STUDIES
ON
SECOND-ORDER CONE
COMPLEMENTARITY PROBLEMS

SHUNSUKE HAYASHI

STUDIES
ON
SECOND-ORDER CONE
COMPLEMENTARITY PROBLEMS

SHUNSUKE HAYASHI

STUDIES
ON
SECOND-ORDER CONE
COMPLEMENTARITY PROBLEMS

by

SHUNSUKE HAYASHI

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

KYOTO UNIVERSITY
KYOTO 606–8501, JAPAN
DECEMBER 2004

Preface

In this thesis, we study the second-order cone complementarity problem, SOCCP for short. This problem is to find a solution satisfying a system of equations and a complementarity condition defined on the Cartesian product of second-order cones, simultaneously

Classical complementarity problems, such as linear complementarity problems, nonlinear complementarity problems, and mixed complementarity problems, are defined on the nonnegative orthant, and have been studied extensively so far. For linear complementarity problems, Lemke's method was proposed in the 1960's as an approach to solve convex quadratic programming problems. For nonlinear complementarity problems, studies on the nonsmooth reformulation approach flourished in the 1990's, in which Fischer-Burmeister function or the min function were employed to construct an equivalent system of nonsmooth equations.

On the other hand, studies for SOCCPs have begun only recently. For example, Fukushima, Luo and Tseng introduced a scheme of analyzing SOCCPs by using Jordan algebra. It requires, however, quite different analysis from the classical complementarity problems since the Jordan product is not associative in general. Hence, many problems remain unsolved, and efficient algorithms have yet to be developed. Meanwhile, the SOCCP has a big potential of applications. Although efficient algorithms based on interior point methods have been developed for linear second-order cone programming problems (linear SOCPs), studies on algorithms for solving nonlinear SOCPs are scarce. Since the Karush-Kuhn-Tucker conditions for an SOCP take the form of SOCCP, we may apply algorithms for SOCCPs to solve SOCPs. In addition, there are many applications that are peculiar to SOCCP. For example, the robust Nash equilibrium problem to be studied in this thesis is one of such applications.

The main contribution of this thesis is to propose efficient algorithms for solving SOCCP. We propose two different types of algorithms. One is based on Newton's method in which smoothing and regularization methods are combined. This algorithm enjoys a very nice convergence property under mild conditions. The other algorithm is based on the matrix splitting method for linear complementarity problems, which is particularly effective for large-scale problems with sparse matrices. The practical efficiency of both algorithms are confirmed by means of numerical experiments.

Another contribution is to define a new concept for a non-cooperative game with incomplete information. This concept is called the robust Nash equilibrium, which is defined when each player determines a strategy by presuming the uncertainty of information. Moreover, we show that the problem of finding such an equilibrium reduces to SOCCP under appropriate assumptions.

It is not long since the studies on SOCCP has begun, and much possibility may be hidden. The author hopes that the thesis will be helpful for further studies on SOCCPs and other related problems.

Shunsuke Hayashi
December 2004

Acknowledgment

First of all, I would like to express my sincere appreciation to Professor Masao Fukushima of Kyoto University, who supervised me throughout my master's and doctoral course. Although I may have often troubled him, he kindly supported me, read my draft manuscripts carefully, and gave me precise and detailed advices. Moreover, he often spared his precious time for me to answer my trivial questions. His eager and earnest attitude to the studies was respectable and incentive. I would like to bear such a spirit in my mind, and make effort to improve my skill of research in the future.

I would also like to express my gratitude to Assistant Professor Nobuo Yamashita of Kyoto University. He suggested some ideas that led to some results shown in this thesis. When I could not make progress of my study, he gave me valuable hints to help me. Also, when he stayed in Canada for a year, he gave me plenty of kind advices by E-mail.

I would also like to tender my acknowledgments to Professor Tetsuya Takine of Osaka University. When he was an associate professor of Kyoto University, he gave me a lot of comments at seminars in the laboratory. Owing to his advices from the different realm, I could see my study not only from my own major but also from other various viewpoints.

I would like to thank all the present members, OBs and OGs of Fukushima Laboratory. It was my pleasure to have belonged to the laboratory with homely atmosphere and cheerful members. When I joined this laboratory five years ago, my seniors and contemporaries helped me to get acclimatized to the environment. Recently, my juniors have often helped me with various affairs and activities. I will never forget pleasant memories with them such as amusements, parties, recreations, travels, etc.

Finally, I would like to dedicate this thesis to my parents with my appreciation for their gratuitous and affectionate supports.

Contents

Preface **v**

Acknowledgment **vii**

List of Tables **xi**

1 Introduction **1**
1.1 Second-order cone complementarity problem 1
1.2 Overview of the thesis 3

2 Preliminaries **5**
2.1 Notations 5
2.2 Background 6
2.2.1 Subdifferential and semismoothness 7
2.2.2 Convexity and monotonicity 8
2.2.3 Jordan algebra associated with SOCs 10
2.3 Existence and uniqueness of solution 13
2.4 SOCCP-functions and nonsmooth reformulation 16

3 A combined smoothing and regularization method for monotone second-order cone complementarity problems **19**
3.1 Introduction 19
3.2 Merit function 20
3.3 Smoothing and regularization 21
3.3.1 Smoothing functions 22
3.3.2 Regularization method 23
3.4 Algorithm 24
3.4.1 Prototype algorithm 24
3.4.2 Semismoothness and strong semismoothness 26
3.4.3 Jacobian consistency of the smoothing function 27
3.4.4 Quadratically convergent algorithm with Newton's method 33
3.5 Numerical experiments 36
3.5.1 Linear case 37

3.5.2 Comparison with interior point method . . . 37
3.5.3 Nonlinear case . . . 38
3.6 Concluding remarks . . . 40

4 A matrix splitting method for symmetric affine second-order cone complementarity problems 43
4.1 Introduction . . . 43
4.2 Matrix splitting method and its convergence . . . 44
4.3 Block SOR method . . . 47
4.4 Solving subproblems . . . 51
4.5 Numerical results . . . 57
4.6 Concluding remarks . . . 60

5 Robust Nash equilibria and second-order cone complementarity problems 61
5.1 Introduction . . . 61
5.2 Robust Nash equilibria and its existence . . . 62
5.3 SOCCP formulation of robust Nash equilibrium . . . 64
5.3.1 Uncertainty in the opponent's strategy . . . 65
5.3.2 Component-wise uncertainty in the cost matrices . . . 66
5.3.3 Column/row-wise uncertainty in the cost matrices . . . 67
5.3.4 General uncertainty in the cost matrices . . . 69
5.4 Numerical examples of robust Nash equilibria . . . 71
5.4.1 Uncertainty in opponent's strategy . . . 71
5.4.2 Uncertainty in cost matrices . . . 72
5.5 Concluding remarks . . . 72

6 Conclusion 75

A Level-boundedness of the merit function and its smoothing function 77

B Alternative proof of Theorem 3.4.1 79

C Detailed proof of Proposition 3.4.2 83

D Proof of Proposition 3.4.3 87

E Well-definedness of Algorithm 3.4.2 91

F A sufficient condition for the solvability of linear SOCCP 93

List of Tables

3.1 Results for linear SOCCPs of various problem sizes 38
3.2 Results for linear SOCCPs with various degrees of rank deficiency 39
3.3 Comparison of Algorithm 3.4.2 and the interior point method on linear SOCPs . . . 39
3.4 Results for nonlinear SOCCPs with various choices of $(\eta, \overline{\eta})$ 40

4.1 Results for Procedure 4.4.1 applied to SOCCPs (4.4.1) of various problem sizes . . . 58
4.2 Results for Procedure 4.4.1 applied to SOCCPs (4.4.1) with various degrees of sparsity 58
4.3 Results for Algorithm 4.2.1 applied to SOCCPs (4.1.1) with various choices of (ω, γ) 59
4.4 Results for Algorithm 4.2.1 applied to SOCCPs (4.1.1) with various Cartesian structures of $\mathcal{K}$ 60

5.1 Robust Nash equilibria for various values of ρ_y and ρ_z 73
5.2 Robust Nash equilibria for various values of ρ_A and ρ_B (cost matrices A_1 and B_1) . 73
5.3 Robust Nash equilibria for various values of ρ_A and ρ_B (cost matrices A_2 and B_2) . 73

Chapter 1

Introduction

1.1 Second-order cone complementarity problem

In the realm of mathematical programming, there have been a large number of studies on complementarity problems. The linear complementarity problem (LCP) [9, 10, 24, 48, 61, 67, 86, 87] is the most fundamental class of complementarity problems, of which concentrated studies began in the 1960's for solving convex quadratic programming problems [23, 27]. Thereafter, Nash equilibrium problem for the bimatrix game was formulated as an LCP, and some efficient algorithms like Lemke's method has been proposed [73, 74].

The nonlinear complementarity problem (NCP) [11, 26, 70, 111, 118] is the problem where the linear function in LCP is replaced by a nonlinear function. Many practical problems such as traffic equilibrium problems [35, 41, 42, 44, 62, 72], spatial equilibrium problems [21, 94], and nonlinear Nash equilibrium problems [38, 45, 88, 89, 122] reduce to NCP. In the 1990's, studies for solving NCP made significant progress. Especially, the nonsmooth reformulation approach was one of the main research streams, which is to reformulate NCP into a system of nonsmooth equations by using Fischer-Burmeister function or the min function [36, 37, 106, 110], and to solve the nonsmooth equation by the generalized Newton methods [64, 68, 96, 98] or the smoothing Newton methods [12, 16, 69, 75, 99, 100]. Thanks to such innovative studies, a lot of results for NCP have been obtained.

Recently, the complementarity problem on second-order cones (SOCs) has attracted much attention. This problem is called the second-order cone complementarity problem (SOCCP) [15, 19, 43, 57, 92], which is to find a triple $(x, y, \zeta) \in \Re^n \times \Re^n \times \Re^\nu$ such that

$$x \in \mathcal{K},\ y \in \mathcal{K},\ x^T y = 0,\ F(x, y, \zeta) = 0, \tag{1.1.1}$$

where $F : \Re^n \times \Re^n \times \Re^\nu \to \Re^n \times \Re^\nu$ is a continuously differentiable mapping, and $\mathcal{K} \subset \Re^n$ is the Cartesian product of second-order cones, that is, $\mathcal{K} = \mathcal{K}^{n_1} \times \mathcal{K}^{n_2} \times \cdots \times \mathcal{K}^{n_m}$ with $n = n_1 + \cdots + n_m$ and the n_i-dimensional second-order cone $\mathcal{K}^{n_i} \subset \Re^{n_i}$ defined by

$$\mathcal{K}^{n_i} := \left\{ (z_1, z_2^T)^T \in \Re \times \Re^{n_i - 1} \;\middle|\; \|z_2\|_2 \le z_1 \right\}.$$

SOCCP is a wide class of problems containing NCP and second-order cone programming problem (SOCP) [2, 76, 82, 85, 112]. For a given function $f : \Re^n \to \Re^n$, NCP is to find a vector $x \in \Re^n$

such that

$$x \geq 0, \;\; f(x) \geq 0, \;\; x^T f(x) = 0. \tag{1.1.2}$$

It can be easily seen that NCP (1.1.2) is equivalent to SOCCP (1.1.1) with $n_1 = n_2 = \cdots = n_m = 1$ and $F(x, y, \zeta) = f(x) - y$ since $\mathcal{K}^1$ is the set of nonnegative reals. On the other hand, SOCP is written as

$$\begin{array}{ll} \text{minimize} & \theta(z) \\ \text{subject to} & \gamma(z) \in \mathcal{K}, \end{array} \tag{1.1.3}$$

where $\theta : \Re^n \to \Re$ and $\gamma : \Re^n \to \Re^\nu$ are continuously differentiable functions. The KKT conditions for SOCP (1.1.3) are given as

$$\lambda \in \mathcal{K}, \;\; \gamma(z) \in \mathcal{K}, \;\; \lambda^T \gamma(z) = 0, \;\; \nabla\theta(z) + \nabla\gamma(z)\lambda = 0 \tag{1.1.4}$$

with the Lagrange multiplier $\lambda \in \Re^m$. By setting $x := \lambda$, $y := \mu$, $\zeta := z$, and

$$F(\lambda, \mu, z) := \begin{pmatrix} \mu - \gamma(z) \\ \nabla\theta(z) + \nabla\gamma(z)\lambda \end{pmatrix},$$

the KKT conditions (1.1.4) reduces to SOCCP (1.1.1).

Let $\mathcal{S}^n$ be a set of $n \times n$ matrices, and $F : \mathcal{S}^n \times \mathcal{S}^n \times \Re^\nu \to \mathcal{S}^n \times \Re^\nu$ be a given function. Then, the semidefinite complementarity problem (SDCP) [49, 50, 71, 104, 120] is to find a triple $(X, Y, \zeta) \in \mathcal{S}^n \times \mathcal{S}^n \times \Re^\nu$ such that

$$X \succeq 0, \;\; Y \succeq 0, \;\; X \bullet Y = 0, \;\; F(X, Y, \zeta) = 0,$$

where $X \succeq 0$ means the positive semidefiniteness of X and the operator $\bullet$ denotes the inner product associated with matrices, i.e., $X \bullet Y := \mathrm{trace}(X^T Y)$. The KKT conditions for a semidefinite programing problem (SDP) [1, 28, 108, 115, 116, 117] can be written in the form of SDCP. Moreover, SOCCP is the special case of SDCP, which can be seen by the fact that, for $x = (x_1, x_2) \in \Re \times \Re^{n-1}$ and the matrix defined by

$$X := \begin{pmatrix} x_1 & x_2^T \\ x_2 & x_1 I \end{pmatrix},$$

we have the relation

$$X \succeq 0 \iff x \in \mathcal{K}^n.$$

However, it is not advantageous in general to reformulate an SOCCP to an equivalent SDCP and solve it as an SDCP, since SDCP deals with matrices, and therefore, is much more expensive computationally than SOCCP.

For NCP, there has been much study on algorithms and applications. On the other hand, study for SOCCP has begun recently, and hence, there are many issues to be studied. Although some

results on SOCCP-functions and their smoothing functions are shown in [43, 19, 15], any concrete algorithm has not been developed and many properties remain unclear. The main reason why more problems on SOCCP remain unsolved than NCP is that SOCCP is defined on the Cartesian product space of SOCs, which is difficult to analyze, in contrast with the fact that NCP is defined on a nonnegative orthant. For example, Fukushima, Luo and Tseng [43] introduced a scheme for analyzing SOCCP by using Jordan algebra, which requires quite different analysis from NCP since the Jordan product is not associative in general.

Another motivation to study SOCCP is its possibility of wide applications. For linear SOCPs, many efficient algorithms based on primal-dual interior point methods have been developed [76, 82, 112]. However, studies on nonlinear SOCPs are relatively scarce. Since the KKT conditions for SOCP can be regarded as an SOCCP, we may apply algorithms for SOCCPs to solve SOCPs. On the other hand, there are many applications that are peculiar to SOCCP. The robust Nash equilibrium problem [58] to be studied in Chapter 5 is one of such applications. In the field of architecture, a certain kind of equilibrium problem with frictional contact can be formulated as an SOCCP [66]. As these instances show, SOCCP is applicable not only to the KKT conditions for SOCP but also to many other practical problems.

1.2 Overview of the thesis

The thesis is organized as follows. In Chapter 2, we give some preliminaries. Especially, we give some notations, basic properties, mathematical techniques and existing results that are necessary for the later arguments. In Chapter 3, we construct an efficient algorithm for solving SOCCP, by combining a regularization method with a smoothing method, in which the smoothing function proposed in [43] is adopted. Moreover, we show global and quadratic convergence of the proposed algorithm under the monotonicity assumption. In Chapter 4, we propose an algorithm effective for large-scale SOCCPs, by extending the matrix splitting method for LCP to SOCCP. The algorithm is based on the SOR method for LCP. However, the proposed algorithm utilizes a special techniques for solving subproblems, of which structure is much more complicated than that of NCP. In Chapter 5, we define the concept of robust Nash equilibrium and show that the problem of finding such equilibria can be reformulated as an SOCCP. The robust Nash equilibrium is a new concept for the non-cooperative game, in which each player determines each strategy by taking the uncertainty of information into consideration. More detailed overview of Chapters 3 – 5 will be stated in the introduction of each chapter. In Chapter 6, we summarize the thesis and mention some future issues.

Chapter 2

Preliminaries

In this chapter, we give some definitions and basic results that will be useful in the subsequent chapters.

2.1 Notations

Throughout the thesis, we use the following notations. For a vector $x \in \Re^n$, x_i denotes the i-th component, and $\|x\|$ denotes the Euclidean norm defined by

$$\|x\| := \sqrt{x_1^2 + \cdots + x_n^2}.$$

For vectors $x \in \Re^n$ and $y \in \Re^n$, the inequality $x \geq y$ means $x_i \geq y_i$ for each $i = 1, \ldots, n$, and the max operator $\max\{x, y\}$ is defined by

$$\max\{x, y\} := \begin{pmatrix} \max\{x_1, y_1\} \\ \vdots \\ \max\{x_n, y_n\} \end{pmatrix}.$$

The min operator $\min\{x, y\}$ is defined in a similar way. For a matrix $M \in \Re^{n \times m}$, M_{ij} denotes the i-th row and j-th column component, and $\|M\|$ and $\|M\|_F$ denote the 2-norm and the Frobenius norm, respectively, that is,

$$\|M\| := \max_{\|u\|=1} \|Mu\| \quad \text{and} \quad \|M\|_F := \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{m} (M_{ij})^2}.$$

Furthermore, M_i^r and M_j^c denote the i-th row vector and the j-th column vector, respectively. For square matrices X and $Y \in \Re^{n \times n}$, $X \succ (\succeq)\, 0$ denotes the positive (semi)definiteness of X, and $X \succ (\succeq)\, Y$ means $X - Y \succ (\succeq)\, 0$. For a nonsingular matrix $X \in \Re^{n \times n}$, $X^{-T} := (X^{-1})^T = (X^T)^{-1}$. For a function $f : \Re^n \times \Re^m \to \Re$, $f(\cdot, z) : \Re^n \to \Re$ and $f(y, \cdot) : \Re^m \to \Re$ denote the functions with z and y, respectively, being fixed. For a differentiable real-valued function $g : \Re^n \to \Re$, its gradient

$\nabla g(x)$ is defined by

$$\nabla g(x) := \begin{pmatrix} \dfrac{\partial g(x)}{\partial x_1} \\ \vdots \\ \dfrac{\partial g(x)}{\partial x_n} \end{pmatrix},$$

where $\partial g(x)/\partial x_i$ $(i = 1, \ldots, n)$ denotes the partial derivative of g at x associated with its i-th component. In addition, when g is twice differentiable, its Hessian matrix $\nabla^2 g(x)$ is defined by

$$\nabla^2 g(x) := \begin{pmatrix} \dfrac{\partial^2 g(x)}{\partial x_1^2} & \cdots & \dfrac{\partial^2 g(x)}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial^2 g(x)}{\partial x_n \partial x_1} & \cdots & \dfrac{\partial^2 g(x)}{\partial x_n^2} \end{pmatrix} \in \Re^{n \times n}.$$

For a differentiable vector-valued function $G : \Re^n \to \Re^m$, $\nabla G(x)$ denotes the transposed Jacobian matrix, that is,

$$\begin{aligned} \nabla G(x) &:= \Big(\nabla G_1(x), \ldots, \nabla G_m(x)\Big) \\ &= \begin{pmatrix} \dfrac{\partial G_1(x)}{\partial x_1} & \cdots & \dfrac{\partial G_m(x)}{\partial x_1} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial G_1(x)}{\partial x_n} & \cdots & \dfrac{\partial G_m(x)}{\partial x_n} \end{pmatrix} \in \Re^{n \times m} \end{aligned}$$

For a set X, $\mathcal{P}(X)$ denotes the set consisting of all the subsets of X. We define the sets $\Re^n_+$, $\mathcal{S}^n$ and $\mathcal{S}^n_+$ by

$$\begin{aligned} \Re^n_+ &:= \{x \in \Re^n \,|\, x \geq 0\}, \\ \mathcal{S}^n &:= \{X \in \Re^{n \times n} \,|\, X = X^T\}, \\ \mathcal{S}^n_+ &:= \{X \in \Re^{n \times n} \,|\, X = X^T,\ X \succeq 0\}, \end{aligned}$$

respectively. We often denote the vector $(x^T, y^T)^T \in \Re^{n+m}$ as $(x, y) \in \Re^n \times \Re^m$. However, for n-dimensional vectors $x^1, \ldots, x^m$, we may regard $(x^1, \ldots, x^m)$ as an $n \times m$ matrix. In this case, we will mention its dimension explicitly. For any positive integer n, $I_n \in \Re^{n \times n}$ denotes the identity matrix, and $e_n \in \Re^n$ denotes the vector of ones. $B(z, \varepsilon)$ denotes the open sphere with center z and radius ε. For a given set S, $\mathrm{bd}\, S$, $\mathrm{int}\, S$, $\mathrm{cl}\, S$ and $\mathrm{co}\, S$ denote the boundary, the interior, the closure and the convex hull of S, respectively.

2.2 Background

In this section, we give some definitions and mathematical background that are necessary in later discussions. In particular, Subsections 2.2.1 and 2.2.2 are devoted to the basic properties of functions, and Subsection 2.2.3 is devoted to those of second-order cones.

2.2.1 Subdifferential and semismoothness

We first focus on the properties related to differentiation. Especially, we give some concepts which extend the idea of differentiability to nondifferentiable functions.

As a preparation, we define local and global Lipschitzian properties.

Definition 2.2.1 *Function $H : \Re^n \to \Re^m$ is said to be locally Lipschitzian if, for any bounded set $\Omega \subset \Re^n$, there exists $\kappa > 0$ such that*

$$\|H(x) - H(y)\| \leq \kappa \|x - y\| \quad \forall x, y \in \Omega.$$

Moreover, if the constant κ does not depend on Ω, the function H is said to be globally Lipschitzian.

Any affine function is globally Lipschitzian, and most of the functions dealt with in the thesis are locally Lipschitzian. For a counterexample, function $f : \Re \to \Re$ defined by $f(\alpha) := \sqrt[3]{\alpha}$ is not locally Lipschitzian. Next, we define some notions of generalized Jacobian for nonsmooth locally Lipschitzian functions.

Definition 2.2.2 *Let $H : \Re^n \to \Re^m$ be a Lipschitzian function. Then, the set-valued mappings $\partial_B H$ and $\partial H : \Re^n \to \mathcal{P}(\Re^{n \times m})$ defined by*

$$\begin{aligned} \partial_B H(w) &:= \Big\{ \lim_{k \to \infty} \nabla H(w^k) \;\Big|\; \lim_{k \to \infty} w^k = w,\ \{w^k\} \subseteq \mathcal{D}_H \Big\}, \\ \partial H(w) &:= \operatorname{co} \partial_B H(w) \end{aligned}$$

are said to be B(ouligant) subdifferential and Clarke subdifferential of H, respectively, where $\mathcal{D}_H := \{w \in \Re^n \,|\, H \text{ is differentiable at } w\}$ and 'co' stands for the convex hull. In particular, Clarke subdifferential is merely called 'subdifferential'.

For instance, when $H : \Re \to \Re$ is defined by $H(w) = |w|$, we have $\partial_B H(0) = \{-1, 1\}$ and $\partial H(0) = [-1, 1]$. These generalized Jacobians are natural extensions of the original Jacobian for differentiable functions. Indeed, if H is continuously differentiable at w, then it is easily seen $\partial_B H(w) = \partial H(w) = \{\nabla H(w)\}$. Next, we give the definition of directional differentiability.

Definition 2.2.3 *Let $H : \Re^n \to \Re^m$ be a locally Lipschitzian function. Let $w \in \Re^n$ and $d \in \Re^n \backslash \{0\}$. Then, the function H is said to be directionally differentiable at w along d, if there exists a limit $H'(w; d)$ defined by*

$$H'(w; d) := \lim_{h \downarrow 0} \frac{H(w + hd) - H(w)}{h}.$$

Moreover, $H'(w; d)$ is said to be the directional derivative of H at w along d.

For a differentiable function, $H'(w; d)$ is simply given by $\nabla H(w)^T d$.

Finally, by using the subdifferential and the directional derivative, we define semismoothness and strong semismoothness, which play crucial role in establishing the rapid convergence of Newton-type methods for nonsmooth functions. Especially, semismoothness is related to the superlinear convergence and strong semismoothness is related to the quadratic convergence.

Definition 2.2.4 *A directionally differentiable and locally Lipschitzian function $H : \Re^n \to \Re^m$ is said to be semismooth at w if*

$$V^T d - H'(w\,;d) \;=\; o(\|d\|)$$

for any $d \in \Re^n \backslash \{0\}$ sufficiently small and $V \in \partial H(w+d)$. Moreover, if $o(\|d\|)$ can be replaced by $O(\|d\|^2)$, then function H is said to be strongly semismooth.

It is known that any piecewise smooth function[1] is semismooth [91]. Moreover, any piecewise smooth function is strongly semismooth if all the derivatives of smooth pieces are locally Lipschitzian.

2.2.2 Convexity and monotonicity

In this subsection, we define the concepts of convexity and monotonicity, which play an important role throughout the thesis. We first define the convexity for sets and real-valued functions.

Definition 2.2.5 *A set $S \subseteq \Re^n$ is said to be convex if $(1-\alpha)x + \alpha y \in S$ holds for any vectors $x, y \in S$ and scalar $\alpha \in (0,1)$.*

Definition 2.2.6 *Let $X \subseteq \Re^n$ be a nonempty and convex set. Then, a function $\theta : X \to \Re$ is said to be*

(a) *convex (on X) if $\theta((1-\alpha)x+\alpha y) \le (1-\alpha)\theta(x)+\alpha\theta(y)$ holds for any $x, y \in X$ and $\alpha \in (0,1)$.*

(b) *strictly convex (on X) if $\theta((1-\alpha)x+\alpha y) < (1-\alpha)\theta(x)+\alpha\theta(y)$ holds for any $x, y \in X$ with $x \neq y$ and $\alpha \in (0,1)$.*

(c) *strongly convex (on X) with modulus $\varepsilon > 0$ if $\theta((1-\alpha)x+\alpha y) \le (1-\alpha)\theta(x)+\alpha\theta(y)+\frac{1}{2}\varepsilon(1-\alpha)\alpha\|x-y\|^2$ holds for any $x, y \in X$ and $\alpha \in (0,1)$.*

It is obvious that any strongly convex function is strictly convex, and any strictly convex function is convex. For example, a linear function is convex but not strictly convex, $\theta(\alpha) = e^\alpha$ is strictly convex but not strongly convex, and $\theta(\alpha) = \alpha^2$ is strongly convex. Convexity plays a crucial role in the field of optimization. In particular, if functions $\gamma_1, \ldots, \gamma_m$ and θ in the nonlinear programming problem :

$$\text{minimize } \theta(z) \quad \text{subject to } \gamma_i(z) \le 0 \;\; (i = 1, \ldots, m)$$

are convex, then any local minimum of the problem is a global minimum.

Next, we define monotonicity for vector-valued mappings from a subset of $\Re^n$ to $\Re^n$.

Definition 2.2.7 *Let $X \subseteq \Re^n$ be a nonempty set. Then, a function $f : X \to \Re^n$ is said to be*

[1] Function $H : \Re^n \to \Re^m$ is said to be piecewise smooth at z if there exist $\varepsilon > 0$ and a finite number of continuously differentiable functions $H_i : B(z,\varepsilon) \to \Re^m \;\; (i = 1, \ldots, N)$ such that for any $z' \in B(z,\varepsilon)$ there exists an index i such that $H(z') = H_i(z')$, where $B(z,\varepsilon)$ denotes the sphere with center z and radius ε.

(a) *monotone (on X) if $(x-y)^T(f(x)-f(y)) \geq 0$ holds for any $x, y \in X$.*

(b) *strictly monotone (on X) if $(x-y)^T(f(x)-f(y)) > 0$ holds for any $x, y \in X$ with $x \neq y$.*

(c) *strongly monotone (on X) with modulus $\varepsilon > 0$ if $(x-y)^T(f(x)-f(y)) \geq \varepsilon\|x-y\|^2$ holds for any $x, y \in X$.*

It is obvious that any strongly monotone function is strictly monotone, and strictly monotone function is monotone. For $n = 1$, the monotonicity in the above sense corresponds to monotonic nondecrease and the strict monotonicity corresponds to monotonic increase. As will be stated in Subsection 2.3, monotonicity plays an important role in analyzing the solution set of SOCCP.

The following proposition shows the relation between convexity and monotonicity.

Proposition 2.2.1 *Let $X \subseteq \Re^n$ be an open convex set, and $\theta : X \to \Re$ be a continuously differentiable function. Then,*

(a) *θ is convex if and only if $\nabla\theta$ is monotone.*

(b) *θ is strictly convex if and only if $\nabla\theta$ is strictly monotone.*

(c) *θ is strongly convex with modulus $\varepsilon > 0$ if and only if $\nabla\theta$ is strongly monotone with modulus $\varepsilon > 0$.*

As the above proposition shows, there is close relation between convexity and monotonicity. Moreover, these properties also have much relevance to the positive (semi)definiteness of matrices.

Proposition 2.2.2 *Let $X \subseteq \Re^n$ be an open convex set, and $f : X \to \Re^n$ be a continuously differentiable function. Then,*

(a) *f is monotone if and only if $\nabla f(x)$ is positive semidefinite for any $x \in X$.*

(b) *f is strictly monotone if $\nabla f(x)$ is positive definite for any $x \in X$.*

(c) *f is strongly monotone if and only if there exists $\varepsilon > 0$ such that*

$$\min_{\|e\|=1} e^T \nabla f(x) e \geq \varepsilon \tag{2.2.1}$$

for any $x \in X$.

Note that (b) does not hold when "if" is replaced by "if and only if". For example, though a function $f : \Re \to \Re$ defined by $f(\alpha) = \alpha^3$ is monotonically increasing on $\Re$, $\nabla f(\alpha) = 3\alpha^2$ is not positive when $\alpha = 0$. The above two propositions directly lead to the following corollary which mentions the relation between the convexity of a real-valued function and the positive (semi)definiteness of its Hessian matrix.

Corollary 2.2.1 *Let $X \subseteq \Re^n$ be an open convex set, and $\theta : X \to \Re$ be a twice continuously differentiable function. Then,*

(a) *θ is convex if and only if $\nabla^2\theta(x)$ is positive semidefinite for any $x \in X$.*

(b) *θ is strictly convex if $\nabla^2\theta(x)$ is positive definite for any $x \in X$.*

(c) *θ is strongly convex if and only if there exists $\varepsilon > 0$ such that*

$$\min_{\|e\|=1} e^T \nabla^2\theta(x) e \geq \varepsilon \tag{2.2.2}$$

for any $x \in X$.

In addition, we easily obtain the following corollary on monotonicity of affine functions and convexity of quadratic functions.

Corollary 2.2.2 *Let $M \in \Re^{n\times n}$ and $q \in \Re^n$ be a given matrix and a vector, respectively. Let $f : \Re^n \to \Re$ and $\theta : \Re \to \Re$ be defined by $f(x) = Mx + q$ and $\theta(x) = \frac{1}{2}x^T Mx + q^T x$, respectively. Then, we have*

(a) *M is positive semidefinite $\iff$ f is monotone $\iff$ θ is convex,*

(b) *M is positive definite $\iff$ f is strongly monotone $\iff$ θ is strongly convex.*

This corollary implies that the strong monotonicity is equivalent to the strict monotonicity for affine functions, and that the strong convexity is equivalent to the strict convexity for quadratic functions.

2.2.3 Jordan algebra associated with SOCs

In this subsection, we introduce Jordan algebra, which provides a useful methodology of dealing with SOCs. Although Jordan algebra originally targets on not only SOCs but also general sets in Euclidean space, we restrict ourselves to Jordan algebra associated with SOCs. For more detail, see [34, 43].

For two n-dimensional vectors $x = (x_1, x_2) \in \Re \times \Re^{n-1}$ and $y = (y_1, y_2) \in \Re \times \Re^{n-1}$, Jordan product is defined by

$$x \circ y := (x^T y,\ x_1 y_2 + y_1 x_2).$$

Note that Jordan product generates an n-dimensional vector from two n-dimensional vectors. This is one of the main differences from inner product, which generates a scalar. For convenience, we denote $x \circ x$ by x^2, and define the vector e by

$$e := (1, 0, \ldots, 0).$$

Then, Jordan product enjoys the following properties.

Property 2.2.1 *For any x, y, $z \in \Re^n$, we have*

(a) $e \circ x = x$,

(b) $x \circ y = y \circ x$,

(c) $(x + y) \circ z = x \circ z + y \circ z$,

(d) $x \circ (x^2 \circ y) = x^2 \circ (x \circ y)$.

(a) implies vector e plays the role of identity. (b) and (c) imply commutativity and distributivity, respectively. (d) is a certain kind of commutativity, which is useful in identifying the Jordan product. Indeed, (b) and (d) are the original axioms of Jordan algebra. Although Jordan product can be regarded as the multiplication, associativity does not hold, i.e., $x \circ (y \circ z) \neq (x \circ y) \circ z$ in general. On the other hand, associativity holds under the inner product in the sense that

$$x^T(y \circ z) \,=\, y^T(z \circ x) = z^T(x \circ y).$$

When $x \in \mathcal{K}^n$, the square root can be defined by

$$x^{1/2} := \left(s,\, \frac{x_2}{2s}\right), \quad \text{where} \quad s = \sqrt{\left(x_1 + \sqrt{x_1^2 - \|x_2\|^2}\right)/2}.$$

By an easy calculation, it can be verified that $x = x^{1/2} \circ x^{1/2}$ and $x^{1/2} \in \mathcal{K}^n$. The following proposition shows another expression of SOC complementarity.

Proposition 2.2.3 *For any $(x, y) \in \Re^n \times \Re^n$, the following relation holds:*

$$x \in \mathcal{K}^n,\ y \in \mathcal{K}^n,\ x^T y = 0 \iff x \in \mathcal{K}^n,\ y \in \mathcal{K}^n,\ x \circ y = 0.$$

From this proposition, SOCCP (1.1.1) can be rewritten equivalently as

$$x \in \mathcal{K},\ y \in \mathcal{K},\ x \circ_{\mathcal{K}} y = 0,\ F(x, y, \zeta) = 0,$$

where the operator $\circ_{\mathcal{K}}$ denotes the Jordan product for the Cartesian product $\mathcal{K} = \mathcal{K}^{n_1} \times \cdots \times \mathcal{K}^{n_m}$ given by $x \circ_{\mathcal{K}} y = (x^1 \circ y^1, \ldots, x^m \circ y^m)$ with $x = (x^1, \ldots, x^m)$ and $y = (y^1, \ldots, y^m) \in \Re^{n_1} \times \cdots \times \Re^{n_m}$.

Next we define spectral factorization, which is one of the basic concepts in Jordan algebra. For any vector $z = (z_1, z_2) \in \Re \times \Re^{n-1}$ $(n \geq 2)$, its spectral factorization with respect to the second-order cone $\mathcal{K}^n$ is defined as

$$z = \lambda_1 u^{\{1\}} + \lambda_2 u^{\{2\}}, \tag{2.2.3}$$

where λ_1 and λ_2 are the *spectral values* given by

$$\lambda_i \,=\, z_1 + (-1)^i \|z_2\|, \quad i = 1,\, 2, \tag{2.2.4}$$

and $u^{\{1\}}$ and $u^{\{2\}}$ are the *spectral vectors* given by

$$u^{\{i\}} \,=\, \begin{cases} \dfrac{1}{2}\left(1, (-1)^i \dfrac{z_2}{\|z_2\|}\right) & (z_2 \neq 0), \\ \dfrac{1}{2}(1, (-1)^i w) & (z_2 = 0), \end{cases} \qquad i = 1,\, 2, \tag{2.2.5}$$

with $w \in \Re^{n-1}$ such that $\|w\| = 1$. The spectral vectors $u^{\{1\}}$ and $u^{\{2\}}$ are obtained as the directional vectors of two half lines derived from the intersection of $\mathrm{bd}\,\mathcal{K}^n$ and Ω, where Ω is the 2-dimensional subspace spanned by two vectors z and e. Hence, the spectral factorization can be regarded as the linear factorization in 2-dimensional subspace Ω with bases $u^{\{1\}}$ and $u^{\{2\}}$. We note that the spectral values and vectors possess the following properties.

