Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 12, No. 1, pp. 185–206, January 2016

NEW SMOOTHING FUNCTIONS FOR SOLVING A SYSTEM OF EQUALITIES AND INEQUALITIES

Jein-Shan Chen*, Chun-Hsu Ko, Yan-Di Liu and Sheng-Pen Wang†

Abstract: In this paper, we propose a family of new smoothing functions for solving a system of equalities and inequalities, which is a generalization of [13]. We then investigate an algorithm based on a new reformation $\widehat{H}$ with less dimensionality and show, as in [13], that it is globally and locally convergent under suitable assumptions. Numerical evidence shows the better performance of the algorithm in the sense that some unsolved examples in [13] can be solved by our proposed method. Moreover, the involved parameters in the family of new smoothing functions does not have influence in the algorithm, which is a new discovery to the literature.

Key words: *smoothing function, system of equations and inequalities, convergence.*

Mathematics Subject Classification: *26B05, 26B35, 65K05, 90C33.*

---

1 Introduction and Motivation

The target problem of this paper is the following system of equalities and inequalities:

$$\left\{ \begin{array}{lcl} f_I(x) & \leq & 0 \\ f_E(x) & = & 0 \end{array} \right. \tag{1.1}$$

where $I = \{1, 2, \cdots, m\}$ and $E = \{m+1, m+2, \cdots, n\}$. In other words, the function $f_I : \mathbb{R}^n \to \mathbb{R}^m$ is given by

$$f_I(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_m(x) \end{bmatrix}$$

where $f_i : \mathbb{R}^n \to \mathbb{R}$ for $i = \{1, 2, \cdots, m\}$; and the function $f_E : \mathbb{R}^n \to \mathbb{R}^{n-m}$ is given by

$$f_E(x) = \begin{bmatrix} f_{m+1}(x) \\ f_{m+2}(x) \\ \vdots \\ f_n(x) \end{bmatrix}$$

---

*The author's work is supported by Ministry of Science and Technology, Taiwan.
†Corresponding author.

where $f_j : \mathbb{R}^n \to \mathbb{R}$ for $j = \{m+1, m+2, \cdots, n\}$. For simplicity, throughout this paper, we denote $f : \mathbb{R}^n \to \mathbb{R}^n$ as

$$f(x) := \begin{bmatrix} f_I(x) \\ f_E(x) \end{bmatrix} = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_m(x) \\ f_{m+1}(x) \\ f_{m+2}(x) \\ \vdots \\ f_n(x) \end{bmatrix}$$

and assume that f is continuously differentiable. When E is empty set, the system (1.1) reduces to a system of inequalities; whereas it reduces to a system of equations when I is empty.

Problems in form of (1.1) arise in real applications, including data analysis, computer-aided design problems, image reconstructions, and set separation problems, etc.. Many optimization methods have been proposed for solving the system (1.1), for instance, non-interior continuation method [14], smoothing-type algorithm [7, 13], Newton algorithm [8], and iteration methods [5, 9, 10, 12]. In this paper, we consider the similar smoothing-type algorithm studied in [7, 13] for solving the system (1.1). In particular, we propose a family of smoothing functions, investigate its properties, and report numerical performance of an algorithm in which this family of new smoothing functions is involved.

As seen in [7, 13], the main idea of smoothing-type algorithm for solving the system (1.1) is to reformulate system (1.1) as a system of smoothing equations via projection function, More specifically, for any $x = (x_1, x_2, \cdots, x_n) \in \mathbb{R}^n$, one defines

$$(x)_+ := \begin{bmatrix} \max\{0, x_1\} \\ \vdots \\ \max\{0, x_n\} \end{bmatrix}.$$

Then, the system (1.1) is equivalent to the following system of equations:

$$\begin{cases} (f_I(x))_+ & = & 0 \\ f_E(x) & = & 0. \end{cases} \tag{1.2}$$

Note that the function $(f_I(x))_+$ in the reformulation (1.2) is nonsmooth, the classical Newton methods cannot be directly applied to solve (1.2). To conquer this, a smoothing algorithm was considered in [7, 13], in which the following smoothing function was employed:

$$\phi(\mu, t) = \begin{cases} t & \text{if} \quad t \geq \mu, \\ \frac{(t+\mu)^2}{4\mu} & \text{if} \quad -\mu < t < \mu, \\ 0 & \text{if} \quad t \leq -\mu, \end{cases} \tag{1.3}$$

where $\mu > 0$.

In this paper, we propose a family of new smoothing functions, which include the function $\phi(\mu, t)$ given as in (1.3) as a special case, for solving the reformulation (1.2). More specifically, we consider the family of smoothing functions as below:

$$\phi_p(\mu,t) = \begin{cases} t & \text{if} \quad t \geq \frac{\mu}{p-1}, \\ \frac{\mu}{p-1}\left[\frac{(p-1)(t+\mu)}{p\mu}\right]^p & \text{if} \quad -\mu < t < \frac{\mu}{p-1}, \\ 0 & \text{if} \quad t \leq -\mu, \end{cases} \tag{1.4}$$

where $\mu > 0$ and $p \geq 2$. Note that ϕ_p reduces to the smoothing function studied in [13] when $p = 2$. The graphs of ϕ_p with different values of p and various μ are depicted as in Figures 1-3.

Proposition 1.1. *Let ϕ_p be defined as in (1.4). For any $(\mu,t) \in \mathbb{R}_{++} \times \mathbb{R}$, we have*

(a) $\phi_p(.,.)$ *is continuously differentiable at any $(\mu,t) \in \mathbb{R}_{++} \times \mathbb{R}$.*

(b) $\phi_p(0,t) = (t)_+$.

(c) $\frac{\partial \phi_p(\mu,t)}{\partial t} \geq 0$ *for any $(\mu,t) \in \mathbb{R}_{++} \times \mathbb{R}$.*

(d) $\lim_{p\to\infty} \phi_p(\mu,t) \to (t)_+$.

Proof. (a) First, we calculate $\frac{\partial \phi_p(\mu,t)}{\partial t}$ and $\frac{\partial \phi_p(\mu,t)}{\partial \mu}$ as below:

$$\frac{\partial \phi_p(\mu,t)}{\partial t} = \begin{cases} 1 & \text{if} \quad t \geq \frac{\mu}{p-1}, \\ \left[\frac{(p-1)(t+\mu)}{p\mu}\right]^{p-1} & \text{if} \quad -\mu < t < \frac{\mu}{p-1}, \\ 0 & \text{if} \quad t \leq -\mu, \end{cases}$$

$$\frac{\partial \phi_p(\mu,t)}{\partial \mu} = \begin{cases} 0 & \text{if} \quad t \geq \frac{\mu}{p-1}, \\ \left[\frac{(p-1)(t+\mu)}{p\mu}\right]^{p-1} \frac{(t+\mu-pt)}{p\mu} & \text{if} \quad -\mu < t < \frac{\mu}{p-1}, \\ 0 & \text{if} \quad t \leq -\mu, \end{cases}$$

Then, we see that $\frac{\partial \phi_p(\mu,t)}{\partial t} \in C^1$ because

$$\begin{aligned} \lim_{t\to\frac{\mu}{p-1}} \frac{\partial \phi_p(\mu,t)}{\partial t} &= \lim_{t\to\frac{\mu}{p-1}} \left[\frac{(p-1)(\frac{\mu}{p-1}+\mu)}{p\mu}\right]^{p-1} = 1, \\ \lim_{t\to-\mu} \frac{\partial \phi_p(\mu,t)}{\partial t} &= \lim_{t\to-\mu} \left[\frac{(p-1)(-\mu+\mu)}{p\mu}\right]^{p-1} = 0. \end{aligned}$$

and $\frac{\partial \phi_p(\mu,t)}{\partial \mu} \in C^1$ since

$$\begin{aligned} \lim_{t\to\frac{\mu}{p-1}} \frac{\partial \phi_p(\mu,t)}{\partial \mu} &= \lim_{t\to\frac{\mu}{p-1}} \left[\frac{(p-1)(\frac{\mu}{p-1}+\mu)}{p\mu}\right]^{p-1} \frac{(\frac{\mu}{p-1}+\mu-p\frac{\mu}{p-1})}{p\mu} = 0, \\ \lim_{t\to-\mu} \frac{\partial \phi_p(\mu,t)}{\partial \mu} &= \lim_{t\to-\mu} \left[\frac{(p-1)(-\mu+\mu)}{p\mu}\right]^{p-1} \frac{(-\mu+\mu-p(-\mu))}{p\mu} = 0. \end{aligned}$$

The above verifications imply that $\phi_p(.,.)$ is continuously differentiable.

(b) From the definition of $\phi_p(\mu,t)$, it is clear that

$$\phi_p(0,t) = \begin{cases} t & \text{if} \quad t \geq 0 \\ 0 & \text{if} \quad t \leq 0 \end{cases} = (t)_+$$

which is the desired result.

(c) When $-\mu < t < \frac{\mu}{p-1}$, we have $t + \mu > 0$. Hence, from the expression of $\frac{\partial \phi_p(\mu,t)}{\partial t}$, it is obvious that $\left[\frac{(p-1)(t+\mu)}{p\mu}\right]^{p-1} \geq 0$, which says $\frac{\partial \phi_p(\mu,t)}{\partial t} \geq 0$.

