2

Studies
on
Optimization Based Solution Methods for Variational Inequality Problems

Kouichi TAJI

STUDIES
ON
OPTIMIZATION BASED SOLUTION METHODS FOR VARIATIONAL INEQUALITY PROBLEMS

Kouichi TAJI

Submitted in partial fulfillment of
the requirement for the degree of
DOCTOR OF ENGINEERING
(Applied Mathematics and Physics)

KYOTO UNIVERSITY
KYOTO 606-01, JAPAN

Preface

In the last fifteen years, the variational inequality problem has been used to study and formulate various equilibrium problems arising in engineering, economics, operations research, transportation and regional sciences. The variational inequality problem was originally introduced by G. Stampacchia in the middle of 1960's to formulate and study partial differential equations, and hence, the early work was mainly focused on infinite-dimensional variational inequality problems.

Studies on finite-dimensional variational inequality problems started in 1979, when M. J. Smith formulated an equilibrium condition for traffic assignment problem in the form of variational inequalities. Since then, study of the variational inequality problem has become active and large progress has been made from both theoretical and practical point of views.

Among various research subjects, it is important to construct solution methods for the variational inequality problem. It is well known that the variational inequality problem is a generalization of a system of nonlinear equations and the nonlinear complementarity problem. So it is natural that various iterative algorithms, such as projection methods, linearized Jacobi method, successive over-relaxation method and

Newton's method have been developed as generalizations of iterative algorithms for nonlinear equations and nonlinear complementarity problems, and their convergence theorems have been established.

On the other hand, another approach for solving variational inequality problems has recently attracted much attention. This approach exploits various merit functions for the variational inequality problem. The purpose of introducing a merit function is to formulate a variational inequality problem as an equivalent optimization problem, and hence, many descent methods proposed for nonlinear programming problems are applicable.

Our studies focuses on the optimization formulation of the variational inequality problem. One of the main aims of this thesis is to develop both theoretically and practically efficient algorithms for the variational inequality problem, which is based on an equivalent optimization formulation. Another aim of this thesis is to construct a new merit function, which is particularly useful to deal with variational inequality problems with general nonlinear constraints.

Importance of the optimization formulation approach to the variational inequality problem is now being recognized. The author hopes that the results obtained in this thesis will help further improve the study in this field.

Kouichi Taji

Acknowledgments

The author is sincerely grateful to Professor Toshihide Ibaraki of Kyoto University for his continual encouragement and invaluable suggestions ever since my student days.

The author is deeply indebted to Professor Masao Fukushima of Kyoto University for his enthusiastic guidance, discussion and persistent encouragement. He would also like to thank him for his careful reading of this manuscript and accurate comments.

Thanks are due to Associate Professor Shigeru Masuyama of Toyohashi University of Technology, Associate Professor Masamitsu Ohnishi of Tohoku University and Associate Professor Hiroshi Nagamochi of Kyoto University for their helpful comments and suggestions. Furthermore, the author is grateful to all his friends and colleagues in Professor Fukushima's laboratory and in Professor Ibaraki's laboratory.

Finally, but not least, the author would like to express highest gratitude to his family for their heartfelt cooperation and encouragement.

Contents

List of Tables

List of Figures

Notation

We provide nonations that will be frequently used in this thesis. For other mathematical concepts and definitions, see Appendix.

We denote by R^n the real n-dimensional Euclidian space. Throughout the thesis, every vector is a *column* vector, i.e. $x \in R^n$ represents the n-dimensional column vector

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}.$$

For a mapping $F : R^n \rightarrow R^n$, $F(x)$ is also considered an n-dimensional column vector whose i-th component $F_i(x)$, where F_i is a function from R^n into R. The *transpose* of an $m \times n$ matrix A and a vector x is denoted by A^t and x^t, respectively. We often write $x \geq 0$ if $x_i \geq 0$ for all $i = 1, \cdots, n$. The *nonnegative orthant* of R^n, denoted by R^n_+, is the set of vectors $x \in R^n$ such that $x \geq 0$. We denote by e_i the i-th unit vector such that the i-th element of e_i is 1 and the other elements are zero. E_j represents the $j \times j$ *identity* matrix.

The symbol $\langle \cdot, \cdot \rangle$ denotes the *inner product* in R^n defined by

$$\langle x, y \rangle = \sum_{i=1}^{n} x_i y_i.$$

$\|\cdot\|$ denotes the *Euclidean norm* in R^n defined by

$$\| x \| = \langle x, x\rangle^{\frac{1}{2}}.$$

We often use the *G-norm* in R^n, defined by

$$\| x \|_G = \langle x, Gx\rangle^{\frac{1}{2}}, \tag{0.1}$$

for an $n \times n$ symmetric positive definite matrix G. The norm of an $n \times n$ matrix A, also denoted by $\| A \|$, is defined by

$$\| A \| = \min_{x \neq 0} \frac{\| Ax \|}{\| x \|}.$$

The *gradient* of a function $\phi : R^n \to R$ at $x \in R^n$ is defined to be the column vector

$$\nabla\phi(x) = \begin{pmatrix} \frac{\partial\phi(x)}{\partial x_1} \\ \vdots \\ \frac{\partial\phi(x)}{\partial x_n} \end{pmatrix},$$

where $\dfrac{\partial\phi(x)}{\partial x_i}$ denotes a partial derivative. The *Hessian* of ϕ at $x \in R^n$, denoted by $\nabla^2\phi(x)$, is the $n \times n$ symmetric matrix whose (i,j)-th component is $\dfrac{\partial^2\phi(x)}{\partial x_i \partial x_j}$. The *Jacobian* of the mapping $F : R^n \to R^n$ at $x \in R^n$, denoted by $\nabla F(x)$, is the $n \times n$ matrix defined by

$$\nabla F(x) = (\nabla F_1(x), \ldots, \nabla F_n(x)).$$

Generally x^* denotes a solution to the problem under consideration and $\{x^k\}$ denotes a sequence generated by algorithms, where the superscript k represents the k-th iterate. In particular, x^0 denotes an initial iterate.

Chapter 1

Introduction

1.1 Historical background on study for the variational inequality problem

In the last fifteen years, the variational inequality problem has been widely used to formulate and study various equilibrium models arising in engineering, economics and operations research. The finite dimensional variational inequality problem is to find a vector $x^* \in S$ such that

$$\langle F(x^*), x - x^*\rangle \geq 0 \quad \text{for all} \quad x \in S, \tag{1.1}$$

where the set S is a nonempty closed convex subset of R^n and the mapping F is a continuous mapping from R^n into R^n.

The history of the variational inequality problem dates back to the work of Stampacchia et al. [HaS66, LiS67], who formulated partial differential equations as variational inequality problems. As the early studies were tied with boundary value problems, Stepfan problem and fluid dynamics, the attention was mainly paid to infinite dimensional variational inequality problems, i.e., the set S in (1.1) is replaced by a closed convex subset of a Hilbert space V and F represents the mapping from V into its dual. The book of Kinderlehrer and Stampacchia [KiS80] provides many references and applications concerning infinite dimensional variational inequality problems.

On the other hand, the finite dimensional variational inequality problem may be viewed as a generalization of systems of nonlinear equations, convex programming problems and complementarity problems. Marcino and Stampacchia [MaS72] investigated the relation between finite dimensional variational inequality problems and convex programming problems. They also developed a computational method for solving finite dimensional variational inequality problems. Karamardian [Kar71, Kar72] showed that the variational inequality problem includes complementarity problems which arise in various fields such as quadratic programming, game theory and economic equilibria [Kar69a, Kar69b]. He derived some existence results for the nonlinear complementarity problem from those for the variational inequality problem [Kar71]. Before the 1980's, in spite of these, studies on the finite dimensional variational inequality problem were not so active, compared with those for convex programming and complementarity problems.

In the 1980's, the study on the variational inequality problem in a finite dimensional space became more popular and attracted much attention in connection with various equilibrium problems arising in engineering, economic and operations research. In 1979, Smith [Smi79] presented a formulation of an equilibrium condition for the traffic equilibrium problem. Dafermos [Daf80] first pointed out that Smith's formulation is a finite dimensional variational inequality problem. In the same paper,

Dafermos also presented a solution method, which belongs to the class of projection methods, for the variational inequality problem. Since then, finite dimensional variational inequality problems have been used to formulate and study various equilibrium problems, such as traffic assignment problems [AaM82, BeG82, Nag93], spatial price equilibrium problems [Har84, NaA88, Tob88], Walrasian equilibrium problems [Mat87], Nash-Cournot production problems [MSS82], Nash price equilibrium problems [CDH90] and other equilibrium problems [Flo89, Nag87, NaA89]. Also, many important results on algorithms [Daf83, HaP90, PaC82], sensitivity analysis [Daf88, DaN84, Kyp87, Kyp90, QiM89, Tob86] and generalizations of the problem [ChP82, FaP82, Fuk85] have been investigated.

Throughout this thesis, we focus on the finite dimensional variational inequality problem; hence we shall simply call it the variational inequality problem.

1.2 Merit functions for the variational inequality problem

The variational inequality problem can be regarded as a generalization of a system of nonlinear equations and the nonlinear complementarity problem. So it is natural to generalize various iterative algorithms developed for a system of nonlinear equations and the nonlinear complementarity problem to the variational inequality problem. Such generalized algorithms include projection methods [Aus76, BeG82, Fuk86], linearized Jacobi method [PaC82], nonlinear Jacobi method [FlS82, PaC82], successive over-relaxation method [Pan85], Newton's method [Jos79a, PaC82], a quasi-Newton method [Jos79b] and generalized descent methods [HaM87, IFI88, Smi84]. Iterative algorithms and their convergence properties are summarized in [Daf83, HaP90, PaC82].

Recently, another approach that, by introducing a merit function, reformulates the variational inequality problem as an optimization problem has attracted much attention. The function $\varphi : \Omega \to R \cup \{+\infty\}$, where Ω is subset of R^n such that $S \subseteq \Omega$, is said to be a *merit function* for the variational inequality problem (1.1) when φ has the following property:

(a) $\varphi(x) > 0$ for all $x \in \Omega$.

(b) $\varphi(x) = 0$ if and only if x is a solution of (1.1).

A merit function enable us to formulate an equivalent optimization problem for a variational inequality problem:

$$\begin{array}{ll} \text{minimize} & \varphi(x) \\ \text{subject to} & x \in \Omega. \end{array}$$

One of the advantages of introducing a merit function is that many descent methods proposed for nonlinear programming problems become applicable.

Various merit functions for the variational inequality problem have been proposed and studied. In 1976, Auslender [Aus76] introduced the so-called *gap function* $g : S \to R \cup \{+\infty\}$ for the variational inequality problem, defined by

$$g(x) = \sup_y \{\langle F(x), x - y \rangle \mid y \in S\}. \tag{1.2}$$

(The name 'gap function' was first used by Hearn [Hea82] in studying the duality gap of convex programming problems.) It can be seen that the gap function has the above properties (a) and (b) with $\Omega = S$. So by using the gap function, the variational inequality problem can be formulated as an optimization problem:

$$\text{minimize } g(x) \text{ subject to } x \in S. \tag{1.3}$$

Note that the function g may be infinite-valued when the set S is unbounded. Auslender [Aus76] showed that, when the constraint set is bounded and strongly convex, the gap function is everywhere differentiable. Moreover he proposed a descent method which uses the derivatives of the gap function. But the assumption of strong convexity is too restrictive and, for example, excludes the case in which the constraint set is a polyhedral convex set. Hearn, Lawphongpanich and Nguyen [HLN84] discussed the convexity of the gap function. Based on the gap function, Marcotte [Mar85] proposed another descent method for monotone variational inequality problems, and Marcotte and Dussault [MaD87] proposed a globally convergent modification of Newton's method.

For monotone variational inequality problems, Hearn and Nguyen [HeN82] introduced the *dual gap function* $\hat{g} : S \to R \cup \{+\infty\}$, defined by

$$\hat{g}(x) = \max_y \{\langle F(y), x - y\rangle \mid y \in S\}. \tag{1.4}$$

Since the dual gap function is defined as the pointwise maximum of a set of linear functions, the dual gap function leads to an equivalent *convex* minimization problem. Based on this formulation, Nguyen and Dupuis [NgD84] proposed a solution method which is closely related to the cutting plane algorithm in nonlinear programming [Zan69].

The gap function and the dual gap function are in general non-differentiable. It has been well known that, under the symmetry assumption, an equivalent differentiable optimization formulation of the variational inequality problem exists [HaP90]. Whether or not there exists an equivalent differentiable optimization formulation for general asymmetric variational inequality problems had been open for a long time. Recently by introducing the *regularized gap function*, Fukushima [Fuk92] solved this question affirmatively. The regularized gap function is defined by

$$f_S(x) = \max_y \left\{ -\langle F(x), y - x\rangle - \frac{1}{2}\langle y - x, G(y - x)\rangle \;\middle|\; y \in S \right\}, \tag{1.5}$$

where G is an $n \times n$ symmetric positive definite matrix. Fukushima [Fuk92] showed that the regularized gap function has properties (a) and (b) with $\Omega = S$ and that the regularized gap function is *differentiable* whenever the mapping involved in the variational inequality problem is differentiable. Based on the regularized gap function,

Fukushima [Fuk92] also proposed a descent method for solving monotone variational inequality problems. The regularized gap function was also used in a globally convergent modification of Newton's method by Taji, Fukushima and Ibaraki [TFI93].

By replacing the quadratic term $\frac{1}{2}\langle y - x, G(y - x)\rangle$ in (1.5) with a general strongly convex function $\Phi(y - x)$, Wu, Florian and Marcotte [WFM93] generalized the regularized gap function and proposed a general descent framework for monotone variational inequality problems. Zhu and Marcotte [ZhM93] also proposed a similar generalization of the regularized gap function. By using their merit function, Zhu and Marcotte [ZhM93] proposed descent methods and globally convergent modifications of Newton's method and nonlinear Jacobi method [MaZ95]. Independently of Fukushima, Auchmuty [Auc89] proposed a class of merit functions, which includes the gap function, dual gap function and Fukushima's regularized gap function. Larsson and Patriksson [LaP94] generalized Auchmuty's class of merit functions.

More recently, Peng [Pen95] introduced the *D-gap function* for variational inequality problems. The D-gap function is defined on R^n as the difference of two regularized gap functions. Peng [Pen95] showed that the D-gap function is nonnegative on R^n and its zero points coincide with solutions to the variational inequality problem, and hence the D-gap function leads to an *unconstrained* optimization reformulation of the variational inequality problem. Various interesting properties of the D-gap function was investigated by Yamashita, Taji and Fukushima [YTF95].

Pang [Pan90] proposed another equivalent optimization formulation for variational inequality problems, which is based on B-differentiable equations, and developed a damped Newton method for variational inequality problems [PaG93]. Xiao and Harker [XiH94a, XiH94b] considered a similar B-differentiable optimization formulation, and proposed a globally convergent Newton method for solving variational inequality problems.

Recent developments of merit functions and related algorithms for variational inequality problems and complementarity problems are summarized in a survey paper by Fukushima [Fuk96]

In this thesis, we study optimization reformulations for the variational inequality problem and develop practically efficient algorithms. In particular, we show that the regularized gap function and its modification are useful in designing globally convergent Newton method.

1.3 Research objective and outline of the thesis

One of the main aims of the thesis is to develop efficient algorithms, based on equivalent optimization reformations, for solving the variational inequality problem and to examine their convergence properties. Another aim of this thesis is to construct a new merit function which enables us to deal effectively with variational inequality problems involving general nonlinear constraints. Furthermore, we shall demonstrate the practical usefulness of optimization reformulations of the variational inequality problem.

Chapter 2 introduces the definition of the variational inequality problem and related concepts and notations which will be necessary for the development of subsequent chapters.

In Chapter 3, we propose a globally convergent Newton method for solving variational inequality problems [TFI93]. We first consider the differentiable merit function introduced by Fukushima [Fuk92] to formulate the variational inequality problem as an optimization problem, and show some properties of the merit function. Using this function, we propose to modify Newton's method for variational inequality problems. The purpose of introducing this merit function is to provide some measure of the discrepancy between the solution of the variational inequality problem and the current iterate. It is shown that, under the strong monotonicity assumption, the method is globally convergent and, under some additional assumptions, the rate of convergence is quadratic.

The nonlinear complementarity problem is a special case of the variational inequality problem, and has also been used to study and formulate various equilibrium problems [Aas79, Flo89, FTSH83, Kar69a, Kar69b, Lem65, Mat87, Tob88]. In Chapter 4, we propose globally convergent methods for solving nonlinear complementarity problems [TaF94], based on a differentiable optimization reformulation of the nonlinear complementarity problem. These are applications of the methods proposed in [Fuk92] and in Chapter 3 for solving variational inequality problems, but they take full advantage of the special structure of the nonlinear complementarity problem. We establish global convergence of the proposed methods. Some computational experience indicates that the proposed methods are practically efficient.

In Chapter 5, we propose a new merit function for the variational inequality problem with general convex constraints [TaF96]. The proposed function is defined as an optimal value of a quadratic programming problem whose constraints consists of a linear approximation of the given nonlinear constraints. We show that the set of constrained minima of the proposed merit function coincides with the set of solutions to the variational inequality problem. We also show that this function is directionally differentiable in all directions and, under suitable assumptions, any stationary point of the function over the constraint set actually solves the variational inequality problem. Furthermore, we propose a descent method for solving the variational inequality problem and prove its global convergence.

In Chapter 6, we propose a new globally convergent Newton method for solving variational inequality problems with general inequality constraints [TaF95]. The method solves at each iteration an affine variational inequality problem, in which not

only the mapping of the problem but also the constraints are linearized. The algorithm has the property that a subproblem can be solved finitely at each iteration even if the constraints of the given problem are nonlinear. To establish global convergence, we make use of the merit function proposed in Chapter 5. We show that, when the mapping involved in the given problem is strongly monotone, the method is globally convergent to the solution, and that, under some additional assumptions, the rate of convergence is superlinear.

Finally, in Chapter 7, we summarize the results obtained in the thesis.

Chapter 2

Problem Definitions and Basic Concepts

2.1 Variational inequality problem and its applications

In this section, we define the variational inequality problem and state its relation to nonlinear equations, optimization problems and complementarity problems. We also present its applications to economic equilibrium problems.

Definition 2.1 Let S be a nonempty closed convex subset of R^n and let F be a continuous mapping from R^n into R^n. The *variational inequality problem* is to find a vector $x^* \in S$ such that

$$\langle F(x^*), x - x^* \rangle \geq 0 \quad \text{for all} \quad x \in S. \tag{2.1}$$

In geometric terms, inequality (2.1) states that the vector $F(x^*)$ is *normal* to the set S at the point x^*. Figure 2.1 illustrates a variational inequality problem in R^2 graphically. In the figure, the shaded region represents the set S. The mapping F can be thought as a vector field. At a solution x^*, the vector $F(x^*)$ is inward normal to

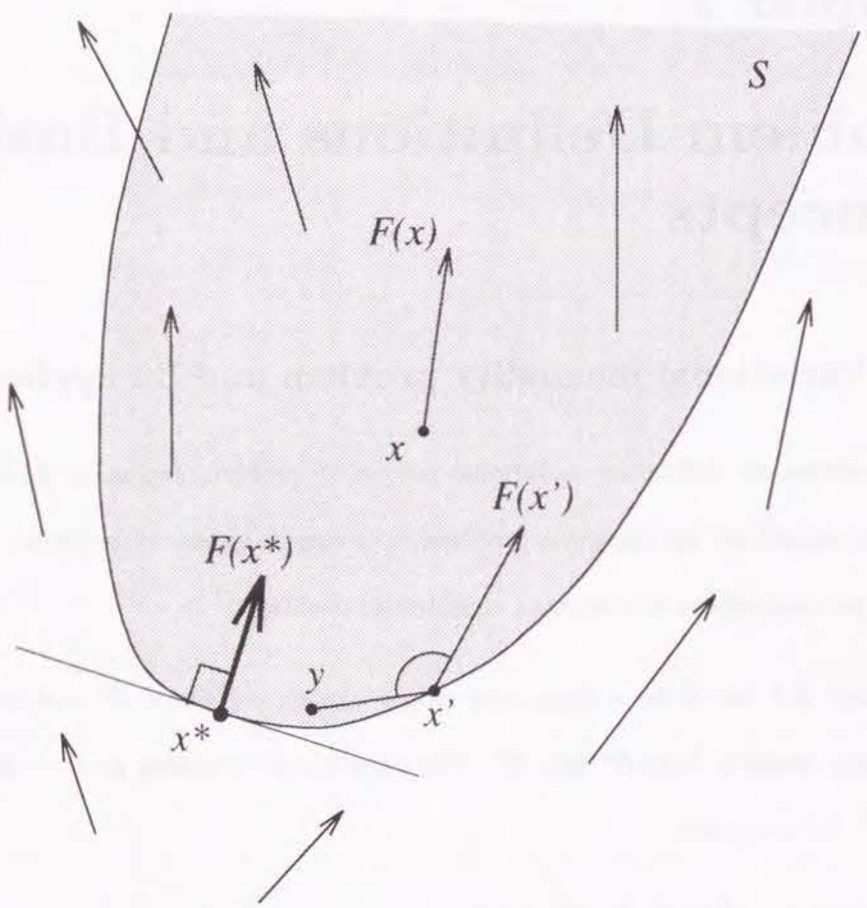

Figure 2.1: Illustration of a variational inequality problem in R^2

the boundary of S, while at the point x' which is not a solution, there is a $y \in S$ such that the vector $y - x'$ is at an obtuse angle to $F(x')$.

If the set S is defined by a system of inequalities and equalities of the form

$$S = \{x \in R^n \mid c_i(x) \leq 0, \quad i = 1, \ldots, m, \quad h_j(x) = 0, \quad j = 1, \ldots, l\}, \tag{2.2}$$

where $c_i : R^n \to R$ are continuously differentiable convex functions and $h_i : R^n \to R$ are affine functions, the following proposition holds under Slater's constrained qualification: there exists an $\hat{x} \in R^n$ such that

$$c_i(\hat{x}) < 0 \text{ for } i = 1, \ldots, m \text{ and } h_j(\hat{x}) = 0, \text{ for } j = 1, \ldots, l. \tag{2.3}$$

For the proof of the proposition, see [Tob86].

Proposition 2.1 *Suppose that $c_i : R^n \to R$, $i = 1, \ldots, m$ are continuously differentiable convex functions, $h_j : R^n \to R$, $j = 1, \ldots, l$, are affine functions and S is defined by* (2.2)*. Suppose also that Slater's constrained qualification* (2.3) *holds. Then x^* is a solution to* (2.1) *if and only if there exist Lagrange multipliers λ_i^*, $i = 1, \ldots, m$, and π_j^*, $j = 1, \ldots, l$, such that the vector (x^*, λ^*, π^*) satisfies the following conditions:*

$$\begin{aligned} & F(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla c_i(x^*) + \sum_{i=1}^{l} \pi_j^* \nabla h_j(x^*) = 0, \\ & c_i(x^*) \leq 0, \ \lambda_i^* \geq 0, \ \lambda_i^* c_i(x^*) = 0, \ i = 1, \ldots, m, \\ & h_j(x^*) = 0, \ j = 1, \ldots, l. \end{aligned} \tag{2.4}$$

The condition (2.4) corresponds to the Karush-Kuhn-Tucker condition in an optimization problem (cf. Appendix A.3.2) and is useful for the analyses in Chapters 5 and 6.

Various mathematical problems can be formulated as variational inequality problems. In the following, we explain how these problems relate variational inequality problems.

Nonlinear equation

The simplest example of a variational inequality problem is a system of *nonlinear equations.*

Proposition 2.2 *Let $S = R^n$ and let F be a continuous mapping from R^n into R^n. Then the variational inequality problem (2.1) is equivalent to the system of nonlinear equations:*

$$F(x^*) = 0. \tag{2.5}$$

Proof. If $F(x^*) = 0$, then inequality (2.1) holds with equality for all $x \in R^n$. Conversely, if x^* satisfies (2.1), then, by setting $x = x^* - F(x^*)$, we have

$$\langle F(x^*), -F(x^*)\rangle = - \| F(x^*) \|^2 \geq 0,$$

and hence, $F(x^*) = 0$. □

Optimization problem

The second example of a variational inequality problem arises from an *optimization problem.* Let us consider the optimization problem:

$$\begin{array}{ll} \text{minimize} & \varphi(x) \\ \text{subject to} & x \in S, \end{array} \tag{2.6}$$

where $\varphi : R^n \to R$ is a continuously differentiable function and S is a closed convex subset of R^n.

Definition 2.2 A vector x^* is a *stationary point* of the optimization problem (2.6) if $x^* \in S$ and

$$\langle \nabla\varphi(x^*), x - x^* \rangle \geq 0 \quad \text{for all } x \in S. \tag{2.7}$$

In the optimization theory, inequality (2.7) is often called the *first order necessary optimality condition* for problem (2.6). We note that inequality (2.7) is just the variational inequality problem (2.1) with $F = \nabla\varphi$.

The following two propositions clarify the relationship an optimization problem and a variational inequality problem.

Proposition 2.3 *Let x^* be a solution to the optimization problem (2.6), i.e., $\varphi(x) \geq \varphi(x^*)$ for all $x \in S$. Then x^* solves the variational inequality problem (2.7).*

Proof. Since S is convex, $x^* + t(x - x^*) \in S$ for any $x \in S$ and $0 \leq t \leq 1$. Then we have that $\varphi(x^* + t(x - x^*)) \geq \varphi(x^*)$. Hence,

$$\langle \nabla\varphi(x^*), x - x^* \rangle = \lim_{t \to 0+} \frac{\varphi(x^* + t(x - x^*)) - \varphi(x^*)}{t} \geq 0$$

holds for any $x \in S$. □

Proposition 2.4 *If the function φ is pseudo-convex on S, then a vector x^* satisfying (2.7) is a solution of (2.6).*

Proof. Since x^* satisfies (2.7), it follows from the definition of pseudo-convexity (cf. Appendix A.2) that

$$\varphi(x) \geq \varphi(x^*) \text{ for all } x \in S,$$

and hence, x^* is a solution to the optimization problem (2.6) $\square$

Complementarity problem

An important special case of the variational inequality problem (2.1) is the *complementarity problem.*

Definition 2.3 Let F be the mapping from R^n into itself. The complementarity problem is to find a vector $x^* \in R^n$ such that

$$x^* \geq 0, F(x^*) \geq 0 \text{ and } \langle x^*, F(x^*)\rangle = 0. \tag{2.8}$$

When the mapping F is affine, problem (2.8) is called a *linear complementarity problem.* When F is a general nonlinear mapping, problem (2.8) is called a *nonlinear complementarity problem.* The following proposition illustrates the relationship between a complementarity problem (2.8) and a variational inequality problem (2.1).

Proposition 2.5 *The vector x^* is a solution to the complementarity problem* (2.8) *if and only if $x^* \in R^n_+$ is a solution of the variational inequality problem:*

$$\langle F(x^*), x - x^*\rangle \geq 0 \quad \textit{for all} \quad x \in R^n_+. \tag{2.9}$$

Proof. Suppose that x^* is a solution of (2.8). Then for all $x \in R^n_+$, we have

$$0 = \langle x^*, F(x^*)\rangle \leq \langle x, F(x^*)\rangle,$$

that is

$$\langle F(x^*), x - x^*\rangle \geq 0 \quad \text{for all} \quad x \in R^n_+.$$

Therefore, x^* is a solution of (2.9).

Conversely, suppose that x^* is a solution of (2.9). Substituting $x = x^* + e_i$ into (2.9), we have that $\langle F(x^*), e_i\rangle = F_i(x^*) \geq 0$ for all $i = 1, \ldots, n$, and hence, $F(x^*) \geq 0$. Substituting $x = 2x^*$ into (2.9), we have

$$\langle F(x^*), x^*\rangle \geq 0. \tag{2.10}$$

We also have

$$\langle F(x^*), -x^*\rangle \geq 0 \tag{2.11}$$

by substituting $x = 0$ into (2.9). From (2.10) and (2.11), we have $\langle x^*, F(x^*)\rangle = 0$. $\square$

Economic equilibrium problems

Variational inequality problems are used to formulate and study various economic equilibrium problems. Here, we briefly explain how the Nash equilibrium problem and the traffic assignment problem can be formulated as variational inequality problems. For detailed expositions and for other economic equilibrium problems, the reader may refer to the book of Nagurney [Nag93].

The first example is a Nash equilibrium in an oligopolistic market [MSS82, Har84]. Let there be n firms which supply a homogeneous product in a noncooperative fashion.

Let the function $p(q)$ represent the price at which the consumers will demand a quantity q and let x_i denote the i-th firm's supply. Finally, let $s_i(x_i)$ denote the i-th firm's total cost of supplying x_i units. Then a *Nash equilibrium* solution for the market is a vector $x^* = (x_1^*, \ldots, x_n^*)^t$ such that x_i^* is a solution to the following optimization problem for all $i = 1, \ldots, n$:

$$\begin{array}{ll} \text{maximize} & x_i p(x_i + q_i^*) - s_i(x_i) \\ \text{subject to} & x_i \geq 0, \end{array}$$

where $q_i^* = \sum_{j \neq i} x_j^*$. Harker [Har84] shows that, when $s_i(x_i)$ is continuously differentiable and convex for all $i = 1, \ldots, n$, $p(\cdot)$ is strictly decreasing and continuously differentiable and $qp(q)$ is concave with respect to q, then x^* is a Nash equilibrium solution if and only if x^* is a solution to the variational inequality problem

$$\langle F(x^*), x - x^* \rangle \geq 0 \quad \text{for all} \quad x \in R_+^n,$$

where $F_i(x) = s_i'(x_i) - p\left(\sum_{j=1}^n x_j\right) - x_i p'\left(\sum_{j=1}^n x_j\right)$. Note that it follows from Proposition 2.5 that this problem can be reformulated to the complementarity problem:

$$\begin{array}{l} x_i^* \geq 0, \; s_i'(x_i^*) - p\left(\sum_{j=1}^n x_j^*\right) - x_i^* p'\left(\sum_{j=1}^n x_j^*\right) \geq 0 \\ \text{and} \;\; x_i^*\left(s_i'(x_i^*) - p\left(\sum_{j=1}^n x_j^*\right) - x_i^* p'\left(\sum_{j=1}^n x_j^*\right)\right) = 0 \\ \text{for all } i = 1, \ldots, n. \end{array}$$

The next example is a traffic assignment problem [Aas79, BeG82, Daf80, Smi79]. Consider a network consisting of a set of nodes $\mathcal{N}$ and a set of directed links $\mathcal{L}$ together with a set W of node pairs referred to as origin-destination (O/D) pairs. For each O/D

pair $w \in \mathcal{W}$, $\mathcal{P}_w$ denotes a set of simple directed paths joining w. We denote by x_p the flow on the path p. Then the feasible path flow vectors x whose components are x_p, $p \in \mathcal{P}_w$, $w \in \mathcal{W}$, is given by

$$X = \left\{ x \;\middle|\; \sum_{p \in \mathcal{P}_w} x_p = d_w, \; x_p \geq 0, \; \text{for all } p \in \mathcal{P}_w, \; w \in \mathcal{W} \right\}, \tag{2.12}$$

where $d_w > 0$ is a given demand for an O/D pair w. For each link $l \in \mathcal{L}$, the link flow y_l is defined as the sum of all the path flows on the paths p containing the link l, that is,

$$y_l = \sum_{p \in \mathcal{P}_w, w \in \mathcal{W}} \delta_{lp} x_p,$$

where $\delta_{lp} = 1$ if link l is contained in path p and $\delta_{lp} = 0$ otherwise.

Let c_l denote the user cost associated with traversing link l, and C_p the user cost associated with traversing the path p. For example, the value of c_l represents the travel time in traversing link l. We have

$$C_p = \sum_{l \in \mathcal{L}} \delta_{lp} c_a,$$

which may be viewed as the total travel time of path p. We assume that the cost associated with a link depends on the entire link flow pattern, that is,

$$c_a = c_a(y),$$

where y is a vector whose components are y_l, $l \in \mathcal{L}$.