Property 2.2.2 *Let λ_1 and λ_2 be the spectral values of z, and $u^{\{1\}}$ and $u^{\{2\}}$ be the spectral vectors of z. Then the following properties hold.*

(a) $u^{\{1\}} \circ u^{\{2\}} = 0, \quad \|u^{\{1\}}\| = \|u^{\{2\}}\| = 1/\sqrt{2}$

(b) $u^{\{i\}} \in \mathrm{bd}\,\mathcal{K}^n, \quad u^{\{i\}} \circ u^{\{i\}} = u^{\{i\}} \quad i = 1, 2$

(c) $\lambda_1 \le \lambda_2, \quad \lambda_1 \ge 0 \iff z \in \mathcal{K}^n$

In the above properties, (a) indicates that $u^{\{1\}}$ and $u^{\{2\}}$ play a role like orthonormal bases.

Let $\hat{g}$ be a function from $\Re$ to $\Re$. Then, we define a vector-valued function $g : \Re^n \to \Re^n$ associated with $\hat{g}$ by

$$g(z) := \hat{g}(\lambda_1)u^{\{1\}} + \hat{g}(\lambda_2)u^{\{2\}}, \tag{2.2.6}$$

where z is expressed as (2.2.3). From (a) and (b) of Property 2.2.2, we obtain

$$z^2 = \lambda_1^2 u^{\{1\}} + \lambda_2^2 u^{\{2\}},$$

which leads us the following proposition.

Proposition 2.2.4 *Suppose that $\hat{g} : \Re^n \to \Re^n$ can be written in a power series expansion, i.e., $\hat{g}(\alpha) = \sum_{k=0}^{\infty} a_k \alpha^k$ with real coefficients $a_0, a_1, \ldots$ Then the function g defined by* (2.2.6) *is written as*

$$g(z) = \sum_{k=0}^{\infty} a_k z^k,$$

where z^k denotes k power of z by Jordan product and x^0 equals $e = (1, 0, \ldots, 0)$.

By this proposition, $\exp(z)$, $\ln(z)$, $\sin(z)$ and $\sinh(z)$ can be defined by using power series expansion for scalar-valued functions. On the other hand, the square root $z^{1/2}$ and the projection $P_{\mathcal{K}^n}$ onto $\mathcal{K}^n$, i.e.,

$$P_{\mathcal{K}^n}(z) := \operatorname*{argmin}_{z' \in \mathcal{K}^n} \|z' - z\|, \tag{2.2.7}$$

can be expressed explicitly by using the spectral factorization as follows:

$$\begin{aligned} z^{1/2} &= \sqrt{\lambda_1}u^{\{1\}} + \sqrt{\lambda_2}u^{\{2\}}, \\ P_{\mathcal{K}^n}(z) &= \max\{0, \lambda_1\}u^{\{1\}} + \max\{0, \lambda_2\}u^{\{2\}}. \end{aligned} \tag{2.2.8}$$

The following proposition, which was first proved by Chen, Chen and Tseng [15] and alternatively proved by Chen [14], asserts that the vector-valued function $g : \Re^n \to \Re^n$ inherits several properties from the scalar-valued function $\hat{g}$.

Proposition 2.2.5 *Let $\hat{g} : \Re \to \Re$ be a given scalar-valued function, and g be defined by* (2.2.6). *Then, the following statements hold.*

(a) *g is continuous at $x \in \Re^n$ with spectral values λ_1 and λ_2 if and only if $\hat{g}$ is continuous at λ_1 and λ_2.*

(b) *g is directionally differentiable at $x \in \Re^n$ with spectral values λ_1 and λ_2 if and only if $\hat{g}$ is directionally differentiable at λ_1 and λ_2.*

(c) *g is differentiable at $x \in \Re^n$ with spectral values λ_1 and λ_2 if and only if $\hat{g}$ is differentiable at λ_1 and λ_2.*

(d) *g is continuously differentiable at $x \in \Re^n$ with spectral values λ_1 and λ_2 if and only if $\hat{g}$ is continuously differentiable at λ_1 and λ_2.*

(e) *g is globally Lipschitzian with modulus κ if and only if $\hat{g}$ is globally Lipschitzian with modulus κ.*

(f) *g is semismooth at $x \in \Re^n$ with spectral values λ_1 and λ_2 if and only if $\hat{g}$ is semismooth at λ_1 and λ_2.*

(g) *g is strongly semismooth at $x \in \Re^n$ with spectral values λ_1 and λ_2 if and only if $\hat{g}$ is strongly semismooth at λ_1 and λ_2.*

2.3 Existence and uniqueness of solution

The purpose of this section is to give sufficient conditions for the existence and the uniqueness of a solution to SOCCP. To this end, we introduce a class of problems called the variational inequality problem, and discuss the feature of its solution set.

For a continuous function $f : \Re^n \to \Re^n$ and a closed convex set $S \subseteq \Re^n$, the variational inequality problem (VI) [8, 25, 33, 39, 40, 52, 107] is to find a vector $z \in S$ such that

$$f(z)^T(z' - z) \geq 0, \quad \forall z' \in S. \tag{2.3.1}$$

VI is a very large class of problems containing systems of equations, convex programming problems, and complementarity problems. In particular, when $S = \Re^n$, VI (2.3.1) is equivalent to the equation :

$$f(z) = 0.$$

For a continuously differentiable convex function $\theta : \Re^n \to \Re$ such that

$$f(z) = \nabla\theta(z) \quad \forall z \in \Re^n,$$

VI (2.3.1) is equivalent to the constrained minimization problem :

$$\text{minimize } \theta(z) \quad \text{subject to } z \in S. \tag{2.3.2}$$

If S is a closed convex cone such that $-S \cap S = \{0\}$, then VI (2.3.1) can be rewritten as the complementarity problem [33], which is to find a vector $z \in \Re^n$ such that

$$z \in S, \quad f(z) \in S^*, \quad z^T f(z) = 0, \tag{2.3.3}$$

where S^* is the dual cone of S defined by

$$S^* := \left\{ x \;\middle|\; x^T y \geq 0, \ \forall y \in S \right\}.$$

A cone S satisfying $S = S^*$ is called self-dual. For example, the nonnegative cone R^n_+, the second-order cone $\mathcal{K}^n$, and the semidefinite cone $\mathcal{S}^n_+$ are self-dual. Moreover, the Cartesian product of self-dual cones is also self-dual. Thus, VI (2.3.1) with $S = \Re^n_+$ is NCP (1.1.2), and VI (2.3.1) with $S = \mathcal{K}$ is the special SOCCP of finding vectors $(x, y) \in \Re^n \times \Re^n$ such that

$$x \in \mathcal{K}, \ \ y \in \mathcal{K}, \ \ x^T y = 0, \ \ f(x) - y = 0. \tag{2.3.4}$$

Furthermore, by setting

$$z := (x, y, \zeta), \quad f(z) := \begin{pmatrix} y \\ F(x, y, \zeta) \end{pmatrix}, \quad S = \mathcal{K} \times \Re^n \times \Re^\nu,$$

VI (2.3.1) reduces to the original SOCCP (1.1.1).

As we mentioned above, VI contains many classes of problems with various functions f and sets S. On the other hand, it is difficult to solve VI (2.3.1) itself in general, and therefore, we have to restrict ourselves to special S like the nonnegative cone $\Re^n_+$ and the second-order cone $\mathcal{K}$. But, even if S is a general closed convex set, it is possible to discuss the solvability of VI. The result for VI can be applied to SOCCP directly, since SOCCP is a subclass of VI.

Next, we discuss the feature of the solution set to VI (2.3.1). We first give the following proposition.

Proposition 2.3.1 *Let $f : \Re^n \to \Re^n$ be a continuous function, and $S \subseteq \Re^n$ be a nonempty closed convex set. If S is bounded, then VI (2.3.1) has at least one solution.*

This proposition guarantees the solvability of VI under the boundedness assumption on S. However, in the case of SOCCP, the set S is obviously unbounded since it is a cone. In order to consider the solvability of VI with an unbounded set, we introduce the following property for function f.

Definition 2.3.1 *Let $f : \Re^n \to \Re^n$. Suppose that S is an unbounded set. Then, the function f is said to be coercive on S if there exists $x^0 \in S$ such that*

$$\lim_{\substack{\|x\| \to \infty \\ x \in S}} \frac{f(x)^T (x - x^0)}{\|x\|} = +\infty.$$

The following proposition shows the existence of a solution under the coerciveness of f.

Proposition 2.3.2 *Let $f : \Re^n \to \Re^n$ be a continuous function, and $S \subseteq \Re^n$ be a nonempty closed convex set. If f is coercive on S, then VI (2.3.1) has at least one solution.*

Next we discuss the convexity and singleton property of the solution set. Recall that VI (2.3.1) is equivalent to the minimization problem (2.3.2) if $f(x) = \nabla\theta(x)$ for all x. Moreover, from Proposition 2.2.1, the gradient mapping $\nabla\theta$ is monotone if θ is convex. Hence, it can be expected that a similar result holds between the solution set of a minimization problem with a convex objective function and that of VI with a monotone function. The following proposition indicates the validity of this expectation.

Proposition 2.3.3 *Let $f : \Re^n \to \Re^n$ be a continuous function, and $S \subseteq \Re^n$ be a nonempty closed convex set. Then,*

(a) *the solution set of VI*(2.3.1) *is convex* (*possibly empty*) *if f is monotone on S.*

(b) *VI*(2.3.1) *has at most one solution if f is strictly monotone on S.*

(c) *VI*(2.3.1) *has a unique solution if f is strongly monotone on S.*

We emphasize that neither (a) nor (b) guarantees the existence of solution, though (c) guarantees the existence. Indeed, it is easily seen that any strongly monotone function is coercive. The above discussions readily yield the following corollary on the existence and uniqueness of a solution to SOCCP.

Corollary 2.3.1 *Let $f : \Re^n \to \Re^n$ be a continuous function. Then,*

(a) *SOCCP*(2.3.4) *has at least one solution if f is coercive on $\mathcal{K}$.*

(b) *the solution set of SOCCP*(2.3.4) *is closed and convex* (*possibly empty*) *if f is monotone on $\mathcal{K}$.*

(c) *SOCCP*(2.3.4) *has at most one solution if f is strictly monotone on $\mathcal{K}$.*

(d) *SOCCP*(2.3.4) *has a unique solution if f is strongly monotone on $\mathcal{K}$.*

2.4 SOCCP-functions and nonsmooth reformulation

In this section, we introduce an SOCCP-function, which is useful in reformulating SOCCP into an equivalent system of equations. A function $\varphi : \Re^n \times \Re^n \to \Re^n$ satisfying the following condition is called an SOCCP-function :

$$\varphi(x, y) = 0 \iff x \in \mathcal{K}^n,\ y \in \mathcal{K}^n,\ x^T y = 0. \tag{2.4.1}$$

It is important to consider such functions to construct an algorithm for solving SOCCP.

Several SOCCP-functions can be defined by using Jordan algebra. Recall that the projection of vector z onto $\mathcal{K}^n$ can be written explicitly as

$$P_{\mathcal{K}^n}(z) = \max\{0, \lambda_1\}u^{\{1\}} + \max\{0, \lambda_2\}u^{\{2\}},$$

where λ_1 and λ_2 are the spectral values and $u^{\{1\}}$ and $u^{\{2\}}$ are the spectral vectors of z, which are given by (2.2.4) and (2.2.5), respectively. By using the above formula, the natural residual, which serves as an SOCCP-function and will be used in Chapter 3, is defined by

$$\varphi_{\mathrm{NR}}(x, y) = x - P_{\mathcal{K}^n}(x - y).$$

Originally, the natural residual is proposed for VI (2.3.1), which is given by $\varphi_{\mathrm{NR}}(x) = x - P_S(x - f(x))$. For a general closed convex set S, the projection P_S cannot be expressed explicitly. However, in the case of SOCCP, the projection can be expressed explicitly by using Jordan algebra. For NCP, the projection is given by $P_{\Re^n_+}(x-y) = \max\{0, x-y\}$, and hence, the natural residual is calculated as

$$\begin{aligned}\varphi_{\mathrm{NR}}(x, y) &= x - \max\{0, x - y\} \\ &= \min\{x, y\}.\end{aligned}$$

Actually, the natural residual for NCP is called the min function.

Chen, Sun and Sun [19] proposed a penalized natural residual, in which a penalty term is added to the natural residual. This function is defined by

$$\varphi_{\mathrm{PNR}}(x, y) = x - P_{\mathcal{K}^n}(x - y) + P_{\mathcal{K}^n}(x) \circ P_{\mathcal{K}^n}(y),$$

which also serve as a SOCCP-function. Chen, Sun and Sun [19] have proved that function ψ defined by $\psi(x) := \|\varphi_{\mathrm{PNR}}(x, f(x))\|$ is level-bounded[2] under monotonicity assumption on f. This result is due to the added penalty term $P_{\mathcal{K}^n}(x) \circ P_{\mathcal{K}^n}(y)$. Indeed, the level-boundedness is a fundamental property in establishing the global convergence of an algorithm.

Next we give another SOCCP-function by using the square and the square root in Jordan algebra. The following function is called Fischer-Burmeister function :

$$\varphi_{\mathrm{FB}}(x, y) = (x^2 + y^2)^{1/2} - x - y.$$

[2]Function $\psi : \Re^n \to \Re$ is said to be level-bounded if the level set $L_\alpha := \{x \,|\, \psi(x) \le \alpha\}$ is bounded for any $\alpha \in \Re$.

This function also satisfies (2.4.1). In addition, Fischer-Burmeister function is nondifferentiable only at the origin $(0,0)$, while the natural residual is nondifferentiable at every point (x,y) such that $x-y \in \mathrm{bd}\,\mathcal{K}^n$. Note that, when $n=1$, Fischer-Burmeister function is given by

$$\varphi_{\mathrm{FB}}(\alpha,\beta) = \sqrt{\alpha^2+\beta^2} - \alpha - \beta,$$

which is the well-known Fischer-Burmeister function for NCP.

By using an arbitrary SOCCP-function φ, SOCCP (1.1.1) can be reformulated into a system of equations. Actually, let function $H : \Re^{2n+\nu} \to \Re^{2n+\nu}$ be defined by

$$H(w) := \begin{pmatrix} \varphi(x^1,y^1) \\ \vdots \\ \varphi(x^m,y^m) \\ F(x,y,\zeta) \end{pmatrix}, \tag{2.4.2}$$

where $w := (x,y,\zeta)$ and $x=(x^1,\ldots,x^m)$, $y=(y^1,\ldots,y^m) \in \Re^{n_1}\times\cdots\times\Re^{n_m}$. Then, SOCCP (1.1.1) reduces to the equation:

$$H(w) = 0. \tag{2.4.3}$$

If the SOCCP-function φ is continuously differentiable, then Newton's method with the direction

$$d^k = -\nabla H(w^k)^{-T} H(w^k)$$

is applicable. However, all SOCCP-functions proposed so far are nondifferentiable, and hence, the conventional Newton's method cannot be applied. In order to solve a nonsmooth equation, we should adopt special techniques such as the generalized Newton method and the smoothing Newton method.

The generalized Newton method was first proposed by Qi and Sun [98], in which the following iteration was employed:

$$w^{k+1} := w^k - V_k^{-T} H(w^k), \quad V_k \in \partial H(w^k). \tag{2.4.4}$$

If H is differentiable at w^k, then the iteration (2.4.4) is the same as that of Newton's method. Qi and Sun [98] showed that, if function H is semismooth at the solution w^* of equation (2.4.3) and every element of $\partial H(w^*)$ is nonsingular, then the sequence generated by (2.4.4) converges to w^* superlinearly. We note that, since all the aforementioned SOCCP-functions are semismooth [57, 19, 15], function H defined by (2.4.2) is semismooth everywhere.

For the rapid convergence of the above-mentioned method, every element of $\partial H(w^*)$ needs to be nonsingular. Since this demand is rather strict in practical, Qi [96] proposed another generalized Newton method with the iteration:

$$w^{k+1} := w^k - V_k^{-T} H(w^k), \quad V_k \in \partial_B H(w^k), \tag{2.4.5}$$

where Clarke subdifferential in (2.4.4) is replaced by B-subdifferential. Qi [96] showed that the above generalized Newton method is superlinearly convergent if H is semismooth at the solution

w^* and every element of $\partial_B H(w^*)$ is nonsingular. Note that the nonsingularity assumption of $\partial_B H(w^*)$ is much weaker than that of $\partial H(w^*)$. For example, when $H : \Re \to \Re$ is given by $H(w) = |w|$, 0 does not belong to $\partial_B H(0) = \{-1, 1\}$ though it belongs to $\partial H(0) = [-1, 1]$.

The smoothing Newton method is a Newton-type method in which a smooth approximating function is used instead of the nonsmooth function. For a nonsmooth function $h : \Re^n \to \Re^m$, a function $h_\mu : \Re^n \to \Re^m$ with a parameter $\mu > 0$ is called a smoothing function, if h_μ is differentiable for any $\mu > 0$ and $\lim_{\mu \downarrow 0} h_\mu(x) = h(x)$ for any $x \in \Re^n$. Let φ_μ be a smoothing function of an SOCCP-function φ. Actually, concrete smoothing functions for the natural residual $\varphi_{\rm NR}$ and Fischer-Burmeister function $\varphi_{\rm FB}$ are proposed in [43]. Then, the parameterized function $H_\mu : \Re^{2n+\nu} \to \Re^{2n+\nu}$ defined by

$$H_\mu(w) := \begin{pmatrix} \varphi_\mu(x^1, y^1) \\ \vdots \\ \varphi_\mu(x^m, y^m) \\ F(x, y, \zeta) \end{pmatrix}$$

serves as a smoothing function of H defined by (2.4.2). In the smoothing Newton method, we solve a system of smoothed equations

$$H_\mu(w) = 0 \tag{2.4.6}$$

to obtain its solution w_μ, hoping that w_μ converges to a solution of the original problem (2.4.2) by letting $\mu \downarrow 0$. In order to solve the smoothed subproblem (2.4.6), it is natural to adopt the ordinary Newton iteration

$$d^k = -\nabla H_\mu(w^k)^{-T} H_\mu(w^k).$$

It is also possible to use the modified Newton direction

$$d^k = -\nabla H_\mu(w^k)^{-T} H(w^k), \tag{2.4.7}$$

where the smoothing function appears only in the Jacobian. The latter method may generate a sequence that eventually converges to a solution of the original problem (2.4.2) as $\mu \downarrow 0$. It is generally more difficult to discuss the convergence property of the smoothing Newton method than the generalized Newton method. In practice, the convergence behavior largely depends on the employed smoothing function.

Chapter 3

A combined smoothing and regularization method for monotone second-order cone complementarity problems

3.1 Introduction

In this chapter, we focus on the special SOCCP : Find $(x, y) \in \Re^n \times \Re^n$ such that

$$x \in \mathcal{K},\ y \in \mathcal{K},\ x^T y = 0,\ y = f(x), \tag{3.1.1}$$

where f is a continuously differentiable mapping from $\Re^n$ to $\Re^n$. This SOCCP may seem rather restrictive. However, the KKT conditions for any SOCP with continuously differentiable functions can be written in the form of SOCCP (3.1.1). For more detail, see Section 2.6.

Recently, a number of methods for solving SOCCP have been proposed. For SOCP including affine functions, primal-dual interior-point methods [76, 82, 112] are shown to be effective. For SOCCP, Fukushima, Luo and Tseng [43], Chen, Sun and Sun [19], and Chen, Chen and Tseng [15] studied smoothing and nonsmooth approaches. Especially, Fukushima, Luo and Tseng [43] showed that the min function and the Fischer-Burmeister function for the NCP can be extended to the SOCCP by means of Jordan algebra. Furthermore, they constructed smoothing functions for those functions and analyzed the properties of their Jacobians.

In this chapter, we introduce not only smoothing methods but also regularization methods for SOCCPs. Smoothing methods have effectively been adopted to deal with nondifferentiable reformulations of complementarity problems [13, 16, 17, 69, 95, 97, 101, 119]. On the other hand, regularization methods have provided a fundamental tool to deal with ill-posed problems [4, 30, 32, 95, 105]. By combining these methods, we develop a hybrid algorithm for solving monotone SOCCPs. Moreover, we show conditions for the algorithm to be globally and quadratically convergent. Particularly, we highlight the two properties called *strong semismoothness* and *Jacobian consistency*, which play a crucial role in establishing quadratic convergence of the algorithm.

This chapter is organized as follows. In Section 3.2, we construct a merit function by means of the natural residual for the SOCCP. In Section 3.3, we introduce smoothing and regularization methods, which smoothen the natural residual and weaken the condition for the merit function to be level-bounded. In Section 3.4, we propose a prototype algorithm for solving the SOCCP and show that it has global convergence when applied to monotone SOCCPs. Moreover, by incorporating Newton's method, we construct a concrete algorithm and establish quadratic convergence. In Section 3.5, we present some numerical results with the latter algorithm. In Section 3.6, we conclude this chapter with some remarks.

3.2 Merit function

We consider the unconstrained optimization reformulation of SOCCP (3.1.1) :

$$\text{Minimize } \Psi(x,y),$$

where Ψ is a real-valued function on $\Re^n \times \Re^n$. The objective function Ψ is called a merit function for SOCCP (3.1.1). In order to construct a merit function for SOCCP (3.1.1), it is convenient to introduce a function $\Phi : \Re^n \times \Re^n \to \Re^n$ satisfying

$$\Phi(x,y) = 0 \iff x \in \mathcal{K},\ y \in \mathcal{K},\ x^T y = 0. \tag{3.2.1}$$

By using such a function, we define $H : \Re^n \times \Re^n \to \Re^{2n}$ by

$$H(x,y) := \begin{pmatrix} \Phi(x,y) \\ f(x) - y \end{pmatrix}.$$

It is obvious that SOCCP (3.1.1) is equivalent to the equation $H(x,y) = 0$. Moreover, we define function $\Psi : \Re^n \times \Re^n \to \Re$ by

$$\Psi(x,y) := \frac{1}{2}\|H(x,y)\|^2 = \frac{1}{2}\|\Phi(x,y)\|^2 + \frac{1}{2}\|f(x) - y\|^2. \tag{3.2.2}$$

Then, it is easy to see that $\Psi(x,y) \geq 0$ for any $(x,y) \in \Re^n \times \Re^n$, and that $\Psi(x,y) = 0$ if and only if (x,y) is a solution of SOCCP (3.1.1). Therefore, the function Ψ defined by (3.2.2) can serve as a merit function for SOCCP (3.1.1).

Note that the complementarity condition on $\mathcal{K} = \mathcal{K}^{n_1} \times \cdots \times \mathcal{K}^{n_m}$ can be decomposed into complementarity conditions on each $\mathcal{K}^{n_i}$, that is,

$$x \in \mathcal{K},\ y \in \mathcal{K},\ x^T y = 0 \iff x^i \in \mathcal{K}^{n_i},\ y^i \in \mathcal{K}^{n_i},\ (x^i)^T y^i = 0 \quad (i = 1, \ldots, m), \tag{3.2.3}$$

where $x = (x^1, \ldots, x^m) \in \Re^{n_1} \times \cdots \times \Re^{n_m}$ and $y = (y^1, \ldots, y^m) \in \Re^{n_1} \times \cdots \times \Re^{n_m}$. This fact naturally leads us to construct a function Φ satisfying (3.2.1) as

$$\Phi(x,y) := \begin{pmatrix} \varphi^1(x^1, y^1) \\ \vdots \\ \varphi^m(x^m, y^m) \end{pmatrix},$$

where $\varphi^i : \Re^{n_i} \times \Re^{n_i} \to \Re^{n_i}$ is a function satisfying

$$\varphi^i(x^i, y^i) = 0 \iff x^i \in \mathcal{K}^{n_i},\ y^i \in \mathcal{K}^{n_i}, (x^i)^T y^i = 0 \tag{3.2.4}$$

for each $i = 1, \ldots, m$. Fukushima, Luo and Tseng [43] showed that (3.2.4) holds for the *natural residual* function $\varphi^i_{\text{NR}} : \Re^{n_i} \times \Re^{n_i} \to \Re^{n_i}$ defined by

$$\varphi^i_{\text{NR}}(x^i, y^i) := x^i - P_{\mathcal{K}^{n_i}}(x^i - y^i). \tag{3.2.5}$$

Using this function, we define function $\Phi_{\text{NR}} : \Re^n \times \Re^n \to \Re^n$ by

$$\Phi_{\text{NR}}(x, y) := \begin{pmatrix} \varphi^1_{\text{NR}}(x^1, y^1) \\ \vdots \\ \varphi^m_{\text{NR}}(x^m, y^m) \end{pmatrix}$$

and $H_{\text{NR}} : \Re^n \times \Re^n \to \Re^{2n}$ by

$$H_{\text{NR}}(x, y) := \begin{pmatrix} \Phi_{\text{NR}}(x, y) \\ f(x) - y \end{pmatrix}.$$

Then, we can construct a merit function $\Psi_{\text{NR}} : \Re^n \times \Re^n \to \Re$ for SOCCP (3.1.1) as

$$\Psi_{\text{NR}}(x, y) := \frac{1}{2}\|H_{\text{NR}}(x, y)\|^2 = \frac{1}{2}\sum_{i=1}^{m} \|\varphi^i_{\text{NR}}(x^i, y^i)\|^2 + \frac{1}{2}\|f(x) - y\|^2.$$

In what follows, we write φ_{NR} for φ^i_{NR} for simplicity of notation.

3.3 Smoothing and regularization

In the previous section, we have constructed the merit function Ψ_{NR} from the natural residual φ_{NR}. We can solve SOCCP (3.1.1) by minimizing Ψ_{NR} by an appropriate descent algorithm. However, the function Ψ_{NR} is not differentiable, and hence, methods that use the gradient of the function, such as the steepest descent method and Newton's method, are not applicable. In order to get rid of this difficulty, we first introduce a smoothing method that solves a sequence of differentiable approximations to the original nondifferentiable problem. To ensure global convergence of a descent method, the level-boundedness of the objective function plays an important role. If the function involved in the SOCCP is strongly monotone, then the merit function Ψ_{NR} is level-bounded. (See Appendix A.) But the assumption of strong monotonicity is quite restrictive from a practical standpoint. To be amenable to a merely monotone problem, we propose to combine a regularization method with a smoothing method.

In the remainder of this chapter, we assume $\mathcal{K} = \mathcal{K}^n$. Then we can rewrite SOCCP (3.1.1) as follows: Find $(x, y) \in \Re^n \times \Re^n$ such that

$$x \in \mathcal{K}^n,\ y \in \mathcal{K}^n,\ x^T y = 0,\ y = f(x). \tag{3.3.1}$$

The assumption $\mathcal{K} = \mathcal{K}^n$ is only for simplicity of presentation. In view of (3.2.3), the results obtained in the following can be extended to the general $\mathcal{K}$ in a straightforward manner (See Section 2.6). Under this assumption, $\Phi_{\text{NR}}(x, y)$ is reduced to $\varphi_{\text{NR}}(x, y)$, and hence, the vector-valued function H_{NR} and the merit function Ψ_{NR} are respectively rewritten as

$$\begin{aligned} H_{\text{NR}}(x, y) &= \begin{pmatrix} \varphi_{\text{NR}}(x, y) \\ f(x) - y \end{pmatrix}, \\ \Psi_{\text{NR}}(x, y) &= \frac{1}{2}\|H_{\text{NR}}(x, y)\|^2 = \frac{1}{2}\|\varphi_{\text{NR}}(x, y)\|^2 + \frac{1}{2}\|f(x) - y\|^2. \end{aligned} \tag{3.3.2}$$

3.3.1 Smoothing functions

In this subsection, we introduce a class of smoothing functions of the merit function Ψ_{NR}. For a nondifferentiable function $h : \Re^n \to \Re^m$, we consider a function $h_\mu : \Re^n \to \Re^m$ with a parameter $\mu > 0$ that has the following properties:

(a) h_μ is differentiable for any $\mu > 0$,

(b) $\lim_{\mu \downarrow 0} h_\mu(x) = h(x)$ for any $x \in \Re^n$.

Such a function h_μ is called a *smoothing function* of h. Instead of handling the nonsmooth equation $h(x) = 0$ directly, the smoothing method solves a family of smoothed subproblems $h_\mu(x) = 0$ for $\mu > 0$, and obtain a solution of the original problem by letting $\mu \downarrow 0$. Fukushima, Luo and Tseng [43] extended Chen and Mangasarian's class [13] of smoothing functions for NCP to SOCCP, which may be regarded as a smoothing function of the natural residual φ_{NR}.

First we define a smoothing function of the projection function $P_{\mathcal{K}^n}$ defined by (2.2.7). To this end, we consider a continuously differentiable convex function $\hat{g} : \Re \to \Re$ such that

$$\lim_{\alpha \to -\infty} \hat{g}(\alpha) = 0, \quad \lim_{\alpha \to \infty} (\hat{g}(\alpha) - \alpha) = 0, \quad 0 < \hat{g}'(\alpha) < 1. \tag{3.3.3}$$

For example, $\hat{g}_1(\alpha) = (\sqrt{\alpha^2 + 4} + \alpha)/2$ and $\hat{g}_2(\alpha) = \ln(e^\alpha + 1)$ satisfy (3.3.3). By using $\hat{g}$, we define function $P_\mu : \Re^n \to \Re^n$ by

$$P_\mu(z) := \mu\hat{g}(\lambda_1/\mu)u^{\{1\}} + \mu\hat{g}(\lambda_2/\mu)u^{\{2\}}, \tag{3.3.4}$$

where λ_1 and λ_2 are the spectral values of z given by (2.2.4), and $u^{\{1\}}$ and $u^{\{2\}}$ are the spectral vectors of z given by (2.2.5). Fukushima, Luo and Tseng [43] showed that the function P_μ defined by (3.3.4) is a smoothing function of $P_{\mathcal{K}^n}$, by using the facts that $\lim_{\mu \downarrow 0} \mu\hat{g}(\lambda/\mu) = \max\{0, \lambda\}$ and that $\gamma_\mu(\lambda) := \mu\hat{g}(\lambda/\mu)$ is differentiable for any $\mu > 0$. Hence, from the definition (3.2.5) of φ_{NR}, the function $\varphi_\mu : \Re^n \to \Re^n$ defined by

$$\varphi_\mu(x, y) := x - P_\mu(x - y)$$

becomes a smoothing function of φ_{NR}. In particular, by [43, Proposition 5.1], there exists a positive constant ν such that

$$\|\varphi_\mu(x, y) - \varphi_{\text{NR}}(x, y)\| \leq \nu\mu \tag{3.3.5}$$

for any $\mu > 0$ and $(x, y) \in \Re^n \times \Re^n$. Likewise, function $\Psi_\mu : \Re^n \times \Re^n \to \Re$ given by $\Psi_\mu(x, y) := (1/2)\,\|\varphi_\mu(x, y)\|^2 + (1/2)\|f(x) - y\|^2$ serves as a smoothing function of the merit function Ψ_{NR}. In the following, we denote $P_0(z) := P_{\mathcal{K}^n}(z)$, $\varphi_0(x, y) := \varphi_{\text{NR}}(x, y)$ and $\Psi_0(x, y) := \Psi_{\text{NR}}(x, y)$.

3.3.2 Regularization method

Since Ψ_μ is differentiable, we may apply an appropriate gradient-based descent method to obtain a minimum (x_μ, y_μ) of the function Ψ_μ for $\mu > 0$. In order that a sequence generated by such a descent method has an accumulation point, it is desirable that Ψ_μ is level-bounded, that is, the level sets $\mathcal{L}_\alpha := \{(x, y) \mid \Psi_\mu(x, y) \le \alpha\}$ are bounded for all $\alpha \in \Re$. Actually, by using (3.3.5) and the results for natural residual of variational inequality problems [93, 121], we can show that Ψ_μ is level-bounded for any $\mu \ge 0$ if f is strongly monotone. (See Appendix A.) However, the strong monotonicity is quite a severe condition. As a remedy for this inconvenience, we employ a regularization method.

Let the function $f_\varepsilon : \Re^n \to \Re^n$ be defined by $f_\varepsilon(x) := f(x) + \varepsilon x$ with a positive parameter ε. The regularization method solves a sequence of SOCCPs involving f_ε with $\varepsilon > 0$, so that a solution of the original SOCCP is obtained by taking the limit $\varepsilon \downarrow 0$. Define functions $H_{\mu,\varepsilon} : \Re^n \times \Re^n \to \Re^{2n}$ and $\Psi_{\mu,\varepsilon} : \Re^n \times \Re^n \to \Re$ by

$$H_{\mu,\varepsilon}(x, y) := \begin{pmatrix} \varphi_\mu(x, y) \\ f_\varepsilon(x) - y \end{pmatrix}, \tag{3.3.6}$$

$$\Psi_{\mu,\varepsilon}(x, y) := \frac{1}{2}\|H_{\mu,\varepsilon}(x, y)\|^2 = \frac{1}{2}\|\varphi_\mu(x, y)\|^2 + \frac{1}{2}\|f_\varepsilon(x) - y\|^2. \tag{3.3.7}$$

If f is monotone, then f_ε is strongly monotone for any $\varepsilon > 0$, and hence, the function $\Psi_{\mu,\varepsilon}$ is level-bounded for any $\mu \ge 0$ and $\varepsilon > 0$.

Finally, we give explicit expressions of functions $\nabla H_{\mu,\varepsilon}$ and ∇P_μ, which will be useful for the subsequent analysis.

Remark 3.3.1 *From the definition of $H_{\mu,\varepsilon}$, φ_μ and f_ε, $\nabla H_{\mu,\varepsilon}(x, y)$ can be written as*

$$\nabla H_{\mu,\varepsilon}(x, y) = \begin{pmatrix} I - \nabla P_\mu(x - y) & \nabla f(x) + \varepsilon I \\ \nabla P_\mu(x - y) & -I \end{pmatrix}. \tag{3.3.8}$$

Let $g : \Re^n \to \Re^n$ be defined by $g(z) := \hat{g}(\lambda_1)u^{\{1\}} + \hat{g}(\lambda_2)u^{\{2\}}$, where λ_1 and λ_2 are the spectral values of z, and $u^{\{1\}}$ and $u^{\{2\}}$ are the spectral vectors of z. Then we have $\nabla P_\mu(z) = \nabla g(z/\mu)$ since $P_\mu(z) = \mu g(z/\mu)$ from (3.3.4)*. Therefore, by* [43, Proposition 5.2]*, $\nabla P_\mu(z)$ is written as*

$$\nabla P_\mu(z) = \begin{cases} \hat{g}'(z_1/\mu)I & \text{if } z_2 = 0, \\ \begin{pmatrix} b_\mu & \dfrac{c_\mu z_2^T}{\|z_2\|} \\ \dfrac{c_\mu z_2}{\|z_2\|} & a_\mu I + (b_\mu - a_\mu)\dfrac{z_2 z_2^T}{\|z_2\|^2} \end{pmatrix} & \text{if } z_2 \ne 0, \end{cases} \tag{3.3.9}$$

where

$$\begin{cases} a_\mu = \dfrac{\hat{g}(\lambda_2/\mu) - \hat{g}(\lambda_1/\mu)}{\lambda_2/\mu - \lambda_1/\mu}, \\ b_\mu = \dfrac{1}{2}(\hat{g}'(\lambda_2/\mu) + \hat{g}'(\lambda_1/\mu)), \\ c_\mu = \dfrac{1}{2}(\hat{g}'(\lambda_2/\mu) - \hat{g}'(\lambda_1/\mu)). \end{cases} \tag{3.3.10}$$

3.4 Algorithm

In this section, we present a globally and quadratically convergent algorithm for solving SOCCP (3.3.1). We first give a globally convergent prototype algorithm that utilizes smoothing and regularization techniques. Next, we study two properties called strong semismoothness and Jacobian consistency, which play an important role in establishing quadratic convergence of the algorithm. Finally, we construct a quadratically convergent algorithm by incorporating Newton's method to the prototype algorithm.