(d) Part(d) is clear from the definition. □

The properties of ϕ_p in Proposition 1.1 can be verified via the graphs. In particular, in Figures 1-2, we see that when $\mu \to 0$, $\phi_p(\mu, t)$ goes to $(t)_+$ which verifies Proposition 1.1(b).

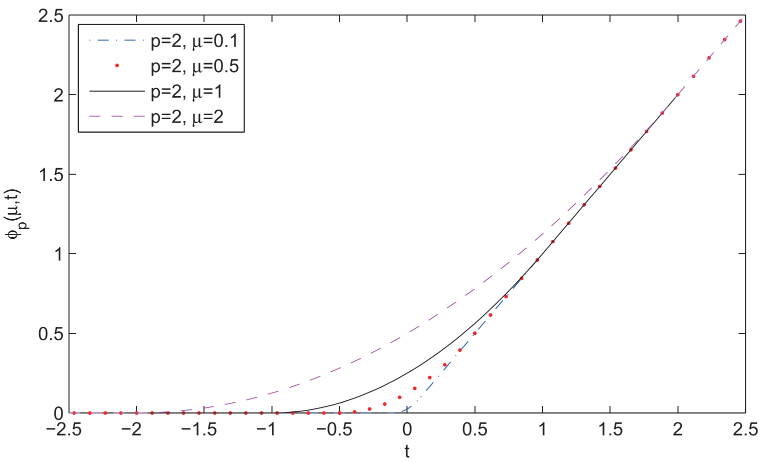

Figure 1: Graphs of $\phi_p(\mu, t)$ with $p = 2$ and $\mu = 0.1, 0.5, 1, 2$.

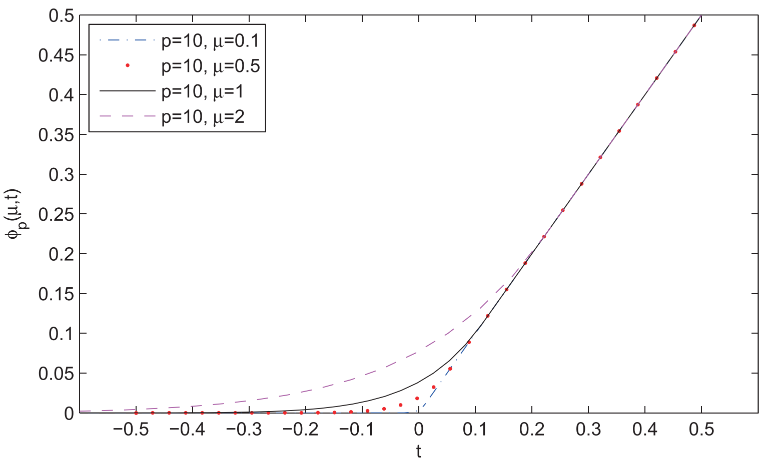

Figure 2: Graphs of $\phi_p(\mu, t)$ with $p = 10$ and $\mu = 0.1, 0.5, 1, 2$.

Figure 3 says that for fixed $\mu > 0$, $\phi_p(\mu, t)$ approaches to $(t)_+$ as $p \to \infty$. This also verifies Proposition 1.1(d).

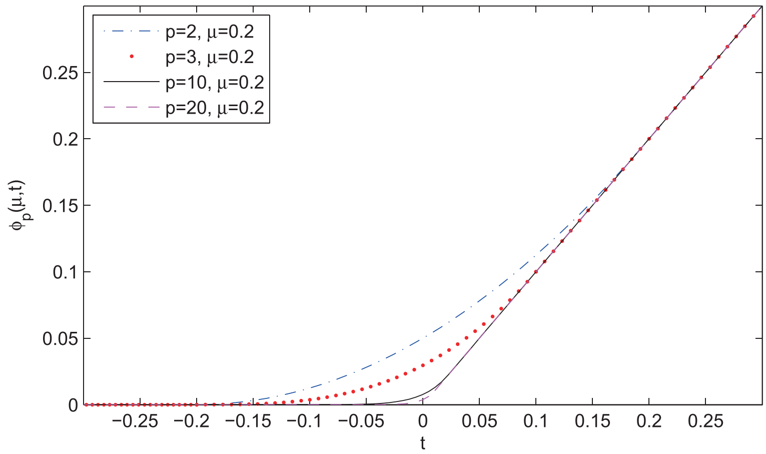

Figure 3: Graphs of $\phi_p(\mu,t)$ with $p=2,3,10,20$ and $\mu=0.2$.

Next, we will form another reformulation for problem (1.1). To this end, we define

$$F(z) := \begin{bmatrix} f_I(x)-s \\ f_E(x) \\ \Phi_p(\mu,s) \end{bmatrix} \quad \text{with} \quad \Phi_p(\mu,s) := \begin{bmatrix} \phi_p(\mu,s_1) \\ \vdots \\ \phi_p(\mu,s_m) \end{bmatrix} \quad \text{and} \quad z=(\mu,x,s) \tag{1.5}$$

where Φ_p is a mapping from $\mathbb{R}^{1+m} \to \mathbb{R}^m$. Then, in light of Proposition 1.1(b), we see that

$$F(z)=0 \text{ and } \mu=0 \quad \Longleftrightarrow \quad s=f_I(x),\ s_+=0,\ f_E(x)=0.$$

This, together with Proposition 1.1(a), indicates that one can solve system (1.1) by applying Newton-type methods to solve $F(z)=0$ by letting $\mu \downarrow 0$. Furthermore, by introducing an extra parameter p, we define a function $H:\mathbb{R}^{1+n+m} \to \mathbb{R}^{1+n+m}$ by

$$H(z) := \begin{bmatrix} \mu \\ f_I(x)-s+\mu x_I \\ f_E(x)+\mu x_E \\ \Phi_p(\mu,s)+\mu s \end{bmatrix} \tag{1.6}$$

where $x_I=(x_1,x_2,\cdots,x_m)$, $x_E=(x_{m+1},x_{m+2},\cdots,x_n)$, $s\in\mathbb{R}^m$, $x:=(x_I,x_E)\in\mathbb{R}^n$ and functions ϕ_p and Φ_p are defined as in(1.4) and (1.5), respectively. Thereby, it is obvious that if $H(z)=0$, then $\mu=0$ and x solves the system (1.1). It is not difficult to see that, for any $z\in\mathbb{R}_{++}\times\mathbb{R}^n\times\mathbb{R}^m$, the function H is continuously differentiable. Let H' denote the Jacobian of the function H. Then, for any $z\in\mathbb{R}_{++}\times\mathbb{R}^n\times\mathbb{R}^m$, we have

$H^{'}(z) =$

$$
\begin{bmatrix}
1 & O_{1\times n} & O_{1\times m} \\
\begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix}_{m\times 1} & A & \begin{bmatrix} -1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & -1 \end{bmatrix}_{m\times m} \\
\begin{bmatrix} x_{m+1} \\ \vdots \\ x_n \end{bmatrix}_{(n-m)\times 1} & B & \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix}_{(n-m)\times m} \\
\begin{bmatrix} s_1 + \frac{\partial}{\partial\mu}\phi^{'}(\mu, s_1) \\ \vdots \\ s_m + \frac{\partial}{\partial\mu}\phi^{'}(\mu, s_m) \end{bmatrix}_{m\times 1} & \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & 0 & \vdots \\ 0 & \cdots & 0 \end{bmatrix}_{m\times n} & \begin{bmatrix} \frac{\partial}{\partial s}\phi^{'}(\mu, s_1) + \mu & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{\partial}{\partial s}\phi^{'}(\mu, s_m) + \mu \end{bmatrix}_{m\times m}
\end{bmatrix}
$$

where

$$
A = \begin{bmatrix} \frac{\partial f_1(x_1)}{\partial x_1} + \mu & \cdots & 0 & \\ \vdots & \ddots & \vdots & 0_{m\times(n-m)} \\ 0 & \cdots & \frac{\partial f_m(x_m)}{\partial x_m} + \mu & \end{bmatrix}_{m\times n}
$$

and

$$
B = \begin{bmatrix} & \frac{\partial f_{m+1}(x_{m+1})}{\partial x_{m+1}} + \mu & \cdots & 0 \\ 0_{(n-m)\times m} & \vdots & \ddots & \vdots \\ & 0 & \cdots & \frac{\partial f_n(x_n)}{\partial x_n} + \mu \end{bmatrix}_{(n-m)\times n}
$$

With the above, we can simplify the matrix $H'(z)$ as

$$
H^{'}(z) = \begin{bmatrix} 1 & 0_n & 0_m \\ x_I & f_I^{'}(x) + \mu U & -I_m \\ x_E & f_E^{'}(x) + \mu V & 0_{(n-m)\times m} \\ s + \Phi_\mu^{'}(\mu, s) & 0_{m\times n} & \Phi_s^{'}(\mu, s) + \mu I_m \end{bmatrix} \tag{1.7}
$$

where

$$
U := [I_m \quad 0_{m\times(n-m)}], \qquad V := [0_{(n-m)\times m} \quad I_{n-m}],
$$

$$
s + \Phi_\mu^{'}(\mu, s) = \begin{bmatrix} s_1 + \frac{\partial}{\partial\mu}\phi^{'}(\mu, s_1) \\ \vdots \\ s_m + \frac{\partial}{\partial\mu}\phi^{'}(\mu, s_m) \end{bmatrix}_{m\times 1},
$$

$$
\Phi_s^{'}(\mu, s) + \mu I_m = \begin{bmatrix} \frac{\partial}{\partial s}\phi^{'}(\mu, s_1) + \mu & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{\partial}{\partial s}\phi^{'}(\mu, s_m) + \mu \end{bmatrix}_{m\times m}.
$$

Here, we use 0_I to denote the I-dimensional zero vector and $0_{l\times q}$ to denote the $l \times q$ zero matrix for any positive integers l and q. Thus, we might apply some Newton-type methods to solve the system of smooth equations $H(z) = 0$ at each iteration by letting $\mu > 0$ and $H(z) \to 0$ so that a solution of (1.1) can be found. This is the main idea of smoothing approach for solving system (1.1).