Then the traffic equilibrium problem is to find a vector $x \in X$ such that for each path $p \in P_w$ and every O/D pair w

$$C_p \begin{cases} = \lambda_w & \text{if} \quad x_p^* > 0 \\ \geq \lambda_w & \text{if} \quad x_p^* = 0, \end{cases} \tag{2.13}$$

where λ_w is an indicator, whose value is not known a priori. This condition is based on the user optimization principle in which no user may decrease one's travel time by changing one's route unilaterally. In fact, the condition (2.13) asserts that only those paths connecting an O/D pair that have minimal user costs are used.

Let C be a mapping with components C_p and let $\Delta = (\delta_{lp})$ be a matrix. It can be shown [Daf80, Smi79] that the equilibrium condition (2.13) is equivalent to the variational inequality problem

$$\langle F(x^*), x - x^* \rangle \geq 0 \quad \text{for all} \quad x \in X,$$

where $F(x) = \Delta^t C(\Delta x)$.

2.2 Monotone mapping and projection

We first introduce the notion of monotonicity.

Definition 2.4 A mapping $F : R^n \to R^n$ is said to be

(a) *monotone* on S if

$$\langle F(x) - F(y), x - y \rangle \geq 0 \quad \text{for all} \quad x, y \in S, \tag{2.14}$$

(b) *strictly monotone* on S if

$$\langle F(x) - F(y), x - y \rangle > 0 \quad \text{for all} \quad x, y \in S, \ x \neq y, \tag{2.15}$$

(c) *strongly monotone* with modulus $\mu > 0$ on S if

$$\langle F(x) - F(y), x - y \rangle \geq \mu \parallel x - y \parallel^2 \quad \text{for all} \quad x, y \in S. \tag{2.16}$$

In the one-dimensional case, F is monotone if and only if F is nondecreasing and strictly monotone if and only if strictly increasing. Moreover, suppose that $F = \nabla\varphi$ where $\varphi : R^n \to R$ is a differentiable function. Then F is monotone if φ is convex; F is strictly monotone if φ is strict convex; and F is strongly monotone if φ is strongly convex. For definitions of convexity, strict convexity and strong convexity, see Appendix A.2.

Next proposition shows the relationship between the mapping F and its Jacobian $\nabla F(x)$.

Proposition 2.6 [OrR70, Chapter 5.4] *Suppose the mapping F is differentiable. Then F is strictly monotone on S if the Jacobian matrix $\nabla F(x)$ is positive definite for all $x \in S$, and F is strongly monotone on S if and only if $\nabla F(x)$ satisfies*

$$\langle x - y, \nabla F(x)(x - y)\rangle \geq \mu \, \|x - y\|^2 \quad \textit{for all } x, y \in S. \tag{2.17}$$

The variational inequality problem (2.1) does not necessarily have a solution. But when the solution set of (2.1) is actually nonempty, it is convex if F is monotone, and it is a singleton if F is strictly monotone. Furthermore, if F is strongly monotone, then problem (2.1) is guaranteed to have a unique solution [HaP90, Corollary 3.2].

Next we define the projection under the G-norm.

Definition 2.5 Let G be an $n \times n$ symmetric positive definite matrix. The *projection under the G-norm* of a vector $x \in R^n$ onto the set S, denoted by $\mathrm{Proj}_{S,G}(x)$, is defined as the unique solution y to the following optimization problem:

$$\text{minimize} \quad \| y - x \|_G \quad \text{subject to} \quad y \in S.$$

We note that, when $G = E_n$, $\mathrm{Proj}_{S,G}(x)$ reduces to the orthogonal projection. Figure 2.2 illustrates the difference between $\mathrm{Proj}_{S,G}(x)$ and the orthogonal projection. In the figure, dotted ellipses represent contours of the function $\varphi(y) = \| y - x \|_G^2$. Figure 2.2 illustrates that $\mathrm{Proj}_{S,G}(x)$ is a minimum of $\varphi(y)$ over S and is different from orthogonal projection x'.

Using this notation, we define a mapping $H_S \colon R^n \to R^n$ as

$$H_S(x) = \mathrm{Proj}_{S,G}(x - G^{-1}F(x)). \tag{2.18}$$

The next proposition characterizes a solution of the variational inequality problem as a fixed point of the mapping H_S.

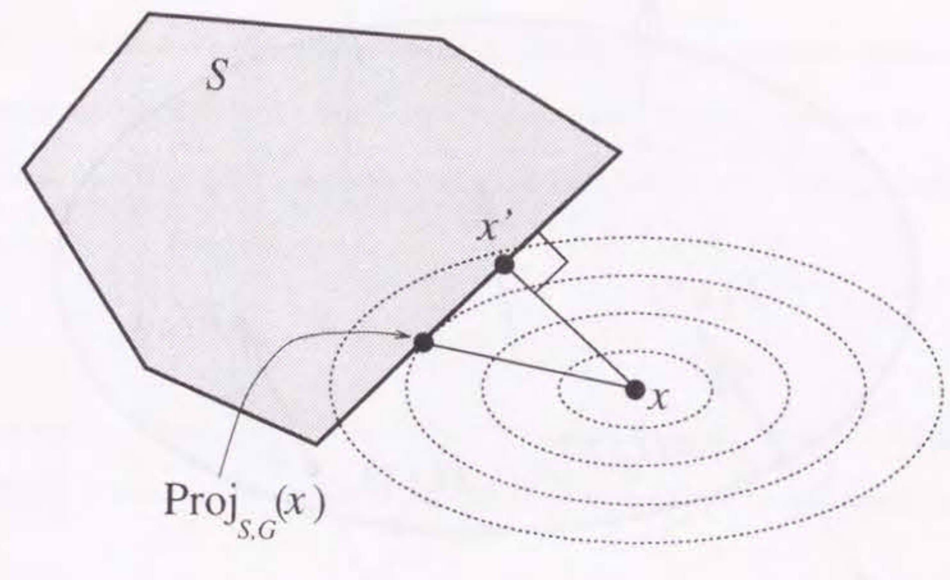

Figure 2.2: Illustration of the projection under the G-norm

Proposition 2.7 [BeT89, page 267] *Let G be an $n \times n$ symmetric positive definite matrix. Then x^* solves problem (2.1) if and only if*

$$x^* = \mathrm{Proj}_{S,G}(x^* - G^{-1}F(x^*)),$$

i.e., if and only if x^* is a fixed point of the mapping H_S.

Figure 2.3 illustrates H_S with $G = E_n$. In the case, $\mathrm{Proj}_{S,G}(x)$ reduces to the orthogonal projection. At the solution x^*, the vector $F(x^*)$ is inward normal to the boundary, and hence, $x^* = H_S(x^*)$ holds.

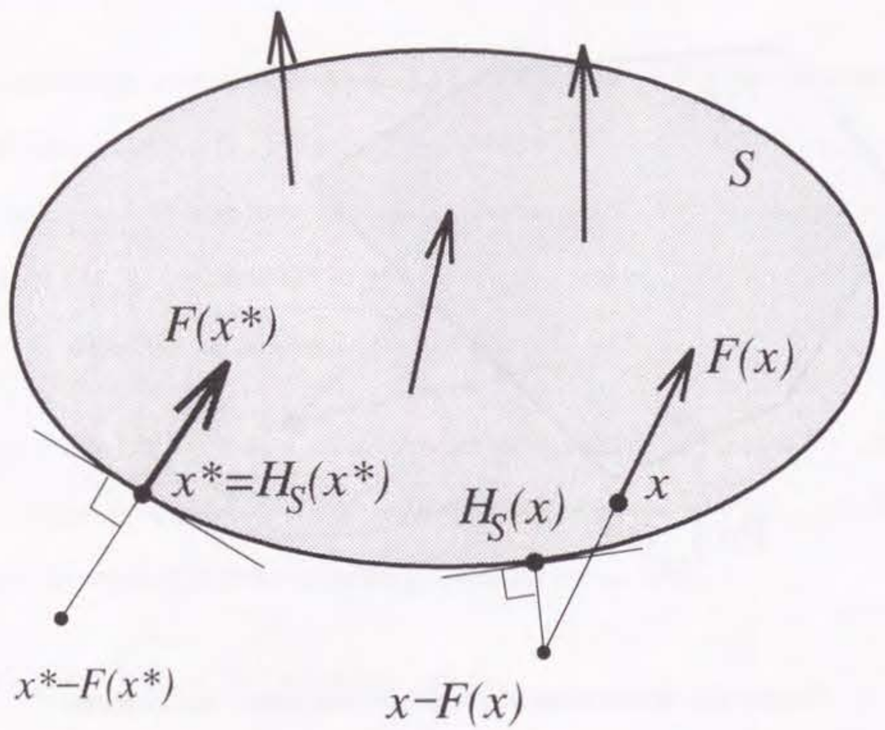

Figure 2.3: Illustration of a mapping H_S with $G = E_n$

It is known [BeT89, page 217] that the projection operator $\mathrm{Proj}_{S,G}(\cdot)$ is nonexpansive, i.e.,

$$\| \mathrm{Proj}_{S,G}(x) - \mathrm{Proj}_{S,G}(y) \|_G \leq \| x - y \|_G \quad \text{for all} \quad x, y \in R^n.$$

Hence, if the mapping F is continuous, so is the mapping H_S defined by (2.18).

2.3 Newton's method

Newton's method is a classical but useful method for solving nonlinear equations and unconstrained minimization problems. For example, Newton's method for solving nonlinear equations (2.5) generates a sequence $\{x^k\}$, where x^{k+1} is determined to be a solution to the linearized equations:

$$F(x^k) + \nabla F(x^k)^t (x - x^k) = 0. \tag{2.19}$$

It was shown [DeS83, Theorem 5.2.1] that the sequence $\{x^k\}$ converges quadratically if $\nabla F(x^*)$ is nonsingular and an initial iterate x^0 is chosen to be sufficiently close to x^*.

An early attempt to generalize Newton's method to solve variational inequality problems was made by Josephy [Jos79a]. We describe the basic Newton method for the variational inequality problem.

The basic Newton method for the variational inequality problem

> Choose an initial iterate $x^0 \in S$ and determine x^{k+1} to be a solution of the variational inequality problem obtained by linearizing F at the current iterate x^k, i.e., $x^{k+1} \in S$ and

$$\left\langle F(x^k) + \nabla F(x^k)^t (x^{k+1} - x^k), x - x^{k+1} \right\rangle \geq 0 \quad \text{for all} \quad x \in S. \tag{2.20}$$

The basic Newton method was originally proposed by Josephy [Jos79a] for the problem of generalized equations, which was introduced by Robinson [Rob79, Rob80, Rob83] and known to contain the variational inequality problem as a special case. But Josephy only considered the generalized equation equivalent to the variational inequality problem.

By the same argument of Proposition 2.2, it is easy to see that (2.20) reduces to (2.19) when $S = R^n$, so this Newton method is a natural generalization of Newton's method for nonlinear equations. The next Theorem establishes local and quadratic convergence of the basic Newton method for the variational inequality problem.

Theorem 2.1 [Jos79a, PaC82] *Let S be a nonempty closed convex subset of R^n, F be a continuous mapping from R^n into R^n, and x^* be a solution to the variational inequality problem (2.1). Suppose that F is continuously differentiable with $\nabla F(x^*)$ being positive definite and that ∇F is Lipschitz continuous in some neighborhood of x^*. Then there exists a neighborhood of x^* such that, if the initial iterate x^0 is chosen there, the sequence $\{x^k\}$ generated by the basic Newton method converges to the solution x^* quadratically, i.e., there exists a constant $\zeta > 0$ such that*

$$\| x^{k+1} - x^* \| \leq \zeta \| x^k - x^* \|^2 . \tag{2.21}$$

Proposition 2.5 says that, if $S = R^n_+$, the variational inequality problem (2.1) is equivalent to the nonlinear complementarity problem (2.8). The same proposition also says that inequality (2.20) can be written as a linear complementarity problem. Thus the above Newton method can be naturally transmitted to the complementarity problem.

The basic Newton method for the complementarity problem

Choose an initial iterate $x^0 \geq 0$ and determine x^{k+1} to be a solution $\bar{x}$ of the linearized complementarity problem:

$x \geq 0,\ F(x^k) + \nabla F(x^k)^t(x - x^k) \geq 0$

and $\left\langle x, F(x^k) + \nabla F(x^k)^t(x - x^k)\right\rangle = 0.$

An analogue to Theorem 2.1 holds for the above Newton method for nonlinear complementarity problems, that is, under the assumptions that $\nabla F(x^*)$ is positive definite and ∇F is Lipschitz continuous, then the sequence generated by the above algorithm quadratically converges to the solution x^* of the complementarity problem (2.8), provided that an initial iterate $x^0 \geq 0$ is chosen sufficiently close to x^*.

Chapter 3

A Globally Convergent Newton Method for Solving Strongly Monotone Variational Inequality Problems

3.1 Introduction

In this chapter, we consider the variational inequality problem:

$$\langle F(x^*), x - x^* \rangle \geq 0 \quad \text{for all} \quad x \in S \tag{3.1}$$

where S denotes a nonempty closed convex subset of R^n and F denotes a continuous mapping from R^n into R^n (cf. Definition 2.1).

By incorporating a line search strategy, Marcotte and Dussault [MaD87] have modified the basic Newton method (2.20) to obtain a globally convergent algorithm. Their method is based on the use of the gap function $g : R^n \to R \cup \{+\infty\}$ defined by (1.2). When F is monotone, the method of Marcotte and Dussault [MaD87] is shown to converge globally to a solution of problem (3.1) and, under suitable assumptions, the rate of convergence is quadratic. It is noted that the set S is assumed to be compact in order that the function g is well-defined. Moreover, global convergence has been proved with the exact line search to the gap function (1.2) [MaD87].

In this chapter we propose another modification of Newton's method. The method makes use of the regularized gap function defined by (1.5). It is shown that, when F is strongly monotone, the algorithm is globally convergent to a solution of problem (3.1), and, under some additional assumptions, the rate of convergence is quadratic. In particular, the method allows *inexact* line search and does not rely upon the compactness assumption on the set S. Limited computational experience indicates that the proposed method is well comparable to the method of Marcotte and Dussault [MaD87].

3.2 Regularized gap function

In this section, we introduce the regularized gap function for variational inequality problem (3.1) and present some of its properties.

Definition 3.1 For an arbitrarily chosen positive definite symmetric $n \times n$ matrix G, we define the *regularized gap function* $f_S : R^n \to R$ for a variational inequality problem (3.1) by

$$f_S(x) = \max\left\{ -\langle F(x), y - x\rangle - \frac{1}{2}\langle y - x, G(y - x)\rangle \;\middle|\; y \in S \right\} \tag{3.2}$$

$$= -\langle F(x), H_S(x) - x\rangle - \frac{1}{2}\langle H_S(x) - x, G(H_S(x) - x)\rangle, \tag{3.3}$$

where the mapping $H_S : R^n \to R^n$ is defined by (2.18).

Note that, by the positive definiteness of G, the maximum in (3.2) is always uniquely attained by $y = H_S(x)$. Using the regularized gap function, an equivalent optimization problem can be obtained for any variational inequality problem.

Proposition 3.1 [Fuk92] *Let f_S be the regularized gap function defined by* (3.2). *Then $f_S(x) \geq 0$ for all $x \in S$, and $f_S(x) = 0$ if and only if x solves* (3.1). *Hence x solves* (3.1) *if and only if it solves the following optimization problem and its optimal value is zero:*

$$\text{minimize } f_S(x) \text{ subject to } x \in S. \tag{3.4}$$

Remark 3.1 When problem (3.1) has no solution, the optimization problem (3.4) may have a minimizer which does not zero the function f_S. For example, consider the case where $F : R \to R$, $F(x) = -1$ and $S = \{x \in R \,|\, x \geq 0\}$. The corresponding variational inequality problem is to find an $x^* \geq 0$ such that $x^* \geq x$ holds for all $x \geq 0$. But the existence of such a number is impossible and hence the variational inequality problem has no solution. The function f_S associated with $G = 1$ is $f_S(x) = \frac{1}{2}$ for all $x \geq 0$, so any $x \geq 0$ is a global minimizer of problem (3.4).

It can be shown that, for any closed convex set S, the regularized gap function is continuously differentiable whenever so is the mapping F.

Proposition 3.2 [Fuk92] *If a mapping $F: R^n \to R^n$ is continuous, then the function f_S defined by 3.2 is also continuous. Furthermore, if F is continuously differentiable, then f_S is also continuously differentiable and its gradient is given by*

$$\nabla f_S(x) = F(x) - [\nabla F(x) - G](H_S(x) - x). \tag{3.5}$$

The regularized gap function has an interesting property that, when $\nabla F(x)$ is positive definite for all $x \in S$, *any* stationary point of problem (3.4) is a *global* optimal solution of problem (3.4). The function f_S is in general not convex.

Proposition 3.3 [Fuk92] *Assume that a mapping $F: R^n \to R^n$ is continuously differentiable and its Jacobian $\nabla F(x)$ is positive definite for all $x \in S$. If x is a stationary point of problem* (3.4), *i.e.,*

$$\langle \nabla f_S(x), y - x\rangle \geq 0 \quad \text{for all} \quad y \in S, \tag{3.6}$$

then x is a global optimal solution of problem (3.4), *and hence it solves the variational inequality problem* (3.1).

This proposition indicates that the regularized gap function can used to construct a descent method for solving monotone variational inequality problems. Moreover, when the mapping F is strongly monotone on S, the following result holds. Note that this result does not require the differentiability of F.

Proposition 3.4 *Let x^* be a solution to* (3.1). *If F is strongly monotone with modulus μ on S, then the regularized gap function f_S satisfies the inequality*

$$f_S(x) \geq \left(\mu - \frac{1}{2}\|G\|\right) \| x - x^* \|^2 \quad \text{for all} \quad x \in S. \tag{3.7}$$

In particular, if the matrix G is chosen sufficiently small to satisfy $\|G\| < 2\mu$, then

$$\lim_{x \in S,\ \|x\| \to \infty} f_S(x) = +\infty.$$

Proof. Since x^* is a solution to (3.1), we have

$$\langle F(x^*), x - x^*\rangle \geq 0 \quad \text{for any} \quad x \in S.$$

From this inequality and the definition of strong monotonicity (2.16), we obtain

$$\langle F(x), x - x^*\rangle \geq \mu \| x - x^* \|^2 \quad \text{for any} \quad x \in S. \tag{3.8}$$

Since $\langle x^* - x, G(x^* - x)\rangle \leq \|G\| \, \| x^* - x \|^2$, it follows from (3.2) and (3.8) that

$$\begin{aligned} f_S(x) &\geq -\langle F(x), x^* - x\rangle - \frac{1}{2}\langle x^* - x, G(x^* - x)\rangle \\ &\geq \mu \| x - x^* \|^2 - \frac{1}{2}\|G\| \, \| x - x^* \|^2 \\ &= \left(\mu - \frac{1}{2}\|G\|\right) \| x - x^* \|^2 . \end{aligned}$$

The last half of the proposition then follows immediately. □

3.3 Globally convergent Newton method

In this section we present a globally convergent Newton method for solving the variational inequality problem (3.1), which incorporates a line search strategy of Armijo-type to the basic Newton method (2.20). Throughout this section we assume that the mapping F is continuously differentiable and strongly monotone with modulus μ (cf. (2.16)).

We first define the global convergence formally.

Definition 3.2 Consider the optimization problem:

$$\begin{array}{ll} \text{minimize} & \varphi(x) \\ \text{subject to} & x \in \Omega, \end{array} \tag{3.9}$$

where $\varphi : R^n \to R \cup \{+\infty\}$ is a function and Ω is a subset of R^n. Then an algorithm $\mathcal{A}$ is said to be *globally convergent* if, for any initial iterate $x^0 \in \Omega$, every accumulation point of the sequence $\{x^k\}$ generated by $\mathcal{A}$ is a solution x^* to problem (3.9), i.e. $\lim_{k\to\infty, x\in K} x^k = x^*$ for some subsequence $\{x^k\}_{k\in K}$.

For given $x \in S$, we consider the following linearized variational inequality problem which is to find $\bar{x} \in S$ such that

$$\left\langle F(x) + \nabla F(x)^t(\bar{x} - x), y - \bar{x} \right\rangle \geq 0 \quad \text{for all} \quad y \in S. \tag{3.10}$$

The strong monotonicity of F ensures that the linearized problem (3.10) always has a unique solution $\bar{x}$ in S. The linearized problem (3.10) is usually easier to solve than the original problem (3.1). In particular, if the set S is polyhedral convex, problem

(3.10) can be rewritten as a linear complementarity problem, which can be solved in a finite number of steps using Lemke's complementary pivoting method [Eav78].

In the remainder of this chapter, the linearized problem (3.10) is called LVI(x) and its unique solution is denoted $N(x)$.

Now we explicitly describe the algorithm.

Algorithm 3.1

Step 0 Choose $x^0 \in S$, $0 < \beta < 1$, $0 < \gamma < 1$, $0 < \sigma < 1$, and a symmetric positive definite matrix G. Let $k := 0$.

Step 1 Find the unique solution $N(x^k) \in S$ that solves LVI(x^k), i.e.,

$$\left\langle F(x^k) + \nabla F(x^k)^t(N(x^k) - x^k), x - N(x^k) \right\rangle \geq 0 \text{ for all } x \in S.$$

Let $d^k := N(x^k) - x^k$.

Step 2 If $f_S(x^k + d^k) \leq \gamma f_S(x^k)$, then set $\alpha_k := 1$ and go to Step 3.

Otherwise set $\alpha_k := \beta^{l_k}$ where l_k is the smallest nonnegative integer l such that

$$f_S(x^k) - f_S(x^k + \beta^l d^k) \geq -\sigma\beta^l \left\langle \nabla f_S(x^k), d^k \right\rangle.$$

Step 3 Set $x^{k+1} := x^k + \alpha_k d^k$. Let $k := k + 1$. Return to Step 1.

The next result shows that the Newton direction $d^k = N(x^k) - x^k$ obtained by solving LVI(x^k) is a feasible descent direction for f_S at x^k. This in particular implies that the step size α_k can be found in a finite number of steps at each iteration of the algorithm.

Theorem 3.1 *Let S be a nonempty closed convex subset of R^n, F be a mapping from R^n into itself, G be an $n \times n$ symmetric positive definite matrix and $f_S : R^n \to R$ be a function defined by (3.2). Suppose that F is continuously differentiable and strongly monotone with modulus μ on S. If x^k is not a solution to (3.1), then the vector $d^k = N(x^k) - x^k$ satisfies the inequality*

$$\left\langle \nabla f_S(x^k), d^k \right\rangle < -\left(\mu - \frac{1}{2} \| G \| \right) \| d^k \|^2 . \tag{3.11}$$

In particular, if the matrix G is chosen sufficiently small to satisfy $\| G \| \le 2\mu$, then d^k is a feasible descent direction of f_S at x^k.

Proof. For simplicity of presentation, we omit the superscript k in x^k and d^k. Since $d = N(x) - x$, it follows from (3.5) that

$$\begin{aligned}
\langle \nabla f_S(x), d \rangle &= \langle F(x), N(x) - x \rangle - \langle (\nabla F(x)^t - G)(N(x) - x), H_S(x) - x \rangle \\
&= \langle F(x) + \nabla F(x)^t (N(x) - x), N(x) - x \rangle \\
&\quad - \langle N(x) - x, \nabla F(x)^t (N(x) - x) \rangle \\
&\quad + \langle F(x) + \nabla F(x)^t (N(x) - x), x - H_S(x) \rangle \\
&\quad - \langle F(x), x - H_S(x) \rangle - \langle G(N(x) - x), x - H_S(x) \rangle \\
&= -\langle F(x) + \nabla F(x)^t (N(x) - x), H_S(x) - N(x) \rangle \\
&\quad + \left\{ \langle F(x), H_S(x) - x \rangle + \frac{1}{2} \langle H_S(x) - x, G(H_S(x) - x) \rangle \right\} \\
&\quad - \langle d, \nabla F(x) d \rangle + \frac{1}{2} \langle d, G d \rangle - \frac{1}{2} \| N(x) - H_S(x) \|_G^2 .
\end{aligned} \tag{3.12}$$

Since $N(x)$ is a solution to LVI(x), the first term of (3.12) is nonpositive. From (3.3), the second term of (3.12) is equal to $-f_S(x)$, which is *strictly* negative since x is not a solution to (3.1) (see Proposition 3.1). Hence, we have

$$\langle \nabla f_S(x), d \rangle < -\langle d, \nabla F(x) d \rangle + \frac{1}{2} \langle d, G d \rangle .$$

But since strong monotonicity of F implies $\langle d, \nabla F(x) d \rangle \ge \mu \| d \|^2$ and since $\langle d, G d \rangle \le \| G \| \| d \|^2$, we have the inequality

$$\langle \nabla f_S(x), d \rangle < -\left(\mu - \frac{1}{2} \| G \| \right) \| d \|^2 .$$

The last half of the theorem then follows immediately. □

The following theorem is the main result of this section.

Theorem 3.2 (global convergence) *Let S be a nonempty closed convex subset of R^n, F be a mapping from R^n into itself, G be an $n \times n$ symmetric positive definite matrix and $f_S : R^n \to R$ be a function defined by (3.2). Suppose that the mapping F is continuously differentiable and strongly monotone with modulus μ on S. If the matrix G is chosen sufficiently small to satisfy $\| G \| < 2\mu$, then the sequence $\{x^k\}$ generated by Algorithm 3.1 is globally convergent.*

Before proving the theorem, we show the following lemma.

Lemma 3.1 *If F is continuously differentiable and strongly monotone on S, then the mapping $N : S \to S$ is continuous on S. Furthermore, x is the solution of problem (3.1) if and only if x satisfies $x = N(x)$.*

Proof. The first half of the proposition follows from [HaP90, Theorem 5.4]. To prove the second half, suppose first that $x = N(x)$. Then it follows from (3.10) that

$$\langle F(x), y - x\rangle \geq 0 \quad \text{for all} \quad y \in S.$$

Thus, x is the solution to (3.1). Conversely, suppose that x solves (3.1). Then, since $N(x) \in S$, it follows from (3.1) that

$$\langle F(x), N(x) - x\rangle \geq 0.$$

Also it follows from (3.10) that

$$\langle F(x) + \nabla F(x)^t (N(x) - x), x - N(x)\rangle \geq 0.$$

Adding the last two inequalities and rearranging terms, we obtain

$$\langle N(x) - x, \nabla F(x)(N(x) - x)\rangle \leq 0.$$

But since the strong monotonicity of F ensures that $\nabla F(x)$ is positive definite (see Proposition 2.6), it follows that $x = N(x)$. □

Proof of Theorem 3.2. Since x^k and $x^k + d^k$ both belong to S, and since $0 < \alpha_k \leq 1$, it follows from the convexity of S that the sequence $\{x^k\}$ is contained in S. Moreover, by Theorem 3.1 and the line search rule of the algorithm, the sequence $\{f_S(x^k)\}$ is nonincreasing. This, together with Proposition 3.4, implies that the sequence $\{x^k\}$ is bounded, and hence has at least one accumulation point.

If $f_S(x^k + d^k) \leq \gamma f_S(x^k)$ holds infinitely often, then $\lim_{k\to\infty} f_S(x^k) = 0$. Since f_S is continuous by Proposition 3.2, we have $f_S(\bar{x}) = 0$ for any accumulation point $\bar{x}$ of

$\{x^k\}$. Moreover, by Proposition 3.1, $\bar{x}$ is a solution of (3.1). Since (3.1) has a unique solution by the strong monotonicity of F [HaP90, Corollary 3.2], it then follows that the whole sequence $\{x^k\}$ necessarily converges to the unique solution of problem (3.1).

Next we consider the case where $f_S(x^k + d^k) \leq \gamma f_S(x^k)$ holds for only finitely many k. Let $\{x^k\}_{k\in K}$ be any convergent subsequence of $\{x^k\}$ and let $\bar{x} \in S$ be its limit point. Since $d^k = N(x^k) - x^k$ and since $N(\cdot)$ is continuous by Lemma 3.1, $\{d^k\}_{k\in K}$ converges to the vector $\bar{d} = N(\bar{x}) - \bar{x}$. Therefore, by Lemma 3.1, to prove that $\bar{x}$ is a solution of (3.1), it is sufficient to show that

$$\bar{d} = 0.$$

Assume the contrary. Then there exist $\varepsilon > 0$ and an index $\bar{k}$ such that

$$\| d^k \| \geq \varepsilon \quad \text{for all} \quad k \in K,\ k \geq \bar{k}.$$

Thus it follows from (3.11) that

$$\langle \nabla f_S(x^k), d^k\rangle < -\left(\mu - \frac{1}{2}\,\|G\|\right)\varepsilon^2 \quad \text{for all} \quad k \in K,\ k \geq \bar{k}, \tag{3.13}$$

from which we obtain

$$\langle \nabla f_S(\bar{x}), \bar{d}\rangle \leq -\left(\mu - \frac{1}{2}\,\|G\|\right)\varepsilon^2 < 0. \tag{3.14}$$

On the other hand, it follows from the line search rule that

$$f_S(x^k) - f_S(x^{k+1}) \geq -\sigma\alpha_k \left\langle \nabla f_S(x^k), d^k \right\rangle \tag{3.15}$$

and

$$f_S(x^k) - f_S(x^k + \frac{\alpha_k}{\beta} d^k) < -\sigma \frac{\alpha_k}{\beta} \left\langle \nabla f_S(x^k), d^k \right\rangle \tag{3.16}$$

for all $k \in K$ sufficiently large. Since, by Proposition 3.1, the sequence $\{f_S(x^k)\}$ is nonnegative and since $\{f_S(x^k)\}$ is monotonically decreasing, (3.15) implies

$$\alpha_k \left\langle \nabla f_S(x^k), d^k \right\rangle \to 0.$$

Hence, from (3.13), we have

$$\{\alpha_k\}_{k \in K} \to 0.$$

Then, dividing both sides of (3.16) by α_k/β and taking limit, we obtain

$$-\left\langle \nabla f_S(\bar{x}), \bar{d} \right\rangle \le -\sigma \left\langle \nabla f_S(\bar{x}), \bar{d} \right\rangle.$$

Since $\sigma < 1$, this implies

$$\left\langle \nabla f_S(\bar{x}), \bar{d} \right\rangle \ge 0,$$

which contradicts (3.14).

Consequently, we have $\bar{d} = 0$, i.e., $\bar{x}$ is a solution of (3.1). Thus, it follows from Lemma 3.1 that any accumulation point of $\{x^k\}$ is a solution to (3.1). Moreover, since strong monotonicity of F ensures that problem (3.1) has a unique solution, we can conclude that the entire sequence converges to the unique solution of (3.1). □

Remark 3.2 Marcotte and Dussault [MaD87] have obtained a globally convergent Newton method that uses the gap function g defined by (1.2) as a merit function. In their method, F is assumed monotone but not necessary strongly monotone. To obtain global convergence, however, the method assumes the following exact line search rule:

Step 2' If $g(x^k + d^k) \le \gamma \frac{1}{2} g(x^k)$, then set $\alpha_k := 1$ and go to Step 3.

Otherwise find α_k such that

$$\alpha_k \in \arg\min_{\alpha \in [0,1]} g(x^k + \alpha d^k).$$

Remark 3.3 In Algorithm 3.1, the inexact line search rule may be replaced by the exact rule of Marcotte and Dussault's algorithm [MaD87]. Under the same assumption of Theorem 3.2, global convergence of the algorithm with exact line search can be proved in a way similar to [MaD87]. Moreover, the analysis of the rate of convergence [MaD87] to be given in the next section also remains in force for the algorithm with exact line search.

3.4 Rate of convergence

In this section, we show that, under suitable assumptions, Algorithm 3.1 proposed in the previous section is locally quadratically convergent. We show the quadratic rate of convergence only for the case that S is polyhedral convex. For a general convex set S, whether or not Algorithm 3.1 is quadratically convergent is unknown.

To obtain a rate of convergence result, we need the following strict complementarity condition [MaD87].

Definition 3.3 Suppose that S is polyhedral and that problem (3.1) has a unique solution x^*. Let S^* denote the minimal face of S containing x^*. Then we say that the *geometric strict complementarity* holds at x^* if $x \in S$ and $\langle F(x^*), x - x^* \rangle = 0$ imply $x \in S^*$.

