axioms

Article

Strong Convergence of a Two-Step Modified Newton Method for Weighted Complementarity Problems

Xiangjing Liu [1,*,†] and Jianke Zhang [2,†]

[1] School of Sciences, Xi'an Technological University, Xi'an 710021, China
[2] School of Science, Xi'an University of Posts and Telecommunications, Xi'an 710121, China; jiankezh@163.com
* Correspondence: liuxiangjing504@163.com
† These authors contributed equally to this work.

Abstract: This paper focuses on the weighted complementarity problem (WCP), which is widely used in the fields of economics, sciences and engineering. Not least because of its local superlinear convergence rate, smoothing Newton methods have widespread application in solving various optimization problems. A two-step smoothing Newton method with strong convergence is proposed. With a smoothing complementary function, the WCP is reformulated as a smoothing set of equations and solved by the proposed two-step smoothing Newton method. In each iteration, the new method computes the Newton equation twice, but using the same Jacobian, which can avoid consuming a lot of time in the calculation. To ensure the global convergence, a derivative-free line search rule is inserted. At the same time, we develop a different term in the solution of the smoothing Newton equation, which guarantees the local strong convergence. Under appropriate conditions, the algorithm has at least quadratic or even cubic local convergence. Numerical experiments indicate the stability and effectiveness of the new method. Moreover, compared to the general smoothing Newton method, the two-step smoothing Newton method can significantly improve the computational efficiency without increasing the computational cost.

Keywords: weighted complementarity problem; derivative-free line search; two-step smoothing Newton method; superquadratic convergence property

MSC: 65K05; 90C33

Citation: Liu, X.; Zhang, J. Strong Convergence of a Two-Step Modified Newton Method for Weighted Complementarity Problems. *Axioms* **2023**, *12*, 742. https://doi.org/10.3390/axioms12080742

Academic Editors: Weifeng Pan, Hua Ming and Dae-Kyoo Kim

Received: 15 June 2023
Revised: 18 July 2023
Accepted: 26 July 2023
Published: 28 July 2023

1. Introduction

The weighted complementarity problem (WCP for short) is

$$x \geq 0,\ s \geq 0,\ G(x,s,y) = 0,\ xs = w, \tag{1}$$

in which $x,\ s \in R^n$, $y \in R^m$, $w \in R^n_+$ is a known weighted vector, $G(x,s,y) : R^{2n+m} \to R^{n+m}$ is a nonlinear mapping and xs represents the vector obtained by multiplying the components of x with s, respectively.

The concept of WCP was introduced first by Potra [1], is an extension of the complementarity problem (CP) [2,3], and is widely used in engineering, economics and science. As shown in [1], Fisher market equilibrium problems from economics can be transformed into WCPs, and quadratic programming and weighted centering problems can be equivalently converted to monotone WCPs. Not only that, the WCP has the potential to be developed into atmospheric chemistry [4,5] and multibody dynamics [6,7].

When $G(x,s,y) : R^{2n+m} \to R^{n+m}$ is a linear mapping, the WCP (1) can be degenerated into the linear weighted complementarity problem (WLCP) as

$$x \geq 0,\ s \geq 0,\ Mx + Ns + Py = t,\ xs = w, \tag{2}$$

where $M,\ N \in R^{(m+n)\times n}$, $P \in R^{(m+n)\times m}$, $t \in R^{m+n}$. Many scholars have studied the WLCP and have put forward many effective algorithms. Potra [1] proposed two interior-point algorithms and discussed their computational complexity and convergence based on the methods by Mcshane [8] and Mizuno et al. [9]. Gowda [10] discussed a class of WLCP over Euclidean Jordan algebra. Chi et al. [11,12] proposed infeasible interior-point methods for WLCPs, which have good computational complexity. Asadi et al. [13] presented a modified interior-point method and obtained an iteration bound for the monotone WLCP.

On the other hand, recent years have witnessed a growing development of smoothing Newton methods for WCPs whose basic idea is to convert the problem to a smoothing set of nonlinear equations by employing a smoothing function, which is then solved by Newton methods [14–19]. Zhang [20] proposed a smoothing Newton method for the WLCP. For WCPs over Euclidean Jordan algebras, Tang et al. [21] presented a smoothing method and analyzed its convergence property under some weaker assumptions.

The two-step Newton method [22–24], which typically achieves third-order convergence when solving nonlinear equations $H(x) = 0$, has a higher order of convergence than the classical Newton method. The two-step Newton algorithm computes not only a Newton step defined as

$$d_1^k = -H'(x^k)^{-1}H(x^k),$$

but also an approximate Newton step as

$$d_2^k = -H'(x^k)^{-1}H(y^k),$$

where $y^k = x^k + d_1^k$ and $H'(x)$ represents the Jacobian matrix of $H(x)$. Compared with classical third-order methods such as Halley's method [25] or super-Newton's method [26], the two-step Newton algorithm does not need to compute the second-order Hessen matrix, and its computational cost is lower. Without adding additional derivatives and inverse operators, it is possible to raise the order of convergence from second to third order by evaluating the function only once.

In light of those considerations, we present here a two-step Newton algorithm possessing a high-order convergence rate for the WCP (1) on a smoothing complementarity function and an equivalent smoothing system of equations. The new algorithm has the following advantageous properties:

- The proposed method computes the Newton direction twice in each iteration. The first calculation yields a Newton direction, and the second yields an approximate Newton direction. Moreover, both calculations employ the same Jacobian matrix (see Section 3), which saves computing costs.
- The new algorithm utilizes a new term $\zeta_k = \min\{\gamma, \varepsilon_k^{\varrho}\}$ where $\varrho \in [1, 2]$ (see Section 3), when computing the Newton direction, unlike existing Newton algorithms for the WCP [20,21], which determine the local strong convergence. In particular, when $\varrho = 2$, the algorithm has local cubic convergence properties.
- To obtain global convergence properties, we employ a derivative-free line search rule.

This paper is structured as follows. Section 2 presents a smoothing function and discusses its basic properties. Section 3 presents a derivative-free two-step smoothing Newton algorithm for the WCP, which is shown to be feasible. Section 4 deals with convergence properties. Section 5 shows some experiment results. Section 6 gives some concluding remarks.

2. Preliminaries

We define a smoothing function as

$$\theta_\varepsilon(u, v, r) = \sqrt{u^2 + v^2 + 2r + 2\varepsilon} - (u + v), \tag{3}$$

where $\varepsilon \in (0, 1)$ and $r \geq 0$ is a given constant. It readily follows that $\theta_0(u, v, r) = 0$ if and only if $u \geq 0$, $v \geq 0$, $uv = r$.

By simple reasoning and calculations, we can conclude the following.

