Numerical algorithms for a class of obstacle problems

Yuwei Zhao

October 14, 2009

Abstract

This paper concerns the nonsmooth Newton method and its applications in solving obstacle problems. A detailed description of the nonsmooth Newton method and its corresponding convergence theorem is presented. We show that Howard's algorithm generates the same sequence as the nonsmooth Newton method does in solving obstacle problems. We also consider a nonsmooth Newton method based on the so-called Fischer-Burmeister (FB) function. The penalty approximations are used to solve obstacle problems with the nonsmooth Newton method. Obstacle problems that arise in financial mathematics background, such as pricing American options, pricing American warrants and portfolio selcetion problems, are taken as examples to examine the algorithms and numerical tests are performed.

1 Introduction

The general form of the obstacle problem is formulated as follows:

$$\text{Find a } x \text{ such that } \min(F(x), G(x)) = 0.$$

The difficulty of solving the obstacle problem is the nonsmoothness of the min function. A direct consequence of the nonsmoothness is the absence of the Jacobian at the point x where $F(x) = G(x)$, even if the Jacobian exists, it is still questionable that the Jacobian gives a "good" approximation in the Newton iterations. To circumvent this difficulty, two methods are available. One method is to generalize the concept of the Jacobian and to use the generalized Jacobian in the Newton iterations. The other method is to find a zero x of a smooth function which is also a zero of the complementarity problem. The FB-based nonsmooth Newton method, where FB means the Fischer-Burmeister function, is of this kind. Both methods are proposed and well explained in Facchinei and Pang [11]. Pang and Qi [19] also summarizes the algorithms and motivations of various nonsmooth equations.

It is very difficult to find explicit solutions to the obstacle problems. In contrast, several numerical methods are available. One natural technique is to use the dynamic programming method and its variations, such as the Howard's algorithm, since the problem can be treated as optimal stopping problems (Bellman [3], Bertsekas [4], Howard [15]). Bokanowski et al. [5] studied convergence results of Howard's algorithm for the solution of the obstacle problem $\min_{\alpha\in\mathcal{A}}(B^\alpha x - b^\alpha) = 0$, where B^α is a matrix and b^α is a vector. A globally Q-superlinear convergence result is shown under the monotonicity assumption of B^α.

Another method is to use the binomial model to simulate the stochastic process embedded in the formulation of the problem and to find an optimal strategy based on the simulation result. Davis et al. [10] study the problem of pricing European options with proportional transaction costs with the binomial model. Dai and Yi [7] use the binomial model to study the portfolio selection problem with transaction costs. Despite of its simplicity, it is of little interest theoretically. The disadvantages are that this method undermines the continuity property of the problem and as the number of time steps increases, the number of nodes to calculate rises exponentially. Moreover, it becomes too complex to apply this method in higher dimensions if two or more related stochastic processes are dealt with at the same time. So this method is not considered in this paper.

Although the penalty approximation is still a nonsmooth equation, it might cure the singularity of the generalized Jacobian in certain cases. Moreover, it is easy to apply this method when there are more than two functions inside the min function. Zvan et al. [21] use the penalty method to value American options with stochastic volatility. Forsyth and Vetzal [12] provide a penalty method for valuing American options. By adopting variant timesteps, quadratic convergence is achieved with respect to the number of the grids. Dai et al. [8] propose two schemes which are modified from the scheme in Forsyth and Vetzal [12]. Both schemes achieve a second order convergence even when constant timesteps are adopted. Dai and Zhong [6], Dai and Yi [7] use the penalty method to solve the portfolio selection problem with transaction costs. The corresponding complementarity problem has three functions (in one-dimensional case) or five functions (in two-dimensional case) inside the min function.

In American-type financial contracts, the holder can exercise the contract any time before the mature time. Consequently, finding an optimal stopping strategy is of importance to value the American-type financial contract. Similarly, the portfolio selection problem, which is to maximize the portfolio value by choosing the best proportion taken by stocks in the portfolio, is also important to find the optimal trading strategy. Both the valuing American-type financial contracts and the portfolio selection problem can be formulated as obstacle problems.

The rest of this paper is organized as follows. In Section 2, the derivation of the generalized Jacobian and the description of the nonsmooth Newton method are presented. A series of associated propositions and theorems are given as well. In Section 3, we focus on valuing American options with different numerical methods. In Section 4, valuing the American warrant subject to issuer's calling is studied. In Section 5, we study a portfolio selection problem with transaction costs.

2 Nonsmooth Newton method[1]

Newton method for smooth functions is a very powerful technique to find a zero of the smooth function, for which the convergence rate is quadratic. This is the reason why it is very attractive to develop an algorithm for nonsmooth functions based on Newton method. Let $G : \Omega \subset \mathbb{R}^n \to \mathbb{R}^n$ be a continuously differentiable function. The key idea in the classical Newton method is to replace the function G by an approximation depending on the current iterate, resulting in an approximation problem that can be solved more easily. The solution of this approximation problem is then taken as a new iterate and the process is

[1] All the definitions, propositions and theorems in this section are cited from Facchinei and Pang [11]

repeated. For the function G, a natural approximation is available, due to Taylor's expansion. Suppose that an iterate x^k is given, the approximation problem is

$$G(x^k) + JG(x^k)(x - x^k) = 0$$

and calculate x^{k+1} as the zero of this linear approximation. If G is a nonsmooth function, the Jacobian might not exist or the above approximation might not be "good". We want to define the kind of the "good" linear approximations, which satisfy the desirable features in the Newton iteration.

Definition 2.1. Let G be a locally Lipschitz function from an open subset Ω of $\mathbb{R}^n$ to $\mathbb{R}^m$. We say that G has a Newton approximation at a point $\bar{x} \in \Omega$ if there exists a neighborhood $\Omega' \subseteq \Omega$ and a function $\Delta : (0, \infty) \to [0, \infty)$ with

$$\lim_{t \downarrow 0} \Delta(t) = 0, \tag{2.1}$$

such that for every point x in Ω' there is a family $\mathcal{A}(x)$ of functions each mapping $\mathbb{R}^n$ to $\mathbb{R}^m$ and satisfying the following two properties:

(a) $A(x, 0) = 0$ for every $A(x, \cdot) \in \mathcal{A}(x)$;

(b) for any $x \in \Omega'$ different from $\bar{x}$ and for any $A(x, \cdot) \in \mathcal{A}(x)$

$$\frac{\|G(x) + A(x, \bar{x} - x) - G(\bar{x})\|}{\|x - \bar{x}\|} \leq \Delta(\|x - \bar{x}\|). \tag{2.2}$$

We call $\mathcal{A}$ a (Newton) approximation scheme for G at $\bar{x}$. If the requirement (b) is strengthened to

(b') there exists a positive constant L' such that for each $x \in \Omega'$ different from $\bar{x}$ and for every $A(x, \cdot) \in \mathcal{A}(x)$,

$$\frac{\|G(x) + A(x, \bar{x} - x) - G(\bar{x})\|}{\|x - \bar{x}\|^2} \leq L', \tag{2.3}$$

then we say that F has a strong Newton approximation at $\bar{x}$ and that $\mathcal{A}$ is a strong (Newton) approximation scheme. Furthermore, if the following additional condition is met:

(c) ($m = n$ and) $\mathcal{A}$ is a family of uniformly Lipschitz homeomorphisms on Ω', by which we mean that there exist positive constants $L_{\mathcal{A}}$ and $\epsilon_{\mathcal{A}}$ such that for each x in Ω' and for each $A(x, \cdot) \in \mathcal{A}(x)$, there are two open sets U_x and V_x, both containing $\mathbb{B}(0, \epsilon_{\mathcal{A}})$, such that $A(x, \cdot)$ is a Lipschitz homeomorphism mapping U_x onto V_x with $L_{\mathcal{A}}$ being the Lipschitz modulus of the inverse of the restricted map $A(x, \cdot)\,|_{U_x}$,

we say that the (strong) Newton approximation is nonsingular and that the (strong) approximation $\mathcal{A}$ is nonsingular. If $\mathcal{A}$ contains only one element for every x in Ω', we say that G admits a single-valued (strong) Newton approximation at $\bar{x}$ and the approximation scheme $\mathcal{A}$ is single valued. □

It is easy to see that Condition (b) and Condition (b') are playing the same role as Taylor's expansion does for smooth functions. Now the primal question becomes to find a suitable approximation for nonsmooth functions. We start from the complementarity problems we intend to solve.

2.1 Complementarity problems

An obstacle problem is also called a complementarity problem. "Complementarity" explains the relationship between two terms inside the min function. A basic form of a complementarity problem is that $\min(F(x), x) = 0$, where $x \in K \subset \mathbb{R}^n$ and $F : K \to \mathbb{R}^n$. We can replace x inside the min function by any function $G : K \subset \mathbb{R}^n \to \mathbb{R}^n$ which is invertible locally, and then the problem $\min(F(x), G(x)) = 0$ is equivalent to the problem $\min((G^{-1} \circ F)(x), x) = 0$. For the purpose of convenience, the basic form of complementarity problems is used only in Section 2.1 and the complementarity problems in the next sections will not be transformed into the basic form. In the following definition, an important class of complementarity problems is given.

Definition 2.2. Given a mapping $F : \mathbb{R}^n_+ \to \mathbb{R}^n$, the NCP(F) is to find a vector $x \in \mathbb{R}^n$ satisfying

$$0 \le x \perp F(x) \ge 0. \quad \square$$

By expressing the orthogonality condition $x^T F(x) = 0$ in terms of the componentwise products, we obtain the following equivalent formulation of the NCP(F):

$$0 \le x, F(x) \ge 0$$
$$x_i F_i(x) = 0, \forall i = 1, \ldots, n$$

This formulation provides an explanation for term "complementarity"; namely, x_i and $F_i(x)$ are complementary in the sense that if one of them is positive, then the other must be zero. If F is an affine function, We call such NCP(F) an linear complementarity problem (LCP). Using the min function is not the only way to define a complementarity problem. A class of functions, named C-functions can capture the essence of all unconstrained equation formulations of complementarity problems. We define them explicitly below.

Definition 2.3. A function $\psi : \mathbb{R}^2 \to \mathbb{R}$ is called a C-function, where C stands for complementarity, if for any pair $(a, b) \in \mathbb{R}^2$,

$$\psi(a, b) = 0 \Leftrightarrow [(a, b) \ge 0 \text{ and } ab = 0];$$

equivalently, ψ is a C-function if the set of its zeros are the two nonnegative semi-axes. $\square$

In general, given any C-function ψ, we can immediately obtain an equivalent formulation of the NCP(F) as a system of equations:

$$0 = F_\psi(x) \equiv \begin{pmatrix} \psi(x_1, F_1(x)) \\ \vdots \\ \psi(x_n, F_n(x)) \end{pmatrix}$$

Let ψ be the min function. Applying this equivalence, we obtain

$$x \text{ solves the NCP(F)} \Leftrightarrow \mathbf{F}_{\min} \equiv \min(x, F(x)) = 0,$$

where "min" is the componentwise minimum vector function.

Another important C-function is denoted by ψ_{FB} and referred to as the Fischer-Burmeister C-function:

$$\psi_{FB}(a, b) \equiv \sqrt{a^2 + b^2} - a - b, \quad (a, b) \in \mathbb{R}^2.$$

This function is convex, differentiable everywhere except the origin $(0, 0)$; moreover $\psi^2_{FB}(a, b)$ is a continuously differentiable function on the entire plane. Based on this property, we can develop an efficient algorithm for the solution of the NCP(F), which will be presented in next section.

2.2 Monotonicity

Monotonicity plays a very important role in our following examples. It ensures a global convergence of the sequence generated by the nonsmooth Newton method. Moreover, when we try to solve the LCP, the monotonicity implies the global non-singularity of the Newton approximation. To begin with, we define the monotonicity for an arbitrary function F.

Definition 2.4. A mapping $F : K \subseteq \mathbb{R}^n \to \mathbb{R}^n$ is said to be

(a) monotone on K if

$$(F(x) - F(y))^T(x - y) \geq 0, \quad \forall x, y \in K;$$

(b) strictly monotone on K if

$$(F(x) - F(y))^T(x - y) > 0, \quad \forall x, y \in K.$$

□

If an affine map $F(x) = Ax + b$ with $K = \mathbb{R}^n$, where A is a $n \times n$ matrix, not necessarily symmetric, and b is an n-vector, we have

$$(y - x)^T A^T (y - x) \geq (>)0, \text{ if } F \text{ is (strongly) monotone for all } (x, y) \in \mathbb{R}^n.$$

Consequently, we have the following results:

(a) F is strictly monotone if and only if A is positive definite;

(b) F is monotone if and only if A is positive semi-definite.

In most cases, the matrices in obstacle problems are not symmetric or positive definite. We defines a broader class of matrices, M-matrices(see Appendix A). The matrix A has an porperty: for a vector x,

$$Ax \geq 0 \text{ implies } x \geq 0.$$

If A is also invertible, we call A a monotone matrix. It is easily to show that the positive (semi-)definite matrices are M-matrices. In the following sections, we can see that the sequence generated by Howard's algorithm is monotone if the matrix calculated in each iterate is monotone.

2.3 Generalized Jacobians

In this section, we intend to propose a generalized Jacobian for locally continuous functions which are also directionally differentiable. We introduce a very important definition of differentiability.

Definition 2.5. A function $\Phi : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}^m$ defined on the open set $\mathcal{D}$ is said to be B(ouligand)-differentiable at a vector $x \in \mathcal{D}$ if Φ is Lipschitz continuous in a neighborhood of x and directionally differentiable at x. If Φ is B-differentiable at x, we call the directional derivative $\Phi'(x; d)$ the B-derivative of Φ at x along d. The B-derivative $\Phi(x; \cdot)$ is strong if the error function

$$e(y) \equiv \Phi(y) - \Phi(x) - \Phi'(x; y - x)$$

satisfies

$$\lim_{y^1 \neq y^2, (y^1, y^2) \to (x,x)} \frac{e(y^1) - e(y^2)}{\|y^1 - y^2\|} = 0$$

In this case, we say that Φ is strongly B-differentiable at x. We say that Φ is B-differentiable near x if Φ is B-differentiable at every point in a certain neighborhood of x. □

An important consequence of the B-differentiability is that the following statement on the limit of the directional derivative holds for any $x \in \mathcal{D}$:

$$\lim_{y \neq x, y \to x} \frac{\Phi(y) - \Phi(x) - \Phi'(x; y - x)}{\|y - x\|} = 0.$$

It is easy to verify that the C-functions we mentioned, e.g. the min and FB functions, are all B-differentiable. The next proposition is the chain rule for the B-differentiable functions. Furthermore, under an appropriate restriction, the B-derivative of the composite map is strong.

Propositon 2.1. *Let $\mathcal{D}$ and $\mathcal{D}'$ be open sets in $\mathbb{R}^n$ and $\mathbb{R}^m$ respectively. Let $\Phi : \mathcal{D} \to \mathbb{R}^m$ and $\Psi : \mathcal{D}' \to \mathbb{R}^p$ be B-differentiable at $x \in \mathcal{D}$ and $\Phi(x) \in \mathcal{D}'$ respectively. Suppose that $\Phi(\mathcal{D}) \subseteq \mathcal{D}'$. The following two statements hold.*

(a) *The composite map $\Gamma \equiv \Psi \circ \Phi : \mathcal{D} \to \mathbb{R}^p$ is B-differentiable at x; moreover,*

$$\Gamma'(x; d) = \Psi'(\Phi(x); \Phi'(x; d)), \quad \forall d \in \mathbb{R}^n.$$

(b) *If Ψ is strongly F(réchet)-differentiable at $\Phi(x)$ and Φ has a strong B-derivative at x, then Γ has a strong B-derivative at x.* □

Consider the problem $\min(F(x), H(x)) = 0$, where two functions F and H are C^1 functions which map $\mathbb{R}^n$ into $\mathbb{R}^m$. Let $G(x) = \min(F(x), H(x))$. We can take G as a composition of 2^m C^1 function G^i obtained by letting $G^i_J \equiv F_J$ and $G^i_{\bar{J}} \equiv H_{\bar{J}}$, where J and $\bar{J}$ are any pair of complementary subsets of $1, \ldots, m$. We call the function G a PC^1 function. The definition of the PC^1 function is given below.