Alternatively, one may have another smoothing reformulation for system (1.1) without introducing the extra variable s. More specifically, we can define $\widehat{H} : \mathbb{R}^{1+n} \to \mathbb{R}^{1+n}$ as

$$\widehat{H}(\mu, x) := \begin{bmatrix} \mu \\ f_E(x) + \mu x_E \\ \Phi_p(\mu, f_I(x)) + \mu x_I \end{bmatrix} \tag{1.8}$$

The Jacobian of $\widehat{H}(\mu, x)$ is similar to $H'(z)$ and indeed is a bit tedious, so we omit its presentation here. The reformulation of $\widehat{H}(\mu, x) = 0$ has less dimension than $H(z) = 0$, whereas the expression of $\widehat{H}'(\mu, x)$ is more tedious than $H'(z)$. Both smoothing approaches can lead to the solution to system (1.1). The numerical results based on $H(z) = 0$ and $\widehat{H}(\mu, x) = 0$ are compared in this paper. Moreover, we also investigate how the parameter p affect the numerical performance when different ϕ_p is employed. Proposing the new family of smoothing functions as well as the above two aspects of numerical points are the main motivation and contribution of this paper.

2 A smoothing-type algorithm

In this section, we consider a non-monotone smoothing-type algorithm whose similar framework has been discussed in [7, 13]. In particular, we correct a flaw in Step 5 in [13] and show that only this modification can really make the algorithm well-defined. Moreover, for $\widehat{H}(\mu, x)$, a new reformulation of $H(z)$ with lower dimensionality, we will use the function $\psi(\cdot) := \|H(z)\|^2$ or $\psi(\cdot) := \|\widehat{H}(\mu, x)\|^2$ alternatively. Below are the details of the algorithm.

Algorithm 2.1. (A Nonmonotone Smoothing-Type Algorithm)

Step 0 Choose $\delta \in (0, 1), \sigma \in (0, 1/2), \beta > 0$. Take $\tau \in (0, 1)$ such that $\tau\beta < 1$. Let $\mu_0 = \beta$ and $(x^0, s^0) \in \mathbb{R}^{n+m}$ be an arbitrary vector. Set $z^0 := (\mu_0, x^0, s^0)$. Take $e^0 := (1, 0, \cdots, 0) \in \mathbb{R}^{1+n+m}$, $R_0 := \|H(z^0)\|^2 = \psi(z^0)$ and $Q_0 = 1$.
Choose $\eta_{\min}$ and $\eta_{\max}$ such that $0 \le \eta_{\min} \le \eta_{\max} < 1$. Set $\theta(z^0) := \tau \min\{1, \psi(z^0)\}$ and $k := 0$.

Step 1 If $\|H(z^k)\| = 0$, stop.

Step 2 Compute $\triangle z^k := (\triangle\mu_k, \triangle x^k, \triangle s^k) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m$ by using

$$H^{'} \triangle z^k = -H(z^k) + \beta\theta(z^k)e^0 \tag{2.1}$$

Step 3 Let α_k be the maximum of the values $1, \delta, \delta^2, \cdots$ such that

$$\psi(z^k + \alpha_k \triangle z^k) \le [1 - 2\sigma(1 - \tau\beta)\alpha_k]\, R_k \tag{2.2}$$

Step 4 Set $z^{k+1} := z^k + \alpha_k \triangle z^k$. If $\|H(z^{k+1})\| = 0$, stop.

Step 5 Choose $\eta_k \in [\eta_{\min}, \eta_{\max}]$. Set

$$\begin{array}{rcl} Q_{k+1} & := & \eta_k Q_k + 1 \\ \theta(z^{k+1}) & := & \min\left\{\tau, \tau\psi(z^{k+1}), \theta(z^k)\right\} \\ R_{k+1} & := & \dfrac{(\eta_k Q_k R_k + \psi(z^{k+1}))}{Q_{k+1}} \end{array} \tag{2.3}$$

and $k := k + 1$. Go to Step 2

In Algorithm 2.1, a nonmonotone line search technique is adopted. It is easy to see that R_{k+1} is a convex combination of R_k and $\psi(z^{k+1})$. Since $R_0 = \psi(z^0)$, it follows that R_k is a convex combination of the function values $\psi(z^0), \psi(z^1), \cdots, \psi(z^k)$. The choice of η_k controls the degree of nonmonotonicity. If $\eta_k = 0$ for every k, then the line search is the usual monotone Armijo line search. The scheme of Algorithm 2.1 is not exactly the same as the one in [13]. In particular, $\theta(z^{k+1}) := \min\left\{\tau, \tau\psi(z^{k+1}), \theta(z^k)\right\}$ which is different from $\theta(z^{k+1}) := \min\left\{\tau, \tau\psi(z^k), \theta(z^k)\right\}$ given in [13]. Only this modification can really make the algorithm well-defined as shown in the following Theorem 2.3. For convenience, we denote

$$f'(x) := \begin{bmatrix} f_I'(x) \\ f_E'(x) \end{bmatrix}$$

and make the following assumption.

Assumption 2.1. *$f'(x) + \mu I_n$ is invertible for any $x \in \mathbb{R}^n$ and $\mu \in \mathbb{R}_{++}$.*

Some basic properties of Algorithm 2.1 are stated in the following lemma. Since the proof arguments are almost the same as those in [13], they are thus omitted.

Lemma 2.1. *Let the sequence $\{R_k\}$ and $\left\{z^k\right\}$ be generated by Algorithm 2.1. Then, the following hold.*

(a) *The sequence $\{R_k\}$ is monotonically decreasing.*

(b) *The function $\psi(z^k) \leq R_k$ for all $k \in \mathscr{J}$.*

(c) *The sequence $\theta(z^k)$ is monotonically decreasing.*

(d) *$\beta\theta(z^k) \leq \mu_k$ for all $k \in \mathscr{J}$.*

(e) *$\mu_k > 0$ for all $k \in \mathscr{J}$ and the sequence $\{\mu_k\}$ is monotonically decreasing.*

Lemma 2.2. *Suppose $A \in \mathbb{R}^{n\times n}$ which is partitioned as $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$ where A_{11} and A_{22} are square matrices. If A_{12} or A_{21} is zero matrix, then* $\det(A) = \det(A_{11}) \cdot \det(A_{22})$.

Proof. This a well known result in matrix analysis, which is a special case of Fischer's ineqiality [2, 6]. Please refer to [11, Theorem 7.3] for a proof. □

Theorem 2.3. *Suppose that f is a continuously differentiable function and Assumption 2.1 is satisfied. Then Algorithm 2.1 is well defined.*

Proof. Applying Lemmas 2.1-2.2 and mimicking the arguments as in [13], it is easy to achieve the desired result. However, we point it out again that $\theta(z^{k+1})$ in step 5 is different from the one in [13]. Only this modification can really make the algorithm well-defined. □

3 Convergence analysis

In this section, we analyze the convergence of the algorithm proposed in previous section. To this end, the following assumption is needed which was introduced in [7].

Assumption 3.1. *For an arbitrary sequence* $\{(\mu_k, x^k)\}$ *with* $\lim_{k\longrightarrow\infty} \|x\| = +\infty$ *and the sequence* $\{\mu_k\} \subset \mathbb{R}_+$ *bounded, then either*

(i) *there is at least an index* i_0 *such that* $\limsup_{k\longrightarrow\infty}\{f_{i_0}(x^k) + \mu_k x^k_{i_0}\} = +\infty$*; or*

(ii) *there is at least an index* i_0 *such that* $\limsup_{k\longrightarrow\infty}\{\mu_k(f_{i_0}(x^k) + \mu_k x^k_{i_0})\} = -\infty$.

It can be seen that many functions satisfy Assumption 3.1, see [7]. The global convergence of Algorithm 2.1 is stated as follows. In fact, with Proposition 1.1, the main idea for the proof is almost the same as that in [13, Theorem 4.1], only a few technical parts are different. Thus, we omit the details.