3.4.1 Prototype algorithm

As we have stated in the previous section, for any $\mu > 0$ and $\varepsilon > 0$, the function $\Psi_{\mu,\varepsilon}$ defined by (3.3.7) is differentiable and level-bounded, provided f is monotone. Therefore, by applying an appropriate descent method, we may obtain a minimum $(x_{\mu,\varepsilon}, y_{\mu,\varepsilon})$ of the function $\Psi_{\mu,\varepsilon}$. Moreover, letting (μ, ε) converge to $(0, 0)$, we can expect that $(x_{\mu,\varepsilon}, y_{\mu,\varepsilon})$ converges to a solution of the original SOCCP. However, in practice, it is usually impossible to compute an exact minimum of $\Psi_{\mu,\varepsilon}$. So, we consider the following algorithm in which the function $\Psi_{\mu,\varepsilon}$ is minimized only approximately at each iteration.

Algorithm 3.4.1

Step 0 *Choose* $(x^{(0)}, y^{(0)}) \in \Re^n \times \Re^n$, $\mu_0 \in (0, \infty)$, $\varepsilon_0 \in (0, \infty)$ *and* $\alpha_0 \in (0, \infty)$. *Set* $k := 0$.

Step 1 *Terminate the iteration if* $\Psi_{\mathrm{NR}}(x^{(k)}, y^{(k)}) = 0$.

Step 2 *Find a pair* $(x^{(k+1)}, y^{(k+1)}) \in \Re^n \times \Re^n$ *such that*

$$\Psi_{\mu_k, \varepsilon_k}(x^{(k+1)}, y^{(k+1)}) \leq \alpha_k. \tag{3.4.1}$$

Step 3 *Update parameters* $\mu_{k+1} \in (0, \mu_k)$, $\varepsilon_{k+1} \in (0, \varepsilon_k)$ *and* $\alpha_{k+1} \in (0, \alpha_k)$ *so that they converge to* 0 *eventually. Set* $k := k + 1$. *Go back to Step* 1.

To obtain $(x^{(k+1)}, y^{(k+1)})$ in Step 2, we may use any suitable unconstrained minimization technique. These issues will be discussed in detail in Subsection 3.4.4.

In order for Algorithm 3.4.1 to be well-defined, there must exist a pair $(x^{(k+1)}, y^{(k+1)})$ satisfying (3.4.1) for any $\alpha_k > 0$. For this purpose, it suffices to show that any stationary point of $\Psi_{\mu,\varepsilon}$ is a global minimum of $\Psi_{\mu,\varepsilon}$.

Proposition 3.4.1 *If $f : \Re^n \to \Re^n$ is monotone, then, for any $\mu > 0$ and $\varepsilon \geq 0$, every stationary point $(\overline{x}, \overline{y})$ of the function $\Psi_{\mu,\varepsilon}$ satisfies $\Psi_{\mu,\varepsilon}(\overline{x}, \overline{y}) = 0$.*

Proof. Note that $\nabla\Psi_{\mu,\varepsilon}(\overline{x}, \overline{y}) = \nabla H_{\mu,\varepsilon}(\overline{x}, \overline{y})\, H_{\mu,\varepsilon}(\overline{x}, \overline{y}) = 0$. By Proposition 6.1 of [43], $\nabla H_{\mu,\varepsilon}(\overline{x}, \overline{y})$ is nonsingular. Hence, we have $H_{\mu,\varepsilon}(\overline{x}, \overline{y}) = 0$, that is, $\Psi_{\mu,\varepsilon}(\overline{x}, \overline{y}) = (1/2)\|H_{\mu,\varepsilon}(\overline{x}, \overline{y})\|^2 = 0$. ∎

We proceed to showing the global convergence property of Algorithm 3.4.1. To this end, we introduce the weak univalence property. We say function $H : \mathcal{D} \subseteq \Re^n \to \Re^n$ to be weakly univalent if it is continuous and there exists a sequence of continuous and injective functions $\{H_k\}$ converging to H uniformly on a bounded subset of $\mathcal{D}$. In particular, if f is monotone, then $H_{\rm NR}$ is weakly univalent from the following reasons: For any $\mu > 0$ and $\varepsilon \geq 0$, $\nabla H_{\mu,\varepsilon}(x, y)$ is nonsingular for any $(x, y) \in \Re^n \times \Re^n$ [43, Proposition 6.1], and hence, $H_{\mu,\varepsilon}$ is injective. Moreover, $H_{\mu,\varepsilon}$ converges to $H_{\rm NR}$ uniformly on a bounded set as $(\mu, \varepsilon) \downarrow (0, 0)$ since we have from (3.3.5)

$$\|H_{\mu,\varepsilon}(x, y) - H_{\rm NR}(x, y)\| \leq \|\varphi_\mu(x, y) - \varphi_{\rm NR}(x, y)\| + \|f_\varepsilon(x) - f(x)\| \leq \nu\mu + \varepsilon\|x\|. \quad (3.4.2)$$

The next lemma indicates a property which weakly univalent functions possess.

Lemma 3.4.1 [33, Corollary 3.6.5] *Let $H : \Re^n \to \Re^n$ be a weakly univalent function such that the inverse image $H^{-1}(0)$ is nonempty and compact. Then, for any $\varepsilon > 0$, there exists $\delta > 0$ such that the following statement holds: For every weakly univalent function $\tilde{H} : \Re^n \to \Re^n$ such that $\|H(z) - \tilde{H}(z)\| \leq \delta$ for any $z \in \mathrm{cl}(H^{-1}(0) + B(0, \varepsilon))$, we have $\emptyset \neq \tilde{H}^{-1}(0) \subseteq H^{-1}(0) + B(0, \varepsilon)$, and $\tilde{H}^{-1}(0)$ is connected.*

By using this lemma, we establish the global convergence of Algorithm 3.4.1.

Theorem 3.4.1 *Let $f : \Re^n \to \Re^n$ be a monotone function. Assume that the solution set $\mathcal{S}$ of SOCCP (3.3.1) is nonempty and bounded. Let $\{(x^{(k)}, y^{(k)})\}$ be a sequence generated by Algorithm 3.4.1. Then, $\{(x^{(k)}, y^{(k)})\}$ is bounded, and every accumulation point is a solution of SOCCP (3.3.1).*

Proof. From a simple continuity argument, we can easily show that every accumulation point of $\{(x^{(k)}, y^{(k)})\}$ is a solution of SOCCP (3.3.1). So we only show the boundedness of $\{(x^{(k)}, y^{(k)})\}$. For any $\varepsilon > 0$, let $\Omega := \mathrm{cl}(H_{\rm NR}^{-1}(0) + B(0, \varepsilon))$, which is nonempty and compact by the given assumption. Then, there exists $\delta > 0$ such that Lemma 3.4.1 holds for $H = H_{\rm NR}$. Let continuous and injective functions $G_k : \Re^n \times \Re^n \to \Re^{2n}$ be defined by $G_k(x, y) := H_{\mu_{k-1},\varepsilon_{k-1}}(x, y) - H_{\mu_{k-1},\varepsilon_{k-1}}(x^{(k)}, y^{(k)})$. From (3.4.1) and (3.4.2), there exists $\overline{k}$ such that $\|G_k(x, y) - H_{\rm NR}(x, y)\| \leq \|H_{\mu_{k-1},\varepsilon_{k-1}}(x, y) - H_{\rm NR}(x, y)\| + \|H_{\mu_{k-1},\varepsilon_{k-1}}(x^{(k)}, y^{(k)})\| \leq \nu\mu_{k-1} + \varepsilon_{k-1}\|x\| + \sqrt{2\alpha_{k-1}} \leq \delta$ for any $k \geq \overline{k}$ and $(x, y) \in \Omega$. Moreover, G_k is a weakly univalent function. Hence, by Lemma 3.4.1 with $\tilde{H} = G_k$ and $H = H_{\rm NR}$, we have $G_k^{-1}(0) \subseteq \Omega$ for all $k \geq \overline{k}$. This together with $(x^{(k)}, y^{(k)}) \in G_k^{-1}(0)$ implies the boundedness of $\{(x^{(k)}, y^{(k)})\}$. ∎

An alternative proof of this theorem is given in Appendix B, in which the Mountain Pass Theorem is used instead of the weak univalence property. Although the proof in the appendix is more comprehensible, the above proof is more elegant and extendable to other cases.

3.4.2 Semismoothness and strong semismoothness

Semismoothness is a generalized concept of the smoothness, which was originally introduced by Mifflin [80] for functionals and extended to vector-valued functions by Qi and Sun [98]. Strong semismoothness is also a generalized concept of the smoothness, which is stronger than the semismoothness. These concepts play an important role in establishing fast local convergence of the algorithm presented later. In this subsection, we first show that the function $P_{\mathcal{K}^n}$ is strongly semismooth at an arbitrary point. Then, by using this result, we show the strong semismoothness of the function H_{NR} defined by (3.3.2).

Let $H : \Re^n \to \Re^m$ be a locally Lipschitzian function. Then H is differentiable almost everywhere by Rademacher's Theorem [20]. Let $D_H \subseteq \Re^n$ be the set of points where H is differentiable. Then, Clarke subdifferential $\partial H(x)$ of H at x is defined by $\partial H(x) := \text{co}\,\{\lim_{\hat{x}\to x,\, \hat{x}\in D_H} \nabla H(\hat{x})\}$ [20, 96]. We note that $\partial H(x) = \{\nabla H(x)\}$ if H is continuously differentiable at x. By using the concept of subdifferential, we give the definitions of semismoothness and strong semismoothness.

Definition 3.4.1 *A directionally differentiable and locally Lipschitzian function $H : \Re^n \to \Re^m$ is said to be semismooth at z if $V^T d - H'(z\,;d) = o(\|d\|)$ for any $d \in \Re^n\backslash\{0\}$ sufficiently small and $V \in \partial H(z+d)$, where $H'(z;d) := \lim_{h\downarrow 0}(H(z+hd) - H(z))/h$ is the directional derivative of H at z along the direction d. In particular, if $o(\|d\|)$ can be replaced by $O(\|d\|^2)$, then function H is said to be strongly semismooth.*

It is known that any piecewise smooth function[1] is semismooth. Moreover, any piecewise smooth function is strongly semismooth if all the derivatives of smooth pieces are locally Lipschitzian. Next we show that the projection function $P_{\mathcal{K}^n}$ is strongly semismooth everywhere.

Proposition 3.4.2 *The projection function $P_{\mathcal{K}^n}$ is strongly semismooth at any $z \in \Re^n$.*

Proof. Let λ_1 and λ_2 be the spectral values of z. Since we have from (2.2.8)

$$P_{\mathcal{K}^n}(z) = \begin{cases} z & (\lambda_1 \geq 0,\, \lambda_2 \geq 0) \\ \dfrac{1}{2}(z_1 + \|z_2\|)\left(1, \dfrac{z_2}{\|z_2\|}\right) & (\lambda_1 < 0,\, \lambda_2 > 0) \\ 0 & (\lambda_1 \leq 0,\, \lambda_2 \leq 0)\,, \end{cases}$$

$P_{\mathcal{K}^n}$ is continuously differentiable at any z such that $\lambda_1 \neq 0$ and $\lambda_2 \neq 0$, and its derivative is locally Lipschitzian. Moreover, $P_{\mathcal{K}^n}$ is piecewise smooth at any z such that $\lambda_2 > \lambda_1 = 0$ or $\lambda_1 < \lambda_2 = 0$. Hence, we have only to show the strong semismoothness at the origin. Note that $P_{\mathcal{K}^n}(tz) = tP_{\mathcal{K}^n}(z)$ for any $t \geq 0$. Since any locally Lipschitzian positively homogeneous function is strongly semismooth at the origin [103], $P_{\mathcal{K}^n}$ is strongly semismooth at the origin. ■

In Appendix C, we give a more detailed proof, in which we divide the whole space into six subsets and discuss the strong semismoothness in each subset. Recently, Chen, Sun and Sun [19] also showed

[1] Function $H : \Re^n \to \Re^m$ is said to be piecewise smooth at z if there exist $\varepsilon > 0$ and a finite number of continuously differentiable functions $H_i : B(z,\varepsilon) \to \Re^m$ $(i = 1, \ldots, N)$ such that for any $z' \in B(z,\varepsilon)$ there exists an index i such that $H(z') = H_i(z')$.

the strong semismoothness of $P_{\mathcal{K}^n}$, and Chen, Chen and Tseng [15] gave more general results. We note, however, that their approaches are quite different from ours. Now, we are ready to prove the strong semismoothness of H_{NR}.

Theorem 3.4.2 *The function H_{NR} defined by (3.3.2) is semismooth at every point $(x, y) \in \Re^n \times \Re^n$. Moreover, if $\nabla f : \Re^n \to \Re^{n\times n}$ is locally Lipschitzian, then H_{NR} is strongly semismooth at every point $(x, y) \in \Re^n \times \Re^n$.*

Proof. Let $(\xi, \eta) \in \Re^n \times \Re^n$ be a pair of vectors sufficiently small, and U be an arbitrary element of $\partial H_{\text{NR}}(x + \xi, y + \eta)$. Note that $P_{\mathcal{K}^n}$ is strongly semismooth at $x - y$. Then, since

$$\partial H_{\text{NR}}(x+\xi, y+\eta) = \left\{ \begin{pmatrix} I - V & \nabla f(x+\xi) \\ V & -I \end{pmatrix} \middle| V \in \partial P_{\mathcal{K}^n}((x-y)+(\xi-\eta)) \right\}$$

and

$$H'_{\text{NR}}((x,y);(\xi,\eta)) = \begin{pmatrix} \xi - P'_{\mathcal{K}^n}(x-y;\xi-\eta) \\ \nabla f(x)^T\xi - \eta \end{pmatrix},$$

we have

$$\begin{aligned} U^T\begin{pmatrix}\xi\\ \eta\end{pmatrix} - H'_{\text{NR}}((x,y);(\xi,\eta)) &= \begin{pmatrix} I - V & \nabla f(x) \\ V & -I \end{pmatrix}^T \begin{pmatrix}\xi\\ \eta\end{pmatrix} - \begin{pmatrix} \xi - P'_{\mathcal{K}^n}(x-y;\xi-\eta) \\ \nabla f(x)^T\xi - \eta \end{pmatrix} \\ &= \begin{pmatrix} P'_{\mathcal{K}^n}(x-y;\xi-\eta) - V^T(\xi-\eta) \\ (\nabla f(x+\xi) - \nabla f(x))^T\xi \end{pmatrix} \\ &= \begin{pmatrix} O(\|\xi-\eta\|^2) \\ o(\|\xi\|) \end{pmatrix} \\ &= o(\|(\xi,\eta)\|), \end{aligned} \tag{3.4.3}$$

where the last equality follows from $\|\xi - \eta\|^2 \le 2\|\xi\|^2 + 2\|\eta\|^2 = 2\|(\xi, \eta)\|^2$ and $\|\xi\| \le \|(\xi, \eta)\|$. Hence, H_{NR} is semismooth at (x, y). Furthermore, if ∇f is locally Lipschitzian, then the strong semismoothness of H_{NR} at (x, y) readily follows since $o(\|\xi\|)$ in (3.4.3) can be replaced by $O(\|\xi\|^2)$. ■

3.4.3 Jacobian consistency of the smoothing function

Jacobian consistency, which was first introduced by Chen, Qi and Sun [17], is a concept relating the generalized Jacobian of a nonsmooth function with Jacobian of a smoothing function. Like the strong semismoothness, the Jacobian consistency plays an important role in establishing rapid convergence of smoothing methods. In this subsection, we show that the function $H_{\mu,\varepsilon}$ defined by (3.3.6) enjoys the Jacobian consistency, and give some results that will be useful in constructing a rapidly convergent algorithm.

Definition 3.4.2 *Let $F : \Re^n \to \Re^n$ be a continuous function. Let $F_{\mu,\varepsilon} : \Re^n \to \Re^n$ be a function such that $F_{\mu,\varepsilon}$ is continuously differentiable for any $\mu > 0$ and $\varepsilon \geq 0$, and that $\lim_{(\mu,\varepsilon)\downarrow(0,0)} F_{\mu,\varepsilon}(z) = F(z)$ for any $z \in \Re^n$. Then we say $F_{\mu,\varepsilon}$ satisfies the Jacobian consistency if*

$$\lim_{(\mu,\varepsilon)\downarrow(0,0)} \mathrm{dist}(\nabla F_{\mu,\varepsilon}(z),\, \partial F(z)) \,=\, 0$$

holds for any $z \in \Re^n$. Here $\mathrm{dist}(X, S)$ denotes $\min\{\|X - Y\| \,|\, Y \in S\}$.

We note that Chen, Qi and Sun [17] define the Jacobian consistency in terms of $\partial_C F(z) := \partial F_1(z) \times \cdots \times \partial F_n(z) \supseteq \partial F(z)$ instead of $\partial F(z)$. Moreover, their definition contains only a smoothing parameter μ.

In order to show the Jacobian consistency of $H_{\mu,\varepsilon}$, we need the following proposition. Since it can be proved from (2.2.8) and (3.3.9) in a straightforward manner, we postpone the proof to Appendix D.

Proposition 3.4.3 *Clarke subdifferential of the projection function $P_{\mathcal{K}^n}$ and the matrix $J^o_P(z) := \lim_{\mu\downarrow 0} \nabla P_\mu(z)$ are given as follows:*

$$\partial P_{\mathcal{K}^n}(z) \,=\, \begin{cases} \{I\} & (\lambda_1 > 0, \lambda_2 > 0) \\ \{(\lambda_2/(\lambda_1+\lambda_2))I + Z\} & (\lambda_1 < 0, \lambda_2 > 0) \\ \{O\} & (\lambda_1 < 0, \lambda_2 < 0) \\ \mathrm{co}\{I, I+Z\} & (\lambda_1 = 0, \lambda_2 > 0) \\ \mathrm{co}\{O, Z\} & (\lambda_1 < 0, \lambda_2 = 0) \\ \mathrm{co}\,(\{O\} \cup \{I\} \cup \mathcal{S}) & (\lambda_1 = 0, \lambda_2 = 0)\,, \end{cases} \tag{3.4.4}$$

$$J^o_P(z) \,=\, \begin{cases} I & (\lambda_1 > 0, \lambda_2 > 0) \\ (\lambda_2/(\lambda_1+\lambda_2))I + Z & (\lambda_1 < 0, \lambda_2 > 0) \\ O & (\lambda_1 < 0, \lambda_2 < 0) \\ I + (1 - \hat{g}'(0))Z & (\lambda_1 = 0, \lambda_2 > 0) \\ \hat{g}'(0)Z & (\lambda_1 < 0, \lambda_2 = 0) \\ \hat{g}'(0)I & (\lambda_1 = 0, \lambda_2 = 0)\,, \end{cases} \tag{3.4.5}$$

where $(r_1, r_2) := (z_1, z_2)/\|z_2\|$,

$$Z = \frac{1}{2}\begin{pmatrix} -r_1 & r_2^T \\ r_2 & -r_1 r_2 r_2^T \end{pmatrix},\ \mathcal{S} := \left\{ \frac{1}{2}(1+\beta)I + \frac{1}{2}\begin{pmatrix} -\beta & w^T \\ w & -\beta w w^T \end{pmatrix} \middle| -1 \leq \beta \leq 1, \|w\| = 1 \right\}.$$

By using this proposition, we show the Jacobian consistency of $H_{\mu,\varepsilon}$.

Theorem 3.4.3 *$H_{\mu,\varepsilon}$ satisfies the Jacobian consistency.*

Proof. The formulas (3.4.4), (3.4.5) and $0 < \hat{g}'(0) < 1$ from (3.3.3) yield $J_P^o(z) \in \partial P_{\mathcal{K}^n}(z)$ for any $z \in \Re^n$. Moreover, we have

$$J_H^o(x,y) := \lim_{(\mu,\varepsilon)\downarrow(0,0)} \nabla H_{\mu,\varepsilon}(x,y) = \begin{pmatrix} I - J_P^o(x-y) & \nabla f(x) \\ J_P^o(x-y) & -I \end{pmatrix} \tag{3.4.6}$$

from (3.3.8), and

$$\partial H_{\mathrm{NR}}(x,y) = \left\{ \begin{pmatrix} I - D & \nabla f(x) \\ D & -I \end{pmatrix} \middle| \, D \in \partial P_{\mathcal{K}^n}(x-y) \right\}.$$

We thus have $J_H^o(x,y) \in \partial H_{\mathrm{NR}}(x,y)$ for any $(x,y) \in \Re^n \times \Re^n$, which implies that $H_{\mu,\varepsilon}$ satisfies the Jacobian consistency. ∎

The Jacobian consistency of $H_{\mu,\varepsilon}$ guarantees that, for any $(x,y) \in \Re^n \times \Re^n$ and $\delta' > 0$, there exist $\mu > 0$ and $\varepsilon > 0$ such that $\mathrm{dist}(\nabla H_{\mu,\varepsilon}(x,y), \partial H_{\mathrm{NR}}(x,y)) < \delta'$. To find such μ and ε for given δ' and (x,y) is an essential issue for constructing a rapidly convergent algorithm. In the remainder of this subsection, we study how we can find such μ and ε.

In what follows, we use the following notation for convenience. Let $\hat{g} : \Re \to \Re$ be a continuously differentiable convex function satisfying (3.3.3). Then we denote for any $\mu > 0$ that

$$\gamma_\mu(\alpha) := \mu \hat{g}(\alpha/\mu). \tag{3.4.7}$$

Note that, as $\mu \downarrow 0$, there exist limits of $\gamma_\mu(\alpha)$ and $\gamma'_\mu(\alpha)$ for any fixed α, which we denote

$$\gamma_0(\alpha) := \lim_{\mu\downarrow 0} \gamma_\mu(\alpha) = \max\{0, \alpha\}, \tag{3.4.8}$$

$$\gamma_0^+(\alpha) := \lim_{\mu\downarrow 0} \gamma'_\mu(\alpha) = \begin{cases} 0 & (\alpha < 0) \\ \hat{g}'(0) & (\alpha = 0) \\ 1 & (\alpha > 0). \end{cases} \tag{3.4.9}$$

Moreover, we suppose that the function $\hat{g}$ satisfies not only (3.3.3) but also

$$\hat{g}(\alpha) - \alpha = \hat{g}(-\alpha) \tag{3.4.10}$$

for any $\alpha \in \Re$. For example, $\hat{g}_1(\alpha) = (\sqrt{\alpha^2 + 4} + \alpha)/2$ and $\hat{g}_2(\alpha) = \ln(e^\alpha + 1)$ satisfy (3.3.3) and (3.4.10). For such functions, we have the following lemma.

Lemma 3.4.2 *Let $\hat{g}$ be a continuously differentiable convex function satisfying* (3.3.3) *and* (3.4.10). *Let γ_μ, γ_0 and γ_0^+ be defined by* (3.4.7)–(3.4.9). *Then it holds that* (a) $\gamma_\mu(\alpha) - \gamma_0(\alpha) = \gamma_\mu(-\alpha) - \gamma_0(-\alpha)$ *for any $\alpha \in \Re$, and* (b) $|\gamma'_\mu(0) - \gamma_0^+(0)| = 0 < |\gamma'_\mu(\alpha_2) - \gamma_0^+(\alpha_2)| \le |\gamma'_\mu(\alpha_1) - \gamma_0^+(\alpha_1)|$ *for any $(\alpha_1, \alpha_2) \in \Re \times \Re$ such that $0 < |\alpha_1| \le |\alpha_2|$.*

Proof. Since (a) can be easily seen from (3.4.7), (3.4.8) and (3.4.10), we only show (b). Let $(\alpha_1, \alpha_2) \in \Re \times \Re$ be arbitrary scalars such that $0 < |\alpha_1| \le |\alpha_2|$. From (3.4.10), we have $\hat{g}'(\alpha) - 1 =$

$-\hat{g}'(-\alpha)$ for any $\alpha \in \Re$. This, together with $\gamma'_\mu(\alpha) = \hat{g}'(\alpha/\mu)$ and (3.4.9), yields $\gamma'_\mu(\alpha) - \gamma_0^+(\alpha) = -\{\gamma'_\mu(-\alpha) - \gamma_0^+(-\alpha)\}$. We therefore have $|\gamma'_\mu(0) - \gamma_0^+(0)| = 0$ and

$$|\gamma'_\mu(\alpha_i) - \gamma_0^+(\alpha_i)| = |\gamma'_\mu(|\alpha_i|) - \gamma_0^+(|\alpha_i|)| \tag{3.4.11}$$

for each $i = 1, 2$. Now, we note that, for any $\beta > 0$, $|\gamma'_\mu(\beta) - \gamma_0^+(\beta)| = 1 - \gamma'_\mu(\beta) > 0$, where the inequality follows from $\gamma'_\mu(\beta) = \hat{g}'(\beta/\mu)$ and (3.3.3). Moreover, $1 - \gamma'_\mu(\beta)$ is monotonically nonincreasing since γ_μ is convex. Hence, we have $0 < |\gamma'_\mu(|\alpha_2|) - \gamma_0^+(|\alpha_2|)| \le |\gamma'_\mu(|\alpha_1|) - \gamma_0^+(|\alpha_1|)|$. Combining this inequality with (3.4.11), we have (b). ∎

We further define function $\tilde{\lambda} : \Re^n \to [0, +\infty)$ by

$$\tilde{\lambda}(z) := \begin{cases} \min\limits_{i \in \mathcal{I}(z)} |\lambda_i(z)| & (\mathcal{I}(z) \neq \emptyset) \\ 0 & (\mathcal{I}(z) = \emptyset), \end{cases} \tag{3.4.12}$$

where $\lambda_i(z)$ $(i = 1, 2)$ are the spectral values of z, and $\mathcal{I}(z) \subseteq \{1, 2\}$ is the index set defined by $\mathcal{I}(z) := \{i \,|\, \lambda_i(z) \neq 0\}$. Then we can easily see from Lemma 3.4.2 (b) that

$$|\gamma'_\mu(\lambda_i(z)) - \gamma_0^+(\lambda_i(z))| \le |\gamma'_\mu(\tilde{\lambda}(z)) - \gamma_0^+(\tilde{\lambda}(z))| \quad (i = 1, 2). \tag{3.4.13}$$

From this fact, we can estimate the upper bound of $\mathrm{dist}(\nabla H_{\mu,\varepsilon}(x, y), \partial H_{\mathrm{NR}}(x, y))$.

Proposition 3.4.4 *Let $\hat{g}$ be a function satisfying* (3.3.3) *and* (3.4.10)*, and $\tilde{\lambda}$ be the function defined by* (3.4.12)*. Then, there exists $M > 0$ such that*

$$\mathrm{dist}(\nabla H_{\mu,\varepsilon}(x, y), \partial H_{\mathrm{NR}}(x, y)) \le M(|\gamma'_\mu(\tilde{\lambda}(x - y)) - \gamma_0^+(\tilde{\lambda}(x - y))| + \varepsilon)$$

for any $\mu > 0$, $\varepsilon \ge 0$ and $(x, y) \in \Re^n \times \Re^n$.

Proof. By (3.3.8), (3.4.6) and the Jacobian consistency of $H_{\mu,\varepsilon}$, there exists $M' > 0$ such that $\mathrm{dist}(\nabla H_{\mu,\varepsilon}(x, y), \partial H_{\mathrm{NR}}(x, y)) \le M'(\|\nabla P_\mu(x - y) - J_P^o(x - y)\| + \varepsilon)$ for any $\mu > 0$, $\varepsilon \ge 0$ and $(x, y) \in \Re^n \times \Re^n$. So it suffices to show the existence of $N > 0$ such that

$$\|\nabla P_\mu(z) - J_P^o(z)\| \le N|\gamma'_\mu(\tilde{\lambda}(z)) - \gamma_0^+(\tilde{\lambda}(z))|$$

for any $\mu > 0$ and $z \in \Re^n$. Let λ_1 and λ_2 be the spectral values of z, and a_μ, b_μ and c_μ be defined by (3.3.10). Moreover, let

$$a_0 := \lim_{\mu \downarrow 0} a_\mu = \frac{\gamma_0(\lambda_2) - \gamma_0(\lambda_1)}{\lambda_2 - \lambda_1}, \tag{3.4.14}$$

$$b_0 := \lim_{\mu \downarrow 0} b_\mu = \frac{1}{2}(\gamma_0^+(\lambda_2) + \gamma_0^+(\lambda_1)), \tag{3.4.15}$$

$$c_0 := \lim_{\mu \downarrow 0} c_\mu = \frac{1}{2}(\gamma_0^+(\lambda_2) - \gamma_0^+(\lambda_1)). \tag{3.4.16}$$

When $z_2 = 0$, that is, $z_1 = \lambda_1 = \lambda_2$, we have $\|\nabla P_\mu(z) - J_P^o(z)\| = \|g'(z_1/\mu)I - \gamma_0^+(z_1)I\| = |\gamma'_\mu(z_1) - \gamma_0^+(z_1)| = |\gamma'_\mu(\tilde{\lambda}(z)) - \gamma_0^+(\tilde{\lambda}(z))|$, where the first equality follows from (3.3.9), (3.4.5) and (3.4.9), and the last equality follows from $\tilde{\lambda}(z) = |z_1|$ and (3.4.11).

When $z_2 \neq 0$, that is, $\lambda_1 < \lambda_2$, by (3.3.9) and (3.4.5), we have for some $N' > 0$

$$\begin{aligned}
&\|\nabla P_\mu(z) - J_P^o(z)\| \\
&= \left\| (a_\mu - a_0)I + \begin{pmatrix} (b_\mu - b_0) - (a_\mu - a_0) & (c_\mu - c_0)r_2^T \\ (c_\mu - c_0)r_2 & ((b_\mu - b_0) - (a_\mu - a_0))r_2 r_2^T \end{pmatrix} \right\| \\
&\leq N'(|a_\mu - a_0| + |b_\mu - b_0| + |c_\mu - c_0|),
\end{aligned} \tag{3.4.17}$$

where $r_2 := z_2/\|z_2\|$. First we consider the case where $0 \leq \lambda_1 < \lambda_2$. Note that there exists $\overline{\lambda} \in [\lambda_1, \lambda_2]$ such that $\gamma'_\mu(\overline{\lambda}) = (\gamma_\mu(\lambda_2) - \gamma_\mu(\lambda_1))/(\lambda_2 - \lambda_1) = a_\mu$. Moreover, we have from (3.4.9), (3.4.14) and $\overline{\lambda} \in [\lambda_1, \lambda_2]$ that $a_0 = 1 = \gamma_0^+(\overline{\lambda})$. Hence, we obtain $|a_\mu - a_0| = |\gamma'_\mu(\overline{\lambda}) - \gamma_0^+(\overline{\lambda})|$. Furthermore, by (3.3.10) and (3.4.15), we have $|b_\mu - b_0| \leq (1/2)|\gamma'_\mu(\lambda_2) - \gamma_0^+(\lambda_2)| + (1/2)|\gamma'_\mu(\lambda_1) - \gamma_0^+(\lambda_1)|$, and, by (3.3.10) and (3.4.16), we have $|c_\mu - c_0| \leq (1/2)|\gamma'_\mu(\lambda_2) - \gamma_0^+(\lambda_2)| + (1/2)|\gamma'_\mu(\lambda_1) - \gamma_0^+(\lambda_1)|$. Combining these inequalities, we have

$$\begin{aligned}
&|a_\mu - a_0| + |b_\mu - b_0| + |c_\mu - c_0| \\
&\leq |\gamma'_\mu(\lambda_2) - \gamma_0^+(\lambda_2)| + |\gamma'_\mu(\lambda_1) - \gamma_0^+(\lambda_1)| + |\gamma'_\mu(\overline{\lambda}) - \gamma_0^+(\overline{\lambda})| \\
&\leq 3|\gamma'_\mu(\tilde{\lambda}(z)) - \gamma_0^+(\tilde{\lambda}(z))|,
\end{aligned}$$

where the last inequality follows from $0 \leq \lambda_1 < \lambda_2$, $\overline{\lambda} \in [\lambda_1, \lambda_2]$ and Lemma 3.4.2 (b). Hence, by (3.4.17), we have for some $N > 0$ that $\|\nabla P_\mu(z) - J_P^o(z)\| \leq N|\gamma'_\mu(\tilde{\lambda}(z)) - \gamma_0^+(\tilde{\lambda}(z))|$. When $\lambda_1 < \lambda_2 \leq 0$, we can obtain the desired result in a similar way.

Finally, we consider the case where $\lambda_1 < 0 < \lambda_2$. First we assume $0 < |\lambda_1| \leq \lambda_2$. When $|\lambda_1| < \lambda_2$, there exists $\overline{\lambda} \in [|\lambda_1|, \lambda_2] = [-\lambda_1, \lambda_2]$ such that

$$\begin{aligned}
|a_\mu - a_0| &= \left| \frac{\gamma_\mu(\lambda_2) - \gamma_\mu(\lambda_1)}{\lambda_2 - \lambda_1} - \frac{\gamma_0(\lambda_2) - \gamma_0(\lambda_1)}{\lambda_2 - \lambda_1} \right| \\
&= \left| \frac{\gamma_\mu(\lambda_2) - \gamma_\mu(-\lambda_1)}{\lambda_2 - \lambda_1} - \frac{\gamma_0(\lambda_2) - \gamma_0(-\lambda_1)}{\lambda_2 - \lambda_1} \right| \\
&= \frac{\lambda_2 + \lambda_1}{\lambda_2 - \lambda_1} \left| \frac{\gamma_\mu(\lambda_2) - \gamma_\mu(-\lambda_1)}{\lambda_2 - (-\lambda_1)} - 1 \right| \\
&= \frac{\lambda_2 + \lambda_1}{\lambda_2 - \lambda_1} |\gamma'_\mu(\overline{\lambda}) - \gamma_0^+(\overline{\lambda})| \\
&\leq |\gamma'_\mu(\overline{\lambda}) - \gamma_0^+(\overline{\lambda})|,
\end{aligned} \tag{3.4.18}$$

where the second equality follows from Lemma 3.4.2 (a), the third equality follows from (3.4.8), the fourth equality follows from (3.4.9), and the inequality follows from $(\lambda_2 + \lambda_1)/(\lambda_2 - \lambda_1) = (|\lambda_2| - |\lambda_1|)/(|\lambda_2| + |\lambda_1|) \leq 1$. When $|\lambda_1| = \lambda_2$, we have $|a_\mu - a_0| = 0$ from (3.4.18) and Lemma 3.4.2 (a). In a way similar to the previous case, we have $|b_\mu - b_0| \leq (1/2)|\gamma'_\mu(\lambda_2) - \gamma_0^+(\lambda_2)| + (1/2)|\gamma'_\mu(\lambda_1) - \gamma_0^+(\lambda_1)|$ and $|c_\mu - c_0| \leq (1/2)|\gamma'_\mu(\lambda_2) - \gamma_0^+(\lambda_2)| + (1/2)|\gamma'_\mu(\lambda_1) - \gamma_0^+(\lambda_1)|$. Hence, by $0 < |\lambda_1| \leq \lambda_2$, $\overline{\lambda} \in [|\lambda_1|, \lambda_2]$, Lemma 3.4.2 (b) and (3.4.13), we have $\|\nabla P_\mu(z) - J_P^o(z)\| \leq N|\gamma'_\mu(\tilde{\lambda}(z)) - \gamma_0^+(\tilde{\lambda}(z))|$ for some $N > 0$. In the case where $\lambda_1 < 0 < \lambda_2$ and $0 < \lambda_2 \leq |\lambda_1|$, we can also show the desired result in a similar manner. ∎

Proposition 3.4.4 indicates that, for given $\alpha \in \Re$, if we can reduce the value of $|\gamma'_\mu(\alpha) - \gamma_0^+(\alpha)|$ arbitrarily close to zero by choosing a sufficiently small $\mu > 0$, then we can make $\nabla H_{\mu,\varepsilon}(x,y)$ arbitrarily close to $\partial H_{\rm NR}(x,y)$. Since $\gamma_0^+(\alpha) = \lim_{\mu \downarrow 0} \gamma'_\mu(\alpha)$, for any $\delta > 0$ and $\alpha \in \Re$, there exists $\overline{\mu}(\alpha,\delta) > 0$ such that $|\gamma'_\mu(\alpha) - \gamma_0^+(\alpha)| < \delta$ for any $\mu \in (0, \overline{\mu}(\alpha,\delta))$. The following proposition gives an explicit expression of such $\overline{\mu}(\alpha,\delta)$ when $\hat{g}(\alpha)$ is given by $\hat{g}(\alpha) = (\alpha + \sqrt{\alpha^2+4})/2$.