Definition 2.6. A continuous function $F : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}^m$ is said to be a PC^1 function near the vector $x \in \mathcal{D}$ if there exist an open neighborhood $\mathcal{N} \subseteq \mathcal{D}$ of x and a finite family of C^1 functions defined on $\mathcal{N}$, $\{G^1, G^2, \ldots, G^k\}$, for some positive integer k, such that $G(y)$ is an element of $\{G^1(y), G^2(y), \ldots, G^k(y)\}$ for all $y \in \mathcal{N}$. Each function G^i is called a C^1 piece of G at x. Let $\mathcal{P}(y)$ denote the set of indices $i \in \{1, \ldots, k\}$ such that $G(y) = G^i(y)$. □

The PC^1 function is of importance, because it provides a bridge between nonsmooth functions and smooth functions. For every $x \in \mathcal{N}$, there exists an index $i \in \{1, \ldots, k\}$ such that $G(x) = G^i(x)$. This is to say that if x is a zero of G, then x is also a zero of some piece G^i where $i \in \{1, \ldots, k\}$. There are only a finite number of candidates for the zero of a PC^1 function G. This result will be vital to the proof of the converge result of the algorithms. This next proposition gives that PC^1 functions are locally B-differentiable.

Propositon 2.2. *Let G be a PC^1 map near x with C^1 pieces $\{G^1, G^2, \ldots, G^k\}$. The following statements are valid.*

(a) *G is B-differentiable at all points near x.*

(b) *$G'(x;\cdot)$ is piecewise linear with linear pieces $\{JG^1(x), JG^2(x), \ldots, JG^k(x)\}$.* □

We know that if a function G is continuously differentiable in a neighborhood of x, then G is a locally Lipschitz homeomorphism at x if and only if the Jacobian matrix $JG(x)$ is invertible. It turns out that necessary and sufficient conditions for a PC^1 function to be a Lipschitz homeomorphism near a point x can be obtained in terms of an object that generalizes the Jacobian matrix of a smooth function. The cornerstone of this object is Rademacher's theorem, named after Hans Adolph Rademacher, which is given below.

Theorem 2.3. *Let $F : \mathcal{D} \subseteq \mathbb{R}^m \to \mathbb{R}^n$ be a locally Lipschitz continuous function defined on the open set $\mathcal{D}$ in $\mathbb{R}^n$. Let $\mathcal{D}'$ be the subset of $\mathcal{D}$ consisting of points where F is Fréchet differentiable. The set $\mathcal{D} \setminus \mathcal{D}'$ is negligible; thus $\mathcal{D}'$ is dense in $\mathcal{D}$; i.e., $\mathcal{D} \subseteq cl\ \mathcal{D}'$, where $cl\ \mathcal{D}'$ means the closure of the set $\mathcal{D}'$.* □

Rademacher's theorem implies that every locally Lipschitz continuous function is F-differentiable almost everywhere. Then a generalized Jacobian is introduced below.

Definition 2.7. Let $G : \mathcal{U} \to \mathbb{R}^n$, where $\mathcal{U}$ is an open subset of $\mathbb{R}^n$, be a given function that is locally Lipschitz continuous in a neighborhood $\mathcal{N} \subseteq \mathcal{U}$ of a vector x. Define the limiting Jacobian $Jac(G, x)$ to be the (nonempty) set of limit points of sequences $\{JG(x^v)\}$, where each $x^v \in \mathcal{U}$ is a F-differentiable point of G and the sequence $\{x^v\}$ converges to x. Another term for $Jac(G, x)$ is the B-subdifferential of G at x, denoted $\partial_B G(x)$. □

Although the concept of the limiting Jacobian of G is well defined and employed successfully when G is a PC^1 function, there are a few facts that we can not deny: for an arbitrary nonsmooth function, it is difficult to calculate and manipulate the limiting Jacobian; secondly, the Jacobian does not allow us to obtain optimality conditions in a nonsmooth optimization problem; lastly, the limiting Jacobian can not be based solely on to obtain mean-value theorems or other useful results. The following definition intend to circumvent these difficulties.

Definition 2.8. Let $G : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}^m$, with $\mathcal{D}$ open, be locally Lipschitz at a vector $\bar{x} \in \mathcal{D}$. The Clarke generalized Jacobian of G at $\bar{x}$ is:

$$\partial G(\bar{x}) = conv Jac\, G(\bar{x})$$

where conv A means the convex hull of the set A. When $m = 1$, that is when G is a real-valued function $g : \mathbb{R}^n \to \mathbb{R}$, $\partial g(\bar{x})$ is called the generalized gradient of g at $\bar{x}$. Furthermore, in this case, consistently with the notation of the gradient of a smooth function, the elements of $\partial g(\bar{x})$ are viewed as column vectors. □

The generalized Jacobian can be viewed as a multifunction from $\mathcal{D}$ into subsets of $\mathbb{R}^{n\times m}$:

$$\partial G : x \in \mathcal{D} \mapsto \partial G(x) \subset \mathbb{R}^{n\times m}.$$

Below we report some basic properties of this multifunctions at a vector x when G is locally Lipschitz.

Propositon 2.4. *Let a function $G : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}^m$ be given, with G locally Lipschitz on the open set $\mathcal{D}$. The following statements are valid for any $x \in \mathcal{D}$:*

(a) *$\partial G(x)$ is nonempty, convex and compact;*

(b) *the mapping ∂G is upper semicontinuous at x; thus for every $\varepsilon > 0$ there is a $\delta > 0$ such that, for all $y \in \mathbb{B}(x, \delta)$,*

$$\partial G(y) \subseteq \partial G(x) + \mathbb{B}(0, \varepsilon).$$

Therefore ∂G is closed at x, that is, if $\{x^k\} \to x$, $H^k \in \partial G(x^k)$ and $H^k \to H$, then $H \in \partial G(x)$. □

We propose an proposition on the composition map next.

Propositon 2.5. *Let $f = g \circ G$, where $G : \mathbb{R}^n \to \mathbb{R}^m$ is locally Lipschitz continuous at x and where $g : \mathbb{R}^m \to \mathbb{R}^n$ is locally Lipschitz at $G(x)$. Then f is locally Lipschitz continuous at x and*

$$\partial f(x) \subseteq conv\{\xi = H^T\zeta : H \in \partial G(x), \zeta \in \partial g(G(x))\}.$$

If in addition either one of the following two conditions is satisfied, then equality holds and the conv is superfluous.

(a) *g is continuously differentiable at $G(x)$;*

(b) *g is C-regular at $G(x)$ and G is continuously differentiable at x.* □

In the next propositions, an optimal condition on generalized gradient and the mean-value theorem is reported.

Propositon 2.6. *Suppose that $g : \mathbb{R}^n \to \mathbb{R}$ is Lipschitz continuous in a neighborhood of an (unconstrained) local minimum x of g. Then*

$$0 \in \partial g(x) \tag{2.4}$$

If g is convex, the condition (2.4) *is necessary and sufficient for x to be a global (unconstrained) minimum point of g.* □

We call a vector x satisfying (2.4) a C-stationary point of g, where C stands for Clarke.

Propositon 2.7. *Let a function $g : \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz on an open set containing the line segment $[x, y]$. There exists a point z in (x, y) such that*

$$g(y) = g(x) + \xi^T(y - x)$$

for some ξ belonging to $\partial g(z)$. □

The above results is confined to real-valued functions. We present a result that abridge between the generalized Jacobian of a vector-valued function and the generalized gradients of its component functions.

Propositon 2.8. *Let $G : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}^m$ be a locally Lipschitz continuous function on the open set $\mathcal{D}$. If $x \in \mathcal{D}$, then*

$$\partial G(x) \subseteq (\partial G_1(x) \times \partial G_2(x) \times \cdots \times \partial G_m(x))^T.$$

We apply Proposition 2.8 to Proposition 2.7, and we get an integral version of the mean-value theorem in the nonsmooth case.

Propositon 2.9. *Let a function $G : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}^m$, with $\mathcal{D}$ open, be B-differentiable at a point x in $\mathcal{D}$. For every vector $d \in \mathbb{R}^n$, there exists $H \in \partial G(x)$ such that $G'(x; d) = Hd$.* □

We say that the generalized Jacobian $\partial G(x)$ is nonsingular if all matrices in this Jacobian are nonsingular.

Propositon 2.10. *Let $G : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}^n$ be locally Lipschitz at $x \in \mathcal{D}$. If the generalized Jacobian $\partial G(x)$ is nonsingular, then G is a locally Lipschitz homeomorphism at x.*

2.4 Semismooth functions

Although the generalized Jacobian allows us to extend the useful results for smooth functions to locally continuous functions, it fails to define a Newton approximation scheme of an arbitrary locally Lipschitz continuous function G; that is the limit condition

$$\lim_{\bar{x}\neq x\to\bar{x}, H\in\partial G(x)} \frac{G(x)+H(\bar{x}-x)-G(\bar{x})}{\|x-\bar{x}\|}=0$$

does not generally hold for any locally Lipschitz continuous function G. We need locally Lipschitz continuous functions with the legitimate Newton approximation scheme defined by the (Clarke) generalized Jacobians. They are called semismooth functions and the definition is below.

Definition 2.9. Let $G:\mathcal{D}\subseteq\mathbb{R}^n\to\mathbb{R}^m$, with $\mathcal{D}$ open, be a locally Lipschitz continuous function on $\mathcal{D}$. We say that G is semismooth at a point $\bar{x}\in\mathcal{D}$ if G is directionally differentiable near $\bar{x}$ and there exists a neighborhood $\mathcal{D}'\subseteq\mathcal{D}$ of $\bar{x}$ and a function $\Delta:(0,\infty)\to[0,\infty)$ with

$$\lim_{t\downarrow 0}\Delta(t)=0$$

such that for any $x\in\mathcal{D}'$ different from $\bar{x}$,

$$\frac{\|G'(x;x-\bar{x})-G'(\bar{x};x-\bar{x})\|}{\|x-\bar{x}\|}\leq\Delta(\|x-\bar{x}\|).$$

If the above requirement is strengthened to

$$\limsup_{\bar{x}\neq x\to\bar{x}}\frac{\|G'(x;x-\bar{x})-G'(\bar{x};x-\bar{x})\|}{\|x-\bar{x}\|^2}<\infty,$$

we say that G is strongly semismooth at $\bar{x}$. If G is (strongly) semismooth at each point of Ω, then we say that G is (strongly) semismooth on Ω. □

By definition, G is semismooth at $\bar{x}$ if and only if G is B-differentiable near $\bar{x}$ and the following limit holds:

$$\limsup_{\bar{x}\neq x\to\bar{x}}\frac{\|G'(x;x-\bar{x})-G'(\bar{x};x-\bar{x})\|}{\|x-\bar{x}\|}=0$$

Thus, for all x close enough to $\bar{x}$, a good approximation to the function G is provided by the directional derivative $G'(\bar{x};x-\bar{x})$. Moreover, the directional derivative $G'(\bar{x};x-\bar{x})$ can be approximated with a good degree of precision by using any element of the generalized Jacobian of G at x. Taking this assertion and other results as basis, we can use the generalized Jacobians to build linear Newton approximations to G at $\bar{x}$. The theorem formally presents this statement.

Theorem 2.11. *Let $G:\mathcal{D}\subseteq\mathbb{R}^n\to\mathbb{R}^m$, with $\mathcal{D}$ open, be B-differentiable near $\bar{x}\in\mathcal{D}$. The following three statements are equivalent:*

(a) *G is semismooth at $\bar{x}$*

(b) *the following limit holds:*

$$\lim_{\bar{x}\neq x\to\bar{x}}\frac{G'(x;x-\bar{x})-H(x-\bar{x})}{\|x-\bar{x}\|}=0; \tag{2.5}$$

(c) *the following limit holds:*

$$\lim_{\bar{x}\neq x\to\bar{x}, H\in\partial G(x)} \frac{G(x)+H(\bar{x}-x)-G(\bar{x})}{\|x-\bar{x}\|}=0; \tag{2.6}$$

If G is strongly semismooth at $\bar{x}$, then

$$\limsup_{\bar{x}\neq x\to\bar{x}} \frac{G(x)-G(\bar{x})-G'(\bar{x};x-\bar{x})}{\|x-\bar{x}\|^2}<\infty, \tag{2.7}$$

$$\limsup_{\bar{x}\neq x\to\bar{x}, H\in\partial G(x)} \frac{G(x)+H(\bar{x}-x)-G(\bar{x})}{\|x-\bar{x}\|^2}<\infty. \tag{2.8}$$

□

Consequently when $m = n$, by (2.5), we obtain that $\mathcal{A}(x) \equiv \partial G(x)$ is a Newton approximation scheme of G at $\bar{x}$ if G is semismooth at $\bar{x}$. Furthermore, if G is strongly semismooth at $\bar{x}$, by (2.8), the approximation scheme is strong. These results justify our interest in the class of semismooth functions.

As we know, the composition of smooth functions preserves the smoothness, which implies in particular that the sum and difference of two smooth functions are also smooth. We extend this result to the semismooth functions in the following proposition.

Propositon 2.12. *Let a function $F : \Omega_F \subseteq \mathbb{R}^n \to \mathbb{R}^m$, with Ω_F open, a point $\bar{x}$ belonging to Ω_F , and a function $g : \Omega_g \subseteq \mathbb{R}^m \to \mathbb{R}$, with Ω_g being a neighborhood of $F(\bar{x})$, be given. If F and g are (strongly) semismooth at $\bar{x}$ and $F(\bar{x})$ respectively, then the composite function $g \circ F$ is (strongly) semismooth at $\bar{x}$.*

As we mentioned, this theorem implies that the sum and difference of (strongly) semismooth functions are (strongly) semismooth. It is easy to check that a vector-valued function is (strongly) semismooth if and only if each of its component functions is (strongly) semismooth. Combining with Proposition 2.12, we obtain that the composition of two vector semismooth functions is semismooth. In the next propositions, a broad class of functions is shown to be semismooth. The functions we deal with in the example can all be viewed as a composition of these functions.We take real-valued functions as the beginning.

Propositon 2.13. *Let $f : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}$, with Ω open, and a point $\bar{x}$ belonging to Ω be given.*

(a) *If f is continuously differentiable in a neighborhood of $\bar{x}$, then f is semismooth at $\bar{x}$.*

(b) *If f is continuously differentiable with a Lipschitz continuous gradient in a neighborhood of $\bar{x}$, then f is strongly semismooth at $\bar{x}$.*

(c) *If f is convex on a neighborhood of $\bar{x}$, then f is semismooth at $\bar{x}$.* □

This proposition implies that PC^1 functions are semismooth. Moreover like PC^1 functions, we say that a continuous function G is piecewise semismooth near a vector x in the domain of G if G satisfies the conditions in Definition 2.6 except the pieces $\{G^i : i = 1, \ldots, k\}$ are semismooth functions near x.

Propositon 2.14. *Let $G : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^m$, with Ω open, be piecewise semismooth near the vector $\bar{x}$ in Ω. Then G is semismooth at $\bar{x}$.* □

The class of piecewise semismooth functions seems to be broader than the class of semismooth functions. Proposition 2.14 shows that the class of semismooth functions includes all the piecewise semismooth functions, and of course, PC^1 functions.

Proposition 2.13 also implies that every norm function, being convex, is semismooth. The next proposition shows that the norm functions of a particular class are strongly semismooth.

Propositon 2.15. *The norm function $\|\cdot\|_p : \mathbb{R}^n \to \mathbb{R}_+$ is strongly semismooth for every $p \in [1, \infty]$.* □

In fact, it is easy to see that for any $\mu \in (0, 4)$, the modified FB function:

$$\psi_{FB_\mu}(a, b) = \sqrt{(a-b)^2 + \mu ab} - a - b, \quad (a, b) \in \mathbb{R}^2$$

is strongly semismooth. The function ψ_{FB_μ} is a C-function and includes both the min function (when $\mu \to 0+$) and the FB function (when $\mu = 2$). Thus the min function and the FB function are strongly semismooth.