Lemma 1. *For any* $0 < \varepsilon < 1$, $\theta_\varepsilon(u,v,r)$ *is continuously differentiable, where*

$$(\theta_\varepsilon(u,v,r))'_\varepsilon = \frac{1}{\sqrt{u^2+v^2+2r+2\varepsilon}},$$
$$(\theta_\varepsilon(u,v,r))'_u = \frac{u}{\sqrt{u^2+v^2+2r+2\varepsilon}} - 1,$$
$$(\theta_\varepsilon(u,v,r))'_v = \frac{v}{\sqrt{u^2+v^2+2r+2\varepsilon}} - 1.$$

In addition, $(\theta_\varepsilon(u,v,r))'_u < 0$ *and* $(\theta_\varepsilon(u,v,r))'_v < 0$.

Let $z = (\varepsilon, x, s, y) \in R \times R^{2n+m}$ and $w \in R^n_+$; we define $M(z)$ by

$$M(z) = \begin{pmatrix} \varepsilon \\ \theta_\varepsilon(x,s,w) \\ G(x,s,y) \end{pmatrix}, \tag{4}$$

where

$$\theta_\varepsilon(x,s,w) = \begin{pmatrix} \theta_\varepsilon(x_1,s_1,w_1) \\ \vdots \\ \theta_\varepsilon(x_n,s_n,w_n) \end{pmatrix}. \tag{5}$$

It follows that the WCP (1) can be transformed into an equivalent equation:

$$M(z) = 0. \tag{6}$$

The following lemma states the continuous differentiability of $M(z)$.

Lemma 2. *Define* $M(z)$ *and* $\theta_\varepsilon(x,s,w)$ *by (4) and (5), respectively. For any* $\varepsilon > 0$, $M(z)$ *is continuously differentiable with*

$$M'(z) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ D_1 & D_2 & D_3 & 0 \\ 0 & G'_x & G'_s & G'_y \end{pmatrix}, \tag{7}$$

where

$$D_1 = \text{vec}\left\{ \frac{1}{\sqrt{x_i^2+s_i^2+2w_i+2\varepsilon}} \right\},\ i = 1,2,\ldots,n, \tag{8}$$

$$D_2 = \text{diag}\left\{ \frac{x_i}{\sqrt{x_i^2+s_i^2+2w_i+2\varepsilon}} - 1 \right\},\ i = 1,2,\ldots,n, \tag{9}$$

$$D_3 = \text{diag}\left\{ \frac{s_i}{\sqrt{x_i^2+s_i^2+2w_i+2\varepsilon}} - 1 \right\},\ i = 1,2,\ldots,n. \tag{10}$$

In order to discuss the nonsingularity of Jacobian matrix $M'(z)$, it is necessary to make some assumption.

Assumption 1. *Assuming that* G'_y *is column full rank, then it holds that any* $(\Delta x, \Delta s, \Delta y) \in R^{2n+m}$ *with*

$$G'_x \Delta x + G'_s \Delta s + G'_y \Delta y = 0$$

yields $\langle \Delta x, \Delta s \rangle \geq 0$.

For the WLCP (2), i.e., $G(x,s,y) : R^{2n+m} \to R^{n+m}$ is a linear mapping, then Assumption 1 reduces to

$$M\Delta x + N\Delta s + P\Delta y = 0,$$

which shows that $G(x,s,y)$ is monotone, and this case has been discussed for the feasibility of smoothing algorithms for the WLCP, see [1,20,27] and the reference therein.

Theorem 1. *If Assumption 1 holds, then for any* $\varepsilon > 0$*,* $M'(z)$ *is nonsingular.*

Proof of Theorem 1. It only needs to verify that there exists $\Delta z = (\Delta\varepsilon, \Delta x, \Delta s, \Delta y) \in R^{2n+m+1}$ such that

$$M'(z)\Delta z = 0, \tag{11}$$

with $\Delta z = 0$. Substituting (7) into (11) yields

$$\begin{aligned} \Delta\varepsilon &= 0, \\ G'_x\Delta x + G'_s\Delta s + G'_y\Delta y &= 0, \\ D_1\Delta\varepsilon + D_2\Delta x + D_3\Delta s &= 0. \end{aligned} \tag{12}$$

By Lemmas 1 and 2, we obtain that the diagonal matrices D_2 and D_3 are both negative definite. Upon (12), we get

$$\Delta x = -D_2^{-1}D_3\Delta s, \tag{13}$$

and then

$$\langle \Delta x, \Delta s \rangle = -\Delta s^T D_3 D_2^{-1}\Delta s \leq 0. \tag{14}$$

Using Assumption 1 yields that $\langle \Delta x, \Delta s \rangle \geq 0$, which, together with (14), implies

$$\langle \Delta x, \Delta s \rangle = -\Delta s^T D_3 D_2^{-1}\Delta s = 0,$$

and then $\Delta s = 0$. We conclude from (13) that $\Delta x = 0$; hence, $\Delta y = 0$ due to the second equation in (12). We complete the proof. □

3. A Two-Step Newton Method

Now, we state the two-step smoothing Newton method. In order to understand Algorithm 1 more intuitively, we also give the flow chart of the new algorithm, as shown in Figure 1.

Algorithm 1 A Two-Step Newton Method.

Initial Step. Choose $\varepsilon_0 > 0$ and γ, $\eta \in (0,1)$. Choose $c \in (0,1)$, $l \in (0,1)$ and $\varrho \in [1,2]$. $\{\xi_k\} \subseteq R_+$ satisfies that $\sum_{k=0}^{\infty}\xi_k \leq \xi < \infty$. Choose any $(x^0, s^0, y^0) \in R^{2n+m}$ as a starting point and let $\mu_0 = (\varepsilon_0, 0, 0, 0)^T \in R \times R^{2n+m}$. Set $z^0 = (\varepsilon_0, x^0, s^0, y^0)$ and $k = 0$.

Step 1. If $\|M(z^k)\| = 0$, stop. Else, calculate Δz_1^k by

$$M'(z^k)\Delta z_1^k = -M(z^k) + \zeta_k\mu_k, \tag{15}$$

where $\zeta_k = \min\{\gamma, \varepsilon_k^{\varrho}\}$ and $\mu_k = (\varepsilon_k, 0, 0, 0)^T$. Let $\bar{z}^k = z^k + \Delta z_1^k$.

Step 2. Calculate Δz_2^k by

$$M'(z^k)\Delta z_2^k = -M(\bar{z}^k) + \zeta_k\mu_k. \tag{16}$$

Step 3. If

$$\|M(z^k + \Delta z_1^k + \Delta z_2^k)\| \leq c \cdot \|M(z^k)\|, \tag{17}$$

set $\beta_k = 1$ and go to Step 5.