Figure 3.1 illustrates the geometric strict complementarity condition. In the figure, x^* denotes the unique solution. Then S^* is denoted by the bold line of (a). In (b), the geometric strict complementarity condition holds. On the other hand, in (c) the vector $F(x^*)$ is normal to the face S_2, and hence, the geometric strict complementarity condition does not hold.

When the set S is defined by (2.2), a solution x^* satisfies the condition (2.4) with vectors λ^* and π^*. Then the geometric strict complementarity condition holds if c_i, $i = 1, \ldots, m$, are all affine and $c_i(x^*) = 0$ implies $\lambda_i^* > 0$ for all $i = 1, \ldots, m$.

Now we can establish the following result of the rate of convergence.

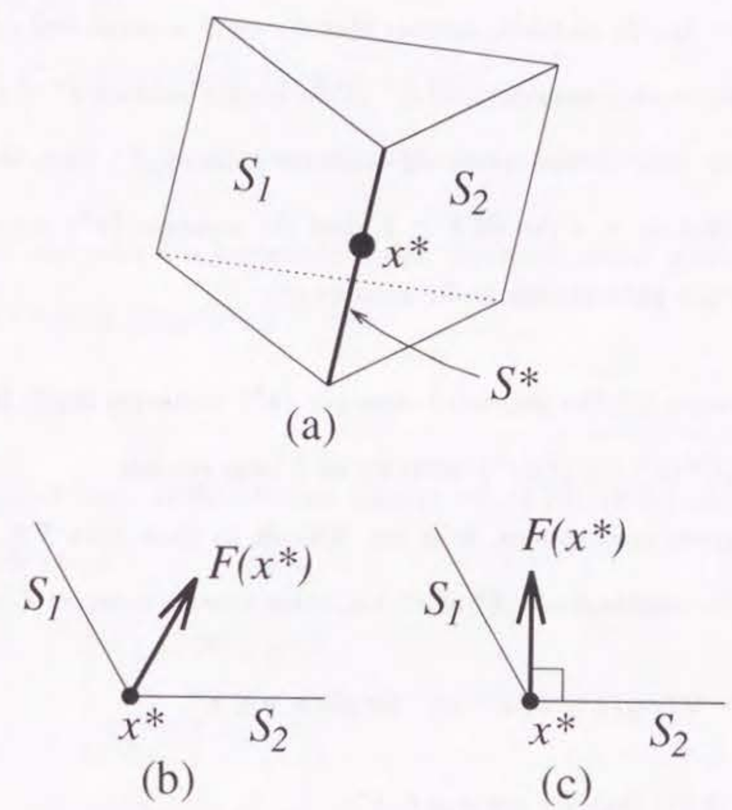

Figure 3.1: Illustration of the geometric strict complementarity condition

Theorem 3.3 (quadratic convergence) *Let S be a nonempty closed convex subset of R^n, F be a mapping from R^n into itself, G be an $n \times n$ symmetric positive definite matrix and $f_S : R^n \to R$ be a function defined by (3.2). Suppose that F is continuously differentiable and strongly monotone with modulus μ on S and that the matrix G satisfies $\| G \| < 2\mu$. In addition, suppose that the set S is polyhedral convex, ∇F is Lipschitz continuous on a neighborhood X^* of the unique solution x^* of problem (3.1) and the geometric strict complementarity condition holds at x^*. Then there exists an integer $\bar{k}$ such that $\alpha_k = 1$ for all $k \geq \bar{k}$, and the sequence $\{x^k\}$ generated by the algorithm converges quadratically to the solution x^*.*

Proof. By Theorem 3.2, the generated sequence $\{x^k\}$ converges to x^*. It is sufficient to prove that $f_S(N(x^k)) \leq \gamma f_S(x^k)$ holds for all k large enough.

Under the given assumptions, it is not difficult to show that ∇f_S is Lipschitz continuous on the neighborhood X^* of x^*, i.e., there exists a constant $L > 0$ such that

$$\| \nabla f_S(x) - \nabla f_S(y) \| \leq L \| x - y \| \quad \text{for all} \quad x, y \in X^*. \tag{3.17}$$

It follows from (3.17) that, for any $x, y \in X^*$,

$$\begin{aligned} f_S(x) - f_S(y) &= \int_0^1 \langle \nabla f_S(y + s(x-y)), x - y \rangle \, ds \\ &= \langle \nabla f_S(y), x - y \rangle + \int_0^1 \langle \nabla f_S(y + s(x-y)) - \nabla f_S(y), x - y \rangle \, ds \\ &\leq \langle \nabla f_S(y), x - y \rangle + \int_0^1 Ls \, \| x - y \|^2 \, ds \\ &= \langle \nabla f_S(y), x - y \rangle + \frac{1}{2} L \, \| x - y \|^2 . \end{aligned} \tag{3.18}$$

Since $f_S(x^*) = 0$ by Proposition 3.1, and since $\nabla f_S(x^*) = F(x^*)$ by (3.5) and Proposition 2.7, it follows from (3.18) that

$$f_S(x) \leq \langle F(x^*), x - x^* \rangle + \frac{1}{2} L \, \| x - x^* \|^2 \tag{3.19}$$

holds for all $x \in S \cap X^*$.

Since $x^k \to x^*$, it follows from Theorem 2.1 that

$$\| N(x^k) - x^* \| \leq \zeta \, \| x^k - x^* \|^2 \tag{3.20}$$

for some $\zeta > 0$, whenever k is sufficiently large. Moreover, under given assumptions, it follows from [MaD89, Proposition 1] that

$$\langle F(x^*), N(x^k) - x^* \rangle = 0 \tag{3.21}$$

for all k sufficiently large. It then follows from (3.19), (3.20), (3.21) and (3.7) that for any k sufficiently large

$$\begin{aligned} f_S(N(x^k)) &\leq \frac{1}{2} L \, \| N(x^k) - x^* \|^2 \\ &\leq \frac{1}{2} L\zeta^2 \, \| x^k - x^* \|^4 \\ &\leq \frac{L\zeta^2}{2\mu - \| G \|} \, \| x^k - x^* \|^2 \, f_S(x^k). \end{aligned}$$

Therefore, $f_S(N(x^k)) \leq \gamma f_S(x^k)$ holds for all k large enough to satisfy

$$\frac{L\zeta^2}{2\mu - \| G \|} \, \| x^k - x^* \|^2 \leq \gamma.$$

The quadratic rate of convergence follows from the fact that the convergence rate of the basic Newton iteration $x^{k+1} = N(x^k)$ is quadratic (see Theorem 2.1). □

3.5 Computational results

In this section, we report numerical results of Algorithm 3.1. All computer programs were coded in FORTRAN and run in double precision on a personal computer called Fujitsu FMR-70.

Throughout the computational experiments, the parameters used in the algorithm were set as $\beta = 0.5$, $\gamma = 0.5$ and $\sigma = 0.01$. The symmetric positive definite matrix G was chosen to be the identity matrix multiplied by 0.01. The convergence criterion was

$$f_S(x^k) \leq 10^{-6}.$$

For comparison purposes, we also coded the basic Newton method (cf. (2.20)) and the algorithm GNEW of Marcotte and Dussault [MaD87]. It is noted that we implemented the latter method with an inexact line search of Armijo-type, though its global convergence has been proved in [MaD87] only with exact line search rule (cf. Remark 3.2). In all test examples, the constraint sets S are polyhedral convex sets specified by linear inequalities. In solving the linearized subproblem LVI(x^k) at each iteration of the above-mentioned algorithms, we first transformed it into a linear complementarity problem, and then applied Lemke's complementary pivoting method to the latter problem [Lem65].

Example 3.1 is a modification of the test problem used by Marcotte and Dussault [MaD87]. In this problem, the constraint set S and the mapping F are taken respectively as

Table 3.1: Data for Example 3.1

$$F(x) = \begin{pmatrix} 0.726 & -0.949 & 0.266 & -1.193 & -0.504 \\ 1.645 & 0.678 & 0.333 & -0.217 & -1.443 \\ -1.016 & -0.225 & 0.769 & 0.934 & 1.007 \\ 1.063 & 0.567 & -1.144 & 0.550 & -0.548 \\ -0.259 & 1.453 & -1.073 & 0.509 & 1.026 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} + \rho \begin{pmatrix} \arctan(x_1 - 2) \\ \arctan(x_2 - 2) \\ \arctan(x_3 - 2) \\ \arctan(x_4 - 2) \\ \arctan(x_5 - 2) \end{pmatrix} + \begin{pmatrix} 5.308 \\ 0.008 \\ -0.938 \\ 1.024 \\ -1.312 \end{pmatrix}$$

Solution $x^* = (2.0, 2.0, 2.0, 2.0, 2.0)^t$

$$S = \left\{ x \in R^5 \;\middle|\; \sum_{i=1}^{5} x_i \geq 10,\ x_i \geq 0,\ i = 1, \ldots, 5 \right\}$$

and

$$F(x) = Px + \rho\Phi(x) + q,$$

where P is a 5×5 asymmetric positive definite matrix and $\Phi(x)$ is a nonlinear mapping with components $\Phi_i(x) = \arctan(x_i - 2)$, $i = 1, \ldots, 5$. The parameter ρ is used to vary the degree of asymmetry and nonlinearity. The data of Example 3.1 are given in Table 3.1. Numerical results for this example are shown in Tables 3.2 $\sim$ 3.5. It is noted that, since algorithm GNEW requires the set S to be compact, we had to include the extra

Table 3.2: Results for Example 3.1 ($\rho = 10$)

Initial Iterate	Algorithm	#Iterations
	Newton	12
$(25, 0, 0, 0, 0)$	GNEW	5
	Algorithm 3.1	5
	Newton	10
$(10, 0, 10, 0, 10)$	GNEW	7
	Algorithm 3.1	6
	Newton	12
$(10, 0, 0, 0, 0)$	GNEW	5
	Algorithm 3.1	5
	Newton	5
$(0, 2.5, 2.5, 2.5, 2.5)$	GNEW	4
	Algorithm 3.1	4

constraint $\sum_{i=1}^{5} x_i \leq 50$ when the problem was solved by this algorithm.

Example 3.2 consists of several test problems of various sizes, whose data are randomly generated. Specifically, in each problem, the constraint set S takes the form

$$S = \{x \in R^n \mid Ax \leq b, \ x \geq 0\},$$

and the mapping F is given by

$$F(x) = Px + \Psi(x) + q,$$

where P is an $n \times n$ asymmetric positive definite matrix and $\Psi(x)$ is a nonlinear mapping with components $\Psi_i(x) = p_i x_i^4$, where p_i are positive constants. The data of the

Table 3.3: Results for Example 3.1 ($\rho = 20$)

Initial Iterate	Algorithm	#Iterations
	Newton	failed
$(25, 0, 0, 0, 0)$	GNEW	6
	Algorithm 3.1	6
	Newton	failed
$(10, 0, 10, 0, 10)$	GNEW	7
	Algorithm 3.1	6
	Newton	failed
$(10, 0, 0, 0, 0)$	GNEW	6
	Algorithm 3.1	6
	Newton	9
$(0, 2.5, 2.5, 2.5, 2.5)$	GNEW	4
	Algorithm 3.1	4

smallest problem are given in Table 3.6. Numerical results of this example are shown in Table 3.7. We note that the both mappings of Examples 3.1 and 3.2 is strongly monotone on S.

In Example 3.1, the behavior of the basic Newton method is rather unstable (see Table 3.2 and 3.3). When the parameter ρ is so large that the mapping F is highly nonlinear, the Newton's method has failed for several initial iterates as shown in Table 3.3 though the mapping is strongly monotone. The same table also shows that the algorithms using line search strategies are always convergent, even if the initial iterates are chosen far from the solution. Moreover, Tables 3.2 and 3.3 show that, even when

Table 3.4: Result for Example 3.1 ($\rho = 10$): Algorithm 3.1

Iteration	$f_S(x^k)$	$\|x^k - x^*\|$	α_k	x_1	x_2	x_3	x_4	x_5
0	88721	23.345		25.000	0.0000	0.0000	0.0000	0.0000
1	13078	4.4570	1.0	0.0000	5.1395	2.6209	4.3643	1.8197
2	7492.9	3.9010	1.0	4.4516	0.0000	2.7069	0.0000	2.8416
3	71.933	0.5011	0.5	2.2258	2.2829	2.0183	1.8215	1.7034
4	1.0540	0.0211	1.0	1.9930	1.9894	1.9969	2.0050	2.0157
5	0.0000	0.0000	1.0	2.0000	2.0000	2.0000	2.0000	2.0000

Table 3.5: Result for Example 3.1 ($\rho = 20$): Algorithm 3.1

Iteration	$f_S(x^k)$	$\|x^k - x^*\|$	α_k	x_1	x_2	x_3	x_4	x_5
0	96697	8.9443		10.000	0.0000	0.0000	0.0000	0.0000
1	42955	5.6054	1.0	0.0000	5.7212	3.4167	5.1752	3.2181
2	31025	4.4870	1.0	5.4586	0.0000	2.1595	0.0000	2.3820
3	99.815	1.0188	0.5	2.7293	2.6397	1.9501	2.3001	1.9335
4	43.972	0.1708	1.0	1.8725	1.9510	2.0489	2.0637	2.0639
5	0.0342	0.0013	1.0	2.0011	1.9998	1.9998	1.9996	1.9997
6	0.0000	0.0000	1.0	2.0000	2.0000	2.0000	2.0000	2.0000

Table 3.6: Data for Example 3.2 ($n = 5$)

$$F(x) = \begin{pmatrix} 3.0 & -4.0 & -16.0 & -15.0 & -4.0 \\ 4.0 & 1.0 & -5.0 & -10.0 & -11.0 \\ 16.0 & 5.0 & 2.0 & -11.0 & -7.0 \\ 15.0 & 10.0 & 11.0 & 3.0 & -10.0 \\ 4.0 & 11.0 & 7.0 & 10.0 & 1.0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} + \begin{pmatrix} 0.004x_1^4 \\ 0.007x_2^4 \\ 0.005x_3^4 \\ 0.009x_4^4 \\ 0.008x_5^4 \end{pmatrix} + \begin{pmatrix} -15 \\ 10 \\ -50 \\ -30 \\ -25 \end{pmatrix}$$

$$A = \begin{pmatrix} 0.0 & 0.0 & -0.5 & 0.0 & -2.0 \\ -2.0 & -2.0 & 0.0 & -0.5 & -2.0 \\ 2.0 & 2.0 & -4.0 & 2.0 & -3.0 \\ -5.0 & 3.0 & -2.0 & 0.0 & 2.0 \end{pmatrix} \qquad b = \begin{pmatrix} -10 \\ -10 \\ 13 \\ 18 \end{pmatrix}$$

Solution $x^* = (9.08, 4.84, 0.00, 0.00, 5.00)^t$

Table 3.7: Results for Example 3.2

	Initial Iterate	Algorithm	#Iterations
$n = 5$	$(0, 0, 100, 0, 0)$	Newton	14
		GNEW	13
		Algorithm 3.1	13
$n = 10$	$(100, 0, 0, \dots, 0)$	Newton	14
		GNEW	13
		Algorithm 3.1	13
$n = 15$	$(100, 0, 0, \dots, 0)$	Newton	12
		GNEW	12
		Algorithm 3.1	12
$n = 20$	$(100, 0, 0, \dots, 0)$	Newton	12
		GNEW	12
		Algorithm 3.1	12
$n = 25$	$(100, 0, 0, \dots, 0)$	Newton	17
		GNEW	13
		Algorithm 3.1	13

the basic Newton method converges, it can happen that the number of iterations of Algorithm 3.1 is less than that of the basic Newton method. It can be observed from Tables 3.4 and 3.5 that Algorithm 3.1 converges quadratically to the solution. Note that, in this example, the unit step size is chosen at all but one iterations. On the other hand, Table 3.7 shows that the three versions of Newton's method all converge to the solution for each test problem of Example 3.2. However it is observed that the number of iterations is less than or equal to that of the basic Newton method. Note also that, in this example, the number of iterations is almost independent of the problem size, at least up to $n = 25$.

Finally, we may conclude that, as far as our limited computational experience is concerned, Algorithm 3.1 is well comparable to the algorithm GNEW of Marcotte and Dussault [MaD87].

Chapter 4

Optimization Based Globally Convergent Methods for the Nonlinear Complementarity Problem

4.1 Introduction

In this chapter we consider the nonlinear complementarity problem, which is to fined a vector $x \in R^n$ such that

$$x^* \geq 0, F(x^*) \geq 0 \text{ and } \langle x^*, F(x^*) \rangle = 0, \tag{4.1}$$

where F is a continuous mapping from R^n into itself. To solve the nonlinear complementarity problem (4.1), various iterative algorithms, such as fixed point algorithms, projection methods, nonlinear Jacobi method, successive over-relaxation methods and Newton's method, have been proposed [GaZ81, HaP90, PaC82]. Many of these methods are generalizations of classical methods for systems of nonlinear equations. Their convergence results have also been studied extensively [HaP90, PaC82].

Assuming the monotonicity of mapping F, Fukushima [Fuk92] has recently proposed a differentiable optimization formulation for variational inequality problem (cf. Section 3.2) and proposed a decent algorithm to solve variational inequality problem [Fuk92]. Based on this optimization formulation, we have proposed in Chapter 3 a modification of Newton's method for solving the variational inequality problem, and proved that, under the strong monotonicity assumption, the method is globally and quadratically convergent.

In this chapter, we apply the methods of Fukushima [Fuk92] and Algorithm 3.1 proposed in Chapter 3 to the nonlinear complementarity problem. We show that those methods can take full advantage of the special structure of problem (4.1), thereby yielding new algorithms for solving strongly monotone complementarity problems. We establish global convergence of the proposed methods, which are refinements of the results obtained for the variational inequality counterparts in several respects. In this chapter we show that the compactness assumption made in [Fuk92] can be removed for the strongly monotone complementarity problem. Moreover, some computational results shows that the proposed methods are practically efficient for solving monotone complementarity problems, though the convergence of the proposed methods is theoretically proved only under the strong monotonicity assumption.

4.2 Equivalent optimization problem

In this section, we introduce a merit function for the nonlinear complementarity problem (4.1) and present some of its properties.

Choose positive parameters $\delta_i > 0$, $i = 1, \ldots, n$ and define function $f_C : R^n \to R$ by

$$f_C(x) = \sum_{i=1}^{n} \frac{1}{2\delta_i} \left\{ F_i(x)^2 - (\max(0, F_i(x) - \delta_i x_i))^2 \right\}. \tag{4.2}$$

This function is a special case of the regularized gap function (3.2) originally introduced by Fukushima [Fuk92] for variational inequality problem (see Definition 3.1). Although some of its properties can be immediately derived from the results of [Fuk92], we give here simple and direct proofs for these properties, which utilize a special structure of problem (4.1).

For convenience, we define

$$f_C^i(x) = \frac{1}{2\delta_i} \left\{ F_i(x)^2 - (\max(0, F_i(x) - \delta_i x_i))^2 \right\}, \tag{4.3}$$

hence f_C is written as $f_C(x) = \sum_{i=1}^{n} f_C^i(x)$. We denote by D the diagonal matrix such that

$$D = \begin{pmatrix} \delta_1 & & 0 \\ & \ddots & \\ 0 & & \delta_n \end{pmatrix}. \tag{4.4}$$

We also denote

$$H_C(x) = \max\left(0, x - D^{-1}F(x)\right), \tag{4.5}$$

where the max operator is taken component-wise, i.e.,

$$(H_C)_i(x) = \max\left(0, x_i - \delta_i^{-1} F_i(x)\right).$$

Using these notations, we have the following result.

Lemma 4.1 *Let the mapping $H_C : R^n \to R^n_+$ be defined by (4.5). Then x^* solves (4.1) if and only if $x^* = H_C(x^*)$.*

Proof. Suppose that x^* is a solution to (4.1). Then either

$$\begin{cases} x_i^* = 0 \\ F_i(x^*) \geq 0 \end{cases} \quad \text{or} \quad \begin{cases} x_i^* \geq 0 \\ F_i(x^*) = 0 \end{cases}$$

holds for each $i = 1, \ldots, n$. Since $\delta_i > 0$, we have

$$x_i^* = 0 \Longrightarrow (H_C)_i(x^*) = \max(0, -\delta_i^{-1} F_i(x)) = 0,$$

and

$$F_i(x^*) = 0 \Longrightarrow (H_C)_i(x^*) = \max(0, x_i^*) = x_i^*.$$

Thus $(H_C)_i(x^*) = x_i^*$ for all i.

Conversely, suppose that x^* satisfies $x^* = H_C(x^*)$. Then, for each i, either

$$\begin{cases} x_i^* = 0 \\ x_i^* - \delta_i^{-1} F_i(x^*) \leq 0 \end{cases} \quad \text{or} \quad \begin{cases} x_i^* = x_i^* - \delta_i^{-1} F_i(x^*) \\ x_i^* - \delta_i^{-1} F_i(x^*) \geq 0 \end{cases}$$

holds. Hence, it follows from $\delta_i > 0$ that either $x_i^* = 0$ and $F_i(x^*) \geq 0$, or $x_i^* \geq 0$ and $F_i(x^*) = 0$ holds for each i. Thus x^* solves (4.1). □

Using the function (4.2), an equivalent optimization problem is obtained for the complementarity problem (4.1).

Proposition 4.1 *Let the function $f_C : R^n \to R$ be defined by (4.2). Then $f_C(x) \geq 0$ for all $x \geq 0$, and $f_C(x^*) = 0$ if and only if x^* solves (4.1). Hence, x solves (4.1) if and only if it is a solution to the following optimization problem and its optimal value is zero:*

$$\text{minimize} \quad f_C(x) \quad \text{subject to} \quad x \geq 0. \tag{4.6}$$

Proof. We first show that $f_C^i(x) \geq 0$ for all $x \geq 0$, so that $f_C(x) \geq 0$ for all $x \geq 0$. If $F_i(x) - \delta_i x_i \leq 0$, then $f_C^i(x) = (2\delta_i)^{-1} F_i(x)^2 \geq 0$. So we consider the case $F_i(x) - \delta_i x_i > 0$. Since $\delta_i > 0$, $x_i \geq 0$ and $F_i(x) > \delta_i x_i$ hold, we see from (4.3) that

$$\begin{aligned} f_C^i(x) &= \frac{1}{2\delta_i}\left\{F_i(x)^2 - (F_i(x) - \delta_i x_i)^2\right\} \\ &= x_i F_i(x) - \frac{\delta_i}{2} x_i^2 \\ &\geq \frac{\delta_i}{2} x_i^2 \\ &\geq 0. \end{aligned}$$

Therefore, $f_C(x) \geq 0$ holds for all $x \geq 0$.

Next, suppose $f_C(x^*) = 0$. Then $f_C^i(x^*) = 0$ must hold for all i. Hence, as shown in the above, either $F_i(x^*) = 0$ and $F_i(x^*) - \delta_i x_i^* \leq 0$, or $x_i^* = 0$ and $F_i(x^*) - \delta_i x_i^* > 0$ holds for each i. Therefore, x^* is a solution of (4.1).

Conversely, suppose that x^* solves (4.1). Then either $F(x_i^*) = 0$ or $x_i^* = 0$ holds for all i. If $F(x_i^*) = 0$, then from (4.3) we have

$$f_C^i(x^*) = -\frac{1}{2\delta_i}(\max(0, -\delta_i x_i^*))^2 = 0.$$

Also if $x_i^* = 0$, then we have

$$\begin{aligned} f_C^i(x^*) &= \frac{1}{2\delta_i}\left\{F_i(x^*)^2 - (\max(0, F_i(x^*)))^2\right\} \\ &= \frac{1}{2\delta_i}\left\{F_i(x^*)^2 - F_i(x^*)^2\right\} \\ &= 0. \end{aligned}$$

Therefore, we have $f_C(x^*) = 0$. □

Remark 4.1 When problem (4.1) has no solution, the optimization problem (4.6) may have a minimizer which does not zero the function f_C. For example, let us consider the case of R^1 and $F(x) = -x - 1$. Clearly, the complementarity problem has no solution. However, given $\delta > 0$, the corresponding optimization problem (4.6) is formulated as

$$\text{minimize} \quad \frac{1}{2\delta}(x+1)^2 \quad \text{subject to} \quad x \geq 0.$$

The unique optimal solution to this problem is $x = 0$, at which the function value is $\frac{1}{2\delta} > 0$.

It can be shown that the function f is continuously differentiable whenever so is the mapping F.

Proposition 4.2 *If the mapping F is continuously differentiable, then so is the function f_C defined by (4.2), and the gradient of f_C is given by*

$$\nabla f_C(x) = F(x) - (\nabla F(x) - D)(H_C(x) - x). \tag{4.7}$$

Proof. We first note that, if a function $\varphi : R^n \to R$ is continuously differentiable and $\Phi(x) = (\max(0, \varphi(x)))^2$, then Φ is continuously differentiable and the gradient of Φ is given by

$$\nabla\Phi(x) = 2\max(0, \varphi(x))\nabla\varphi(x).$$

Hence, we have from (4.3) that

$$\begin{aligned} \nabla f_C^i(x) &= \frac{1}{\delta_i}(F_i(x) - \max(0, F_i(x) - \delta_i x_i))\nabla F_i(x) \\ &\quad + \max(0, F_i(x) - \delta_i x_i)e_i. \end{aligned} \tag{4.8}$$

Since

$$\begin{aligned} \max(0, F_i(x) - \delta_i x_i) &= F_i(x) - \delta_i x_i + \delta_i \max(0, x_i - \delta_i^{-1}F_i(x)) \\ &= F_i(x) - \delta_i x_i + \delta_i (H_C)_i(x) \end{aligned}$$

holds, we have from (4.8) that

$$\begin{aligned} \nabla f_C^i(x) &= \frac{1}{\delta_i}(F_i(x) - \max(0, F_i(x) - \delta_i x_i))\nabla F_i(x) + \max(0, F_i(x) - \delta_i x_i)e_i \\ &= (x_i - (H_C)_i(x))\nabla F_i(x) + (F_i(x) - \delta_i x_i + \delta_i (H_C)_i(x))e_i. \end{aligned} \tag{4.9}$$

Therefore, we have from (4.9) that

$$\begin{aligned} \nabla f_C(x) &= \sum_{i=1}^{n} \nabla f_C^i(x) \\ &= \sum_{i=1}^{n} \{(x_i - (H_C)_i(x))\nabla F_i(x) + (F_i(x) - \delta_i x_i + \delta_i (H_C)_i(x))e_i\} \\ &= F(x) - (\nabla F(x) - D)(H_C(x) - x). \end{aligned}$$

This completes the proof. □

Proposition 4.1 says that finding a global optimal solution to (4.6) amounts to solving the complementarity problem (4.1). However, in general, optimization algorithms may only find a stationary point of the problem. Thus it is desirable to clarify conditions under which any stationary point of problem (4.6) actually solves problem (4.1). The next proposition answers for this question.

Proposition 4.3 *Suppose that $\nabla F(x)$ is positive definite for all $x \geq 0$. If $x \geq 0$ is a stationary point of problem* (4.6), *i.e.,*

$$\langle \nabla f_C(x), y - x \rangle \geq 0 \quad \text{for all} \quad y \geq 0, \tag{4.10}$$

then x is a global optimal solution of problem (4.6), *and hence it solves the nonlinear complementarity problem* (4.1).

Proof. Suppose that x satisfies (4.10). Then from (4.7) we have

$$\begin{aligned}
&\langle \nabla f_C(x), H_C(x) - x \rangle \\
&= \langle (F(x) - Dx + DH_C(x)) - \nabla F(x)(H_C(x) - x), H_C - x \rangle \\
&= \langle F(x) - Dx + DH_C(x), H_C(x) - x \rangle \\
&\quad - \langle H_C(x) - x, \nabla F(x)(H_C(x) - x) \rangle . \qquad (4.11)
\end{aligned}$$

It is easy to see that

$$\langle F(x) - Dx + DH_C(x), H_C(x) - x \rangle$$

$$\begin{aligned}
&= \sum_{i=1}^{n} \left\{ \delta_i (\max(0, x_i - \delta_i^{-1} F_i(x)) - x_i)^2 + (\max(0, x_i - \delta_i^{-1} F_i(x)) - x_i) F_i(x) \right\} \\
&\leq 0. \qquad (4.12)
\end{aligned}$$

Since x satisfies (4.10), it follows from (4.11) and (4.12) that

$$\langle H_C(x) - x, \nabla F(x)(H_C(x) - x) \rangle \leq 0.$$

However, since $\nabla F(x)$ is positive definite, we have $x = M(x)$. Therefore, it follows from Lemma 4.1 that x is a solution to (4.1). □

According to Definition 2.4, we call the mapping F a strongly monotone on R^n_+, if there exists a positive constant $\mu > 0$ such that

$$\langle F(x) - F(y), x - y \rangle \geq \mu \parallel x - y \parallel^2 \quad \text{for all} \quad x, y \geq 0. \tag{4.13}$$

When F is strongly monotone, we have the following result which establishes an asymptotic behavior of the function f_C. Note that similar results have not been obtained for the general variational inequality problem.

Proposition 4.4 *If F is strongly monotone with modulus μ on R^n_+, then*

$$\lim_{x \geq 0,\ \|x\| \to \infty} f_C(x) = +\infty.$$

Proof. Let $\{x^k\}$ be a sequence such that $x^k \geq 0$ and $\| x^k \| \to \infty$. Taking a subsequence if necessary, we may suppose that there exists a set $I \subset \{1, \ldots, n\}$ such that $x_i^k \to +\infty$ for $i \in I$ and $\{x_i^k\}$ is bounded for $i \notin I$. From $\{x^k\}$, we define another sequence $\{y^k\}$ such that $y_i^k = 0$ if $i \in I$ and $y_i^k = x_i^k$ if $i \notin I$. From (4.13) and the definition of y^k, we have

$$\sum_{i\in I}(F_i(x^k) - F_i(y^k))x_i^k \geq \mu \sum_{i\in I}(x_i^k)^2. \tag{4.14}$$

By Schwartz inequality we have

$$\left(\sum_{i\in I}(F_i(x^k) - F_i(y^k))^2\right)^{\frac{1}{2}} \left(\sum_{i\in I}(x_i^k)^2\right)^{\frac{1}{2}} \geq \sum_{i\in I}(F_i(x^k) - F_i(y^k))x_i^k. \tag{4.15}$$

It then follows from (4.14) and (4.15) that

$$\sum_{i\in I}(F_i(x^k) - F_i(y^k))^2 \geq \mu^2 \sum_{i\in I}(x_i^k)^2. \tag{4.16}$$

Since $\{y^k\}$ is bounded by definition, $\{F_i(y^k)\}$ is also bounded. Therefore, since $x_i^k \to +\infty$ for all $i \in I$, (4.16) implies

$$\sum_{i\in I} F_i(x^k)^2 \to \infty.$$

As shown at the beginning of the proof of Proposition 4.1, $f_C^i(x^k) = \dfrac{1}{2\delta_i}F_i(x^k)^2 \geq 0$ if $F_i(x^k) - \delta_i x_i^k \leq 0$, and $f_C^i(x^k) \geq \dfrac{\delta_i}{2}(x_i^k)^2$ if $F_i(x^k) - \delta_i x_i^k > 0$. Therefore we have

$$\begin{aligned} f_C(x^k) &= \sum_{i=1}^{n} f_C^i(x^k) \\ &\geq \sum_{i\in I} f_C^i(x^k) \\ &\geq \sum_{i\in I} \frac{1}{2\delta_i} \min\left(F_i(x^k)^2, (\delta_i x_i^k)^2\right). \end{aligned}$$

Since $x_i^k \to +\infty$ for all $i \in I$ and $\sum_{i\in I} F_i(x^k)^2 \to \infty$, it follows that $f_C(x^k) \to +\infty$. □

4.3 Algorithms

In this section, we propose two globally convergent methods for solving the complementarity problem (4.1). One is based on the method of Fukushima [Fuk92] and the other Algorithm 3.1 of Chapter 3, both of which were originally proposed for the variational inequality problem. Throughout this section, we suppose that the mapping F is strongly monotone on R_+^n with modulus $\mu > 0$ (see (4.13)).