2.5 Nonsmooth Newton Method

We first give a basic form of the nonsmooth Newton method. Let us suppose that the function G admits a Newton approximation scheme $\mathcal{A}(x)$ at x where $G(x) = 0$. We start from x^0 sufficiently close to x and use the Newton iteration to find x.

Algorithm 1 Nonsmooth Newton Method[a]

Data: $x^0 \in \mathbb{R}^n$ and $\varepsilon > 0$.

Step 1: Set $k = 0$.

Step 2: If $G(x^k) = 0$, stop.

Step 3: Select an element $A(x^k, \cdot)$ in $\mathcal{A}(x)$ and find a vector d^k in $\mathbb{B}(0, \varepsilon)$ such that

$$G(x^k) + A(x^k, d^k) = 0.$$

Step 4: Set $x^{k+1} \equiv x^k + d^k$ and $k \leftarrow k + 1$; go to Step 2.

[a] Facchinei and Pang [11]

Basically the Newton method is an iterative algorithm. Several concepts of converge rates play an important role in the convergence analysis of iterative algorithms.

Definition 2.10. Let $\{x^k\} \subset \mathbb{R}^n$ be a sequence of vectors tending to the limit $x^\infty \neq x^k$ for all k. The convergence rate is said to be (at least)

(a) Q-linear if

$$\limsup_{k \to \infty} \frac{\|x^{k+1} - x^\infty\|}{\|x^k - x^\infty\|} < \infty$$

(b) Q-superlinear if

$$\lim_{k\to\infty} \frac{\|x^{k+1} - x^\infty\|}{\|x^k - x^\infty\|} = 0$$

(c) Q-quadratic if

$$\limsup_{k\to\infty} \frac{\|x^{k+1} - x^\infty\|}{\|x^k - x^\infty\|^2} < \infty$$

In each case, we say that $\{x^k\}$ converges to x^∞ (at least) Q-linearly, Q-superlinearly, and Q-quadratically respectively. □

The theorem below shows that the nonsmooth Newton method pertain the fast convergence rate of the classical Newton method. Recall that $\varepsilon_{\mathcal{A}}$ is a positive constant in Definition 2.1.

Theorem 2.16. *Let $G : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^n$, with Ω open, be a locally Lipschitz function in a neighborhood of $x^* \in \Omega$ satisfying $G(x^*) = 0$. Assume that G admits a nonsingular Newton approximation $\mathcal{A}$ at x^*. For every $\varepsilon \in (0, \varepsilon_{\mathcal{A}}]$, there exists a neighborhood $\mathbb{B}(x^*, \delta)$ of x^* such that if x^0 belongs to $\mathbb{B}(x^*, \delta)$ then Algorithm 1 generates a unique sequence $\{x^k\}$ that converges Q-superlinearly to x^*. If the Newton approximation $\mathcal{A}$ is strong, the convergence rate is Q-quadratic.*

This theorem is a local convergence result; that is, it is assumed that the initial iterate x^0 is chosen from a suitable neighborhood of a desired but unknown solution. If the sequence $\{x^k\}$ generated by Algorithm 1 is monotone and bounded, then this theorem is strengthened to be a global convergence result.

Now we modify Algorithm 1 for semismooth functions particularly.

Algorithm 2 Semismooth Newton Method[a]

Data: $x^0 \in \mathbb{R}^n$.

Step 1: Set $k = 0$.

Step 2: If $G(x^k) = 0$, stop.

Step 3: Select an element $H^k \in \partial G(x^k)$.Find a direction $d^k \in \mathbb{R}^n$ such that

$$G(x^k) + H^k d^k = 0.$$

Step 4: Set $x^{k+1} \equiv x^k + d^k$ and $k \leftarrow k + 1$; go to Step 2.

[a] Facchinei and Pang [11]

The condition that H^k is nonsingular is very vital to Algorithm 2, because Algorithm 2 generates a unique sequence by solving a system of linear equations with a unique solution. The lemma below essentially guarantee the nonsingularity of H^k if all generalized Jacobians of G at a zero x^* are nonsingular and that x^k is sufficiently close to x^*.

Lemma 2.17. *Let $J : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^{n\times n}$, with Ω open, be a compact-valued, upper semicontinuous set-valued mapping. Suppose that at a point $\bar{x} \in \Omega$ all the matrices in $J(\bar{x})$ are nonsingular. There exist*

positive constants κ and δ such that

$$\sup_{x\in\mathbb{B}(\bar{x},\delta),H\in J(x)} \max\{\|H\|, \|H^{-1}\|\} \leq \kappa.$$

In particular, all the matrices $H \in J(x)$ for $x \in \mathbb{B}(\bar{x}, \delta)$ are nonsingular.

Based on the above lemma, we state the convergence result of Algorithm 2.

Theorem 2.18. *Let $G : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^n$, with Ω open, be semismooth at $x^* \in \Omega$ satisfying $G(x^*) = 0$. If $\partial G(x^*)$ is nonsingular, then there exists a $\delta > 0$ such that, if $x^0 \in \mathbb{B}(x^*, \delta)$, the sequence $\{x^k\}$ generated by Algorithm 2 is well defined and converges Q-superlinearly to x^*. Furthermore, if G is strongly semismooth at x^*, then the convergence rate is Q-quadratic.* □

As one can expect, the semismooth functions are not the only class of functions with linear Newton approximation schemes. The definition is an attempt to capture when G has a Newton approximation scheme defined by linear functions such that the nonsingularity of the scheme is the immediate consequence of the nonsingularity of a set of distinguished matrices.

Definition 2.11. Let $G : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^m$, with Ω open, be locally Lipschitz on Ω. We say that G admits a linear Newton approximation at a vector $\bar{x} \in \Omega$ if there exists a multifunction $T : \Omega \to \mathbb{R}^{n\times n}$ such that $T(x)$ is a Newton approximation scheme of G at $\bar{x}$ and T has nonempty compact images and is upper semicontinuous at $\bar{x}$. If T is a strong Newton approximation scheme, then we say the G admits a strong linear Newton approximation scheme, then we say the G admits a strong linear Newton approximation at $\bar{x}$. We also say that T is a (strong) linear Newton approximation scheme of G. □

Based on the definition, we describe the basic Newton method and its convergence properties.

Algorithm 3 Linear Newton method[a]

Data: Let $x^0 \in \mathbb{R}^n$.

Step 1: Set $k = 0$.

Step 2: If $G(x^k) = 0$, stop.

Step 3: Select an element $H^k \in T(x^k)$. Find a direction $d^k \in \mathbb{R}^n$ such that

$$G(x^k) + H^k d^k = 0.$$

Step 4: Set $x^{k+1} \equiv x^k + d^k$ and $k \leftarrow k + 1$; go to Step 2.

[a] Facchinei and Pang [11]

The following theorem is the convergence result for Algorithm 3.

Theorem 2.19. *Assume that $G : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^n$, with Ω open, is locally Lipschitz continuous on Ω and has a linear Newton approximation scheme T at $x^* \in \Omega$, which satisfies $G(x^*) = 0$. Suppose that all the matrices H belonging to $T(x^*)$ are nonsingular. There exists a $\delta > 0$ such that, if $x^0 \in \mathbb{B}(x^*, \delta)$, the sequence $\{x^k\}$ generated by Algorithm 3 is well defined and converges Q-superlinearly to the solution x^*. Furthermore, if the linear approximation scheme T is strong, then the convergence rate is Q-quadratic.* □

In the next theorem, a systematic way of generating linear Newton approximation of a composite map is considered.

Theorem 2.20. *Suppose that $G : \mathbb{R}^n \to \mathbb{R}^n$ is a locally Lipschitz continuous map given by the composition of two maps: $G \equiv A \circ B$, where $B : \mathbb{R}^n \to \mathbb{R}^m$ and $A : \mathbb{R}^m \to \mathbb{R}^n$ are both locally Lipschitz. Suppose that T_A and T_B are (strong) linear Newton approximation schemes of A and B at $B(\bar{x})$ and $\bar{x}$ respectively. Then*

$$T(x) \equiv \{VW : V \in T_A(B(x)), W \in T_B(x)\},$$

is a (strong) linear Newton approximation scheme of G at $\bar{x}$. □

3 Valuing American options

As a classical problem in financial mathematics, several methods are developed to value American options. In this section, three numerical methods are applied to solve this problem and the corresponding numerical results are also presented. Here, we use the formulation in Bokanowski et al. [5]. Let $u(s,t)$ be the value of the American put option where $s \in [0, S_{\max}]$ is the price of the underlying stock and $t \in [0,T]$ is the time before maturity, and when $t = 0$ the option matures. The value function $u(s,t)$ satisfies the following nonlinear partial differentiable equation:

$$\begin{aligned} \min\left(\partial_t u - \frac{1}{2}\sigma^2 s^2 \partial_{ss} u - rs\partial_s u + ru, u - \varphi(s)\right) = 0, \\ t \in (0,T], s \in (0, S_{\max}) \\ u(t, S_{\max}) = 0, t \in [0,T] \\ u(0,s) = \varphi(s), s \in (0, S_{\max}) \end{aligned} \tag{3.1}$$

where $\varphi(s) = \max(K - s, 0)$ is the value when the option exercises, $K > 0$ is the strike price, $\sigma > 0$ is the volatility of the underlying stock, $r > 0$ is the interest rate of the bank account. The upper bound $S_{\max}$ should be the infinity, but for numerical purpose, we consider $S_{\max}$ positive, large enough but finite. This assumption is suitable, and it is proved by the numerical results.

3.1 The discretization scheme[2]

We insert N_t grids into the interval $[0,T]$ and N_s grids into the interval $[0, S_{\max}]$. Let $\Delta t = \frac{T}{N_t}$ and $\Delta S = \frac{S_{\max}}{N_s}$. Let $S_j = j\Delta S$ and $t_n = n\Delta t$. Denote $U_j^n \equiv u(S_j, t_n)$. We use the implicit Euler method to discretize (3.1); that is, when $N_t \geq n \geq 1$ and $N_s - 1 \geq j \geq 1$,

$$\begin{aligned} \partial_t u(\cdot, t_n) &\sim \frac{U^n - U^{n-1}}{\Delta t} \\ \partial_{ss} u(s_j, \cdot) &\sim \frac{U_{j+1} - 2U_j + U_{j-1}}{\Delta S^2} \\ \partial_s u(s_j, \cdot) &\sim \frac{U_{j+1} - U_j}{\Delta S} \end{aligned} \tag{3.2}$$

[2] Bokanowski et al. [5]

When $n = 0$, $U_j^n = \varphi(S_j)$. When $j = N_s$, $U_j = 0$. When $j = 0$, the equation (3.1) degenerates and its discretized approximation is

$$\min\left(\frac{U_j^n - U_j^{n-1}}{\Delta t} - rU_j^n, U_j^n - \varphi(S_0)\right) = 0.$$

Remark 3.1. In Section 3.3, a modified discretization scheme is used. To improve the order of convergence and the stability, the Crank-Nicolson method is applied. Moreover, $\partial_s u(s_j, \cdot) \sim \dfrac{U_{j+1} - U_{j-1}}{\Delta S}$. □

Since the vector U^n is known, we can simply view (3.2) as $\min(A^1x - b^1, A^2x - b^2) = 0$, where the two $(N_s + 1) \times (N_s + 1)$ matrices A^1 and A^2 has the following property:

$$A_{ii}^k \le \delta + \sum_{i \ne j} |A_{ij}^k| \text{ and } A_{ij}^k < 0,\ k \in \{1, 2\}, 1 \le i, j \le N_s + 1,$$

where $\delta > 0$ is a constant. Thus they are strictly diagonal dominant M-matrices. The strictly diagonal dominant M-matrix A is invertible and satisfies that $Ax \ge 0$ implies $x \ge 0$ (see Horn and Johnson [13; 14] and Appendix A). Moreover, if a square matrix is composed of the rows of strictly diagonal dominant M-matrices, this square matrix is also a strictly diagonal dominant M-matrix.(see Appendix A)

3.2 Howard's algorithm

Howard's algorithm was proposed in Howard [15]. Bokanowski et al. [5] use Howard's algorithm to solve the problem of valuing American put option. We describe Howard's algorithm for the NCP $\min(A^1x - b^1, A^2x - b^2) = 0$ where A^1 and A^2 are $m \times m$ matrices.

Algorithm 4 Howard's algorithm for NCP[a]

Data: x^0 is given and $k = 0$;

Step 1: Find a matrix B^k and a vector c^k. For each integer $i \in \{1, \ldots, m\}$, if $A_{i\cdot}^1 x^k - b_i^1 < A_{i\cdot}^2 x^k - b_i^2$, let $B_{i\cdot}^k = A_{i\cdot}^1$ and $b_i^k = b_i^1$; otherwise, let $B_{i\cdot}^k = A_{i\cdot}^2$ and $b_i^k = b_i^2$;

Step 2: Find x^{k+1} solution of $B^k x = b^k$;

Step 3: if $x^k = x^{k+1}$, stop. Otherwise, $k \leftarrow k + 1$, and go to Step 1.

[a] Facchinei and Pang [11]

There is an interconnection between semismooth Newton method and Howard's algorithm. It is easy to see that $G(x) \equiv \min(A^1x - b^1, A^2x - b^2)$ is a PC^1 function and B^k is also one element of the generalized Jacobian of $G(x^k)$ at x^k. Moreover, if the iteration does not stop at x^k, $x^{k+1}s$ generated by two algorithms are the same. Thus we can obtain the convergence results directly from the convergence results of semismooth Newton method. The proposition below shows that under the monotonicity assumption on the matrices A^1 and A^2, a global convergence result is obtained.

Propositon 3.1. *Assume that for any if for any $i \in \{1, \ldots, m\}$, the i-th row of a matrix A equals to the i-th row of the matrix A^1 or the i-th row of the matrix A^2, then A are monotone; that is, $Ax \ge 0$ implies $x \ge 0$. Then there exists a unique x^* solution of the NCP* $\min(A^1x - b^1, A^2x - b^2) = 0$. *Moreover, the sequence $\{x^k\}$ given by Howard's algorithm satisfies*

(a) $x^k \le x^{k+1}$ *for all* $k \ge 1$*;*

(b) $\{x^k\}$ *converges to* x^* *pointwisely;*

(c) *there exists a finite number* k *such that* $x^k = x^*$.

This proof is given in Appendix B.

Remark 3.2. The finite termination property of Howard's algorithm is a consequence of the fact that F and G are affine functions and the function $G(x) \equiv \min(F(x), G(x))$ is regular at the limit point x^*. Due to the finite termination, it is meaningless to check the convergence rate of Howard's algorithm. As we know, in the finite dimensional case, the Q-superlinear (Q-quadratic) convergence is achieved if the function G in Algorithm 2 is (strongly) semismooth. But in the infinite dimensional case, this result does not hold. It is because if a sequence of vectors converges componentwisely, we can not immediately have the result that this sequence converges. More assumptions are necessary. Suppose that x is in the infinite dimensional vector space $\mathbb{R}^{\mathbb{N}^*}$. If several assumptions are satisfied, we can show that Howard's algorithm is globally superlinearly convergent. Moreover, if one more assumption is satisfied, Howard's algorithm is finite terminated. (see Appendix C)

3.3 Penalty method

We introduce one scheme in Dai et al. [8], which is modified from the scheme Forsyth and Vetzal [12]. The penalty approximation of (3.1) is written as:

$$\partial_t u = -\frac{1}{2}\sigma^2 s^2 \partial_{ss} u - rs\partial_s u + ru + \rho \min(u - \varphi(s), 0), \tag{3.3}$$

where ρ is the penalty parameter and $\rho \min(u - \varphi(s), 0)$ is the penalty term. As we mentioned in Remark 3.1, the discretization scheme for (3.3) excluding the penalty term is a little different from (3.2), and to approximate the penalty term at U_j^n, we use the following discretization scheme:

$$\xi\Big(\frac{U_j^{n-1} + U_j^n}{2} - \varphi(S_j)\Big)$$

where

$$\xi = \begin{cases} \rho, & \text{if } \frac{U_j^{n-1}+U_j^n}{2} - \varphi(S_j) < 0; \\ 0, & \text{otherwise.} \end{cases}$$

This penalty approximation can be viewed as a PC^1 function, if we take the outside terms into the min function. Thus we can perform the semismooth Newton method (Algorithm 2). In Dai et al. [8], Forsyth and Vetzal [12] it is shown that the iteration has a finite termination property; when an iterate is sufficiently close to x^*, the iteration converges in one step.