Theorem 3.1. *Suppose that* f *is a continuously differentiable function and Assumptions 2.1 and 3.1 are satisfied. Then, the sequence* $\{z^k\}$ *generated by Algorithm 2.1 is bounded. Moreover, any accumulation point of* x^k *is a solution to (1.1).*

Next, we analyse the convergence rate for Algorithm 2.1. Before presenting the result, we introduce some concepts that will used in the subsequent analysis as well as a technical lemma.

A locally Lipschitz function $F : \mathbb{R}^n \to \mathbb{R}^m$, which has the generalized Jacobian $\partial F(x)$, is said to be semismooth (or strongly semismooth) at $x \in \mathbb{R}^n$ if F is directionally differentiable at x and

$$F(x + h) - F(x) - Vh = o(\|h\|) \quad (\text{or} = O(\|h\|^2)$$

holds for any $V \in \partial F(x + h)$. It is well known that convex functions, smooth functions, and piecewise linear functions are examples of semismooth functions. The composition of (strongly) semismooth functions is still a (strongly) semismooth function. It can be verified that the function ϕ_p defined by (1.4) is strongly semismooth on $\mathbb{R}^2$. Thus, f being continuously differentiable implies that the function H defined by (1.6) and $\widehat{H}$ defined by (1.8)are semismooth (or strongly semismooth if $f^{'}$ is Lipschitz continuous on $\mathbb{R}^n$.

Lemma 3.2. *For any* $\alpha, \beta \in \mathbb{R}_{++}$, $\alpha = O(\beta)$ *represents that* $\frac{\alpha}{\beta}$ *is uniformly bounded, and* $\alpha = o(\beta)$ *denotes* $\frac{\alpha}{\beta} \to 0$ *as* $\beta \to 0$*. Then, we have*

(a) $O(\beta) \pm O(\beta) = O(\beta)$;

(b) $o(\beta) \pm o(\beta) = o(\beta)$;

(c) *If* $c \neq 0$ *then* $O(c\beta) = O(\beta)$, $o(c\beta) = o(\beta)$;

(d) $O(o(\beta)) = o(\beta)$, $o(O(\beta)) = o(\beta)$;

(e) $O(\beta_1)O(\beta_2) = O(\beta_1\beta_2)$, $O(\beta_1)o(\beta_2) = o(\beta_1\beta_2)$, $o(\beta_1)O(\beta_2) = o(\beta_1\beta_2)$.

(f) *If* $\alpha = O(\beta_1)$ *and* $\beta_1 = o(\beta_2)$*, then* $\alpha = o(\beta_2)$*.*

Proof. For parts (a)-(e), please refer to [1] for a proof. Part (f) can be verified straightforwardly. $\square$

With Proposition 1.1 and Lemma 3.2, mimicking the arguments as in [13, Theorem 5.1] gives the following theorem.

Theorem 3.3. *Suppose that f is a continuously differentiable function and Assumptions 2.1 and 3.1 are satisfied. Let $z^* = (\mu_*, x^*, s^*)$ be an accumulation point of $\{z^k\}$ generated by Algorithm 2.1. If all $V \in \partial H(z^*)$ are nonsingular, then the following hold.*

(a) $\alpha_k \equiv 1$ *for all* z^k *sufficiently close to* z^**;*

(b) *the whole sequence* $\{z^k\}$ *converges to* z^**;*

(c) $\|z^{k+1} - z^*\| = o(\|z^k - z^*\|)$ *(or* $\|z^{k+1} - z^*\| = O(\|z^k - z^*\|^2)$*) provided* f' *is Lipschitz continuous on* $\mathbb{R}^n$ *);*

(d) $\mu_{k+1} = o(\mu_k)$ *(or* $\mu_{k+1} = O(\mu_k^2)$ *if* f' *is Lipschitz continuous on* $\mathbb{R}^n$*).*

4 Numerical Results

In this section, we present our test problems and report numerical results. In this paper, the function f is assumed to be a mapping from $\mathbb{R}^n$ to $\mathbb{R}^n$, which means the dimension of x is exactly the same as the total number of inequalities and equalities. In reality, this may not be the case. In other words, there may have a system like this:

$$\left\{ \begin{array}{lcll} f_I(x) & \leq & 0, & I = \{1, 2, \cdots, m\} \\ f_E(x) & = & 0, & E = \{m, m+1, \cdots, l\} \end{array} \right. \tag{4.1}$$

This says f could be a mapping from $\mathbb{R}^n$ to $\mathbb{R}^l$. When $l \neq n$, the scheme in Algorithm 2.1 cannot be applied to the system (4.1) because the dimension of x is not equal to the total number of inequalities and equalities. To make system (4.1) solvable under the proposed algorithm, as remarked in [13, Sec. 6], some additional inequality or variable needs to be added. For example,

(i) if $l < n$, we may add a trivial inequality like

$$\sum_{i=1}^{n} x_i^2 \leq M$$

where M is sufficiently large, into system (4.1) so that Algorithm 2.1 can be applied.

(ii) if $l > n$ and $m \geq 1$, we may add a variable x_{n+1} into the inequalities so that

$$f_i(x) \leq 0 \quad \to \quad f_i(x) + x_{n+1}^2 \leq 0.$$

(iii) if $l > n$ and $m = 0$, we may add a trivial inequality like

$$\sum_{i=1}^{n+2} x_i^2 \leq M$$

where M is sufficiently large, into system (4.1) so that Algorithm 2.1 can be applied.

In real implementation, the $H(z)$ given as in (1.6) is replaced by

$$H(z) := \begin{bmatrix} \mu \\ f_I(x) - s + c\mu x_I \\ f_E(x) + c\mu x_E \\ \Phi_p(\mu, s) + c\mu s \end{bmatrix} \tag{4.2}$$

where c is a given constant. Likewise, the $\widehat{H}(\mu, x)$ given as in (1.8) is replaced by

$$\widehat{H}(\mu, x) := \begin{bmatrix} \mu \\ f_E(x) + c\mu x_E \\ \Phi_p(\mu, f_I(x)) + c\mu x_I \end{bmatrix}. \tag{4.3}$$

Adding such a constant c is helpful when coding the algorithm because μ approaches to zero eventually. The theoretical results will not be affected in any case. In practice, in order to obtain an interior solution x^* for inequalities ($f_I(x^*) < 0$), the following system

$$\begin{cases} f_I(x) + \varepsilon e \leq 0 \\ f_E(x) = 0 \end{cases}$$

is considered, where ε is a small number and e is the vector of all ones. Now, we list the test problems which are employed from [7, 13].

Example 4.1. Consider $f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix}$ with $x \in \mathbb{R}^2$ where

$$\begin{aligned} f_1(x) &= x_1^2 + x_2^2 - 1 + \varepsilon \leq 0, \\ f_2(x) &= -x_1^2 - x_2^2 + (0.999)^2 + \varepsilon \leq 0. \end{aligned}$$

Example 4.2. Consider $f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \\ f_5(x) \\ f_6(x) \end{bmatrix}$ with $x \in \mathbb{R}^2$ where

$$\begin{aligned} f_1(x) &= \sin(x_1) + \varepsilon \leq 0, \\ f_2(x) &= -\cos(x_2) + \varepsilon \leq 0, \\ f_3(x) &= x_1 - 3\pi + \varepsilon \leq 0, \\ f_4(x) &= x_2 - \frac{\pi}{2} - 2 + \varepsilon \leq 0, \\ f_5(x) &= -x_1 - \pi + \varepsilon \leq 0, \\ f_6(x) &= -x_2 - \frac{\pi}{2} + \varepsilon \leq 0. \end{aligned}$$

Example 4.3. Consider $f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix}$ with $x \in \mathbb{R}^2$ where

$$\begin{aligned} f_1(x) &= \sin(x_1) + \varepsilon \le 0, \\ f_2(x) &= -\cos(x_2) + \varepsilon \le 0. \end{aligned}$$

Example 4.4. Consider $f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \\ f_5(x) \end{bmatrix}$ with $x \in \mathbb{R}^5$ where

$$\begin{aligned} f_1(x) &= x_1 + x_3 - 1.6 + \varepsilon \le 0, \\ f_2(x) &= 1.333x_2 + x_4 - 3 + \varepsilon \le 0, \\ f_3(x) &= -x_3 - x_4 + x_5 + \varepsilon \le 0, \\ f_4(x) &= x_1^2 + x_3^2 - 1.25 = 0, \\ f_5(x) &= x_2^{1.5} + 1.5x_4 - 3 = 0. \end{aligned}$$

Example 4.5. Consider $f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \end{bmatrix}$ with $x \in \mathbb{R}^3$ where

$$\begin{aligned} f_1(x) &= x_1 + x_2 e^{0.8x_3} + e^{1.6} + \varepsilon \le 0, \\ f_2(x) &= x_1^2 + x_2^2 + x_3^2 - 5.2675 + \varepsilon \le 0, \\ f_3(x) &= x_1 + x_2 + x_3 - 0.2605 = 0. \end{aligned}$$

Example 4.6. Consider $f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \end{bmatrix}$ with $x \in \mathbb{R}^2$ where

$$\begin{aligned} f_1(x) &= 0.8 - e^{x_1 + x_2} + \varepsilon \le 0, \\ f_2(x) &= 1.21e^{x_1} + e^{x_2} - 2.2 = 0, \\ f_3(x) &= x_1^2 + x_2^2 + x_2 - 0.1135 = 0. \end{aligned}$$

Example 4.7. Consider $f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix}$ with $x \in \mathbb{R}^2$ where

$$\begin{aligned} f_1(x) &= x_1 - 0.7\sin(x_1) - 0.2\cos(x_2) = 0 \\ f_2(x) &= x_2 - 0.7\cos(x_1) + 0.2\sin(x_2) = 0 \end{aligned}$$

Moreover, in light of the aforementioned discussions, there have corresponding modified problems for Example 4.2', Example 4.6', and Example 4.7', which are stated as below. The other examples are kept unchanged. In other words, Example 4.1 and Example 4.1' are the same , so are Example 4.3 and Example 4.3', Example 4.4 and Example 4.4', Example 4.5 and Example 4.5'.