Step 4. Set β_k be the maximum of $1, l, l^2, \ldots$ that satisfies the following inequality

$$\|M(z^k + \beta_k\Delta z_1^k + \beta_k^2\Delta z_2^k)\|^2 \leq (1+\xi_k)\|M(z^k)\|^2 - \eta\beta_k^2(\|\Delta z_1^k\|^2 + \|\Delta z_2^k\|^2 + \|M(z^k)\|^2). \tag{18}$$

Step 5. Set $z^{k+1} = z^k + \beta_k\Delta z_1^k + \beta_k^2\Delta z_2^k$, $k = k+1$ and return to Step 1.

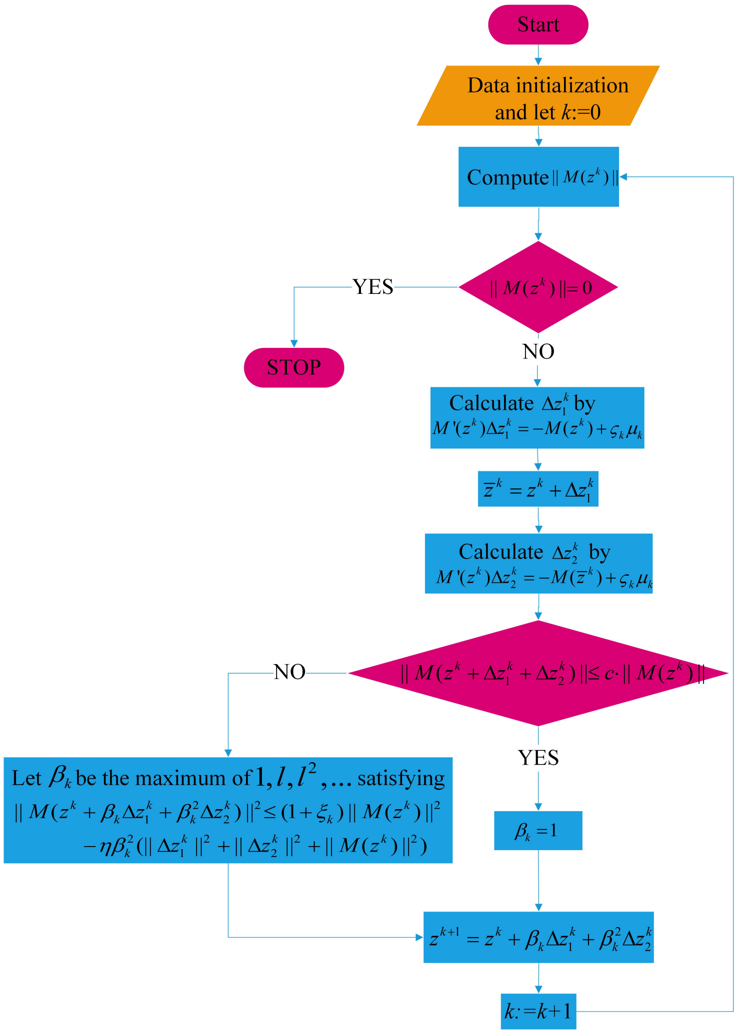

Figure 1. Flow chart of Algorithm 1.

Remark 1.

1. *In each iteration, Algorithm 1 computes the Newton direction by the equations*

$$M'(z^k)\Delta z_1^k = -M(z^k) + \zeta_k \mu_k,$$

and

$$M'(z^k)\Delta z_2^k = -M(\bar{z}^k) + \zeta_k \mu_k,$$

using a new term $\zeta_k = \min\{\gamma, \varepsilon_k^{\varrho}\}$, *which is of significance for discussing the local strong convergence of Algorithm 1. Moreover, although Algorithm 1 computes the Newton direction twice, its computational cost is comparable to the classical Newton method.*

2. *Algorithm 1 employs a derivative-free line search rule, a variant of that in [28]. As is shown in Theorem 2, the new derivative-free line search is feasible.*

Theorem 2. *If Assumption 1 holds, then Algorithm 1 is feasible. Moreover, we have*

1. $\varepsilon_k \geq 0$ *for any* $k \geq 0$.
2. $\{\varepsilon_k\}$ *is non-increasing monotonically.*

Proof of Theorem 2. With Theorem 1, we get that $M'(z)$ is invertible. Then, both Step 1 and Step 2 are feasible. Next, we show the feasibility of Step 4. Supposing not, then for any $\beta_k \geq 0$,

$$\begin{aligned}
&\|M(z^k + \beta_k \Delta z_1^k + \beta_k^2 \Delta z_2^k)\|^2 \\
&> (1+\xi_k)\|M(z^k)\|^2 - \eta\beta_k^2(\|\Delta z_1^k\|^2 + \|\Delta z_2^k\|^2 + \|M(z^k)\|^2) \\
&\geq \|M(z^k)\|^2 - \eta\beta_k^2(\|\Delta z_1^k\|^2 + \|\Delta z_2^k\|^2 + \|M(z^k)\|^2).
\end{aligned} \tag{19}$$

Hence,

$$\|M(z^k + \beta_k \Delta z_1^k + \beta_k^2 \Delta z_2^k)\|^2 - \|M(z^k)\|^2 > \eta\beta_k^2(\|\Delta z_1^k\|^2 + \|\Delta z_2^k\|^2 + \|M(z^k)\|^2) \tag{20}$$

Dividing (20) by β_k and taking the limit as $k \to \infty$, we can conclude that

$$\lim_{k\to\infty} \frac{\|M(z^k + \beta_k \Delta z_1^k + \beta_k^2 \Delta z_2^k)\|^2 - \|M(z^k)\|^2}{\beta_k} \geq 0.$$

Therefore, we get

$$\begin{aligned}
&\lim_{k\to\infty} \frac{\|M(z^k + \beta_k \Delta z_1^k + \beta_k^2 \Delta z_2^k)\|^2 - \|M(z^k)\|^2}{\beta_k} \\
&= 2M(z^k)^T(M'(z^k)\Delta z_1^k) \\
&\geq 0.
\end{aligned} \tag{21}$$

On the other hand, if z^k is not the solution of (1), then it follows from Step 1 that

$$\begin{aligned}
&M(z^k)^T(M'(z^k)\Delta z_1^k) \\
&= M(z^k)^T(-M(z^k) + \zeta_k \mu_k) \\
&\leq (\gamma - 1)]\|M(z^k)\|^2 \\
&< 0,
\end{aligned} \tag{22}$$

where the first equality comes from the fact that $\zeta_k = \min\{\gamma, \varepsilon_k^{\varrho}\} \leq \gamma \leq 1$. This contradicts (21). Hence, Step 4 is feasible, and then Algorithm 1 is well-defined.