3.4 Nonsmooth Newton method based on FB function

With the monotonicity assumption, both Howard's algorithm and the penalty method have the finite termination property. In contrast, the nonsmooth Newton method based on FB functional does not have

such a property, but when the monotonicity assumption omitted, the nonsmooth Newton method based on FB functional is still a Q-quadratic global method.

We consider the NCP $\min(F(x), H(x)) = 0$, where F and H are continuously differentiable functions. The nonsmooth Newton method based on FB function is basically a line search method, proposed in Facchinei and Pang [11]. Some concepts are necessary. Let

$$F_{FB}(x) = \begin{bmatrix} \psi_{FB}(F_1(x), G_1(x)) \\ \vdots \\ \psi_{FB}(F_m(x), G_m(x)) \end{bmatrix}$$

where ψ_{FB} is a FB C-function, defined as

$$\psi_{FB}(a, b) \equiv \sqrt{a^2 + b^2} - a - b, \quad (a, b) \in \mathbb{R}^2.$$

The associate merit function is the square Euclidean norm of F_{FB}, that is,

$$\theta_{FB}(x) = \frac{1}{2} F_{FB}(x)^T F_{FB}(x).$$

The differentiability results of the FB function F_{FB} and its according merit function θ_{FB} is listed in the following proposition.

Propositon 3.2. *Let $F, G : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}^m$ be continuous differentiable functions on the open set $\mathcal{D}$. The following statements hold.*

(a) *The generalized Jacobian of $F_{FB}(x)$ satisfies*

$$\partial F_{FB}(x) \subseteq \mathcal{D}_a(x) JF(x) + \mathcal{D}_b(x) JG(x),$$

where $\mathcal{D}_a(x)$ and $\mathcal{D}_b(x)JG(x)$ are the sets of $n \times n$ diagonal matrices $diag(a_1(x), \ldots, a_n(x))$ and $diag(b_1(x), \ldots, b_n(x))$ respectively, with

$$(a_i(x), b_i(x)) \begin{cases} \equiv \frac{(F_i(x), G_i(x))}{\sqrt{F_i(x)^2 + G_i(x)^2}} - (1, 1), & \text{if } (F_i(x), G_i(x)) \neq 0; \\ \in cl\mathbb{B}(0, 1) - (1, 1), & \text{if } (F_i(x), G_i(x)) = 0. \end{cases}$$

where clA means the closure of the set A.

(b) *$F_{FB}(x)$ is semismooth on $\mathcal{D}$.*

(c) *$\theta_{FB}(x)$ is continuously differentiable on $\mathcal{D}$ and its gradient $\nabla\theta_{FB}(x)$ is equal to $H^T F_{FB}(x)$ for every H in $\partial F_{FB}(x)$.*

(d) *If the Jacobian $JF(x)$ is locally Lipschitz on $\mathcal{D}$, then $F_{FB}(x)$ is strongly semismooth on $\mathcal{D}$.* □

Let $T(x)$ be a linear approximation of the C-function F_{FB} at x.

Algorithm 5 FB line search algorithm[a]

Data: $x^0 \in \mathbb{R}^m$, $\rho > 0$, $p > 1$ and $\gamma \in (0, 1)$

Step 1: Set $k = 0$.

Step 2: If $\theta_{FB}(x^k) = 0$, stop.

Step 3: Select an element H^k in $T(x^k)$ and find a solution d^k of the system

$$F_{FB}(x^k) + H^k d = 0; \tag{3.4}$$

if (3.4) is not solvable or if the condition $\nabla\theta_{FB}(x^k)^T d^k \leq -\rho\|d^k\|^p$ is not satisfied, set $d^k = -\nabla\theta_{FB}(x^k)$

Step 4: Find the smallest nonnegative integer i_k such that, with $i = i_k$,

$$\theta_{FB}(x^k + 2^{-i}d^k) \leq \theta_{FB}(x^k) + \gamma 2^{-i}\nabla\theta_{FB}(x^k)^T d^k. \tag{3.5}$$

Set $d^k \leftarrow 2^{-i}d^k$.

Step 5: Set $x^{k+1} \equiv x^k + d^k$ and $k \leftarrow k + 1$; go to Step 2.

[a]Facchinei and Pang [11]

To ease computation and for other practical reasons, we describe a simple procedure to compute a matrix H in $Jac\, F_{FB}(x)$, which is a good choice for the linear Newton approximation T in Algorithm 5.

Procedure to calculate an element H in $Jac\, F_{FB}(x)$[a]

Step 1: Set $\beta \equiv \{i : x_i = 0 = F_i(x)\}$.

Step 2: Choose $z \in \mathbb{R}^n$ such that $z_i \neq 0$ for all i belonging to β.

Step 3: For each $i \notin \beta$ set the i-th column of H^T equal to

$$\Big(\frac{G_i(x)}{\sqrt{G_i(x)^2 + F_i(x)^2}} - 1\Big)\nabla G_i(x) + \Big(\frac{F_i(x)}{\sqrt{G_i(x)^2 + F_i(x)^2}} - 1\Big)\nabla F_i(x).$$

Step 4: For each $i \in \beta$ set the i-th column of H^T equal to

$$\Big(\frac{\nabla G_i(x)^T z}{\sqrt{(\nabla G_i(x)^T z)^2 + (\nabla F_i(x)^T z)^2}} - 1\Big)\nabla G_i(x) + \Big(\frac{\nabla F_i(x)^T z}{(\nabla G_i(x)^T z)^2 + (\nabla F_i(x)^T z)^2} - 1\Big)\nabla F_i(x).$$

[a]Facchinei and Pang [11]

Such H is proved to be an element in $Jac\, F_{FB}(x)$ in Facchinei and Pang [11]. We summarize the convergence results of Algorithm 5 in the following theorem[3].

[3]Facchinei and Pang [11]

Theorem 3.3. *Let $F, G : \mathbb{R}^n \to \mathbb{R}^n$ be continuously differentiable and let T be linear Newton approximation scheme of $F_{FB}(x)$. Let $\{x^k\}$ be an infinite sequence generated by Algorithm 4.*

(a) *Every accumulation point of $\{x^k\}$ is a stationary point of the merit function $\theta_{FB}(x)$.*

(b) *If x^* is an accumulation point of $\{x^k\}$ such that x^* is regular, then x^* is a solution of* $\min(F(x), G(x)) = 0$.

(c) *If $\{x^k\}$ has an isolated limit point, then the whole sequence $\{x^k\}$ converges to that point.*

(d) *Suppose x^* is a limit point of $\{x^k\}$ and a solution of* $\min(F(x), G(x)) = 0$. *Assume that $T(x) \subseteq \partial F_{FB}(x)$ for every x in a neighborhood of x^* and that all the matrices belonging to $T(x^*)$ are nonsingular. The whole sequence $\{x^k\}$ converges to x^*; furthermore, if $p > 2$ and $\gamma < \frac{1}{2}$ in Algorithm 4, the following statements hold:*

  (i) *Eventually d^k is always the solution of* (3.4)*;*

  (ii) *Eventually a unit step size is accepted so that $x^{k+1} = x^k + d^k$;*

  (iii) *The convergence rate is Q-superlinear; furthermore, if the Jacobian $JF(x)$ is Lipschitz continuous in a neighborhood of x^*, the convergence rate is Q-quadratic.* □

3.5 Numerical results

We primarily concern the numerical properties of different methods. To easily compare the results, we only consider the number of iterations required to obtain the value of U^1. The reason is that due to the existence of the time derivative, as t increases, the error in calculating U^t is accumulating which undermines the accuracy of the analysis. We use the numbers of iterations to compare the performance of algorithms, since solving linear systems is the main calculation task for algorithms and the numbers show how much work algorithms are required. We mainly examine two properties: (a) the global convergence and (b) the convergence rate. To examine the global convergence property of algorithms, we use different initial values and observe whether the solutions of different algorithms are the same. We also concern the numbers of iterations the algorithms take to stop. We check the effect of increasing the numbers of spatial grids on the numbers of iterations to stop to study the convergence speed.

We start from the analysis of Howard's algorithm. For different initial values, the results of U^1 generated with different initial values are the same. In Table 1, the numbers of iterations is relatively small, comparing with the number of grids. If the initial value x^0 is smaller than the value U^1, fewer iterations requires. This is because of the monotonicity property. With x^0 small enough, the x^1 generated by Howard's algorithm is much closer to the limit point. This result also appears in the next section. As the number of grids doubles, the number of iterations almost doubles in Table 2. Since the increase of the spatial grids results in a heavier burden of calculation required in each iteration, the proportional increase in numbers of iterations means that Howard's algorithm costs more time and requires much more calculations. This explains why much longer time is taken when the number of grids is more than 500 in practice.

From Table 3 and Table 4, we can tell that both the initial values and numbers of spatial grids do not affect the algorithm. A fast converge rate is always achieved. The drawback of the penalty method is that the

solution is only an approximation. If the error brought by the discretization scheme is much larger than the error brought by the penalty approximation, it is an acceptable solution. As the penalty parameter ρ is large enough, the distance between the solutions of the penalty method and Howard's algorithm is small. Thus, the penalty approximation provides a good solution in this problem.

For the problem of valuing American put options, when the initial value is far away from the solution, the convergence rage of the FB nonsmooth Newton method is too slow (see Table 6). As the number of spatial grids increases, the number of iterations does not increase rapidly, but the number of iterations is very large when the number of spatial grids is 50 (see Table 6). This shows that the FB nonsmooth Newton method requires a good initial value close to the expected solution and a larger number of grids to ensure a fast convergence rate. We also examine the convergence rate when the initial value is U^0 and the number of spatial grids is 100. Obviously the Q-superlinear convergence is ascertained from Figure 1, but the Q-quadratic convergence is not clear from Figure 2.

4 Valuing American warrant subject to issuer's calling

We use the formulation of the problem of valuing American warrant subject to issuer's calling in Dai et al. [8]. Let $V(S,t)$ be the price function of the American warrant, where S is the stock price. Let $\phi(S)$ be payoff upon exercise and $\psi(S)$ be the rebate received by the holder upon calling. Assume the arrival of calling by issuer to be governed by a Poisson process with intensity $\rho\mathbf{1}_{\{V>\psi\}}$, where ρ is a constant. The indicator function $\mathbf{1}_{\{V>\psi\}}$ is included since the issuer calls only when $V > \psi$. In our callable American warrant, we have the exercise payoff $\phi(S) = S - K$ and rebate $\psi(S) = R$. Under the risk neutral measure Q, the dynamics of the stock price are assumed to be governed by

$$\frac{\mathrm{d}S}{S} = (r-q)\,\mathrm{d}t + \sigma\,\mathrm{d}Z,$$

where q is the constant dividend yield of the stock. The governing equation for $V(S,t)$ in the continuation region where the warrant remains alive is:

$$rV\,\mathrm{d}t = \left[\frac{\partial V}{\partial t} + \frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2} + (r-q)S\frac{\partial V}{\partial S}\right]\mathrm{d}t + \rho\,\mathrm{d}t\max(V-\psi,0).$$

In the stopping region where it is optimal for the holder to exercise the warrant, we have

$$V(S,t) = \phi(S).$$

We write $\tau \equiv T - t$, where τ is the time to expiry and let $\mathbf{L}_w$ be the differential operator defined as

$$\mathbf{L}_w \equiv \frac{\sigma^2}{2}S^2\frac{\partial^2}{\partial S^2} + (r-q)S\frac{\partial}{\partial S} - r.$$

The complementarity formulation for $V(S,t)$ is given by

$$\min\left(\frac{\partial V}{\partial \tau} - \mathbf{L}_w V + \rho\max(V-\psi,0),\ V-\phi\right) = 0,$$

subject to the termination condition $V(S,0) = \max(\phi(S),0)$. We use the finite element method (3.2) to discretize this problem. Then we can view this problem as a double obstacle problem $\min(\max(A^1x - b^1, A^2x - b^2), A^3x - b^3) = 0$, where A^1,A^2 and A^3 are all strictly row diagonal dominant M-matrices.

4.1 Howard's algorithm for double-obstacle problems[4]

Similar as Algorithm 4, we present Howard's algorithm for double-obstacle problem $\min(\max(A^1x - b^1, A^2x - b^2), A^3x - b^3) = 0$.

Algorithm 6 Howard's algorithm for double-obstacle problems[a]

Data: $x^0 \in \mathbb{R}^n$

Step 1: Set $k = 0$.

Step 2: Find a matrix B^k and a vector c^k satisfying for all $i = 1, \ldots, n$,

- if $A^1_{i\cdot}x^k - b^1_i \geq A^2_{i\cdot}x^k - b^2_i$ and $A^1_{i\cdot}x^k - b^1_i \leq A^3_{i\cdot}x^k - b^3_i$, let $B^k_{i\cdot} = A^1_{i\cdot}$ and $c^k_i = b^1_i$;
- if $A^2_{i\cdot}x^k - b^2_i > A^1_{i\cdot}x^k - b^1_i$ and $A^2_{i\cdot}x^k - b^2_i \leq A^3_{i\cdot}x^k - b^3_i$, let $B^k_{i\cdot} = A^2_{i\cdot}$ and $c^k_i = b^2_i$;
- otherwise, let $B^k_{i\cdot} = A^3_{i\cdot}$ and $c^k_i = b^3_i$.

Step 3: Let x^{k+1} be the solution of the equation $B^kx = c^k$.

Step 4: If $x^{k+1} = x^k$, stop; otherwise, $k \leftarrow k + 1$.

[a] Bokanowski et al. [5]

As we explain in Section 3.2, the double-obstacle problem can also be viewed as a PC^1 function with 3^n pieces. Similarly we have the following proposition.

Propositon 4.1. *Assume that for any if $i \in \{1, \ldots, n\}$, the i-th row of a matrix A equals to the i-th row of one of the matrices A^1, A^2 and A^3, then A is monotone; that is, $Ax \geq 0$ implies $x \geq 0$. Then there exists a unique x^* solution of the double-obstacle problem* $\min(\max(A^1x - b^1, A^2x - b^2), A^3x - b^3) = 0$. *Moreover, the sequence $\{x^k\}$ given by Howard's algorithm satisfies*

(a) *$x^k \leq x^{k+1}$ for all $k \geq 1$;*

(b) *$\{x^k\}$ converges to x^* pointwisely;*

(c) *there exists a finite number k such that $x^k = x^*$.* □

We put the proof in Appendix D.

4.2 The min-FB line search method

Pang and Gabriel [18] proposed a damped Newton algorithm for the problem $\min(F(x), G(x)) = 0$, named NE/SQP, applied to the NCP. This algorithm involves solving a sequence of convex quadratic programming problems of the least-square type. Theoretically, there are two drawbacks for NE/SQP comparing with the nonsmooth Newton method based on FB function introduced below: (a) a quadratic programming problem has to be solve at each iteration comparing with FB-based algorithm only need to solve a linear programming problem; (b) the solution of this algorithm is not automatically the stationary

[4] Bokanowski et al. [5]

point, without the regularity conditions, while the stationary result is a natural consequence of the continuously differentiability of the merit function for the FB-based algorithm.

We consider the NCP $F_{\min}(x) \equiv \min(F(x), G(x)) = 0$. Suppose that $F_{\min}$ admits a linear Newton approximation at the zero x^* of the function $F_{\min}$. We denote this linear Newton approximation as $T_{\min}$ We describe th algorithm below.