Example 4.2'. Consider $f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \\ f_5(x) \\ f_6(x) \end{bmatrix}$ with $x \in \mathbb{R}^6$ where

$$\begin{aligned} f_1(x) &= \sin(x_1) + \varepsilon \leq 0 \\ f_2(x) &= -\cos(x_2) + \varepsilon \leq 0 \\ f_3(x) &= x_1 - 3\pi + x_3^2 + \varepsilon \leq 0 \\ f_4(x) &= x_2 - \frac{\pi}{2} - 2 + x_4^2 + \varepsilon \leq 0 \\ f_5(x) &= -x_1 - \pi + x_5^2 + \varepsilon \leq 0 \\ f_6(x) &= -x_2 - \frac{\pi}{2} + x_6^2 + \varepsilon \leq 0 \end{aligned}$$

Example 4.6'. Consider $f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \end{bmatrix}$ with $x \in \mathbb{R}^3$ where

$$\begin{aligned} f_1(x) &= 0.8 - e^{x_1+x_2} + x_3^2 + \varepsilon \leq 0, \\ f_2(x) &= 1.21e^{x_1} + e^{x_2} - 2.2 = 0, \\ f_3(x) &= x_1^2 + x_2^2 + x_2 - 0.1135 = 0. \end{aligned}$$

Example 4.7'. Consider $f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \end{bmatrix}$ with $x \in \mathbb{R}^3$ where

$$\begin{aligned} f_1(x) &= x_1^2 + x_2^2 + x_3^2 - 10000 + \varepsilon \leq 0, \\ f_2(x) &= x_1 - 0.7\sin(x_1) - 0.2\cos(x_2) = 0, \\ f_3(x) &= x_2 - 0.7\cos(x_1) + 0.2\sin(x_2) = 0. \end{aligned}$$

The numerical implementations are coded in Matlab. In the numerical reports, x^0 is the stating point, NI is the total number of iterations, NF denotes the number of function evaluations for $H(z^k)$ or $\widehat{H}(\mu_k, x^k)$, and SOL means the solution obtained from the algorithm. The parameters used in the algorithm are set as

$$\varepsilon = 0.00001, \quad \delta = 0.3, \quad \sigma = 0.001, \quad \beta = 1.0, \quad \mu_0 = 1.0, \quad Q_0 = 1.0.$$

In Table 1 and Table 2, we adapt the same x^0, c, τ, η used as in [13] for $p = 2$. Basically, in Table 1 and Table 2, the bottom half data for the modified problems are the same as those in [13], respectively. Below are our numerical observations and conclusions.

- From Table 1 and Table 2, we see that, when employing formulation $H(z) = 0$, solving the modified problems is more successful than solving the original problems.
- Table 3 indicates that the numerical results are the same for original problems and modified problems, when $\widehat{H}(\mu, x) = 0$ is employed. Hence, in Tables 4-11, we focus on the modified problems when formulation $H(z) = 0$ is employed, whereas we only test original problems whenever the implementations are based on $\widehat{H}(\mu, x) = 0$.

- From Table 5 ($p = 2$), we see that the algorithm based on $\widehat{H}(\mu, x) = 0$ can solve more problems than the one in [13] does. In view of the lower dimensionality of $\widehat{H}(\mu, x) = 0$ and this performance, we can confirm the merit of this new reformulation.
- In Table 4 and Table 5, the initial point and other parameters are the same as those in [13]. In Tables 6-7, we fix the initial point x^0 for all test problems. In Table 8 and Table 9, even x^0, c, τ and η are all fixed for all test problems. In Table 10 and Table 11, x^0 is fixed for all test problems and parts of c, τ and η are fixed. In general, we observe that the numerical performance based on the formulation $\widehat{H}(\mu, x) = 0$ is better than the one based on $H(z) = 0$.
- Moreover, the changing of parameter p seems have no influence on the numerical performance no matter $\widehat{H}(\mu, x) = 0$ or $H(z) = 0$ is adapted. This indicates that the smoothing approach may not be affected when p is perturbed. This phenomenon is different from the one for other approaches observed in [3, 4] and is a new discovery to the literature. We guess that the main reason comes from μ dominating the algorithm in the smoothing approach even various p is perturbed. This conjecture still needs further verification and investigation.

In summary, the main contribution of this paper is to propose a new family of smoothing functions and correct a flaw in an algorithm studied in [13], which is used to guarantee its convergence. We believe that the proposed new smoothing functions can be also employed in other contexts where the projection function is involved. The related numerical performance can be investigated accordingly. We leave them as future research topics.

Acknowledgements. We are gratefully indebted to anonymous referees for their valuable suggestions that help us to improve the presentation of the paper.

References

[1] R.G. Bartle, *The Elements of Real Analysis, Second Edition* John Wiley, New Jersey, 1976.

[2] R. Bhatia, *Matrix Analysis*, Springer-Verlag, New York, 1997.

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

[4] J-S. Chen, S-H. Pan and C-Y. Yang, Numerical comparison of two effective methods for mixed complementarity problems, *J. Comput. Appl. Math.* 234 (2010) 667-683.

[5] J.W. Daniel, Newton's method for nonlinear inequalities, *Numer. Math.* 21 (1973) 381-387.

[6] R. A. Horn and C.R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, 1986.

[7] Z-H. Huang, Y. Zhang and W. Wu, A smoothing-type algorithm for solving stsyem of inequalities, *J. Comput. Appl. Math.* 220 (2008) 355-363.

[8] S.L. HU, Z-H. Huang and P. Wang , A non-monotone smoothing Newton algorithm for solving nonlinear complementarity problems, *Optim. Method. Softw.* 24 (2009) 447-460.

[9] D.Q. Mayne, E. Polak and A.J. Heunis, Solving nonlinear inequalities in a finite number of iterations, *J. Optimiz. Theory. App.* 33 (1981) 207-221.

[10] M. Sahba, On the solution of nonlinear inequalities in a finite number of iterations, *Numer. Math.* 46 (1985) 229-236.

[11] J.M. Schott, *Matrix Analysis for Statistics*, 2nd edition, John Wiley, New Jersey, 2005.

[12] H-X. Ying, Z-H. Huang and L. Qi, The convergence of a Levenberg-Marquard method for the l_2-norm solution of nonlinear inequalities, *Numer. Func. Anal. Opt.* 29 (2008) 687-716.

[13] Y. Zhang and Z-H. Huang, A nonmonotone smoothing-type algorithm for solving a system of equalities and inequalities, *J. Comput. Appl. Math.* 233 (2010) 2312-2321.

[14] J. Zhu and B. Hao, A new non-interior continuation method for solving a system of equalities and inequalities, *J. Appl. Math.*, (2014) Article Number 592540.

Manuscript received 3 December 2014
revised 12 December 2014
accepted for publication 15 December 2014

Jein-Shan Chen
Department of Mathematics
National Taiwan Normal University
Taipei 11677, Taiwan
E-mail address: `jschen@math.ntnu.edu.tw`

Chun-Hsu Ko
Department of Electrical Engineering
I-Shou University
Kaohsiung 840, Taiwan
E-mail address: `chko@isu.edu.tw`

Yan-Di Liu
Department of Mathematics
National Taiwan Normal University
Taipei 11677, Taiwan
E-mail address: `60140026S@ntnu.edu.tw`

Sheng-Pen Wang
Department of Industrial and Business Management
Chang Gung University
Taoyuan 333, Taiwan
E-mail address: `wangsp@mail.cgu.edu.tw`

Table 1: Numerical performance when $p = 2$, stop criterion: 0.001.

Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 5)	100	0.006	0.01	8	12	(-0.6188, 0.7853)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	Fail	Fail	Fail
Ex 4.3	(0, 0)	0.5	0.2	0.01	3	4	(-0.01516, 0.7207)
Ex 4.4	(0.5, 2, 1, 0, 0)	5	0.02	0.8	4	5	(0.5557, 1.324, 0.9703, 0.984, 1.156)
Ex 4.5	(-1, -1, 1)	0.5	0.2	0.8	5	6	(-0.8301, -0.8662, 1.957)
Ex 4.6	(0, 0, 0)	0.5	0.02	0.8	7	8	(0.2743, -0.4975, 1.5e+006)
Ex 4.7	(0, 1, 0)	0.5	0.006	0.8	9	15	(0.5268, 0.5084, -100)
Ex 4.1'	(0, 5)	100	0.006	0.01	8	12	(-0.6188, 0.7853)
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	6	9	(-0.009654,1.428,2.846,1.28,1.639,1.666)
Ex 4.3'	(0, 0)	0.5	0.2	0.01	3	4	(-0.01516, 0.7207)
Ex 4.4'	(0.5, 2, 1, 0, 0)	5	0.02	0.8	4	5	(0.5557, 1.324, 0.9703, 0.984, 1.156)
Ex 4.5'	(-1, -1, 1)	0.5	0.2	0.8	5	6	(-0.8301, -0.8662, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.02	0.8	5	7	(-0.09533, 0.09533, 0.3321)
Ex 4.7'	(0, 1, 0)	0.5	0.006	0.8	9	15	(0.5268, 0.5084, -100)

Note: Based on $H(z) = 0$ given as in (4.2).