Then, we show $\varepsilon_k \geq 0$ by induction. Suppose that $\varepsilon_k \geq 0$ for some $k > 0$, we obtain from (15) and (16) that

$$\Delta\varepsilon_k^1 = -\varepsilon_k + \zeta_k \varepsilon_k, \tag{23}$$

and

$$\Delta\varepsilon_k^2 = -\varepsilon_k - \Delta\varepsilon_k^1 + \zeta_k \varepsilon_k. \tag{24}$$

Then, it holds by Step 5 that

$$\begin{aligned}
\varepsilon_{k+1} &= \varepsilon_k + \beta_k \Delta\varepsilon_k^1 + \beta_k^2 \Delta\varepsilon_k^2 \\
&= \varepsilon_k + \beta_k(-\varepsilon_k + \zeta_k \varepsilon_k) + \beta_k^2(-\varepsilon_k - \Delta\varepsilon_k^1 + \zeta_k \varepsilon_k) \\
&= (1 - \beta_k)\varepsilon_k + \beta_k \zeta_k \varepsilon_k \\
&= [1 - \beta_k(1 - \zeta_k)]\varepsilon_k,
\end{aligned} \tag{25}$$

which means that $\varepsilon_{k+1} \geq 0$ due to the fact that $\beta_k \leq 1$ and $\zeta_k \leq 1$. Moreover, it follows that

$$\varepsilon_{k+1} = [1 - \beta_k(1 - \zeta_k)]\varepsilon_k \leq \varepsilon_k,$$

i.e., $\{\varepsilon_k\}$ is non-increasing monotonically. $\square$

Lemma 3. *If Assumption 1 holds, then $\{\|M(z^k)\|\}$ is convergent, and the sequence $\{z^k\}$ remains in the level set $L(z)$ of $\|M(z)\|$*

$$L(z) = \{z \in R_+ \times R^{2n+m} | \|M(z)\| \leq e^{\frac{\xi}{2}} \|M(z^0)\|\}.$$

Proof of Lemma 3. According to (18), we have

$$\|M(z^{k+1})\|^2 \leq (1+\xi_k)\|M(z^k)\|^2. \tag{26}$$

Since $\sum_{k=0}^{\infty} \xi_k \leq \xi < \infty$, it follows from Lemma 3.3 in [29] that $\{\|M(z^k)\|^2\}$ is convergent. Then, $\{\|M(z^k)\|\}$ is also convergent.

Moreover, we have

$$\begin{aligned}
\|M(z^{k+1})\| &\leq \sqrt{1+\xi_k}\|M(z^k)\| \\
&\leq \sqrt{1+\xi_k} \cdot \sqrt{1+\xi_{k-1}} \cdot \dots \cdot \sqrt{1+\xi_0}\|M(z^0)\| \\
&= \prod_{j=0}^{k} \sqrt{1+\xi_j}\|M(z^0)\| \\
&\leq \left[\sum_{j=0}^{k} \frac{1}{k+1}(1+\xi_j)\right]^{\frac{k+1}{2}} \|M(z^0)\| \\
&\leq \left(1+\frac{\xi}{k+1}\right)^{\frac{k+1}{2}} \|M(z^0)\| \\
&\leq e^{\frac{\xi}{2}}\|M(z^0)\|.
\end{aligned} \tag{27}$$

□

4. Convergence Properties

4.1. Global Convergence

We first show a statement of great significance before analyzing the convergence properties of Algorithm 1.

Theorem 3. *If Assumption 1 holds, then it holds that $\lim_{k\to\infty} \beta_k \|M(z^k)\| = 0$.*

Proof of Theorem 3. Define $S(k)$ and $R(k)$ by

$$S(k) = \{j \leq k | (17) \text{ is satisfied}\}$$

and

$$R(k) = \{0, 1, \dots, k\} \backslash S(k).$$

Let $|S(k)|$ be the number of elements in $S(k)$.

If (17) holds for infinite k, then $|S(k)| \to \infty$ as $k \to \infty$. By (17), (18) and (27), we get

$$\begin{aligned}
\|M(z^{k+1})\|^2 &\leq \prod_{i\in R(k)} (1+\xi_i) \prod_{i \in S(k)} l^2 \|M(z^0)\|^2 \\
&= \prod_{i\in R(k)} (1+\xi_i) l^{2|S(k)|}\|M(z^0)\|^2 \\
&\leq e^{\xi} l^{2|S(k)|}\|M(z^0)\|^2.
\end{aligned}$$

As $k \to \infty$, $\|M(z^{k+1})\|^2 \to 0$, i.e., $\|M(z^{k+1})\| \to 0$, and then $\lim_{k\to\infty} \beta_k\|M(z^k)\| = 0$.

Assume that (17) holds for finite k. Then, we have from (18) that

$$\sum_{k=0}^{\infty} \beta_k^2 \|M(z^k)\| = 0,$$

which implies

$$\lim_{k\to\infty} \beta_k \|M(z^k)\| = 0.$$

The proof is completed. □

Theorem 4. *If Assumption 1 holds, then $\{z^k\}$ converges to a solution of the WCP (1).*

Proof of Theorem 4. According to Lemma 3, we know that $\{\|M(z^k)\|\}$ is convergent. Suppose, without loss of generality, that $\{z^k = (\varepsilon_k, x^k, s^k, y^k)\}$ converges to $z^* = (\varepsilon_*, x^*, s^*, y^*)$ and $\lim_{k\to\infty} \|M(z^k)\| = \|M(z^*)\| \geq 0$. Next, we show $\|M(z^*)\| = 0$ by contradiction. Assume that $\|M(z^*)\| > 0$, then $\lim_{k\to\infty} \beta_k = 0$ due to Theorem 3.

Let $\hat{\beta} = \frac{\beta_k}{l}$, it follows from Step 4 that

$$\begin{aligned} \|M(z^k + \hat{\beta}\Delta z_1^k + \hat{\beta}^2 \Delta z_2^k)\|^2 >& (1+\xi_k)\|M(z^k)\|^2 \\ &- \eta\hat{\beta}^2(\|\Delta z_1^k\|^2 + \|\Delta z_2^k\|^2 + \|M(z^k)\|^2), \end{aligned} \tag{28}$$

for sufficiently large k.