4.3.1 Descent method

The first method uses the vector

$$\begin{aligned} d &= H_C(x) - x \\ &= \max\left(0, x - D^{-1}F(x)\right) - x \end{aligned} \tag{4.17}$$

as a search direction at x. When the mapping F is strongly monotone, it can be shown that the vector d given by (4.17) is a descent direction.

Lemma 4.2 *If F is strongly monotone with modulus μ, then the vector d given by (4.17) satisfies the descent condition*

$$\langle \nabla f_C(x), d\rangle \leq -\mu \|d\|^2.$$

Proof. From (4.11) and (4.12), we have

$$\langle \nabla f_C(x), H_C(x) - x\rangle \leq -\langle H_C(x) - x, \nabla F(x)(H_C(x) - x)\rangle. \tag{4.18}$$

It follows from Proposition 2.6 that, when F is differentiable and strongly monotone on R_+^n, ∇F satisfies

$$\langle y-x, \nabla F(x)(y-x)\rangle \geq \mu \, \| y-x \|^2 \text{ for all } x, y \geq 0.$$

Therefore, from (4.18) we have

$$\langle \nabla f_C(x), d\rangle \leq -\mu \, \| d \|^2 . \qquad \square$$

Thus the direction d can be used to determine the next iterate by using the following Armijo-type line search rule: Let $\alpha := \beta^{\bar{l}}$, where $\bar{l}$ is the smallest nonnegative integer l such that

$$f_C(x) - f_C(x + \beta^l d) \geq \sigma \beta^l \, \| d \|^2,$$

where $0 < \beta < 1$ and $\sigma > 0$. Note that, in the descent method originally proposed by Fukushima [Fuk92] for the variational inequality problem, the line search only allows step sizes shorter than unity. Here, we propose the algorithm that allows longer step sizes at each iteration.

Algorithm 4.1

Step 0 Choose $x^0 \geq 0$, $\beta_1 > 1$, $0 < \beta_2 < 1$, $\sigma > 0$, and a positive diagonal matrix D.

Let $k := 0$.

Step 1 Set $d^k := \max(0, x^k - D^{-1}F(x^k)) - x^k$ and $\bar{\alpha}_k := \max\{s \mid x^k + sd^k \geq 0, \ s \geq 0\}$.

Let $l := 0$.

Step 2a If $f_C(x^k) - f_C(x^k + d^k) \geq \sigma \, \| d^k \|^2$, then set $\alpha_k := \beta_1^{l_k}$ where l_k is the largest nonnegative integer l such that

$$\beta_1^l \leq \bar{\alpha}_k, \quad f_C(x^k) - f_C(x^k + \beta_1^l d^k) \geq \sigma \beta_1^l \, \| d^k \|^2$$
$$\text{and } f_C(x^k + \beta_1^{l+1} d^k) > f_C(x^k + \beta_1^l d^k).$$

Go to Step 3.

Step 2b Otherwise set $\alpha_k := \beta_2^{l_k}$ where l_k is the smallest nonnegative integer l such that

$$f_C(x^k) - f_C(x^k + \beta_2^l d^k) \geq \sigma \beta_2^l \, \| d^k \|^2 .$$

Step 3 Set $x^{k+1} := x^k + \alpha_k d^k$. Let $k := k+1$. Return to Step 1.

Note that the vector $H_C(x^{k+1}) = \max(0, x^{k+1} - D^{-1}F(x^{k+1}))$ has already been found at the previous iteration as a by-product of evaluating f_C. Therefore one need not compute again the search direction d^k at the beginning of each iteration.

Theorem 4.1 *Let F be the mapping from R^n into itself. Suppose that F is continuously differentiable and strongly monotone with modulus μ on R^n_+. Suppose also that ∇F is Lipschitz continuous on any bounded subset of R^n_+. Then, Algorithm 4.1 is globally convergent if the positive constant σ is chosen to be sufficiently small such that $\sigma < \mu$.*

Proof. By Proposition 4.4, the level set $B = \{x \mid f_C(x) \le f_C(x^0)\}$ is bounded. Hence ∇F is Lipschitz continuous on B. Since F is continuously differentiable, it is easy to show that F is also Lipschitz continuous on B. Under these conditions, it is not difficult to show that ∇f_C is Lipschitz continuous on B, i.e., there exists a constant $L > 0$ such that

$$\| \nabla f_C(x) - \nabla f_C(y) \| \le L \, \| x - y \| \quad \text{for all} \quad x, y \in S.$$

Therefore, as shown in the proof of [Fuk92, Theorem 4.2], any accumulation point of $\{x^k\}$ satisfies $x = H_C(x)$, and hence solves (4.1) by Lemma 4.1. Since strong monotonicity of F ensures that problem (4.1) has a unique solution, we can conclude that the entire sequence converges to the unique solution of (4.1). □

Remark 4.2 In Fukushima [Fuk92], the global convergence theorem assumes not only the strong monotonicity of mapping F but the *compactness* of the constraint set, which is not the case for the nonlinear complementarity problem. Theorem 4.1 above establishes global convergence under the strong monotonicity of F only.

4.3.2 Modification of Newton's method

The second method to solve the complementarity problem is a modification of Newton's method, which incorporates a line search strategy. The basic Newton method for solving the nonlinear complementarity problem (4.1) generates a sequence $\{x^k\}$ such that $x^0 \ge 0$ and x^{k+1} is determined as $x^{k+1} := \bar{x}$, where $\bar{x}$ is a solution to the following linearized complementarity problem (see Section 2.3):

$$\begin{aligned} & x \ge 0, \; F(x^k) + \nabla F(x^k)^t (x - x^k) \ge 0 \\ & \text{and} \; \left\langle x, F(x^k) + \nabla F(x^k)^t (x - x^k) \right\rangle = 0. \end{aligned} \tag{4.19}$$

It can be shown that, when F is monotone, the Newton direction $d^k := \bar{x} - x^k$ obtained by solving the linearized complementarity problem (4.19) is a feasible descent direction of f_C.

Lemma 4.3 *When the mapping F is strongly monotone with modulus μ, the vector $d^k := \bar{x} - x^k$ obtained by solving the linearized complementarity problem (4.19) satisfies the inequality*

$$\left\langle \nabla f_C(x^k), d^k \right\rangle < - \left(\mu - \frac{1}{4} \| D \| \right) \| d^k \|^2 .$$

Therefore, d^k is actually a feasible descent direction of f_C at x^k, if the matrix D is chosen to satisfy $\| D \| = \max_i(\delta_i) < 4\mu$.

Proof. For simplicity of presentation, we omit the superscript k in x^k and d^k. Since $d := \bar{x} - x$, it follows from (4.7) that

$$\langle \nabla f_C(x), d \rangle \; = \; \langle F(x), \bar{x} - x \rangle + \langle (\nabla F(x) - D)(x - H_C(x)), \bar{x} - x \rangle$$

$$
\begin{aligned}
&= \left\langle F(x) + \nabla F(x)^t(\bar{x} - x), \bar{x} - x \right\rangle - \left\langle \bar{x} - x, \nabla F(x)^t(\bar{x} - x) \right\rangle \\
&\quad + \left\langle F(x) + \nabla F(x)^t(\bar{x} - x), x - H_C(x) \right\rangle \\
&\quad - \langle F(x), x - H_C(x) \rangle - \langle \bar{x} - x, D(x - H_C(x)) \rangle \\
&= -\left\langle F(x) + \nabla F(x)^t(\bar{x} - x), H_C(x) - \bar{x} \right\rangle \\
&\quad + \langle F(x), H_C(x) - x \rangle + \langle \bar{x} - x, D(H_C(x) - x) \rangle \\
&\quad - \left\langle \bar{x} - x, \nabla F(x)^t(\bar{x} - x) \right\rangle. \qquad (4.20)
\end{aligned}
$$

Since $\bar{x}$ is a solution to (4.19) and $H_C(x) \geq 0$, the first term of (4.20) is nonpositive. From (4.12), we have

$$\langle F(x), H_C(x) - x \rangle \leq -\langle H_C(x) - x, D(H_C(x) - x) \rangle.$$

Then it follows from the second term of (4.20) that

$$
\begin{aligned}
&\langle F(x), H_C(x) - x \rangle + \langle \bar{x} - x, D(H_C(x) - x) \rangle \\
&\leq \langle \bar{x} - x, D(H_C(x) - x) \rangle - \langle H_C(x) - x, D(H_C(x) - x) \rangle \\
&= \sum_{i=1}^{n} \delta_i \left\{ (\bar{x}_i - x_i)((H_C)_i(x) - x_i) - ((H_C)_i(x) - x_i)^2 \right\} \\
&= \sum_{i=1}^{n} \frac{\delta_i}{2} \left\{ (\bar{x}_i - x_i)^2 - ((H_C)_i(x) - x_i)^2 - (\bar{x}_i - (H_C)_i(x))^2 \right\} \\
&\leq \sum_{i=1}^{n} \frac{\delta_i}{2} \left\{ (\bar{x}_i - x_i)^2 - \frac{1}{2}(\bar{x}_i - x_i)^2 \right\} \\
&= \sum_{i=1}^{n} \frac{\delta_i}{4} (\bar{x}_i - x_i)^2 \\
&= \frac{1}{4} \langle \bar{x} - x, D(\bar{x} - x) \rangle. \qquad (4.21)
\end{aligned}
$$

Hence, we have from (4.20) and (4.21) that

$$\langle \nabla f_C(x), d \rangle \leq -\left\langle d, \nabla F(x)^t d \right\rangle + \frac{1}{4} \langle d, Dd \rangle.$$

However, since strong monotonicity of F implies $\langle d, \nabla F(x) d \rangle \geq \mu \, \| d \|^2$ and since $\langle d, Dd \rangle \leq \| D \| \| d \|^2$, we have

$$\langle \nabla f_C(x), d \rangle < -\left(\mu - \frac{1}{4} \| D \| \right) \| d \|^2 .$$

The last half of the proposition then follows immediately. □

Using this result, we can construct a modified Newton method for solving the nonlinear complementarity problem (4.1).

Algorithm 4.2

Step 0 Choose $x^0 \geq 0$, $0 < \beta < 1$, $0 < \sigma < \frac{1}{2}$, and a positive diagonal matrix D. Let $k := 0$.

Step 1 Find the unique solution $\bar{x}^k$ that satisfies

$$\bar{x}^k \geq 0,\ F(x^k) + \nabla F(x^k)^t(\bar{x}^k - x^k) \geq 0,$$
$$\text{and } \left\langle \bar{x}^k, (F(x^k) + \nabla F(x^k)^t(\bar{x}^k - x^k)) \right\rangle = 0.$$

Let $d^k := \bar{x}^k - x^k$.

Step 2 Set $\alpha_k := \beta^{l_k}$ where l_k is the smallest nonnegative integer l such that

$$f_C(x^k) - f_C(x^k + \beta^l d^k) \geq -\sigma\beta^l \left\langle \nabla f_C(x^k), d^k \right\rangle.$$

Step 3 Set $x^{k+1} := x^k + \alpha_k d^k$. Let $k := k+1$. Return to Step 1.

When the mapping F is strongly monotone, we can establish the global convergence of Algorithm 4.2.

Theorem 4.2 *Let F be a mapping from R^n into itself and D be a positive definite matrix defined by (4.4). Suppose that the mapping F is continuously differentiable and strongly monotone with modulus μ. If the matrix D is chosen such that $\| D \| = \max_i(\delta_i) < 4\mu$, then, Algorithm 4.2 is globally convergent.*

Proof. By Theorem 4.3 and the Armijo line search rule, the sequence $\{f_C(x^k)\}$ is nonincreasing. It then follows from Proposition 4.4 that the sequence $\{x^k\}$ is bounded, and hence it contains at least one accumulation point. As shown in the proof of Theorem 3.2, any accumulation point of $\{x^k\}$ is a solution of (4.1). Since strong monotonicity of F ensures that problem (4.1) has a unique solution, we can conclude that the entire sequence converges to the unique solution of (4.1). □

We can also show that the rate of convergence of Algorithm 4.2 is quadratic if $F \in C^2$, that is, all F_i, $i = 1, \ldots, n$ are twice continuously differentiable, and the strict complementarity condition holds at the unique solution x^* of (4.1).

Theorem 4.3 *Let F be a mapping from R^n into itself. Suppose that the sequence $\{x^k\}$ generated by Algorithm 4.2 converges to the solution x^* to the nonlinear complementarity problem (4.1). Suppose also that the mapping F belongs to class C^2, $\nabla F(x^*)$ is positive definite and $\nabla^2 F$ is Lipschitz continuous on some neighborhood of x^*. If the strict complementarity condition holds at x^*, i.e., $x_i^* = 0$ implies $F_i(x^*) > 0$ for all $i = 1, \ldots, n$, then there exists an integer $\bar{k}$ such that the unit step size is accepted for all $k \geq \bar{k}$. Therefore, the sequence $\{x^k\}$ converges quadratically to the solution x^*.*

Before proving Theorem 4.3, we show the following lemma.

Lemma 4.4 *Let x^* be a solution to problem (4.1). If $F \in C^2$ and the strict complementarity condition holds at x^*, then f_C^i is twice continuously differentiable on a neighborhood of x^*, and the gradient and the Hessian of f_C^i are given by*

$$\nabla f_C^i(x) = \begin{cases} (x_i \nabla F_i(x) + F_i(x)e_i) - \delta_i x_i e_i & \text{if } i \in I^* \quad (4.22a) \\ \frac{1}{\delta_i} F_i(x)\nabla F_i(x) & \text{if } i \in \bar{I}^* \quad (4.22b) \end{cases}$$

and

$$\nabla^2 f_C^i(x) = \begin{cases} x_i \nabla^2 F_i(x) + 2\nabla F_i(x) e_i^t - \delta_i e_i e_i^t & \text{if } i \in I^* \quad (4.23a) \\ \frac{1}{\delta_i}\left(F_i(x)\nabla^2 F_i(x) + \nabla F_i(x) \nabla F_i(x)^t\right) & \text{if } i \in \bar{I}^*, \quad (4.23b) \end{cases}$$

respectively, where $I^* = \{i \mid x_i^* = 0\}$ *and* $\bar{I}^* = \{i \mid x_i^* > 0\}$.

Proof. We have from the strict complementarity and $\delta_i > 0$ that

$$\begin{cases} F_i(x^*) - \delta_i x_i^* > 0 & \text{if } i \in I^* \\ F_i(x^*) - \delta_i x_i^* < 0 & \text{if } i \in \bar{I}^* \end{cases}$$

Hence the continuity of F ensure that there is a neighborhood X^* of x^* such that

$$\begin{cases} F_i(x) - \delta_i x_i > 0 & \text{if } i \in I^* \\ F_i(x) - \delta_i x_i < 0 & \text{if } i \in \bar{I}^* \end{cases} \qquad (4.24)$$

holds for all $x \in X^*$. Hence, we have from (4.3)

$$f_C^i(x) = \begin{cases} x_i F_i(x) - \frac{\delta_i}{2} x_i^2 & \text{if } i \in I^* \quad (4.25a) \\ \frac{1}{2\delta_i} F_i(x)^2 & \text{if } i \in \bar{I}^*. \quad (4.25b) \end{cases}$$

Therefore, by differentiating (4.25a) and (4.25b) directly, we have (4.22a), (4.22b) and (4.23a), (4.23b). □

Proof of Theorem 4.3 It is sufficient to show that

$$f_C(x^k) - f_C(\bar{x}^k) \geq -\sigma \left\langle \nabla f_C(x^k), \bar{x}^k - x^k \right\rangle$$

holds for a sufficiently large k. For simplicity, we consider the case of $\delta_1 = \cdots = \delta_n = \delta > 0$, i.e., the diagonal matrix D is the identity matrix multiplied by $\delta > 0$. It is not difficult to extend the result to the general case. Without loss of generality, we assume $\bar{I}^* = \{j, j+1, \ldots, n\}$, where $1 \leq j \leq n$, and denote

$$x = \begin{pmatrix} x_{I^*} \\ x_{\bar{I}^*} \end{pmatrix}, \quad F(x) = \begin{pmatrix} F_{I^*}(x) \\ F_{\bar{I}^*}(x) \end{pmatrix}, \quad \nabla F(x) = \begin{pmatrix} \nabla_{I^*} F_{I^*}(x) & \nabla_{I^*} F_{\bar{I}^*}(x) \\ \nabla_{\bar{I}^*} F_{I^*}(x) & \nabla_{\bar{I}^*} F_{\bar{I}^*}(x) \end{pmatrix}.$$

Since the strict complementarity holds at x^*, there is an integer K_1 such that x^k satisfies

$$\begin{cases} F_i(x^k) - \delta x_i^k > 0 & \text{if } i \in I^* \quad (4.26a) \\ F_i(x^k) - \delta x_i^k < 0 & \text{if } i \in \bar{I}^* \quad (4.26b) \end{cases}$$

for all $k \geq K_1$. Under the given assumptions, Newton's method (4.19) is locally quadratically convergent to the solution x^* [PaC82]. Hence, it follows from the strict complementarity and the continuity of F and ∇F, that there is an integer K_2 such that

$$\begin{cases} \bar{x}_i^k = 0 \text{ and } F_i(x^k) + \left\langle \nabla F_i(x^k), \bar{x}^k - x^k \right\rangle > 0 & \text{if } i \in I^* \quad (4.27a) \\ \bar{x}_i^k > 0 \text{ and } F_i(x^k) + \left\langle \nabla F_i(x^k), \bar{x}^k - x^k \right\rangle = 0 & \text{if } i \in \bar{I}^* \quad (4.27b) \end{cases}$$

for all $k \geq K_2$.

Now suppose $k \geq \max(K_1, K_2)$. For simplicity of presentation, we omit superscript k in x^k and $\bar{x}^k$. For each $i \in I^*$, we have

$$\begin{aligned} & f_C^i(x) - f_C^i(\bar{x}) + \sigma \left\langle \nabla f_C^i(x), \bar{x} - x \right\rangle \\ &= \left(x_i F_i(x) - \frac{\delta}{2} x_i^2\right) - \left(\bar{x}_i F_i(\bar{x}) - \frac{\delta}{2} \bar{x}_i^2\right) \\ &\quad + \sigma \left\langle x_i \nabla F_i(x) + F_i(x) e_i - \delta x_i e_i, \bar{x} - x \right\rangle \\ &= x_i F_i(x) - \frac{\delta}{2} x_i^2 + \sigma \left(x_i \left\langle \nabla F_i(x), \bar{x} - x \right\rangle - x_i F_i(x) + \delta x_i^2\right) \\ &\geq x_i F_i(x) - \frac{\delta}{2} x_i^2 + \sigma \left(-2 x_i F_i(x) + \delta x_i^2\right) \\ &= (1 - 2\sigma) \left(x_i F_i(x) - \frac{\delta}{2} x_i^2\right) \end{aligned}$$

$$\geq \left(\frac{1}{2}-\sigma\right)\delta x_i^2$$
$$= \left(\frac{1}{2}-\sigma\right)\delta(\bar{x}_i - x_i)^2, \tag{4.28}$$

where the first equality follows from (4.22a) and (4.25a), the second equality and the first inequality follow from (4.27a), the second inequality follows from (4.26a) and the last equality follows from (4.27a).

On the other hand, since

$$f_C^i(\bar{x}) - f_C^i(x) = \left\langle \nabla f_C^i(x), \bar{x}-x \right\rangle + \frac{1}{2}\left\langle \bar{x}-x, \nabla^2 f_C^i(\xi)(\bar{x}-x)\right\rangle$$

hold for some ξ in line segment of x and $\bar{x}$ by the mean value theorem, we have

$$\begin{aligned}
&f_C^i(x) - f_C^i(\bar{x}) + \sigma\left\langle \nabla f_C^i(x), \bar{x}-x\right\rangle \\
&= (\sigma-1)\left\langle \nabla f_C^i(x), \bar{x}-x\right\rangle - \frac{1}{2}\left\langle \bar{x}-x, \nabla^2 f_C^i(\xi)(\bar{x}-x)\right\rangle \\
&= (\sigma-1)\left\langle \nabla f_C^i(x), \bar{x}-x\right\rangle - \frac{1}{2}\left\langle \bar{x}-x, \nabla^2 f_C^i(x)(\bar{x}-x)\right\rangle \\
&\quad + \frac{1}{2}\left\langle \bar{x}-x, \left(\nabla^2 f_C^i(x) - \nabla^2 f_C^i(\xi)\right)(\bar{x}-x)\right\rangle .
\end{aligned} \tag{4.29}$$

Then for each $i \in \bar{I}^*$, we have

$$\begin{aligned}
&(\sigma-1)\left\langle \nabla f_C^i(x), \bar{x}-x\right\rangle - \frac{1}{2}\left\langle \bar{x}-x, \nabla^2 f_C^i(x)(\bar{x}-x)\right\rangle \\
&= \frac{\sigma-1}{\delta}F_i(x)\langle \nabla F_i(x), \bar{x}-x\rangle \\
&\quad - \frac{1}{2\delta}\left\langle \bar{x}-x, \left(F_i(x)\nabla^2 F_i(x) + \nabla F_i(x)\nabla F_i(x)^t\right)(\bar{x}-x)\right\rangle \\
&= \frac{1}{\delta}\left(\frac{1}{2}-\sigma\right)\langle \nabla F_i(x), (\bar{x}-x)\rangle^2 - \frac{1}{2\delta}F_i(x)\left\langle \bar{x}-x, \nabla^2 F_i(x)(\bar{x}-x)\right\rangle,
\end{aligned} \tag{4.30}$$

where the first equality follows from (4.22b) and (4.23b), and the second equality follows from (4.27b). Hence, we have from (4.28) and (4.30) that

$$\begin{aligned}
&f_C^i(x) - f_C^i(\bar{x}) + \sigma\left\langle \nabla f_C^i(x), \bar{x}-x\right\rangle \\
&= \frac{1}{\delta}\left(\frac{1}{2}-\sigma\right)\langle \nabla F_i(x), (\bar{x}-x)\rangle^2 - \frac{1}{2\delta}F_i(x)\left\langle \bar{x}-x, \nabla^2 F_i(x)(\bar{x}-x)\right\rangle \\
&\quad + \frac{1}{2}\left\langle \bar{x}-x, \left(\nabla^2 f_C^i(x) - \nabla^2 f_C^i(\xi)\right)(\bar{x}-x)\right\rangle \\
&= \frac{1}{\delta}\left(\frac{1}{2}-\sigma\right)\langle \nabla F_i(x^*), \bar{x}-x\rangle^2 - \frac{F_i(x)}{2\delta}\left\langle \bar{x}-x, \nabla^2 F_i(x)(\bar{x}-x)\right\rangle \\
&\quad + \frac{1}{2}\left\langle \bar{x}-x, \left(\nabla^2 f_C^i(x) - \nabla^2 f_C^i(\xi)\right)(\bar{x}-x)\right\rangle \\
&\quad + \frac{1}{\delta}\left(\frac{1}{2}-\sigma\right)\left\{\langle \nabla F_i(x), \bar{x}-x\rangle^2 - \langle \nabla F_i(x^*), \bar{x}-x\rangle^2\right\}.
\end{aligned}$$

When $\nabla^2 F$ is Lipschitz continuous and is bounded on some neighborhood of x^*, it is not difficult to show that $\nabla^2 f_C^i$ and ∇F_i are also Lipschitz continuous. Moreover, for $i \in \bar{I}^*$ we have $F_i(x) \to 0$ if $x \to x^*$. Hence,

$$\begin{aligned}
&f_C^i(x) - f_C^i(\bar{x}) + \sigma\left\langle \nabla f_C^i(x), \bar{x}-x\right\rangle \\
&\geq \frac{1}{\delta}\left(\frac{1}{2}-\sigma\right)\langle \nabla F_i(x^*), \bar{x}-x\rangle^2 + O\left(\|x-x^*\| + \|\bar{x}-x\|\right)\|\bar{x}-x\|^2
\end{aligned} \tag{4.31}$$

holds on some neighborhood of x^*.

Therefore, it follows from (4.28) and (4.31) that

$$\begin{aligned}
&f_C(x) - f_C(\bar{x}) + \sigma\langle \nabla f_C(x), \bar{x}-x\rangle \\
&\geq \delta\left(\frac{1}{2}-\sigma\right)\sum_{i\in I^*}(\bar{x}_i - x_i)^2 + \frac{1}{\delta}\left(\frac{1}{2}-\sigma\right)\sum_{i\in \bar{I}^*}\langle \nabla F_i(x), \bar{x}-x\rangle^2 \\
&\quad + O\left(\|x-x^*\| + \|\bar{x}-x\|\right)\|\bar{x}-x\|^2
\end{aligned}$$

$$= \frac{1}{\delta}\left(\frac{1}{2}-\sigma\right)\langle \bar{x}-x, J(\bar{x}-x)\rangle + O\left(\|x-x^*\| + \|\bar{x}-x\|\right)\|\bar{x}-x\|^2, \quad (4.32)$$

where J is a matrix such that

$$\begin{aligned} J &= \begin{pmatrix} \delta^2 E_j & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & \nabla_{I^*}F_{\bar{I}^*}(x^*) \\ 0 & \nabla_{\bar{I}^*}F_{\bar{I}^*}(x^*) \end{pmatrix} \begin{pmatrix} 0 & 0 \\ \nabla_{I^*}F_{\bar{I}^*}(x^*)^t & \nabla_{\bar{I}^*}F_{\bar{I}^*}(x^*)^t \end{pmatrix} \\ &= \begin{pmatrix} \delta E_j & \nabla_{I^*}F_{\bar{I}^*}(x^*) \\ 0 & \nabla_{\bar{I}^*}F_{\bar{I}^*}(x^*) \end{pmatrix} \begin{pmatrix} \delta E_j & \nabla_{I^*}F_{\bar{I}^*}(x^*) \\ 0 & \nabla_{\bar{I}^*}F_{\bar{I}^*}(x^*) \end{pmatrix}^t. \end{aligned}$$

Clearly J is positive semi-definite. Moreover, since $\nabla F(x^*)$ is positive definite by assumption, the matrix $\nabla_{\bar{I}^*}F_{\bar{I}^*}(x^*)$ is also positive definite. Hence, matrix $\begin{pmatrix} \delta E_l & \nabla_{I^*}F_{\bar{I}^*}(x^*) \\ 0 & \nabla_{\bar{I}^*}F_{\bar{I}^*}(x^*) \end{pmatrix}$ is nonsingular, and hence J is positive definite.

Therefore, (4.32) is strictly positive provided that x is sufficiently close to x^*. □

Remark 4.3 In Chapter 3, we have obtained a globally convergent Newton method, Algorithm 3.1, for the variational inequality problem. In Algorithm 3.1, to obtain quadratic convergence, the following line search procedure was used:

Step 2 Let $0 < \beta < 1$, $0 < \gamma < 1$ and $\sigma \in (0, 1)$.

If $f_C(x^k + d^k) \le \gamma f_C(x^k)$, then set $\alpha_k := 1$ and go to Step 3.

Otherwise set $\alpha_k := \beta^{l_k}$ where l_k is the smallest nonnegative integer l such that

$$f_C(x^k) - f_C(x^k + \beta^l d^k) \ge -\sigma\beta^l \left\langle \nabla f_C(x^k), d^k \right\rangle.$$

Note that this line search procedure, which is similar to the one used by Marcotte and Dussault [MaD89], first checks if the unit step size is acceptable. On the other hand, Algorithm 4.2 employs the Armijo rule in a more direct manner.

4.4 Computational results I

In the following two sections, we report some numerical results for Algorithms 4.1 and 4.2 discussed in the previous sections. In this section, we present the results for a strongly monotone problem. All computer programs were coded in FORTRAN and the runs in this section were made in double precision on a personal computer called Fujitsu FMR-70.

Throughout the computational experiments, the parameters used in the algorithms were set to $\beta_1 = 2$, $\beta_2 = 0.5$, $\gamma = 0.5$ and $\sigma = 0.0001$. The positive diagonal matrix D was chosen to be the identity matrix multiplied by a positive parameter $\delta > 0$. Therefore the merit function (4.2) can be written simply as

$$f_C(x) = \frac{1}{2\delta}\sum_{i=1}^{n}\left\{F_i(x)^2 - (\max(0, F_i(x) - \delta x_i))^2\right\}. \quad (4.33)$$

The search direction of Algorithm 4.1 can also be written as

$$d^k := \max\left(0, x^k - \frac{1}{\delta}F(x^k)\right) - x^k.$$

The convergence criterion was

$$|\min(x_i, F_i(x))| \le 10^{-5} \quad \text{for all } i = 1, \ldots, n.$$

For comparison purposes, we also tested two popular methods for solving the nonlinear complementarity problem, the projection method [Daf80] and the basic Newton method (cf. Section 2.3). The projection method generates a sequence $\{x^k\}$ such that $x^0 \ge 0$ and x^{k+1} is determined from x^k by

$$x^{k+1} := \max\left(0, x^k - \frac{1}{\delta}F(x^k)\right), \tag{4.34}$$

for all k. Note that this method may be considered a fixed step-size variant of Algorithm 4.1. When the mapping F is strongly monotone and Lipschitz continuous with constants μ and L, respectively, this method is globally convergent if δ is chosen large enough to satisfy $\delta > L^2/2\mu$ (see [PaC82, Corollary 2.11.]).

The mappings tested in this section are of the form

$$F(x) = E_n x + \rho(V - V^t)x + \Psi(x) + q, \tag{4.35}$$

where E_n is the $n \times n$ identity matrix, V is an $n \times n$ matrix such that each row contains only one nonzero element, and $\Psi(x)$ is a nonlinear monotone mapping with components $\Psi_i(x_i) = p_i x_i^4$, where p_i are positive constants. Elements of matrix V and vector q as well as coefficients p_i are randomly generated from uniform distributions such that $-5 \le V_{ij} \le 5$, $-25 \le q_i \le 25$ and $0.001 \le p_i \le 0.006$. The results are shown in Tables 4.1 $\sim$ 4.4. All initial iterates were chosen to be $(0, 0, \ldots, 0)$. In the tables, $\#f_C$ is the total number of evaluating the merit function f_C. All CPU times are in seconds and exclude input/output times. The parameter ρ in (4.35) is used to change the degree of asymmetry of F; namely F deviates from symmetry as ρ becomes large. Since the matrix $E_n + \rho(V - V^t)$ is positive definite for any ρ and $\Psi_i(x_i)$ are monotonically increasing for $x_i \ge 0$, the mapping F defined by (4.35) is strongly monotone on R^n_+.

4.4.1 Comparison of Algorithm 4.1 and the projection method

First we compare Algorithms 4.1 and the projection method (4.34) by using a 10-dimensional example, in which mapping F is given by

$$F(x) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & -2 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2 & -5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 1 & 0 & -5 & 0 & 0 \\ 0 & 0 & -3 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ -5 & 0 & 0 & 0 & 0 & 5 & 0 & 1 & 0 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -5 & 4 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{pmatrix}$$

$$+ \begin{pmatrix} 0.004x_1^4 \\ 0.004x_2^4 \\ 0.003x_3^4 \\ 0.003x_4^4 \\ 0.006x_5^4 \\ 0.006x_6^4 \\ 0.004x_7^4 \\ 0.004x_8^4 \\ 0.004x_9^4 \\ 0.002x_{10}^4 \end{pmatrix} + \begin{pmatrix} 2 \\ 10 \\ 2 \\ 9 \\ -15 \\ 12 \\ -9 \\ 5 \\ 7 \\ -17 \end{pmatrix}.$$

The results for this problem are shown in Table 4.1.