Table 2: Numerical performance when $p = 2$, stop criterion: $1e - 006$.

Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 5)	100	0.006	0.01	Fail	Fail	Fail
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	Fail	Fail	Fail
Ex 4.3	(0, 0)	0.5	0.2	0.01	4	5	(-0.01516, 0.7206)
Ex 4.4	(0.5, 2, 1, 0, 0)	5	0.02	0.8	5	6	(0.5563, 1.326, 0.9698, 0.9822, 1.155)
Ex 4.5	(-1, -1, 1)	0.5	0.2	0.8	6	7	(-0.8299, -0.8663, 1.957)
Ex 4.6	(0, 0, 0)	0.5	0.02	0.8	7	8	(0.2743, -0.4975, 1.5e+006)
Ex 4.7	(0, 1, 0)	0.5	0.006	0.8	10	16	(0.5265, 0.5079, -100)
Ex 4.1'	(0, 5)	100	0.006	0.01	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	7	10	(-0.009276,1.429, 2.846,1.279,1.64, 1.667)
Ex 4.3'	(0, 0)	0.5	0.2	0.01	4	5	(-0.01516, 0.7206)
Ex 4.4'	(0.5, 2, 1, 0, 0)	5	0.02	0.8	5	6	(0.5563, 1.326, 0.9698, 0.9822, 1.155)
Ex 4.5'	(-1, -1, 1)	0.5	0.2	0.8	6	7	(-0.8299, -0.8663, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.02	0.8	6	8	(-0.09533, 0.09533, 0.332)
Ex 4.7'	(0, 1, 0)	0.5	0.006	0.8	10	16	(0.5265, 0.5079, -100)

Note: Based on $H(z) = 0$ given as in (4.2).

Table 3: Numerical performance when $p = 2$, stop criterion: 0.001.

Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 5)	100	0.006	0.01	10	15	(0.5942, -0.8031)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	3	4	(-0.01407, 7.663e-006, 0, 0, 0, 0)
Ex 4.3	(0, 0)	0.5	0.2	0.01	3	4	(-0.01407, 7.663e-006)
Ex 4.4	(0.5, 2, 1, 0, 0)	5	0.02	0.8	3	4	(0.5489, 2.066, 0.9741, 0.0204, 9.748e-007)
Ex 4.5	(-1, -1, 1)	0.5	0.2	0.8	24	39	(0.5031, -1.7, 1.458)
Ex 4.6	(0, 0, 0)	0.5	0.02	0.8	3	4	(-0.09533, 0.09533, 0.08515)
Ex 4.7	(0, 1, 0)	0.5	0.006	0.8	3	4	(0.5271, 0.508, 0)
Ex 4.1'	(0, 5)	100	0.006	0.01	10	15	(0.5942, -0.8031)
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	3	4	(-0.01407, 7.663e-006, 0, 0, 0, 0)
Ex 4.3'	(0, 0)	0.5	0.2	0.01	3	4	(-0.01407, 7.663e-006)
Ex 4.4'	(0.5, 2, 1, 0, 0)	5	0.02	0.8	3	4	(0.5489, 2.066, 0.9741, 0.0204, 9.748e-007)
Ex 4.5'	(-1, -1, 1)	0.5	0.2	0.8	24	39	(0.5031, -1.7, 1.458)
Ex 4.6'	(0, 0, 0)	0.5	0.02	0.8	3	4	(-0.09533, 0.09533, 0.1698)
Ex 4.7'	(0, 1, 0)	0.5	0.006	0.8	3	4	(0.5271, 0.508, 0)

Note: Based on $\widehat{H}(\mu, x) = 0$ given as in (4.3).

Table 4: Numerical performance with different p for modified problems.

	$p = 2$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 5)	100	0.006	0.01	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	7	10	(-0.009276,1.429,2.846,1.279,1.64,1.667)
Ex 4.3'	(0, 0)	0.5	0.2	0.01	4	5	(-0.01516, 0.7206)
Ex 4.4'	(0.5, 2, 1, 0, 0)	5	0.02	0.8	5	6	(0.5563, 1.326, 0.9698, 0.9822, 1.155)
Ex 4.5'	(-1, -1, 1)	0.5	0.2	0.8	6	7	(-0.8299, -0.8663, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.02	0.8	6	8	(-0.09533, 0.09533, 0.332)
Ex 4.7'	(0, 1, 0)	0.5	0.006	0.8	10	16	(0.5265, 0.5079, -100)
	$p = 1.5$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 5)	100	0.006	0.01	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	7	10	(-0.03532,1.428, 2.849, 1.278,1.63,1.666)
Ex 4.3'	(0, 0)	0.5	0.2	0.01	4	5	(-0.0189, 0.7217)
Ex 4.4'	(0.5, 2, 1, 0, 0)	5	0.02	0.8	5	6	(0.5546, 1.329, 0.9708, 0.979, 1.135)
Ex 4.5'	(-1, -1, 1)	0.5	0.2	0.8	6	7	(-0.8288, -0.8673, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.02	0.8	6	8	(-0.09533, 0.09533, 0.3509)
Ex 4.7'	(0, 1, 0)	0.5	0.006	0.8	10	16	(0.5265, 0.5079, -100)
	$p = 3$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 5)	100	0.006	0.01	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	7	10	(-0.007125,1.43,2.848,1.281, 1.641,1.667)
Ex 4.3'	(0, 0)	0.5	0.2	0.01	4	5	(-0.01061, 0.72)
Ex 4.4'	(0.5, 2, 1, 0, 0)	5	0.02	0.8	6	7	(0.5589, 1.33, 0.9683, 0.9771, 1.17)
Ex 4.5'	(-1, -1, 1)	0.5	0.2	0.8	6	7	(-0.8313, -0.8648, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.02	0.8	6	7	(-0.09533, 0.09533, 0.4472)
Ex 4.7'	(0, 1, 0)	0.5	0.006	0.8	10	16	(0.5265, 0.5079, -100)
	$p = 8$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 5)	100	0.006	0.01	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	7	10	(-0.00355,1.431,2.849,1.282,1.643,1.668)
Ex 4.3'	(0, 0)	0.5	0.2	0.01	4	5	(-0.001338, 0.7196)
Ex 4.4'	(0.5, 2, 1, 0, 0)	5	0.02	0.8	6	7	(0.5622, 1.351, 0.9664, 0.9528, 1.18)
Ex 4.5'	(-1, -1, 1)	0.5	0.2	0.8	6	7	(-0.8338, -0.8624, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.02	0.8	5	6	(-0.09533, 0.09533, 0.2985)
Ex 4.7'	(0, 1, 0)	0.5	0.006	0.8	10	16	(0.5265, 0.5079, -100)

Note: Based on $H(z) = 0$ given as in (4.2).

Table 5: Numerical performance with different p for original problems.

	$p = 2$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 5)	100	0.006	0.01	12	20	(0.5821, -0.812)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	4	5	(-0.01407, 7.663e-006,0,0,0,0)
Ex 4.3	(0, 0)	0.5	0.2	0.01	4	5	(-0.01407, 7.663e-006)
Ex 4.4	(0.5, 2, 1, 0, 0)	5	0.02	0.8	4	5	(0.549,2.066,0.974,0.02039,9.747e-007)
Ex 4.5	(-1, -1, 1)	0.5	0.2	0.8	25	40	(0.5029, -1.7, 1.458)
Ex 4.6	(0, 0, 0)	0.5	0.02	0.8	4	5	(-0.09533, 0.09533, 0.08515)
Ex 4.7	(0, 1, 0)	0.5	0.006	0.8	4	5	(0.5265, 0.5079, 0)
	$p = 1.5$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 5)	100	0.006	0.01	12	20	(0.5821, -0.812)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	4	5	(-0.01739, 0.002797,0,0,0,0)
Ex 4.3	(0, 0)	0.5	0.2	0.01	4	5	(-0.01739, 0.002797)
Ex 4.4	(0.5, 2, 1, 0, 0)	5	0.02	0.8	4	5	(0.547,2.061,0.9751,0.02811,0.000359)
Ex 4.5	(-1, -1, 1)	0.5	0.2	0.8	Fail	Fail	Fail
Ex 4.6	(0, 0, 0)	0.5	0.02	0.8	4	5	(-0.09533, 0.09533, 0.1086)
Ex 4.7	(0, 1, 0)	0.5	0.006	0.8	4	5	(0.5265, 0.5079, 0)
	$p = 3$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 5)	100	0.006	0.01	12	20	(0.5821, -0.812)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	4	5	(-0.009902, 6.814e-011,0,0,0,0)
Ex 4.3	(0, 0)	0.5	0.2	0.01	4	5	(-0.009902, 6.814e-011)
Ex 4.4	(0.5, 2, 1, 0, 0)	5	0.02	0.8	4	5	(0.5513, 2.071,0.9727,0.01239, 8.581e-012)
Ex 4.5	(-1, -1, 1)	0.5	0.2	0.8	Fail	Fail	Fail
Ex 4.6	(0, 0, 0)	0.5	0.02	0.8	4	5	(-0.09533, 0.09533, 0.06131)
Ex 4.7	(0, 1, 0)	0.5	0.006	0.8	4	5	(0.5265, 0.5079, 0)
	$p = 8$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 5)	100	0.006	0.01	12	20	(0.5821, -0.812)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	4	5	(-0.002048, 6.022e-036, 0, 0, 0, 0)
Ex 4.3	(0, 0)	0.5	0.2	0.01	4	5	(-0.002048, 6.022e-036)
Ex 4.4	(0.5, 2, 1, 0, 0)	5	0.02	0.8	4	5	(0.557, 2.079,0.9694,0.001483,7.47e-037)
Ex 4.5	(-1, -1, 1)	0.5	0.2	0.8	Fail	Fail	Fail
Ex 4.6	(0, 0, 0)	0.5	0.02	0.8	4	5	(-0.09533, 0.09533, 0.01803)
Ex 4.7	(0, 1, 0)	0.5	0.006	0.8	4	5	(0.5265, 0.5079, 0)