On the other hand, since

$$M(z^k + \hat{\beta}(\Delta z_1^k + \hat{\beta}\Delta z_2^k)) = M(z^k) + \hat{\beta}M'(z^k)(\Delta z_1^k + \hat{\beta}\Delta z_2^k) + o(\hat{\beta}),$$

it follows that

$$\begin{aligned} &\|M(z^k + \hat{\beta}(\Delta z_1^k + \hat{\beta}\Delta z_2^k))\|^2 \\ =& \|M(z^k) + \hat{\beta}M'(z^k)(\Delta z_1^k + \hat{\beta}\Delta z_2^k)\|^2 + o(\hat{\beta}) \\ =& \|M(z^k)\|^2 + 2\hat{\beta}M(z^k)^T\Big(M'(z^k)(\Delta z_1^k + \hat{\beta}\Delta z_2^k)\Big) + o(\hat{\beta}). \end{aligned} \tag{29}$$

Combining (15) and (16) with (29), we obtain

$$\begin{aligned} \|M(z^k + \hat{\beta}(\Delta z_1^k + \hat{\beta}\Delta z_2^k))\|^2 &= \|M(z^k)\|^2 + 2\hat{\beta}M(z^k)^T\Big(-M(z^k) + \zeta_k\mu_k\Big) + o(\hat{\beta}) \\ &= (1-2\hat{\beta})\|M(z^k)\|^2 + 2\hat{\beta}\zeta_k M(z^k)^T\mu_k + o(\hat{\beta}) \\ &\leq [1 - 2\hat{\beta}(1-\gamma)]\|M(z^k)\|^2 + o(\hat{\beta}). \end{aligned} \tag{30}$$

Then, from (28) and (30), we get

$$\begin{aligned} &[1 - 2\hat{\beta}(1-\gamma)]\|M(z^k)\|^2 + o(\hat{\beta}) \\ >& (1+\xi_k)\|M(z^k)\|^2 - \eta\hat{\beta}^2(\|\Delta z_1^k\|^2 + \|\Delta z_2^k\|^2 + \|M(z^k)\|^2) \\ \geq& \|M(z^k)\|^2 - \eta\hat{\beta}^2(\|\Delta z_1^k\|^2 + \|\Delta z_2^k\|^2 + \|M(z^k)\|^2). \end{aligned}$$

It follows by simple calculation that

$$2(1-\gamma)\|M(z^k)\|^2 + \frac{o(\hat{\beta})}{\hat{\beta}} < \eta\hat{\beta}(\|\Delta z_1^k\|^2 + \|\Delta z_2^k\|^2 + \|M(z^k)\|^2). \tag{31}$$

Passing to the limit in (31), then

$$2(\gamma - 1)\|M(z^*)\|^2 \geq 0.$$

As $\|M(z^*)\| > 0$, it follows that

$$\gamma > 1,$$

a contradiction. Thus, $\|M(z^*)\| = 0$, which means that $\{z^k\}$ converges to a solution of the WCP (1). □

4.2. Local Convergence

We then discuss the local superquadratical convergence properties of Algorithm 1.

Theorem 5. *If Assumption 1 holds, all $J \in \partial M(z^*)$ are nonsingular. Suppose that $G'(x, s, y)$ and $M'(x, s, y)$ are both Lipschitz continuous on some neighborhood of z^*, then*

1. *$\beta_k \equiv 1$ for any sufficiently large k.*
2. *$\{z^k\}$ converges to z^* locally superquadratically. In particular, $\{z^k\}$ converges to z^* locally cubically if $\varrho = 2$.*

Proof of Theorem 5. Upon Theorem 4, we have that $\|M(z^*)\| = 0$. All $J \in \partial M(z^*)$ are nonsingular, so we have for any sufficiently large k that

$$\|M'(z^k)^{-1}\| = O(1). \tag{32}$$

Since $G'(x, s, y)$ is Lipschitz continuous on some neighborhood of z^*, $M(z)$ is strongly semismooth and Lipschitz continuous on some neighborhood of z^*, namely,

$$\|M(z^k) - M(z^*) - M'(z^k)(z^k - z^*)\| = O(\|z^k - z^*\|^2), \tag{33}$$

and

$$\|M(z^k)\| = \|M(z^k) - M(z^*)\| = O(\|z^k - z^*\|), \tag{34}$$

for any sufficiently large k.

By the definition of ζ_k and μ_k, it follows that

$$\|\zeta_k \mu_k\| \leq \varepsilon_k^{\varrho+1} \leq \|M(z^k)\|^{\varrho+1}. \tag{35}$$

Then, combining (15) and (32)–(35) implies

$$\begin{aligned}
\|z^k + \Delta z_1^k - z^*\| &= \|z^k + M'(z^k)^{-1}(-M(z^k) + \zeta_k \mu_k) - z^*\| \\
&= O\Big(\|M(z^k) - M(z^*) - M'(x^k)(z^k - z^*) + \zeta_k \mu_k\|\Big) \\
&\leq O(\|z^k - z^*\|^2) + O(\|M(z^k)\|^3) \\
&= O(\|z^k - z^*\|^2),
\end{aligned} \tag{36}$$

which means that $z^k + \Delta z_1^k$ is sufficiently close to z^* for sufficiently large k. Then, according to (34) and (36), we have that

$$\begin{aligned}
\|M(z^k + \Delta z_1^k)\| &= \|M(z^k + \Delta z_1^k) - M(z^*)\| \\
&= O(\|z^k + \Delta z_1^k - z^*\|) \\
&= O(\|z^k - z^*\|^2) = O(\|M(z^k)\|^2)
\end{aligned} \tag{37}$$

Hence, since $\varrho \geq 1$, it follows from (16), (32), (35) and (37) that

$$\begin{aligned} \|\Delta z_2^k\| &= \left\| M'(z^k)^{-1}\left(-M(z^k+\Delta z_1^k)+\zeta_k\mu_k\right)\right\| \\ &\leq O\left(\|M(z^k+\Delta z_1^k)\|+\|\zeta_k\mu_k\|\right) \\ &= O(\|M(z^k)\|^2)+O(\|M(z^k)\|^{\varrho+1}) \\ &= O(\|M(z^k)\|^2), \end{aligned} \tag{38}$$

combining with (34) and (36) yields

$$\begin{aligned} \|z^k+\Delta z_1^k+\Delta z_2^k-\Delta z^*\| &\leq \|z^k+\Delta z_1^k-z^*\|+\|\Delta z_2^k\| \\ &= O(\|z^k-z^*\|^2), \end{aligned} \tag{39}$$

for any sufficiently large k, which also means that $z^k+\Delta z_1^k+\Delta z_2^k$ is sufficiently close to z^* for a sufficiently large k, which, together with the Lipschitz continuity of $M(z)$ on some neighborhood of z^*, implies

$$\|M(z^k+\Delta z_1^k+\Delta z_2^k)-M(z^k)-M'(z^k)(\Delta z_1^k+\Delta z_2^k)\| = O(\|\Delta z_1^k+\Delta z_2^k\|^2). \tag{40}$$

Then, it holds that

$$\begin{aligned} &\|M(z^k+\Delta z_1^k+\Delta z_2^k)\| \\ \leq\ & \|M(z^k)+M'(z^k)(\Delta z_1^k+\Delta z_2^k)\| \\ &+\|M(z^k+\Delta z_1^k+\Delta z_2^k)-M(z^k)-M'(z^k)(\Delta z_1^k+\Delta z_2^k)\| \\ =\ & \|M(z^k)+M'(z^k)\Delta z_1^k\|+\|M(\bar{z}^k)\| \\ &+\|M(\bar{z}^k)+M'(z^k)\Delta z_2^k\|+O(\|\Delta z_1^k+\Delta z_2^k\|^2). \end{aligned} \tag{41}$$