In general, the projection method is guaranteed to converge only if the parameter δ is chosen sufficiently large. In fact, Table 4.1 shows that when δ is large, the projection

Table 4.1: Comparison of Algorithm 4.1 and the projection method ($n = 10$, $\rho = 1$)

	Algorithm 4.1			projection method*	
δ	#Iterations	$\#f_C$	CPU	#Iterations	CPU
0.1	1380	9683	18.45	–	–
0.5	307	1527	3.04	–	–
1	328	1305	2.63	–	–
2	338	1009	2.13	–	–
3	353	951	2.04	–	–
4	342	685	1.55	–	–
5	256	511	1.16	–	–
6	351	701	1.72	–	–
6.2	337	674	1.66	–	–
6.3	377	754	1.91	9118	13.94
6.5	376	752	1.93	1594	2.43
7	385	770	2.07	610	0.93
8	337	676	1.98	338	0.51
9	270	542	1.59	271	0.43
10	242	487	1.41	244	0.36
12	232	468	1.36	229	0.34
15	239	488	1.41	239	0.35
20	254	636	1.72	272	0.41
50	229	920	2.17	549	0.79
100	229	1149	2.63	1036	1.47
200	239	1385	3.05	2008	2.88
500	239	1674	3.59	5007	7.08
1000	372	2605	5.55	9998	14.18

*The projection method failed to converge for the value of δ up to 6.2.

method is always convergent, but as δ becomes small, the behavior of the method found to be unstable and eventually it fails to converge.

Table 4.1 also shows that Algorithm 4.1 is always convergent even if δ is chosen small, since the line search determines an adequate step size at each iteration. In Algorithm 4.1, the number of iterations is almost constant. This is because we may choose a larger step size when the magnitude of vector d^k is small, i.e. δ is large.

Algorithm 4.1 spends more CPU times per iteration than the projection method, because the former algorithm requires overheads of evaluating the merit function f_C. But, when δ becomes large, Algorithm 4.1 tends to spend less CPU time than the projection method, because the number of iterations of Algorithm 4.1 increases mildly.

4.4.2 Comparison of Algorithm 4.2 and Newton's method

Next we compare Algorithm 4.2 and the basic Newton method. For each of the problem sizes $n = 30, 50$ and 90, we randomly generated five test problems. The parameters ρ and δ were set to $\rho = 1$ and $\delta = 1$. The initial iterate was chosen to be $x = 0$. In solving the linearized subproblem at each iteration of Algorithm 4.2 and Newton's method, we used Lemke's complementarity pivoting method [Lem65] coded by Fukushima [IbF91]. All parameters and initial iterates were set to the default values used in [IbF91]. The results are given in Table 4.2. All numbers shown in Table 4.2 are the averages of the results for five test problems, each case and #Lemke is the total number of pivotings in Lemke's method.

Table 4.2 shows that the number of iterations of Newton's method is consistently larger than that of Algorithm 4.2 as far as the test problems used in the experiments are concerned. Therefore, since it is time consuming to solve a linear subproblem at each iteration, Algorithm 4.2 required less CPU time than Newton's method in spite of the overheads in line search. Finally we note that Newton's method (4.19) is not guaranteed to be globally convergent, although it actually converged for all test problems reported in Table 4.2.

Table 4.2: Comparison of Algorithm 4.2 and Newton's method ($\rho = 1$)

	Algorithm 4.2				Newton's method		
n	#Iterations	$\#f_C$	#Lemke	CPU	#Iterations	#Lemke	CPU
30	5.6	7.6	80.6	4.294	8	115.2	5.840
50	5.6	7.6	156.0	19.880	8	216.4	26.142
90	6.0	8.0	275.2	105.400	8	358.8	135.690

4.4.3 Comparison of Algorithms 4.1 and 4.2

Finally we compare Algorithms 4.1 and 4.2. Test problems are the same as ones in the previous section. To see how these algorithms behave for different degrees of asymmetry of the mapping F, we have tested several values of ρ between 0.1 and 2.0. The initial iterates was always chosen to be $x = 0$. The results are given in Table 4.3. All numbers shown in Table 4.3 are the averages of the results for five test problems.

Table 4.3 shows that when the mapping F is close to symmetry, Algorithm 4.1 converges very quickly, and when the mapping becomes asymmetric, the number of iterations and CPU time of Algorithm 4.1 increase rapidly. On the other hand, in

Table 4.3: Comparison of Algorithms 4.1 and 4.2

		Algorithm 4.1			Algorithm 4.2				
ρ	n	#Ite.*	$\#f_C$	CPU	#Ite.*	$\#f_C$	#Lemke	CPU	LEMKE
	30	31.6	96.0	0.672	6.0	8.4	87.4	4.956	4.094
0.1	50	28.4	84.6	0.966	6.0	9.2	148.0	19.050	17.989
	90	38.2	114.0	2.322	6.2	8.2	259.0	99.156	96.721
	30	40.0	119.6	0.838	6.2	8.2	92.6	4.872	4.340
0.2	50	37.6	110.0	1.264	6.4	8.8	162.6	20.888	19.633
	90	40.4	120.4	2.448	6.6	8.6	277.2	106.672	103.201
	30	33.8	99.2	0.690	6.4	8.4	90.8	4.812	4.258
0.3	50	39.2	112.2	1.292	6.2	8.6	157.8	20.270	19.055
	90	41.4	119.6	2.408	6.6	8.6	275.2	105.890	102.253
	30	45.8	127.0	0.892	6.0	8.2	88.2	4.662	4.128
0.5	50	58.6	161.8	1.848	6.0	8.0	157.8	20.242	19.134
	90	110.8	273.8	5.780	6.6	8.6	281.4	108.260	105.073
	30	152.8	322.0	2.436	5.8	7.8	84.4	4.474	3.956
0.8	50	290.4	586.4	7.450	6.0	8.0	160.4	20.476	19.368
	90	780.2	1557.4	33.960	6.0	8.0	269.4	103.266	100.296
	30	394.2	792.4	5.270	5.6	7.6	80.6	4.294	3.784
1.0	50	519.6	1077.6	11.068	5.6	7.6	156.0	19.880	18.833
	90	866.0	2129.6	35.880	6.0	8.0	275.2	105.400	102.107
	30	1197.0	3793.2	21.518	5.4	7.4	80.0	4.266	3.752
1.5	50	1604.0	4927.4	45.280	5.2	7.2	145.8	18.648	17.607
	90	2928.0	9777.8	158.162	6.0	8.0	275.6	105.492	102.606
	30	3195.2	12694.8	69.636	5.2	7.2	79.0	4.168	3.712
2.0	50	3842.6	15929.0	141.944	5.2	7.2	145.2	18.494	17.336
	90	4957.6	20905.0	332.510	5.8	7.8	279.0	106.514	103.618

*#Ite. denotes the total number of iterations.

Algorithm 4.2, while the total number of pivotings of Lemke's method increases in proportion to problem size n, the number of iterations stays constant even when the problem size and the degree of asymmetry of F are varied. Hence, when the degree of asymmetry of F is relatively small, that is, when ρ is smaller than 1.0 in our test problems, Algorithm 4.1 requires less CPU time than Algorithm 4.2.

Note that, since the mapping F used in our computational experience is sparse, complexity of each iteration in Algorithm 4.1 is small. On the other hand, the code [IbF91] of Lemke's method used in Algorithm 4.2 to solve a linear subproblem does not make use of sparsity. Moreover, since the code of Lemke's method is restrictive in the choice of initial iterates, we must restart from a priori fixed initial iterate at each iteration even when the iterate becomes close to a solution. Therefore, it may require a significant amount of CPU time at each iteration for large problems. (In Table 4.3, LEMKE is the total CPU time to solve subproblems by Lemke's method.) If a method that can make use of sparsity and can start from arbitrary point is available to solve a linear subproblem, CPU time of Algorithm 4.2 may decrease. The projected Gauss-Seidel method [CPS92, page 397] for solving the linear complementarity problem is one of such methods. In Table 4.4, results of Algorithm 4.2 using the projected Gauss-Seidel method in place of Lemke's method are given. Table 4.4 shows that, if the mapping F is almost symmetric, Algorithm 4.2 converges very fast. We note that the projected Gauss-Seidel method is not guaranteed to be convergent when a problem is not symmetric. Algorithm 4.2 fails to converge when the degree of asymmetry increased, because the projected Gauss-Seidel method failed to solve linear

Table 4.4: Results for Algorithm 4.2 (Gauss-Seidel version)

		Algorithm 4.2			
ρ	n	#Iterations	$\#f_C$	#Gauss	CPU
0.1	30	6.0	8.4	39.2	0.274
	50	6.0	9.2	39.6	0.466
	90	6.2	8.2	48.0	0.916
0.2	30	3 of 5 failed			
	50	3 of 5 failed			
	90	failed			
$0.3 \sim 2.0$	30	failed			
	50				
	90				

subproblems.

Figure 4.1 illustrates how Algorithms 4.1 and 4.2 converged for two typical test problems with $n = 30$ and 50. In the figure, the vertical axis represents the accuracy of a generated iterate to the solution, which is evaluated by

$$\mathrm{ACC} = \max_i \{ |\min(x_i, F_i(x))| \mid i = 1, \dots, n \}.$$

Figure 4.1 indicates that Algorithm 4.2 is quadratically convergent when the iterates come near the solution. Figure 4.1 also indicates that Algorithm 4.1 is linearly convergent though it has not been proved theoretically.

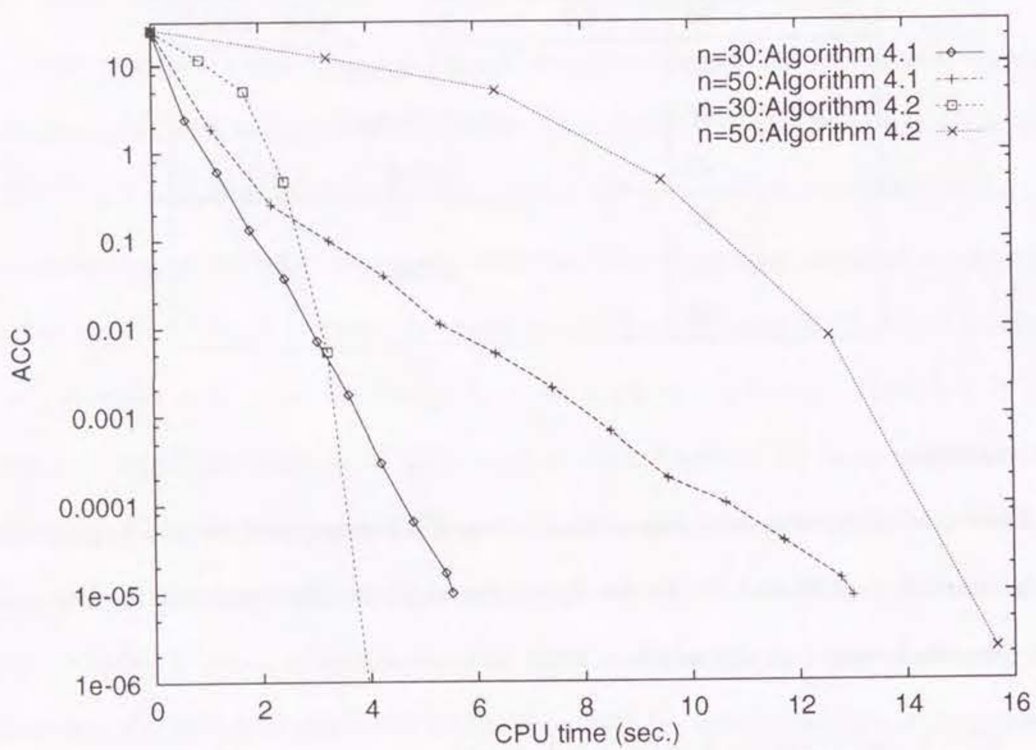

Figure 4.1: Behavior of Algorithms 4.1 and 4.2

4.5 Computational results II

In this section, we present the results of applying Algorithms 4.1 and 4.2 to some examples which arise from an optimization problem, a spatial price equilibrium problem, a noncooperative game and a traffic assignment problem. The algorithms were implemented in FORTRAN and run on a SUN-4 workstation. The parameters in the algorithms were set in the same manner as in Section 4.4. The positive diagonal matrix D was also chosen to be the identity matrix multiplied by $\delta > 0$, and hence the merit function (4.33) was used. The convergence criterion was

$$|\min(x_i, F_i(x))| \leq \mathrm{CC} \quad \text{for all } i = 1, \dots, n,$$

where CC is a parameter used to change accuracy of algorithms. In solving the linearized subproblem of Algorithm 4.2, we used Lemke's complementarity pivoting algorithm coded by Fukushima [IbF91]. The results are shown in Tables 4.5 $\sim$ 4.11.

Some mappings F used in the experiments were only monotone but not strongly monotone. Others were not even monotone, though they could be considered almost monotone. Thus all the problems do not satisfy the convergence conditions of our algorithms. However, for most of the tested case, both Algorithms 4.1 and 4.2 converged and produced satisfactory solutions.

Example 4.1 This is the following 4-variable complementarity problem from Josephy [Jos79a], whose mapping is given by

$$F(x) = \begin{pmatrix} 3x_1^2 + 2x_1x_2 + 2x_2^2 + x_3 + 3x_4 - 6 \\ 2x_1^2 + x_1 + x_2^2 + 3x_3 + 2x_4 - 2 \\ 3x_1^2 + x_1x_2 + 2x_2^2 + 2x_3 + 3x_4 - 1 \\ x_1^2 + 3x_2^2 + 2x_3 + 3x_4 - 3 \end{pmatrix}.$$

The results are shown in Table 4.5. Since the mapping is co-positive but not monotone, Algorithm 4.2 failed when the initial iterates $(0, \ldots, 0)$ and $(10, \ldots, 10)$ were chosen, because the linearized subproblem at $(0, \ldots, 0)$ has no solution and the search direction at $(10, \ldots, 10)$ is not a descent direction. On the other hand, Algorithm 4.1 converged for all of those initial iterates.

Table 4.5: Results for Example 4.1

		Algorithm 4.1			Algorithm 4.2			
CC	Initial Iterate	#Iterations	$\#f_C$	CPU	#Iterations	$\#f_C$	#Lemke	CPU
10^{-5}	$(0, \ldots, 0)$	20	62	0.00	failed			
	$(1, \ldots, 1)$	21	63	0.00	4	5	8	0.00
	$(5, \ldots, 5)$	21	63	0.00	5	6	10	0.00
	$(10, \ldots. 10)$	21	63	0.00	failed			

Example 4.2 This is a 10-variable complementarity problem arising from the Nash-Cournot production problem appeared in Harker [Har88]. In this example, for any $x > 0$, the Jacobian $\nabla F(x)$ of the mapping is a P-matrix, i.e., for any $x \neq 0$, there exists an index $i \in \{1, \ldots, n\}$ such that $x_i(\nabla F(x)x)_i > 0$, but the mapping F is not monotone. Table 4.6 shows that both Algorithms 4.1 and 4.2 converged to the solution

quickly.

Table 4.6: Results for Example 4.2

		Algorithm 4.1			Algorithm 4.2			
CC	Initial Iterate	#Iterations	$\#f_C$	CPU	#Iterations	$\#f_C$	#Lemke	CPU
10^{-5}	$(1, \ldots, 1)$	44	88	0.08	8	10	89	0.08
	$(10, \ldots. 10)$	43	86	0.08	6	7	60	0.05

Example 4.3 This example is the following convex programming problem:

$$\begin{array}{ll} \text{minimize} & (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + 10x_5^4 \\ & +7x_6^2 + 2x_7^2 - 4x_6x_7 - 10x_6 - 8x_7 \\ \text{subject to} & 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \quad \leq 100 \\ & 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \quad \leq 200 \\ & 20x_1 + x_2^2 + 6x_6^2 - 8x_7 \quad \leq 150 \\ & 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \quad \leq 0 \\ & x_i \geq 0,\ i = 1, \ldots, 7, \end{array}$$

which is formulated as an 11-variable complementarity problem. Since the objective function is convex, the mapping is monotone, but not strongly monotone on R_+^n. Table 4.7 shows that Algorithms 4.1 and 4.2 converged for both initial iterates $(0, \ldots, 0)$ and $(10, \ldots, 10)$.

Example 4.4 This example is a 15-variable traffic assignment problem from Bertsekas and Gafni [BeG82]. This problem consists of a traffic network with 25 nodes, 40 arcs, 5 O/D pairs and 10 paths. The mapping is monotone but not strongly monotone. The results are shown in Table 4.8. In this example, Algorithm 4.1 failed to find a descent direction because the mapping is not strongly monotone. But Algorithm 4.2 converged in 4 iterations for both initial iterates.

Table 4.7: Results for Example 4.3

		Algorithm 4.1			Algorithm 4.2			
CC	Initial Iterate	#Iterations	$\#f_C$	CPU	#Iterations	$\#f_C$	#Lemke	CPU
10^{-3}	$(0,\ldots,0)$	263	840	0.13	5	10	30	0.03
	$(10,\ldots.10)$	1213	4504	0.64	9	10	75	0.08
10^{-5}	$(0,\ldots,0)$	375	1119	0.19	6	11	36	0.04
	$(10,\ldots.10)$	1308	4752	0.67	10	11	81	0.08

Table 4.8: Results for Example 4.4

		Algorithm 4.1			Algorithm 4.2			
CC	Initial Iterate	#Iterations	$\#f_C$	CPU	#Iterations	$\#f_C$	#Lemke	CPU
10^{-3}	$(0,\ldots,0)$	17181	failed		4	5	62	0.09
	$(1,\ldots.1)$	16608	failed		4	5	64	0.10
10^{-5}	$(0,\ldots,0)$	17181	failed		4	5	62	0.09
	$(1,\ldots.1)$	16608	failed		4	5	64	0.10

Example 4.5 This example is the following convex programming problem:

$$\begin{array}{lll}
\text{minimize} & x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3-10)^2 & \\
& +4(x_4-5)^2 + (x_5-3)^2 + 2(x_6-1)^2 + 5x_7^2 & \\
& +7(x_8-11)^2 + 2(x_9-10)^2 + (x_{10}-7)^2 + 45 & \\
\text{subject to} & 4x_1 + 5x_2 - 3x_7 + 9x_8 & \le 105 \\
& 10x_1 - 8x_2 - 17x_7 + 2x_8 & \le 0 \\
& -8x_1 + 2x_2 + 5x_9 - 2x_{10} & \le 12 \\
& 3(x_1-2)^2 + 4(x_2-3)^2 + 2x_3^2 - 7x_4 & \le 120 \\
& 5x_1^2 + 8x_2 + (x_3-6)^2 - 2x_4 & \le 40 \\
& \frac{1}{2}(x_1-8)^2 + 2(x_2-4)^2 + 3x_5^2 - x_6 & \le 30 \\
& x_1^2 + 2(x_2-2)^2 - 2x_1x_2 + 14x_5 - 6x_6 & \le 0 \\
& -3x_1 + 6x_2 + 12(x_9-8)^2 - 7x_{10} & \le 0 \\
& x_i \ge 0,\ i = 1,\ldots,10, &
\end{array}$$

which is formulated as an 18-variable complementarity problem. The results are shown in Table 4.9. The mapping is monotone but not strongly monotone. Algorithm 4.1 converged slowly and eventually failed to find a descent direction as the iterate become very close to a solution. On the other hand, Algorithm 4.2 converged in several iterations for both initial iterates.

Table 4.9: Results for Example 4.5

		Algorithm 4.1			Algorithm 4.2			
CC	Initial Iterate	#Iterations	$\#f_C$	CPU	#Iterations	$\#f_C$	#Lemke	CPU
10^{-3}	$(0,\ldots,0)$	31616	414381	71.98	4	5	68	0.14
	$(10,\ldots.10)$	31238	411704	72.43	6	7	97	0.19
10^{-5}	$(0,\ldots,0)$	72050	failed		5	6	84	0.17
	$(10,\ldots.10)$	75257	failed		6	7	97	0.19

Example 4.6 This example is a traffic assignment problem. This is a 40-variable complementarity problem which is Example 6.2 in Aashtiani [Aas79]. The results are shown in Table 4.10. The mapping is monotone but not strongly monotone. For this example, Algorithm 4.1 converged slowly and could not attain the strict convergence criterion CC $= 10^{-5}$. On the other hand, Algorithm 4.2 failed because the linear subproblem became unsolvable after 2 or 3 iterations.

Table 4.10: Results for Example 4.6

		Algorithm 4.1			Algorithm 4.2			
CC	Initial Iterate	#Iterations	$\#f_C$	CPU	#Iterations	$\#f_C$	#Lemke	CPU
10^{-3}	$(0,\ldots,0)$	2907	24925	13.25	2	failed		
	$(10,\ldots.10)$	2981	25797	13.74	3	failed		
10^{-5}	$(0,\ldots,0)$	4218	failed		2	failed		
	$(10,\ldots.10)$	4166	failed		3	failed		

Example 4.7 This example is a spatial price equilibrium problem from Tobin [Tob88] which is formulated as a 42-variable complementarity problem. The mapping is not monotone but is close to be monotone. The results are shown in Table 4.11.

In this example, Algorithms 4.1 and 4.2 converged for all initial iterates chosen in our experiment. Note that the mapping of Example 4.7 is similar to the form (4.35) used in the experiments of Section 4.5, and hence, the mapping is sparse. For the example, Algorithm 4.1 converged mush faster than Algorithm 4.2.

Table 4.11: Results for Example 4.7

		Algorithm 4.1			Algorithm 4.2			
CC	Initial Iterate	#Iterations	$\#f_C$	CPU	#Iterations	$\#f_C$	#Lemke	CPU
10^{-3}	$(0,\ldots,0)$	63	148	0.09	6	11	130	1.34
	$(1,\ldots,1)$	66	155	0.09	7	10	131	1.37
	$(10,\ldots.10)$	63	149	0.10	7	8	157	1.60
10^{-5}	$(0,\ldots,0)$	84	199	0.12	7	12	148	1.53
	$(1,\ldots,1)$	89	209	0.12	7	10	131	1.37
	$(10,\ldots.10)$	83	197	0.12	7	8	157	1.60

4.6 Concluding remarks

When the mapping is strongly monotone with modulus μ, the solution x^* to (4.1) satisfies the inequality

$$\| x^* \| \leq \frac{1}{\mu} \| F(0) \| .$$

Hence, we may reformulate problem (4.1) as a variational inequality problem with bounded constraint by adding an extra constraint

$$\| x \|_\infty \leq \mathrm{UB},$$

where UB is a sufficiently large positive number. Then we may apply the methods of Fukushima [Fuk92] or Algorithm 3.1 directly. In this case, however, the subproblem becomes a linear variational inequality problem with a bound constraint, which is in general more difficult to solve than a linear complementarity problem of the proposed algorithms.

Since the modulus μ is generally a priori unknown, the matrix D may not satisfy $\| D \| < 4\mu$, implies that Algorithm 4.2 may fail because the search direction is not guaranteed to be a descent direction. When we do not know the exact value of μ for the strongly monotone mapping F, we may start Algorithm 4.2 with an arbitrary positive diagonal matrix D, and, if it fails, continue by halving D until convergence is obtained. Eventually we will have $\| D \| < 4\mu$ and hence Algorithm 4.2 converges by Theorem 4.2.

Chapter 5

A New Merit Function and A Successive Quadratic Programming Algorithm for Variational Inequality Problems

5.1 Introduction

In this chapter, we return to the variational inequality problem of finding a vector $x^* \in S$ such that

$$\langle F(x^*), x - x^* \rangle \geq 0 \quad \text{for all} \quad x \in S, \tag{5.1}$$

where S is a nonempty closed convex subset of R^n and F is a continuous mapping from R^n into R^n.

Recently, various merit functions for variational inequality problems have been proposed and their properties have been studied (see Section 1.2). Among them, the gap function g defined by (1.2) first introduced by Auslender [Aus76], has the property that its minima on S coincide with the solutions to the variational inequality problem.

Hence problem (5.1) can be reformulated as the optimization problem (1.3). Based on the gap function, Marcotte [Mar85] proposed a descent algorithm for monotone variational inequality problems, and Marcotte and Dussault [MaD87] presented a globally convergent modification of Newton's method.

Using the regularized gap function f_S defined by (1.5), Fukushima [Fuk92] has proposed another optimization formulation of the variational inequality problem. The function f_S is shown to be differentiable whenever so is F, while the gap function g defined by (1.2) is generally nondifferentiable. Fukushima [Fuk92] has proposed a descent method for solving the variational inequality problem using regularized gap function f_S. The function f_S has also been used in a globally convergent modification of Newton's method in Chapter 3. Independently, Auchmuty [Auc89] has proposed a class of merit functions which includes the gap function g and the regularized gap function f_S. Larsson and Patriksson [LaP94] have developed and generalized Auchmuty's class of merit functions. Wu, Florian and Marcotte [WFM93] have proposed a general descent framework for the variational inequality problem by using a class of gap functions. Unfortunately, however, all of these merit functions are not easy to evaluate unless the constraints of the problem have a relatively simple structure.

In this chapter, we propose a new merit function, which is defined by (1.5) with the set S replaced by its polyhedral outer approximation. The proposed function has the advantage over the function f_S that, even when S is a general convex set specified by nonlinear convex inequalities, we can estimate the value of the function by solving a linearly constrained quadratic programming problem. We show that the proposed

merit function has a property that its minimum on S coincides with a solution to (5.1). So the proposed function leads to another equivalent optimization problem of the variational inequality problem. We also show that the proposed function is directionally differentiable in all directions and, under suitable assumptions, any stationary point of the equivalent optimization problem actually solves the original variational inequality problem. We propose a descent method for solving the variational inequality problem and establish its convergence. We note that the method is closely related to a successive quadratic programming method for solving nonlinear programming problems.

5.2 A new merit function

In this section, we introduce a new merit function for the variational inequality problem (5.1) which is a relaxed version of the regularized gap function introduced by Fukushima [Fuk92]. In the remainder of this chapter, we suppose that the set S of (5.1) is defined by a system of inequalities of the form

$$S = \{x \in R^n \mid c_i(x) \le 0, \quad i = 1, \ldots, m\}, \tag{5.2}$$

where $c_i : R^n \to R$ are continuously differentiable convex functions, and *Slater's constraint qualification* holds; i.e., there exists an $\hat{x} \in R^n$ such that

$$c_i(\hat{x}) < 0 \quad \text{for all} \quad i = 1, \ldots, m. \tag{5.3}$$

Under these assumptions, it follows from Proposition 2.1 that x^* is a solution to (5.1) if and only if there exist Lagrange multipliers λ_i^*, $i = 1, \ldots, m$, for which the following conditions hold:

$$\begin{aligned} &F(x^*) + \sum_{i=1}^m \lambda_i^* \nabla c_i(x^*) = 0, \\ &c_i(x^*) \le 0, \ \lambda_i^* \ge 0, \ \lambda_i^* c_i(x^*) = 0, \ i = 1, \ldots, m. \end{aligned} \tag{5.4}$$

For each $x \in R^n$, we define the set $T(x)$ as

$$T(x) = \{y \in R^n \mid c_i(x) + \langle \nabla c_i(x), y - x\rangle \le 0, \quad i = 1, \ldots, m\}. \tag{5.5}$$

Note that, for all $x \in R^n$, the set $T(x)$ is a polyhedral convex set which always contains S.

Using $T(x)$, we define a function f_T by

$$f_T(x) = \max\left\{ -\langle F(x), y - x\rangle - \frac{1}{2}\langle y - x, G(y - x)\rangle \;\middle|\; y \in T(x) \right\}, \tag{5.6}$$

where G is an $n \times n$ symmetric positive definite matrix. Note that the positive definiteness of G and the convexity of $T(x)$ guarantee that the maximum in (5.6) is always unique. Thus we can rewrite (5.6) as

$$f_T(x) = -\langle F(x), H_T(x) - x\rangle - \frac{1}{2}\langle H_T(x) - x, G(H_T(x) - x)\rangle, \tag{5.7}$$

where $H_T(x)$ is the unique solution y to the quadratic programming problem

$$\begin{aligned} \mathrm{QP}(x): \quad & \text{minimize}_y \quad \tfrac{1}{2}\langle y - x, G(y - x)\rangle + \langle F(x), y - x\rangle \\ & \text{subject to} \quad y \in T(x). \end{aligned} \tag{5.8}$$

Note that it follows from Proposition 2.3 and 2.4 that $H_T(x)$ is the unique solution to the following variational inequality problem

$$\langle F(x) + G(H_T(x) - x), y - H_T(x)\rangle \ge 0 \quad \text{for all} \ y \in T(x). \tag{5.9}$$

The next Lemma characterizes a solution of problem (5.1) as a fixed point of the mapping H_T.

Lemma 5.1 *The vector x is a solution to* (5.1) *if and only if $H_T(x) = x$.*

Proof. Let x^* solve (5.1). It is known [BaS76, page 143] that if Slater's constraint qualification (5.3) holds, then x^* also satisfies the inequality

$$\langle F(x^*), y - x^*\rangle \ge 0 \quad \text{for all} \quad y \in T(x^*). \tag{5.10}$$

Since $H_T(x^*)$ solves QP(x^*), $H_T(x^*)$ satisfies

$$\langle F(x^*) + G(H_T(x^*) - x^*), y - H_T(x^*)\rangle \geq 0 \quad \text{for all } y \in T(x^*). \tag{5.11}$$

Since $H_T(x^*) \in T(x^*)$, it follows from (5.10) that

$$\langle F(x^*), H_T(x^*) - x^*\rangle \geq 0, \tag{5.12}$$

while, since $x^* \in S \subset T(x^*)$, (5.11) implies

$$\langle F(x^*) + G(H_T(x^*) - x^*), x^* - H_T(x^*)\rangle \geq 0. \tag{5.13}$$

Adding (5.12) and (5.13), we have

$$\langle G(H_T(x^*) - x^*), H_T(x^*) - x^*\rangle \leq 0.$$

Hence, we must have $x^* = H_T(x^*)$ because of the positive definiteness of G.

Conversely, suppose $x^* = H_T(x^*)$. Then we have from (5.9) that

$$\langle F(x^*), y - x^*\rangle \geq 0 \quad \text{for all} \quad y \in T(x^*),$$

which implies

$$\langle F(x^*), y - x^*\rangle \geq 0 \quad \text{for all} \quad y \in S,$$

because $S \subset T(x^*)$. Therefore, to prove that x^* solves (5.1), we only need to show that $x^* \in S$. Since $H_T(x^*) \in T(x^*)$ and $x^* = H_T(x^*)$, it follows from the definition (5.5) of $T(x)$ that $c_i(x^*) \leq 0$ for all $i = 1, \ldots, m$. This completes the proof. □

Now let us consider the optimization problem:

$$\text{minimize} \quad f_T(x) \quad \text{subject to} \quad x \in S, \tag{5.14}$$

where the function f_T is defined by (5.6). The next theorem establishes the equivalence between variational inequality problem (5.1) and optimization problem (5.14).

Theorem 5.1 *Let F of (5.1) be a continuous mapping from R^n into itself and S be a convex subset of R^n defined by (5.2). Let $f_T : R^n \to R$ be a function defined by (5.6). Then $f_T(x) \geq 0$ for all $x \in S$. Moreover, $f_T(x) = 0$ with $x \in S$ holds if and only if x solves variational inequality problem (5.1). Hence x solves (5.1) if and only if it solves optimization problem (5.14) and satisfies $f_T(x) = 0$.*

Proof. Recall that $x \in T(x)$ whenever $x \in S$. So it follows from (5.6) that

$$\begin{aligned} f_T(x) &\geq -\langle F(x), x - x\rangle - \frac{1}{2}\langle x - x, G(x - x)\rangle \\ &= 0, \end{aligned}$$

for any $x \in S$. We shall show that $f_T(x) = 0$ with $x \in S$ holds if and only if $H_T(x) = x$. This along with Lemma 5.1 proves the rest of the theorem. First suppose that $x \in S$ and $f_T(x) = 0$. Since $x \in T(x)$, (5.9) implies

$$\langle F(x) + G(H_T(x) - x), x - H_T(x)\rangle \geq 0,$$

namely

$$-\langle F(x), H_T(x) - x\rangle \geq \langle H_T(x) - x, G(H_T(x) - x)\rangle. \tag{5.15}$$

Hence, it follows from (5.7) and (5.15) that

$$f_T(x) \geq \frac{1}{2}\langle H_T(x) - x, G(H_T(x) - x)\rangle.$$

But since G is positive definite and $f_T(x) = 0$, we have $H_T(x) = x$. Next suppose that $H_T(x) = x$. Then, it follows from (5.7) that $f_T(x) = 0$. Moreover, by the feasibility of $H_T(x)$ to problem (5.8), we have

$$c_i(x) + \langle \nabla c_i(x), H_T(x) - x \rangle \leq 0,\ i = 1, \ldots, m.$$

Thus $H_T(x) = x$ implies $x \in S$. This completes the proof. □

Remark 5.1 This theorem also says that, if optimization problem (5.14) has a global minimizer which does not zero the function f_T, then problem (5.1) has no solution. On the other hand, even if problem (5.1) does not have a solution, problem (5.14) may have a global minimum x with $f_T(x) > 0$. For example, consider the case in which $F : R \rightarrow R$, $F(x) = -1$ and $S = \{x \in R \mid x \geq 0\}$. Clearly, the corresponding variational inequality problem has no solution. The function f_T associated with $G = 1$ is $f_T(x) = \dfrac{1}{2}$ for all $x \geq 0$; hence any $x \geq 0$ is a global minimizer of problem (5.14). The same remark also holds for the regularized gap function f_S (See Remark 3.1).