Algorithm 7 min-FB line search algorithm[a]

Data: $x^0 \in \mathbb{R}^n$, $\varepsilon > 0$, $\rho > 0$, $p > 1$ and $\gamma \in (0, 1)$.

Step 1: Set $k = 0$.

Step 2: If $\theta_{FB}(x^k) = 0$, stop.

Step 3: Select an element H^k in $T_{\min}(x^k)$ and find a solution of the system

$$F_{\min}(x^k) + H^k d = 0. \tag{4.1}$$

If (4.1) is solvable and the solution d^k satisfies $\|F_{FB}(x^k + d^k)\| \leq \gamma \|F_{FB}(x^k)\|$. Set $\tau = 1$ and to Step 5. Otherwise, if this system is not solvable or if the condition $\nabla\theta_{FB}(x^k)^T d^k \leq -\rho\|d^k\|^p$ is not satisfied, set $d^k = -\nabla\theta_{FB}(x^k)$.

Step 4: Find the smallest nonnegative integer i_k such that with $i = i_k$, $\theta_{FB}(x^k + 2^{-i}d^k) \leq \theta_{FB}(x^k) - \gamma 2^{-i}\nabla\theta_{FB}(x^k)^T d^k$ and set $\tau = 2^{-i_k}$.

Step 5: Set $x^{k+1} \equiv x^k + \tau d^k$ and $k \leftarrow k + 1$; go to Step 2.

[a] Facchinei and Pang [11]

In Algorithm 6, F_{FB} and θ_{FB} is the same to what we define in Algorithm 4. To use Algorithm 6, we need to transform the double-obstacle problem into a NCP. Here we can replace the max function $\max(F(x), H(x))$ with the FB-type function $\varphi(F(x), H(x)) \equiv \sqrt{F^2(x) + H^2(x)} + F(x) + H(x)$. $\varphi(F(x), H(x))$ pertains the sign of the max function $\max(F(x), H(x))$ (see Appendix E). For the case when F, G are continuously differentiable, Algorithm 6 has several good convergence results, which inherits from the FB line search algorithm. We do not repeat these results.

The double-obstacle problem $\min(\max(F(x), H(x)), G(x)) = 0$ becomes $\min(\varphi(F(x), H(x)), G(x)) = 0$. As we know, φ is not continuously differentiable at the origin. Thus the corresponding merit function θ_{FB} is not differentiable everywhere.(see Appendix F) Consequently, Algorithm 7 does not have the properties (a),(b),(c) and (d)(i)(ii) in Theorem 3.3. Fortunately, φ admits a strong Newton approximation generated similarly by the procedure below Algorithm 5 with F and G replaced by $-F$ and $-H$ and θ_{FB} is also a strong semismooth function. Due to Theorem 2.20, $\theta_{FB}(\varphi(F(x), H(x)), G(x))$ admits a strong linear Newton approximation. Thus, by Theorem 2.19, a local convergence result ((d)(iii) in Theorem 3.3) of Algorithm 7 is assured. Moreover, if there exists a number k such that for any $s > k$, $G(x^s) \geq k$ holds, by the conclusion in Appendix F, we have that θ_{FB} is continuously differentiable, and thus (d)(i)(ii) in Theorem 3.3 holds. (a),(b) and (c) in Theorem 3.3 need more condition to hold in this case.

4.3 Numerical results

We perform the same numerical examination as in Section 3.5. For Howard's algorithm, we have the similar results. The number of iterations increases almost proportionally with the increase in the number of spatial grids. With the small initial value, the number of iterations before stop are required.

The min-FB line search method has similar results to the FB line search method. It seems more stable than the FB line search method with different initial values. All iterations with different initial values take a reasonable small number of steps before stop. We also run the examination on the convergence rate. From Figure 3 and Figure 4, it seems that two limits

$$\lim_{k\to\infty} \frac{\|x^{k+1} - x^{\infty}\|}{\|x^{k+1} - x^{\infty}\|} = 0 \text{ and } \limsup_{k\to\infty} \frac{\|x^{k+1} - x^{\infty}\|}{\|x^{k+1} - x^{\infty}\|^2} < \infty$$

hold.

5 Portfolio selection problem with transaction costs

Merton initiated the study of continuous-time portfolio selection problems in Merton [16; 17]. He show that the optimal strategy of a constant relative risk aversion (CRRA) investor is to keep a constant fraction of total wealth in each asset and to consume at a constant rate when there is no transaction cost. Such a strategy leads to incessant tradings, which seems impracticable in a real market with transaction costs. A series of papers follow and study the portfolio selection problem with transaction costs, which can be viewed as a stochastic control problem and the value function is consider as a solution of a generalized complementarity problem with gradient constraints. Most theoretical studies of the problem focus on a market consisting of one single risky asset and a bank account. Several numerical methods are used to study the market with one or even two risky assets and one bank account. Davis and Norman [9] propose a theoretical analysis for an infinite horizon investment and consumption decision with transaction costs. Shreve and Soner [20] show that viscosity solutions of optimal trading strategies in the infinite horizon case exist. Davis et al. [10] suggest using the binomial model to study the problem of pricing European options with transaction costs. Using the binomial method, Dai and Yi [7] verify their analysis on the buying and selling boundary in the optimal investment problem of a CRRA who faces the proportional transaction costs and finite time horizon. Akian et al. [1; 2] propose an algorithm by combining the Howard algorithm with the multigrid method to overcome the volatility brought by gradient constraints in the finite horizontal case. Dai and Zhong [6] use penalty method and several times of changing variables to enhance the performance of the nonsmooth Newton method.

5.1 Portfolio selection problem with transaction costs

Suppose that the market consists of one risky asset and one bank account. Their prices, denoted by $S(t)$ and $B(t)$ respectively, evolve according to the following equations:

$$\begin{aligned} \mathrm{d}S &= \alpha S \,\mathrm{d}t + \sigma S \,\mathrm{d}W, \\ \mathrm{d}B &= rB \,\mathrm{d}t. \end{aligned}$$

where $r > 0$ is the constant interest rate, $\alpha > r$ and $\sigma > 0$ are constant expected rate of return and volatility of the risky asset respectively. The process $\{W(t) : t > 0\}$ is the standard Brownian motions on a filtered probability space $(S, \mathscr{F}, \{\mathscr{F}_t\}_{t\geq 0}, P)$ with $W_0 = 0$ almost surely. We assume that the filtration $\{\mathscr{F}_t\}_{t\geq 0}$ is right-continuous and each $\mathscr{F}_t$ contains all $\mathscr{F}$-nulls of $\mathscr{F}_\infty$.

Assume that an investor hold a portfolio $X(t) = \Big(X_0(t), X_1(t)\Big)$ where X_0 and X_1 are respectively the dollar values in bank and in the risky asset at time t. In the presence of proportional transaction costs, the equations of their values are

$$\begin{aligned} \mathrm{d}X_0 &= rX_0\,\mathrm{d}t - (1+\lambda)\,\mathrm{d}L + (1-\mu)\,\mathrm{d}M, \\ \mathrm{d}X_1 &= \alpha X_1\,\mathrm{d}t + \sigma X_1\,\mathrm{d}W + \mathrm{d}L - \mathrm{d}M. \end{aligned}$$

Here, $L(t)$ and $M(t)$ are right-continuous, nonnegative and nondecreasing $\{\mathscr{F}_t\}_{t\geq 0}$-adapted processes with $L(0) = M(0) = 0$, representing cumulative dollar values for the purpose of buying and selling the risky asset respectively. The constant $\lambda \in [0, \infty)$ and $\mu \in [0, 1)$ account for proportional transaction costs incurred on purchase and sale of the risky asset, respectively. We assume that $\lambda + \mu > 0$.

Due to the existance of transaction costs, the investor's net wealth in monetary terms is $X_0 + (1-\mu)X_1^+ - (1+\lambda)X_1^-$. With the requirement that the net wealth at any time always be positive, the solvency region is

$$\{x = (x_0, x_1) \in \mathbb{R}^2 : x_0 + (1-\mu)x_1^+ - (1+\lambda)x_1^- > 0\}. \tag{5.1}$$

An investment strategy (L, M) is admissible for a positive x from $s \in [0, T]$ if $X(t)$ governed by (5.3) with $X(s) = x$ is in the solvency region. We denote by $\mathcal{A}(x)$ the set of all admissible investment strategies for x from time s. We consider CRRA investors whose utility function takes the following form:

$$u(V) = \begin{cases} \frac{V^\gamma}{\gamma}, & \gamma \neq 0, \gamma < 1; \\ \log V, & \gamma = 0. \end{cases}$$

Define the value function by

$$V(x,t) = \sup_{(L,M)\in\mathcal{A}(x)} E_t^{X(t)=x}[e^{-\beta T}u(V_T)].$$

where $\beta > 0$ is the discount rate. The value function satisfies the following HJB equation

$$\max\left(\frac{\partial V}{\partial t} + \mathbf{A}V, \mathbf{L}V, \mathbf{M}V\right) = 0,\ t \in [0, T)$$

with the terminal condition $V(x, T) = u(x_0 + (1-\mu)x_1^+ - (1+\lambda)x_1^-)$, where

$$\begin{aligned} \mathbf{A}V &= \frac{1}{2}\sigma^2 x_1^2 \frac{\partial^2 V}{\partial x_1^2} + \alpha x_1 \frac{\partial V}{\partial x_1} + r x_0 \frac{\partial V}{\partial x_0} - \beta V, \\ \mathbf{L}V &= -(1+\lambda)\frac{\partial V}{\partial x_0} + \frac{\partial V}{\partial x_1}, \\ \mathbf{M}V &= (1-\mu)\frac{\partial V}{\partial x_0} - \frac{\partial V}{\partial x_1}. \end{aligned}$$

Let

$$y = \frac{x_1}{x_0 + x_1} \text{ and } \varphi(y,t) = V(1-y,y,t).$$

Due to the homotheticity of the utility function, we obtain that for $\gamma \neq 0$ and $\gamma < 1$,

$$\begin{cases} \max\left(\frac{\partial \varphi}{\partial t} + \mathbf{A}_1\varphi, \mathbf{L}_1\varphi, \mathbf{M}_1\varphi\right) = 0, \\ \varphi(y,T) = \frac{(1-\mu y^+ - \lambda y^-)^\gamma}{\gamma}, y \in \Omega, t \in [0,T), \end{cases} \tag{5.2}$$

where $\Omega = \{y \in \mathbb{R} : 1 - \mu y^+ - \lambda y^- > 0\}$,

$$\begin{aligned} \mathbf{A}_1\varphi &= a\frac{\partial^2 \varphi}{\partial y^2} + b\frac{\partial \varphi}{\partial y} - \theta\gamma\varphi, \\ \mathbf{L}_1\varphi &= (1+\lambda y)\frac{\partial \varphi}{\partial y} - \lambda\gamma\varphi, \\ \mathbf{M}_1\varphi &= (-1+\mu y)\frac{\partial \varphi}{\partial y} - \mu\gamma\varphi, \end{aligned}$$

with

$$\begin{aligned} a &= \frac{1}{2}\sigma^2 y^2 (1-y)^2, \\ b &= y(1-y)(\alpha - r + (\gamma-1)\sigma^2 y), \\ \theta &= \frac{\beta}{\gamma} - \left(r + y\left(\alpha - r - \frac{1-\gamma}{2}\sigma^2 y\right)\right). \end{aligned}$$

For this above formulation, we have the following penalty approximation:

$$\frac{\partial \varphi}{\partial t} + \mathbf{A}_1\varphi + K\Big[\max(\mathbf{L}_1\varphi, 0) + \max(\mathbf{M}_1\varphi, 0)\Big] = 0 \tag{5.3}$$

We can make the further transformation of variables. Let $w(y,t) = \frac{\log(\gamma\varphi)}{\gamma}$ when $\gamma < 1$ and $\delta \neq 0$, and when $\gamma = 0$, let $w(y,t) = \frac{\varphi(y,t)}{g(t)}$ where $g(t) = e^{-\beta(T-t)}$. Moreover, let $v = w_y$. Thus, we obtain for $\gamma < 1$,

$$\begin{cases} \min\left(\max\left(-v_t - \mathcal{T}v, v - \frac{\lambda}{1+\lambda y}\right), v + \frac{\mu}{1-\mu y}\right) = 0, \\ v(y,T) = -\frac{\mu}{1-\mu y} \text{ in } y \in [0, \frac{1}{\mu}), t \in [0,T). \end{cases} \tag{5.4}$$

where

$$\begin{aligned} \mathcal{T}v =& \frac{1}{2}\sigma^2 y^2 (1-y)^2 v_{yy} + (\alpha - r + (\gamma-1)\sigma^2 y + (1-2y)\sigma^2) y(1-y)v_y \\ &+ ((\alpha - r)(1-2y) + (\gamma-1)\sigma^2 y(2-3y))v + (\alpha - r + (\gamma-1)\sigma^2 y) \\ &+ \gamma\sigma^2 y(1-y)v((1-2y)v + y(1-y)v_y). \end{aligned}$$

6 Numerical Methods

We present the methods targeting (5.3) and (5.4). Although (5.3) is formed by the max function, we can easily use the relationship between the max function and the min function:

$$\max(a, b) = -\min(-a, -b), \quad (a, b) \in \mathbb{R}^2,$$

to transform the max function into the min function. Moreover, to discretize (5.3), we only need to modify the approximation of the derivative $\frac{\partial W}{\partial y}$ in (3.1) by

$$b\frac{\partial W}{\partial y} \backsim \begin{cases} b\frac{W_{j+1}^n - W_j^n}{\Delta y}, & \text{if } b > 0 \\ b\frac{W_j^n - W_j^n - 1}{\Delta y}, & \text{if } b < 0 \end{cases}$$

For other derivatives, we will use the same approximation as in (3.1). Then, we can use the nonsmooth Newton method to calculate the solution.
For (5.4), we consider the case when $\gamma = 0$. Then the nonlinear term is eliminated. We use Howard's algorithm to solve (5.4).
Both methods converge in less than 10 iterations, but the solutions do not prove the conclusion on the non-trade region in Dai and Yi [7].

Comment 6.1. There is another formulation of the portfolio selection problem with transaction costs, provided in Akian et al. [1]. One main differences of the formulation in Akian et al. [1] comparing with the formulation in Dai and Zhong [6] are that the variable y is nonnegative and scaled, that is, $0 \le y \le 1$. The other difference is that the boundary conditions on the bounds of y are different. In Dai and Zhong [6] it is assumed that $\mathbf{L}_1 W = 0$ at the lower bound of y and $\mathbf{M}_1 W = 0$ at the upper bound of y. In Akian et al. [1], it is assumed that $\max(\mathbf{A}_1 W, \mathbf{M}_1 W) = 0$ at $y = 1$ and no specific condition is set at $y = 0$. Due to numerical consideration, similarly, we set $\max(\mathbf{A}_1 W, \mathbf{L}_1 W) = 0$ at $y = 0$.

We also use other methods to solve the portfolio selection problem. We can use the FB-type function (see Appendix E) to transform the problem $\min(F(x), G(x), H(x)) = 0$ into a nonlinear complementarity problem. It takes a long time to run the algorithm and the results do not satisfy the conclusion on the non-trade region in Dai and Yi [7]. Moreover, the results differs significantly when the number of grids changes. Thus this problem is very sensitive to the error brought by the approximation scheme. To achieve higher accuracy, better techniques in solving linear systems are required. In Akian et al. [1], the multigrid method is introduced in solving linear systems. □

7 Conclusion

In this paper, we introduce the basic nonsmooth Newton method with several necessary concepts and convergence theorems. Then we specify the basic nonsmooth Newton method for different complementarity problems and then explore the interconnection between different numerical methods. We see that the concept of the linear Newton approximation scheme and the linear Newton method (Algorithm 3) play a vital role in solving complementarity problems. If the object function G admits a (strong) linear Newton approximation at its zero x^*, the linear Newton method generates a sequence locally converging to x^* Q-linearly (Q-quadratically). Moreover, there is a broad class of locally Lipschitz function, named

semismooth functions, which take their generalized Jacobian as their linear Newton approximation naturally. The class of semismooth functions is so broad that it includes all the functions, i.e, the min and max functions and the FB-type functions. So all the algorithms in this paper are based on or equivalent to the linear Newton method.