Now, we consider the term

$$\|M(z^k)+M'(z^k)\Delta z_1^k\|+\|M(\bar{z}^k)\|+\|M(\bar{z}^k)+M'(z^k)\Delta z_2^k\|+O(\|\Delta z_1^k+\Delta z_2^k\|^2).$$

By (15), (32) and (34), we obtain

$$\begin{aligned} \|\Delta z_1^k\| &= \left\| M'(z^k)^{-1}\left(-M(z^k)+\zeta_k\mu_k\right)\right\| \\ &= O(\|M(z^k)\|), \\ &= O(\|z^k-z^*\|). \end{aligned} \tag{42}$$

On the other hand, according to (15), (16) and (35), we obtain

$$\|M(z^k)+M'(z^k)\Delta z_1^k\| = \|\zeta_k\mu_k\| = O(\|M(z^k)\|^{\varrho+1}), \tag{43}$$

and

$$\|M(\bar{z}^k)+M'(z^k)\Delta z_2^k\| = \|\zeta_k\mu_k\| = O(\|M(z^k)\|^{\varrho+1}). \tag{44}$$

So, combining (34), (38) and (41)–(44), we have

$$\begin{aligned} &\|M(z^k+\Delta z_1^k+\Delta z_2^k)\| \\ =\ & O(\|M(z^k)\|^{\varrho+1})+O(\|z^k-z^*\|^2) \\ =\ & o(\|M(z^k)\|) \\ =\ & \rho_k\|M(z^k)\|, \end{aligned} \tag{45}$$

where $\rho_k \to 0$. This means that (17) makes sense for a sufficiently large k, which shows that $\beta_k \equiv 1$ when z^k is sufficiently close to z^*, i.e.,

$$z^{k+1} = z^k + \Delta z_1^k + \Delta z_2^k. \tag{46}$$

By using (16), (32), (35), (46) and the Lipschitz continuity of $M'(z)$ on some neighborhood of z^*, we get

$$\begin{aligned}
&\|z^{k+1} - z^*\| \\
&= \|\bar{z}^k - z^* + M'(z^k)^{-1}(-M(\bar{z}^k) + \zeta_k \mu_k)\| \\
&= O\Big(\|M(\bar{z}^k) - M(z^*) - M'(z^k)(\bar{z}^k - z^*)\| + \|\zeta_k \mu_k\|\Big) \\
&= O\Big(\|M(\bar{z}^k) - M(z^*) - M'(\bar{z}^k)(\bar{z}^k - z^*) + (M'(\bar{z}^k) - M'(z^k))(\bar{z}^k - z^*)\| + \|\zeta_k \mu_k\|\Big) \\
&= O(\|\bar{z}^k - z^*\|^2) + O(\|M(z^k)\|^{\varrho+1}) \\
&= O(\|z^k - z^*\|^{\varrho+1}).
\end{aligned} \tag{47}$$

Moreover, we have from (34) that

$$\|M(z^{k+1})\| = O(\|z^{k+1} - z^*\|) = O(\|z^k - z^*\|^{\varrho+1}) = O(\|M(z^k)\|^{\varrho+1}),$$

which means that $\{z^k\}$ converges to z^* locally and superquadratically since $\varrho \in [1,2]$. In particular, if $\varrho = 2$, then

$$\|z^{k+1} - z^*\| = O(\|z^k - z^*\|^3)$$

and

$$\|M(z^{k+1})\| = O(\|M(z^k)\|^3),$$

which means that $\{z^k\}$ converges to z^* locally cubically. □

5. Numerical Experiments

We implement Algorithm 1 in practice and use it to solve some numerical examples to verify the feasibility and effectiveness in this section. All programs are implemented on Matlab R2018b and a PC with 2.30 GHz CPU and 16.00 GB RAM. We also code the algorithm in [20], denoted as SNM_Z, and compare it with the new algorithm. To illustrate the performance of the new algorithm, we also code and compare the algorithm in [20], denoted as SNM_Z, with Algorithm 1. The stopping criterion is $\|M(z^k)\| \le 10^{-6}$ and the parameters are set as

$$l = 0.5,\ c = 0.8,\ \gamma = 0.01,\ \varepsilon_0 = 0.1,\ \eta = 0.001 \text{ and } \xi_k = 1/2^{k+2}.$$

For SNM_Z, the stopping criterion is the same as that in Algorithm 1, and the parameters are the same as [20].

Example 1. *Consider an example of the WLCP (2) with*

$$M = \begin{pmatrix} A \\ B \end{pmatrix}, N = \begin{pmatrix} 0 \\ -I \end{pmatrix}, P = \begin{pmatrix} 0 \\ -A^T \end{pmatrix}, t = \begin{pmatrix} Af \\ g \end{pmatrix},$$

where $B \in R^{n\times n}$, $A \in R^{m\times n}$ whose elements are produced by the normal distribution on $[0,1]$, $f, g \in R^n$ are chosen uniformly from $[0,1]$ and $[-1,0]$, respectively. The weighted vector $w \in R^n$ is generated by $w = uv$ with $v = Bu - g$, where $u, v \in R^n$ are generated uniformly from $[0,1]$.

We test two kinds of problems using Algorithm 1 with different B, denoted by B_1 and B_2. B_1 is produced by setting $B_1 = QQ^T/\|QQ^T\|$, where Q is generated uniformly from $[0,1]$. The diagonal matrix B_2 is generated randomly on $[0,1]$. The initial points x^0, s^0 and y^0 are chosen as $(1,0,\dots,0)^T$ with relevant dimensions in every experiment.

First, in order to state the influence of ϱ on the local convergence, we perform different ϱ for each case on B_1. We perform three experiments for each problem and present the numerical results in Table 1. In what follows, (AIT)IT represents the (average) number of iterations, (ATime)Time is the (average) time taken for the algorithm to run in seconds, and (AERO)ERO represents the (average) value of $\|M(z^k)\|$ in the last iteration. As we can see in Table 1, Algorithm 1 has different local convergence rates with different values of ϱ. Moreover, Algorithm 1 has at least a local quadratic rate of convergence.

Then, we test an example of $m = 400$ and $n = 800$ for B_1 to visually demonstrate the local convergence properties of Algorithm 1 and SNM_Z. In what follows, we set $\varrho = 2$ in Algorithm 1. The results are shown in Table 2, which shows that Algorithm 1 has a local cubic convergence rate whose convergence rate is actually faster than SNM_Z, which possesses the local quadratic convergence rate.

Finally, we randomly performed 10 trials for each case. The tested results are shown in Table 3, which demonstrates that Algorithm 1 carries out fewer iterations than SNM_Z. In addition, although Algorithm 1 calculates the Newton direction twice in each iteration, it does not consume too much time compared with SNM_Z.