Next we consider the continuity and differentiability of f_T.

Definition 5.1 [Hog73b] Let Z be a point-to-set-mapping from X into 2^X, where X is a subset of R^n.

1. Z is *open* at a point $\bar{x}$ if, for any sequence $\{x^k\}$ such that $x^k \rightarrow \bar{x}$, $\bar{y} \in Z(\bar{x})$ implies that there exists a sequence $\{y^k\}$ such that $y^k \in Z(x^k)$ and $y^k \rightarrow \bar{y}$.

2. Z is *closed* at a point $\bar{x}$ if, for any sequence $\{x^k\}$ such that $x^k \rightarrow \bar{x}$, $y^k \in Z(x^k)$ and $y^k \rightarrow \bar{y}$ imply $\bar{y} \in Z(\bar{x})$.

3. Z is *continuous* at a point $\bar{x}$ if it is both open and closed at $\bar{x}$. Z is continuous on X if it is continuous at any point in X.

Proposition 5.1 *Let c_i, $i = 1, \ldots, m$, be continuously differentiable and satisfy Slater's constraint qualification* (5.3). *Then the point-to-set mapping T defined by* (5.5) *is continuous on R^n.*

Proof. Since $c_i(x) + \langle \nabla c_i(x), y - x \rangle$ is continuous with respect to (x, y) for each i, it follows from [Hog73b, Theorem 10] that T is closed at each x. By Slater's constraint qualification (5.3), there is an $\hat{x}$ such that $c_i(\hat{x}) < 0$ for all $i = 1, \ldots, m$. Moreover from the convexity of c_i, we have $c_i(x) + \langle \nabla c_i(x), \hat{x} - x \rangle \leq c_i(\hat{x}) < 0$. Hence, it follows from [Hog73b, Theorem 12] that T is open at each x. Therefore, T is continuous on R^n. □

Lemma 5.2 *Suppose that a mapping F of* (5.1) *is continuous. If the Slater's constraint qualification* (5.3) *holds, then the mapping H_T defined by* (5.9) *is bounded on any bounded set.*

Proof. Suppose that H_T is not bounded on some bounded set B. Then there is a sequence $\{x^k\}$ in B such that $\| H_T(x^k) \| \rightarrow \infty$. Since sequence $\{F(x^k)\}$ is bounded on B by the continuity of F, it follows from (5.7) and the positive definiteness of G that $f_T(x^k) \rightarrow -\infty$. On the other hand, since $\hat{x} \in T(x^k)$ for all k, we have

$$\begin{aligned} f_T(x^k) &= \max \left\{ -\langle F(x^k), y - x^k \rangle - \frac{1}{2} \langle y - x^k, G(y - x^k) \rangle \;\middle|\; y \in T(x^k) \right\} \\ &\geq -\langle F(x^k), \hat{x} - x^k \rangle - \frac{1}{2} \langle \hat{x} - x^k, G(\hat{x} - x^k) \rangle, \end{aligned}$$

which is bounded below since $\{x^k\}$ and $\{F(x^k)\}$ are both bounded. This is a contradiction. $\square$

Given $x \in R^n$ and $\lambda \in R^m$, we define matrix $M(x, \lambda)$ by

$$M(x,\lambda) = \nabla F(x) + \sum_{i=1}^{m} \lambda_i \nabla^2 c_i(x). \tag{5.16}$$

The next theorem demonstrates the directional differentiability of f_T.

Theorem 5.2 *Suppose that the mapping $F : R^n \to R^n$ of* (5.1) *is continuous and the convex functions $c_i : R^n \to R$, $i = 1, \ldots, m$, of* (5.2) *are continuously differentiable. Suppose also that Slater's constraint qualification* (5.3) *holds. Then function f_T defined by* (5.7) *is continuous on R^n. Moreover, if F is continuously differentiable and c_i, $i = 1, \ldots, m$, are twice continuously differentiable, then f_T is directionally differentiable in any direction $d \in R^n$, and its directional derivative $f'_T(x; d)$ is given by*

$$f'_T(x;d) = \min_{\lambda \in \Lambda(x)} \langle F(x) - [M(x,\lambda) - G](H_T(x) - x),\ d \rangle, \tag{5.17}$$

where $\Lambda(x)$ is defined by

$$\Lambda(x) = \left\{ \lambda \in R^m \;\middle|\; \begin{array}{l} F(x) + G(H_T(x) - x) + \sum_{i=1}^{m} \lambda_i \nabla c_i(x) = 0, \\ \lambda_i [c_i(x) + \langle \nabla c_i(x), H_T(x) - x \rangle] = 0, \\ \lambda_i \geq 0,\ i = 1, \ldots, m \end{array} \right\}. \tag{5.18}$$

Proof. To prove the first half, it is sufficient to show that mapping $H_T(x)$ is continuous. Under the given assumptions, the point-to-set mapping T is continuous by Proposition 5.1 and the objective function of (5.8) is continuous with respect to (x, y).

Therefore, by [Hog73b, Theorem 8], mapping H_T is closed. However, since H_T is everywhere defined and single-valued, H_T must be continuous for any x. The second half follows from [Hog73a, Theorem 2]. $\square$

Remark 5.2 If the set $\Lambda(x)$ is a singleton, f_T is differentiable at x and the gradient is given by

$$\nabla f_T(x) = F(x) - [M(x,\lambda) - G](H_T(x) - x).$$

A sufficient condition [FiM90, Theorem 6] for $\Lambda(x)$ to be a singleton is that the vectors $\nabla c_i(x), i \in I(x)$, are linearly independent, where $I(x) = \{i \mid c_i(x) + \langle \nabla c_i(x), H_T(x) - x \rangle = 0\}$, and the strict complementarity condition holds; i.e., $\lambda_i = 0$ implies $c_i(x) + \langle \nabla c_i(x), H_T(x) - x \rangle < 0$.

Remark 5.3 The regularized gap function f_S defined by (1.5) also has the property that the set of zeros of f_S on S coincides with the set of solutions to problem (5.1). Moreover, f_S is continuously differentiable when so is mapping F. It is easy to see that $f_T(x) \geq f_S(x)$ because $T(x)$ contains S for all x. In particular, when c_i are all linear, (1.5) coincides with (5.6).

To obtain a solution to (5.1) by solving problem (5.14), we need to find a global minimizer of f_T on S. It is therefore desirable to know conditions under which a stationary point of (5.14) is a global optimal solution, because most optimization algorithms only find a stationary point of the problem. The next theorem answers this question.

Theorem 5.3 *Let F of (5.1) be a continuously differentiable mapping from R^n into itself, c_i, $i = 1, \ldots, m$, of (5.2) be twice continuously differentiable convex functions from : R^n into R and f_T be a function defined by (5.7). Suppose that ∇F is positive definite and Slater's constraint qualification (5.3) is satisfied. If $x \in S$ and*

$$f_T'(x; y - x) \geq 0 \quad \text{for all } y \in S, \tag{5.19}$$

then x is a solution to (5.1).

Before proving the theorem, we show the next lemma, which will also be used to derive a descent condition in the next section.

Lemma 5.3 *Let mapping F of (5.1) be continuously differentiable and convex functions c_i, $i = 1, \ldots, m$, of (5.2) be twice continuously differentiable. If $d = H_T(x) - x$, then we have*

$$f_T'(x; d) \leq -\langle d, \nabla F(x) d\rangle + \min_{\lambda \in \Lambda(x)} \left(\sum_{i \in I_+} \lambda_i c_i(x) \right), \tag{5.20}$$

where $I_+ = \{i | c_i(x) > 0\}$.

Proof. Since $d = H_T(x) - x$, it follows from the Karush-Kuhn-Tucker conditions for problem (5.8) that d together with a Lagrange multiplier vector λ satisfies

$$F(x) + Gd + \sum_{i=1}^{m} \lambda_i \nabla c_i(x) = 0, \tag{5.21a}$$

$$c_i(x) + \langle \nabla c_i(x), d\rangle \leq 0, \qquad \lambda_i \geq 0, \tag{5.21b}$$

$$\lambda_i [c_i(x) + \langle \nabla c_i(x), d\rangle] = 0, \qquad i = 1, \ldots, m. \tag{5.21c}$$

From (5.16), (5.17), (5.21a) and (5.21c), we have

$$\begin{aligned} & f_T'(x; d) \\ &= \min_{\lambda \in \Lambda(x)} \left\langle F(x) - [\nabla F(x) - G]d - \sum_{i=1}^{m} \lambda_i \nabla^2 c_i(x) d,\, d \right\rangle \\ &= \min_{\lambda \in \Lambda(x)} \left\{ -\langle d, \nabla F(x) d\rangle - \left\langle d, \sum_{i=1}^{m} \lambda_i \nabla c_i(x) \right\rangle - \left\langle d, \sum_{i=1}^{m} \lambda_i \nabla^2 c_i(x) d \right\rangle \right\} \\ &= \min_{\lambda \in \Lambda(x)} \left\{ -\langle d, \nabla F(x) d\rangle + \sum_{i=1}^{m} \lambda_i c_i(x) - \sum_{i=1}^{m} \lambda_i \langle d, \nabla^2 c_i(x) d\rangle \right\}. \end{aligned} \tag{5.22}$$

Since the convexity of c_i ensures that $\nabla^2 c_i(x)$ is positive semi-definite, i.e., $\langle d, \nabla c_i^2(x) d\rangle \geq 0$, and since $\sum_{i \notin I_+} \lambda_i c_i(x) \leq 0$, we have from (5.22) that

$$f_T'(x; d) \leq -\langle d, \nabla F(x) d\rangle + \min_{\lambda \in \Lambda(x)} \left(\sum_{i \in I_+} \lambda_i c_i(x) \right).$$

This completes the proof. □

Proof of Theorem 5.3. Suppose that $x \in S$ satisfies (5.19). First note that, under Slater's constraint qualification, we have

$$T(x) \subset x + \mathrm{cl}\{\tau(y - x) \mid y \in S, \tau > 0\}, \tag{5.23}$$

where cl denotes closure of a set [BaS76, page 143]. Thus, from (5.19), (5.23) and the positive homogeneity of $f_T'(x; \cdot)$ (see Appendix A.1.3), we can deduce that

$$f_T'(x; y - x) \geq 0 \quad \text{for all } y \in T(x). \tag{5.24}$$

Since $x \in S$ implies $I_+ = \phi$, it follows from Lemma 5.3 that

$$f'_T(x; H_T(x) - x) \leq -\langle H_T(x) - x, \nabla F(x)(H_T(x) - x)\rangle. \tag{5.25}$$

Since $H_T(x) \in T(x)$, we have from (5.24) and (5.25) that

$$\langle H_T(x) - x, \nabla F(x)(H_T(x) - x)\rangle \leq 0.$$

But since $\nabla F(x)$ is positive definite, we have $x = H_T(x)$. Therefore, it follows from Lemma 5.1 that x is a solution to (5.1). □

5.3 Successive quadratic programming algorithm

In this section, we present a successive quadratic programming algorithm for solving the variational inequality problem (5.1) and prove its convergence.

First, we show that the vector

$$d = H_T(x) - x \tag{5.26}$$

is a descent direction of the penalty function $\theta_r : R^n \to R$ defined by

$$\theta_r(x) = f_T(x) + r\sum_{i=1}^{m} \max(0, c_i(x)), \tag{5.27}$$

where r is a sufficiently large positive parameter (cf. Appendix A.3.3).

Theorem 5.4 *Let F of (5.1) be a mapping from R^n into itself, c_i, $i = 1, \ldots, m$, of (5.2) be convex functions from R^n into R, H_T be a mapping defined by (5.9) and θ_r be a penalty function defined by (5.27) with penalty parameter $r > 0$. Suppose that F is continuously differentiable and c_i, $i = 1, \ldots, m$, are twice continuously differentiable. If $\nabla F(x)$ is positive definite and*

$$\| \lambda \|_\infty \leq r \quad \text{for all } \lambda \in \Lambda(x),$$

then the vector $d = H_T(x) - x$ satisfies the descent condition

$$\theta'_r(x; d) < 0,$$

whenever $d \neq 0$.

Proof. Let $I_+ = \{i \mid c_i(x) > 0\}$ and $I_0 = \{i \mid c_i(x) = 0\}$. By Theorem 5.2 and [Han77, Lemma 3.1], θ_r is directionally differentiable and the directional derivative is given by

$$\theta_r'(x;d) = f_T'(x;d) + r\sum_{i\in I_+}\langle \nabla c_i(x), d\rangle + r\sum_{i\in I_0}\max(0, \langle\nabla c_i(x), d\rangle). \tag{5.28}$$

First note that $d = H_T(x) - x$ together with a Lagrange multiplier vector λ satisfies

$$F(x) + Gd + \sum_{i=1}^{m}\lambda_i \nabla c_i(x) = 0, \tag{5.29a}$$

$$c_i(x) + \langle \nabla c_i(x), d\rangle \le 0, \qquad \lambda_i \ge 0, \tag{5.29b}$$

$$\lambda_i[c_i(x) + \langle\nabla c_i(x), d\rangle] = 0, \qquad i = 1, \ldots, m. \tag{5.29c}$$

Then (5.29b) yields

$$\sum_{i\in I_0}\max(0, \langle\nabla c_i(x), d\rangle) = 0. \tag{5.30}$$

By Lemma 5.3, we have

$$f_T'(x;d) \le -\langle d, \nabla F(x) d\rangle + \min_{\lambda\in\Lambda(x)}\left(\sum_{i\in I_+}\lambda_i c_i(x)\right). \tag{5.31}$$

Hence, it follows from (5.28), (5.29b), (5.30) and (5.31) that

$$\begin{aligned} \theta_r'(x;d) &\le -\langle d, \nabla F(x) d\rangle + \min_{\lambda\in\Lambda(x)}\left(\sum_{i\in I_+}\lambda_i c_i(x)\right) + r\sum_{i\in I_+}\langle\nabla c_i(x), d\rangle \\ &\le -\langle d, \nabla F(x) d\rangle + \min_{\lambda\in\Lambda(x)}\left(\sum_{i\in I_+}(\lambda_i - r)c_i(x)\right) \\ &< 0, \end{aligned}$$

because, by assumption, $\nabla F(x)$ is positive definite and $\| \lambda \|_\infty \le r$ for all $\lambda \in \Lambda(x)$. The proof is complete. □

Next we describe a successive quadratic programming algorithm for solving the variational inequality problem (5.1). The proposed algorithm uses the vector d defined

by (5.26) as a search direction and incorporates the exact line search to the penalty function θ_r defined by (5.27). Note that, since $H_T(x)$ is the unique solution to the quadratic programming problem (5.8), the search direction $d = H_T(x) - x$ is obtained by solving a convex quadratic programming problem.

Algorithm 5.1

Step 0 Choose $x^0 \in R^n$, $r > 0$ and a symmetric positive definite matrix G. Let $k := 0$

Step 1 Find the unique solution d^k of the quadratic programming problem:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\langle d, Gd\rangle + \langle F(x^k), d\rangle \\ \text{subject to} & c_i(x^k) + \langle\nabla c_i(x^k), d\rangle \le 0, \;\; i = 1, 2, \ldots, m. \end{array}$$

Step 2 Find an α_k such that

$$\theta_r(x^k + \alpha_k d^k) = \min_{0\le\alpha\le 1}\theta_r(x^k + \alpha d^k).$$

Step 3 Set $x^{k+1} := x^k + \alpha_k d^k$. Let $k := k+1$. Return to Step 1.

The next theorem establishes the global convergence of this algorithm.

Theorem 5.5 *Let F of (5.1) be a continuously differentiable mapping from R^n into itself, c_i, $i = 1, \ldots, m$, of (5.2) be continuously differentiable convex functions from R^n into R, H_T be a mapping defined by (5.9) and θ_r be a penalty function defined by (5.27) with penalty parameter $r > 0$. Suppose that $\nabla F(x)$ is positive definite on R^n. Suppose also that r is chosen sufficiently large. If the sequence $\{x^k\}$ generated by Algorithm*

5.1 is bounded, then $\{x^k\}$ converges to the unique solution to the variational inequality problem (5.1).

Proof. To prove the global convergence, we shall use Zangwill's global convergence Theorem A [Zan69, page 91]. We denote by $\mathcal{A}$ an algorithmic map defined by Algorithm 5.1, i.e., $x^{k+1} = \mathcal{A}(x^k)$. Since $\{x^k\}$ is bounded, it follows from [Han77, Lemma 3.3] that there exists a positive number $\bar{r}$ such that $\| \lambda^k \|_\infty \leq \bar{r}$ for all k, where λ^k is an arbitrary optimal Lagrange multiplier vector of the quadratic program solved in Step 1 at iteration k. Assuming that $r \geq \bar{r}$, we have from Theorem 5.4 that d^k satisfies the descent condition $\theta_r'(x^k; d^k) < 0$, whenever x^k is not a solution to (5.1). Moreover, since the map $d = H_T(x) - x$ and the function θ_r are continuous with respect to x, and, since the exact line search strategy on a bounded interval is closed (cf. Definition 5.1) the overall algorithmic map $\mathcal{A}$ is closed. Therefore, Zangwill's global convergence Theorem A [Zan69, page 91] guarantees that any accumulation point of $\{x^k\}$ is a solution to (5.1). Since problem (5.1) has at most one solution by the positive definiteness of $\nabla F(x)$, we conclude that the entire sequence converges to the solution to (5.1). □

Remark 5.4 When F is a gradient of some differentiable convex function φ, problem (5.1) corresponds to a necessary and sufficient optimality condition for the convex programming problem

$$\begin{array}{ll} \text{minimize} & \varphi(x) \\ \text{subject to} & c_i(x) \leq 0, \quad i = 1, \ldots, m. \end{array} \tag{5.32}$$

We may therefore apply Algorithm 5.1 to (5.32) with the identification $F = \nabla\varphi$. Then the subproblem solved in Step 1 becomes the same as that of the standard successive

quadratic programming problem [Han77]. But the merit function used in the line search is quite different from [Han77].

5.4 Computational results

In this section, we report some numerical results for Algorithm 5.1. All computer programs were coded in FORTRAN and run in double precision on a SUN SuperSPARC Station.

Throughout the computational experiments, the positive definite matrix G was chosen to be the identity matrix. The convergence criterion was

$$f_T(x^k) \le 10^{-6} \text{ and } c_i(x^k) \le 10^{-6} \quad \text{for } i = 1, \ldots, m.$$

For each example, we tested three values of the penalty parameter: $r = 1, 10$ and 100. It is noted that, though the global convergence was proved only with the exact line search, we implemented with an inexact line search rule of Armijo-type:

Step 2' Set $\alpha_k := \beta^{l_k}$, where l_k is the smallest nonnegative integer l such that

$$\theta_r(x^k) - \theta_r\left(x^k + \frac{1}{2^l}d^k\right) \ge \frac{0.0001}{2^l} \| d^k \|^2 . \tag{5.33}$$

Example 5.1 This example is a two dimensional variational inequality problem, where the mapping F is given by

$$F(x) = \begin{pmatrix} x_1 + 2x_2 + 7 \\ -2x_1 + x_2 + 5 \end{pmatrix}$$

and the set S is given by

$$S = \left\{ \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \middle| \; x_1^2 + x_2^2 \le 9 \right\}.$$

This problem has the unique solution $x^* = (-0.533144, -2.952246)^t$. The results for Example 5.1 are shown in Tables 5.1~5.3.

Tables 5.1~5.3 show that Algorithm 5.1 converges to the solution for all cases. From the tables, we see that the generated sequence converged from outside the set S. The same tables also show that Algorithm 5.1 does not necessarily decrease the value of the merit function f_T, while the value of the penalty function θ_r decreases monotonically. In Figure 5.1, we plot the iterates $\{x^k\}$ for the cases $r = 1$ and $r = 10$ in the two dimensional plane.

Table 5.1: Result for Example 5.1 ($r = 1$)

Iteration	x_1	x_2	$f_T(x)$	$c(x)$	$\theta_r(x)$
0	0.000000	0.000000	37.000000	−9.000000	37.000000
1	−1.750000	−1.250000	15.295186	−4.375000	15.295186
2	−0.913851	−3.295608	−0.342954	2.696157	2.353203
3	−0.171937	−3.296810	−0.624539	1.898517	1.273978
4	−0.284299	−3.003017	0.064855	0.098939	0.163794
5	−0.514843	−2.972955	−0.053349	0.103524	0.050175
6	−0.521524	−2.954387	0.000087	0.000389	0.000476
7	−0.533386	−2.952260	−0.000179	0.000340	0.000161
8	−0.532789	−2.952310	0.000000	0.000000	0.000000

Table 5.2: Result for Example 5.1 ($r = 10$)

Iteration	x_1	x_2	$f_T(x)$	$c(x)$	$\theta_r(x)$
0	0.000000	0.000000	37.000000	−9.000000	37.000000
1	−1.750000	−1.250000	15.295186	−4.375000	15.295186
2	−0.913851	−3.295608	−0.342954	2.696157	13.137831
3	0.569977	−3.298011	0.375274	2.201753	11.384041
4	−0.526062	−3.153634	−0.521104	1.222148	5.589636
5	−0.186360	−3.016532	0.147328	0.134194	0.818300
6	−0.501974	−2.985912	−0.085023	0.167647	0.753212
7	−0.515694	−2.955532	0.000069	0.001111	0.005625
8	−0.533415	−2.952346	−0.000464	0.000880	0.003935
9	−0.532599	−2.952345	0.000000	0.000001	0.000004

Table 5.3: Result for Example 5.1 ($r = 100$)

Iteration	x_1	x_2	$f_T(x)$	$c(x)$	$\theta_r(x)$
0	0.000000	0.000000	37.000000	−9.000000	37.000000
1	−1.750000	−1.250000	15.295186	−4.375000	15.295186
2	−1.331926	−2.272804	3.508426	−2.060336	3.508426
3	−0.672712	−2.885751	0.158826	−0.219901	0.158826
4	−0.558027	−2.931536	0.051589	−0.094702	0.051589
5	−0.546346	−2.941836	0.025286	−0.047108	0.025286
6	−0.539727	−2.947068	0.012494	−0.023483	0.012494
7	−0.533148	−2.952257	−0.000037	0.000070	0.006984
8	−0.533120	−2.952250	0.000000	0.000000	0.000000

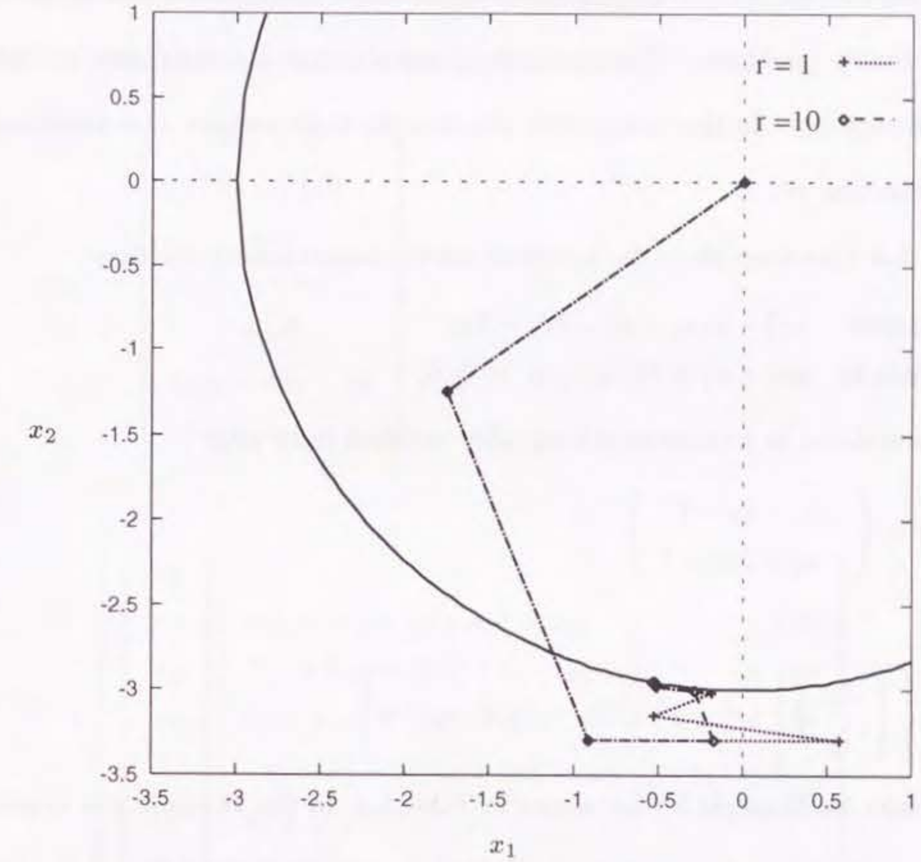

Figure 5.1: Behavior of Algorithms 5.1

Examples 5.2∼5.4 are convex programming problems which are formulated as variational inequality problems. The results for Examples 5.2∼5.4 are shown in Tables 5.4∼5.6, respectively. In the tables, $\#f_T$ denotes the total number of evaluations of the merit function f_T.

Example 5.2 This example is the following convex programming problem:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}x_1^2 - x_1x_2 + x_2^2 - 7x_1 - 7x_2 \\ \text{subject to} & 4x_1^2 + x_2^2 \le 25,\ x_1 \ge 0,\ x_2 \ge 0, \end{array}$$

which is formulated as a variational inequality problem (5.1) with

$$F(x) = \begin{pmatrix} x_1 - x_2 - 7 \\ -x_1 + 2x_2 - 7 \end{pmatrix}$$

and

$$S = \left\{ \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \,\middle|\, 4x_1^2 + x_2^2 \le 25,\ x_1 \ge 0,\ x_2 \ge 0 \right\}.$$

The results for Example 5.2 are shown in Table 5.4. In this example, the objective function is quadratic convex and hence $\nabla F(x)$ is positive definite for all x. Table 5.4 shows that Algorithm 5.1 converged for all cases, but when $r = 1$, the number of iterations is extremely large.

Table 5.4: Results for Example 5.2

Initial Iterate	r	#Iterations	$\#f_T$
	1	149	298
$(0,\dots,0)$	10	12	27
	100	14	35

Example 5.3 This example is the convex programming problem given in Example 4.3 which is formulated as a variational inequality problem (5.1) where

$$F(x) = \begin{pmatrix} 2x_1 - 20 \\ 10x_2 - 120 \\ 4x_3^3 \\ 6x_4 - 66 \\ 40x_5^3 \\ 14x_6 - 4x_7 - 10 \\ 4x_7 - 4x_6 - 8 \end{pmatrix}$$

and

$$S = \left\{ \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} \,\middle|\, \begin{array}{ll} 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 & \le 100 \\ 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 & \le 200 \\ 20x_1 + x_2^2 + 6x_6^2 - 8x_7 & \le 150 \\ 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 & \le 0 \\ x_i \ge 0,\ i = 1,\dots,7, & \end{array} \right\}.$$

The results for Example 5.3 are shown in Table 5.5. Since the mapping F is monotone but not strongly monotone on R^7, $\nabla F(x)$ is not necessarily positive definite

Table 5.5: Results for Example 5.3

Initial Iterate	r	#Iterations	$\#f_T$
	1	failed	
$(0,\dots,0)$	10	378	1988
	100	302	1627

for all x. Table 5.5 shows that Algorithm 5.1 converged when the penalty parameter was $r = 10$ and 100. But when $r = 1$, Algorithm 5.1 stalled because the search direction d^k failed to be a descent direction for the penalty function θ_r at the 37th iteration.

Example 5.4 This example is the convex programming problem given in Example 4.5 which is formulated as a variational inequality problem (5.1) with

$$F(x) = \begin{pmatrix} 2x_1 + x_2 - 14 \\ x_1 + 2x_2 - 16 \\ 2x_3 - 10 \\ 8x_4 - 40 \\ 2x_5 - 6 \\ 4x_6 - 4 \\ 10x_7 \\ 14x_8 - 154 \\ 4x_9 - 40 \\ 2x_{10} - 14 \end{pmatrix}$$

and

Table 5.6: Results for Example 5.4

Initial Iterate	r	#Iterations	$\#f_T$
	1	142	423
$(0, \ldots, 0)$	10	147	435
	100	162	482

$$S = \left\{ \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{pmatrix} \middle| \begin{array}{ll} 4x_1 + 5x_2 - 3x_7 + 9x_8 & \leq 105 \\ 10x_1 - 8x_2 - 17x_7 + 2x_8 & \leq 0 \\ -8x_1 + 2x_2 + 5x_9 - 2x_{10} & \leq 12 \\ 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 & \leq 120 \\ 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 & \leq 40 \\ \frac{1}{2}(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 & \leq 30 \\ x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 & \leq 0 \\ -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} & \leq 0 \\ x_i \geq 0,\ i = 1, \ldots, 10, & \end{array} \right\}.$$

The results for Example 5.4 are shown in Table 5.6. Since the objective function is quadratic convex, $\nabla F(x)$ is positive definite for all x. Table 5.6 shows that Algorithm 5.1 converged for all cases. In this example, the number of iterations is almost constant.

Chapter 6

A Globally Convergent Newton Method for Solving Variational Inequality Problems with Inequality Constraints

6.1 Introduction

We consider the variational inequality problem of finding $x^* \in S$ such that

$$\langle F(x^*), x - x^* \rangle \geq 0 \quad \text{for all} \quad x \in S, \tag{6.1}$$

where S is a nonempty closed convex subset of R^n and F is a continuously differentiable mapping from R^n into R^n. In this chapter, we suppose that the set S is specified by

$$S = \{x \in R^n \mid c_i(x) \leq 0, \quad i = 1, \dots, m\}, \tag{6.2}$$

where $c_i : R^n \to R$ are twice continuously differentiable convex functions. Throughout this chapter, we assume that *Slater's constraint qualification* holds; i.e., there exists an $\hat{x} \in R^n$ such that

$$c_i(\hat{x}) < 0 \quad \text{for all} \quad i = 1, \ldots, m. \tag{6.3}$$

Many iterative methods, such as Newton's method, projection methods, the linearized Jacobi method and the successive over-relaxation methods, have been proposed to solve the variational inequality problem (6.1). Among them, Newton's method generates a sequence $\{x^k\}$, where x^{k+1} is a solution to the linearized variational inequality problem

$$\left\langle F(x^k) + \nabla F(x^k)^t(x^{k+1} - x^k), x - x^{k+1} \right\rangle \geq 0 \quad \text{for all} \quad x \in S. \tag{6.4}$$

It can be shown that, under suitable assumptions, Newton's method converges quadratically to a solution x^*, provided that an initial point x^0 is chosen sufficiently close to x^* (see Theorem 2.1).

For the variational inequality problem (6.1), various merit functions have been proposed and has been used to globalize Newton's method (6.4). Marcotte and Dussault [MaD87] obtained a globally convergent Newton method by incorporating an exact line search strategy for the gap function g defined by (1.2). Another modification is Algorithm 3.1 proposed in Chapter 3, which makes use of Armijo line search for the regularized gap function f_S defined by (1.5).

Note that the above methods tacitly assume that the constraint set S has a relatively simple structure. For example, when S is a polyhedral convex set, that is, functions c_i are all affine, the variational inequality subproblem (6.4) of Newton's method becomes an affine variational inequality problem and the functions g and f_S can be evaluated by solving linear and quadratic programming problems, respectively.

However, when S is a general convex set defined by nonlinear convex functions, solving the linearized subproblem (6.4) and evaluating $g(x)$ and $f_S(x)$ should be considered difficult tasks.