References

[1] M. Akian, P. Squier, and A. Sulem. A finite horizon multidimensional portfolio selection problem with singular transactions. *Proceedings of the 34th Conference on Decision and Control, New Orlans*, 3:2193–2198, 1995.

[2] M. Akian, A. Sulem, and M. Taksar. Dynamic optimisation of long term growth rate for a portfolio with transaction costs and logarithmic utility. *Mathematical Finance*, 11(2):153–188, 2001.

[3] Richard Bellman. *Dynamic Programming*. Princeton University Press, 1961.

[4] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 1995.

[5] Olivier Bokanowski, Stefania Maroso, and Hasnaa Zidani. Some convergence results for howard's algorithm. *SIAM J. Numer. Anal*, 47(4):3001–3026, 2009.

[6] M. Dai and Y.F. Zhong. Penalty methods for continuous-time portfolio selection with proportional transaction costs. accepted for publication in Journal of Computational Finance, 2008.

[7] Min Dai and Fahuai Yi. Finite-Horizon Optiomal investment with transaction costs:a Parabolic Double Obstacle problem. *Journal of Differential Equations*, 246(4), 2009.

[8] Min Dai, Yue Kuen Kwok, and Hong You. Intensity-based framework and penalty formulation of optimal stopping problems. *Journal of Economic Dynamics and Control*, 31(12), 2007.

[9] M. H.A. Davis and A. R. Norman. Portfolio selection with transaction costs. *Mathematics of Operations Research*, 15(4):676 – 713, 1990.

[10] Mark H. A. Davis, Vasssilios G.Panas, and Thaleia Zariphopoulou. European option pricing with transaction costs. *SIAM J.Control and Optimization*, 31(2), 1993.

[11] Francisco Facchinei and Jong-Shi Pang. *Finite-dimensional variational inequalities and complementarity problems*. Springer series in operations research. Springer-Verlag New York, Inc., 2003.

[12] P. A. Forsyth and K. R. Vetzal. Quadratic convergence for valuing american options using a penalty method. *SIAM J. Sci. Comput.*, 23(6):2095–2122, 2002.

[13] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[14] Roger A. Horn and Charles R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, New York, 1991.

[15] R.A. Howard. *Dynamic programming and Markov processes*. The Technology Press of the MIT. J. Wiley. Cambridge MA, 1960.

[16] Robert C. Merton. Lifetime portfolio selection under uncertainty: the continuous time case. *The Review of Economics and Statistics*, 51(3):247–257, 1969.

[17] Robert C. Merton. Optimum consumption and portfolio rules in a continuous-time model. *Journal of Economic Theory*, 3:373–413, 1971.

[18] Jong-Shi Pang and Steven A. Gabriel. Ne/sqp: A robust algorithm for the nonlinear complementarity problem. *Mathematical Programming*, 60(1-3):295–337, 1993.

[19] Jong-Shi Pang and Liqun Qi. Nonsmooth equations: Motivation and algorithms. *SIAM J. Optim*, 3 (3):443–465, 1993.

[20] S.E. Shreve and H.M. Soner. Optimal Investment and consumption with transaction costs. *The Annals of Applied Probability*, 4(3), 1994.

[21] R. Zvan, P. A. Forsyth, and K. R. Vetzal. Penalty methods for american options with stochastic volatility. *Journal of Computational and Applied Mathematics*, 91(2):199 – 218, 1998.

A Introduction to the M-matrices[5]

Let M_n be the set of complex $n \times n$ matrices and $M_n(\mathbb{R})$ be the set of real $n \times n$ matrices. The following definiton gives an important class of M-matries.

Definition A.1. Let $A = [a_{ij}] \in M_n$. The matrix A is said to be diagonally dominant if

$$|a_{ii}| \geq \sum_{j=1, j\neq i}^{n} |a_{ij}|, \text{ for all } i = 1, \ldots, n$$

It is said to be strictly diagonally dominant if

$$|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}|, \text{ for all } i = 1, \ldots, n$$

For the strictly diagonally dominant matrices, we have the following theorem.

Theorem A.1. *Let $A = [a_{ij}] \in M_n$ be strictly diagonally dominant. Then*

(a) *A is invertible;*

(b) *If all main diagonal entries of A are positive, then all the eigenvalues of A have positive real part.*

(c) *If A is Hermitian and all main diagonal entries of A are positive, then all the eigenvalues of A are real and positive.*

[5] This is a summary of important definitions and theories in Horn and Johnson [13; 14]

This theorem is a consequence of the Gergorin disc theorem. Moreover, the part (a) is known as the Levy-Desplanques theorem.

To introduce the M-matrices, we start from considering the general notion of inertia for matrices in M_n

Definition A.2. If $A \in M_n$, define:

$i_+(A) \equiv$ the number of eigenvalues of A, counting multiplicities, with positive real part;

$i_-(A) \equiv$ the number of eigenvalues of A, counting multiplicities, with negative real part;

$i_0(A) \equiv$ the number of eigenvalues of A, counting multiplicities, with zero real part.

Then, $i_+(A) + i_-(A) + i_0(A) = n$ and the row vector

$$i(A) \equiv [i_+(A), i_-(A), i_0(A)]$$

is called the inertia of A.

With the notion of the inertia, we give the definition of the positive stable matrix.

Definition A.3. A matrix $A \in M_n$ is said to be positive stable if $i(A) = [n, 0, 0]$, that is, if $i_+(A) = n$.

Next we consider a class of real matrices.

Definition A.4. The set $Z_n \subset M_n(\mathbb{R})$ is defined by

$$Z_n = \{A = [a_{ij}] \in M_n(\mathbb{R}) : a_{ij} \leq 0 \text{ if } i \neq j, i, j = 1, \ldots, n\}.$$

The set of M-matrices is the intersection of two sets defined above. We present the definition of the M-matrix below.

Definition A.5. A matrix A is called an M-matrix if $A \in Z_n$ and A is positive stable.

The next theorem lists some equivalent formulations of the M-matrix.

Theorem A.2. *If $A \in Z_n$, the following statement are equivalen:*

1. *A is positive stable, that is, A is an M-matrix.*
2. *$A = \alpha I - P, P \geq 0, \alpha > \rho(P)$, where $\rho(P)$ is the spectral radius.*
3. *Every real eigenvalue of A is positive.*
4. *$A + rI$ is nonsingular for all $t \geq 0$.*
5. *$A + D$ is nonsingular for every nonnegative diagonal matrix D.*
6. *All principal minors of A are positive.*
7. *The sum of all k-by-k principal minors of A is positive for $k = 1, \ldots, n$.*

8. *The leading principal minors of A are positive.*

9. *$A = LU$, when L is lower triangular and U is upper triangular and all the diagonal entries of each are positive.*

10. *For each nonzero $x \in \mathbb{R}^n$, there is an index $1 \le i \le n$ such that $x_i(Ax)_i > 0$.*

11. *For each nonzero $x \in \mathbb{R}^n$, there is a positive diagonal matrix D such that $x^T ADx > 0$.*

12. *There is a positive vector $x \in \mathbb{R}^n$ with $Ax > 0$.*

13. *The diagonal entries of A are positive and AD is strictly row diagonally dominant for some positive diagonal matrix D.*

14. *The diagonal entries of A are positive and $D^{-1}AD$ is strictly row diagonally dominant for some positive diagonal matrix D.*

15. *The diagonal entries of A are positive and there exist positive diagonal matrices D, E such that DAE is both strictly diagonally dominant.*

16. *There is a positive diagonal matrix D such that $DA + A^T D$ is positive definite.*

17. *A is nonsingular and $A^{-1} \ge 0$.*

18. *$Ax \ge 0$ implies $x \ge 0$.*

B Proof of Proposition 3.1

We notice that when $k \ge 1$, $B^k x^k - c^k \le 0 = B^k x^{k+1} - c^k$. Thus, $B^k(x^{k+1} - x^k) \ge 0$. This implies that $x^{k+1} \ge x^k$. Part (a) is proved.

To prove (b), let $G(x) \equiv \min(A^1 x - b^1, A^2 x - b^2)$. As we mentioned in Section 2, we can view $G(x)$ as a PC^1 function with pieces $\{G^1, \ldots, G^{2^n}\}$. When $k \ge 1$, there exists a number $i \in \{1, \ldots, 2^n\}$ such that $G^i(x^k) = 0$. Since each equation $G^i = 0$ has a unique solution, combining the monotonicity of $\{x^k\}$, $\{x^k\}$ is bounded and thus $\{x^k\}$ is convergent. That is, there exists x^* such that $\{x^k\}$ converges to x^* pointwisely.

The value set $\{x \in \mathbb{R}^m : \exists k \ge 0, x = x^k\}$ is finite and $\{x^k\}$ is monotone. Thus $\{x^k\}$ is finite. Suppose that Howard's algorithm (Algorithm 3) stops at x^K and $G(x^K) \ne 0$. Let us consider the function

$$H(x) = \begin{bmatrix} (A^1_{1.}x - b^1_1) \times (A^2_{1.}x - b^2_1) \\ \vdots \\ (A^1_{m.}x - b^1_m) \times (A^2_{m.}x - b^2_m) \end{bmatrix}.$$

H is obviously continuous and for $k \ge 1$, we have $H(x^k) = 0$. Thus, $H(x^K) = 0$. Then $G(x^K) \le 0$. By Howard's algorithm (Algorithm 3), we can generate $x^{K+1} \ge x^K$. Moreover, $x^{K+1} \ne x^K$. This is a contradiction to the assumption that Howard's algorithm (Algorithm 3) stops at x^K. Due to this contradiction, we have $G(x^K) = 0$. Part(c) is proved. □

C The convergence result of Howard's algorithm

As we metioned before, in the finite dimensional case, the finite termination property is ascertained. In this appendix, we study the convergence rate of Howard's algorithm in the infinite dimensional case. Bokanowski et al. [5] establish the superlinear convergence of Howard's algorithm. We first introduce the notations and assumptions in Bokanowski et al. [5]. Then, some necessary assumptions are added and then the quadratic convergence is gained. Let $\mathcal{X}$ denote the infinite dimensional vector space $\mathbb{R}^{\mathbb{N}^*}$. We define the set of matrices $\mathcal{M}$ as $\mathcal{M} \equiv \mathbb{R}^{\mathbb{N}^*\times\mathbb{N}^*}$. The notation of the index set $\mathbb{I} \equiv \mathbb{N}^*$. We fix two real numbers $p, q \in [1, \infty]$ such that $\frac{1}{p} + \frac{1}{q} = 1$, and the following norms are used:

- For $x \in \mathcal{X}$, $\|x\|_p \equiv \left(\sum_{i\in\mathbb{I}}|x_i|^p\right)^{\frac{1}{p}}$ if $p < \infty$ and $\|x\|_\infty \equiv \sup_{i\in\mathbb{I}}|x_i|$.
- For $A \in \mathcal{M}$, we denote
$$\|A\|_{q,\infty} \equiv \max_{i\in\mathbb{I}}\left(\sum_{j\in\mathbb{I}}|A_{ij}|^q\right)^{\frac{1}{q}} \text{ (if } q < \infty).$$

We also denote
$$\ell^p \equiv \{x \in \mathcal{X} : \|x\|_p < \infty\}$$
and
$$\mathcal{M}_{q,\infty} \equiv \{A \in \mathcal{M} : \|A\|_{q,\infty} < \infty\}.$$

We consider the problem to find $x \in \mathcal{X}$ such that
$$G(x) \equiv \min(A^1x - b^1, A^2x - b^2) = 0,$$
where $A_1, A_2 \in \mathcal{M}_{q,\infty}$ and $b_1, b_2 \in \ell^\infty$. $G(x)$ can also be considered as a composition of smooth functions $Hx - b$, where H is a composition of rows of A_1 and A_2 and b is a composition of rows of b_1 and b_2. We define the set of matrices $\mathcal{H}$ as $\mathcal{H} \equiv \{H \in \mathcal{M}_{q,\infty} : H_{i\cdot} = A^1_{i\cdot} \text{ or } A^2_{i\cdot},\ i \in \mathbb{I}\}$ and $\mathcal{B}$ as $\mathcal{B} \equiv \{b \in \ell^\infty : b_i = b^1_i \text{ or } b^2_i,\ i \in \mathbb{I}\}$. Two assumptions on $H \in \mathcal{H}$ and $b \in \mathcal{B}$ is given in Bokanowski et al. [5]:

(A1) For any $H \in \mathcal{H}$, H is monotone, that is, for any $x \in \mathcal{X}$, $Ax \geq 0$ implies $x \geq 0$. There exists $C \geq 0$ such that for any $H \in \mathcal{H}$ and $y \in \ell^\infty$, there exists $x \in \ell^p$ such that $Hx = y$ and $\|x\|_p \leq C\|y\|_\infty$

(A2) $\epsilon^H_J \equiv \sup_{H\in\mathcal{H}} \max_{i\geq 1}\left(\sum_{|j-i|\geq J}|H_{ij}|^q\right)^{\frac{1}{q}} \xrightarrow{J\to+\infty} 0$, if $q < \infty$;
or $\epsilon^H_J \equiv \sup_{H\in\mathcal{H}} \max_{i\geq 1,|j-i|\geq J}|H_{ij}| \xrightarrow{J\to+\infty} 0$, if $q = \infty$.

With these assumptions, Bokanowski et al. [5] show that the sequence $\{x^k\}$ generated by Howard's algorithm is monotone and converges to a unique x^* in ℓ^p which satisfies $G(x^*) = 0$. Moreover, x^* is a regular point of the function G, that is, $JG(x^*)$ satisfies the following condition: for any $i \in \mathbb{I}$,
$$JG_{i\cdot}(x^*) = \begin{cases} A^1_{i\cdot}, & \text{if } A^1_{i\cdot}x^* - b^1_i < A^2_{i\cdot}x^* - b^2_i; \\ A^1_{i\cdot} \text{ or } A^2_{i\cdot}, & \text{if } A^1_{i\cdot}x^* - b^1_i = A^2_{i\cdot}x^* - b^2_i; \\ A^2_{i\cdot}, & \text{if } A^2_{i\cdot}x^* - b^1_i < A^1_{i\cdot}x^* - b^2_i. \end{cases} \tag{C.1}$$

We notice such $JG(x^*)$ is not unique, and thus we define the set $\mathcal{H}^*$ by

$$\mathcal{H}^* \equiv \{H \in \mathbb{R}^{\mathbb{N}^* \times \mathbb{N}^*} : H \text{ satisfies the condition (C.1)}\}.$$

For the sequence $\{x^k\}$, we define that $d^k \equiv x^{k+1} - x^k$ and $e^k \equiv x^k - x^*$. Then, $d^k \equiv e^{k+1} - e^k$. Let H^k be the matrix we obtain at x^k in Howard's algorithm. The following lemma is an alterative experession of Lemma 3.2 in Bokanowski et al. [5] and a different proof is given.

Lemma C.1. *For any $i \in \mathbb{I}$, the limit*

$$\lim_{k\to\infty} \inf_{H^* \in \mathcal{H}^*} \|H^k_{i\cdot} - H_{i\cdot}\|_q = 0$$

holds.

