatrix} \in \mathbb{R}^{m \times m},$$

$I \in \mathbb{R}^{m \times m}$ is the unit matrix and $\otimes$ the Kronecker product. The nonlinear term is $\phi(u) = u - \sin u + b$, in which the entries of b are ranging from 0 to 10. More precisely,

$$b = -\big[0, h_1, 2h_1, \ldots, 10, 0, h_1, 2h_1, \ldots, 10, \ldots \ldots, 0, h_1, 2h_1, \ldots, 10\big] \in \mathbb{R}^n,$$

where $h_1 = 10/(m - 1)$.

We test the problem for different mesh-step sizes h. Table 4, where the symbol '-' indicates that the number of iterations exceeds 10000, shows that the proposed DADM is the fastest method and it requires the smallest number of iterations in all cases. MSADM and SADM are still faster than MJ, MGS, MSOR and IADM despite longer computation time than MAOR. Besides, DADM can be at least 30 times faster than the existing methods at M=9, which is a huge advantage. We also note that MJ, MGS, MSOR were not able to determine the solution within 10000 iterations, and all methods, except of DADM, required more iterations to converge than in Example 4.1. It is related to the difference of vectors b, nonlinear terms and the form of the matrix A. Our experiments demonstrate that the structure of A is the major factor influencing the behavior of the methods. In contrast to Example 4.1, the problem considered arises from the discretisation of a free boundary problem. Therefore, one needs to divide the matrix A by h^2 here and this is the main reason why more iterations than in Example 4.1 are needed.

Table 4: Example 4.2. Numerical results.

M	7			8			9		
	CPU	IT	RES	CPU	IT	RES	CPU	IT	RES
DADM	**0.06**	**3**	7.85E-07	**0.32**	**3**	5.74E-07	**1.51**	**3**	7.43E-07
SADM	0.46	636	9.73E-07	4.54	1329	9.85E-07	67.69	2776	9.99E-07
MSADM	0.45	636	9.73E-07	4.54	1329	9.84E-07	65.06	2776	9.98E-07
IADM	3.78	624	9.80E-07	48.92	1257	9.88E-07	577.95	2551	9.98E-07
MAOR	0.34	541	9.86E-07	3.23	1161	9.90E-07	49.52	2386	1.00E-06
MSOR	-	-	-	-	-	-	-	-	-
MGS	-	-	-	-	-	-	-	-	-
MJ	-	-	-	-	-	-	-	-	-

Summarising we note the following issues.

1. All new methods considered here converge in all cases. They have advantage in different situations — cf. the discussion above. In particular, for large size problems they are more efficient that the existing MJ, MGS, MSOR, MAOR and IADM methods.

2. In practical implementation, we can consider a small size problem first to decide which one of new methods is the fastest and use it to solve the corresponding large size problem.

5. Conclusions

In order to find more efficient and feasible ways to solve symmetric positive definite weakly nonlinear complementarity problems, we improve inexact alternating direction methods. We prove the convergence of the methods. Numerical experiments show that the methods proposed have a number of advantages over similar existing methods. They

are more effective, faster, and provide more accurate results in a shorter time, especially for large size problems.

Acknowledgments

References

[1] L. Badea, X.C. Tai and J. Wang, *Convergence rate analysis of a multiplicative Schwarz method for variational inequalities*, SIAM J. Numer. Anal. **41**, 1052–1073 (2003).

[2] Z.Z. Bai, *The monotone convergence of a class of parallel nonlinear relaxation methods for nonlinear complementarity problems*, Comput. Math. Appl. **31**, 17–33 (1996).

[3] Z.Z. Bai, *New comparison theorem for the nonlinear multisplitting relaxation method for the nonlinear complementarity problems*, Comput. Math. Appl. **32**, 41–48 (1996).

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

[5] Z.Z. Bai, *A class of asynchronous parallel nonlinear accelerated over relaxation methods for the nonlinear complementarity problem*, J. Comput. Appl. Math. **93**, 35–44 (1998).

[6] Z.Z. Bai, *Asynchronous parallel nonlinear multisplitting relaxation methods for the large sparse nonlinear complementarity problems*, Appl. Math. Comput. **92**, 85–100 (1998).

[7] Z.Z. Bai, *Modulus-based matrix splitting iteration methods for linear complementarity problems*, Numer. Linear Algebra Appl. **17**, 917–933 (2010).

[8] Z.Z. Bai, V. Migallón, J. Penadés and D.B. Szyld, *Block and asynchronous two-stage methods for mildly nonlinear systems*, Numer. Math. **82**, 1–20 (1999).

[9] Z.Z. Bai and D.R. Wang, *A class of parallel nonlinear multisplitting relaxation methods for the large sparse nonlinear complementarity problems*, Comput. Math. Appl. **32**, 79–95 (1996).