Table 1. Numerical results for a WLCP with B_1.

m	n	$\varrho = 1$			$\varrho = 1.5$			$\varrho = 2$		
		IT	Time	ERO	IT	Time	ERO	IT	Time	ERO
500	1000	5	2.0413	1.7219×10^{-10}	4	1.9238	5.4462×10^{-7}	4	1.8250	5.8769×10^{-12}
		6	2.3358	3.9422×10^{-7}	5	2.4771	3.1306×10^{-13}	4	1.7943	3.3544×10^{-12}
		6	2.3764	1.3445×10^{-11}	5	2.4997	3.1950×10^{-13}	4	1.7302	3.5464×10^{-12}
1000	2000	6	14.0543	1.7549×10^{-12}	5	11.9847	5.9945×10^{-7}	4	12.4097	1.8342×10^{-12}
		6	14.5208	2.6955×10^{-12}	5	11.8457	3.0670×10^{-9}	4	12.3936	1.1576×10^{-8}
		6	14.2352	1.3315×10^{-9}	5	12.5786	2.7924×10^{-10}	4	12.4111	1.8763×10^{-12}
1500	3000	5	39.7915	5.0310×10^{-7}	5	38.5589	3.0246×10^{-12}	4	36.4203	6.2360×10^{-12}
		7	48.1246	2.9957×10^{-7}	6	41.4926	3.2203×10^{-12}	4	39.9579	6.7978×10^{-11}
		7	46.4935	1.3715×10^{-9}	6	41.8099	3.4429×10^{-10}	4	46.6347	2.0403×10^{-11}
2000	4000	6	110.1033	5.6894×10^{-10}	5	81.3782	5.9703×10^{-7}	4	100.4127	2.7987×10^{-11}
		6	107.8945	2.2694×10^{-8}	6	104.6634	3.3573×10^{-7}	4	110.2353	1.2117×10^{-11}
		7	128.1354	2.6992×10^{-11}	7	136.8423	5.7307×10^{-12}	4	103.2185	2.2580×10^{-10}
2500	5000	6	178.1547	3.4099×10^{-10}	5	176.0884	6.7770×10^{-7}	4	100.4127	2.7987×10^{-11}
		6	261.9206	2.5631×10^{-10}	6	178.2723	1.1975×10^{-10}	4	110.2353	1.2117×10^{-11}
		6	286.3750	6.7117×10^{-12}	6	209.1771	6.7177×10^{-12}	4	130.8529	8.7520×10^{-12}
3000	6000	6	371.0119	4.0263×10^{-10}	6	356.2151	1.8624×10^{-8}	4	316.1594	4.5234×10^{-11}
		6	332.6154	2.4767×10^{-8}	6	306.6382	9.6163×10^{-12}	4	320.2794	6.5542×10^{-11}
		6	352.4338	1.7184×10^{-10}	6	369.1693	8.3990×10^{-8}	4	320.2625	1.8997×10^{-11}

Table 2. Variation of the value of $\|M(z^k)\|$ with the number of iterations for B_1.

k	Algorithm 1	SNM_Z
1	3.5025×10^{0}	9.0554×10^{0}
2	1.4006×10^{-1}	1.5244×10^{0}
3	1.7885×10^{-4}	1.3534×10^{-1}
4	3.0450×10^{-12}	4.6924×10^{-3}
5	\	1.2604×10^{-5}
6	\	1.5665×10^{-10}

Table 3. Numerical comparison results for a WLCP.

B	m	n	Algorithm 1			SNM_Z		
			AIT	ATime	AERO	AIT	ATime	AERO
B_1	500	1000	4.0	1.6388	2.2736×10^{-7}	5.9	2.5001	3.8463×10^{-8}
	1000	2000	4.0	15.8679	3.5679×10^{-11}	5.6	17.8929	7.1401×10^{-8}
	1500	3000	4.1	42.0557	3.5400×10^{-7}	5.9	31.5197	3.7355×10^{-8}
	2000	4000	4.1	90.4867	5.3932×10^{-7}	5.5	80.4471	5.7230×10^{-8}
	2500	5000	4.1	179.8909	4.8241×10^{-7}	6.0	142.9854	7.1523×10^{-8}
	3000	6000	4.1	300.2717	5.7601×10^{-7}	6.8	215.0043	2.5449×10^{-8}
B_2	500	1000	4.2	1.5968	7.5550×10^{-8}	6.8	1.5040	1.0939×10^{-7}
	1000	2000	4.2	9.5536	2.7452×10^{-7}	7.0	9.4423	9.6079×10^{-8}
	1500	3000	4.4	32.8206	2.2022×10^{-7}	5.8	35.8170	3.9237×10^{-8}
	2000	4000	4.5	80.3553	1.9778×10^{-9}	7.1	73.1503	1.2470×10^{-7}
	2500	5000	4.6	159.6309	7.6437×10^{-9}	7.5	127.3718	1.2020×10^{-7}
	3000	6000	4.8	270.6326	5.9255×10^{-8}	7.3	204.2898	1.1762×10^{-7}

Example 2. *Consider an example of the WCP (1), where*

$$G(x, s, y) = \begin{pmatrix} Bx + C^T y - s + d \\ C(x - t) \end{pmatrix},$$

with $B = \mathrm{diag}(b)$ *where* $b \in R^n$ *is generated uniformly from* $[0, 1]$, $C \in R^{m \times n}$ *whose entries are produced from the standard normal distribution randomly.* $d, t \in R^n$ *and* $w \in R^n$ *are all generated randomly from* $[0, 1]$.

We also generated 10 trials for each case. The initial points x^0, s^0 and y^0 are all chosen as $(1, 0, \ldots, 0)^T$ with relevant dimensions. The test results are shown in Table 4, which also indicates that Algorithm 1 is more stable and efficient than SNM_Z.

Table 4. Numerical comparison results for a WCP.

m	n	Algorithm 1			SNM_Z		
		AIT	ATime	AERO	AIT	ATime	AERO
500	1000	4.4	1.8729	1.7379×10^{-7}	6.7	2.4799	2.4252×10^{-8}
1000	2000	4.6	13.3430	7.0375×10^{-9}	6.9	17.1639	1.1370×10^{-7}
1500	3000	5.0	44.8153	1.2673×10^{-7}	7.4	61.5095	4.7525×10^{-8}
2000	4000	5.2	120.9742	4.0044×10^{-8}	6.8	114.9068	1.7140×10^{-8}
2500	5000	5.3	220.2452	3.6690×10^{-8}	7.3	217.9106	2.0620×10^{-8}
3000	6000	5.2	371.6272	2.2190×10^{-9}	6.6	404.5870	1.1662×10^{-7}

6. Conclusions

The two-step Newton method, known for its efficiency in solving nonlinear equations, is adopted to solve the WCP in this paper. A novel two-step Newton method designed specifically for solving the WCP is proposed. The best property of this method is its consistent Jacobian matrix in each iteration, resulting in an improved convergence rate without additional computational expenses. To guarantee the global convergence, a new derivative-free line search rule is introduced. With appropriate conditions and parameter selection, the algorithm achieves cubic local convergence. Numerical results show that the two-step Newton method significantly improves the computational efficiency without increasing the computational cost compared to the general smoothing Newton method.