In this chapter we propose a new globally convergent Newton method for solving variational inequality problems with general inequality constraints. The method solves at each iteration an affine variational inequality subproblem, in which not only the mapping F but also the constraint functions c_i are linearized. Moreover it makes use of the merit function f_T introduced in the previous chapter to obtain global convergence. The proposed method has a clear advantage over the method of Marcotte and Dussault [MaD87] and Algorithm 3.1 that solve subproblems (6.4) and use the gap function and the regularized gap function respectively, in that each step of the algorithm is a finite computation even if the set S is specified by nonlinear inequalities. It is shown that, when the mapping F of (6.1) is strongly monotone, the method converges globally to the solution, and that, under some additional assumptions, the rate of convergence is superlinear. The proposed method is closely related to a successive quadratic programming method for solving nonlinear programming problems.

6.2 Globally convergent Newton method

In this section, we present a globally convergent Newton method for the variational inequality problem (6.1), which incorporates an Armijo line search procedure for the penalty function $\theta_r : R^n \to R$ defined by

$$\theta_r(x) = f_T(x) + r\sum_{i=1}^{m} \max(0, c_i(x)), \tag{6.5}$$

where f_T is defined by (5.6) and r is a sufficiently large positive parameter. By Theorem 5.2 and [Han77, Lemma 3.1], θ_r is directionally differentiable and the directional derivative is given by

$$\theta_r'(x; d) = f_T'(x; d) + r\sum_{i\in I_+} \langle \nabla c_i(x), d\rangle + r\sum_{i\in I_0} \max(0, \langle \nabla c_i(x), d\rangle), \tag{6.6}$$

where $I_+ = \{i \,|\, c_i(x) > 0\}$ and $I_0 = \{i \,|\, c_i(x) = 0\}$. Throughout this section, we assume that the mapping F is continuously differentiable and strongly monotone with modulus μ (cf. (2.16)), so that ∇F satisfies

$$\langle d, \nabla F(x)d\rangle \geq \mu \, \|d\|^2 \quad \text{for all } \; x, d \in R^n \tag{6.7}$$

(see Proposition 2.6). Note that, since the convexity of c_i guarantees that $\nabla^2 c_i(x)$ is positive semi-definite, (6.7) implies that the matrix $M(x, \lambda)$ defined by (5.16) satisfies

$$\langle d, M(x, \lambda)d\rangle \geq \mu \, \|d\|^2 \quad \text{for all } x, d \in R^n, \tag{6.8}$$

whenever $\lambda \geq 0$.

Now we state the algorithm. The proposed algorithm uses a search direction obtained by solving a linearized variational inequality problem and determines the next iterate by performing the Armijo line search for the penalty function θ_r.

Algorithm 6.1

Step 0 Choose $x^0 \in R^n$, $r > 0$, $0 < \beta < 1$, $0 < \sigma < 1$, and a symmetric positive definite matrix G. Let $k := 0$

Step 1 Find the unique solution $\bar{x}^k \in T(x^k)$ of the linearized variational inequality problem

$$\left\langle F(x^k) + M(x^k, \bar{\lambda}^k)^t(\bar{x}^k - x^k), x - \bar{x}^k \right\rangle \geq 0 \;\; \text{for all } \; x \in T(x^k), \tag{6.9}$$

where $\bar{\lambda}^k$ is an arbitrary vector in $\Lambda(x^k)$. Let $d^k := \bar{x}^k - x^k$.

Step 2 Set $\alpha_k := \beta^{l_k}$, where l_k is the smallest nonnegative integer l such that

$$\theta_r(x^k) - \theta_r(x^k + \beta^l d^k) \geq -\sigma\beta^l\theta_r'(x^k; d^k). \tag{6.10}$$

Step 3 Set $x^{k+1} := x^k + \alpha_k d^k$. Let $k := k + 1$. Return to Step 1.

Note that in Step 1 we need an optimal Lagrange multiplier vector $\bar{\lambda}^k$ for the quadratic programming problem QP(x^k) (cf. (5.8)). However, this has already been obtained in the previous iteration as a by-product of evaluating the function value $f_T(x^k)$. Note also that, by the positive definiteness of M, the linearized problem (6.9) always has a unique solution. Moreover problem (6.9) can be rewritten as a linear complementary problem, which can be solved in a finite number of steps using Lemke's complementarity pivoting algorithm [Lem65]. The following theorem shows that the vector d^k generated by Algorithm 6.1 is a descent direction of θ_r at x^k.

Theorem 6.1 *Suppose that the mapping $F : R^n \to R^n$ of (6.1) is continuously differentiable and strongly monotone on R^n with modulus μ, that the convex functions $c_i : R^n \to R$, $i = 1, \ldots, m$, of (6.2) are twice continuously differentiable and that Slater's constraint qualification (6.3) holds. Let θ_r be a penalty function defined by (6.5) where $r > 0$ is a penalty parameter. If*

$$\|\lambda\|_\infty \leq r \quad \text{for all } \lambda \in \Lambda(x^k),$$

where $\Lambda(x^k)$ is defined by (5.18), then the vector $d^k = \bar{x}^k - x^k$ obtained by Step 1 of Algorithm 6.1 satisfies the inequality

$$\theta_r'(x^k; d^k) \leq -\left(\mu - \frac{1}{2}\|G\|\right)\|d^k\|^2, \tag{6.11}$$

where G is an $n \times n$ positive definite matrix of (5.7). In particular, if G is chosen sufficiently small to satisfy $\|G\| < 2\mu$, then d^k is a descent direction of θ_r at x^k.

Before proving the theorem, we give the following lemma.

Lemma 6.1 *For any $x \in R^n$, we have*

$$\begin{aligned} f_T(x) &= \frac{1}{2}\langle H_T(x) - x, G(H_T(x) - x)\rangle - \sum_{i=1}^m \lambda_i c_i(x) \\ &\geq -\sum_{i=1}^m \lambda_i c_i(x) \end{aligned}$$

for any $\lambda \in \Lambda(x)$, where $H_T(x)$ is the unique solution to a quadratic programming problem (5.8). In particular, if $x \in S$, then

$$f_T(x) \geq \frac{1}{2}\langle H_T(x) - x, G(H_T(x) - x)\rangle.$$

Proof. Since $H_T(x)$ solves (5.8), it follows from the definition (5.18) of $\Lambda(x)$ that each vector $\lambda \in \Lambda(x)$ satisfies

$$\begin{aligned} & F(x) + G(H_T(x) - x) + \sum_{i=1}^m \lambda_i \nabla c_i(x) = 0, \\ & c_i(x) + \langle \nabla c_i(x), H_T(x) - x\rangle \leq 0, \qquad \lambda_i \geq 0, \\ & \lambda_i[c_i(x) + \langle \nabla c_i(x), H_T(x) - x\rangle] = 0, \qquad i = 1, \ldots, m. \end{aligned}$$

Hence, we have from (5.7) that

$$\begin{aligned} f_T(x) &= -\langle F(x), H_T(x) - x\rangle - \frac{1}{2}\langle H_T(x) - x, G(H_T(x) - x)\rangle \\ &= \langle G(H_T(x) - x), H_T(x) - x\rangle + \left\langle \sum_{i=1}^m \lambda_i \nabla c_i(x), H_T(x) - x \right\rangle \\ &\quad -\frac{1}{2}\langle H_T(x) - x, G(H_T(x) - x)\rangle \\ &= \frac{1}{2}\langle H_T(x) - x, G(H_T(x) - x)\rangle + \left\langle \sum_{i=1}^m \lambda_i \nabla c_i(x), H_T(x) - x \right\rangle \\ &= \frac{1}{2}\langle H_T(x) - x, G(H_T(x) - x)\rangle - \sum_{i=1}^m \lambda_i c_i(x) \\ &\geq -\sum_{i=1}^m \lambda_i c_i(x), \end{aligned}$$

where the last inequality follows from the positive definiteness of G. Since $\lambda_i \geq 0$ and $c_i(x) \leq 0$, $i = 1, \ldots, m$, for all $x \in S$, the last part of the lemma follows immediately. $\square$

Proof of Theorem 6.1. For simplicity of notation, we omit the superscript k in x^k, $\bar{x}^k$ and d^k. Let $I_+ = \{i \mid c_i(x) > 0\}$ and $I_0 = \{i \mid c_i(x) = 0\}$. Note that $d = \bar{x} - x$

together with some Lagrange multiplier vector $\hat{\lambda} \geq 0$ satisfies

$$F(x) + M(x, \bar{\lambda})^t d + \sum_{i=1}^{m} \hat{\lambda}_i \nabla c_i(x) = 0, \tag{6.12a}$$

$$c_i(x) + \langle \nabla c_i(x), d \rangle \leq 0, \tag{6.12b}$$

$$\hat{\lambda}_i [c_i(x) + \langle \nabla c_i(x), d \rangle] = 0, \qquad i = 1, \ldots, m. \tag{6.12c}$$

Then (6.12b) yields

$$\sum_{i \in I_0} \max(0, \langle \nabla c_i(x), d \rangle) = 0. \tag{6.13}$$

Since $d = \bar{x} - x$, and since $M(x, \lambda) = \nabla F(x) + \sum_{i=1}^{m} \lambda_i \nabla^2 c_i(x)$, it follows from (5.17) that

$$\begin{aligned}
f_T'(x; d) &= \min_{\lambda \in \Lambda(x)} \langle F(x) - [M(x, \lambda) - G](H_T(x) - x), \bar{x} - x \rangle \\
&\leq \langle F(x) - [M(x, \bar{\lambda}) - G](H_T(x) - x), \bar{x} - x \rangle \\
&= \langle F(x), \bar{x} - x \rangle - \Big\langle H_T(x) - x, M(x, \bar{\lambda})^t (\bar{x} - x) \Big\rangle \\
&\quad + \langle G(H_T(x) - x), \bar{x} - x \rangle \\
&= \Big\langle F(x) + M(x, \bar{\lambda})^t (\bar{x} - x), \bar{x} - x \Big\rangle - \Big\langle M(x, \bar{\lambda})^t (\bar{x} - x), \bar{x} - x \Big\rangle \\
&\quad - \Big\langle F(x) + M(x, \bar{\lambda})^t (\bar{x} - x), H_T(x) - x \Big\rangle \\
&\quad + \langle F(x), H_T(x) - x \rangle + \langle G(H_T(x) - x), \bar{x} - x \rangle \\
&= - \Big\langle F(x) + M(x, \bar{\lambda})^t (\bar{x} - x), H_T(x) - \bar{x} \Big\rangle \\
&\quad + \Big\{ \langle F(x), H_T(x) - x \rangle + \frac{1}{2} \langle H_T(x) - x, G(H_T(x) - x) \rangle \Big\} \\
&\quad - \langle d, M(x, \bar{\lambda}) d \rangle + \frac{1}{2} \langle d, Gd \rangle - \frac{1}{2} \langle \bar{x} - H_T(x), G(\bar{x} - H_T(x)) \rangle,
\end{aligned} \tag{6.14}$$

where the last equality follows from the equality

$$\begin{aligned}
2\langle \bar{x} - x, G(H_T(x) - x) \rangle &= \langle H_T(x) - x, G(H_T(x) - x) \rangle + \langle \bar{x} - x, G(\bar{x} - x) \rangle \\
&\quad - \langle \bar{x} - H_T(x), G(\bar{x} - H_T(x)) \rangle.
\end{aligned}$$

Since $\bar{x}$ is a solution to (6.9), the first term of (6.14) is nonpositive. From (5.7), the second term of (6.14) equals $-f_T(x)$. The last term is nonpositive by the positive definiteness of G. Hence, we have

$$f_T'(x; d) \leq -f_T(x) - \langle d, M(x, \bar{\lambda}) d \rangle + \frac{1}{2} \langle d, Gd \rangle.$$

Moreover, since $\bar{\lambda} \in \Lambda(x)$, it follows from Lemma 6.1 that

$$f_T'(x; d) \leq -\langle d, M(x, \bar{\lambda}) d \rangle + \frac{1}{2} \langle d, Gd \rangle + \sum_{i=1}^{m} \bar{\lambda}_i c_i(x). \tag{6.15}$$

Hence, we have

$$\begin{aligned}
\theta_r'(x; d) &\leq -\langle d, M(x, \bar{\lambda})^t d \rangle + \frac{1}{2} \langle d, Gd \rangle + \sum_{i=1}^{m} \bar{\lambda}_i c_i(x) + r \sum_{i \in I_+} \langle \nabla c_i(x), d \rangle \\
&\leq -\langle d, M(x, \bar{\lambda})^t d \rangle + \frac{1}{2} \langle d, Gd \rangle + \sum_{i \in I_+} (\bar{\lambda}_i - r) c_i(x) \\
&\leq -\left(\mu - \frac{1}{2} \| G \| \right) \| d^k \|^2,
\end{aligned}$$

where the first inequality follows from (6.6), (6.13) and (6.15), the second inequality follows from (6.12b) together with the fact that $\bar{\lambda}_i \geq 0$ for all i and $c_i(x) \leq 0$ for $i \notin I_+$, and the third inequality follows from (6.8) and $\| \lambda \|_\infty \leq r$ for all $\lambda \in \Lambda(x)$. This proves (6.11). The last part of the theorem follows immediately. □

Next we show the global convergence of Algorithm 6.1.

Theorem 6.2 *Suppose that the mapping $F : R^n \to R^n$ of (6.1) is continuously differentiable and strongly monotone on R^n with modulus μ, that convex functions $c_i : R^n \to R$, $i = 1, \ldots, m$, of (6.2) are twice continuously differentiable and that Slater's constraint qualification (6.3) holds. Suppose also that the parameter r of Algorithm 6.1 is chosen sufficiently large and that the matrix G of Algorithm 6.1 is chosen to satisfy $\| G \| < 2\mu$. If the sequence $\{x^k\}$ generated by Algorithm 6.1 is bounded, then $\{x^k\}$ converges to the unique solution to the variational inequality problem (6.1).*

Proof. Since the sequence $\{x^k\}$ is bounded, it follows from [Han77, Lemma 3.3] that there exists a positive number $\bar{r} > 0$ such that $\| \lambda^k \|_\infty \le \bar{r}$ for all k, where λ^k is any vector in $\Lambda(x^k)$. Assuming that $r \geq \bar{r}$, we have from Theorem 6.1 that d^k satisfies the descent condition (6.11), whenever x^k is not a solution to (6.1). Hence, by the line search rule (6.10), the sequence $\{\theta_r(x^k)\}$ is decreasing. This together with the boundedness of $\{x^k\}$ implies that there is at least an accumulation point. In a way similar to the proof of Theorem 3.2, it can be shown that any accumulation point is a solution to (6.1). Moreover, under the strong monotonicity assumption, problem (6.1) must have a unique solution. Therefore we conclude that the entire sequence $\{x^k\}$ converges to the unique solution to (6.1). □

Next we examine the asymptotic rate of convergence of Algorithm 6.1. To this end, we consider the iterates (x^k, λ^k) generated by Newton's method directly applied to the mixed nonlinear complementarity problem (5.4), namely

$$\begin{aligned} &F(x^k) + M(x^k, \lambda^k)^t(x^{k+1} - x^k) + \sum_{i=1}^{m} \lambda_i^{k+1} \nabla c_i(x^k) = 0, \\ &c_i(x^k) + \left\langle \nabla c_i(x^k), x^{k+1} - x^k \right\rangle \le 0, \qquad \lambda_i^{k+1} \ge 0, \\ &\lambda_i^{k+1} \left[c_i(x^k) + \left\langle \nabla c_i(x^k), x^{k+1} - x^k \right\rangle \right] = 0, \quad i = 1, \ldots, m. \end{aligned} \tag{6.16}$$

It can be shown [GaM76] that, if $\nabla F(x^*)$ is positive definite, the strict complementarity holds at x^*, i.e., $c_i(x^*) = 0$ implies $\lambda^* > 0$, and if the linear independence of the active constraints hold (cf. Appendix A.3.2), then the sequence generated by Newton's method (6.16) is quadratically convergent, provided that the starting point is chosen sufficiently close to the solution. (Note that [GaM76] deals with the nonlinear programming problem, which corresponds to a special case of problem (6.1) where F is a gradient mapping of some scalar function, so that F is symmetric. But the symmetry assumption is not used in the proof of the theorem in [GaM76].)

Note that a solution x^{k+1} to (6.16) is a solution of the variational inequality problem

$$\left\langle F(x^k) + M(x^k, \lambda^k)^t(x^{k+1} - x^k), x - x^{k+1} \right\rangle \ge 0 \quad \text{for all} \quad x \in T(x^k), \tag{6.17}$$

which is the same problem as (6.9) solved in Step 1 of Algorithm 6.1, except for the choice of λ^k. Therefore, if $\| M(x^k, \bar{\lambda}^k) - M(x^k, \lambda^k) \|$ tends to zero as $x^k \to x^*$, then the sequence $\{x^k\}$ generated by solving the linearized variational inequality problem

$$\left\langle F(x^k) + M(x^k, \bar{\lambda}^k)^t(x^{k+1} - x^k), x - x^{k+1} \right\rangle \ge 0 \quad \text{for all} \quad x \in T(x^k)$$

with an arbitrary $\bar{\lambda}^k \in \Lambda(x^k)$, is locally superlinearly convergent.

Since the vector $\bar{\lambda}^k$ belongs to $\Lambda(x^k)$ defined by (5.18), and λ^k in (6.17) is determined in the previous Newton iteration (6.16), both $\bar{\lambda}^k$ and λ^k approach the set $\Lambda(x^*)$

whenever x^k converges to x^*. In particular, if $\Lambda(x^*)$ consists of the unique vector λ^*, then both $\bar{\lambda}^k$ and λ^k converge to λ^*, and hence we have $\| M(x^k, \bar{\lambda}^k) - M(x^k, \lambda^k) \| \to 0$. Note that the uniqueness of the Lagrange multiplier vector λ^* is ensured by the linear independence of the active constraints.

These observations are summarized in the following theorem.

Theorem 6.3 *Let the assumptions of Theorem 6.2 be satisfied. In addition, suppose that the strict complementarity and the linear independence of the active constraints hold at the solution x^*. If there is an integer $\bar{k} > 0$ such that the unit step size is accepted in Step 2 of Algorithm 6.1 for all $k \geq \bar{k}$, then the sequence $\{x^k\}$ generated by Algorithm 6.1 converges superlinearly to the solution x^*.*

6.3 Computational results

In this section, we report some numerical results for Algorithm 6.1. All computer programs were coded in FORTRAN and run in double precision on a SUN SuperSPARC Station.

Throughout the computational experiments, the positive definite matrix G was chosen to be the identity matrix multiplied by 0.1. The convergence criterion was

$$f_T(x^k) \leq 10^{-6} \text{ and } c_i(x^k) \leq 10^{-6} \quad \text{for } i = 1, \dots, m.$$

In solving the linearized subproblem (6.9), we first transformed it into a linear complementarity problem, and then applied Lemke's complementarity pivoting method [Lem65] coded by Fukushima [IbF91].

For each example, we tested three values of the penalty parameter: $r = 1, 10$ and 100. It is noted that, though the convergence of Algorithm 6.1 was proved only with the Armijo line search rule (6.10), we implemented with the simpler line search rule (5.33).

All examples in this chapter are convex programming problems which are formulated as variational inequality problems. The results are shown in Tables 6.1~6.3. In the tables, $\#f_T$ denotes the total number of evaluations of the merit function f_T.

Example 6.1 This example is the two dimensional convex programming problem given in Example 5.2. The results for Example 6.1 are shown in Table 6.1. In this example, the objective function is quadratic convex and hence F is strongly monotone on R^2.

Table 6.1 shows that Algorithm 6.1 converged for all cases.

Table 6.1: Results for Example 6.1

Initial Iterate	r	#Iterations	$\#f_T$
$(0,\ldots,0)$	1	7	9
	10	5	9
	100	9	19

Example 6.2 This example is the 7-dimensional convex programming problem given in Example 4.3 which is formulated as a variational inequality problem (cf. Example 5.3). The results for Example 6.2 are shown in Table 6.2. It is noted that the mapping F is monotone but not strongly monotone on R^7. Table 6.2 shows that Algorithm 6.1 converged when the penalty parameter was $r = 10$ and 100. But when $r = 1$, Algorithm 6.1 stalled because the search direction d^k failed to be a descent direction of the penalty function θ_r at 9th iteration.

Table 6.2: Results for Example 6.2

Initial Iterate	r	#Iterations	$\#f_T$
$(0,\ldots,0)$	1	failed	
	10	11	21
	100	12	25

Example 6.3 This example is the 10-dimensional convex programming problem given in Example 4.3 which is formulated as a variational inequality problem (cf. Example

Table 6.3: Results for Example 6.3

Initial Iterate	r	#Iterations	$\#f_T$
$(0,\ldots,0)$	1	5	6
	10	5	6
	100	5	6

5.4). The results for Example 6.3 are shown in Table 6.3. Table 6.3 shows that Algorithm 6.1 converged for all cases.

Figures 6.1~6.3 illustrate how Algorithm 6.1 converged for Examples 6.1~6.3, respectively. In the figures, the vertical axis represents the distance from a generated iterate to the solution, i.e.,

$$\mathrm{DIST} = \| x^k - x^* \| .$$

From Figures 6.1~6.3, it is observed that, for all test problems in this section, the rate of convergence is superlinear when Algorithm 6.1 converges to the solution.

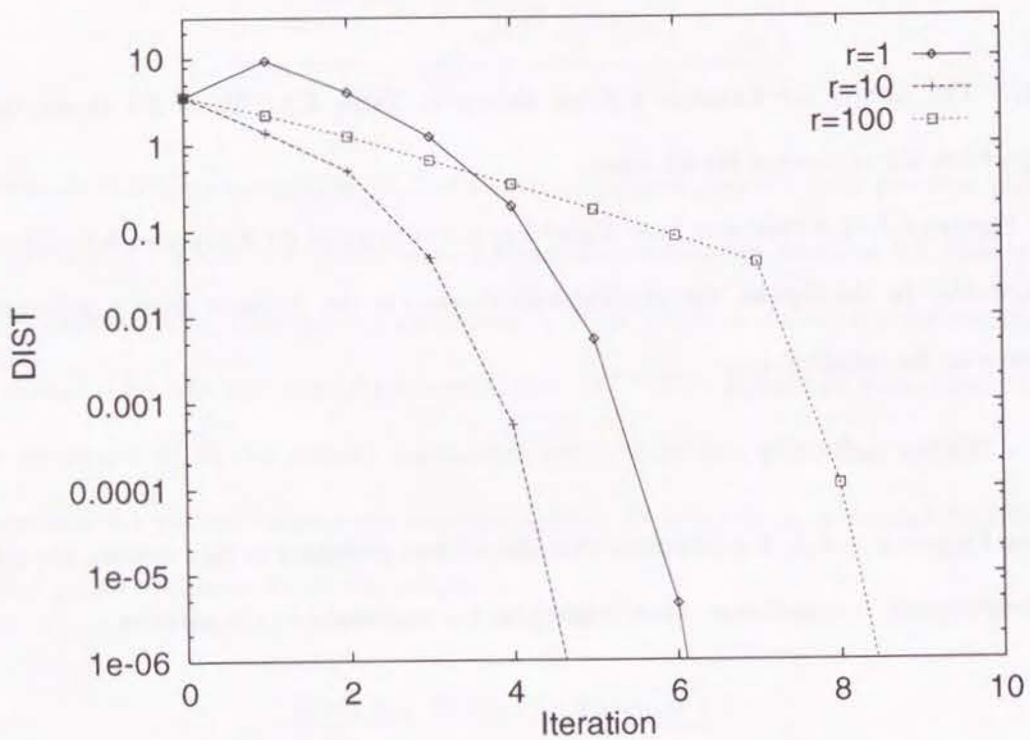

Figure 6.1: Results for Example 6.1

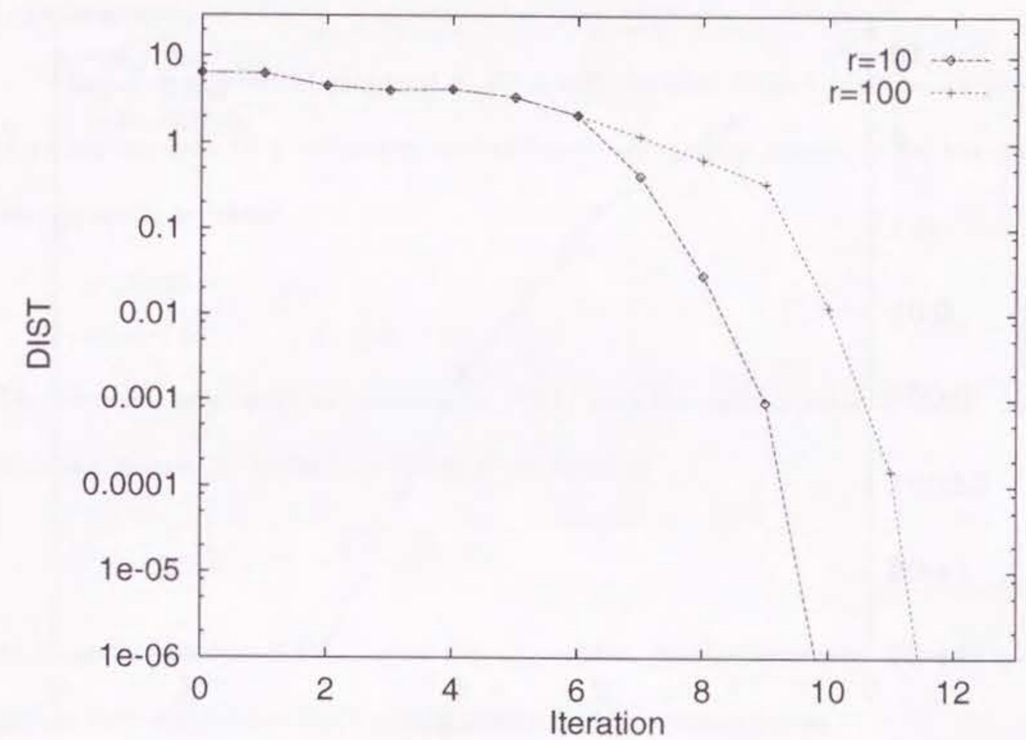

Figure 6.2: Results for Example 6.2

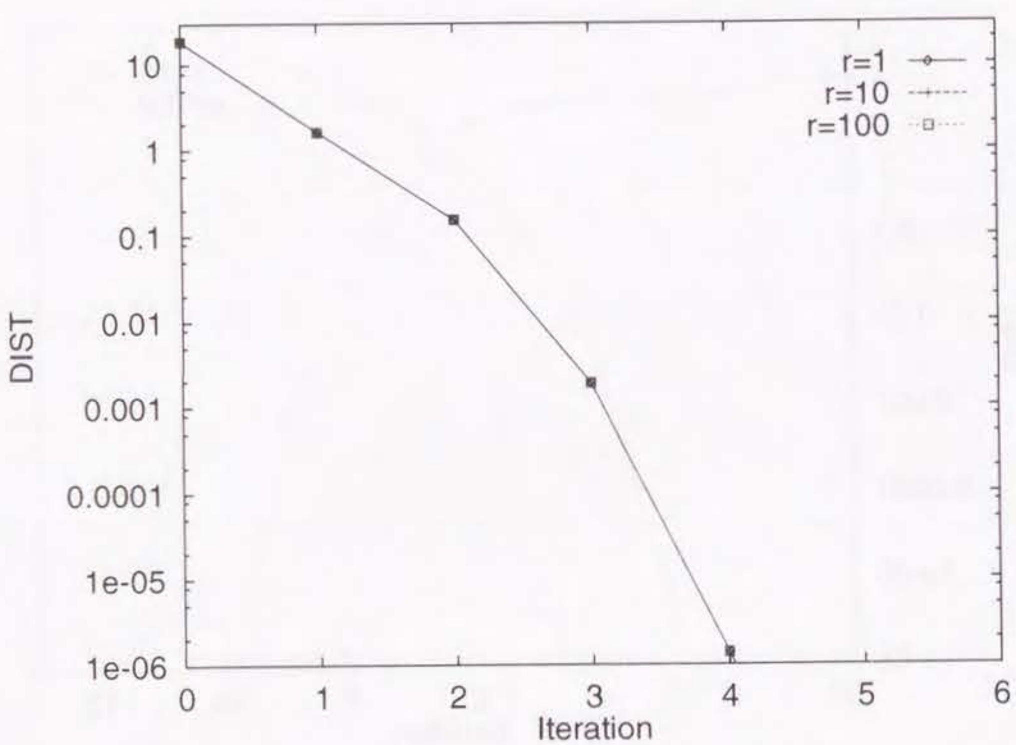

Figure 6.3: Results for Example 6.3

6.4 Concluding remarks

We have proposed a Newton's method for solving the variational inequality problem, and shown that, under the strong monotonicity assumption, the method is globally convergent and that, under some additional assumptions, the rate of convergence is superlinear.

When F is a gradient mapping of some differentiable convex function φ, problem (6.1) corresponds to a necessary and sufficient optimality condition for the convex programming problem

$$\begin{array}{ll} \text{minimize} & \varphi(x) \\ \text{subject to} & c_i(x) \leq 0, \ \ i = 1, \dots, m. \end{array} \tag{6.18}$$

Therefore we may apply our method to (6.18) with the identification $F = \nabla\varphi$. In this case, the matrix M defined by (5.16) is rewritten as

$$M(x, \lambda) = \nabla^2\varphi(x) + \sum_{i=1}^{m} \lambda_i \nabla^2 c_i(x),$$

which is the Hessian of the Lagrangian of problem (6.18). Moreover, since M is symmetric, the subproblem (6.9) solved in Step 1 can be rewritten as

$$\begin{array}{ll} \text{minimize}_d & \frac{1}{2}\left\langle d, M(x^k, \bar{\lambda}^k)d\right\rangle + \left\langle F(x^k), d\right\rangle \\ \text{subject to} & c_i(x^k) + \langle \nabla c_i(x^k), d\rangle \leq 0 \ \ i = 1, \dots, m. \end{array}$$

Thus Algorithm 6.1 reduces to a successive quadratic programming (SQP) method. A major difference from other SQP methods is that Algorithm 6.1 makes use of the function f_T as a merit function to globalize the convergence, instead of using a penalty function associated with problem (6.18).

In the last two chapters, we have assumed that the set S is specified by a system of inequalities (cf. (6.2)). In general, the convex set S may be defined by a system of inequalities and equalities of the form

$$S = \{x \in R^n \mid c_i(x) \leq 0, \;\; i = 1, \ldots, m, \;\; h_j(x) = 0, \;\; j = 1, \ldots, l\}, \tag{6.19}$$

where $h_j : R^n \to R$ are affine functions. In this case, by replacing T with

$$T(x) = \left\{ y \in R^n \;\middle|\; \begin{array}{ll} c_i(x) + \langle \nabla c_i(x), y - x \rangle \leq 0, & i = 1, \ldots, m, \\ h_j(y) = 0, & j = 1, \ldots, l \end{array} \right\}, \tag{6.20}$$

we can define the merit function f_T. Under Slater's constraint qualification for (6.19), i.e., there exists an $\hat{x} \in R^n$ such that

$$c_i(\hat{x}) < 0 \text{ for } i = 1, \ldots, m \text{ and } h_j(\hat{x}) = 0, \text{ for } j = 1, \ldots, l,$$

it is not difficult to show that Theorems 5.1 and 5.2 hold for f_T with T defined by (6.20). We can also apply Algorithms 5.1 and 6.1 to the variational inequality problem with S defined by (6.19) and establish their global convergence.

Chapter 7

Conclusion

In this thesis, we have developed efficient algorithms for solving the variational inequality problem based on its optimization reformulations.

In Chapter 3, for the variational inequality problem with general convex constraints, we proposed a globally convergent modification of Newton's method by incorporating a line search strategy to minimize the regularized gap function. In Chapter 4, this method was specialized to solve the nonlinear complementarity problem. In the same chapter, we also proposed a descent method for solving the nonlinear complementarity problem and proved its global convergence. Through some computational experiments, these algorithms were shown to be practically efficient.