[10] J.S. Chen and S. Pan, *A family of NCP functions and a descent method for the nonlinear complementarity problem*, Comput. Optim. Appl. **40**, 389–404 (2008).

[11] R.W. Cottle and G.B. Dantzig, *Complementary pivot theory of mathematical programming*, Linear Algebra Appl. **1**, 103–125 (1968).

[12] C. Cryer, *The solution of a quadratic programming using systematic overrelaxation*, J. SIAM Control. **9**, 385–392 (1971).

[13] T. Deluca, F. Facchinei and C. Kanzow, *A semismooth equation approach to the solution of nonlinear complementarity problems*, Math. Program. **75**, 407–439 (1996).

[14] J.L. Dong, J. Gao, F. Ju and J. Shen, *Modulus methods for nonnegatively constrained image restoration*, SIAM J. Imaging Sci. **9**, 1226–1246 (2016).

[15] J.L. Dong and M.Q. Jiang, *A modified modulus method for symmetric positive-definite linear complementarity problems*, Numer. Linear Algebra Appl. **16**, 129–143 (2009).

[16] S.Q. Du and Y. Gao, *Merit functions for nonsmooth complementarity problems and related descent algorithm*, Appl. Math. J. Chinese Univ. **25**, 78–84 (2010).

[17] F. Facchinei and J.S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer-Verlag (2003).

[18] M.C. Ferris and C. Kanzow, *Complementarity and related problems: A survey*, in: *Handbook of Applied Optimization*, P.M. Pardalos and M.G.C. Resende (Eds), pp. 514–530, Oxford University Press (2002).

[19] M.C. Ferris, O.L. Mangasarian and J.S. Pang, *Complementarity: Applications, Algorithms and Extensions*, Springer (2011).

[20] M.C. Ferris and J.S. Pang, *Engineering and economic applications of complementarity problems*, SIAM Rev. **39**, 669–713 (1997).

[21] D. Gabay and B. Mercier, *A dual algorithm for the solution of nonlinear variational problems via finite element approximation*, Comput. Math. Appl. **2**, 17–40 (1976).

[22] R. Glowinski, *Numerical Methods for Nonlinear Variational Problems*, Springer-Verlag (1984).

[23] P.T. Harker and J.S. Pang, *Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications*, Math. Program. **48**, 161–220 (1990).

[24] J.W. He, C.C. Lei, C.Y. Shi and S. Vong, *An inexact alternating direction method of multipliers for a kind of nonlinear complementarity problems*, Numer. Algebra, Control. Optim. **11**, 353–362 (2021).

[25] N. Huang and C.F. Ma, *The modulus-based matrix splitting algorithms for a class of weakly nonlinear complementarity problems*, Numer. Linear Algebra Appl. **23**, 558–569 (2016).

[26] Y.J. Jiang and J.P. Zeng, *A multiplicative Schwarz algorithm for the nonlinear complementarity problem with an M-function*, Bull. Aust. Math. Soc. **82**, 353–366 (2010).

[27] C. Kanzow, *Inexact semismooth Newton methods for large-scale complementarity problems*, Optim. Methods Softw. **19**, 309–325 (2004).

[28] C.E. Lemke and J.T. Howson, *Equilibrium points of bimatrix games*, SIAM J. Appl. Math. **12**, 413–423 (1964).

[29] R. Li and J.F. Yin, *Accelerated modulus-based matrix splitting iteration methods for a restricted class of nonlinear complementarity problems*, Numer. Algorithms **75**, 339–358 (2017).

[30] R. Li and J.F. Yin, *On the convergence of modulus-based matrix splitting iteration methods for a class of nonlinear complementarity problems with H_+-matrices*, J. Comput. Appl. Math. **342**, 202–209 (2018).

[31] W. Li, *A general modulus-based matrix splitting method for linear complementarity problems of H-matrices*, Appl. Math. Lett. **26**, 1159–1164 (2013).

[32] Z.Z. Li, Y.F. Ke, H. Zhang and R.S. Chu, *SOR-like iteration methods for second-order cone linear complementarity problems*, East Asian J. Appl. Math. **10**, 295–315 (2020).

[33] C.F. Ma and N. Huang, *Modified modulus-based matrix splitting algorithms for a class of weakly nondifferentiable nonlinear complementarity problems*, Appl. Numer. Math. **108**, 116–124 (2016).

[34] O.L. Mangasarian, *Equivalence of the complementarity problem to a system of nonlinear equations*, SIAM J. Appl. Math. **31**, 89–92 (1976).

[35] O.L. Mangasarian and M.V. Solodov, *Nonlinear complementarity as unconstrained and constrained minimization*, Math. Program. **62**, 277–297 (1993).

[36] F. Mezzadri and E. Galligani, *Modulus-based matrix splitting methods for horizontal linear complementarity problems*, Numer. Algorithms **83**, 201–219 (2020).