Proposition 3.4.5 *Let the function $\hat{g}$ be defined by $\hat{g}(\alpha) = (\alpha + \sqrt{\alpha^2+4})/2$, and γ_μ and γ_0^+ be defined by* (3.4.7) and (3.4.9)*, respectively. Moreover, let $\overline{\mu}(\alpha,\delta)$ be defined by*

$$\overline{\mu}(\alpha,\delta) := \begin{cases} +\infty & (\delta \geq 1/2 \ \text{ or } \ \alpha = 0) \\ \dfrac{1}{2}|\alpha|\sqrt{\delta} & (\delta < 1/2 \ \text{ and } \ \alpha \neq 0)\,. \end{cases}$$

Then, for any $\alpha \in \Re$, $\delta > 0$ and $\mu \in (0, \overline{\mu}(\alpha,\delta))$, we have

$$|\gamma'_\mu(\alpha) - \gamma_0^+(\alpha)| < \delta. \tag{3.4.19}$$

Proof. Let μ be an arbitrary scalar in the interval $(0, \overline{\mu}(\alpha,\delta))$. By easy calculation, we have $\gamma_\mu(\alpha) = (\alpha + \sqrt{\alpha^2+4\mu^2})/2$, $\gamma'_\mu(\alpha) = (1 + \alpha/\sqrt{\alpha^2+4\mu^2})/2$ and

$$\gamma_0^+(\alpha) = \begin{cases} 1 & (\alpha > 0) \\ 1/2 & (\alpha = 0) \\ 0 & (\alpha < 0)\,. \end{cases}$$

Moreover, these equalities yield

$$|\gamma'_\mu(\alpha) - \gamma_0^+(\alpha)| = \begin{cases} 0 & (\alpha = 0) \\ \dfrac{1}{2}\left(1 - \dfrac{|\alpha|}{\sqrt{\alpha^2+4\mu^2}}\right) & (\alpha \neq 0)\,. \end{cases}$$

When $\alpha = 0$, (3.4.19) trivially holds since $|\gamma'_\mu(\alpha) - \gamma_0^+(\alpha)| = 0 < \delta$. When $\delta \geq 1/2$, (3.4.19) always holds since $|\gamma'_\mu(\alpha) - \gamma_0^+(\alpha)| < 1/2$ for any $\delta > 0$ and $\mu > 0$. When $\alpha \neq 0$ and $0 < \delta < 1/2$, we have

$$\begin{aligned} |\gamma'_\mu(\alpha) - \gamma_0^+(\alpha)| - \delta &= \frac{1}{2}\left(1 - \frac{|\alpha|}{\sqrt{\alpha^2+4\mu^2}}\right) - \delta \\ &< \frac{1}{2}\left(1 - \frac{|\alpha|}{\sqrt{\alpha^2+\alpha^2\delta}}\right) - \delta \\ &= \frac{1}{2}\left(1 - \frac{1}{\sqrt{1+\delta}}\right) - \delta \\ &< 0, \end{aligned}$$

where the first inequality follows from $\mu < \overline{\mu}(\alpha,\delta) = (1/2)|\alpha|\sqrt{\delta}$, and the last inequality follows since the function $f(\delta) := (1/2)(1-(1+\delta)^{-1/2}) - \delta$ satisfies $f(0) = 0$ and $f'(\delta) = (1/4)(1+\delta)^{-3/2} - 1 < 0$ for any $\delta > 0$. This completes the proof. ∎

3.4.4 Quadratically convergent algorithm with Newton's method

In this subsection, based on Algorithm 3.4.1, we propose a more specific algorithm in which Newton's method is applied for solving subproblem (3.4.1). Moreover, we show that the algorithm is quadratically convergent under appropriate assumptions including the strong semismoothness and the Jacobian consistency. We first state the algorithm. For convenience, we denote

$$w := \begin{pmatrix} x \\ y \end{pmatrix}, \quad w^{(k)} := \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}.$$

Algorithm 3.4.2 *Choose η, $\rho \in (0,1)$, $\overline{\eta} \in (0, \eta]$, $\sigma \in (0, 1/2)$, $\kappa > 0$ and $\hat{\kappa} > 0$.*

Step 0 *Choose $w^{(0)} \in \Re^{2n}$ and $\beta_0 \in (0, \infty)$. Let $\mu_0 := \|H_{\rm NR}(w^{(0)})\|$ and $\varepsilon_0 := \|H_{\rm NR}(w^{(0)})\|$. Set $k := 0$.*

Step 1 *Terminate if $\|H_{\rm NR}(w^{(k)})\| = 0$.*

Step 2

Step 2.0 *Set $v^{(0)} := w^{(k)}$ and $j := 0$.*

Step 2.1 *Find a vector $\hat{d}^{(j)}$ such that*

$$H_{\mu_k,\varepsilon_k}(v^{(j)}) + \nabla H_{\mu_k,\varepsilon_k}(v^{(j)})^T \hat{d}^{(j)} = 0.$$

Step 2.2 *If $\|H_{\mu_k,\varepsilon_k}(v^{(j)} + \hat{d}^{(j)})\| \le \beta_k$, then let $w^{(k+1)} := v^{(j)} + \hat{d}^{(j)}$ and go to* Step 3. *Otherwise, go to* Step 2.3.

Step 2.3 *Find the smallest nonnegative integer m such that*

$$\Psi_{\mu_k,\varepsilon_k}(v^{(j)} + \rho^m \hat{d}^{(j)}) \le (1 - 2\sigma\rho^m)\Psi_{\mu_k,\varepsilon_k}(v^{(j)}).$$

Let $m_j := m$, $\tau_j := \rho^{m_j}$ and $v^{(j+1)} := v^{(j)} + \tau_j \hat{d}^{(j)}$.

Step 2.4 *If*

$$\|H_{\mu_k,\varepsilon_k}(v^{(j+1)})\| \le \beta_k, \tag{3.4.20}$$

then let $w^{(k+1)} := v^{(j+1)}$ and go to Step 3. *Otherwise, set $j := j + 1$ and go back to* Step 2.1.

Step 3 *Update the parameters as follows*:

$$\begin{aligned}
\mu_{k+1} &:= \min\left\{\kappa\|H_{\rm NR}(w^{(k+1)})\|^2, \mu_0\overline{\eta}^{k+1}, \overline{\mu}\Big(\tilde{\lambda}(x^{(k+1)} - y^{(k+1)}), \hat{\kappa}\|H_{\rm NR}(w^{(k+1)})\|\Big)\right\}, \\
\varepsilon_{k+1} &:= \min\left\{\kappa\|H_{\rm NR}(w^{(k+1)})\|^2, \varepsilon_0\overline{\eta}^{k+1}\right\}, \\
\beta_{k+1} &:= \beta_0\eta^{k+1}.
\end{aligned}$$

Set $k := k + 1$. Go back to Step 1.

In Step 3, $\tilde{\lambda}$ is the function given by (3.4.12), and $\overline{\mu}(\alpha,\delta)$ is determined so that $|\gamma'_\mu(\alpha) - \gamma_0^+(\alpha)| < \delta$ for any $\mu \in (0, \overline{\mu}(\alpha,\delta))$. An explicit formula of $\overline{\mu}(\alpha,\delta)$ is given as in Proposition 3.4.5 when $\hat{g}(\alpha) = (\alpha + \sqrt{\alpha^2+4})/2$.

In the inner iterations Steps 2.0 – 2.4, a damped Newton method seeks a point $w^{(k+1)}$ such that $\|H_{\mu_k,\varepsilon_k}(w^{(k+1)})\| \le \beta_k$. Note that, by letting $\alpha_k := \beta_k^2/2$, the termination criterion (3.4.20) for the inner iterations becomes equivalent to (3.4.1) in Algorithm 3.4.1. Step 3 specifies the updating rule of the parameters, where $\{\beta_k\}$, $\{\mu_k\}$ and $\{\varepsilon_k\}$ converge to 0 since $0 < \overline{\eta} \le \eta < 1$. Algorithm 3.4.2 is well-defined in the sense that Steps 2.0 – 2.4 find $v^{(j+1)}$ satisfying (3.4.20) in a finite number of iterations for each k. (For more detail, see Appendix E.)

Now, we show that Algorithm 3.4.2 is quadratically convergent under appropriate assumptions. Note that, from (3.4.2) and $\|x\| \le \|(x,y)\|$, there exists $\nu > 0$ such that

$$\|H_{\mu,\varepsilon}(w) - H_{\rm NR}(w)\| \le \nu\mu + \varepsilon\|w\| \tag{3.4.21}$$

for any $w \in \Re^{2n}$. By using this inequality and the two properties shown in Subsections 2.4.2 and 2.4.3, we establish the main theorem of this subsection.

Theorem 3.4.4 *Let $f : \Re^n \to \Re^n$ be a monotone function such that ∇f is locally Lipschitzian. Let $\{w^{(k)}\}$ be a sequence generated by Algorithm* 3.4.2. *Moreover, suppose that the following assumptions hold true*:

(i) *The solution set of SOCCP* (3.3.1) *is nonempty and bounded.*

(ii) *Every accumulation point of $\{\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})\}$ is nonsingular.*

Then we have the following two statements:

(a) *For all k sufficiently large, the inequality in* Step 2.2 *of Algorithm* 3.4.2 *holds for $j = 0$;*

(b) *The sequence $\{w^{(k)}\}$ converges to a solution w^* of SOCCP* (3.3.1) *quadratically.*

Proof. By assumption (i) and Theorem 3.4.1, $\{w^{(k)}\}$ is bounded and an arbitrary accumulation point w^* is a solution of SOCCP (3.3.1), that is, $H_{\rm NR}(w^*) = 0$. Let B be a positive number such that $\|w^{(k)}\| \le B$ for all k. Let $L > 0$ be a Lipschitzian constant of $H_{\rm NR}$ on a bounded set $\Omega \supset \{w^{(k)}\}$. Let $d^{(k)} := \hat{d}^{(0)} = -\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})^{-T} H_{\mu_k,\varepsilon_k}(w^{(k)})$, where J^{-T} denotes $J^{-T} := (J^T)^{-1} = (J^{-1})^T$. From the nonsingularity of $\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})$ [43, Proposition 6.1] and assumption (ii), there exists $C > 0$ such that

$$\|\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})^{-T}\| = \|\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})^{-1}\| \le C \tag{3.4.22}$$

for all k.

First we show $\|w^{(k)} + d^{(k)} - w^*\| = O(\|w^{(k)} - w^*\|^2)$. Let V_k be an element of $\partial H_{\rm NR}(w^{(k)})$ such that $\|\nabla H_{\mu_k,\varepsilon_k}(w^{(k)}) - V_k\| = \mathrm{dist}(\nabla H_{\mu_k,\varepsilon_k}(w^{(k)}), \partial H_{\rm NR}(w^{(k)}))$. It then follows from (3.4.22) and

$H_{\rm NR}(w^*)=0$ that

$$\begin{aligned}
&\|w^{(k)}+d^{(k)}-w^*\|\\
&=\left\|w^{(k)}-\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})^{-T}H_{\mu_k,\varepsilon_k}(w^{(k)})-w^*\right\|\\
&\le\left\|\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})^{-T}\right\|\left\|\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})^T(w^{(k)}-w^*)-H_{\mu_k,\varepsilon_k}(w^{(k)})\right\|\\
&\le C\Big\{\left\|(\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})-V_k)^T(w^{(k)}-w^*)\right\|+\left\|V_k^T(w^{(k)}-w^*)-H'_{\rm NR}(w^*;w^{(k)}-w^*)\right\|\\
&\quad+\left\|H'_{\rm NR}(w^*;w^{(k)}-w^*)-H_{\rm NR}(w^{(k)})+H_{\rm NR}(w^*)\right\|+\left\|H_{\rm NR}(w^{(k)})-H_{\mu_k,\varepsilon_k}(w^{(k)})\right\|\Big\}. \qquad (3.4.23)
\end{aligned}$$

Moreover, each term of (3.4.23) is $O(\|w^{(k)}-w^*\|^2)$ by the following reasons. Since $H_{\mu,\varepsilon}$ satisfies the Jacobian consistency, we have for some $M>0$ that

$$\begin{aligned}
\|\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})-V_k\| &\le M\Big(\left|\gamma'_{\mu_k}(\tilde\lambda(x^{(k)}-y^{(k)}))-\gamma_0^+(\tilde\lambda(x^{(k)}-y^{(k)}))\right|+\varepsilon_k\Big)\\
&\le M(\hat\kappa\|H_{\rm NR}(w^{(k)})\|+\kappa\|H_{\rm NR}(w^{(k)})\|^2)\\
&\le M(\hat\kappa+\kappa\|H_{\rm NR}(w^{(k)})\|)\|H_{\rm NR}(w^{(k)})-H_{\rm NR}(w^*)\|\\
&\le M(\hat\kappa+\kappa\|H_{\rm NR}(w^{(k)})\|)L\|w^{(k)}-w^*\|, \qquad (3.4.24)
\end{aligned}$$

where the first inequality follows from Proposition 3.4.4, the second inequality follows from Step 3 of the algorithm, the third inequality follows from $H_{\rm NR}(w^*)=0$, and the last inequality is due to the local Lipschitz continuity of $H_{\rm NR}$. By (3.4.24) and the boundedness of $\{\|H_{\rm NR}(w^{(k)})\|\}$, we have $\|(\nabla H_{\mu_k,\varepsilon_k}(w^{(k)})-V_k)^T(w^{(k)}-w^*)\|=O(\|w^{(k)}-w^*\|^2)$. The second and third terms of (3.4.23) are also $O(\|w^{(k)}-w^*\|^2)$ since $H_{\rm NR}$ is strongly semismooth and directionally differentiable. Finally, we consider the fourth term of (3.4.23). From (3.4.21) and Step 3 of Algorithm 3.4.2, we have $\|H_{\rm NR}(w^{(k)})-H_{\mu_k,\varepsilon_k}(w^{(k)})\|\le\nu\mu_k+B\varepsilon_k=O(\|H_{\rm NR}(w^{(k)})\|^2)$. Moreover, $O(\|H_{\rm NR}(w^{(k)})\|^2)=O(\|w^{(k)}-w^*\|^2)$ from $H_{\rm NR}(w^*)=0$ and local Lipschitz continuity of $H_{\rm NR}$. Therefore, the fourth term of (3.4.23) is $O(\|w^{(k)}-w^*\|)$. Consequently, we have

$$\|w^{(k)}+d^{(k)}-w^*\|=O(\|w^{(k)}-w^*\|^2). \qquad (3.4.25)$$

Next we show (a), that is, $\|H_{\mu_k,\varepsilon_k}(w^{(k)}+d^{(k)})\|\le\beta_k$ for all k sufficiently large. Note that $\{d^{(k)}\}$ is bounded from the inequality $\|d^{(k)}\|\le\|w^{(k)}+d^{(k)}-w^*\|+\|w^{(k)}-w^*\|$ and (3.4.25). We therefore have for some $\Gamma>0$ that

$$\begin{aligned}
\|H_{\mu_k,\varepsilon_k}(w^{(k)}+d^{(k)})\| &\le \|H_{\mu_k,\varepsilon_k}(w^{(k)}+d^{(k)})-H_{\rm NR}(w^{(k)}+d^{(k)})\|+\|H_{\rm NR}(w^{(k)}+d^{(k)})\|\\
&\le \Gamma\|H_{\rm NR}(w^{(k)})\|^2+\|H_{\rm NR}(w^{(k)}+d^{(k)})\|, \qquad (3.4.26)
\end{aligned}$$

where the second inequality follows from (3.4.21), the boundedness of $\{w^{(k)}+d^{(k)}\}$ and Step 3 of Algorithm 3.4.2. Hence, it suffices to show $\|H_{\rm NR}(w^{(k)}+d^{(k)})\|=O(\|H_{\rm NR}(w^{(k)})\|^2)$ and $\|H_{\rm NR}(w^{(k)})\|=O(\eta^k)$. Notice that assumption (ii) implies that $H_{\rm NR}$ satisfies the local error bound property at w^*, that is, $\|w^{(k)}-w^*\|=O(\|H_{\rm NR}(w^{(k)})\|)$. Then, we have $\|H_{\rm NR}(w^{(k)}+d^{(k)})\|\le\overline{L}\|w^{(k)}+d^{(k)}-w^*\|=O(|w^{(k)}-w^*\|^2)=O(\|H_{\rm NR}(w^{(k)})\|^2)$, where $\overline{L}$ is a Lipschitzian constant for

H_{NR} on a bounded set $\overline{\Omega} \supset \{w^{(k)} + d^{(k)}\}$. On the other hand, $\|H_{\mathrm{NR}}(w^{(k)})\| = O(\eta^k)$ since

$$\begin{aligned}\|H_{\mathrm{NR}}(w^{(k)})\| &\leq \|H_{\mu_{k-1},\varepsilon_{k-1}}(w^{(k)})\| + \|H_{\mu_{k-1},\varepsilon_{k-1}}(w^{(k)}) - H_{\mathrm{NR}}(w^{(k)})\| \\ &\leq \beta_{k-1} + \nu\mu_{k-1} + B\varepsilon_{k-1} \\ &\leq \beta_0\eta^{k-1} + \nu\mu_0\eta^{k-1} + B\varepsilon_0\eta^{k-1} \\ &= \eta^{-1}(\beta_0 + \nu\mu_0 + B\varepsilon_0)\eta^k,\end{aligned}$$

where the second inequality follows from Step 2.4 of Algorithm 3.4.2 and (3.4.21), and the third inequality follows from Step 3 in Algorithm 3.4.2 and $0 < \overline{\eta} \leq \eta < 1$. Combining these results with (3.4.26), we have $\|H_{\mu_k,\varepsilon_k}(w^{(k)} + d^{(k)})\| \leq \|H_{\mathrm{NR}}(w^{(k)})\|\, O(\eta^k)$, which implies (a). Finally, (b) directly follows from (3.4.25) and (a). ∎

We note that assumption (ii) holds if f is strongly monotone around a solution. This can be observed as follows. From [43, Proposition 6.1], we have $O \prec \nabla P_\mu(z) \prec I$ for any $\mu > 0$ and $z \in \Re^n$, which implies $O \preceq J_P^o(z) \preceq I$. Moreover, the matrix $J_H^o(x, y)$ defined by (3.4.6) is nonsingular since the matrix $I - J_P^o(x - y) + J_P^o(x - y)\nabla f(x)$ is nonsingular by the positive definiteness of $\nabla f(x)$ and [119, Proposition 2.1(b)].

There have been studied many Newton-type methods with smoothing and regularization techniques for solving NCPs and box constrained variational inequalities (BVI). Our algorithm is regarded as an extension of those methods to SOCCP. However, it is different from the existing Newton-type methods for NCP and BVI in some respects. The Jacobian smoothing method [17, 69] is one of the popular Newton-type methods. In this method, an approximate Newton direction $d = -\nabla H_\mu(z)^{-T} H(z)$ of H is used instead of Newton direction $-\nabla H_\mu(z)^{-T} H_\mu(z)$ of H_μ itself, which our algorithm adopts. Another approach [95, 102] is to consider the nonlinear equation $G(x, \mu) = 0$ where $G(x, \mu) := \binom{H_\mu(z)}{\mu}$, and solve it by Newton's method that treats parameter μ as a variable. Moreover, in [95], a regularization parameter is also included. On the other hand, our method distinguishes between parameters and variables strictly.

3.5 Numerical experiments

In order to evaluate the efficiency of Algorithm 3.4.2, we have conducted some numerical experiments. In our experiments, we chose $\hat{g}(\alpha) = (\sqrt{\alpha^2 + 4} + \alpha)/2$ as the function satisfying (3.3.3) and (3.4.10), and employed $\|H_{\mathrm{NR}}(w^{(k)})\| < 10^{-8}$ as the termination criterion. Moreover, we adopted

$$\overline{\mu}(\alpha, \delta) := \begin{cases} 10^{10} & (\delta \geq 1/2 \ \text{ or } \ \alpha = 0) \\ \dfrac{1}{2}|\alpha|\sqrt{\delta} & (\delta < 1/2 \ \text{ and } \ \alpha \neq 0)\,. \end{cases}$$

in Step 3 of the algorithm. Note that, when $\tilde{\lambda}(x^{(k)} - y^{(k)})$ defined by (3.4.12) is very small but positive while $\|H_{\mathrm{NR}}(x^{(k)}, y^{(k)})\|$ is not small enough, μ_k becomes almost as small as $\tilde{\lambda}(x^{(k)} - y^{(k)})$, although $(x^{(k)}, y^{(k)})$ is not sufficiently close to a solution. Since this may diminish the effects of smoothing, we regard $\lambda_i(x^{(k)} - y^{(k)})$ as 0 when $\lambda_i(x^{(k)} - y^{(k)})/\|H_{\mathrm{NR}}(x^{(k)}, y^{(k)})\| \leq \overline{\delta}$ holds for a

sufficiently small $\overline{\delta} > 0$. In our implementation, we therefore modified the definition (3.4.12) of $\tilde{\lambda}(z)$ such that the index set $\mathcal{I}(z) := \{i \,|\, |\lambda_i(z)| > 10^{-4}\|H_{\rm NR}(x,y)\|\}$ is used instead of $\mathcal{I}(z) = \{i \,|\, \lambda_i(z) \neq 0\}$. The program was coded in MATLAB 6.5 and run on a machine with AMD Athlon(tm) XP 2000+ CPU and 1GB RAM.

3.5.1 Linear case

We solved the following linear SOCCP : Find $(x, y) \in \Re^n \times \Re^n$ such that

$$x \in \mathcal{K}^n,\ y \in \mathcal{K}^n,\ x^T y = 0,\ y = Mx + q, \tag{3.5.1}$$

where $M \in \Re^{n\times n}$ is a rank-deficient positive semidefinite matrix. We chose $\eta = 0.01$, $\overline{\eta} = 0.001$, $\rho = 0.5$, $\sigma = 0.4$, $\kappa = 0.01$ and $\hat{\kappa} = 1$ in Algorithm 3.4.2.

In order to obtain a positive semidefinite matrix M with $\operatorname{rank} M = r < n$, we let $M := nBB^T/\|BB^T\|$, where $B \in \Re^{n\times r}$ is a matrix of which components are randomly chosen from the interval $[-1, 1]$. Furthermore, we let $q := 10^{\alpha} n^{1/2} p - Me$, where $e = (1, 0, \ldots, 0)^T \in \operatorname{int} \mathcal{K}^n$, p is a vector such that $p \in \operatorname{int} \mathcal{K}^n$ and $\|p\| = 1$, and α is randomly chosen from the interval $[-1, 1]$. Then, SOCCP (3.5.1) has a solution, since M is positive semidefinite and there exist $\overline{x} \in \operatorname{int} \mathcal{K}^n$ and $\overline{y} \in \operatorname{int} \mathcal{K}^n$ such that $\overline{y} = M\overline{x}+q$. (See Appendix F.) In the experiments, we determined p as follows : Let θ be randomly chosen from $(0, \pi/2)$, let w be an $(n-1)$ - dimensional vector whose components are randomly chosen from $[-1, 1]$, and put $p := 2^{-1/2} \cos\theta(1, w/\|w\|) + 2^{-1/2} \sin\theta(1, -w/\|w\|)$. It is not difficult to see that a vector p thus determined satisfies $\|p\| = 1$ and $p \in \operatorname{int} \mathcal{K}^n$.

In our first experiments, we generated 100 problem instances for each of $n = 100, 200, \ldots, 1000$, and solved each problem instance by using 100 randomly chosen initial points. We let the rank r of M be an integer randomly chosen from $[0.9n, n-1]$, and selected an initial point $(x^{(0)}, y^{(0)})$ as $(x^{(0)}, y^{(0)}) := 10^{\beta}(a, b)/\|(a, b)\|$, where β is randomly chosen from $[-3, 3]$ and each component of $(a, b) \in \Re^n \times \Re^n$ is randomly chosen from $[-1, 1]$. Table 3.1 shows the results of our experiments, in which n denotes the size of problems, $\sharp$Ite denotes the number of outer iterations, $\sharp$Newton denotes the total number of inner Newton iterations, and cpu(s) denotes the CPU time in second. The values of $\sharp$Ite, $\sharp$Newton and cpu(s) are the average of 100 runs for each n. Table 3.1 reveals that the problem size only slightly affects the number of iterations.

In our second experiments, we fix the size of problems at 100 and varied the rank of $M \in \Re^{100\times 100}$ as $r = 10, 20, \ldots, 90$ and 99. We solved 1000 different problem instances for each r, where initial points are selected in a way similar to the previous experiments. Table 3.2 shows the results of the experiments. The numbers of iterations, $\sharp$Ite and $\sharp$Newton, are the average of 1000 runs for each r. Table 3.2 indicates that neither $\sharp$Ite nor $\sharp$Newton is affected by the rank of matrix M.

3.5.2 Comparison with interior point method

We solved the following SOCP by our method and interior point method.

$$\begin{aligned} &\text{Minimize} && c^T z \\ &\text{subject to} && z \in \mathcal{K}_1,\ \ Az + b \in \mathcal{K}_2, \end{aligned} \tag{3.5.2}$$

Table 3.1: Results for linear SOCCPs of various problem sizes

n	♯Ite	♯Newton	cpu(s)
100	5.28	7.12	0.151
200	5.58	7.93	1.026
300	5.86	8.48	2.848
400	5.65	8.80	5.960
500	5.77	8.89	10.603
600	5.80	9.12	17.346
700	5.96	9.37	26.252
800	5.99	9.36	37.185
900	5.93	9.71	52.223
1000	5.98	9.52	67.316

where $A \in \Re^{m\times n}$, $b \in \Re^m$, $c \in \Re^n$, $\mathcal{K}_1 := \mathcal{K}^{n_1} \times \cdots \times \mathcal{K}^{n_N} \subset \Re^n$ and $\mathcal{K}_2 := \mathcal{K}^{m_1} \times \cdots \times \mathcal{K}^{m_M} \subset \Re^m$. Since the KKT conditions for SOCP (3.5.2) are written as

$$\begin{pmatrix} z \\ \lambda \end{pmatrix} \in \mathcal{K}_1 \times \mathcal{K}_2, \ \begin{pmatrix} O & -A^T \\ A & O \end{pmatrix}\begin{pmatrix} z \\ \lambda \end{pmatrix} + \begin{pmatrix} c \\ b \end{pmatrix} \in \mathcal{K}_1 \times \mathcal{K}_2, \ \begin{pmatrix} z \\ \lambda \end{pmatrix}^T \left\{ \begin{pmatrix} O & -A^T \\ A & O \end{pmatrix}\begin{pmatrix} z \\ \lambda \end{pmatrix} + \begin{pmatrix} c \\ b \end{pmatrix} \right\} = 0, \tag{3.5.3}$$

we can obtain a solution of SOCP (3.5.2) by solving SOCCP (3.5.3). In this experiment, we chose eight pairs of $(\mathcal{K}_1, \mathcal{K}_2)$ such that $n_1 = \cdots = n_N$ and $m_1 = \cdots = m_M$ and generated 100 problem instances for each $(\mathcal{K}_1, \mathcal{K}_2)$, where matrix A is randomly chosen so that its components are contained in the interval $[-1, 1]$, and vectors b and c are randomly chosen so that SOCP (3.5.2) is solvable.

In applying Algorithm 3.4.2 to SOCCP (3.5.3), we chose parameters and initial points in a way similar to the previous experiment. As the interior point method for SOCP, we used SDPT3 solver [109, 113]. We give the results in Table 3.3, in which (n, N) denotes the dimension of $\mathcal{K}_1$ and the number of second-order cones comprising $\mathcal{K}_1$, (m, M) applies similarly to $\mathcal{K}_2$, and the numbers of iterations are the average of 100 runs for each $(\mathcal{K}_1, \mathcal{K}_2)$. As this table shows, our method tends to require fewer iterations than the interior point method when the dimension of second-order cone is large.

3.5.3 Nonlinear case

We solved the following nonlinear SOCCP : Find $(x, y) \in \Re^n \times \Re^n$ such that

$$x \in \mathcal{K}, \ y \in \mathcal{K}, \ x^T y = 0, \ y = f(x), \tag{3.5.4}$$

Table 3.2: Results for linear SOCCPs with various degrees of rank deficiency

r	♯Ite	♯Newton
99	5.396	7.346
90	5.361	7.260
80	5.284	7.207
70	5.245	7.115
60	5.242	7.189
50	5.172	7.004
40	5.233	7.128
30	5.116	6.988
20	5.072	6.809
10	4.894	6.371

Table 3.3: Comparison of Algorithm 3.4.2 and the interior point method on linear SOCPs

(n, N)	(m, M)	Our method		SDPT3
		♯Ite	♯Newton	♯Ite
$(100, 2)$	$(100, 4)$	5.07	11.07	12.58
$(100, 20)$	$(100, 25)$	5.21	18.55	15.81
$(400, 8)$	$(300, 5)$	5.30	9.23	14.05
$(400, 40)$	$(300, 50)$	5.53	11.52	15.24
$(800, 4)$	$(900, 3)$	5.74	9.67	15.19
$(800, 50)$	$(900, 60)$	5.71	12.24	16.12

where $\mathcal{K} = \mathcal{K}^3 \times \mathcal{K}^2$ and $f : \Re^5 \to \Re^5$ is given by

$$f(x) := \begin{pmatrix} 24(2x_1 - x_2)^3 + \exp(x_1 - x_3) - 4x_4 + x_5 \\ -12(2x_1 - x_2)^3 + 3(3x_2 + 5x_3)/\sqrt{1 + (3x_2 + 5x_3)^2} - 6x_4 - 7x_5 \\ -\exp(x_1 - x_3) + 5(3x_2 + 5x_3)/\sqrt{1 + (3x_2 + 5x_3)^2} - 3x_4 + 5x_5 \\ 4x_1 + 6x_2 + 3x_3 - 1 \\ -x_1 + 7x_2 - 5x_3 + 2 \end{pmatrix}.$$

We note that, in view of the KKT conditions (3.6.4) for the SOCP (3.6.1) (see Section 2.6), SOCCP (3.5.4) is equivalent to the following SOCP:

$$\begin{array}{ll} \text{Minimize} & \exp(z_1 - z_3) + 3(2z_1 - z_2)^4 + \sqrt{1 + (3z_2 + 5z_3)^2} \\ \text{subject to} & \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix} \in \mathcal{K}^3, \quad \begin{pmatrix} 4 & 6 & 3 \\ -1 & 7 & -5 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix} + \begin{pmatrix} -1 \\ 2 \end{pmatrix} \in \mathcal{K}^2. \end{array}$$

Since the objective function of this SOCP is convex, we can easily see that the function f is monotone. In the experiments, we set the parameters as $\rho = 0.5$, $\sigma = 0.4$, $\kappa = 0.01$ and $\hat{\kappa} = 1$.

We used nine different pairs $(\eta, \overline{\eta})$ and ran Algorithm 3.4.2 with 100 initial points $(x^{(0)}, y^{(0)})$ for each $(\eta, \overline{\eta})$ determined as $(x^{(0)}, y^{(0)}) := \Gamma(a, b)/\|(a, b)\|$, where Γ is randomly chosen from $[0, 10]$ and each component of $(a, b) \in \Re^n \times \Re^n$ is randomly chosen from $[-1, 1]$. Table 3.4 shows the results of the experiments, in which $\sharp$Ite and $\sharp$Newton are the average of 100 runs for each $(\eta, \overline{\eta})$. As Table 3.4 shows, the number of outer iterations decreases as η becomes smaller. On the other hand, the number of inner iterations decreases as $\overline{\eta}$ becomes smaller and η becomes larger. Since the computation time is largely dependent on the number of inner iterations, it would be recommended to choose η as large as possible and $\overline{\eta}$ as small as possible. However, it should be kept in mind that, when $\overline{\eta}$ is too small, μ and ε tend to 0 very rapidly, which may deteriorate the advantage of smoothing and regularization methods.

Table 3.4: Results for nonlinear SOCCPs with various choices of $(\eta, \overline{\eta})$

η	$\overline{\eta}$	$\sharp$Ite	$\sharp$Newton
0.5	0.5	16.79	26.34
0.5	0.1	12.20	13.70
0.5	0.01	10.50	10.75
0.5	0.001	9.58	9.85
0.1	0.1	8.35	19.11
0.1	0.01	8.13	13.51
0.1	0.001	7.66	11.25
0.01	0.01	5.99	14.32
0.01	0.001	5.73	12.35

3.6 Concluding remarks

In this section, we make some comments that complement the results obtained in this chapter.

In this chapter, the function F in SOCCP (1.1.1) is assumed to be of the form $F(x, y, \zeta) = f(x) - y$. This assumption may seem rather restrictive. However, the KKT conditions for the SOCP

$$\begin{array}{ll} \text{Minimize} & \theta(z) \\ \text{subject to} & \gamma(z) \in \mathcal{K} \end{array} \tag{3.6.1}$$

can be written as the SOCCP with $F(x, y, \zeta) = f(x) - y$ as follows: In SOCP (3.6.1), let $z = z' - z''$ with $z' \in \Re^s_+$ and $z'' \in \Re^s_+$, and denote $\hat{z} := \binom{z'}{z''} \in \Re^{2s}_+$, where $\Re^n_+$ is the n-dimensional nonnegative orthant. Moreover, define $\hat{\theta} : \Re^{2s} \to \Re$ by $\hat{\theta}(\hat{z}) = \theta(z' - z'')$, and $\hat{\gamma} : \Re^{2s} \to \Re^t$ by $\hat{\gamma}(\hat{z}) = \gamma(z' - z'')$.

Then SOCP (3.6.1) can be reformulated as

$$\begin{array}{ll} \text{Minimize} & \hat{\theta}(\hat{z}) \\ \text{subject to} & \begin{pmatrix} \hat{\gamma}(\hat{z}) \\ \hat{z} \end{pmatrix} \in \mathcal{K} \times \Re^{2s}_+, \end{array} \tag{3.6.2}$$

and the KKT conditions for (3.6.2) are written as

$$\begin{aligned} & \nabla\hat{\theta}(\hat{z}) - \begin{pmatrix} \nabla\hat{\gamma}(\hat{z}) & I \end{pmatrix} \begin{pmatrix} \hat{\lambda}_1 \\ \hat{\lambda}_2 \end{pmatrix} = 0, \\ & \begin{pmatrix} \hat{\lambda}_1 \\ \hat{\lambda}_2 \end{pmatrix} \in \mathcal{K} \times \Re^{2s}_+, \quad \begin{pmatrix} \hat{\gamma}(\hat{z}) \\ \hat{z} \end{pmatrix} \in \mathcal{K} \times \Re^{2s}_+, \quad \begin{pmatrix} \hat{\lambda}_1 \\ \hat{\lambda}_2 \end{pmatrix}^T \begin{pmatrix} \hat{\gamma}(\hat{z}) \\ \hat{z} \end{pmatrix} = 0. \end{aligned} \tag{3.6.3}$$

Now, let $\hat{\mu}_1 = \hat{\gamma}(\hat{z})$, and notice that (3.2.3) holds. Then, (3.6.3) can be rewritten as

$$\begin{aligned} & \begin{pmatrix} \hat{\gamma}(\hat{z}) \\ \nabla\hat{\theta}(\hat{z}) - \nabla\hat{\gamma}(\hat{z})\hat{\lambda}_1 \end{pmatrix} = \begin{pmatrix} \hat{\mu}_1 \\ \hat{\lambda}_2 \end{pmatrix}, \\ & \begin{pmatrix} \hat{\lambda}_1 \\ \hat{z} \end{pmatrix} \in \mathcal{K} \times \Re^{2s}_+, \quad \begin{pmatrix} \hat{\mu}_1 \\ \hat{\lambda}_2 \end{pmatrix} \in \mathcal{K} \times \Re^{2s}_+, \quad \begin{pmatrix} \hat{\lambda}_1 \\ \hat{z} \end{pmatrix}^T \begin{pmatrix} \hat{\mu}_1 \\ \hat{\lambda}_2 \end{pmatrix} = 0. \end{aligned} \tag{3.6.4}$$

Setting

$$x = \begin{pmatrix} \hat{\lambda}_1 \\ \hat{z} \end{pmatrix}, \ y = \begin{pmatrix} \hat{\mu}_1 \\ \hat{\lambda}_2 \end{pmatrix}, \ f(x) = \begin{pmatrix} \hat{\gamma}(\hat{z}) \\ \nabla\hat{\theta}(\hat{z}) - \nabla\hat{\gamma}(\hat{z})\hat{\lambda}_1 \end{pmatrix},$$

the KKT conditions (3.6.4) for SOCP (3.6.1) can be reduced to the SOCCP with $F(x, y, \zeta) = f(x) - y$. Note that the KKT conditions (3.6.4) for SOCP (3.6.1) contain more variables than the original KKT conditions. Furthermore, some desirable properties of the functions involved in SOCP (3.6.1) may be lost. For example, even if θ and γ in SOCP (3.6.1) are strictly convex, $\hat{\theta}$ and $\hat{\gamma}$ in (3.6.2) are merely convex. Hence, it would be useful to develop a method that can directly deal with the original KKT conditions, or more generally, SOCCP involving the function $F(x, y, \zeta)$ which is not restricted to be of the form $F(x, y, \zeta) = f(x) - y$.