In Chapters 5 and 6, we considered the variational inequality problem in which the feasible set was specified by nonlinear convex inequalities. We proposed a new merit function which was a modification of the regularized gap function and had a property that the value of the function could be evaluated by solving a convex quadratic programming problem. Based on the new merit function, we proposed a successive quadratic programming algorithm for solving variational inequality problem and proved its global convergence. The proposed merit function was also used to construct another globally convergent modification of Newton's method. In this method, not only the mapping of the problem but also the constraints were linearized, while in the method proposed in Chapter 3, linearization was performed only for the mapping of the problem.

In this thesis, among various merit functions which lead to an optimization formulation of the variational inequality problem, we have only focused on the regularized gap function and its modification. But we believe that the results of this thesis have revealed that such an optimization formulation serves as a promising vehicle for solving the variational inequality problem from both theoretical and practical points of view. In particular, the results obtained in Chapters 5 and 6 contribute to constructing novel efficient algorithms for variational inequality problems with general convex constraints.

We hope that this thesis contributes toward the progress of the field of variational inequality problems.

Appendix A

A.1 Mathematical review

In this section, we provide some mathematical concepts and definitions used in this thesis. For detailed expositions, one should refer [Berg63, OrR70].

A.1.1 Vectors and Matrices

Inner Product

The *inner product* of two vectors x and y in R^n is defined by $\langle x, y\rangle = \sum_{i=1}^{n} x_i y_i$. If the inner product of two vectors is zero, then the two vectors are said to be *orthogonal*.

The Euclidean norm

The *Euclidean norm* of a vector x in R^n is defined by $\| x \| = \langle x, x\rangle^{\frac{1}{2}}$. The Euclidean norm $\| \cdot \|$ has the following properties:

(a) $\| x \| \geq 0$ for all $x \in R^n$ and $\| x \| = 0$ if and only if $x = 0$.

(b) For any scalar $\alpha > 0$, we have that $\| \alpha x \| = |\alpha| \; \| x \|$.

(c) For any two vectors $x, y \in R^n$, we have that $\| x + y \| \leq \| x \| + \| y \|$.

Schwartz inequality

Let x and y be vectors in R^n. Then the following inequality, referred as the *Schwartz inequality*, holds:

$$\langle x, y\rangle \leq \| x \| \| y \| .$$

Transposition

Let A be an $m \times n$ matrix. The *transpose* of A, denoted by A^t, is an $n \times m$ matrix whose (i, j)-element is a_{ji}. An $n \times n$ matrix A is said to be *symmetric* if $A^t = A$.

Norm of matrix

The *norm* of an $n \times n$ matrix A, denoted by $\| A \|$, is defined by

$$\| A \| = \min_{x \neq 0} \frac{\| Ax \|}{\| x \|},$$

where $\| Ax \|$ and $\| x \|$ are the Euclidean norms of the corresponding vectors. It follows from the definition that, for any vector $x \in R^n$, $\| Ax \| \leq \| A \| \| x \|$.

Positive definite matrix

An $n \times n$ matrix A is said to be *positive semidefinite* if, for any vector $x \in R^n$, the inequality

$$\langle x, Ax\rangle \geq 0$$

holds. We say that A is *positive definite* if the above inequality holds strictly whenever $x \neq 0$.

A.1.2 Sets and sequences

For a vector $x \in R^n$ and a scalar $\epsilon > 0$, we denote the open sphere centered at x with radius ϵ by $B_\epsilon(x)$, i.e.,

$$B_\epsilon(x) = \{y \mid \| y - x \| < \epsilon\}.$$

Accumulation point

Consider a sequence $\{x^k\}$ in R^n. A vector $\bar{x} \in R^n$ is said to be an *accumulation point* of the sequence $\{x^k\}$ if there is a subsequence $\{x^k\}_{k \in K}$ of $\{x^k\}$ such that $\{x^k\}_{k \in K}$ converges to $\bar{x}$. Equivalently $\bar{x}$ is an accumulation point of $\{x^k\}$ if, for any $\epsilon > 0$, $B_\epsilon(x)$ contains infinitely many point of $\{x^k\}$.

Open, closed and compact sets

A subset X of R^n is said to be *open* if for every vector $x \in X$ there is an $\epsilon > 0$ such that $B_\epsilon(x) \subset X$. If X is open and if $x \in X$, then X is sometimes called a *neighborhood* of x. A set X is *closed* if and only if its complement in R^n is open. Equivalently X is closed if and only if every convergent sequence $\{x^k\}$ in X converges to a point which belongs to X. A subset X of R^n is said to be *bounded* if there is a number $L > 0$ such that $\| x \| \leq L$ for all $x \in X$. A set X is *compact* if and only if it is both closed and bounded. It is well known that every sequence $\{x^k\}$ in a compact set X have at least one accumulation point in X.

A.1.3 Functions and Mappings

Continuous functions and mappings

A function $\varphi : R^n \to R$ is said to be *continuous* at $x \in R^n$ if $\varphi(x^k) \to \varphi(x)$ whenever $x^k \to x$. Equivalently φ is continuous at x if, for any $\epsilon > 0$, there is a $\delta > 0$ such that $\| y - x \| < \delta$ implies $|\varphi(y) - \varphi(x)| < \epsilon$. The function φ is said to be continuous on R^n if it is continuous at every point $x \in R^n$. A mapping $F : R^n \to R^n$ is said to be continuous at $x \in R^n$ if all component functions F_i, $i = 1, \dots, n$, are continuous at $x \in R^n$. Also F is continuous on R^n if it is continuous at every point $x \in R^n$.

Differentiable functions and mappings

A function $\varphi : R^n \to R$ is said to be *continuously differentiable* if the partial derivatives $\partial\varphi(x)/\partial x_1, \dots, \partial\varphi(x)/\partial x_n$ exist for each $x \in R^n$ and are continuous functions of x over R^n. A *gradient* of a function φ at a point $x \in R^n$ is defined to be a column vector

$$\nabla\varphi(x) = \begin{pmatrix} \partial\varphi(x)/\partial x_1 \\ \vdots \\ \partial\varphi(x)/\partial x_n \end{pmatrix}.$$

If the second partial derivatives $\partial^2\varphi(x)/\partial x_i \partial x_j$ exist for all i, j and are continuous, then we call φ *twice continuously differentiable*. The *Hessian* of φ is defined to be an $n \times n$ symmetric matrix whose (i,j)-th component is $\partial^2\varphi(x)/\partial x_i \partial x_j$.

A mapping $F : R^n \to R^n$ is continuously differentiable if all components F_i, $i = 1, \dots, n$, are continuously differentiable and F is twice continuously differentiable if all F_i are twice continuously differentiable.

Directional derivative

Let φ be a function from R^n into $[-\infty, +\infty]$, and let x be a point where φ is finite. We say that φ is *directionally differentiable* at x in the direction d if the limit

$$\lim_{\tau \to 0, \tau > 0} \frac{\varphi(x + \tau d) - \varphi(x)}{\tau} \tag{A.1}$$

exists, and we call the limit the *directional derivative* and denote it by $\varphi'(x; d)$. It is known that a directional derivative is *positively homogeneous*, i.e., $\varphi'(x; \alpha d) = \alpha\varphi'(x; d)$ holds for any $d \in R^n$ and $\alpha > 0$.

Lipschitz continuous

Let X be a subset of R^n. A mapping $F : R^n \to R^n$ is *Lipschitz continuous* on X if there is a constant $L > 0$ such that

$$\| F(x) - F(y) \| \le L \| x - y \| \quad \text{for all } x, y \in X.$$

The Lipschitz continuity of a Jacobian ∇F is also defined as

$$\| \nabla F(x) - \nabla F(y) \| \le L \| x - y \| \quad \text{for all } x, y \in X,$$

where the norm of the left hand side represents a matrix norm.

Mean value theorems and Taylor series expansion

Let a function $\varphi : R^n \to R$ be continuously differentiable. Then, for any $x, y \in R^n$, there exists an α with $0 < \alpha < 1$ such that

$$\varphi(y) = \varphi(x) + \langle \nabla\varphi(x + \alpha(y - x)), y - x \rangle .$$

If, in addition, φ is twice continuously differentiable, there exists an α with $0 < \alpha < 1$ such that

$$\varphi(y) = \varphi(x) + \langle \nabla\varphi(x), y - x\rangle + \frac{1}{2}\left\langle y - x, \nabla^2\varphi(x + \alpha(y - x))(y - x)\right\rangle.$$

Let φ be a continuously differentiable function from R^n into R. The *first-order Taylor series expansion* of φ around x is given by the equation

$$\varphi(y) = \varphi(x) + \int_0^1 \langle \nabla\varphi(y + \tau(x - y)), x - y\rangle \, d\tau.$$

A.2 Convex sets and convex functions

This section summarizes some concepts of convexity. For more details, see [HiL93, Roc70].

Convex sets

A subset S of R^n is *convex* if, for any $x, y \in S$ and any $0 \le \alpha \le 1$, the vector $\alpha x + (1 - \alpha)y$ is contained in S. An important special case of convex set is a *polyhedral set*. A polyhedral set S of R^n is defined by

$$S = \{x \in R^n | Ax \le b\},$$

where A is an $m \times n$ matrix and b is a vector in R^m. Every polyhedral set is closed and convex. A simple example of polyhedral convex is R^n_+.

Convex functions

Let S be a convex subset of R^n. A function $\varphi : R^n \to R$ is said to be *convex* on S if, for all $x, y \in S$ and all $0 < \alpha < 1$, the inequality

$$\alpha\varphi(x) + (1 - \alpha)\varphi(y) \ge \varphi(\alpha x + (1 - \alpha)y)$$

holds. We say that φ is *strictly convex* if the above inequality holds strictly whenever $x \ne y$. A function φ is said to be *strongly convex* with modulus μ if there exist a $\mu > 0$ such that

$$\alpha\varphi(x) + (1 - \alpha)\varphi(y) \ge \varphi(\alpha x + (1 - \alpha)y) + \frac{1}{2}\mu\alpha(1 - \alpha)\,\|x - y\|^2$$

holds for all $x, y \in S$ and $0 < \alpha < 1$. A function $\varphi : R^n \to R$ is said to be *concave* (*strictly* or *strongly concave*) if $-\varphi$ is convex (strictly or strongly convex). A differentiable function φ is *pseudo convex* if $\langle \nabla\varphi(x), y - x \rangle \geq 0$ implies $\varphi(y) \geq \varphi(x)$ for all $x, y \in R^n$.

When a convex function φ is continuously differentiable on the convex set S, we have the following proposition (cf. [HiL93, page 183]).

Proposition A.1 *Let a function $\varphi : R^n \to R$ be differentiable on the convex subset S of R^n. Then*

(a) φ is convex on S if and only if

$$\varphi(y) - \varphi(x) \geq \langle \nabla\varphi(x), y - x \rangle$$

holds for all $x, y \in S$,

(b) φ is strictly convex if and only if the above inequality holds strictly whenever $x \neq y$,

(c) φ is strongly convex with modulus μ on S if and only if

$$\varphi(y) - \varphi(x) \geq \langle \nabla\varphi(x), y - x \rangle + \frac{1}{2}\mu \, \| x - y \|^2$$

holds for all $x, y \in S$.

Affine functions and mappings

A function $\varphi : R^n \to R$ is said to be *affine* if φ is both convex and concave. Each affine function φ can be represented as $\varphi(x) = \langle a, x \rangle + b$ where a is a vector in R^n and

b is a scalar (cf. [Roc70, Section 4]). An affine function is convex and pseudo-convex but neither strictly nor strongly convex. A mapping $F : R^n \to R^n$ is affine if all components F_i, $i = 1, \ldots, n$, are affine.

A.3 Nonlinear programming

In this section, we collect some concepts of the nonlinear programming problem which have been frequently appeared in this thesis. For details and many other results, see [BaS76, BSS93, Lue84].

A.3.1 Descent method

Consider the following mathematical programming problem:

$$\text{minimize } \varphi(x) \text{ subject to } x \in R^n, \tag{A.2}$$

where φ is a continuously differentiable function from R^n into R. Most typical iterative algorithms for solving (A.2) generates a sequence $\{x^k\}$ determined to be

$$x^{k+1} = x^k + \alpha_k d^k, \tag{A.3}$$

where d^k is a search direction at x^k and α_k is a positive step size parameter. An iterative algorithm (A.3) is said to be a *descent method* if the generated sequence $\{x^k\}$ satisfies $\varphi(x^{k+1}) < \varphi(x^k)$ for all k. We often call (A.3) a *descent gradient method* (or simply a *gradient method*) if the search direction d^k satisfies $\langle \nabla\varphi(x^k), d^k\rangle < 0$ whenever $\nabla\varphi(x^k) \neq 0$.

A *line search* is a procedure which determines a step size α_k of (A.3). Among various line search rule, we introduce two rules which are often used in theory and practice:

(a) *Exact line search*: α_k is chosen so that

$$\varphi(x^k + \alpha_k d^k) = \min_{0 \le \alpha \le 1} \varphi(x^k + \alpha d^k).$$

(b) *Armijo rule*: Parameters $0 < \beta < 1$ and $0 < \sigma < \frac{1}{2}$ are selected. We set $\alpha_k = \beta^{l_k}$, where l_k is the smallest nonnegative integer l such that

$$\varphi(x^k) - \varphi(x^k + \beta^l d^k) \ge -\sigma\beta^l \langle \nabla\varphi(x^k), d^k\rangle.$$

A.3.2 Karush-Kuhn-Tucker condition

Consider the following mathematical programming problem:

$$\begin{array}{ll} \text{minimize} & \varphi(x) \\ \text{subject to} & c_i(x) \le 0, \quad i = 1, \dots, m, \\ & h_j(x) = 0, \quad j = 1, \dots, l, \end{array} \tag{A.4}$$

where each f, c_i for $i = 1, \dots, m$, and h_j for $j = 1, \dots, l$, is continuously differentiable function from R^n into R. Let x^* be a solution to problem (A.4). Then, under suitable *constraint qualification*, there exist *Lagrange multipliers* λ_i^*, $i = 1, \dots, m$, and π_j^*, $j = 1, \dots, l$, such that the vector (x^*, λ^*, π^*) satisfies the *Karush-Kuhn-Tucker condition*:

$$\nabla\varphi(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla c_i(x^*) + \sum_{i=1}^{l} \pi_j^* \nabla h_j(x^*) = 0,$$

$$c_i(x^*) \le 0, \ \lambda_i^* \ge 0, \ \lambda_i^* c_i(x^*) = 0, \ i = 1, \dots, m,$$

$$h_j(x^*) = 0, \ j = 1, \dots, l.$$

Followings are the list of constraint qualifications useful in practice.

(a) *Linear independence constraint qualification*: The vectors $\nabla c_i(x^*)$ for $i \in I$ and $\nabla h_j(x^*)$ for $j = 1, \dots, l$ are linearly independent, where $I = \{i | c_i(x^*) = 0\}$.

(b) *Slater's constraint qualification*: The functions c_i, $i = 1, \ldots, m$, are all convex and h_j, $j = 1, \ldots, l$ are all affine. Furthermore, there exists an $\hat{x} \in R^n$ such that

$$c_i(\hat{x}) < 0 \text{ for } i = 1, \ldots, m \text{ and } h_j(\hat{x}) = 0 \text{ for } j = 1, \ldots, l.$$

A.3.3 Penalty function

A *penalty function* is used to transform a constrained optimization problem into a single unconstrained optimization problem. Consider the following mathematical programming problem:

$$\begin{array}{ll} \text{minimize} & \varphi(x) \\ \text{subject to} & x \in X, \end{array} \tag{A.5}$$

where f is continuous function from R^n into R and X is a subset of R^n. A *penalty function* associated with problem (A.5) is defined by

$$\varphi(x) + r\Phi(x),$$

where a parameter $r > 0$ is said to be a *penalty parameter* and Φ is a continuous function from R^n into R which has a property that $\Phi(x) \geq 0$ for all $x \in R^n$ and $\Phi(x) = 0$ if and only if $x \in X$.

Suppose that X is defined as

$$X = \{x \in R^n \mid c_i(x) \leq 0, \ i = 1, \ldots, m\}.$$

The following functions are the examples of penalty functions:

(a) *l_1 penalty function*:

$$\varphi(x) + r\sum_{i=1}^{m} \max(0, c_i(x)).$$

(b) *Quadratic penalty function*:

$$\varphi(x) + \frac{r}{2}\max(0, c_i(x))^2.$$

The quadratic penalty function is differentiable. On the other hand, the l_1 penalty function is not differentiable but is known to be *exact* [Bert75].

Bibliography

[AaM82] H. Z. Aashtiani and T. L. Magnanti, "A linearization and decomposition algorithm for computing urban traffic equilibria," in *Proceedings of the IEEE Large Scale Systems Symposium* (1982) 8-19.

[Aas79] H. Z. Aashtiani, "The multi-modal traffic assignment problem," Ph. D. thesis, Sloan School of Management, Massachusetts Institute of Technology (Cambridge, MA, 1979).

[Auc89] G. Auchmuty, "Variational principles for variational inequalities," *Numerical Functional Analysis and Optimization* 10 (1989) 863-874.

[Aus76] A. Auslender, *Optimisation: Méthodes Numériques* (Masson, Paris, 1976).

[BaS76] M. S. Bazaraa and C. M. Shetty, *Foundations of Optimization* (Springer-Verlag, Berlin Heidelberg, 1976).

[BeG82] D. P. Bertsekas and E. M. Gafni, "Projection methods for variational inequalities with application to the traffic assignment problem," *Mathematical Programming Study* 17 (1982) 139-159.

[Berg63] C. Berge, *Topological Spaces* (Oliver & Boyd, Edinburgh and London, 1963).

[Bert75] D. P. Bertsekas, "Necessary and sufficient conditions for a penalty method to be exact," *Mathematical Programming* 9 (1975) 87-99.

[BeT89] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods* (Prentice-Hall, New Jersey, 1989).

[BSS93] M. S. Bazaraa, H. D. Sherali and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms, second edition* (John wiley & Sons, New York, 1993).

[CDH90] S. C. Choi, W. S. Desarbo and P. T. Harker, "Product positioning under price competition," *Management Science* 36 (1990) 175-199.

[ChP82] D. Chan and J. S. Pang, "The generalized quasi-variational inequality problem," *Mathematics of Operations Research* 7 (1982) 211-222.

[CPS92] R. W. Cottle, J. S. Pang and R. E. Stone, *The Linear Complementarity Problem* (Academic Press, San Diego, 1992).

[Daf80] S. Dafermos, "Traffic equilibrium and variational inequalities," *Transportation Science* 14 (1980) 42-54.

[Daf83] S. Dafermos, "An iterative scheme for variational inequalities," *Mathematical Programming* 26 (1983) 40-47.

[Daf88] S. Dafermos, "Sensitivity analysis in variational inequalities," *Mathematics Operations Research* 13 (1988) 421-434.

[DaN84] S. Dafermos and A. Nagurney, "Sensitivity analysis for the asymmetric network equilibrium problem," *Mathematical Programming* 28 (1984) 174-184.

[DeS83] J.E. Dennis, Jr. and R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations* (Prentice-Hall, Englewood Cliffs, NJ, 1983).

[Eav78] B. C. Eaves, "Computing stationary points," *Mathematical Programming Study* 7 (1978) 1-14.

[FaP82] S. C. Fang and E. L. Peterson, "Generalized variational inequalities," *Journal of Optimization Theory and Applications* 38 (1982) 363-383.

[FiM90] A.V. Fiacco and G.P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques* (SIAM, Philadelphia, 1990).

[Flo89] M. Florian, "Mathematical programming applications in national, regional and urban planning," in M. Iri and K. Tanabe eds., *Mathematical Programming: Recent Developments and Applications* (KTK Scientific Publishers, Tokyo, 1989) 283-307.

[FlS82] M. Florian and H. Spiess, "The convergence of diagonalization algorithms for asymmetric network equilibrium problems," *Transportation Research* 16B (1982) 447-483.

[FTSH83] T. L. Friesz, R. L. Tobin, T. E. Smith and P. T. Harker, "A nonlinear complementarity formulation and solution procedure for the general derived demand network equilibrium problem," *Journal of Regional Science* 23 (1983) 337-359.

[Fuk85] M. Fukushima, "On the generalized variational inequality problems," in T. Hasegawa ed., *Proceedings of the International Seminar on Operations Research and Systems, Kyoto, Japan, November 22-25, 1985* (Kyoto University, Kyoto, 1985) 35-50.

[Fuk86] M. Fukushima, "A relaxed projection method for variational inequalities," *Mathematical Programming* 35 (1986) 58-70.

[Fuk92] M. Fukushima, "Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems," *Mathematical Programming*, 53 (1992) 99-110.

[Fuk96] M. Fukushima, "Merit functions for variational inequality and complementarity problems," to appear in G. Di Pillo and F. Giannessi eds., *Nonlinear Optimization and Applications* (Plenum Publishing, New York, 1996).

[GaM76] U. M. Garcia Palomares and O. L. Mangasarian, "Superlinearly convergent quasi-Newton algorithms for nonlinearly constrained optimization problems," *Mathematical Programming* 11 (1976) 1-13.

[GaZ81] C. B. Garcia and W. I. Zangwill, *Pathways to Solutions, Fixed Points, and Equilibria* (Prentice-Hall, Eaglewood Cliffs NJ, 1981).

[HaM87] J. H. Hammond and T. L. Magnanti, "Generalized descent methods for asymmetric systems of equations," *Mathematics of Operations Research* 12 (1987) 678-699.

[Han77] S. P. Han, "A globally convergent method for nonlinear programming," *Journal of Optimization Theory and Applications* 22 (1977) 297-309.

[HaP90] P. T. Harker and J. S. Pang, "Finite-dimensional variational inequality and nonlinear complementarity problems: a survey of theory, algorithms and applications," *Mathematical Programming* 48 (1990) 161-220.

[Har84] P. T. Harker, "A variational inequality approach for the determination of oligopolistic market problem," *Mathematical Programming* 30 (1984) 105-111.

[Har88] P. T. Harker, "Accelerating the convergence of the diagonalization and projection algorithms for finite-dimensional variational inequalities," *Mathematical Programming* 41 (1988) 29-59.

[HaS66] P. Hartman and G. Stampacchia, "On some non-linear elliptic differential-functional equations," *Acta Mathematica* 115 (1966) 271-310.

[Hea82] D. W. Hearn, "The gap function of a convex program," *Operations Research Letters* 1 (1982) 67-71.

[HeN82] D. W. Hearn and S. Nguyen, "Dual and saddle functions related to the gap function," Research Report No. 82-4, Department of Industrial and Systems Engineering, University of Florida (Gainesville, FL, 1982).

[HiL93] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms I* (Springer-Verlag, Berlin Heidelberg, 1993).

[HLN84] D. W. Hearn, S. Lawphongpanich and S. Nguyen, "Convex programming formulations of the asymmetric traffic assignment problem," *Transportation Research B* 18B (1984) 357-365.

[Hog73a] W. Hogan, "Directional derivatives for extremal-value functions with applications to the completely convex case," *Operations Research* 21 (1973) 187-209.

[Hog73b] W. Hogan, "Point-to-set maps in mathematical programming," *SIAM Review* 15 (1973) 591-603.

[IbF91] T. Ibaraki and M. Fukushima, *FORTRAN 77 Optimization Programming* (in Japanese) (Iwanami-Shoten, Tokyo, 1991).

[IFI88] T. Ito, M. Fukushima and T. Ibaraki, "An iterative method for variational inequalities with application to traffic equilibrium problems," *Journal of Operations Research Society of Japan* 31 (1988) 82-104.

[Jos79a] N. H. Josephy, "Newton's method for generalized equations," Technical Report No. 1965, Mathematics Research Center, University of Wisconsin (Madison, WI, 1979).

[Jos79b] N. H. Josephy, "Quasi-Newton method for generalized equations," Technical Report No. 1966, Mathematics Research Center, University of Wisconsin (Madison, WI, 1979).

[Kar69a] S. Karamardian, "The nonlinear complementarity problem with applications, part 1," *Journal of Optimization Theory and Applications* 4 (1969) 87-98.

[Kar69b] S. Karamardian, "The nonlinear complementarity problem with applications, part 2," *Journal of Optimization Theory and Applications* 4 (1969) 167-181.

[Kar71] S. Karamardian, "Generalized complementarity problem," *Journal of Optimization Theory and Applications* 8 (1971) 161-158.

[Kar72] S. Karamardian, "The complementarity problem," *Mathematical Programming* 2 (1972) 107-129.

[KiS80] D. Kinderlehrer and G. Stampacchia, *An Introduction to Variational inequalities and Their Applications* (Academic Press, New York, 1980).

[Kyp87] J. Kyparisis, "Sensitivity analysis framework for variational inequalities," *Mathematical Programming* 38 (1987) 203-213.

[Kyp90] J. Kyparisis, "Sensitivity analysis for nonlinear programs and variational Inequalities with Nonunique Multipliers," *Mathematics of Operations Research* 15 (1990) 286-298.

[LaP94] T. Larsson, M. Patriksson, "A class of gap functions for variational inequalities," *Mathematical Programming* 64 (1994) 53-79.

[LaT89] G. Van der Laan and A. J. J. Talman, "An algorithm for the linear complementarity problem with upper and lower bounds," *Journal of Optimization Theory and Applications* 62 (1989) 151-163.

[Lem65] C. E. Lemke, "Bimatrix equilibrium points and mathematical programming," *Management Science* 11 (1965) 681-689.

[LiS67] J. L. Lions and G. Stampacchia, "Variational inequalities," *Communications on Pure and Applied Mathematics* 20 (1967) 493-519.

[Lue84] D. G. Luenberger, *Linear and Nonlinear Programming: Second edition*, (Addison-Wesley, Massachusetts, 1984).

[MaD87] P. Marcotte and J. P. Dussault, "A note on a globally convergent Newton method for solving variational inequalities," *Operations Research Letters* 6 (1987) 35-42.

[MaD89] P. Marcotte and J. P. Dussault, "A sequential linear programming algorithm for solving monotone variational inequalities," *SIAM Journal on Control and Optimization* 27 (1989) 1260-1278.

[Mar85] P. Marcotte, "A new algorithm for solving variational inequalities with application to the traffic assignment problem," *Mathematical Programming* 33 (1985) 339-351.

[MaS72] O. G. Marcino and G. Stampacchia, "Convex programming and variational inequalities," *Journal of Optimization Theory and Applications* 9 (1972) 3-23.

[Mat87] L. Mathiesen, "An algorithm based on a sequence of linear complementarity problems applied to a Walrasian equilibrium model : an example," *Mathematical Programming* 37 (1987) 1-18.

[MaZ95] P. Marcotte and D. Zhu, "Global convergence of descent process for solving nonstrictly monotone variational inequalities," *Computational Optimization and Applications* 4 (1995) 127-138.

[MSS82] F. H. Murphy, H. D. Sherali and A. L. Soyster, "A mathematical programming approach for determining oligopolistic market equilibrium," *Mathematical Programming* 24 (1982) 92-106.

[NaA88] A. Nagurney and J. Aronson, "A general dynamic spatial price equilibrium model: formulation, solution, and computational results," *Journal of Computational and Applied Mathematics* 22 (1988) 339-357.

[NaA89] A. Nagurney and J. Aronson, "A general dynamic spatial price network equilibrium model with gains and losses," *Networks* 19 (1989) 751-769.

[Nag87] A. Nagurney, "Competitive equilibrium problems, variational inequalities and regional science," *Journal of Regional Science* 27 (1987) 503-517.

[Nag93] A. Nagurney, *Network Economics: A Variational Inequality Approach* (Kluwer Academic Publishers, Massachusetts, 1993).

[NgD84] S. Nguyen and C. Dupuis, "An efficient method for computing traffic equilibria in networks with asymmetric transportation costs," *Transportation Science* 18 (1984) 185-202.

[OrR70] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables* (Academic Press, New York, 1970).

[PaC82] J. S. Pang and D. Chan, "Iterative methods for variational and complementarity problems," *Mathematical Programming* 24 (1982) 284-313.

[PaG93] J. S. Pang and S. A. Gabriel, "NE/SQP: a robust algorithm for the nonlinear complementarity problem," *Mathematical Programming* 60 (1993) 295-337.

[Pan85] J. S. Pang, "Asymmetric variational inequality problems over product sets: applications and iterative Methods," *Mathematical Programming* 31 (1985) 206-219.

[Pan90] J. S. Pang, "Newton's method for B-differentiable equations," *Mathematics of Operations Research* 15 (1990) 311-341.

[Pen95] J. M. Peng, "Equivalence of variational inequality problems to unconstrained optimization," Technical Report, State Key Laboratory of Scientific and Engineering Computing, Academia Sinica (Beijing, China, 1995).

[QiM89] Y. Qiu and T. L. Magnanti, "Sensitivity analysis for variational inequalities defined on polyhedral sets," *Mathematics of Operations Research* 14 (1989) 410-432.

[Rob79] S. M. Robinson, "Generalized equations and their Solutions, part I: basic theory," *Mathematical Programming Study* 10 (1979) 128-141.

[Rob80] S. M. Robinson, "Strongly regular generalized equations," *Mathematics of Operations Research* 5 (1980) 43-62.

[Rob83] S. M. Robinson, "Generalized equations," in A. Bachem, M Grötscel and B. Korte eds., *Mathematical Programming: The State of the Art* (Springer-Verlag, Berlin Heidelberg, 1983) 346-367.

[Roc70] R. T. Rockafellar, *Convex Analysis* (Princeton, New Jersey, 1970).

[Smi79] M. J. Smith, "The existence, uniqueness and stability of traffic equilibria," *Transportation Research* 13B (1979) 295-304.

[Smi84] M. J. Smith, "A descent algorithm for solving monotone variational inequalities and monotone complementarity problems," *Journal of Optimization Theory and Applications* 44 (1984) 485-496.

[TaF94] K. Taji and M. Fukushima, "Optimization based globally convergent methods for the nonlinear complementarity problem," *Journal of Operations Research Society of Japan* 37 (1994) 310-331.

[TaF95] K. Taji and M. Fukushima, "A globally convergent Newton method for solving variational inequality problems with inequality constraints," in D.-Z. Du, L. Qi and R. S. Womersley eds., *Recent Advances in Nonsmooth Optimization* (World Scientific, Singapore, 1995) 405-417.

[TaF96] K. Taji and M. Fukushima, "A new merit function and a successive quadratic programming algorithm for variational inequality problems," to appear in *SIAM Journal on Optimization* 6 (1996).

[TFI93] K. Taji, M. Fukushima and T. Ibaraki, "A globally convergent Newton method for solving strongly monotone variational inequalities," *Mathematical Programming* 58 (1993) 369-383.

[Tob86] R. L. Tobin, "Sensitivity analysis for variational inequalities," *Journal of Optimization Theory and Applications* 48 (1986) 191-204.

[Tob88] R. L. Tobin, "A variable dimension solution approach for the general spatial price equilibrium problem," *Mathematical Programming* 40 (1988) 33-51.

[WFM93] J. H. Wu, M. Florian and P. Marcotte, "A general descent framework for the monotone variational inequality problem," *Mathematical Programming* 61 (1993) 281-300.

[XiH94a] B. Xiao and P. T. Harker, "A nonsmooth Newton method for variational inequalities, I: theory," *Mathematical Programming* 65 (1994) 151-194.

[XiH94b] B. Xiao and P. T. Harker, "A nonsmooth Newton method for variational inequalities, II: numerical results," *Mathematical Programming* 65 (1994) 195-216.

[YTF95] N. Yamashita, K. Taji and M. Fukushima, "Unconstrained optimization reformulation of variational inequality problems," Technical Report NAIST-IS-TR95024, Graduate School of Information Science, Nara Institute of Science and Technology (Nara, Japan, 1995).

[Zan69] W. I. Zangwill, *Nonlinear Programming: A Unified Approach* (Prentice-Hall, NJ, 1969).

[ZhM93] D. L. Zhu and P. Marcotte, "Modified descent methods for solving the monotone variational inequality problem," *Operations Research Letters* 14 (1993) 111-120.

[ZhM94] D. L. Zhu and P. Marcotte, "An extended descent framework for variational inequalities," *Journal of Optimization Theory and Applications* 80 (1994) 349-366.