Note: Based on $\widehat{H}(\mu, x) = 0$ given as in (4.3).

Table 6: Numerical performance with different p for modified problems.

	$p = 1.5$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 0)	100	0.006	0.01	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	7	10	(-0.03532,1.428,2.849,1.278,1.63,1.666)
Ex 4.3'	(0, 0)	0.5	0.2	0.01	4	5	(-0.0189, 0.7217)
Ex 4.4'	(0, 0, 0, 0, 0)	5	0.02	0.8	Fail	Fail	Fail
Ex 4.5'	(0, 0, 0)	0.5	0.2	0.8	8	13	(-0.8355, -0.8607, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.02	0.8	6	8	(-0.09533, 0.09533, 0.3509)
Ex 4.7'	(0, 0, 0)	5	0.02	0.8	17	21	(0.5265, 0.5079, 100)
	$p = 2$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 0)	100	0.006	0.01	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	7	10	(-0.009276,1.429,2.846,1.279,1.64,1.667)
Ex 4.3'	(0, 0)	0.5	0.2	0.01	4	5	(-0.01516, 0.7206)
Ex 4.4'	(0, 0, 0, 0, 0)	5	0.02	0.8	Fail	Fail	Fail
Ex 4.5'	(0, 0, 0)	0.5	0.2	0.8	8	13	(-0.8356, -0.8606, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.02	0.8	6	8	(-0.09533, 0.09533, 0.332)
Ex 4.7'	(0, 0, 0)	5	0.02	0.8	17	21	(0.5265, 0.5079, 100)
	$p = 3$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 0)	100	0.006	0.01	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	7	10	(-0.007125,1.43,2.848,1.281,1.641,1.667)
Ex 4.3'	(0, 0)	0.5	0.2	0.01	4	5	(-0.01061, 0.72)
Ex 4.4'	(0, 0, 0, 0, 0)	5	0.02	0.8	Fail	Fail	Fail
Ex 4.5'	(0, 0, 0)	0.5	0.2	0.8	8	13	(-0.8356, -0.8606, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.02	0.8	6	7	(-0.09533, 0.09533, 0.4472)
Ex 4.7'	(0, 0, 0)	5	0.02	0.8	17	21	(0.5265, 0.5079, 100)

Note: Based on $H(z) = 0$ given as in (4.2), x^0 is fixed.

Table 7: Numerical performance with different p for original problems.

	$p = 1.5$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 0)	100	0.006	0.01	6	8	(9.084e-006, -0.999)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	4	5	(-0.01739, 0.002797, 0, 0, 0, 0)
Ex 4.3	(0, 0)	0.5	0.2	0.01	4	5	(-0.01739, 0.002797)
Ex 4.4	(0, 0, 0, 0, 0)	5	0.02	0.8	6	8	(1.118, 2.08,0,0, -0.003514)
Ex 4.5	(0, 0, 0)	0.5	0.2	0.8	7	11	(0.5029, -1.7, 1.458)
Ex 4.6	(0, 0, 0)	0.5	0.02	0.8	4	5	(-0.09533, 0.09533, 0.1086)
Ex 4.7	(0, 0, 0)	0.5	0.006	0.8	4	5	(0.5265, 0.5079, 0)
	$p = 2$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 0)	100	0.006	0.01	6	8	(2.4e-008, -0.999)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	4	5	(-0.01407, 7.663e-006, 0, 0, 0, 0)
Ex 4.3	(0, 0)	0.5	0.2	0.01	4	5	(-0.01407, 7.663e-006)
Ex 4.4	(0, 0, 0, 0, 0)	5	0.02	0.8	6	8	(1.118, 2.08, 0, 0, -0.003159)
Ex 4.5	(0, 0, 0)	0.5	0.2	0.8	7	11	(0.5029, -1.7, 1.458)
Ex 4.6	(0, 0, 0)	0.5	0.02	0.8	4	5	(-0.09533, 0.09533, 0.08515)
Ex 4.7	(0, 0, 0)	0.5	0.006	0.8	4	5	(0.5265, 0.5079, 0)
	$p = 3$, stop criterion: $1e-006$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 0)	100	0.006	0.01	6	8	(2.133e-013, -0.999)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.2	0.01	4	5	(-0.009902, 6.814e-011, 0, 0, 0, 0)
Ex 4.3	(0, 0)	0.5	0.2	0.01	4	5	(-0.009902, 6.814e-011)
Ex 4.4	(0, 0, 0, 0, 0)	5	0.02	0.8	6	8	(1.118, 2.08, 0, 0, -0.002428)
Ex 4.5	(0, 0, 0)	0.5	0.2	0.8	7	11	(0.5029, -1.7, 1.458)
Ex 4.6	(0, 0, 0)	0.5	0.02	0.8	4	5	(-0.09533, 0.09533, 0.06131)
Ex 4.7	(0, 0, 0)	0.5	0.006	0.8	4	5	(0.5265, 0.5079, 0)

Note: Based on $\widehat{H}(\mu, x) = 0$ given as in (4.3), x^0 is fixed.

Table 8: Numerical performance with different p for modified problems.

	$p = 1.5$, stop criterion: $1e-006$, $c = 0.5$, $\tau = 0.1$, $\eta = 0.8$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 0)	0.5	0.1	0.8	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.1	0.8	7	11	(-0.03229,1.57, 3.075, 1.414, 1.763, 1.772)
Ex 4.3'	(0, 0)	0.5	0.1	0.8	5	6	(-0.003862, 0.7497)
Ex 4.4'	(0, 0, 0, 0, 0)	0.5	0.1	0.8	Fail	Fail	Fail
Ex 4.5'	(0, 0, 0)	0.5	0.1	0.8	8	13	(-0.8356, -0.8606, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.1	0.8	6	7	(-0.09533, 0.09533, 0.4472)
Ex 4.7'	(0, 0, 0)	0.5	0.1	0.8	17	23	(0.5265, 0.5079, 100)
	$p = 2$, stop criterion: $1e-006$, $c = 0.5$, $\tau = 0.1$, $\eta = 0.8$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 0)	0.5	0.1	0.8	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.1	0.8	7	11	(-0.006013,1.57,3.071,1.414,1.77,1.772)
Ex 4.3'	(0, 0)	0.5	0.1	0.8	5	6	(-0.003179, 0.7516)
Ex 4.4'	(0, 0, 0, 0, 0)	0.5	0.1	0.8	Fail	Fail	Fail
Ex 4.5'	(0, 0, 0)	0.5	0.1	0.8	8	13	(-0.8356, -0.8606, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.1	0.8	6	7	(-0.09533, 0.09533, 0.4472)
Ex 4.7'	(0, 0, 0)	0.5	0.1	0.8	17	23	(0.5265, 0.5079, 100)
	$p = 3$, stop criterion: $1e-006$, $c = 0.5$, $\tau = 0.1$, $\eta = 0.8$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 0)	0.5	0.1	0.8	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.1	0.8	7	11	(-0.005507,1.57, 3.071,1.414,1.771,1.772)
Ex 4.3'	(0, 0)	0.5	0.1	0.8	5	6	(-0.002279, 0.7528)
Ex 4.4'	(0, 0, 0, 0, 0)	0.5	0.1	0.8	Fail	Fail	Fail
Ex 4.5'	(0, 0, 0)	0.5	0.1	0.8	8	13	(-0.8356, -0.8606, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.1	0.8	6	7	(-0.09533, 0.09533, 0.4472)
Ex 4.7'	(0, 0, 0)	0.5	0.1	0.8	17	23	(0.5265, 0.5079, 100)
	$p = 8$, stop criterion: $1e-006$, $c = 0.5$, $\tau = 0.1$, $\eta = 0.8$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 0)	0.5	0.1	0.8	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.1	0.8	7	11	(-0.002994,1.57, 3.07,1.414,1.771,1.772)
Ex 4.3'	(0, 0)	0.5	0.1	0.8	5	6	(-0.0007597, 0.7535)
Ex 4.4'	(0, 0, 0, 0, 0)	0.5	0.1	0.8	Fail	Fail	Fail
Ex 4.5'	(0, 0, 0)	0.5	0.1	0.8	8	13	(-0.8356, -0.8606, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.1	0.8	5	6	(-0.09533, 0.09533, 0.3771)
Ex 4.7'	(0, 0, 0)	0.5	0.1	0.8	17	23	(0.5265, 0.5079, 100)

Note: Based on $H(z) = 0$ given as in (4.2), x^0, c, τ and η are fixed.