Proof. For any $i \in \mathbb{I}$, if $A^1_{i\cdot}x^* - b^1_i = A^2_{i\cdot}x^* - b^2_i$, the limit holds trivially. Thus, we only consider the case when $A^1_{i\cdot}x^* - b^1_i \neq A^2_{i\cdot}x^* - b^2_i$. Without loss of generality, we assume that $A^1_{i\cdot}x^* - b^1_i < A^2_{i\cdot}x^* - b^2_i$. Due to the continuity of $A^1_{i\cdot}x - b^1_i$ and $A^2_{i\cdot}x - b^2_i$, we can find a positive number δ such that any point x in the neighborhood of x^*, defined by $\mathbb{B}(x^*, \delta) \equiv \{x \in \mathbb{R}^{\mathbb{N}^*} : \|x - x^*\|_p < \delta\}$, satisfies that $A^1_{i\cdot}x - b^1_i < A^2_{i\cdot}x - b^2_i$. Thus as $k \to \infty$, x^{k+1} falls into the neighborhood $\mathbb{B}(x^*, \delta)$ and $H^k = A^1_{i\cdot}$. □

The above lemma is sufficient to show that the sequence $\{x^k\}$ is superlinearly convergent in the finite dimensional case. In the infinite dimesional case, the componentwise convergence of a sequence of vectors is not sufficient to show the convergence of the sequence of vector. So an convergent assumption on the matrix sequence $\{H^k\}$ is necessary. We give the assumption below.

(A3) Assume that (A1) and (A2) hold. For the sequence $\{H^k\}$, the limit

$$\lim_{k\to\infty} \inf_{H \in \mathcal{H}^*} \|H^k - H\|_{q,\infty} = 0$$

holds.

The following proposition presents the superlinear convergence result of Howard's algorithm.

Propositon C.2. *Assume that (A1),(A2) and (A3) hold. The sequence $\{x^k\}$ generated by Howard's algorithm is superlinearly convergent in the infinite dimensional case if Howard's algorithm is not finite terminated.*

Proof. For any $H \in \mathcal{H}^*$, due to the definition of G, we have the following inequality:

$$G(x^*) + H^k e^k \leq G(x^k) \leq G(x^*) + He^k.$$

Thus we have

$$0 \leq G(x^k) - G(x^*) - H^k e^k \leq (H - H^k)e^k$$

Then due to the definition of $\|\cdot\|_\infty$ and Hölder's inequality, we obtain

$$0 \leq \|G(x^k) - G(x^*) - H^k e^k\|_\infty \leq \|(H - H^k)e^k\|_\infty \leq \|H - H^k\|_{q,\infty}\|e^k\|_p.$$

Since H is arbitrary, we have

$$0 \leq \|G(x^k) - G(x^*) - H^k e^k\|_\infty \leq \inf_{H \in \mathcal{H}^*} \|H - H^k\|_{q,\infty} \|e^k\|_p.$$

With the notations of e^k and d^k, we have

$$G(x^*) = [G(x^k) + H^k d^k] - [G(x^k) - G(x^*) - H^k e^k] - H^k e^{k+1}. \tag{C.2}$$

Taking $G(x^*) = 0$ into (C.2), we have

$$H^k e^{k+1} = [G(x^k) + H^k d^k] - [G(x^k) - G(x^*) - H^k e^k]. \tag{C.3}$$

By (A1), we have

$$\|e^{k+1}\|_p \leq C(\|G(x^k) + H^k d^k\|_\infty + \|G(x^k) - G(x^*) - H^k e^k\|_\infty) \leq C(\|G(x^k) + H^k d^k\|_\infty + \inf_{H \in \mathcal{H}^*} \|H - H^k\|_{q,\infty} \|e^k\|_p). \tag{C.4}$$

Since the sequence $\{x^k\}$ is monotone and Howard's algorithm is not finite terminated, we have $\|e^k\| \neq 0$. We divide both sides of (C.4) by $\|e^k\|$. Then we have

$$\frac{\|e^{k+1}\|_p}{\|e^k\|_p} \leq C\left(\frac{\|G(x^k) + H^k d^k\|_\infty}{\|e^k\|_p} + \inf_{H \in \mathcal{H}^*} \|H - H^k\|_{q,\infty}\right). \tag{C.5}$$

Due to the description of Howard's algorithm, we have $\|G(x^k) + H^k d^k\|_\infty = 0$ for all $k \geq 1$. Combining with (A3), we have

$$\lim_{k \to \infty} \frac{\|e^{k+1}\|_p}{\|e^k\|_p} = 0$$

So we show that the sequence $\{x^k\}$ is superlinear convergent. □

If the following assumption on the matrices A^1 and A^2 is satisfied, we can prove that Howard's algorithm is finite terminated.

(A4) There exists a positive real number δ_0 such that for any $i \in \mathbb{I}$, if $A^1_{i\cdot} \neq A^2_{i\cdot}$, we have $\|A^1_{i\cdot} - A^2_{i\cdot}\|_q > \delta_0$.

Propositon C.3. *Assume (A1), (A2), (A3) and (A4) hold. There exists a number K such that for any $k > K$, $x^k = x^*$.*

Proof. We notice that due to the description of Howard's algorithm, we have $\|G(x^k) + H^k d^k\|_\infty = 0$ for all $k \geq 1$. Moreover, by (A3), we have there exists a number K such that for any number $k \geq K$, $\inf_{H \in \mathcal{H}^*} \|H - H^k\|_{q,\infty} < \delta_0$. For any two matrices $H_1, H_2 \in \mathcal{H}$, we have if $H_1 \neq H_2$, $\|H_1 - H_2\|_{q,\infty} > \delta_0$ by (A4). Thus we have there exists a number K such that for any number $k > K$, $\inf_{H \in \mathcal{H}^*} \|H - H^k\|_{q,\infty} = 0$. Then by (C.4), we obtain that $\|e^{k+1}\| = 0$ for all $k > K$. So $x^k = x^*$ for all $k > K$. □

D Proof of Proposition 4.1

To prove this proposition, the key is to prove that $\{x^k\}$ is monotone, that is, $x^{k+1} \geq x^k$ for all $k \geq 1$. We are going to find a monotone matrix A such that $A(x^{k+1} - x^k) \geq 0$.

Suppose $i \in \{1, \ldots, n\}$ and $k \geq 1$. If $A^3_{i\cdot}x^{k+1} - b^3_i = 0$, we have $A^3_{i\cdot}x^{k+1} - b^3_i \geq A^3_{i\cdot}x^k - b^3_i$. We choose $A_{i\cdot} = A^3_{i\cdot}$. If $\max(A^1_{i\cdot}x^{k+1} - b^1_i, A^1_{i\cdot}x^{k+1} - b^2_i) = 0 < A^3_{i\cdot}x^{k+1} - b^3_i$, then according to the definition of B^k in each iterate, we have $\max(A^1_{i\cdot}x^k - b^1_i, A^1_{i\cdot}x^k - b^2_i) \leq 0$. Suppose $\max(A^1_{i\cdot}x^k - b^1_i, A^1_{i\cdot}x^k - b^2_i) < 0$. If $A^1_{i\cdot}x^{k+1} - b^1_i = 0$, we choose $A_{i\cdot} = A^1_{i\cdot}$. Otherwise, we choose $A_{i\cdot} = A^2_{i\cdot}$.

Suppose $\max(A^1_{i\cdot}x^k - b^1_i, A^1_{i\cdot}x^k - b^2_i) = 0$. If $\max(A^1_{i\cdot}x^{k+1} - b^1_i, A^1_{i\cdot}x^{k+1} - b^2_i) = 0$, then we divide it into two cases:

(a) if $0 = A^1_{i\cdot}x^k - b^1_i \geq A^2_{i\cdot}x^k - b^2_i (A^1_{i\cdot}x^k - b^1_i < A^2_{i\cdot}x^k - b^2_i = 0)$ and $0 = A^1_{i\cdot}x^{k+1} - b^1_i \geq A^2_{i\cdot}x^{k+1} - b^2_i (A^1_{i\cdot}x^{k+1} - b^1_i < A^2_{i\cdot}x^{k+1} - b^2_i = 0)$, we simply choose $A_{i\cdot} = A^1_{i\cdot} (A_{i\cdot} = A^2_{i\cdot})$.

(b) if $0 = A^1_{i\cdot}x^k - b^1_i \geq A^2_{i\cdot}x^k - b^2_i (A^1_{i\cdot}x^k - b^1_i < A^2_{i\cdot}x^k - b^2_i = 0)$ and $0 = A^1_{i\cdot}x^{k+1} - b^1_i \leq A^2_{i\cdot}x^{k+1} - b^2_i (A^1_{i\cdot}x^{k+1} - b^1_i > A^2_{i\cdot}x^{k+1} - b^2_i = 0)$, we simply choose $A_{i\cdot} = A^2_{i\cdot} (A_{i\cdot} = A^1_{i\cdot})$.

The matrix A we construct is monotone since A^1, A^2 and A^2 are strongly row diagonal dominant M-matrices. Moreover, $A(x^{k+1} - x^k) \geq 0$. Thus, we prove that $x^{k+1} \geq x^k$ for all $k \geq 1$. As the proof in B, part (b) and (c) is a consequence of the monotonicity and the fact that $\min(\max(A^1_{i\cdot}x^k - b^1_i, A^1_{i\cdot}x^k - b^2_i), A^3_{i\cdot}x^k - b^3_i)$ is a PC^1 function with 3^n pieces. □

E Two FB-type functions

Propositon E.1. *Let $(a, b) \in \mathbb{R}^2$. Two FB-type functions are defined as follows:*

$$F(a, b) = a + b - \sqrt{a^2 + b^2}; \tag{E.1}$$

$$F(a, b) = a + b + \sqrt{a^2 + b^2}; \tag{E.2}$$

F and G are equivalent to the min and max functions in the sense that

$$\min(a, b) \geq 0 \Leftrightarrow F(a, b) \geq 0; \min(a, b) < 0 \Leftrightarrow F(a, b) < 0; \tag{E.3}$$

$$\max(a, b) \geq 0 \Leftrightarrow G(a, b) \geq 0; \max(a, b) < 0 \Leftrightarrow G(a, b) < 0; \tag{E.4}$$

Proof. We only need to show that (C.3) holds, since $\max(a, b) = -\min(-a, -b)$ and $G(a, b) = -F(-a, -b)$.

Suppose $\min(a, b) > 0$. Then $a > 0$ and $b > 0$. Since $(a + b)^2 - (a^2 + b^2) = (a + b - \sqrt{a^2 + b^2})(a + b + \sqrt{a^2 + b^2}) = 2ab > 0$ and $(a + b + \sqrt{a^2 + b^2}) > 0$, we have $F(a, b) = a + b - \sqrt{a^2 + b^2} > 0$.

Suppose $\min(a, b) < 0$. Then at least one of a and b is negative. Pick $a < 0$. If $b < 0$, it is trivial that $F(a, b) < 0$. If $b \geq 0$, then $b < \sqrt{a^2 + b^2}$. We obtain $F(a, b) = a + (b - \sqrt{a^2 + b^2}) < 0$.

Suppose $\min(a,b)=0$. If $a=0$, then $b\geq 0$. $F(a,b)=b-\sqrt{b^2}=0$. It is similar for the case when $b=0$.

Suppose $F(a,b)>0$. $a+b>\sqrt{a^2+b^2}\geq 0$. Then $ab>0$.Thus $a>0$ and $b>0$. So $\min(a,b)>0$.

Suppose $F(a,b)=0$. $ab=0$. If $a=0$, then $b=\sqrt{b^2}\geq 0$. It is similar for the case when $b=0$. So $\min(a,b)=0$.

Suppose $F(a,b)<0$. There must exist a or b is negative. If not, $F(a,b)\geq 0$ according to the previous analysis. This is a contradiction. So we have $\min(a,b)<0$. □

F Differentiability of the merit function θ

The key problem is to show that the merit function $\theta:\mathbb{R}^3\to\mathbb{R}$ is continuously differentiable. Define θ by

$$\theta(a,b,c)=\left(\sqrt{\varphi^2(a,b)+c^2}-c-\varphi(a,b)\right)^2$$

where $\varphi(a,b)=\sqrt{a^2+b^2}+a+b$. It easy to see that $\varphi(-a,-b)$ is a FB function. Thus, as we explain befog, φ^2 is continuously differential everywhere and $\nabla\varphi^2(0,0)=(0,0)$. It is trivial that θ is continuously differentiable everywhere except the line $a=b=0$. Obviously,

$$\theta(a,b,c)=2\varphi^2(a,b)+2c^2+2\varphi(a,b)c-2(c+\varphi(a,b))\sqrt{\varphi^2(a,b)+c^2}.$$

Thus since $\varphi(a,b)^2+c^2$ is continuously differentiable, we only need to show that $\psi(a,b,c)\equiv\varphi(a,b)c-(c+\varphi(a,b))\sqrt{\varphi^2(a,b)+c^2}$ is continuously differentiable.
Now we consider the origin $(a,b,c)=(0,0,0)$. Let the direction $d=(h\cos\beta\sin\alpha,h\sin\beta\sin\alpha,h\cos\alpha)$, where $h>0,\beta\in[0,2\pi)$ and $\alpha\in[0,\pi)$. Thus,

$$\begin{aligned}\frac{\psi(d)-\psi(0,0,0)}{h}&=\frac{\psi(d)}{h}\\&=\frac{h^2\sin\alpha\cos\alpha(1+\sin\beta+\cos\beta)}{h}\\&\quad-\frac{h^2(\sin\alpha+\sin\alpha\sin\beta+\sin\alpha\cos\theta+\cos\alpha)\sqrt{(\sin\alpha+\sin\alpha\sin\beta+\sin\alpha\cos\beta)^2+\cos^2\alpha}}{h}\\&=h\sin\alpha\cos\alpha(1+\sin\beta+\cos\beta)\\&\quad-h(\sin\alpha+\sin\alpha\sin\beta+\sin\alpha\cos\beta+\cos\alpha)\sqrt{(\sin\alpha+\sin\alpha\sin\beta+\sin\alpha\cos\beta)^2+\cos^2\alpha}.\end{aligned}$$

As $h\to 0$, we have $\frac{\psi(d)-\psi(0,0,0)}{h}\to 0$. Now we are going to prove the differentiability of θ when $a=b=0$ and $c\neq 0$. Let the direction $d=(d_a,d_b,d_c)$. Since

$$|\psi(d_a,d_b,c+d_c)-\psi(0,0,c)|\leq|\psi(d_a,d_b,c+d_c)-\psi(d_a,d_b,c)|+|\psi(d_a,d_b,c)-\psi(0,0,c)|$$

and

$$\lim_{\|d\|\to 0}\frac{|\psi(d_a,d_b,c+d_c)-\psi(d_a,d_b,c)|}{\|d\|}=0,$$

we only need to consider the limit of

$$\frac{\psi(d_a, d_b, c) - \psi(0, 0, c)}{\sqrt{d_a^2 + d_b^2}}$$

Assume the direction is $d = (h \sin\beta, h \cos\beta, 0)$, where $h > 0$ and $\beta \in [0, 2\pi)$. We notice that

$$\begin{aligned}\psi(\varphi, c) &= \varphi c - (c + \varphi)\sqrt{\varphi^2 + c^2} \\ &= -c\sqrt{\varphi^2 + c^2} + \varphi\Big(c - \sqrt{\varphi^2 + c^2}\Big)\end{aligned}$$

If $c > 0$, we have $-c\sqrt{\varphi^2 + c^2}$ is differentiable due to the differentiability of φ^2 and

$$\varphi(c - \sqrt{\varphi^2 + c^2}) = \frac{\varphi^3}{c + \sqrt{\varphi^2 + c^2}}$$

is differentiable since

$$\frac{\varphi'(a, b)}{c + \sqrt{\varphi^2(a, b) + c^2}}$$

is bounded when $(a, b) \to (0, 0)$ and φ^2 is differentiable. So θ is continuously differentiable everywhere except the half line $\{(a, b, c) \in \mathbb{R}^3 : a = b = 0, c < 0\}$.

Initial values	U^0	$-10\mathbf{1}_{101\times1}$	$50\mathbf{1}_{101\times1}$	$100\mathbf{1}_{101\times1}$	$200\mathbf{1}_{101\times1}$
Number of iterations	11	3	12	12	12

Table 1: Numbers of iterations comparing with initial values using Howard's algorithm. The parameter values used in calculation are: $K = 100; \sigma = 1, r = 0.1, T = 1, S_{\max} = 400, N_s = 100, N_t = 20$.