In Sections 2.3 and 2.4, it is also assumed that $\mathcal{K} = \mathcal{K}^n$. For the general case where $\mathcal{K} = \mathcal{K}^{n_1} \times \cdots \times \mathcal{K}^{n_m}$, $H_{\mu,\varepsilon}(x, y)$ and $\nabla H_{\mu,\varepsilon}(x, y)$ are given by

$$\begin{aligned} H_{\mu,\varepsilon}(x, y) &= \begin{pmatrix} \varphi_\mu(x^1, y^1) \\ \vdots \\ \varphi_\mu(x^m, y^m) \\ f_\varepsilon(x) - y \end{pmatrix}, \\ \nabla H_{\mu,\varepsilon}(x, y) &= \begin{pmatrix} I - \mathrm{diag}\{\nabla P_\mu(x^j - y^j)\}_{j=1}^m & \nabla f(x) + \varepsilon I \\ \mathrm{diag}\{\nabla P_\mu(x^j - y^j)\}_{j=1}^m & -I \end{pmatrix}, \end{aligned}$$

where $x = (x^1, \ldots, x^m) \in \Re^{n_1} \times \cdots \times \Re^{n_m}$, $y = (y^1, \ldots, y^m) \in \Re^{n_1} \times \cdots \times \Re^{n_m}$, and $\mathrm{diag}\{\nabla P_\mu(x^j - y^j)\}_{j=1}^m$ denotes the block-diagonal matrix with block elements $\nabla P_\mu(x^j - y^j)$. Moreover, the definition (3.4.12) of $\tilde{\lambda}(z)$ is replaced by

$$\tilde{\lambda}(z) := \begin{cases} \min\limits_{(i,j) \in \mathcal{I}(z)} |\lambda_i(z^j)| & (\mathcal{I}(z) \neq \emptyset) \\ 0 & (\mathcal{I}(z) = \emptyset), \end{cases}$$

where $z = (z^1, \ldots, z^m) \in \Re^{n_1} \times \cdots \times \Re^{n_m}$, $\lambda_i(z^j)$ $(i = 1, 2)$ are the spectral values of z^j, and $\mathcal{I}(z) \subseteq \{1, 2\} \times \{1, \ldots, m\}$ is the index set defined by $\mathcal{I}(z) := \{(i, j) \,|\, \lambda_i(z^j) \neq 0\}$. Then, we can also show that Theorems 3.4.1 and 3.4.4 hold for monotone f in a similar way.

Chapter 4

A matrix splitting method for symmetric affine second-order cone complementarity problems

4.1 Introduction

In this chapter, we focus on the symmetric affine SOCCP :

$$\begin{aligned} \text{Find} \quad & z \in \Re^n \\ \text{such that} \quad & z \in \mathcal{K}, \;\; Mz + q \in \mathcal{K}, \;\; z^T(Mz + q) = 0, \end{aligned} \tag{4.1.1}$$

where $M \in \Re^{n \times n}$ and $q \in \Re^n$ are a given symmetric matrix and a vector, respectively, and $\mathcal{K} \subset \Re^n$ is the Cartesian product of SOCs, i.e.,

$$\mathcal{K} = \mathcal{K}^{n_1} \times \mathcal{K}^{n_2} \times \cdots \times \mathcal{K}^{n_m}$$

with $n = n_1 + n_2 + \cdots + n_m$. For instance, the KKT conditions for the Lagrangian dual problem of the SOCP :

$$\begin{aligned} \text{Minimize} \quad & \frac{1}{2} z^T Q z + c^T z \\ \text{subject to} \quad & Az + b \in \mathcal{K} \end{aligned}$$

can be written in the form of symmetric affine SOCCP (4.1.1), where Q is a symmetric positive definite matrix. Throughout the chapter, we often denote SOCCP (4.1.1) as SOCCP $(q, M, \mathcal{K})$.

Recently, several methods have been proposed for solving SOCCPs. One of the popular approaches is to reformulate the SOCCP into an equivalent nondifferentiable minimization problem and solve it by smoothing methods [15, 19, 43, 57]. Such an approach is motivated by smoothing methods for nonlinear complementarity problems [12, 16, 69, 101]. However, smoothing methods may sometimes be expensive computationally for large scale problems, and hence, methods exploiting particular features of matrices such as the sparsity and the block structure are required. In this chapter, we propose an approach based on a matrix splitting method for the affine SOCCP.

In matrix splitting methods, we represent the matrix M as the sum of two matrices B and C where B has a certain simple structure. Then, by solving subproblems involving the matrix B successively, the method generates a sequence converging to a solution of the original problem. Historically, matrix splitting methods have been used to solve the system of linear equations [46]. Several splitting schemes such as Jacobi, Gauss-Seidel, and the successive overrelaxation (SOR) methods have been suggested and various parallel algorithms have been developed on the basis of those matrix splitting schemes. Subsequently, those methods have been extended to the linear complementarity problem (LCP) and the affine variational inequality problem [24, 63, 77, 78, 79, 123].

In this chapter, we extend the matrix splitting method [24] for LCP to SOCCP. Especially, we propose to adopt the block SOR method. The block SOR method for LCP generates subproblems that can be solved easily by simple substitution operations. However, if the method is applied to SOCCP in a direct manner, generated subproblems need not be tractable in general. We will present a special block SOR method for SOCCP in which generated subproblems possess certain particular structures. In our approach, the subproblems are transformed into equivalent single variable equations by exploiting the special structures of subproblems, and then, those equations are solved efficiently by adopting some idea used in the trust region method.

The chapter is organized as follows: In Section 4.2, we describe the basic matrix splitting method for affine SOCCPs, and give conditions for the method to be convergent. In Section 4.3, we present the block SOR method as a particular realization of the basic splitting method of Section 4.2. In Section 4.4, we give a concrete procedure to solve subproblems of the block SOR method. In Section 4.5, we report numerical results with the proposed method. Finally, we conclude the chapter in Section 4.6.

4.2 Matrix splitting method and its convergence

In this section, we extend the matrix splitting method for LCP to SOCCP, and study its convergence property. Let the symmetric matrix M be represented as the sum of two matrices $B \in \Re^{n\times n}$ and $C \in \Re^{n\times n}$, i.e.,

$$M = B + C, \tag{4.2.1}$$

where B and C need not be symmetric. Such a pair (B, C) is called a splitting of M. The basic algorithm of the matrix splitting method for the SOCCP is stated as follows:

Algorithm 4.2.1

Step 1. *Choose a splitting (B, C) of M and an initial point $z^0 \in \mathcal{K}$. Set $k := 0$.*

Step 2. *Solve the following affine SOCCP*:

$$\begin{aligned} &\textit{Find} \quad z \in \Re^n \\ &\textit{such that} \quad z \in \mathcal{K},\ Bz + q^k \in \mathcal{K},\ z^T(Bz + q^k) = 0, \end{aligned} \tag{4.2.2}$$

where

$$q^k := q + Cz^k. \tag{4.2.3}$$

Let z^{k+1} be a solution of problem (4.2.2).

Step 3. *If $z^{k+1} = z^k$, terminate. Otherwise, return to Step* 2 *with k replaced by $k+1$.*

It is particularly important to choose a splitting (B, C) so that SOCCP (4.2.2) can be solved efficiently and the sequence $\{z^k\}$ generated by the algorithm converges to a solution of SOCCP (4.1.1).

Next, by extending the convergence theory of splitting methods for LCP [24, 63] to SOCCP, we give conditions for Algorithm 4.2.1 to be convergent. The following two definitions are necessary for describing the conditions.

Definition 4.2.1 *If SOCCP $(q, B, \mathcal{K})$ has a solution for any $q \in \Re^n$, then B is called a $\mathcal{K}$-Q-matrix. Moreover, if B is a $\mathcal{K}$-Q-matrix, then (B, C) is called a $\mathcal{K}$-Q-splitting.*

Definition 4.2.2 *If $B-C$ is positive (semi-)definite, then the splitting (B, C) is said to be (weakly) regular.*

It is evident from the definition that SOCCP (4.2.2) always has a solution if (B, C) is a $\mathcal{K}$-Q-splitting. On the other hand, the regularity of the splitting (B, C) plays an important role in discussing the convergence property of Algorithm 4.2.1.

In the remainder of the section, we will show that, if (B, C) is a regular $\mathcal{K}$-Q-splitting, then the sequence generated by Algorithm 4.2.1 converges to a solution of SOCCP (4.1.1). The process of convergence analysis may seem similar to that for LCP [24]. However, they are different in many respects since the complementarity condition for SOCCP cannot be decomposed into more than m blocks (m is the number of SOC comprising $\mathcal{K}$), while LCP can be decomposed into n blocks, i.e. $(Mx+q)_i \geq 0$, $x_i \geq 0$, $x_i(Mx+q)_i = 0,\ i = 1, \ldots, n$. The subsequent analyses are concerned with SOCs, but they can be also applied to general self-dual cones.

Now, we first introduce function $\theta : \Re^n \to \Re$ defined by

$$\theta(z) := \frac{1}{2} z^T M z + q^T z.$$

This function serves as a merit function on $\mathcal{K}$ for SOCCP (4.1.1) since matrix M is symmetric and hence a minimizer of θ on $\mathcal{K}$ solves SOCCP (4.1.1). In the subsequent discussions, we will use function θ in order to monitor the progress of the algorithm. The following lemma gives a sufficient condition for the sequence $\{\theta(z^k)\}$ to be nonincreasing.

Lemma 4.2.1 *Let (B, C) be a weakly regular $\mathcal{K}$-Q-splitting of the symmetric matrix M. Let $\{z^k\}$ be a sequence generated by Algorithm* 4.2.1. *Then, we have for each k*

$$\theta(z^k) - \theta(z^{k+1}) \geq \frac{1}{2}(z^k - z^{k+1})^T (B - C)(z^k - z^{k+1}) \geq 0. \tag{4.2.4}$$

In particular, if (B, C) is a regular $\mathcal{K}$-Q-splitting, then the second inequality in (4.2.4) *holds strictly whenever $z^k \neq z^{k+1}$, and moreover $\theta(z^k) = \theta(z^{k+1})$ if and only if $z^k = z^{k+1}$.*

Proof. By (4.2.1), (4.2.3) and the symmetry of M, we have

$$\begin{aligned}\theta(z^k) - \theta(z^{k+1}) &= (z^k - z^{k+1})^T(q + Mz^{k+1}) + \frac{1}{2}(z^k - z^{k+1})^T M(z^k - z^{k+1}) \\ &= (z^k - z^{k+1})^T(q + (B + C)z^{k+1}) \\ &\quad + \frac{1}{2}(z^k - z^{k+1})^T(B - C)(z^k - z^{k+1}) + (z^k - z^{k+1})^T C(z^k - z^{k+1}) \\ &= (z^k - z^{k+1})^T(q^k + Bz^{k+1}) + \frac{1}{2}(z^k - z^{k+1})^T(B - C)(z^k - z^{k+1}).\end{aligned}$$

Moreover, $(z^k)^T(q^k + Bz^{k+1}) \geq 0$ since $z^k \in \mathcal{K}$ and $q^k + Bz^{k+1} \in \mathcal{K}$, and $(z^{k+1})^T(q^k + Bz^{k+1}) = 0$ since z^{k+1} is a solution of SOCCP (4.2.2). We then have $(z^k - z^{k+1})^T(q^k + Bz^{k+1}) \geq 0$ and hence

$$\theta(z^k) - \theta(z^{k+1}) \geq \frac{1}{2}(z^k - z^{k+1})^T(B - C)(z^k - z^{k+1}).$$

The second inequality in (4.2.4) holds from the positive semidefiniteness of $B-C$. The last assertion of the lemma is obtained by the fact that the regularity of (B, C) implies the positive definiteness of $B - C$. ∎

The above lemma leads to the following theorem.

Theorem 4.2.1 *Let (B, C) be a regular $\mathcal{K}$-Q-splitting of the symmetric matrix M. Then, any accumulation point of the sequence $\{z^k\}$ generated by Algorithm* 4.2.1 *is a solution of SOCCP* (4.1.1).

Proof. Let $\tilde{z}$ be an arbitrary accumulation point of the sequence $\{z^k\}$ generated by Algorithm 4.2.1 and $\{z^{k_i}\}$ be a subsequence of $\{z^k\}$ converging to $\tilde{z}$. Then, by the continuity of θ, the sequence $\{\theta(z^{k_i})\}$ converges to $\theta(\tilde{z})$. In addition, the entire sequence $\{\theta(z^k)\}$ is bounded below, since $\{\theta(z^k)\}$ is nonincreasing from Lemma 4.2.1 and the subsequence $\{\theta(z^{k_i})\}$ converges. Consequently, the sequence $\{\theta(z^k)\}$ itself converges. This fact, along with the positive definiteness of $B - C$ and the inequality (4.2.4), yields that $\{z^k - z^{k+1}\}$ converges to 0. Hence, the sequence $\{z^{k_i+1}\}$ also converges to $\tilde{z}$. Since z^{k_i+1} satisfies

$$\begin{aligned} z^{k_i+1} &\in \mathcal{K} \\ Bz^{k_i+1} + Cz^{k_i} + q &\in \mathcal{K} \\ (z^{k_i+1})^T(Bz^{k_i+1} + Cz^{k_i} + q) &= 0, \end{aligned}$$

passing to the limit reveals that $\tilde{z}$ is a solution of SOCCP (4.1.1). ∎

This theorem indicates that, if a sequence $\{z^k\}$ generated by Algorithm 4.2.1 has an accumulation point, then it is a solution of SOCCP (4.1.1). However, the theorem says nothing about the existence of an accumulation point. In order to show the boundedness of $\{z^k\}$, we introduce the following concept on M.

Definition 4.2.3 *A matrix $M \in \Re^{n\times n}$ is said to be (strictly) $\mathcal{K}$-copositive if $z^T Mz \geq (>) 0$ for all $z \in \mathcal{K}\backslash\{0\}$.*

The concept of (strict) $\mathcal{K}$-copositivity is a natural extension of the (strict) copositivity in the LCP theory [24]. It is easily seen that any positive semidefinite matrix is $\mathcal{K}$-copositive and any positive definite matrix is strictly $\mathcal{K}$-copositive. Using this definition, we establish the main theorem in this section.

Theorem 4.2.2 *Let M be a symmetric matrix, and q be an arbitrary vector. If M is strictly $\mathcal{K}$-copositive, then, for any initial point $z^0 \in \mathcal{K}$, the sequence $\{z^k\}$ generated by Algorithm* 4.2.1 *with regular $\mathcal{K}$-Q-splitting (B, C) is bounded, and its arbitrary accumulation point is a solution of SOCCP* (4.1.1).

Proof. From Theorem 4.2.1, if the generated sequence has an accumulation point, it is a solution of SOCCP (4.1.1). Hence, it suffices to show the boundedness of the sequence $\{z^k\}$.

Since M is strictly $\mathcal{K}$-copositive, we have

$$\sigma := \min_{\substack{\|e\|=1 \\ e \in \mathcal{K}}} e^T M e > 0,$$

that is, we have for any $z \in \mathcal{K}$

$$z^T M z \geq \sigma \|z\|^2. \tag{4.2.5}$$

Therefore, we have for all k

$$\begin{aligned} \theta(z^0) &\geq \theta(z^k) \\ &= \frac{1}{2}(z^k)^T M z^k + q^T z^k \\ &\geq \frac{1}{2}\sigma \|z^k\|^2 - \|q\| \|z^k\| \\ &= \frac{1}{2}\sigma \Big(\|z^k\| - \frac{1}{\sigma}\|q\| \Big)^2 - \frac{1}{2\sigma}\|q\|^2, \end{aligned}$$

where the first inequality follows from Lemma 4.2.1, the second inequality follows from (4.2.5), $\{z^k\} \subset \mathcal{K}$ and the Cauchy-Schwarz inequality. This inequality readily yields

$$\|z^k\| \leq \frac{1}{\sigma}\|q\| + \sqrt{\frac{2}{\sigma}\left(\theta(z^0) + \frac{1}{2\sigma}\|q\|^2\right)}$$

for all k, which shows the boundedness of $\{z^k\}$. ∎

4.3 Block SOR method

In the previous section, we have shown that, under the assumption that M is strictly $\mathcal{K}$-copositive, a sequence generated by Algorithm 4.2.1 with any regular $\mathcal{K}$-Q-splitting (B, C) converges to a solution of SOCCP (4.1.1). In this section, we present the block successive overrelaxation (block SOR) method for solving affine SOCCPs by extending the corresponding method for LCPs [24]. In particular, we give conditions for a splitting (B, C) used in the block SOR method to be a regular $\mathcal{K}$-Q-splitting. In this and the next sections, we suppose that the following assumption holds.

Assumption A *The symmetric matrix $M \in \Re^{n\times n}$ is positive definite.*

Note that the matrix M is strictly $\mathcal{K}$-copositive under this assumption.

First, we give an explicit expression of the splitting (B, C). Let the matrix M be partitioned as

$$M = \begin{pmatrix} M_{11} & M_{12} & \cdots & M_{1m} \\ M_{21} & M_{22} & & M_{2m} \\ \vdots & & \ddots & \vdots \\ M_{m1} & M_{m2} & \cdots & M_{mm} \end{pmatrix}$$

with $M_{ij} \in \Re^{n_i \times n_j}$. Then, the splitting (B, C) used in the block SOR method is represented as

$$B = \begin{pmatrix} B_{11} & & & & 0 \\ M_{21} & B_{22} & & & \\ M_{31} & M_{32} & \ddots & & \\ \vdots & & \ddots & \ddots & \\ M_{m1} & \cdots & \cdots & M_{m,m-1} & B_{mm} \end{pmatrix}, \quad C = M - B, \tag{4.3.1}$$

where B_{ii} are chosen appropriately for $i = 1, \ldots, m$. Since B is chosen to be a block lower triangular matrix, we can solve SOCCP (4.2.2) successively as follows: Let z and q^k in SOCCP (4.2.2) be partitioned as

$$z = \begin{pmatrix} z_1 \\ \vdots \\ z_m \end{pmatrix}, \quad q^k = \begin{pmatrix} q_1^k \\ \vdots \\ q_m^k \end{pmatrix},$$

where $z_i \in \Re^{n_i}$ and $q_i^k \in \Re^{n_i}$, $i = 1, \ldots, m$. Then the decomposable structure of SOC constraints [57, Proposition 2.2] yields that SOCCP (4.2.2) is equivalent to the problem of finding $z \in \Re^n$ such that

$$z_i \in \mathcal{K}^{n_i}, \quad B_{ii}z_i + r_i^k \in \mathcal{K}^{n_i}, \quad z_i^T(B_{ii}z_i + r_i^k) = 0 \quad i = 1, \ldots, m, \tag{4.3.2}$$

where

$$r_i^k := \begin{cases} q_1^k & \text{if } i = 1, \\ \sum_{j=1}^{i-1} M_{ij}z_j + q_i^k & \text{if } i \geq 2\ . \end{cases}$$

We can solve problems (4.3.2) for z_i recursively from $i = 1$ to $i = m$, by regarding $z_1, \ldots, z_{i-1}$ and r_i^k as known constants.

In the block SOR method for LCP, the block diagonal elements B_{ii} are normally chosen as $B_{ii} := \omega^{-1}M_{ii}$ with a constant $\omega \in (0, 2)$. However, in the case of SOCCP, subproblems (4.3.2) may not be solved efficiently if $B_{ii} = \omega^{-1}M_{ii}$. Here, we propose to choose B_{ii} as follows: To simplify the notation, we introduce the function $\Gamma : \Re^{\ell\times\ell} \times (0, +\infty) \times [0, +\infty) \to \Re^{\ell\times\ell}$ defined by

$$\Gamma(A, \omega, \gamma) := \begin{cases} \omega^{-1}a_1 & (l = 1) \\ \begin{pmatrix} \omega^{-1}a_1 & 0^T \\ \gamma a_2 & \omega^{-1}A_3 \end{pmatrix} & (l \geq 2)\,, \end{cases} \tag{4.3.3}$$

where $\omega > 0$, $\gamma \geq 0$, and $A \in \Re^{\ell\times\ell}$ is given by

$$A = \begin{pmatrix} a_1 & a_2^T \\ a_2 & A_3 \end{pmatrix} \tag{4.3.4}$$

with $a_1 \in \Re$, $a_2 \in \Re^{\ell-1}$ and $A_3 \in \Re^{(\ell-1)\times(\ell-1)}$. Using this function, we let

$$B_{ii} := \Gamma(M_{ii}, \omega, \gamma).$$

SOCCP (4.3.2) can be solved efficiently by exploiting the particular structure of $B_{ii} = \Gamma(M_{ii}, \omega, \gamma)$, as will be shown in the next section. Our method involves two parameters, and hence may be called a modified block Gauss-Seidel method with diagonal elements defined by AOR-like splitting [51]. Nevertheless, we emphasize that our method is a natural extension of block SOR method for NCP to SOCCP.

Next we consider conditions for the splitting (4.3.1) with $B_{ii} = \Gamma(M_{ii}, \omega, \gamma)$ to be a regular $\mathcal{K}$-Q-splitting. To this end, we give a lemma and a proposition in the following.

Lemma 4.3.1 *Let $A \in \Re^{\ell\times\ell}$ be a positive definite symmetric matrix given by* (4.3.4) *and (G, H) be a splitting of A given by*

$$G \,=\, \Gamma(A, \omega, \gamma), \quad H \,=\, A - \Gamma(A, \omega, \gamma).$$

Suppose that ω and γ satisfy either of the following conditions:

(a) $\gamma > 1$ *and* $0 < \omega \leq 2/\gamma$,
(b) $\gamma = 1$ *and* $0 < \omega < 2$,
(c) $0 \leq \gamma < 1$ *and* $0 < \omega \leq 2/(2-\gamma)$.

Then the matrix G is positive definite and the splitting (G, H) is regular.

Proof. Since the lemma holds evidently for $\ell = 1$, we only consider the case $\ell \geq 2$. Note that, for any symmetric matrix $A \in \Re^{\ell\times\ell}$ given by (4.3.4), the following relation holds [59, Theorem 7.7.6] :

$$A \text{ is positive definite } \iff a_1 > 0 \text{ and } A_3 - a_1^{-1}a_2a_2^T \text{ is positive definite,} \tag{4.3.5}$$

where the matrix $A_3 - a_1^{-1}a_2a_2^T$ is called the Schur complement of A with respect to a_1.

We first show the positive definiteness of G by showing the positive definiteness of

$$(G + G^T)/2 = \begin{pmatrix} \omega^{-1}a_1 & (\gamma/2)a_2^T \\ (\gamma/2)a_2 & \omega^{-1}A_3 \end{pmatrix}.$$

Since it holds evidently when $\gamma = 0$, we only consider the case where $\gamma > 0$. First we note that

$$\frac{4}{\omega^2\gamma^2} - 1 \,>\, 0 \tag{4.3.6}$$

for all cases (a), (b) and (c). The Schur complement of $(G+G^T)/2$ with respect to $\omega^{-1}a_1$ is written as

$$\omega^{-1}A_3 - \frac{(\gamma^2/4)a_2a_2^T}{\omega^{-1}a_1} = \frac{\omega\gamma^2}{4}\left\{\left(\frac{4}{\omega^2\gamma^2}-1\right)A_3 + \left(A_3 - \frac{a_2a_2^T}{a_1}\right)\right\}, \tag{4.3.7}$$

which is positive definite from (4.3.5) and (4.3.6). This together with $\omega^{-1}a_1 > 0$ implies the positive definiteness of $(G+G^T)/2$.

We next show the regularity of (G,H) by showing the positive definiteness of

$$\frac{1}{2}\left\{(G-H)+(G-H)^T\right\} = \begin{pmatrix} (2\omega^{-1}-1)a_1 & (\gamma-1)a_2^T \\ (\gamma-1)a_2 & (2\omega^{-1}-1)A_3 \end{pmatrix}.$$

Note that the Schur complement of $((G-H)+(G-H)^T)/2$ with respect to $(2\omega^{-1}-1)a_1$ is written as

$$\begin{aligned}&(2\omega^{-1}-1)A_3 - \frac{(\gamma-1)^2a_2a_2^T}{(2\omega^{-1}-1)a_1} \\ &\quad = \frac{1}{2\omega^{-1}-1}\left[\left\{(2\omega^{-1}-1)^2-(\gamma-1)^2\right\}A_3 + (\gamma-1)^2\left(A_3 - \frac{a_2a_2^T}{a_1}\right)\right].\end{aligned} \tag{4.3.8}$$

Moreover, A_3 and $A_3 - a_1^{-1}a_2a_2^T$ are positive definite, and $\omega^{-1}-1>0$ and $(2\omega^{-1}-1)^2-(\gamma-1)^2>0$ for all cases (a), (b) and (c). Hence, the matrix given by (4.3.8) is positive definite. This together with $(2\omega^{-1}-1)a_1>0$ implies the positive definiteness of $((G-H)+(G-H)^T)/2$. ∎

Proposition 4.3.1 *Suppose that $G \in \Re^{\ell\times\ell}$ is a positive definite matrix. Then, G is a $\mathcal{K}^\ell$-Q-matrix, and the solution of SOCCP$(p,G,\mathcal{K}^\ell)$ is unique for any $p \in \Re^\ell$.*

Proof. Let us define function F by $F(x) := Gx + p$. Then, F is strongly monotone since G is positive definite. Moreover, we note that SOCCP $(p,G,\mathcal{K}^\ell)$ is equivalent to the following variational inequality problem:

$$\begin{aligned} &\text{Find} \quad x \in \mathcal{K}^\ell \\ &\text{such that} \quad F(x)^T(y-x) \geq 0 \quad \forall y \in \mathcal{K}^\ell. \end{aligned}$$

Since any variational inequality problem with strongly monotone function has a unique solution [52, Corollary 3.2], we obtain the desired result. ∎

Using the above lemma and proposition, we give conditions for (4.3.1) with $B_{ii} = \Gamma(M_{ii},\omega,\gamma)$ to be a regular $\mathcal{K}$-Q-splitting.

Theorem 4.3.1 *Let the splitting (B,C) of M be given by (4.3.1) with $B_{ii} = \Gamma(M_{ii},\omega,\gamma)$, $i = 1,\ldots,m$. Suppose that ω and γ satisfy either of the following conditions:*

(a) $\gamma > 1$ *and* $0 < \omega \leq 2/\gamma$,

(b) $\gamma = 1$ *and* $0 < \omega < 2$,

(c) $0 \leq \gamma < 1$ *and* $0 < \omega \leq 2/(2-\gamma)$.

Then (B, C) is a regular $\mathcal{K}$-Q-splitting.

Proof. Let (B_{ii}, C_{ii}) be a splitting of M_{ii} where $B_{ii} = \Gamma(M_{ii}, \omega, \gamma)$ and $C_{ii} = M_{ii} - B_{ii}$. By setting $A := M_{ii}$, $G := B_{ii} = \Gamma(M_{ii}, \omega, \gamma)$ and $H := C_{ii} = M_{ii} - \Gamma(M_{ii}, \omega, \gamma)$ in Lemma 4.3.1, we obtain the positive definiteness of B_{ii} and the regularity of the splitting (B_{ii}, C_{ii}).

Using these results, we show that (B, C) is a regular $\mathcal{K}$-Q-splitting. We first show the regularity of (B, C). Since M is symmetric and B is block lower triangular, $B - C$ can be written as

$$B - C = \mathrm{diag}\,\{B_{ii} - C_{ii}\}_{i=1}^{m} + L - L^T,$$

where $\mathrm{diag}\,\{B_{ii} - C_{ii}\}_{i=1}^{m}$ denotes a block diagonal matrix whose diagonal elements are $B_{ii} - C_{ii}$, and $L \in \Re^{n \times n}$ is the strictly block lower triangular part of M. Then, we have for any $z \in \Re^n \backslash \{0\}$

$$\begin{aligned} z^T(B - C)z &= z^T(\mathrm{diag}\,\{B_{ii} - C_{ii}\}_{i=1}^{m})z + z^T L z - z^T L^T z \\ &= \sum_{i=1}^{m} {z_i}^T (B_{ii} - C_{ii}) z_i \\ &> 0, \end{aligned}$$

where the inequality follows from the regularity of (B_{ii}, C_{ii}). Hence, the splitting (B, C) is regular. We next show that B is a $\mathcal{K}$-Q-matrix. Since B_{ii} is positive definite, we have from Proposition 4.3.1 that the matrix B_{ii} is also a $\mathcal{K}^{n_i}$-Q-matrix. Note, moreover, that the solution of SOCCP (4.2.2) is obtained by solving SOCCP (4.3.2) recursively from $i = 1$ to $i = m$, and that each SOCCP is solvable since B_{ii} is a $\mathcal{K}^{n_i}$-Q-matrix. Hence, the whole SOCCP $(B, q, \mathcal{K})$ is solvable for any q, that is, B is a $\mathcal{K}$-Q-matrix.

■

4.4 Solving subproblems

In the previous section, we have shown that SOCCP (4.2.2) can be decomposed into m subproblems (4.3.2) by choosing the splitting (B, C) as in (4.3.1). We have also derived conditions for Algorithm 4.2.1 with the splitting (4.3.1) to be convergent. In this section, we propose a method for solving these m subproblems (4.3.2) efficiently. In order to simplify the notation, we consider the following SOCCP in which superscripts and subscripts are omitted :

$$\begin{aligned} \text{Find} \quad & z \in \Re^{\ell} \\ \text{such that} \quad & z \in \mathcal{K}^{\ell},\ Bz + r \in \mathcal{K}^{\ell},\ z^T(Bz + r) = 0, \end{aligned} \tag{4.4.1}$$

where B is a positive definite matrix of the form

$$B = \begin{pmatrix} b_1 & 0^T \\ b_2 & B_3 \end{pmatrix} \tag{4.4.2}$$

with $b_1 \in \Re$, $b_2 \in \Re^{\ell-1}$ and a symmetric matrix $B_3 \in \Re^{(\ell-1)\times(\ell-1)}$. Notice that each subproblem (4.3.2) reduces to SOCCP (4.4.1) when the parameters ω and γ used in the splitting (4.3.1) with

$B_{ii} = \Gamma(M_{ii}, \omega, \gamma)$ satisfy the assumptions of Theorem 4.3.1. In addition, SOCCP (4.4.1) has a unique solution from the positive definiteness of B and Proposition 4.3.1. When $\ell = 1$, we can easily obtain the solution of (4.4.1) as $z = \max(0, -r/B)$. So we will consider the case $\ell \geq 2$.

The following three cases are possible for a solution z^* of SOCCP (4.4.1):

(i) $z^* = 0$,

(ii) $z^* \in \operatorname{int} \mathcal{K}^\ell$,

(iii) $z^* \in \operatorname{bd} \mathcal{K}^\ell \setminus \{0\}$,

where $\operatorname{int} \mathcal{K}^\ell$ and $\operatorname{bd} \mathcal{K}^\ell$ denote the interior and the boundary of $\mathcal{K}^\ell$, respectively. Since $z^* \in \mathcal{K}^\ell$, it is clear that no other cases are possible for a solution of (4.4.1). To solve SOCCP (4.4.1) efficiently, it will be helpful to detect which case applies to the solution z^*. To this end, we provide the following proposition.

Proposition 4.4.1 *Let z^* be the unique solution of SOCCP (4.4.1). Then,*

(a) *case* (i) *holds if and only if $r \in \mathcal{K}^\ell$;*

(b) *case* (ii) *holds if and only if $-B^{-1}r \in \operatorname{int} \mathcal{K}^\ell$. Moreover, we have $z^* = -B^{-1}r$;*

(c) *case* (iii) *holds if and only if $r \notin \mathcal{K}^\ell$ and $-B^{-1}r \notin \operatorname{int} \mathcal{K}^\ell$.*

Proof. We first show (a). If $z^* = 0$ solves SOCCP (4.4.1), then we have $Bz^* + r = r \in \mathcal{K}^\ell$. Conversely, if $r \in \mathcal{K}^\ell$, it is easily seen that $z^* = 0$ solves SOCCP (4.4.1). We next show (b). If case (ii) holds, then we have $Bz^* + r = 0$, which implies $z^* = B^{-1}r$. Conversely, if $-B^{-1}r \in \operatorname{int} \mathcal{K}^\ell$, it is easily seen that $z^* = -B^{-1}r$ solves SOCCP (4.4.1). We must have (c) from the existence and the uniqueness of the solution. This completes the proof. ∎

This proposition indicates that, if $r \in \mathcal{K}^\ell$ or $-B^{-1}r \in \operatorname{int} \mathcal{K}^\ell$, then we can readily calculate the solution of SOCCP (4.4.1).

Now, we describe a method of finding the solution of SOCCP (4.4.1) when case (iii) holds. Note that we have $Bz^* + r \in \operatorname{bd} \mathcal{K}^\ell$ since the solution z^* belongs to $\operatorname{bd} \mathcal{K}^\ell \setminus \{0\}$ and the inner product of z^* and $Bz^* + r$ is equal to 0. Thus, we can write z^* and $Bz^* + r$ as

$$z^* = \lambda \begin{pmatrix} 1 \\ \tilde{w} \end{pmatrix}, \tag{4.4.3}$$

$$Bz^* + r = \mu \begin{pmatrix} 1 \\ -\tilde{w} \end{pmatrix}, \tag{4.4.4}$$

where $\lambda > 0$, $\mu \geq 0$ and $\tilde{w}$ is an $(\ell - 1)$-dimensional vector such that $\|\tilde{w}\| = 1$. Generally, it is not easy to find z^* satisfying (4.4.3) and (4.4.4) simultaneously. Nevertheless, when B is given by (4.4.2), we can exploit the special structure of B to compute the solution z^* as follows: Substituting (4.4.2) and (4.4.3) into (4.4.4), we have

$$\begin{pmatrix} \lambda b_1 + r_1 \\ \lambda b_2 + \lambda B_3 \tilde{w} + r_2 \end{pmatrix} = \begin{pmatrix} \mu \\ -\mu \tilde{w} \end{pmatrix}, \tag{4.4.5}$$

where $r = (r_1, r_2) \in \Re \times \Re^{\ell-1}$. Eliminating μ in (4.4.5), we have

$$-\{\lambda(b_1 I + B_3) + r_1 I\}\tilde{w} = \lambda b_2 + r_2. \tag{4.4.6}$$

Moreover, $\lambda b_1 + r_1 = \mu$ together with $\lambda > 0$, $\mu \geq 0$ and $b_1 > 0$ from the positive definiteness of B yields

$$\lambda \geq \lambda_L := \max\{0, -r_1/b_1\}.$$

Let us define function $H : \Re \to \Re^{(\ell-1)\times(\ell-1)}$ by

$$H(\lambda) := \lambda(b_1 I + B_3) + r_1 I. \tag{4.4.7}$$

If $H(\lambda)$ is nonsingular, then (4.4.6) can be rewritten as

$$\tilde{w} = -H(\lambda)^{-1}(\lambda b_2 + r_2). \tag{4.4.8}$$

Thus, if we find a λ^* such that $\|H(\lambda^*)^{-1}(\lambda^* b_2 + r_2)\| = 1$ and $\lambda^* \geq \lambda_L$, then we obtain z^* by (4.4.3) and (4.4.8).