Author Contributions: Conceptualization, X.L.; methodology, X.L.; software, J.Z.; validation, X.L. and J.Z.; formal analysis, J.Z.; writing—original draft, X.L.; writing—review and editing, J.Z.; supervision, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data sets used in this paper are available from the corresponding authors upon reasonable request.

Conflicts of Interest: These authors declare no conflict of interest.

References

1. Potra, F. Weighted complementarity problems-a new paradigm for computing equilibria. *SIAM J. Optim.* **2012**, *22*, 1634–1654. [CrossRef]
2. Facchinei, F.; Pang, J. *Finite-Dimensional Variational Inequalities and Complementarity Problems*; Springer: New York, NY, USA, 2003.
3. Che, H.; Wang, Y.; Li, M. A smoothing inexact Newton method for P_0 nonlinear complementarity problem. *Front. Math. China* **2012**, *7*, 1043–1058. [CrossRef]
4. Amundson, N.; Caboussat, A.; He, J.; Seinfeld, J. Primal-dual interior-point method for an optimization problem related to the modeling of atmospheric organic aerosols. *J. Optim. Theory Appl.* **2006**, *130*, 375–407.
5. Caboussat, A.; Leonard, A. Numerical method for a dynamic optimization problem arising in the modeling of a population of aerosol particles. *C. R. Math.* **2008**, *346*, 677–680. [CrossRef]
6. Flores, P.; Leine, R.; Glocker, C. Modeling and analysis of planar rigid multibody systems with translational clearance joints based on the non-smooth dynamics approach. *Multibody Syst. Dyn.* **2010**, *23*, 165–190. [CrossRef]
7. Pfeiffer, F.; Foerg, M.; Ulbrich, H. Numerical aspects of non-smooth multibody dynamics. *Comput. Method Appl. Mech.* **2006**, *195*, 6891–6908. [CrossRef]
8. McShane, K. Superlinearly convergent $O(\sqrt{n}L)$-iteration interior-point algorithms for linear programming and the monotone linear complementarity problem. *SIAM J. Optim.* **1994**, *4*, 247–261. [CrossRef]
9. Mizuno, S.; Todd, M.; Ye, Y. On adaptive-step primal-dual interior-point algorithms for linear programming. *Math. Oper. Res.* **1993**, *18*, 964–981. [CrossRef]
10. Gowda, M.S. Weighted LCPs and interior point systems for copositive linear transformations on Euclidean Jordan algebras. *J. Glob. Optim.* **2019**, *74*, 285–295. [CrossRef]
11. Chi, X.; Wang, G. A full-Newton step infeasible interior-point method for the special weighted linear complementarity problem. *J. Optim. Theory Appl.* **2021**, *190*, 108–129. [CrossRef]
12. Chi, X.; Wan, Z.; Hao, Z. A full-modified-Newton step infeasible interior-point method for the special weighted linear complementarity problem. *J. Ind. Manag. Optim.* **2021**, *18*, 2579–2598. [CrossRef]
13. Asadi, S.; Darvay, Z.; Lesaja, G.; Mahdavi-Amiri, N.; Potra, F. A full-Newton step interior-point method for monotone weighted linear complementarity problems. *J. Optim. Theory Appl.* **2020**, *186*, 864–878. [CrossRef]
14. Liu, L.; Liu, S.; Liu, H. A predictor-corrector smoothing Newton method for symmetric cone complementarity problem. *Appl. Math. Comput.* **2010**, *217*, 2989–2999. [CrossRef]
15. Narushima, Y.; Sagara, N.; Ogasawara, H. A smoothing Newton method with Fischer-Burmeister function for second-order cone complementarity problems. *J. Optim. Theory Appl.* **2011**, *149*, 79–101. [CrossRef]
16. Liu, X.; Liu, S. A new nonmonotone smoothing Newton method for the symmetric cone complementarity problem with the Cartesian P_0-property. *Math. Method Oper. Res.* **2020**, *92*, 229–247. [CrossRef]
17. Chen, P.; Lin, G.; Zhu, X.; Bai, F. Smoothing Newton method for nonsmooth second-order cone complementarity problems with application to electric power markets. *J. Glob. Optim.* **2021**, *80*, 635–659. [CrossRef]
18. Zhou, S.; Pan, L.; Xiu, N.; Qi, H. Quadratic convergence of smoothing Newton's method for 0/1 Loss optimization. *SIAM J. Optim.* **2021**, *31*, 3184–3211. [CrossRef]
19. Khouja, R.; Mourrain, B.; Yakoubsohn, J. Newton-type methods for simultaneous matrix diagonalization. *Calcolo* **2022**, *59*, 38. [CrossRef]
20. Zhang, J. A smoothing Newton algorithm for weighted linear complementarity problem. *Optim. Lett.* **2016**, *10*, 499–509.
21. Tang, J; Zhang, H. A nonmonotone smoothing Newton algorithm for weighted complementarity problem. *J. Optim. Theory Appl.* **2021**, *189*, 679–715. [CrossRef]
22. Potra, F.A.; Ptak, V. Nondiscrete induction and iterative processes. *SIAM Rev.* **1987**, *29*, 505–506.
23. Kou, J.; Li, Y.; Wang, X. A modification of Newton method with third-order convergence. *Appl. Math. Comput.* **2006**, *181*, 1106–1111. [CrossRef]
24. Magrenan Ruiz, A.A.; Argyros, I.K. Two-step Newton methods. *J. Complex.* **2014**, *30*, 533–553. [CrossRef]
25. Gander, W. On Halley's iteration method. *Amer. Math.* **1985**, *92*, 131–134. [CrossRef]
26. Ezquerro, J.A.; Hernandez, M.A. On a convex acceleration of Newton's method. *J. Optim. Theory Appl.* **1999**, *100*, 311–326. [CrossRef]

27. Tang, J. A variant nonmonotone smoothing algorithm with improved numerical results for large-scale LWCPs. *Comput. Appl. Math.* **2018**, *37*, 3927–3936. [CrossRef]
28. Li, D.; Fukushima, M. A globally and superlinearly convergent Gauss-Newton-based BFGS method for symmetric nonlinear equations. *SIAM J. Numer. Anal.* **1999**, *37*, 152–172. [CrossRef]
29. Dennis, J.; More, J. A characterization of superlinear convergence and its applications to quasi-Newton methods. *Math. Comput.* **1974**, *28*, 549–560. [CrossRef]