Numbers of spatial grids(N_s)	50	100	200	300	400
Number of iterations	6	11	21	32	42

Table 2: Numbers of iterations comparing with Numbers of spatial grids using Howard's algorithm. The parameter values used in calculation are: $K = 100; \sigma = 1, r = 0.1, T = 1, S_{\max} = 400, N_t = 20$ and the initial value U^0.

Initial values	U^0	$-10\mathbf{1}_{101\times1}$	$50\mathbf{1}_{101\times1}$	$100\mathbf{1}_{101\times1}$	$1000\mathbf{1}_{101\times1}$
Number of iterations	3	3	3	3	2

Table 3: Numbers of iterations comparing with initial values using the penalty method. The parameter values used in calculation are: $K = 100; \sigma = 1, r = 0.1, T = 1, S_{\max} = 400, N_s = 100, N_t = 20, \rho = 10000000$.

Numbers of spatial grids(N_s)	50	100	200	300	400
Number of iterations	3	3	3	3	3

Table 4: Numbers of iterations comparing with Numbers of spatial grids using the penalty method. The parameter values used in calculation are: $K = 100; \sigma = 1, r = 0.1, T = 1, S_{\max} = 400, N_t = 20, \rho = 10000000$ and the initial value U^0.

Initial values	U^0	$\mathbf{0}_{101\times1}$	$101\mathbf{1}_{101\times1}$	$1001\mathbf{1}_{101\times1}$	$-101\mathbf{1}_{101\times1}$
Number of iterations	14	11	8	> 2000	> 2000

Table 5: Numbers of iterations comparing with initial values using the FB line search algorithm. The parameter values used in calculation are: $K = 100; \sigma = 1, r = 0.1, T = 1, S_{\max} = 400, N_s = 100, N_t = 20, \rho = 10000000$.

Numbers of spatial grids(N_s)	50	100	200	300	400
Number of iterations	32361	14	24	34	44

Table 6: Numbers of iterations comparing with Numbers of spatial grids using the FB line search algorithm. The parameter values used in calculation are: $K = 100; \sigma = 1, r = 0.1, T = 1, S_{\max} = 400, N_t = 20$ and the initial value U^0.

Initial values	U^0	$-1001\mathbf{1}_{101\times1}$	$\mathbf{0}_{101\times1}$	$1001\mathbf{1}_{101\times1}$	$10001\mathbf{1}_{101\times1}$
Number of iterations	14	6	6	16	16

Table 7: Numbers of iterations comparing with initial values using Howard's method. The parameter values used in calculation are: $K = 100, r = 0.02, q = 0.04, \sigma = 0.3, T = 2, R = 30, S_{\max} = 200, N_t = 20, N_s = 100, \rho = 10000$.

Numbers of spatial grids(N_s)	50	100	200	300	400
Number of iterations	7	14	26	39	51

Table 8: Numbers of iterations comparing with Numbers of spatial grids using Howard's algorithm. The parameter values used in calculation are: $K = 100, r = 0.02, q = 0.04, \sigma = 0.3, T = 2, R = 30, S_{\max} = 200, N_t = 20, \rho = 10000$ and the initial value U^0.

Initial values	U^0	$-101\mathbf{1}_{101\times1}$	$\mathbf{0}_{101\times1}$	$501\mathbf{1}_{101\times1}$	$1001\mathbf{1}_{101\times1}$	$2001\mathbf{1}_{101\times1}$
Number of iterations	12	13	12	44	14	14

Table 9: Numbers of iterations comparing with initial values using min-FB line search algorithm. The parameter values used in calculation are: $K = 100, r = 0.02, q = 0.04, \sigma = 0.3, T = 2, R = 30, S_{\max} = 200, N_t = 20, N_s = 100, \rho = 10000$.

Numbers of spatial grids(N_s)	50	100	200
Number of iterations	7	12	21

Table 10: Numbers of iterations comparing with Numbers of spatial grids using min-FB line search algorithm. The parameter values used in calculation are: $K = 100, r = 0.02, q = 0.04, \sigma = 0.3, T = 2, R = 30, S_{\max} = 200, N_t = 20, \rho = 10000$ and the initial value U^0.

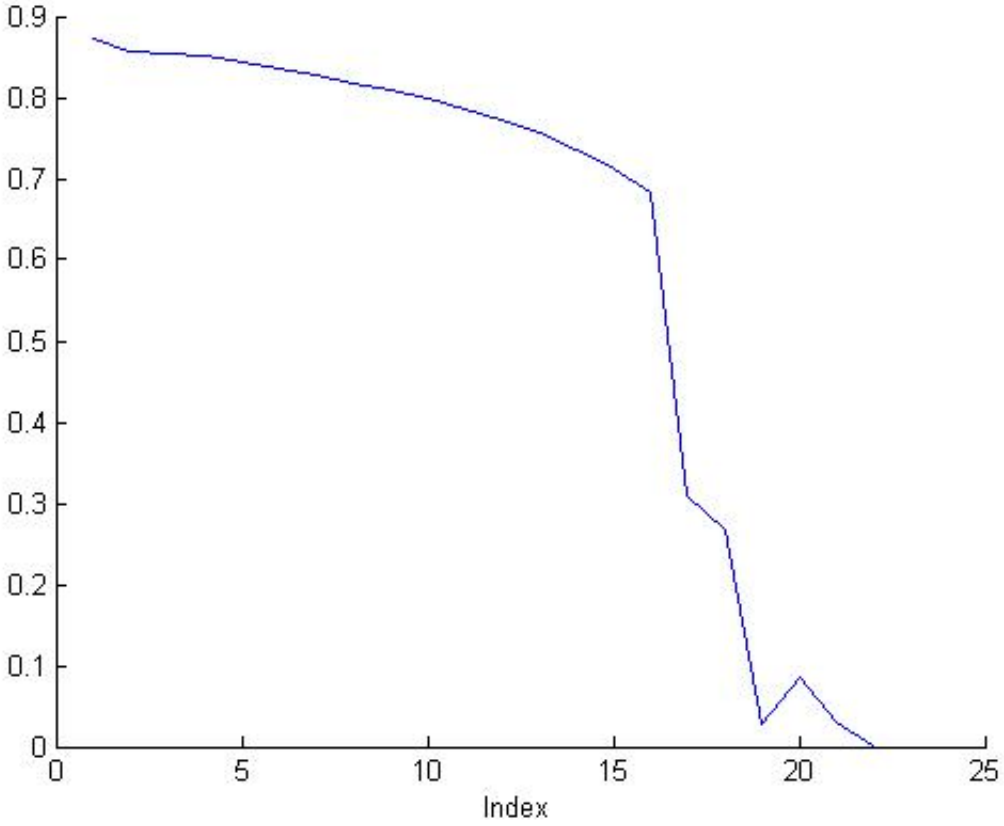

Figure 1: Plot of the values $\frac{\|x^{k+1}-x^{\infty}\|}{\|x^{k}-x^{\infty}\|}$ The parameters used in the calculation are: $K = 100; \sigma = 1, r = 0.1, T = 1, S_{\max} = 400, N_s = 200, N_t = 20$.

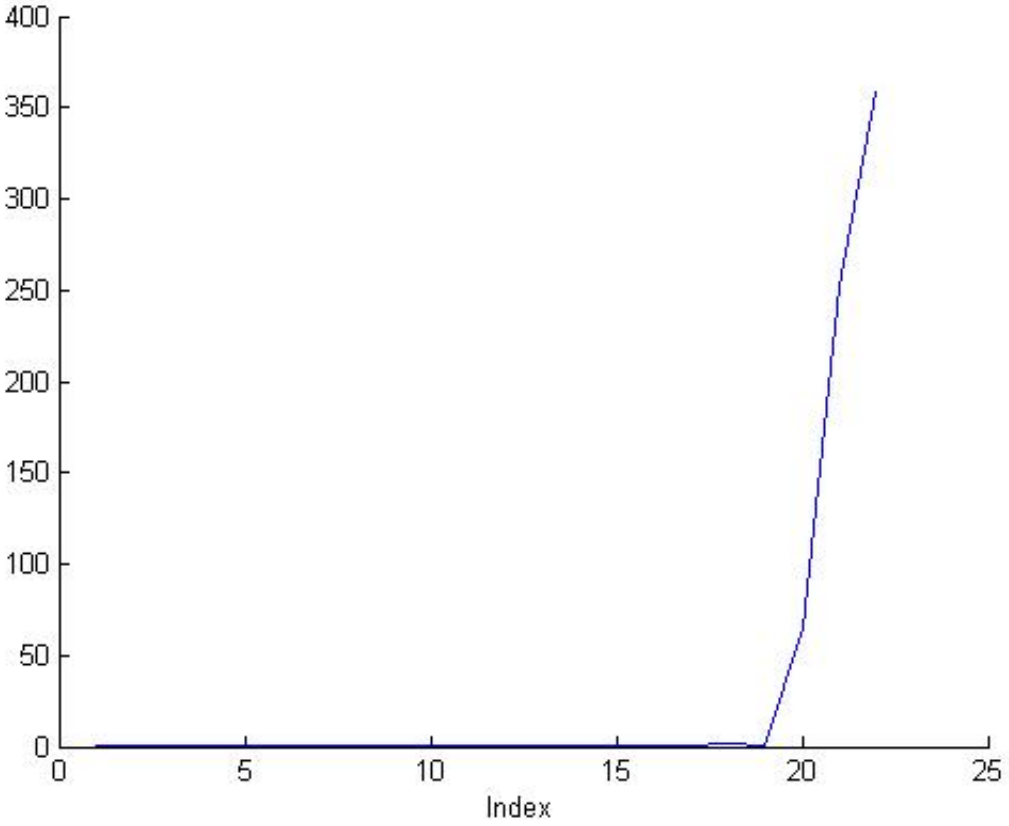

Figure 2: Plot of the values $\frac{\|x^{k+1}-x^{\infty}\|}{\|x^{k}-x^{\infty}\|^{2}n}$ The parameters used in the calculation are: $K = 100; \sigma = 1, r = 0.1, T = 1, S_{\max} = 400, N_s = 200, N_t = 20$.

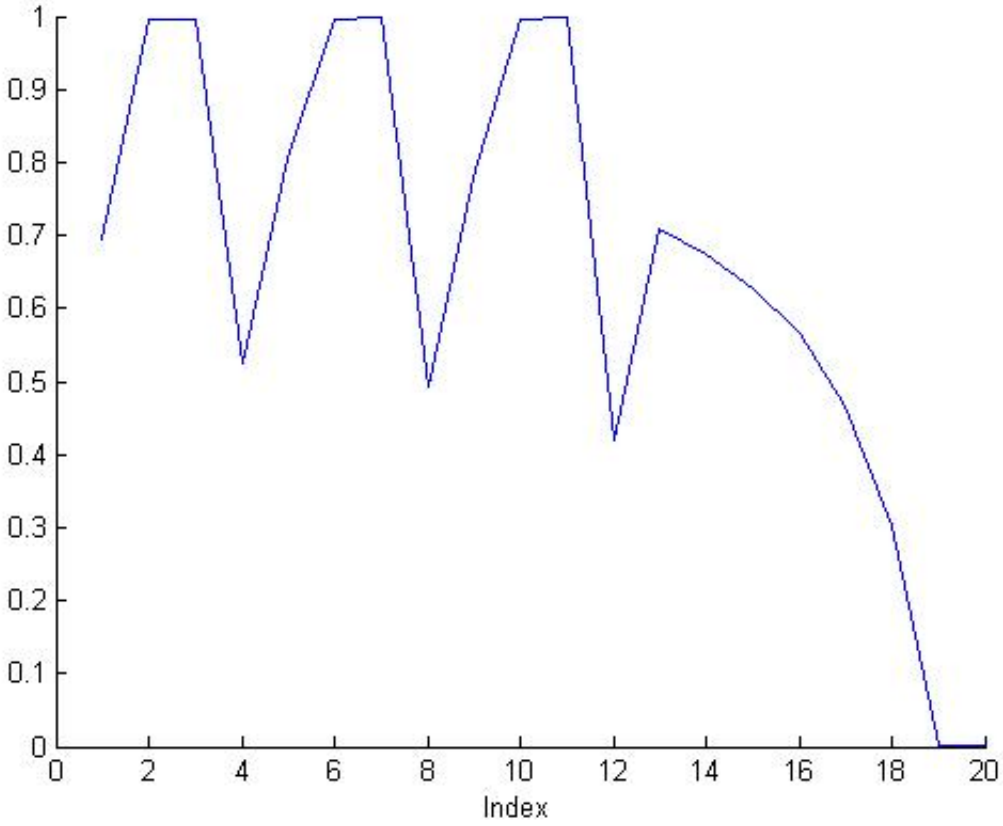

Figure 3: Plot of the values $\frac{\|x^{k+1}-x^{\infty}\|}{\|x^{k}-x^{\infty}\|}$ The parameters used in the calculation are: $K = 100, r = 0.02, q = 0.04, \sigma = 0.3, T = 2, R = 30, S_{\max} = 200, N_t = 20, N_s = 200, \rho = 10000$

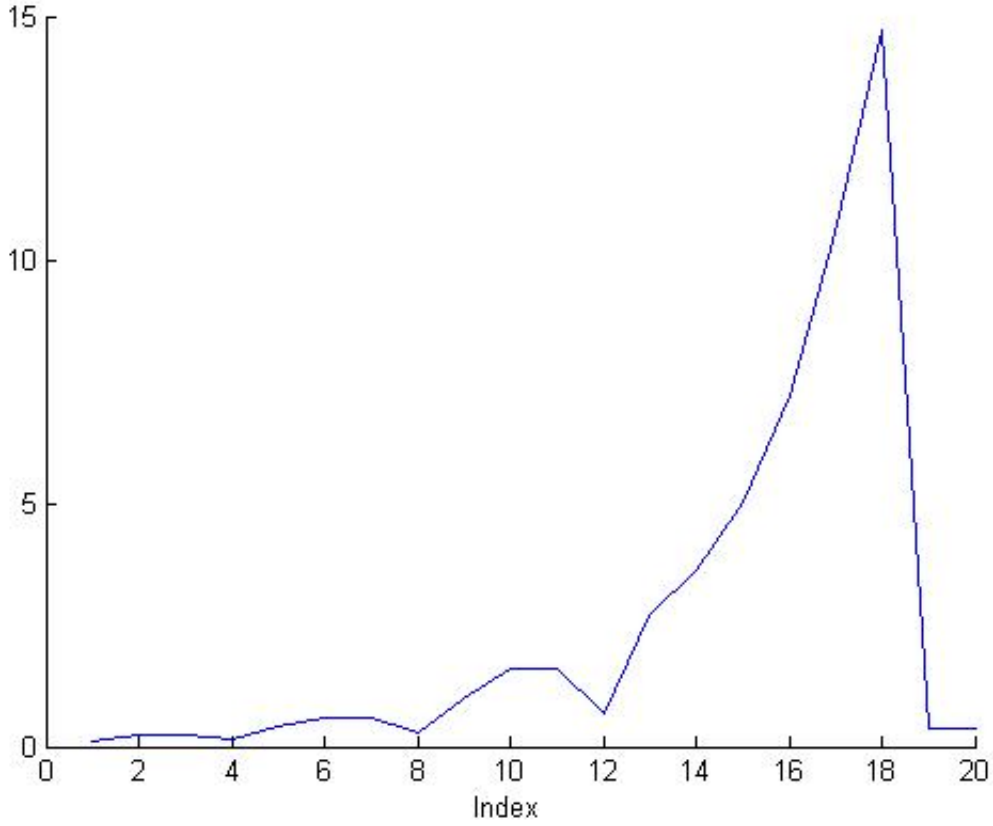

Figure 4: Plot of the values $\frac{\|x^{k+1}-x^{\infty}\|}{\|x^{k}-x^{\infty}\|^2}$ The parameters used in the calculation are: $K = 100, r = 0.02, q = 0.04, \sigma = 0.3, T = 2, R = 30, S_{\max} = 200, N_t = 20, N_s = 200, \rho = 10000$