If $r_1 = 0$, then the equation $\|H(\lambda)^{-1}(\lambda b_2 + r_2)\| = 1$, together with $H(\lambda) = \lambda(b_1 I + B_3)$, yields the single variable quadratic equation $\lambda^2(1 - \|g\|^2) - 2\lambda g^T h - \|h\|^2 = 0$, where $g := (b_1 I + B_3)^{-1} b_2$ and $h := (b_1 I + B_3)^{-1} r_2$. Since $\lambda^* \geq \lambda_L = 0$ and $\|g\| < 1$ from the following lemma, λ^* is given by

$$\lambda^* = \frac{-g^T h + \sqrt{(g^T h)^2 + \|h\|^2(1 - \|g\|^2)}}{1 - \|g\|^2}. \tag{4.4.9}$$

Lemma 4.4.1 *Let B be an arbitrary positive definite matrix given by* (4.4.2)*, and $g := (b_1 I + B_3)^{-1} b_2$. Then, we have $\|g\| < 1$.*

Proof. Let $v := (1, -g^T)^T$. Then we have

$$\begin{aligned} v^T B v &= b_1 - g^T b_2 + g^T B_3 g \\ &= b_1 - g^T(b_1 I + B_3)g + g^T B_3 g \\ &= b_1(1 - \|g\|^2). \end{aligned}$$

Since $v \neq 0$ and B is positive definite, we have $b_1 > 0$ and $b_1(1 - \|g\|^2) > 0$, which imply $\|g\| < 1$. ■

Since z^* can be easily obtained by (4.4.9), (4.4.8) and (4.4.3) when $r_1 = 0$, we suppose $r_1 \neq 0$ in the subsequent discussions. The following proposition gives a sufficient condition for $H(\lambda)$ to be nonsingular.

Proposition 4.4.2 *If $r_1 \neq 0$, then the matrix $H(\lambda)$ defined by* (4.4.7) *is positive definite for any $\lambda \geq \lambda_L$.*

Proof. Note that $b_1 > 0$ and B_3 is positive definite since B is positive definite. If $r_1 > 0$, then $\lambda_L = 0$ and hence $H(\lambda) = \lambda(b_1 I + B_3) + r_1 I$ is positive definite for any $\lambda \geq \lambda_L$. If $r_1 < 0$, then $\lambda_L = -r_1/b_1 > 0$ and hence $H(\lambda) = \lambda B_3 + b_1(\lambda + r_1/b_1)I$ is positive definite for any $\lambda \geq \lambda_L$. ∎

Now, let us define functions $w : [\lambda_L, +\infty) \to \Re^{\ell-1}$ and $\psi : [\lambda_L, +\infty) \to \Re$ by

$$w(\lambda) := -H(\lambda)^{-1}(\lambda b_2 + r_2), \tag{4.4.10}$$

$$\psi(\lambda) := \|w(\lambda)\|. \tag{4.4.11}$$

Then, our purpose is to find a solution $\lambda^* \geq \lambda_L$ of the following single variable equation :

$$\psi(\lambda) = 1. \tag{4.4.12}$$

First we confirm that such a solution always exists.

Proposition 4.4.3 *Suppose that case* (iii) *holds and* $r_1 \neq 0$. *Then we have* (a) $\psi(\lambda_L) \geq 1$ *and* (b) $\lim_{\lambda \to +\infty} \psi(\lambda) = \|(b_1 I + B_3)^{-1} b_2\| < 1$.

Proof. Since (b) can be easily obtained from (4.4.10) and Lemma 4.4.1, we only show (a). Assume to contrary that $\psi(\lambda_L) < 1$. Then, it suffices to show that either case (i) or (ii) holds, since the solution exists uniquely. When $r_1 > 0$, we have from $w(\lambda_L) = w(0) = -r_1^{-1} r_2$ that $1 > \psi(\lambda_L) = \|w(0)\| = \|r_2\|/r_1$. This implies $r \in \operatorname{int} \mathcal{K}^\ell \subset \mathcal{K}^\ell$, that is, case (i) holds. When $r_1 < 0$, (4.4.10) and (4.4.11), together with $b_1 > 0$, $r_1 < 0$ and $\lambda_L = -r_1 b_1^{-1}$, yield

$$\psi(\lambda_L) = \frac{\| - B_3^{-1}(r_2 - r_1 b_1^{-1} b_2)\|}{-r_1 b_1^{-1}} < 1,$$

which implies

$$\begin{pmatrix} -r_1 b_1^{-1} \\ -B_3^{-1}(r_2 - r_1 b_1^{-1} b_2) \end{pmatrix} \in \operatorname{int} \mathcal{K}^\ell. \tag{4.4.13}$$

The vector in (4.4.13) equals $-B^{-1} r$ since the following equality holds identically :

$$\begin{pmatrix} b_1 & 0^T \\ b_2 & B_3 \end{pmatrix} \begin{pmatrix} -r_1 b_1^{-1} \\ -B_3^{-1}(r_2 - r_1 b_1^{-1} b_2) \end{pmatrix} = -\begin{pmatrix} r_1 \\ r_2 \end{pmatrix}.$$

Hence, we have $-B^{-1} r \in \operatorname{int} \mathcal{K}^\ell$, that is, case (ii) holds. ∎

Since function ψ is continuous for $\lambda \geq \lambda_L$, Proposition 4.4.3 guarantees the existence of $\lambda^* \geq \lambda_L$ satisfying (4.4.12). Actually, such a λ^* must exist uniquely since λ^* gives a solution of SOCCP (4.4.1), which exists uniquely from the positive definiteness of B and Proposition 4.3.1.

Now, we present a procedure for finding a solution $\lambda^* \geq \lambda_L$ of equation (4.4.12), which is a Newton type method incorporating the bisection method as a safeguard strategy. In particular, instead of applying Newton's method to (4.4.12) directly, we will adopt a more efficient method that is reminiscent of an approach well-known in the trust region literature [22, Chapter 7].

Define function $\phi : [\lambda_L, +\infty) \to (-1, +\infty]$ by

$$\phi(\lambda) = \begin{cases} \psi(\lambda)^{-1} - 1 & \psi(\lambda) > 0 \\ +\infty & \text{if } \psi(\lambda) = 0. \end{cases}$$

Then, the nonlinear equation

$$\phi(\lambda) = 0 \tag{4.4.14}$$

is equivalent to (4.4.12). We aim to solve (4.4.14) instead of (4.4.12) since function ϕ is expected to behave better than ψ [22].

The iterative formula of Newton's method for the nonlinear equation (4.4.14) is given by

$$\lambda_{j+1} = \lambda_j - \frac{\phi(\lambda_j)}{\phi'(\lambda_j)}, \tag{4.4.15}$$

where ϕ' denotes the derivative of ϕ. The derivative ϕ' is obtained by

$$\begin{aligned} \phi'(\lambda) &= \left\{ \frac{1}{\|w(\lambda)\|} - 1 \right\}' \\ &= -\frac{w(\lambda)^T w'(\lambda)}{\|w(\lambda)\|^3} \\ &= \frac{w(\lambda)^T H(\lambda)^{-1} \{(b_1 I + B_3) w(\lambda) + b_2\}}{\|w(\lambda)\|^3}, \end{aligned} \tag{4.4.16}$$

where the last equality follows from

$$(b_1 I + B_3) w(\lambda) + H(\lambda) w'(\lambda) = -b_2,$$

which is obtained by differentiating both sides of $H(\lambda)w(\lambda) = -(\lambda b_2 + r_1)$. Since $H(\lambda)$ is positive definite for any $\lambda \geq \lambda_L$, by using the Cholesky factorization $H(\lambda) = R(\lambda)R(\lambda)^T$ with $R(\lambda)$ being upper triangular [47], we may rewrite the formula (4.4.15) as follows:

$$\begin{aligned} \lambda_{j+1} &= \lambda_j - \left(\frac{1}{\|w(\lambda_j)\|} - 1 \right) \frac{\|w(\lambda_j)\|^3}{w(\lambda_j)^T (R(\lambda_j)^T)^{-1} R(\lambda_j)^{-1} \{(b_1 I + B_3) w(\lambda_j) + b_2\}} \\ &= \lambda_j + (\|w(\lambda_j)\| - 1) \frac{\|w(\lambda_j)\|^2}{u(\lambda_j)^T v(\lambda_j)}, \end{aligned}$$

where $u(\lambda) = R(\lambda)^{-1} w(\lambda)$ and $v(\lambda) = R(\lambda)^{-1} \{(b_1 I + B_3) w(\lambda) + b_2\}$. Summarizing the above arguments, we have the following procedure for solving SOCCP (4.4.1).

Procedure 4.4.1

Step 1. *If* $r \in \mathcal{K}^\ell$ (case (i)), *set* $z^* := 0$ *and terminate.*

Step 2. *If* $-B^{-1}r \in \operatorname{int} \mathcal{K}^\ell$ (case (ii)), *set* $z^* := -B^{-1}r$ *and terminate.*

Step 3. *Otherwise* (case (iii)), *calculate* z^* *as follows*:

Step 3-0. *If* $r_1 \neq 0$*, then go to Step* 3-1*. Otherwise, calculate vectors* g *and* h *such that* $(b_1 I + B_3)g = b_2$ *and* $(b_1 I + B_3)h = r_2$*. Set* $\lambda^* := (-g^T h + ((g^T h)^2 + \|h\|^2(1 - \|g\|^2))^{1/2})/(1 - \|g\|^2)$ *and* $z^* := \lambda^*(1, w(\lambda^*)^T)^T$*. Terminate.*

Step 3-1. *Let* $\lambda_0 := \lambda_L$*,* $\alpha_0 := \lambda_L$ *and* β_0 *be a scalar such that* $\beta_0 > \lambda_L$ *and* $\|w(\beta_0)\| < 1$*. Set* $j := 0$*.*

Step 3-2. *Factorize* $H(\lambda_j) = RR^T$*. Let* $R^j := R$*.*

Step 3-3. *Calculate* w^j *such that* $R^j(R^j)^T w^j = -(\lambda_j b_2 + r_2)$*.*

Step 3-4. *If* $\|w^j\| = 1$*, then set* $z^* := \lambda_j(1, (w^j)^T)^T$ *and terminate. If* $\|w^j\| > 1$*, then let* $\alpha_{j+1} := \lambda_j$ *and* $\beta_{j+1} := \beta_j$*. If* $\|w^j\| < 1$*, then let* $\alpha_{j+1} := \alpha_j$ *and* $\beta_{j+1} := \lambda_j$*.*

Step 3-5. *Calculate* u^j *such that* $R^j u^j = w^j$*.*

Step 3-6. *Calculate* v^j *such that* $R^j v^j = (b_1 I + B_3)w^j + b_2$*.*

Step 3-7. *Let* $\tilde{\lambda}_{j+1} := \lambda_j + (\|w^j\| - 1)\|w^j\|^2/((u^j)^T v^j)$*. If* $\tilde{\lambda}_{j+1} \in (\alpha_{j+1}, \beta_{j+1})$*, then let* $\lambda_{j+1} := \tilde{\lambda}_{j+1}$*. Otherwise, let* $\lambda_{j+1} := (\alpha_{j+1} + \beta_{j+1})/2$*. Set* $j := j + 1$ *and go back to* Step 3-2*.*

This procedure is used in Step 2 of Algorithm 4.2.1 to solve subproblem (4.3.2). In practice, we may set $\beta_0 := +\infty$, and then compute $\beta_j < +\infty$ such that $\|w(\beta_j)\| < 1$ only when it becomes necessary for the first time.

Since a bisection-type safeguard strategy is combined with Newton's method, Procedure 4.4.1 is guaranteed to be globally convergent. Note that, when the parameter γ is chosen to be 0 in the splitting (4.3.1) with $B_{ii} = \Gamma(M_{ii}, \omega, \gamma)$, SOCCP (4.3.2) reduces to SOCCP (4.4.1) with $b_2 = 0$, and the convergence is ensured even without using the safeguard strategy. In fact, since $H(\lambda)^{-1}(b_1 I + B_3) = \lambda I + r_1(b_1 I + B_3)^{-1}$ is positive definite for any $\lambda \geq \lambda_L$, it follows from (4.4.16) with $b_2 = 0$ that $\phi'(\lambda) > 0$, and hence, function ϕ is monotonically increasing on the interval $[\lambda_L, +\infty)$. We can further show that ϕ is concave on the interval $[\lambda_L, +\infty)$ by calculating the second derivative ϕ'' of ϕ. Thus, starting from $\lambda_0 := \lambda_L$, Newton's method generates a monotonically increasing sequence $\{\lambda_j\}$ converging to a solution λ^*. Furthermore, since ϕ is almost linear for $\lambda \geq \lambda_L$, we may expect that Newton's method converges very rapidly [22].

On the other hand, when $\gamma \neq 0$, convergence is not guaranteed unless a safeguard strategy is employed, since $b_2 \neq 0$ in (4.4.16) and hence ϕ is generally neither monotonically increasing nor concave on the interval $[\lambda_L, +\infty)$. Nevertheless, it is worth mentioning that, as is shown in Theorem 4.3.1, it is allowed to choose an ω such that $\omega > 1$ when $0 < \gamma < 2$, whereas ω must be in $(0, 1]$ when $\gamma = 0$. In view of the fact that convergence of SOR method for linear complementarity problems is accelerated by letting $\omega > 1$ [29], we may expect that Algorithm 4.2.1 performs better when ω is chosen greater than 1.

SOCCP (4.4.1) may also be solved by Newton-type methods like interior point methods. However, Procedure 4.4.1 exploits the special structure of matrix B, and hence, rapid convergence can be expected. Indeed, the numerical results reported in Section 4.5 show that Procedure 4.4.1 finds a solution within several iterations.

4.5 Numerical results

In this section, we present some numerical results with the proposed algorithm. The program was coded in MATLAB 6.5.0 and run on a computer with 3.04GHz CPU and 2GB memory. We have conducted the following experiments:

(A) Testing Procedure 4.4.1 on SOCCP (4.4.1) of various problem sizes.

(B) Testing Procedure 4.4.1 on SOCCP (4.4.1) with various degrees of sparsity.

(C) Testing Algorithm 4.2.1 on SOCCP (4.1.1) with various values of γ and ω.

(D) Testing Algorithm 4.2.1 on SOCCP (4.1.1) with various Cartesian structures of $\mathcal{K}$.

In experiments (A), (B) and (D), we used not only complete Cholesky factorization but also incomplete Cholesky factorization in Step 3-2 of Procedure 4.4.1. The complete Cholesky factorization provides upper triangular R satisfying $H(\lambda) = RR^T$ exactly. But, even if matrix $H(\lambda)$ is sparse, R is not sparse in general. On the other hand, the incomplete Cholesky factorization provides upper triangular R such that $H(\lambda) \approx RR^T$. Therefore, $\|w^j\| = \| - (R^j(R^j)^T)^{-1}(\lambda_j b_2 + r_2)\| = 1$ does not necessarily imply $\|w(\lambda_j)\| = \| - H(\lambda_j)^{-1}(\lambda_j b_2 + r_2)\| = 1$, and the point z^* obtained by Procedure 4.4.1 may only be an approximate solution of SOCCP (4.4.1). However, since the incomplete Cholesky factorization exploits the sparsity of a matrix, computational cost can be reduced drastically when $H(\lambda)$ is sparse. In addition, our computational experiments has revealed that, even though subproblems (4.2.2) are solved only approximately, Algorithm 4.2.1 is able to find a solution of SOCCP (4.1.1) in most cases.

In experiment (A), we generated 100 test problems for each $\ell = 100, 200, \ldots, 1000$ and solved each problem by Procedure 4.4.1 with complete Cholesky factorization and incomplete Cholesky factorization. The termination criterion was $|\|w^j\| - 1| < 10^{-4}$. In generating a test problem, elements of vector r were chosen randomly from the interval $[-1, 1]$, and a positive definite matrix B of the form (4.4.2) was obtained by the following procedure: First, set $A = NN^T + D$, where N is a square matrix whose nonzero elements are chosen randomly from the interval $[-1, 1]$, and D is a diagonal matrix whose diagonal elements are chosen randomly from $[0.01, 1]$. Then, let $B := \Gamma(A, 1, 2)$, where Γ is defined by (4.3.3). In this procedure, the number of nonzero elements of N is determined so that the nonzero density of matrix B becomes approximately 5%. We show the results in Table 4.1, where ℓ denotes the number of variables, $\sharp$iter denotes the number of iterations, and cpu(s) denotes the CPU time in second. The left column labeled comChol corresponds to the case where the complete Cholesky factorization was employed, while the right column labeled incChol corresponds to the case where the incomplete Cholesky factorization was employed. In particular, the number of iterations and the CPU time are the averages of 100 trials for each ℓ. We may observe that the number of iterations stays almost constant regardless of the problem size, although the CPU time grows with the problem size.

In experiment (B), we generated 100 test problems for each nonzero density 0.2%, 0.5%, 1%, 2%, 5% and 10%, where the problem size ℓ was fixed to 1000. The termination criterion and the

Table 4.1: Results for Procedure 4.4.1 applied to SOCCPs (4.4.1) of various problem sizes

ℓ	comChol		incChol	
	♯iter	cpu(s)	♯iter	cpu(s)
100	4.40	0.008	4.53	0.006
200	4.70	0.048	4.84	0.022
300	4.96	0.179	5.12	0.057
400	4.99	0.436	5.33	0.125
500	5.09	0.872	5.28	0.232
600	5.25	1.707	5.51	0.407
700	5.18	2.555	5.36	0.619
800	5.29	3.809	5.36	0.932
900	5.26	5.362	5.48	1.343
1000	5.31	7.264	5.65	1.875

procedure for generating B and r were similar to experiment (A). We show the results in Table 4.2, where dens denotes the (approximate) nonzero density of matrix B, ♯iter denotes the number of iterations, and cpu(s) denotes the CPU time in second. Each number is the average of 100 trials for each nonzero density. As shown in Table 4.2, the sparsity of matrix B affects the number of iterations only slightly, but reduces the computational cost drastically.

Table 4.2: Results for Procedure 4.4.1 applied to SOCCPs (4.4.1) with various degrees of sparsity

dens	comChol		incChol	
	♯iter	cpu(s)	♯iter	cpu(s)
0.2%	3.89	0.024	4.01	0.055
0.5%	4.54	0.561	4.47	0.118
1%	4.83	2.617	4.85	0.277
2%	5.12	5.085	5.12	0.682
5%	5.32	7.531	5.52	1.858
10%	5.43	8.952	5.80	3.533

In experiment (C), we solved SOCCP (4.1.1) with various values of parameters γ and ω used to determine the splitting (B, C), by incorporating a block SOR method and Procedure 4.4.1 into Algorithm 4.2.1. Specifically, we tried all possible combinations of $(\omega, \gamma) \in \{0.1, 0.2, \ldots, 1.9\} \times \{0, 0.5, 1.0, 1.5, 2.0\}$. The underlying SOC $\mathcal{K}$ was fixed to be $\mathcal{K}^{20} \times \cdots \times \mathcal{K}^{20} \subset \Re^{400}$. We solved different 100 test problems for each pair (ω, γ) with data (M, q) randomly generated as follows: Elements of vector q were chosen from the interval $[-1, 1]$, and a symmetric positive definite matrix M, whose nonzero density was controlled to be approximately 1%, was given by $M = NN^T + D$, where N and D are matrices obtained in a way similar to experiment (A). The termination

criterion was $\|z^{k+1} - z^k\| < 10^{-8}$, and the complete Cholesky factorization was used in Step 3-2 of Procedure 4.4.1. We show the results in Table 4.3. The numbers in the table show the average numbers of iterations taken over the successful trials for each pair (ω, γ). We see that the algorithm converges most rapidly when $\omega = 1.1$ for every γ. Note that the combinations of (ω, γ) marked with parentheses in the table do not satisfy the assumptions of Theorem 4.3.1, and hence, the convergence is not guaranteed for such (ω, γ). In fact, the numbers in the parentheses show how often the algorithm with such (ω, γ) failed to converge. Nevertheless, we were able to obtain the solution in most instances even if the convergence of the algorithm was not assured theoretically. The results suggest that it is possible to weaken the assumptions of Theorem 4.3.1.

Table 4.3: Results for Algorithm 4.2.1 applied to SOCCPs (4.1.1) with various choices of (ω, γ)

$\omega\backslash\gamma$	0	0.5	1.0	1.5	2.0
0.1	321.61	323.26	335.16	323.47	336.67
0.2	164.24	169.03	160.09	161.46	164.66
0.3	104.50	104.59	106.34	104.08	104.13
0.4	76.63	77.89	76.73	78.18	78.40
0.5	59.71	57.71	59.35	59.51	59.88
0.6	47.22	45.77	46.62	45.41	45.34
0.7	38.54	37.13	38.10	37.40	35.84
0.8	30.02	30.17	29.02	28.35	29.19
0.9	25.95	24.30	23.68	23.00	22.98
1.0	20.32	19.54	18.30	18.73	19.72
1.1	17.88(0)	17.65	16.97	17.25	18.33(0)
1.2	18.75(0)	18.45	18.41	19.07	20.31(0)
1.3	21.30(0)	20.92	21.21	21.47	22.62(0)
1.4	25.38(0)	24.38(0)	24.43	25.18(0)	26.34(2)
1.5	31.39(0)	30.34(0)	29.53	31.70(0)	31.19(2)
1.6	41.78(2)	37.93(0)	37.81	39.51(0)	39.82(0)
1.7	53.98(0)	51.93(0)	51.99	54.82(1)	55.77(6)
1.8	81.59(2)	77.21(0)	73.64	75.50(1)	79.71(6)
1.9	135.04(2)	147.93(1)	128.84	128.33(2)	152.04(14)

In experiment (D), we solved SOCCP (4.1.1) with various Cartesian structures of $\mathcal{K}$. To construct SOCs of various types, we chose n_i and m such that $n_1 + \cdots + n_m = 1600$ and $n_1 = \cdots = n_m$, where m is the number of SOCs comprising $\mathcal{K}$ and n_i is the dimension of each SOC. For each type of $\mathcal{K}$, we solved 100 test problems, where M and q were generated in a way similar to experiment (C). In view of the results of experiment (C), we set $\gamma = 1$ and $\omega = 1.1$. We let the initial point be $z^0 = 0$, and the termination criterion be $\|z^{k+1} - z^k\| < 10^{-8}$. We show the results in Table 4.4, where the number of iterations and the CPU time are the averages of 100 trials for each type

of $\mathcal{K}$. We find that the number of iterations gradually decreases as the number m of SOCs in $\mathcal{K}$ decreases. However, the whole CPU time does not decrease monotonically as m decreases, since subproblems to be solved at each iteration become more expensive as n_i becomes larger.

Table 4.4: Results for Algorithm 4.2.1 applied to SOCCPs (4.1.1) with various Cartesian structures of $\mathcal{K}$

n_i	m	comChol		incChol	
		♯iter	cpu(s)	♯iter	cpu(s)
1	1600	36.00	7.814	35.02	7.528
2	800	28.25	10.244	27.40	9.662
5	320	23.05	6.562	23.85	6.609
10	160	21.85	4.355	22.00	4.305
20	80	21.10	2.892	21.28	2.890
40	40	21.32	2.144	20.70	2.094
80	20	19.77	1.481	19.87	1.532
160	10	19.34	1.290	19.16	1.355
320	5	17.27	2.988	17.67	1.658
800	2	14.95	36.988	14.63	4.594
1600	1	11.99	175.595	13.13	19.179

4.6 Concluding remarks

In this chapter, we have extended the matrix splitting method for LCPs to SOCCPs and showed that the algorithm converges under the assumption that the matrix M is strictly $\mathcal{K}$-copositive. Furthermore, we have proposed a block SOR method for solving SOCCPs with M being positive definite, and demonstrated its effectiveness through numerical experiments. From a practical viewpoint, there is room for improvement in speeding up the algorithm. In particular, it would be highly effective if subproblems are solved in parallel. Moreover, it is desirable to develop a matrix splitting method for SOCCPs that is convergent under weaker conditions.

Chapter 5

Robust Nash equilibria and second-order cone complementarity problems

5.1 Introduction

We consider a bimatrix game where two players attempt to minimize their own costs. Let $y \in \Re^n$ and $z \in \Re^m$ denote strategies of Players 1 and 2, respectively. Moreover, let Player 1's cost function be given by $f_1(y,z) := y^T Az$ with cost matrix $A \in \Re^{n\times m}$, and let Player 2's cost function be given by $f_2(y,z) := y^T Bz$ with cost matrix $B \in \Re^{n\times m}$. We suppose that the two players choose their strategies y and z from the nonempty closed convex sets $S_1 \subseteq \Re^n$ and $S_2 \subseteq \Re^m$, respectively. Then, the players determine their strategies by solving the following minimization problems with the opponents' strategies fixed:

$$\begin{array}{ll} \underset{y}{\text{minimize}} & y^T Az \quad \text{subject to } y \in S_1, \\ \underset{z}{\text{minimize}} & y^T Bz \quad \text{subject to } z \in S_2. \end{array} \tag{5.1.1}$$

A point $(\overline{y}, \overline{z})$ satisfying $\overline{y} \in \operatorname{argmin}_{y\in S_1} y^T A\overline{z}$ and $\overline{z} \in \operatorname{argmin}_{z\in S_2} \overline{y}^T Bz$ is called a Nash equilibrium [3]. Since the minimization problems (5.1.1) are convex, the problem of finding a Nash equilibrium can be formulated as a variational inequality problem (VI) [33]. Moreover, if S_1 and S_2 are given by $S_1 = \{y \in \Re^n \,|\, g_i(y) \le 0,\ i = 1,\ldots,N\}$ and $S_2 = \{z \in \Re^m \,|\, h_j(z) \le 0, j = 1,\ldots,M\}$ with some convex functions $g_i : \Re^n \to \Re$ and $h_j : \Re^m \to \Re$, respectively, then the VI is further reformulated as a mixed complementarity problem (MCP), which is also called a box-constrained variational problem. Recently, MCP has been extensively studied and many efficient algorithms have been developed for solving it [17, 18, 95].

The concept of Nash equilibrium is premised on the accurate estimation of opponent's strategy and the exact evaluation of player's own cost function. Thus Nash equilibrium may hardly represent the actual situation when those operations are subject to errors. To deal with such situations, we introduce the concept of *robust Nash equilibrium*, which is parallel to that of robust optimization [5, 6, 7, 31].

In the field of game theory, there has been much study on games with incomplete information and the robustness of equilibria. Harsanyi [53, 54, 55] defines a game with incomplete information as a game where each player's payoff function is given in a stochastic manner with its probability distribution. This is one of the most popular formulations of games with incomplete information. Kajii and Morris [65] adopt Harsanyi's formulation to define a concept of robust equilibria to incomplete information. Especially, they show that games with strict equilibria do not necessarily have robust equilibria, and the unique correlated equilibrium of a game is robust. Moreover, they introduce the notion of p-dominance, and show that a p-dominant equilibrium is robust under an appropriate assumption. Ui [114] considers the robustness of equilibria of potential games, which is a class of games involving Monderer and Shapley's potential function [81]. He shows that the action profile uniquely maximizing a potential function is robust. Recently, Morris and Ui [84] have unified the above discussions for p-dominant equilibria and equilibria of potential games. They define a generalized potential function that contains Monderer and Shapley's potential function [81] and Morris's characteristic potential function [83], and show that an action maximizing the generalized potential function is a robust equilibrium to incomplete information.

In the above-mentioned references, the "robustness" means that an equilibrium is stable with respect to estimation errors. On the other hand, the robust Nash equilibrium introduced in this chapter is an equilibrium that results from robust optimization [5, 6, 7, 31] by each player. More precisely, our formulation is premised on the conditions (I) – (III) in the next section. Indeed, these conditions are applicable to several actual problems such as dynamic economic systems based on duopolistic competition with random disturbances, traffic equilibrium problems with incomplete information on travel costs, etc. We note that this concept is different from those considered in [65, 84, 114].

In what follows, we first define a robust Nash equilibrium for a bimatrix game, and discuss its existence. Then, we show that, under certain assumptions, the robust Nash equilibrium problem can be formulated as a second-order cone complementarity problem (SOCCP). The SOCCP is a class of complementarity problems where the complementarity condition is associated with the Cartesian product of second-order cones. Several methods for solving SOCCPs have been proposed recently [15, 19, 43, 56, 57].

5.2 Robust Nash equilibria and its existence

In this section, we define the robust Nash equilibrium of a bimatrix game, and give sufficient conditions for its existence.

Throughout the chapter, we assume that the following three statements hold for each player i $(i = 1, 2)$:

(I) Player 1 cannot estimate Player 2's strategy z exactly, but can only estimate that it belongs to a set $Z(z) \subseteq \Re^m$ containing z. Similarly, Player 2 cannot estimate Player 1's strategy y exactly, but can only estimate that it belongs to a set $Y(y) \subseteq \Re^n$ containing y.

(II) Player 1 cannot estimate his/her cost matrix exactly, but can only estimate that it belongs to a nonempty set $D_A \subseteq \Re^{n\times m}$. Player 2 cannot estimate his/her cost matrix exactly, but can only estimate that it belongs to a nonempty set $D_B \subseteq \Re^{n\times m}$.

(III) Each player tries to minimize his/her worst cost under (I) and (II).

Now, we define the robust Nash equilibrium under the above three assumptions. To realize (III), we define functions $\tilde{f}_i : \Re^n \times \Re^m \to \Re$ $(i = 1, 2)$ by

$$\begin{aligned} \tilde{f}_1(y,z) &:= \max\Big\{ y^T \hat{A}\hat{z} \;\Big|\; \hat{A} \in D_A, \hat{z} \in Z(z) \Big\}, \\ \tilde{f}_2(y,z) &:= \max\Big\{ \hat{y}^T \hat{B} z \;\Big|\; \hat{B} \in D_B, \hat{y} \in Y(y) \Big\}. \end{aligned} \tag{5.2.1}$$

The functions $\tilde{f}_1(\cdot, z)$ and $\tilde{f}_2(y, \cdot)$ represent Player 1's and Player 2's worst costs, respectively, under uncertainty as assumed in (I) and (II). Players 1 and 2 then solve the following minimization problems, respectively :

$$\begin{aligned} &\underset{y}{\text{minimize}} \;\; \tilde{f}_1(y,z) \quad \text{subject to} \;\; y \in S_1, \\ &\underset{z}{\text{minimize}} \;\; \tilde{f}_2(y,z) \quad \text{subject to} \;\; z \in S_2. \end{aligned} \tag{5.2.2}$$

Now, we are in a position to define the robust Nash equilibrium.

Definition 5.2.1 *Let functions $\tilde{f}_1$ and $\tilde{f}_2$ be defined by* (5.2.1). *If $\overline{y}^r \in \operatorname{argmin}_{y\in S_1} \tilde{f}_1(y, \overline{z}^r)$ and $\overline{z}^r \in \operatorname{argmin}_{z\in S_2} \tilde{f}_2(\overline{y}^r, z)$, that is, $(\overline{y}^r, \overline{z}^r)$ is a Nash equilibrium of game* (5.2.2), *then $(\overline{y}^r, \overline{z}^r)$ is called a robust Nash equilibrium of game* (5.1.1).

Next, we give a condition for the existence of a robust Nash equilibrium of game (5.1.1). Note that $Y(\cdot)$ and $Z(\cdot)$ given in (I) can be regarded as set-valued mappings. In what follows, we suppose that $Y(\cdot)$, $Z(\cdot)$, D_A and D_B in (I) and (II) satisfy the following assumption.

Assumption A

(a) *Set-valued mappings $Y : \Re^n \to \mathcal{P}(\Re^n)$ and $Z : \Re^m \to \mathcal{P}(\Re^m)$ are continuous, and $Y(y)$ and $Z(z)$ are nonempty compact for any $y \in \Re^n$ and $z \in \Re^m$.*

(b) *$D_A \subseteq \Re^{n\times m}$ and $D_B \subseteq \Re^{n\times m}$ are nonempty and compact sets.*

The functions $\tilde{f}_1$ and $\tilde{f}_2$ defined by (5.2.1) are well-defined under this assumption. By simple arguments on continuity, we can show that $\tilde{f}_1$ and $\tilde{f}_2$ are continuous everywhere. Furthermore, we have the following lemma on the convexity of $\tilde{f}_1(\cdot, z)$ and $\tilde{f}_2(y, \cdot)$. We omit the proof since it is trivial.

Lemma 5.2.1 *Suppose that Assumption* A *holds. Let $\tilde{f}_1$ and $\tilde{f}_2$ be defined by* (5.2.1). *Then, for any fixed $z \in \Re^m$ and $y \in \Re^n$, the functions $\tilde{f}_1(\cdot, z)$ and $\tilde{f}_2(y, \cdot)$ are convex.*

The next lemma is a fundamental result for noncooperative n-person game [3, Theorem 9.1.1].

Lemma 5.2.2 *Consider a noncooperative two-person game where cost functions are given by $\theta_1 : \Re^n \times \Re^m \to \Re$ and $\theta_2 : \Re^n \times \Re^m \to \Re$. Suppose that functions θ_1 and θ_2 are continuous at any (y, z), and functions $\theta_1(\cdot, z)$ and $\theta_2(y, \cdot)$ are convex. Suppose that S_1 and S_2 are nonempty compact convex sets. Then, the game has a Nash equilibrium.*

By the above lemmas, we obtain the following theorem for the existence of a robust Nash equilibrium of game (5.1.1)

Theorem 5.2.1 *Suppose that Assumption* A *holds, and S_1 and S_2 are nonempty compact convex sets. Then, game* (5.1.1) *has a robust Nash equilibrium.*

Proof. By Lemma 5.2.1, the functions $\tilde{f}_1(\cdot, z)$ and $\tilde{f}_2(y, \cdot)$ are convex. Moreover, as pointed out earlier, $\tilde{f}_1$ and $\tilde{f}_2$ are continuous everywhere. Therefore, from Lemma 5.2.2, game (5.2.2) has a Nash equilibrium. This means, by Definition 5.2.1, that game (5.1.1) has a robust Nash equilibrium. ■

5.3 SOCCP formulation of robust Nash equilibrium

In this section, we focus on the bimatrix game where each player takes a mixed strategy, that is, $S_1 = \{y \,|\, y \geq 0,\ e_n^T y = 1\}$ and $S_2 = \{z \,|\, z \geq 0,\ e_m^T z = 1\}$, and show that the robust Nash equilibrium problem reduces to an SOCCP.

Recall that the SOCCP is to find a vector $(\xi, \eta, \zeta) \in \Re^\ell \times \Re^\ell \times \Re^\nu$ satisfying the conditions

$$\mathcal{K} \ni \xi \perp \eta \in \mathcal{K}, \quad G(\xi, \eta, \zeta) = 0, \tag{5.3.1}$$

where $G : \Re^\ell \times \Re^\ell \times \Re^\nu \to \Re^\ell \times \Re^\nu$ is a given function, $\xi \perp \eta$ denotes $\xi^T \eta = 0$, and $\mathcal{K}$ is a closed convex cone defined by $\mathcal{K} = \mathcal{K}^{\ell_1} \times \mathcal{K}^{\ell_2} \times \cdots \times \mathcal{K}^{\ell_m}$ with ℓ_j-dimensional SOCs.

Consider the bimatrix game where Players 1 and 2 solve the following minimization problems (5.3.2) and (5.3.3), respectively :

$$\underset{y}{\text{minimize}} \ \ y^T A z \quad \text{subject to } \ y \geq 0,\ e_n^T y = 1, \tag{5.3.2}$$

$$\underset{z}{\text{minimize}} \ \ y^T B z \quad \text{subject to } \ z \geq 0,\ e_m^T z = 1. \tag{5.3.3}$$

It is well known that a Nash equilibrium of this game is given as a solution of a mixed linear complementarity problem. In fact, since z and y are fixed in (5.3.2) and (5.3.3), respectively, both problems are linear programming problems, and their KKT conditions are given by

$$\begin{aligned} &0 \leq y \perp Az + e_n s \geq 0, \quad e_n^T y = 1, \\ &0 \leq z \perp B^T y + e_m t \geq 0, \quad e_m^T z = 1, \end{aligned} \tag{5.3.4}$$

where $s \in \Re$ and $t \in \Re$ are Lagrange multipliers associated with the equality constraints in (5.3.2) and (5.3.3), respectively. Thus, if some (y, z) satisfies the above two KKT conditions simultaneously, then it is a Nash equilibrium of the bimatrix game. The problem of finding such a (y, z) can be further formulated as a linear complementarity problem (LCP) [24].