Table 9: Numerical performance with different p for original problems.

	$p = 1.5$, stop criterion: $1e-006$, $c = 0.5$, $\tau = 0.1$, $\eta = 0.8$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 0)	0.5	0.1	0.8	9	12	(1, 0)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.1	0.8	3	4	(-0.004177, 0.003252,0,0,0,0)
Ex 4.3	(0, 0)	0.5	0.1	0.8	3	4	(-0.004177, 0.003252)
Ex 4.4	(0, 0, 0, 0, 0)	0.5	0.1	0.8	4	6	(1.118,1.98,-2.397e-017,0.1421,0.05133)
Ex 4.5	(0, 0, 0)	0.5	0.1	0.8	7	11	(0.5029, -1.7, 1.458)
Ex 4.6	(0, 0, 0)	0.5	0.1	0.8	4	5	(-0.09533, 0.09533, 0.07811)
Ex 4.7	(0, 0, 0)	0.5	0.1	0.8	4	5	(0.5265, 0.5079, 0)
	$p = 2$, stop criterion: $1e-006$, $c = 0.5$, $\tau = 0.1$, $\eta = 0.8$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 0)	0.5	0.1	0.8	4	6	(0.04124, -0.9982)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.1	0.8	3	4	(-0.003477, 8.908e-006,0,0,0,0)
Ex 4.3	(0, 0)	0.5	0.1	0.8	3	4	(-0.003477, 8.908e-006)
Ex 4.4	(0, 0, 0, 0, 0)	0.5	0.1	0.8	4	6	(1.118,2.041, -1.386e-017,0.05605,0.04704)
Ex 4.5	(0, 0, 0)	0.5	0.1	0.8	7	11	(0.5029, -1.7, 1.458)
Ex 4.6	(0, 0, 0)	0.5	0.1	0.8	4	5	(-0.09533, 0.09533, 0.0658)
Ex 4.7	(0, 0, 0)	0.5	0.1	0.8	4	5	(0.5265, 0.5079, 0)
	$p = 3$, stop criterion: $1e-006$, $c = 0.5$, $\tau = 0.1$, $\eta = 0.8$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 0)	0.5	0.1	0.8	4	6	(0.02275, -0.9987)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.1	0.8	3	4	(-0.00251, 7.919e-011,0, 0, 0, 0)
Ex 4.3	(0, 0)	0.5	0.1	0.8	3	4	(-0.00251, 7.919e-011)
Ex 4.4	(0, 0, 0, 0, 0)	0.5	0.1	0.8	4	6	(1.118, 2.074, 0, 0.00922, 0.006294)
Ex 4.5	(0, 0, 0)	0.5	0.1	0.8	7	11	(0.5029, -1.7, 1.458)
Ex 4.6	(0, 0, 0)	0.5	0.1	0.8	4	5	(-0.09533, 0.09533, 0.05056)
Ex 4.7	(0, 0, 0)	0.5	0.1	0.8	4	5	(0.5265, 0.5079, 0)
	$p = 8$, stop criterion: $1e-006$, $c = 0.5$, $\tau = 0.1$, $\eta = 0.8$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 0)	0.5	0.1	0.8	4	6	(0.007717, -0.999)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.1	0.8	3	4	(-0.00102, 6.997e-036,0,0,0,0)
Ex 4.3	(0, 0)	0.5	0.1	0.8	3	4	(-0.00102, 6.997e-036)
Ex 4.4	(0, 0, 0, 0, 0)	0.5	0.1	0.8	4	6	(1.118,2.08,1.658e-019,1.452e-006,-0.0008675)
Ex 4.5	(0, 0, 0)	0.5	0.1	0.8	7	11	(0.5029, -1.7, 1.458)
Ex 4.6	(0, 0, 0)	0.5	0.1	0.8	4	5	(-0.09533, 0.09533, 0.016)
Ex 4.7	(0, 0, 0)	0.5	0.1	0.8	4	5	(0.5265, 0.5079, 0)

Note: Based on $\widehat{H}(\mu, x) = 0$ given as in (4.3), x^0 c, τ and η are fixed.

Table 10: Numerical performance when $p = 2$ for modified problems.

	$p = 2$, stop criterion: $1e - 006$, $c = 1$, $\tau = 0.1$, $\eta = 0.8$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 0)	1	0.1	0.8	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	1	0.1	0.8	8	13	(-0.02205,1.553, 3.071,1.414,1.761,1.763)
Ex 4.3'	(0, 0)	1	0.1	0.8	6	7	(-0.001481, 1.27)
Ex 4.4'	(0, 0, 0, 0, 0)	1	0.1	0.8	Fail	Fail	Fail
Ex 4.5'	(0, 0, 0)	1	0.1	0.8	9	13	(-0.8356, -0.8606, 1.957)
Ex 4.6'	(0, 0, 0)	1	0.1	0.8	5	6	(-0.09533, 0.09533, 0.2995)
Ex 4.7'	(0, 0, 0)	1	0.1	0.8	20	25	(0.5265, 0.5079, 100)
	$p = 2$, stop criterion: $1e - 006$, $c = 0.5$, $\tau = 0.1$, $\eta = 0$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1'	(0, 0)	0.5	0.1	0	Fail	Fail	Fail
Ex 4.2'	(0, 0, 0, 0, 0, 0)	0.5	0.1	0	8	13	(-0.0191, 1.558,3.071,1.414, 1.763,1.765)
Ex 4.3'	(0, 0)	0.5	0.1	0	5	6	(-0.003179, 0.7516)
Ex 4.4'	(0, 0, 0, 0, 0)	0.5	0.1	0	Fail	Fail	Fail
Ex 4.5'	(0, 0, 0)	0.5	0.1	0	8	13	(-0.8356, -0.8606, 1.957)
Ex 4.6'	(0, 0, 0)	0.5	0.1	0	6	8	(-0.09533, 0.09533, 0.3153)
Ex 4.7'	(0, 0, 0)	0.5	0.1	0	Fail	Fail	Fail

Note: Based on $H(z) = 0$ given as in (4.2), x^0 and τ are fixed.

Table 11: Numerical performance when $p = 2$ for original problems.

	$p = 2$, stop criterion: $1e - 006$, $c = 1$, $\tau = 0.1$, $\eta = 0.8$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 0)	1	0.1	0.8	3	4	(0.01673, -0.9989)
Ex 4.2	(0, 0, 0, 0, 0, 0)	1	0.1	0.8	3	4	(-0.003682, 4.453e-006,0,0,0,0)
Ex 4.3	(0, 0)	1	0.1	0.8	3	4	(-0.003682, 4.453e-006)
Ex 4.4	(0, 0, 0, 0, 0)	1	0.1	0.8	5	6	(1.118, 1.414, 0, 0.8795, 0.3904)
Ex 4.5	(0, 0, 0)	1	0.1	0.8	Fail	Fail	Fail
Ex 4.6	(0, 0, 0)	1	0.1	0.8	4	5	(-0.09533, 0.09533, 0.06535)
Ex 4.7	(0, 0, 0)	1	0.1	0.8	4	5	(0.5265, 0.5079, 0)
	$p = 2$, stop criterion: $1e - 006$, $c = 0.5$, $\tau = 0.1$, $\eta = 0$.						
Problem	x^0	c	τ	η	NI	NF	SOL
Ex 4.1	(0, 0)	0.5	0.1	0	4	6	(0.04124, -0.9982)
Ex 4.2	(0, 0, 0, 0, 0, 0)	0.5	0.1	0	3	4	(-0.003477, 8.908e-006,0,0,0,0)
Ex 4.3	(0, 0)	0.5	0.1	0	3	4	(-0.003477, 8.908e-006)
Ex 4.4	(0, 0, 0, 0, 0)	0.5	0.1	0	4	6	(1.118,2.041, -1.386e-017,0.05605,0.04704)
Ex 4.5	(0, 0, 0)	0.5	0.1	0	8	15	(0.5029, -1.7, 1.458)
Ex 4.6	(0, 0, 0)	0.5	0.1	0	4	5	(-0.09533, 0.09533, 0.0658)
Ex 4.7	(0, 0, 0)	0.5	0.1	0	4	5	(0.5265, 0.5079, 0)

Note: Based on $\widehat{H}(\mu, x) = 0$ given as in (4.3), x^0 and τ are fixed.