[37] K.G. Murty, *Linear Complementarity, Linear and Nonlinear Programming*, Heldermann Verlag (1988).

[38] M.A. Noor, *Fixed point approach for complementarity problems*, J. Comput. Appl. Math. **133**, 437–448 (1988).

[39] X.F. Peng, M. Wang and W. Li, *A relaxation two-sweep modulus-based matrix splitting iteration method for linear complementarity problems*, East Asian J. Appl. Math. **9**, 102–121 (2019).

[40] L. Qi, D. Sun and G. Zhou, *A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities*, Math. Program. **87**, 1–35 (2000).

[41] Z. Sun and J.P. Zeng, *A monotone semismooth Newton type method for a class of complementarity problems*, J. Comput. Appl. Math. **235**, 1261–1274 (2011).

[42] M. Tao and X. Yuan, *On Glowinski's open question on the alternating direction method of multipliers*, J. Optim. Theory Appl. **179**, 163–196 (2018).

[43] G. Wang and F. Tan, *Modulus-based multisplitting iteration method for a class of weakly nonlinear complementarity problem*, Commun. Appl. Math. Comput. **3**, 419–427 (2021).

[44] Y.J. Wu, G.L. Yan and A.L. Yang, *Modulus-based synchronous multisplitting iteration methods for a restricted class of nonlinear complementarity problems*, Numer. Math. Theory Methods Appl. **12**, 709–726 (2019).

[45] Z. Xia and C. Li, *Modulus-based matrix splitting iteration methods for a class of nonlinear complementarity problem*, Appl. Math. Comput. **271**, 34–42 (2015).

[46] S.L. Xie, H.R. Xu and J.P. Zeng, *Two-step modulus-based matrix splitting iteration method for a class of nonlinear complementarity problems*, Linear Algebra Appl. **494**, 1–10 (2016).

[47] N.H. Xiu and J.Z. Zhang, *Some recent advances in projection-type methods for variational inequalities*, J. Comput. Appl. Math. **152**, 559–585 (2003).

[48] H.R. Xu, R.L. Chen, S.L. Xie and L. Wu, *Modulus-based multisplitting iteration methods for a class of nonlinear complementarity problems*, East Asian J. Appl. Math. **8**, 519–530 (2018).

[49] W.W. Xu, L. Zhu, X.F. Peng, H. Liu and J.F. Yin, *A class of modified modulus-based synchronous multisplitting iteration methods for linear complementarity problems*, Numer. Algorithms **85**, 1–21 (2020).

[50] L. Yong, *Nonlinear complementarity problem and solution methods*, in: *Proceedings of the 2010 International Conference on Artificial Intelligence and Computational Intelligence, Part I*, pp. 461–469, Springer-Verlag (2010).

[51] J.J. Zhang, *MSSOR-based alternating direction method for symmetric positive-definite linear complementarity problems*, Numer. Algorithms **68**, 631–644 (2015).

[52] J.J. Zhang, J.L. Zhang and W.Z. Ye, *An inexact alternating direction method of multipliers for the solution of linear complementarity problems arising from free boundary problems*, Numer. Algorithms **78**, 895–910 (2018).

[53] H. Zheng, *Improved convergence theorems of modulus-based matrix splitting iteration method for nonlinear complementarity problems of H-matrices*, Calcolo **54**, 1481–1490 (2017).

[54] H. Zheng, W. Li and S. Vong, *An iteration method for nonlinear complementarity problems*, J. Comput. Appl. Math. **372**, 112681 (2020).

[55] H. Zheng and L. Liu, *A two-step modulus-based matrix splitting iteration method for solving nonlinear complementarity problems of H_+-matrices*, Comput. Appl. Math. **37**, 5410–5423 (2018).

[56] H. Zheng and L. Liu, *The sign-based methods for solving a class of nonlinear complementarity problems*, J. Optim. Theory Appl. **180**, 480–499 (2019).

[57] H. Zheng and S. Vong, *The modulus-based nonsmooth Newtons method for solving a class of nonlinear complementarity problems of P-matrices*, Calcolo **55**, 37 (2018).

[58] H. Zheng, S. Vong and L. Liu, *The relaxation modulus-based matrix splitting iteration method for solving a class of nonlinear complementarity problems*, Int. J. Comput. Math. **96**, 1648–1667 (2018).

[59] H. Zheng, S. Vong and L. Liu, *A direct preconditioned modulus-based iteration method for solving nonlinear complementarity problems of H-matrices*, Appl. Math. Comput. **353**, 396–405 (2019).

[60] N. Zheng, K. Hayami and J.F. Yin, *Modulus-type inner outer iteration methods for nonnegative constrained least square problems*, SIAM J. Matrix Anal. Appl. **37**, 1250–1278 (2016).