Now, we consider bimatrix games involving several types of uncertainty, and show that the robust Nash equilibrium problem corresponding to each game reduces to an SOCCP of the form

$$\mathcal{K} \ni M\zeta + q \perp N\zeta + r \in \mathcal{K}, \quad C\zeta = d \tag{5.3.5}$$

with variable $\zeta \in \Re^{\ell+\tau}$ and constants $M, N \in \Re^{\ell\times(\ell+\tau)}$, $q, r \in \Re^{\ell}$, $C \in \Re^{\tau\times(\ell+\tau)}$ and $d \in \Re^{\tau}$. Note that, by introducing new variables $\xi \in \Re^{\ell}$ and $\eta \in \Re^{\ell}$, problem (5.3.5) reduces to SOCCP (5.3.1) with $\nu = \ell + \tau$ and $G : \Re^{3\ell+\tau} \to \Re^{2\ell+\tau}$ defined by

$$G(\xi, \eta, \zeta) := \begin{pmatrix} \xi - M\zeta - q \\ \eta - N\zeta - r \\ C\zeta - d \end{pmatrix}.$$

5.3.1 Uncertainty in the opponent's strategy

In this subsection, we consider the case where each player estimates the cost matrix exactly but opponent's strategy uncertainly. More specifically, we make the following assumption.

Assumption 1

(a) $Y(y) := \{y+\delta y \in \Re^n \,|\, \|\delta y\| \le \rho_y,\ e_n^T\delta y = 0\}$ *and* $Z(z) := \{z+\delta z \in \Re^m \,|\, \|\delta z\| \le \rho_z,\ e_m^T\delta z = 0\}$, *where* ρ_y *and* ρ_z *are given positive constants.*

(b) $D_A = \{A\}$ *and* $D_B = \{B\}$, *where* $A \in \Re^{n\times m}$ *and* $B \in \Re^{n\times m}$ *are given constant matrices.*

Here, the conditions $e_n^T\delta y = e_m^T\delta z = 0$ in the definitions of $Y(y)$ and $Z(z)$ are provided so that $e_n^T(y+\delta y) = e_m^T(z+\delta z) = 1$ holds from $e_n^T y = e_m^T z = 1$. Under this assumption, the following theorem holds.

Theorem 5.3.1 *If Assumption 1 holds, then the bimatrix game has a robust Nash equilibrium.*

Proof. It is easily seen that Assumptions 1(a) and 1(b) imply Assumption A(a) and A(b), respectively. Hence, the theorem readily follows from Theorem 5.2.1. ∎

We now show that the robust Nash equilibrium problem can be formulated as SOCCP (5.3.5) under Assumption 1. Player 1 solves the following minimization problem to determine his/her strategy :

$$\begin{aligned} \underset{y}{\text{minimize}} \quad & \max\left\{ y^T A(z+\delta z) \;\middle|\; \|\delta z\| \le \rho_z,\ e_m^T\delta z = 0 \right\} \\ \text{subject to} \quad & e_n^T y = 1,\ y \ge 0. \end{aligned} \tag{5.3.6}$$

Since the projection of vector $A^T y$ onto hyperplane $\pi := \{z \,|\, e_m^T z = 0\}$ can be represented as $(I_m - m^{-1}e_m e_m^T)A^T y$, the cost function can be written as

$$\begin{aligned} \tilde{f}_1(y,z) &= \max\left\{ y^T A(z+\delta z) \;\middle|\; \|\delta z\| \le \rho_z,\ e_m^T\delta z = 0 \right\} \\ &= y^T Az + \max\left\{ y^T A\delta z \;\middle|\; \|\delta z\| \le \rho_z,\ e_m^T\delta z = 0 \right\} \\ &= y^T Az + \rho_z \|\tilde{A}^T y\|, \end{aligned}$$

where $\tilde{A} := A(I_m - m^{-1}e_m e_m^T)$. Hence, by introducing an auxiliary variable $y_0 \in \Re$, problem (5.3.6) can be reduced to the following convex minimization problem :

$$\begin{aligned} \underset{y_0,\, y}{\text{minimize}} \quad & y^T Az + \rho_z y_0 \\ \text{subject to} \quad & \|\tilde{A}^T y\| \le y_0,\ \ y \ge 0,\ \ e_n^T y = 1. \end{aligned}$$

This is a second-order cone programming problem [2, 76] and its KKT conditions can be written as the following SOCCP :

$$\mathcal{K}^{m+1} \ni \begin{pmatrix} \lambda_0 \\ \lambda \end{pmatrix} \perp \begin{pmatrix} 1 & 0 \\ 0 & \tilde{A}^T \end{pmatrix} \begin{pmatrix} y_0 \\ y \end{pmatrix} \in \mathcal{K}^{m+1},$$
$$\Re^n_+ \ni y \perp Az - \tilde{A}\lambda + e_n s \in \Re^n_+, \quad e_n^T y = 1, \quad \lambda_0 = \rho_z,$$

where $\lambda \in \Re^m$ and $s \in \Re$ are Lagrange multipliers, and $\lambda_0 \in \Re$ is an auxiliary variable. In a similar manner, the KKT conditions for Player 2 can be written as

$$\mathcal{K}^{n+1} \ni \begin{pmatrix} \mu_0 \\ \mu \end{pmatrix} \perp \begin{pmatrix} 1 & 0 \\ 0 & \tilde{B} \end{pmatrix} \begin{pmatrix} z_0 \\ z \end{pmatrix} \in \mathcal{K}^{n+1},$$
$$\Re^m_+ \ni z \perp B^T y - \tilde{B}^T \mu + e_m t \in \Re^m_+, \quad e_m^T z = 1, \quad \mu_0 = \rho_y,$$

where $\mu \in \Re^n$ and $t \in \Re$ are Lagrange multipliers, and $\mu_0 \in \Re$ is an auxiliary variable. Consequently, the problem to find (y, z) satisfying the above two KKT conditions simultaneously can be reformulated as SOCCP (5.3.5) with $\ell = 2m + 2n + 2$, $\tau = 4$, $\mathcal{K} = \mathcal{K}^{n+1} \times \mathcal{K}^{m+1} \times \Re^m_+ \times \Re^n_+$,

$$\zeta = \begin{pmatrix} y_0 \\ y \\ z_0 \\ z \\ \lambda_0 \\ \lambda \\ \mu_0 \\ \mu \\ s \\ t \end{pmatrix}, \quad M = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_m & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_n & 0 & 0 \\ 0 & I_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I_m & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad q = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

$$N = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \tilde{A}^T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \tilde{B} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & A & 0 & -\tilde{A} & 0 & 0 & e_n & 0 \\ 0 & B^T & 0 & 0 & 0 & 0 & 0 & -\tilde{B}^T & 0 & e_m \end{pmatrix}, \quad r = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

$$C = \begin{pmatrix} 0 & e_n^T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & e_m^T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}, \quad d = \begin{pmatrix} 1 \\ 1 \\ \rho_z \\ \rho_y \end{pmatrix}.$$

5.3.2 Component-wise uncertainty in the cost matrices

In the following three subsections 5.3.2, 5.3.3 and 5.3.4, we consider the case where each player estimates the opponent's strategy exactly but his/her own cost matrix uncertainly. In this subsection, we particularly focus on the case where the uncertainty in each cost matrix occurs component-wise independently. That is, we make the following assumption.

Assumption 2

(a) $Y(y) = \{y\}$ *and* $Z(z) = \{z\}$.

(b) $D_A := \{A + \delta A \in \Re^{n\times m} \,|\, |\delta A_{ij}| \le (\Gamma_A)_{ij} \ (i = 1, \ldots, n, \ j = 1, \ldots, m)\}$ *and* $D_B := \{B + \delta B \in \Re^{n\times m} \,|\, |\delta B_{ij}| \le (\Gamma_B)_{ij} \ (i = 1, \ldots, n, \ j = 1, \ldots, m)\}$ *with given constant matrices* $A \in \Re^{n\times m}$ *and* $B \in \Re^{n\times m}$ *and positive constant matrices* $\Gamma_A \in \Re^{n\times m}$ *and* $\Gamma_B \in \Re^{n\times m}$.

Under this assumption, we have the following theorem. We omit the proof, since it is similar to that of Theorem 5.3.1.

Theorem 5.3.2 *If Assumption 2 holds, then the bimatrix game has a robust Nash equilibrium.*

From Assumption 2, together with the constraints $y \ge 0$ and $z \ge 0$, the cost function $\tilde{f}_1$ can be represented as

$$\begin{aligned}
\tilde{f}_1(y,z) &= \max\left\{ y^T \hat{A} z \;\middle|\; \hat{A} \in D_A \right\} \\
&= y^T A z + \max_{|\delta A_{ij}| \le (\Gamma_A)_{ij}} \sum_{i=1}^{n}\sum_{j=1}^{m} \delta A_{ij} y_i z_j \\
&= y^T A z + \sum_{i=1}^{n}\sum_{j=1}^{m} (\Gamma_A)_{ij} y_i z_j \\
&= y^T (A + \Gamma_A) z.
\end{aligned}$$

Analogously, we have $\tilde{f}_2(y,z) = y^T(B + \Gamma_B)z$. Hence, the robust Nash equilibrium problem is simply the problem of finding a Nash equilibrium of the bimatrix game with cost matrices $A + \Gamma_A$ and $B + \Gamma_B$. This problem reduces to the MCP (5.3.4) with A and B replaced by $A + \Gamma_A$ and $B + \Gamma_B$, respectively.

5.3.3 Column/row-wise uncertainty in the cost matrices

In this subsection, we focus on the case where the uncertainty in matrices A and B respectively occur row-wise independently and column-wise independently. That is, we make the following assumption.

Assumption 3

(a) $Y(y) = \{y\}$ *and* $Z(z) = \{z\}$.

(b) $D_A := \{A + \delta A \,|\, \|\delta A^c_j\| \le (\gamma_A)_j \ (j = 1, \ldots, m)\}$ *and* $D_B := \{B + \delta B \,|\, \|\delta B^r_i\| \le (\gamma_B)_i \ (i = 1, \ldots, n)\}$ *with given constant matrices* $A \in \Re^{n\times m}$ *and* $B \in \Re^{n\times m}$ *and positive constant vectors* $\gamma_A \in \Re^m$ *and* $\gamma_B \in \Re^n$.

This assumption implies that the degree of uncertainty in each player's cost depends on the opponent's each pure strategy. Under this assumption, we have the following theorem. We omit the proof, since it is similar to that of Theorem 5.3.1.

Theorem 5.3.3 *If Assumption 3 holds, then the bimatrix game has a robust Nash equilibrium.*

Next, we formulate the Nash equilibrium problem as an SOCCP. From Assumption 3, we have

$$\begin{aligned}\tilde{f}_1(y,z) &= \max_{\hat{A}\in D_A} y^T\hat{A}z \\ &= y^TAz + \max_{\|\delta A_j^c\|\le(\gamma_A)_j}\sum_{j=1}^m z_j\, y^T\delta A_j^c \\ &= y^TAz + \sum_{j=1}^m z_j\|y\|(\gamma_A)_j \\ &= y^TAz + \gamma_A^Tz\|y\|\end{aligned}$$

and $\gamma_A^Tz \ge 0$. Hence, by introducing an auxiliary variable $y_0 \in \Re$, Player 1's problem can be written as

$$\begin{array}{ll}\underset{y_0,\,y}{\text{minimize}} & y^TAz + (\gamma_A^Tz)y_0 \\ \text{subject to} & \|y\| \le y_0,\ \ y \ge 0,\ \ e_n^Ty = 1.\end{array}$$

This is a second-order cone programming problem, and its KKT conditions are given by

$$\begin{gathered}\mathcal{K}^{n+1} \ni \begin{pmatrix} y_0 \\ y\end{pmatrix} \perp \begin{pmatrix} \gamma_A^Tz \\ Az + e_ns - \lambda\end{pmatrix} \in \mathcal{K}^{n+1} \\ \Re_+^n \ni \lambda \perp y \in \Re_+^n,\ \ e_n^Ty = 1\end{gathered}$$

with Lagrange multipliers $\lambda \in \Re^n$ and $s \in \Re$.

In a similar way, the KKT conditions for Player 2's minimization problem are given by

$$\begin{gathered}\mathcal{K}^{m+1} \ni \begin{pmatrix} z_0 \\ z\end{pmatrix} \perp \begin{pmatrix} \gamma_B^Ty \\ B^Ty + e_mt - \mu\end{pmatrix} \in \mathcal{K}^{m+1} \\ \Re_+^m \ni \mu \perp z \in \Re_+^m,\ \ e_m^Tz = 1,\end{gathered}$$

where $\mu \in \Re^m$ and $t \in \Re$ are Lagrange multipliers. Combining the above two KKT conditions, we obtain SOCCP (5.3.5) with $\ell = 2m + 2n + 2$, $\tau = 2$, $\mathcal{K} := \mathcal{K}^{n+1} \times \mathcal{K}^{m+1} \times \Re_+^n \times \Re_+^m$,

$$\zeta = \begin{pmatrix} y_0 \\ y \\ z_0 \\ z \\ \lambda \\ \mu \\ s \\ t\end{pmatrix},\ \ M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & I_n & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I_m & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I_n & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_m & 0 & 0 \end{pmatrix},\ \ q = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0\end{pmatrix},$$

$$N = \begin{pmatrix} 0 & 0 & 0 & \gamma_A^T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & A & -I_n & 0 & e_n & 0 \\ 0 & \gamma_B^T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & B^T & 0 & 0 & 0 & -I_m & 0 & e_m \\ 0 & I_n & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I_m & 0 & 0 & 0 & 0 \end{pmatrix},\ \ r = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0\end{pmatrix},$$

$$C = \begin{pmatrix} 0 & e_n^T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & e_m^T & 0 & 0 & 0 & 0 \end{pmatrix}, \quad d = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

5.3.4 General uncertainty in the cost matrices

In this subsection, we consider the general case of uncertainty in each cost matrix. That is, we make the following assumption.

Assumption 4

(a) $Y(y) = \{y\}$ *and* $Z(z) = \{z\}$.

(b) $D_A := \{A + \delta A \in \Re^{n\times m} \,|\, \|\delta A\|_F \le \rho_A\}$ *and* $D_B := \{B + \delta B \in \Re^{n\times m} \,|\, \|\delta B\|_F \le \rho_B\}$ *with given constant matrices* $A \in \Re^{n\times m}$ *and* $B \in \Re^{n\times m}$ *and positive scalars* ρ_A *and* ρ_B.

Under this assumption, we also have the following theorem.

Theorem 5.3.4 *If Assumption 4 holds, then the bimatrix game has a robust Nash equilibrium.*

Next, we consider the SOCCP formulation of the game. First note that

$$\tilde{f}(y,z) = \max\{y^T \hat{A} z \,|\, \hat{A} \in D_A\} = y^T A z + \max_{\|\delta A\|_F \le \rho_A} y^T (\delta A) z.$$

Moreover, we have

$$\max_{\|\delta A\|_F \le \rho_A} y^T (\delta A) z \;=\; \max_{\|\delta A\|_F \le \rho_A} (z \otimes y)^T \mathrm{vec}\,(\delta A) \;=\; \|z \otimes y\| \rho_A \;=\; \rho_A \|y\| \|z\|,$$

where $\mathrm{vec}\,(\cdot)$ denotes the vec operator that creates an nm-dimensional vector $((p_1^c)^T, \ldots, (p_m^c)^T)^T$ from a matrix $P \in \Re^{n\times m}$ with column vectors $p_1^c, \ldots, p_m^c$, and $\otimes$ denotes Kronecker product (see Sections 4.2 and 4.3 in [60]). Hence, by introducing an auxiliary variable $y_0 \in \Re$, Player 1's minimization problem reduces to the following problem:

$$\begin{array}{ll} \underset{y_0,\, y}{\text{minimize}} & y^T A z + \rho_A \|z\| y_0 \\ \text{subject to} & \|y\| \le y_0, \quad e_n^T y = 1, \quad y \ge 0. \end{array}$$

Again, this is a second-order cone programming problem, and its KKT conditions are given by

$$\mathcal{K}^{n+1} \ni \begin{pmatrix} y_0 \\ y \end{pmatrix} \perp \begin{pmatrix} \rho_A \|z\| \\ Az + e_n s - \lambda \end{pmatrix} \in \mathcal{K}^{n+1},$$
$$\Re^n_+ \ni \lambda \perp y \in \Re^n_+ \quad e_n^T y = 1,$$

where $\lambda \in \Re^n$ and $s \in \Re$ are Lagrange multipliers. Moreover, we can show that this SOCCP is rewritten as the following SOCCP:

$$\begin{aligned} &\mathcal{K}^{n+1} \ni \begin{pmatrix} y_0 \\ y \end{pmatrix} \perp \begin{pmatrix} \rho_A z_1 \\ Az + e_n s - \lambda \end{pmatrix} \in \mathcal{K}^{n+1}, \quad e_n^T y = 1, \\ &\mathcal{K}^{m+1} \ni \begin{pmatrix} z_1 \\ z \end{pmatrix} \perp \begin{pmatrix} y_0 \\ u \end{pmatrix} \in \mathcal{K}^{m+1}, \quad \Re^n_+ \ni \lambda \perp y \in \Re^n_+ \end{aligned} \tag{5.3.7}$$

with an auxiliary variable $u \in \Re^m$. To see this, it suffices to notice that the complementarity condition

$$\mathcal{K}^{m+1} \ni \begin{pmatrix} z_1 \\ z \end{pmatrix} \perp \begin{pmatrix} y_0 \\ u \end{pmatrix} \in \mathcal{K}^{m+1} \tag{5.3.8}$$

in (5.3.7) implies $\|z\| = z_1$. This fact can be verified as follows: On one hand, $\binom{z_1}{z} \in \mathcal{K}^{m+1}$ implies $\|z\| \le z_1$. On the other hand, it holds that $0 = z_1 y_0 + z^T u \ge z_1 y_0 - \|z\|\|u\| \ge z_1 y_0 - \|z\| y_0$, where the equality follows from the perpendicularity in (5.3.8), the first inequality follows from the Cauchy-Schwarz inequality, and the last inequality follows from the condition $\binom{y_0}{u} \in \mathcal{K}^{m+1}$ in (5.3.8). Moreover, $e_n^T y = 1$ and $\binom{y_0}{y} \in \mathcal{K}^{n+1}$ imply $y_0 > 0$. Hence, we have $\|z\| \ge z_1$.

In a similar way, the KKT conditions for Player 2's problem are given by

$$\mathcal{K}^{m+1} \ni \begin{pmatrix} z_0 \\ z \end{pmatrix} \perp \begin{pmatrix} \rho_B y_1 \\ B^T y + e_m t - \mu \end{pmatrix} \in \mathcal{K}^{m+1}, \quad e_m^T z = 1,$$
$$\mathcal{K}^{n+1} \ni \begin{pmatrix} y_1 \\ y \end{pmatrix} \perp \begin{pmatrix} z_0 \\ v \end{pmatrix} \in \mathcal{K}^{n+1}, \quad \Re^m_+ \ni \mu \perp z \in \Re^m_+,$$

where $t \in \Re$, $\mu \in \Re^m$ and $v \in \Re^n$ are auxiliary variables. Combining the above two KKT conditions, we obtain SOCCP (5.3.5) with $\ell = 3m+3n+4$, $\tau = 2$, $\mathcal{K} = \mathcal{K}^{n+1} \times \mathcal{K}^{n+1} \times \mathcal{K}^{m+1} \times \mathcal{K}^{m+1} \times \Re^n_+ \times \Re^m_+$,

$$\zeta = \begin{pmatrix} y_0 \\ y \\ y_1 \\ v \\ z_0 \\ z \\ z_1 \\ u \\ \lambda \\ \mu \\ s \\ t \end{pmatrix}, \quad M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & I_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & I_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_m & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_m & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_n & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_m & 0 & 0 \end{pmatrix}, \quad q = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

$$N = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & \rho_A & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & A & 0 & 0 & -I_n & 0 & e_n & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \rho_B & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & B^T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -I_n & 0 & e_m \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_m & 0 & 0 & 0 & 0 \\ 0 & I_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_m & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad r = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

$$C = \begin{pmatrix} 0 & e_n^T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & e_m^T & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad d = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

As shown thus far, the robust Nash equilibrium problems under Assumptions 1, 3 and 4 are transformed into SOCCPs, while the robust Nash equilibrium problem under Assumption 2 reduces to an LCP. This is a natural consequence of the fact that the Euclidean norm is used in Assumptions 1, 3 and 4 to describe the uncertainty and the absolute value is used in Assumption 2.

5.4 Numerical examples of robust Nash equilibria

In the previous section, we have shown that some robust Nash equilibrium problems for bimatrix games reduce to SOCCPs. In this section, we present some numerical examples for the robust Nash equilibria. Several methods have been proposed for solving SOCCPs. Among them, one of the most popular approaches is to reformulate the SOCCP as an equivalent nondifferentiable minimization problem and solve it by Newton-type method combined with a smoothing technique [15, 19, 43, 57]. In our numerical experiments, we use an algorithm based on the methods proposed in [57].

5.4.1 Uncertainty in opponent's strategy

We first study the case where only the opponents' strategies involve uncertainty, that is, Assumption 1 holds.

We consider the bimatrix game with cost matrices:

$$A_1 = \begin{pmatrix} -1 & -9 & 11 \\ 10 & -1 & 4 \\ 3 & 10 & 1 \end{pmatrix}, \quad B_1 = \begin{pmatrix} -5 & -4 & -8 \\ -1 & 0 & 5 \\ 3 & 1 & 4 \end{pmatrix}. \tag{5.4.1}$$

The Nash equilibrium of the game is given by $\overline{y} = (0.4815, 0.1852, 0.3333)$ and $\overline{z} = (0.1699, 0.2628, 0.5673)$. Robust Nash equilibria $(\overline{y}^r, \overline{z}^r)$ for various values of (ρ_y, ρ_z) are shown in Table 5.1, where $f_i(\overline{y}^r, \overline{z}^r)$ denotes the cost value of each player $i = 1, 2$ at a robust Nash equilibrium. From the table, we see that robust Nash equilibria $(\overline{y}^r, \overline{z}^r)$ approach the Nash equilibrium $(\overline{y}, \overline{z})$ as both ρ_y and ρ_z tend to 0. Note that Players 1 and 2 estimate the opponents' strategies more precisely as ρ_z and ρ_y become smaller. However, the costs of both players for $(\rho_y, \rho_z) = (0.5, 0.5)$ are smaller than those for $(\rho_y, \rho_z) = (0.01, 0.01)$. This implies that an equilibrium may not necessarily be favorable for either of the players even if the estimation is more precise.

Next, we consider the bimatrix game with cost matrices:

$$A_2 = \begin{pmatrix} 5 & 7 & 8 \\ 2 & 3 & 0 \\ -1 & -3 & -2 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 8 & 2 & -7 \\ 5 & 3 & -3 \\ 9 & 1 & -4 \end{pmatrix}. \tag{5.4.2}$$

In this case, the Nash equilibrium comprises $\overline{y} = (0, 0, 1)$ and $\overline{z} = (0, 0, 1)$, and for any pair $(\rho_y, \rho_z) \in \{0.5, 0.1, 0.01\} \times \{0.5, 0.1, 0.01\}$, robust Nash equilibrium $(\overline{y}^r, \overline{z}^r)$ remains the same, that is, $\overline{y}^r = \overline{y}$ and $\overline{z}^r = \overline{z}$. From this result, we may expect that, if the Nash equilibrium is a pure strategy, then the robust Nash equilibrium remains unchanged even if there is uncertainty to some extent.

5.4.2 Uncertainty in cost matrices

We next study the general case where the players' cost matrices involve uncertainty, that is, Assumption 4 holds.

First we consider the bimatrix game with the cost matrices A_1 and B_1 defined by (5.4.1). Robust Nash equilibria for various values of (ρ_A, ρ_B) are shown in Table 5.2, where $f_i(\overline{y}^r, \overline{z}^r)$ denotes the cost value of robust Nash equilibrium. As in the previous case, we see from the table that precise estimation for cost matrices does not necessarily reduce the cost at an equilibrium.

Next we consider the bimatrix game with cost matrices A_2 and B_2 defined by (5.4.2). Robust Nash equilibria for various values of (ρ_A, ρ_B) are shown in Table 5.3, which reveals that $\overline{y}^r = \overline{y}$ and $\overline{z}^r = \overline{z}$ hold when ρ_A and ρ_B are sufficiently small. We also see from the table that precise estimation for cost matrices does not always result in the reduction of the players' costs at an equilibrium.

5.5 Concluding remarks

In this chapter, we have defined the concept of robust Nash equilibrium, and studied a sufficient condition for its existence. Moreover, we have shown that some robust Nash equilibrium problems can be reformulated as SOCCPs. To investigate the behavior of robust Nash equilibria, we have carried out some numerical examples.

Our study is still in the infancy, and many issues remain to be addressed. (1) One is to extend the concept of robust Nash equilibrium to the general N-person game. For the 2-person bimatrix game studied in this chapter, it is sufficient to consider the uncertainty in the cost matrices and the opponent's strategy. To discuss general N-person games, more complicated structure should be dealt with. (2) Another issue is to find other sufficient conditions for the existence of robust Nash equilibria. For instance, it may be possible to consider the existence of robust Nash equilibria without assuming the boundedness of strategy sets. (3) Theoretical study on the relation between Nash equilibrium and robust Nash equilibrium is also worthwhile. For example, it is not known whether the uniqueness of Nash equilibrium is inherited to robust Nash equilibrium. (4) In this chapter, we have formulated several robust Nash equilibrium problems as SOCCPs. However, we have only considered the cases where either the cost matrices or the opponent's strategy is uncertain for each player. It seems interesting to study the case where both of them are uncertain, or the structure of uncertainty is more complicated. (5) In our numerical experiments, we employed an existing algorithm for solving SOCCPs. But, there is room for improvement of solution methods. It may be useful to develop a specialized method for solving robust Nash equilibrium problems.

Table 5.1: Robust Nash equilibria for various values of ρ_y and ρ_z

ρ_y	ρ_z	$\overline{y}^r$	$\overline{z}^r$	$f_1(\overline{y}^r, \overline{z}^r)$	$f_2(\overline{y}^r, \overline{z}^r)$
0.01	0.01	$(0.4896, 0.1814, 0.3290)$	$(0.1702, 0.2697, 0.5601)$	3.650	-1.668
0.1	0.1	$(0.5630, 0.1482, 0.2888)$	$(0.1758, 0.3304, 0.4938)$	3.039	-2.305
0.1	0.5	$(0.5621, 0.1560, 0.2819)$	$(0.1948, 0.6032, 0.2019)$	0.345	-2.122
0.5	0.1	$(0.8891, 0.0011, 0.1098)$	$(0.1812, 0.3272, 0.4916)$	2.506	-5.152
0.5	0.5	$(0.8840, 0.0432, 0.0729)$	$(0.2129, 0.5929, 0.1942)$	-2.424	-4.232

Table 5.2: Robust Nash equilibria for various values of ρ_A and ρ_B (cost matrices A_1 and B_1)

ρ_A	ρ_B	$\overline{y}^r$	$\overline{z}^r$	$f_1(\overline{y}^r, \overline{z}^r)$	$f_2(\overline{y}^r, \overline{z}^r)$
0.1	0.1	$(0.4841, 0.1797, 0.3362)$	$(0.1721, 0.2623, 0.5656)$	3.700	-1.615
1	1	$(0.5097, 0.1376, 0.3527)$	$(0.1969, 0.2552, 0.5479)$	3.640	-1.835
1	10	$(1.0000, 0.0000, 0.0000)$	$(0.2931, 0.2326, 0.4743)$	2.830	-6.190
10	1	$(0.5083, 0.1950, 0.2967)$	$(0.3497, 0.2453, 0.4050)$	3.074	-1.843
10	10	$(0.5934, 0.1961, 0.2105)$	$(0.3326, 0.3002, 0.3672)$	2.396	-2.565

Table 5.3: Robust Nash equilibria for various values of ρ_A and ρ_B (cost matrices A_2 and B_2)

ρ_A	ρ_B	$\overline{y}^r$	$\overline{z}^r$	$f_1(\overline{y}^r, \overline{z}^r)$	$f_2(\overline{y}^r, \overline{z}^r)$
0.1	0.1	$(0.0000, 0.0000, 1.0000)$	$(0.0000, 0.0000, 1.0000)$	-2.000	-4.000
1	1	$(0.0000, 0.0000, 1.0000)$	$(0.0000, 0.0000, 1.0000)$	-2.000	-4.000
1	10	$(0.0000, 0.0000, 1.0000)$	$(0.0000, 0.3110, 0.6890)$	-2.311	-2.445
10	1	$(0.0000, 0.4286, 0.5714)$	$(0.0000, 0.0000, 1.0000)$	-1.143	-3.571
10	10	$(0.0000, 0.3783, 0.6217)$	$(0.0000, 0.1935, 0.8065)$	-1.144	-2.581

Chapter 6

Conclusion

In this thesis, we have proposed two different types of algorithms for solving SOCCPs. One is a Newton-type algorithm in which the smoothing method and the regularization method are combined, and the other is the algorithm based on matrix splitting method. Furthermore, we have introduced the concept of robust Nash equilibrium, and reformulated the problem of finding such an equilibrium as an SOCCP. The results obtained in this thesis are summarized as follows :

(a) In Chapter 3, we proposed a globally and quadratically convergent algorithm, which is based on smoothing and regularization methods, for solving monotone SOCCP. In particular, we studied strong semismoothness and Jacobian consistency, which play an important role in establishing quadratic convergence of the algorithm. Furthermore, we examined effectiveness of the algorithm by means of numerical experiments.

(b) In Chapter 4, we proposed an iterative method for the symmetric affine SOCCP that is based on the idea of matrix splitting. We first gave conditions under which the matrix splitting method converges to a solution of the affine SOCCP. We then presented, as a particular realization of the matrix splitting method, the block successive overrelaxation (SOR) method for the affine SOCCP involving a positive definite matrix, and proposed an efficient method for solving subproblems. Finally, we reported some numerical results with the proposed algorithm, where promising results were obtained especially for problems with sparse matrices.

(c) In Chapter 5, we considered a bimatrix game in which the players can neither evaluate their cost functions exactly nor estimate their opponents' strategies accurately. To formulate such a game, we introduced the concept of robust Nash equilibrium that results from robust optimization by each player, and proved its existence under some mild conditions. Moreover, we showed that a robust Nash equilibrium in the bimatrix game can be characterized as an SOCCP. Some numerical results were presented to illustrate the behavior of robust Nash equilibria.

As we summarized above, we have made some contributions on SOCCP. However, there are several problems that remain unsolved. In the following, we give some future issues.

(a) In Chapter 3, we restricted ourselves to SOCCP (1.1.1) with $F(x, y, \zeta) = f(x) - y$. Moreover, we assumed the monotonicity on f to show the global and quadratic convergence of the algorithm. However, the conditions are rather restrictive from the practical viewpoint. It is worthwhile to improve the algorithm of which the global and rapid convergence are guaranteed under weaker conditions.

(b) In Chapter 4, we have developed the matrix splitting algorithm based on the block SOR method. It may be interesting to construct an algorithm based on matrix splitting methods other than the SOR-type methods. For example, it is challenging to study the (block) Jacobi method for solving SOCCP, since Jacobi method suits parallel computation.

(c) The results derived in Chapter 5 are still inadequate. Especially, theoretical results, such as the uniqueness of a solution or convexity of the solution set, have not been clarified sufficiently. It is certainly important to answer such problems.

(d) The robust Nash equilibrium problem considered in Chapter 5 is one of applications of SOCCP. It may be fascinating to apply SOCCP to other practical problems. The concept of equilibria appears not only in Nash games but also in other various situations (e.g. traffic assignment, economic model, etc). For such situations, we may define a similar concept of robust Nash equilibrium and reformulate the problem of finding such an equilibrium into SOCCP. Especially, if the uncertainty is expressed by using the Euclidean norm, then the problem is expected to reduce to SOCCP.

Appendix A

Level-boundedness of the merit function and its smoothing function

In Section 3.3, we used the fact that, if function f is strongly monotone, then the merit function Ψ_{NR} and its smoothing function Ψ_μ are level-bounded. In this appendix, we give a proof of this fact. Since the level-boundedness of a function $\Psi : \Re^n \to \Re$ is equivalent to

$$\lim_{\|z\|\to\infty} \Psi(z) = +\infty, \tag{A.1}$$

we show (A.1) instead of the boundedness of the level set $\mathcal{L}_\alpha := \{z \mid \Psi(z) \leq \alpha\}$.

Now, let the function $\tilde{\Psi}_{\text{NR}} : \Re^n \to \Re$ be defined by

$$\tilde{\Psi}_{\text{NR}}(x) := \|\varphi_{\text{NR}}(x, f(x))\|. \tag{A.2}$$

Then the following two lemmas hold.

Lemma A.1 *If f is strongly monotone, then $\tilde{\Psi}_{\text{NR}}$ is level-bounded.*

This result was proved for the variational inequality problem by Peng and Fukushima [93]. Recently, Yamashita and Fukushima [121] generalized this result by introducing the concept of strong coerciveness, which is weaker than the strong monotonicity.

Lemma A.2 *$\tilde{\Psi}_{\text{NR}}$ is level-bounded if and only if Ψ_{NR} is level-bounded.*

Proof. We first show "if" part. Suppose that Ψ_{NR} is level-bounded. Then we have

$$\lim_{\|(x,y)\|\to\infty} \Psi_{\text{NR}}(x, y) = +\infty. \tag{A.3}$$

Let $\{x^{(k)}\}$ be an arbitrary sequence such that $\|x^{(k)}\| \to \infty$ and $\{y^{(k)}\}$ be the corresponding sequence such that $y^{(k)} = f(x^{(k)})$ for all k. Then we have

$$\begin{aligned}
\frac{1}{2}\tilde{\Psi}_{\text{NR}}(x^{(k)})^2 &= \frac{1}{2}\|\varphi_{\text{NR}}(x^{(k)}, f(x^{(k)}))\|^2 \\
&= \frac{1}{2}\|\varphi_{\text{NR}}(x^{(k)}, y^{(k)})\|^2 + \frac{1}{2}\|f(x^{(k)}) - y^{(k)}\|^2 \\
&= \Psi_{\text{NR}}(x^{(k)}, y^{(k)}),
\end{aligned}$$

where the second equality follows from $y^{(k)} = f(x^{(k)})$. Noticing that $\|(x^{(k)}, y^{(k)})\| \to \infty$ as $k \to \infty$ and that (A.3) holds, we have $\lim_{k\to\infty} \tilde{\Psi}_{\text{NR}}(x^{(k)}) = +\infty$, that is, $\tilde{\Psi}_{\text{NR}}$ is level-bounded.

We next show the "only if" part. Suppose that $\tilde{\Psi}_{\text{NR}}$ is level-bounded. Then $\tilde{\Psi}_{\text{NR}}$ satisfies (A.1). Noticing $\sqrt{2\|\xi\|^2 + 2\|\eta\|^2} \geq \|\xi\| + \|\eta\|$ for any ξ, $\eta \in \Re^n$, we have

$$\begin{aligned}
2\sqrt{\Psi_{\text{NR}}(x,y)} &\geq \|x - P_{\mathcal{K}^n}(x-y)\| + \|f(x) - y\| \\
&\geq \|x - P_{\mathcal{K}^n}(x - f(x))\| - \|P_{\mathcal{K}^n}(x - f(x)) - P_{\mathcal{K}^n}(x-y)\| + \|f(x) - y\| \\
&\geq \|x - P_{\mathcal{K}^n}(x - f(x))\| - \|f(x) - y\| + \|f(x) - y\| \\
&= \tilde{\Psi}_{\text{NR}}(x),
\end{aligned}$$

where the second inequality follows from the triangle inequality, and the third inequality follows from the nonexpansiveness of the projection operator. Hence, Ψ_{NR} also satisfies (A.1), that is, Ψ_{NR} is level-bounded. ■

We now give the main theorem of this appendix.

Theorem A.1 *If f is strongly monotone, then Ψ_μ is level-bounded for any $\mu \geq 0$.*

Proof. By (3.3.5), we have

$$\begin{aligned}
\sqrt{2\Psi_{\text{NR}}(x,y)} &= \left\| \begin{pmatrix} \varphi_{\text{NR}}(x,y) \\ f(x) - y \end{pmatrix} \right\| \\
&\leq \left\| \begin{pmatrix} \varphi_\mu(x,y) \\ f(x) - y \end{pmatrix} \right\| + \left\| \begin{pmatrix} \varphi_{\text{NR}}(x,y) - \varphi_\mu(x,y) \\ 0 \end{pmatrix} \right\| \\
&\leq \sqrt{2\Psi_\mu(x,y)} + \nu\mu.
\end{aligned}$$

From Lemmas A.1 and A.2, Ψ_{NR} is level-bounded if f is strongly monotone. This implies the level-boundedness of Ψ_μ. ■

Appendix B

Alternative proof of Theorem 3.4.1

In Section 3.4.1, we introduced the weak univalence property of $H_{\rm NR}$ to show the global convergence of Algorithm 3.4.1. In this appendix, we give an alternative proof of Theorem 3.4.1 by using the Mountain Pass Theorem.

We first show that any stationary point of $\Psi_{\mu,\varepsilon}$ is a global minimum of $\Psi_{\mu,\varepsilon}$ by using the nonsingularity of $\nabla H_{\mu,\varepsilon}(x,y)$.

Proposition B.1 *If $f:\Re^n\to\Re^n$ is monotone, then the Jacobian $\nabla H_{\mu,\varepsilon}(x,y)$ is nonsingular for any $\mu>0$, $\varepsilon\geq 0$ and $(x,y)\in\Re^n\times\Re^n$.*

Proof. Firstly, we show that $O\prec\nabla P_\mu(z)\prec I$ for any $z\in\Re^n$. When $z_2=0$, it is clear that $O\prec\nabla P_\mu(z)\prec I$ since $\nabla P_\mu(z)=\hat{g}'(z_1/\mu)I$ from (3.3.9) and $0<\hat{g}'(z_1/\mu)<1$ from (3.3.3). So we only consider the case where $z_2\neq 0$. Noticing that $\nabla P_\mu(z)$ is given by (3.3.9), in order to show $\nabla P_\mu(z)\succ O$, it is sufficient to show that b_μ is positive and the Schur complement of $\nabla P_\mu(z)$ with respect to b_μ is positive definite. Since $b_\mu=(1/2)(\hat{g}'(\lambda_1/\mu)+\hat{g}'(\lambda_2/\mu))$, we have $b_\mu>0$ from (3.3.3). On the other hand, the Schur complement of $\nabla P_\mu(z)$ with respect to b_μ is given by

$$\left\{a_\mu I+(b_\mu-a_\mu)\frac{z_2z_2^T}{\|z_2\|^2}\right\}-\frac{c_\mu^2}{b_\mu}\frac{z_2z_2^T}{\|z_2\|^2}\;=\;a_\mu\left(I-\frac{z_2z_2^T}{\|z_2\|^2}\right)+\frac{b_\mu^2-c_\mu^2}{b_\mu}\frac{z_2z_2^T}{\|z_2\|^2}.$$

If $a_\mu=(\hat{g}(\lambda_2/\mu)-\hat{g}(\lambda_1/\mu))/(\lambda_2/\mu-\lambda_1/\mu)\leq 0$, then the continuous differentiability of $\hat{g}$ and the mean value theorem guarantee the existence of $\tau\in[\lambda_1/\mu,\lambda_2/\mu]$ such that $\hat{g}'(\tau)\leq 0$. Since this fact contradicts (3.3.3), we have $a_\mu>0$. Moreover, we have $(b_\mu^2-c_\mu^2)/b_\mu=2/(1/\hat{g}(\lambda_1)+1/\hat{g}(\lambda_2))>0$ from (3.3.3). Furthermore, both $z_2z_2^T/\|z_2\|^2$ and $I-z_2z_2^T/\|z_2\|^2$ are positive semidefinite and their sum is the identity matrix. Hence, any positive linear combination of $z_2z_2^T/\|z_2\|^2$ and $I-z_2z_2^T/\|z_2\|^2$ is positive definite. Therefore, the Schur complement of $\nabla P_\mu(z)$ with respect to b_μ is positive definite, and hence we obtain $\nabla P_\mu(z)\succ O$. In a similar way, we can show

$$I-\nabla P_\mu(z)\;=\;\begin{pmatrix}1-b_\mu & -\dfrac{c_\mu z_2^T}{\|z_2\|}\\[2ex] -\dfrac{c_\mu z_2}{\|z_2\|} & (1-a_\mu)I-(b_\mu-a_\mu)\dfrac{z_2z_2^T}{\|z_2\|^2}\end{pmatrix}\succ O,$$

from $1-b_\mu>0$ and the positive definiteness of the Schur complement of $I-\nabla P_\mu(z)$ with respect to $1-b_\mu$.

Secondly, we show the nonsingularity of $\nabla H_{\mu,\varepsilon}(x,y)^T$ instead of $\nabla H_{\mu,\varepsilon}(x,y)$. Let us denote $\mathcal{P} := \nabla P_\mu(z)$ and $\mathcal{F} := \nabla f(x)$ for convenience. Then $\nabla H_{\mu,\varepsilon}(x,y)^T$ can be written as

$$\nabla H_{\mu,\varepsilon}(x,y)^T = \begin{pmatrix} I-\mathcal{P} & \mathcal{P} \\ \mathcal{F}+\varepsilon I & -I \end{pmatrix}.$$

Let $\xi, \eta \in \Re^n$ satisfy

$$\nabla H_{\mu,\varepsilon}(x,y)^T \begin{pmatrix} \xi \\ \eta \end{pmatrix} = 0,$$

that is,

$$(I-\mathcal{P})\xi + \mathcal{P}\eta = 0, \tag{B.1}$$

$$(\mathcal{F}+\varepsilon I)\xi - \eta = 0. \tag{B.2}$$

Multiplying the left-hand side of (B.1) by $\mathcal{P}^{-1}$ and combining with (B.2), we have

$$(\mathcal{P}^{-1} - I + \mathcal{F} + \varepsilon I)\xi = 0.$$

Since $O \prec \mathcal{P} \prec I$ implies $\mathcal{P}^{-1} \succ I$ and monotonicity of f implies $\mathcal{F} \succeq O$, $\mathcal{P}^{-1} - I + \mathcal{F} + \varepsilon I$ is positive definite. So we have $\xi = 0$, and then $\eta = 0$ from (B.2). Hence, $\nabla H_{\mu,\varepsilon}(x,y)^T$ is nonsingular, that is, $\nabla H_{\mu,\varepsilon}(x,y)$ is nonsingular. ■

Proposition B.2 *If $f : \Re^n \to \Re^n$ is monotone, then, for any $\mu > 0$ and $\varepsilon \geq 0$, every stationary point $(\overline{x}, \overline{y})$ of the function $\Psi_{\mu,\varepsilon}$ satisfies $\Psi_{\mu,\varepsilon}(\overline{x}, \overline{y}) = 0$.*

Proof. Note that $\nabla \Psi_{\mu,\varepsilon}(\overline{x}, \overline{y}) = \nabla H_{\mu,\varepsilon}(\overline{x}, \overline{y})\, H_{\mu,\varepsilon}(\overline{x}, \overline{y}) = 0$. By Proposition B.1, $\nabla H_{\mu,\varepsilon}(\overline{x}, \overline{y})$ is nonsingular. Hence, we have $H_{\mu,\varepsilon}(\overline{x}, \overline{y}) = 0$, that is, $\Psi_{\mu,\varepsilon}(\overline{x}, \overline{y}) = (1/2)\|H_{\mu,\varepsilon}(\overline{x}, \overline{y})\|^2 = 0$. ■

We proceed to showing the global convergence property of Algorithm 3.4.1 by extending the result of [32] for NCP to SOCCP. To this end, we give two lemmas. The first lemma implies that $\Psi_{\mu,\varepsilon}$ is uniformly continuous on a compact set not only in x and y but also in μ and ε.

Lemma B.3 *Let $C \subset \Re^n \times \Re^n$ be a compact set. Then, for any $\delta > 0$, there exist $\varepsilon' > 0$ and $\mu' > 0$ such that*

$$|\Psi_{\mu,\varepsilon}(x,y) - \Psi_{\mathrm{NR}}(x,y)| \leq \delta$$

for any $(x,y) \in C$, $\varepsilon \in [0, \varepsilon']$ and $\mu \in [0, \mu']$.

Proof. Define the function $\Omega : \Re^n \times \Re^n \times [0,+\infty) \times [0,+\infty) \to \Re$ by $\Omega(x,y,\mu,\varepsilon) := \Psi_{\mu,\varepsilon}(x,y)$. Then, Ω is continuous and satisfies $\Omega(x,y,0,0) = \Psi_{\mathrm{NR}}(x,y)$. Since any continuous function is uniformly continuous on a compact set, Ω is uniformly continuous on $C \times [0,\varepsilon'] \times [0,\mu']$. ■

The next lemma is known as the Mountain Pass Theorem, which is useful for our analysis. For more detail, see Theorem 9.2.7 in [90].

Lemma B.4 (Mountain Pass Theorem) *Let $\theta : \Re^n \to \Re$ be a continuously differentiable and level-bounded function. Let $C \subset \Re^n$ be a nonempty and compact set and let m be the minimum value of θ on the boundary of C, that is,*

$$m := \min_{x \in \partial C} \theta(x).$$

Assume that there exist points $p \in C$ and $q \notin C$ such that $\theta(p) < m$ and $\theta(q) < m$. Then, there exists a point $r \in \Re^n$ such that $\nabla\theta(r) = 0$ and $\theta(r) \geq m$.

Finally, by using the above propositions and lemmas, we establish the global convergence of Algorithm 3.4.1.

Theorem B.2 (Theorem 3.4.1) *Let $f : \Re^n \to \Re^n$ be a monotone function. Assume that the solution set $\mathcal{S}$ of SOCCP* (3.3.1) *is nonempty and bounded. Let $\{(x^{(k)}, y^{(k)})\}$ be a sequence generated by Algorithm* 3.4.1. *Then, $\{(x^{(k)}, y^{(k)})\}$ is bounded, and every accumulation point is a solution of* (3.3.1).

Proof. From a simple continuity argument, we can easily show that every accumulation point of $\{(x^{(k)}, y^{(k)})\}$ is a solution of SOCCP (3.3.1). So we only show the boundedness of $\{(x^{(k)}, y^{(k)})\}$. For a contradiction purpose, we assume that $\{(x^{(k)}, y^{(k)})\}$ is not bounded. Then, there exists a subsequence $\{(x^{(k)}, y^{(k)})\}_{k \in K}$ such that $\lim_{k \to \infty, k \in K} \|(x^{(k)}, y^{(k)})\| = \infty$. Moreover, noticing that there exists a compact set $C \subset \Re^n \times \Re^n$ such that $\mathcal{S} \subset \operatorname{int} C$ because of the boundedness of $\mathcal{S}$, we have the following two facts:

(a) $(x^{(k)}, y^{(k)}) \notin C$ for all $k \in K$ sufficiently large;

(b) $\overline{m} := \min_{(x,y) \in \partial C} \Psi_{\text{NR}}(x, y) > 0.$

Now, applying Lemma B.3 with $\delta := \overline{m}/4 > 0$, we have

$$\Psi_{\mu_k, \varepsilon_k}(x, y) - \Psi_{\text{NR}}(x, y) \leq \frac{1}{4}\overline{m} \tag{B.3}$$

and

$$\Psi_{\mu_k, \varepsilon_k}(x, y) - \Psi_{\text{NR}}(x, y) \geq -\frac{1}{4}\overline{m} \tag{B.4}$$

for any $(x, y) \in C$ and $k \in K$ sufficiently large. Let $(\overline{x}, \overline{y}) \in \mathcal{S} \subset C$ be a solution of SOCCP. Then, from (B.3), we have

$$\Psi_{\mu_k, \varepsilon_k}(\overline{x}, \overline{y}) - \Psi_{\text{NR}}(\overline{x}, \overline{y}) = \Psi_{\mu_k, \varepsilon_k}(\overline{x}, \overline{y}) \leq \frac{1}{4}\overline{m} \tag{B.5}$$

for all $k \in K$ sufficiently large. On the other hand, letting $(\tilde{x}^{(k)}, \tilde{y}^{(k)})$ be a minimizer of $\Psi_{\mu_k, \varepsilon_k}(x, y)$ on the set ∂C, we have, for all $k \in K$ sufficiently large,

$$\begin{aligned} \min_{(x,y) \in \partial C} \Psi_{\mu_k, \varepsilon_k}(x, y) &= \Psi_{\mu_k, \varepsilon_k}(\tilde{x}^{(k)}, \tilde{y}^{(k)}) \\ &\geq -\frac{1}{4}\overline{m} + \Psi_{\text{NR}}(\tilde{x}^{(k)}, \tilde{y}^{(k)}) \\ &\geq -\frac{1}{4}\overline{m} + \overline{m} \\ &= \frac{3}{4}\overline{m}, \end{aligned} \tag{B.6}$$

where the first inequality follows from (B.4) and the second inequality follows from (b) and $(\tilde{x}^{(k)}, \tilde{y}^{(k)}) \in \partial C$. Furthermore, since $\Psi_{\mu_k, \varepsilon_k}(x^{(k)}, y^{(k)}) \leq \alpha_k$ from Step 2 of Algorithm 3.4.1, we have

$$\Psi_{\mu_k, \varepsilon_k}(x^{(k)}, y^{(k)}) \leq \frac{1}{4}\overline{m} \tag{B.7}$$

for all $k \in K$ sufficiently large. Now, let $\overline{k} \in K$ be a sufficiently large integer satisfying (a), (B.5), (B.6) and (B.7). Then, applying Lemma B.4 to $\Psi_{\mu_{\bar{k}}, \varepsilon_{\bar{k}}}$ with $p := (\overline{x}, \overline{y})$, $q := (x^{(\bar{k})}, y^{(\bar{k})})$ and $m := \min_{(x,y)\in\partial C} \Psi_{\mu_{\bar{k}}, \varepsilon_{\bar{k}}}(x, y) \geq (3/4)\overline{m}$, we obtain the existence of $(\hat{x}^{(k)}, \hat{y}^{(k)}) \in \Re^n \times \Re^n$ such that

$$\nabla\Psi_{\mu_{\bar{k}}, \varepsilon_{\bar{k}}}(\hat{x}^{(k)}, \hat{y}^{(k)}) = 0 \quad \text{and} \quad \Psi_{\mu_{\bar{k}}, \varepsilon_{\bar{k}}}(\hat{x}^{(k)}, \hat{y}^{(k)}) \geq \frac{3}{4}\overline{m} > 0.$$

However, this contradicts Proposition B.2. Hence, $\{(x^{(k)}, y^{(k)})\}$ is bounded. ∎

Appendix C

Detailed proof of Proposition 3.4.2

In this appendix, we give a more detailed proof of Proposition 3.4.2, which shows the strong semismoothness of $P_{\mathcal{K}^n}$ defined by (2.2.7). To this end, we first study some differential properties of function $P_{\mathcal{K}^n}$ defined by (2.2.7). For the sake of convenience, we divide $\Re^n$ into six subsets as follows:

$$\begin{array}{lll}
S_1 := \{z \,|\, 0 < \lambda_1 \leq \lambda_2\} = \{z \,|\, z_1 > \|z_2\|\} & = \operatorname{int} \mathcal{K}^n \\
S_2 := \{z \,|\, \lambda_1 < 0 < \lambda_2\} = \{z \,|\, -\|z_2\| < z_1 < \|z_2\|\} & = \Re^n \backslash (\mathcal{K}^n \cup -\mathcal{K}^n) \\
S_3 := \{z \,|\, \lambda_1 \leq \lambda_2 < 0\} = \{z \,|\, z_1 < -\|z_2\|\} & = -\operatorname{int} \mathcal{K}^n \\
S_4 := \{z \,|\, \lambda_1 = 0 < \lambda_2\} = \{z \,|\, 0 < z_1 = \|z_2\|\} & = \operatorname{bd} \mathcal{K}^n \backslash \{0\} \\
S_5 := \{z \,|\, \lambda_1 < 0 = \lambda_2\} = \{z \,|\, -\|z_2\| = z_1 < 0\} & = -\operatorname{bd} \mathcal{K}^n \backslash \{0\} \\
S_6 := \{z \,|\, \lambda_1 = 0 = \lambda_2\} = \{z \,|\, z_1 = \|z_2\| = 0\} & = \{0\},
\end{array} \tag{C.1}$$

where λ_1 and λ_2 are the spectral values of z given by (2.2.4), and bd and int denote the boundary and the interior, respectively. Note that $\cup_{i=1}^6 S_i = \Re^n$, $S_i \cap S_j = \emptyset$ $(i \neq j)$, and that S_1, S_2 and S_3 are open sets. Define, moreover, the functions $\Pi_1 : \Re^n \to \Re^n$, $\Pi_2 : D := \{z \in \Re^n | z_2 \neq 0\} \to \Re^n$ and $\Pi_3 : \Re^n \to \Re^n$ as

$$\Pi_1(z) := \lambda_1 u^{\{1\}} + \lambda_2 u^{\{2\}} = z \tag{C.2}$$

$$\Pi_2(z) := \lambda_2 u^{\{2\}} = \frac{1}{2}(z_1 + \|z_2\|)\left(1, \frac{z_2}{\|z_2\|}\right) \tag{C.3}$$

$$\Pi_3(z) := 0. \tag{C.4}$$

Then (2.2.8) is rewritten as

$$P_{\mathcal{K}^n}(z) = \begin{cases}
\Pi_1(z) & (z \in S_1) \\
\Pi_2(z) & (z \in S_2) \\
\Pi_3(z) & (z \in S_3) \\
\Pi_1(z) = \Pi_2(z) & (z \in S_4) \\
\Pi_2(z) = \Pi_3(z) & (z \in S_5) \\
\Pi_1(z) = \Pi_3(z) & (z \in S_6)\,.
\end{cases} \tag{C.5}$$

The following proposition refers to the differential properties of Π_1, Π_2 and Π_3.

Proposition C.1 *Let $z = (z_1, z_2) \in \Re \times \Re^{n-1}$ and $D = \{z \in \Re^n | z_2 \neq 0\}$. Let Π_1, Π_2 and Π_3 be the functions defined by* (C.2), (C.3) *and* (C.4), *respectively. Then, Π_1 and Π_3 are differentiable on $\Re^n$, and Π_2 is differentiable on D. Moreover, their Jacobians are given as*

$$\nabla \Pi_1(z) = I \tag{C.6}$$

$$\nabla \Pi_2(z) = \frac{1}{2}\left(1 + \frac{z_1}{\|z_2\|}\right) I + \frac{1}{2\|z_2\|} \begin{pmatrix} -z_1 & z_2^T \\ z_2 & -\dfrac{z_1 z_2 z_2^T}{\|z_2\|^2} \end{pmatrix} \tag{C.7}$$

$$\nabla \Pi_3(z) = O. \tag{C.8}$$

Proof. It is evident from $\Pi_1(z) = z$ and $\Pi_3(z) = 0$ that $\nabla \Pi_1(z) = I$ and $\nabla \Pi_3(z) = O$. By using the fact that

$$\nabla_w \left(\frac{w}{\|w\|}\right) = \frac{1}{\|w\|}\left(I - \frac{ww^T}{\|w\|^2}\right)$$

for any $w \in \Re^{n-1} \backslash \{0\}$, we have

$$\nabla_z \left(1, \frac{z_2^T}{\|z_2\|}\right)^T = \frac{1}{\|z_2\|} \begin{pmatrix} 0 & 0 \\ 0 & I - \dfrac{z_2 z_2^T}{\|z_2\|^2} \end{pmatrix}$$

for $z_2 \neq 0$. Moreover, since

$$\nabla_z (z_1 + \|z_2\|) = \left(1, \frac{z_2^T}{\|z_2\|}\right)^T,$$

we have

$$\begin{aligned}
\nabla \Pi_2(z) &= \frac{1}{2}\nabla_z (z_1 + \|z_2\|) \left(1, \frac{z_2^T}{\|z_2\|}\right) + \frac{1}{2}(z_1 + \|z_2\|) \nabla_z \left(1, \frac{z_2^T}{\|z_2\|}\right)^T \\
&= \frac{1}{2}\left(1, \frac{z_2^T}{\|z_2\|}\right)^T \left(1, \frac{z_2^T}{\|z_2\|}\right) + \frac{z_1 + \|z_2\|}{2\|z_2\|} \begin{pmatrix} 0 & 0 \\ 0 & I - \dfrac{z_2 z_2^T}{\|z_2\|^2} \end{pmatrix} \\
&= \frac{1}{2} \begin{pmatrix} 1 & \dfrac{z_2^T}{\|z_2\|} \\ \dfrac{z_2}{\|z_2\|} & \left(1 + \dfrac{z_1}{\|z_2\|}\right) I - \dfrac{z_1}{\|z_2\|} \dfrac{z_2 z_2^T}{\|z_2\|^2} \end{pmatrix} \\
&= \frac{1}{2}\left(1 + \frac{z_1}{\|z_2\|}\right) I + \frac{1}{2\|z_2\|} \begin{pmatrix} -z_1 & z_2^T \\ z_2 & -\dfrac{z_1 z_2 z_2^T}{\|z_2\|^2} \end{pmatrix}.
\end{aligned}$$

■

Next, we give a lemma on the strong semismoothness of the three functions Π_1, Π_2 and Π_3 that constitute $P_{\mathcal{K}^n}$.

Lemma C.1 *Let Π_1, Π_2 and Π_3 be the functions defined by* (C.2), (C.3) *and* (C.4), *respectively. Then, Π_1 and Π_3 are strongly semismooth at any $z \in \Re^n$, and Π_2 is strongly semismooth at any $z \in \Re^n$ such that $z_2 \neq 0$.*

Proof. Obviously, Π_1 and Π_3 are strongly semismooth since their Jacobians are constants. So we only consider the function Π_2. Let $z = (z_1, z_2) \in \Re \times \Re^{n-1}$ be an arbitrary vector such that $z_2 \neq 0$, and $d = (d_1, d_2) \in \Re \times \Re^{n-1}$ be an arbitrarily small and nonzero vector such that $z_2 + d_2 \neq 0$. Moreover, let V be an arbitrary element of $\partial \Pi_2(z + d)$. Note that Π_2 is twice differentiable at z_2 and $z_2 + d_2$ since $z_2 \neq 0$ and $z_2 + d_2 \neq 0$. Then we have

$$\begin{aligned} \|V^T d - \Pi_2'(z\,;d)\| &= \|\nabla \Pi_2(z+d)^T d - \nabla \Pi_2(z)^T d\| \\ &\leq \|\nabla \Pi_2(z+d) - \nabla \Pi_2(z)\|\|d\| \\ &= O(\|d\|^2), \end{aligned}$$

which proves the strong semismoothness of Π_2. ∎

By using the above proposition and lemma, we show the strong semismoothness of $P_{\mathcal{K}^n}$.

Proposition C.2 (Proposition 3.4.2) *The function $P_{\mathcal{K}^n}$ defined by* (2.2.7) *is strongly semismooth at any point $z \in \Re^n$.*

Proof. Let S_i $(i = 1, \ldots, 6)$ be defined by (C.1). Let Π_1, Π_2 and Π_3 be defined by (C.2), (C.3) and (C.4), respectively. When $z \in S_1$, for any $d \in \Re^n$ sufficiently small, $z + d$ is an element of S_1, because S_1 is an open set. Hence, $P_{\mathcal{K}^n}$ is strongly semismooth at any $z \in S_1$ by (C.5) and Lemma C.1. In a similar way, we can show the strong semismoothness of $P_{\mathcal{K}^n}$ at any $z \in S_2 \cup S_3$.

When $z \in S_4$, for any $d \in \Re^n$ sufficiently small, we have $z + d \in S_1 \cup S_2 \cup S_4$. Let V be an arbitrary element of $\partial P_{\mathcal{K}^n}(z + d)$. If $z + d \in S_1$, it then follows from Lemma C.1 that $V^T d - P_{\mathcal{K}^n}'(z\,;d) = (\nabla \Pi_1(z + d) - \nabla \Pi_1(z))^T d = O(\|d\|^2)$. Similarly, if $z + d \in S_2$, it follows that $V^T d - P_{\mathcal{K}^n}'(z\,;d) = (\nabla \Pi_2(z+d) - \nabla \Pi_2(z))^T d = O(\|d\|^2)$. If $z + d \in S_4$, then we can find a scalar $\alpha \in [0, 1]$ such that $V = (1 - \alpha)\nabla \Pi_1(z + d) + \alpha \nabla \Pi_2(z + d)$. Since $P_{\mathcal{K}^n}'(z\,;d) = \nabla \Pi_1(z)^T d = \nabla \Pi_2(z)^T d$, we have $V^T d - P_{\mathcal{K}^n}'(z\,;d) = (1-\alpha)\{\nabla \Pi_1(z+d)^T d - \nabla \Pi_1(z)^T d\} + \alpha\{\nabla \Pi_2(z+d)^T d - \nabla \Pi_2(z)^T d\} = O(\|d\|^2)$. Thus, $P_{\mathcal{K}^n}$ is strongly semismooth at $z \in S_4$. In a similar way, we can show the strong semismoothness of $P_{\mathcal{K}^n}$ at $z \in S_5$.

Finally, we consider the case where $z \in S_6$, that is, $z = 0$. From (C.2), (C.3), (C.4) and (C.5), $P_{\mathcal{K}^n}$ is directionally differentiable at 0, and its directional derivative along the direction $d = (d_1, d_2) \in \Re \times \Re^{n-1} \backslash \{(0, 0)\}$ is given by

$$\begin{aligned} P_{\mathcal{K}^n}'(0\,;d) &= \lim_{h \downarrow 0} P_{\mathcal{K}^n}(hd)/h \\ &= P_{\mathcal{K}^n}(d). \end{aligned} \tag{C.9}$$

When $d \in S_1$, it follows from (C.5), (C.6) and (C.9) that $V^T d - P_{\mathcal{K}^n}'(0\,;d) = \nabla \Pi_1(d)^T d - \Pi_1(d) = Id - d = 0$. When $d \in S_2$, it follows from (C.5), (C.7) and (C.9) that

$$V^T d - P_{\mathcal{K}^n}'(0\,;d)$$

$$
\begin{aligned}
&= \nabla\Pi_2(d)^T d - \Pi_2(d) \\
&= \frac{1}{2}\begin{pmatrix} 1 & \dfrac{d_2^T}{\|d_2\|} \\ \dfrac{d_2}{\|d_2\|} & \left(1+\dfrac{d_1}{\|d_2\|}\right) I - \dfrac{d_1}{\|d_2\|}\dfrac{d_2 d_2^T}{\|d_2\|^2} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} - \frac{1}{2}(d_1 + \|d_2\|) \begin{pmatrix} 1 \\ \dfrac{d_2}{\|d_2\|} \end{pmatrix} \\
&= \frac{1}{2}\begin{pmatrix} d_1 + \|d_2\| \\ \dfrac{d_1 d_2}{\|d_2\|} + d_2 \end{pmatrix} - \frac{1}{2}(d_1 + \|d_2\|) \begin{pmatrix} 1 \\ \dfrac{d_2}{\|d_2\|} \end{pmatrix} \\
&= 0.
\end{aligned}
$$

When $d \in S_3$, it follows from (C.5), (C.8) and (C.9) that $V^T d - P'_{\mathcal{K}^n}(0\,;d) = \nabla\Pi_3(d)^T d - \Pi_3(d) = Od - 0 = 0$. When $d \in S_4$, any $V \in \partial P_{\mathcal{K}^n}(d)$ can be written as $V = (1-\alpha)\nabla\Pi_1(d) + \alpha\nabla\Pi_2(d)$ with $\alpha \in [0,1]$. Furthermore, we have $P'_{\mathcal{K}^n}(0\,;d) = P_{\mathcal{K}^n}(d) = \Pi_1(d) = \Pi_2(d)$ from (C.9) and (C.5). Hence, we have $V^T d - P_{\mathcal{K}^n}(0\,;d) = (1-\alpha)\{\nabla\Pi_1(d)^T d - \Pi_1(d)\} + \alpha\{\nabla\Pi_2(d)^T d - \Pi_2(d)\} = 0$. When $d \in S_5$, we can also show $V^T d - P_{\mathcal{K}^n}(0\,;d) = 0$ in a similar way. Consequently, we have $V^T d - P'_{\mathcal{K}^n}(0\,;d) = 0$ for any $d \in \Re^n \backslash \{0\}$. ■

Appendix D

Proof of Proposition 3.4.3

In Appendix C, we have shown that the function $P_{\mathcal{K}^n}$ is composed of the differentiable functions Π_1, Π_2 and Π_3. By using these functions, we give the proof of Proposition 3.4.3.

Proposition D.1 (Proposition 3.4.3) *Clarke subdifferential of the projection function $P_{\mathcal{K}^n}$ and the matrix $J_P^o(z) := \lim_{\mu \downarrow 0} \nabla P_\mu(z)$ are given as follows*:

$$\partial P_{\mathcal{K}^n}(z) = \begin{cases} \{I\} & (\lambda_1 > 0, \lambda_2 > 0) \\ \{(\lambda_2/(\lambda_1+\lambda_2))I + Z\} & (\lambda_1 < 0, \lambda_2 > 0) \\ \{O\} & (\lambda_1 < 0, \lambda_2 < 0) \\ \mathrm{co}\{I, I+Z\} & (\lambda_1 = 0, \lambda_2 > 0) \\ \mathrm{co}\{O, Z\} & (\lambda_1 < 0, \lambda_2 = 0) \\ \mathrm{co}\left(\{O\} \cup \{I\} \cup \mathcal{S}\right) & (\lambda_1 = 0, \lambda_2 = 0)\,, \end{cases} \tag{D.1}$$

$$J_P^o(z) = \begin{cases} I & (\lambda_1 > 0, \lambda_2 > 0) \\ (\lambda_2/(\lambda_1+\lambda_2))I + Z & (\lambda_1 < 0, \lambda_2 > 0) \\ O & (\lambda_1 < 0, \lambda_2 < 0) \\ I + (1-\hat{g}'(0))Z & (\lambda_1 = 0, \lambda_2 > 0) \\ \hat{g}'(0)Z & (\lambda_1 < 0, \lambda_2 = 0) \\ \hat{g}'(0)I & (\lambda_1 = 0, \lambda_2 = 0)\,, \end{cases} \tag{D.2}$$

where $(r_1, r_2) := (z_1, z_2)/\|z_2\|$,

$$Z = \frac{1}{2}\begin{pmatrix} -r_1 & r_2^T \\ r_2 & -r_1 r_2 r_2^T \end{pmatrix},\ \mathcal{S} := \left\{ \frac{1}{2}(1+\beta)I + \frac{1}{2}\begin{pmatrix} -\beta & w^T \\ w & -\beta w w^T \end{pmatrix} \middle| -1 \le \beta \le 1, \|w\| = 1 \right\}.$$

Proof. First we show (D.1). It follows from (C.2) – (C.5) and Proposition C.1 that $P_{\mathcal{K}^n}$ is continuously differentiable at any point z in the open sets S_i $(i = 1, 2, 3)$. So it suffices to consider the case where $z \in S_i$ $(i = 4, 5, 6)$.

When $z \in S_4$, from (C.1), we have $z_1 = \|z_2\|$, namely $r_1 = 1$. Note that S_4 is adjacent to S_1 and S_2 but not to S_3. Since

$$\lim_{\substack{\hat{z}\to z\\ \hat{z}\in S_1}} \nabla P_{\mathcal{K}^n}(\hat{z}) \;=\; \lim_{\substack{\hat{z}\to z\\ \hat{z}\in S_1}} \nabla \Pi_1(\hat{z}) \;=\; I \tag{D.3}$$

and

$$\lim_{\substack{\hat{z}\to z\\ \hat{z}\in S_2}} \nabla P_{\mathcal{K}^n}(\hat{z}) \;=\; \lim_{\substack{\hat{z}\to z\\ \hat{z}\in S_2}} \nabla \Pi_2(\hat{z}) \;=\; I + \frac{1}{2}\begin{pmatrix} -1 & r_2^T \\ r_2 & -r_2 r_2^T \end{pmatrix},$$

Clarke subdifferential of $P_{\mathcal{K}^n}$ at $z \in S_4$ is given by

$$\partial P_{\mathcal{K}^n}(z) \;=\; \mathrm{co}\left\{ I,\; I + \frac{1}{2}\begin{pmatrix} -1 & r_2^T \\ r_2 & -r_2 r_2^T \end{pmatrix} \right\}.$$

When $z \in S_5$, from (C.1), we have $z_1 = -\|z_2\|$, namely $r_1 = -1$. Note that S_5 is adjacent to S_2 and S_3 but not to S_1. Since

$$\lim_{\substack{\hat{z}\to z\\ \hat{z}\in S_3}} \nabla P_{\mathcal{K}^n}(\hat{z}) \;=\; \lim_{\substack{\hat{z}\to z\\ \hat{z}\in S_3}} \nabla \Pi_3(\hat{z}) \;=\; O \tag{D.4}$$

and

$$\lim_{\substack{\hat{z}\to z\\ \hat{z}\in S_2}} \nabla P_{\mathcal{K}^n}(\hat{z}) \;=\; \lim_{\hat{z}\to z} \Pi_2(\hat{z}) \;=\; \frac{1}{2}\begin{pmatrix} 1 & r_2^T \\ r_2 & r_2 r_2^T \end{pmatrix},$$

Clarke subdifferential of $P_{\mathcal{K}^n}$ at $z \in S_5$ is given by

$$\partial P_{\mathcal{K}^n}(z) \;=\; \mathrm{co}\left\{ O,\; \frac{1}{2}\begin{pmatrix} 1 & r_2^T \\ r_2 & r_2 r_2^T \end{pmatrix} \right\}.$$

Let $z \in S_6$, i.e., $z = 0$. Note that $z = 0$ is adjacent to S_1, S_2 and S_3. Similar to (D.3) and (D.4), we have

$$\lim_{\substack{\hat{z}\to 0\\ \hat{z}\in S_1}} \nabla P_{\mathcal{K}^n}(\hat{z}) \;=\; I \quad \text{and} \quad \lim_{\substack{\hat{z}\to 0\\ \hat{z}\in S_3}} \nabla P_{\mathcal{K}^n}(\hat{z}) \;=\; O.$$

We will show $\mathcal{T} = \mathcal{S}$, where

$$\mathcal{T} := \left\{ \lim_{\substack{\hat{z}\to 0\\ \hat{z}\in S_2}} \nabla \Pi_2(\hat{z}) \right\},$$

$$\mathcal{S} := \left\{ \frac{1}{2}(1+\beta)I + \frac{1}{2}\begin{pmatrix} -\beta & w^T \\ w & -\beta w w^T \end{pmatrix} \middle| \; -1 \le \beta \le 1, \|w\| = 1 \right\}.$$

First we show $\mathcal{T} \subseteq \mathcal{S}$. Let $\{z^{(k)}\} \subseteq \Re^n$ be an arbitrary sequence such that $z^{(k)} \to 0$ and $z^{(k)} \in S_2$ for all k, and denote $(r_1^{(k)}, r_2^{(k)}) := (z_1^{(k)}, z_2^{(k)})/\|z_2^{(k)}\|$. Then we have

$$\nabla \Pi_2(z^{(k)}) \;=\; \frac{1}{2}\left(1 + r_1^{(k)}\right) I + \frac{1}{2}\begin{pmatrix} -r_1^{(k)} & (r_2^{(k)})^T \\ r_2^{(k)} & -r_1^{(k)} r_2^{(k)} (r_2^{(k)})^T \end{pmatrix}.$$

Since $z^{(k)} \in S_2$ implies $-1 < r_1^{(k)} < 1$, every accumulation point of $\{r_1^{(k)}\}$ belongs to the interval $[-1, 1]$. Moreover, every accumulation point of $\{r_2^{(k)}\}$ is a unit vector since $\|r_2^{(k)}\| = 1$ for all k. Hence, we have $\mathcal{T} \subseteq \mathcal{S}$. Next we show $\mathcal{T} \supseteq \mathcal{S}$. Let J be an arbitrary element of $\mathcal{S}$. Then, there exist a scalar $\beta \in [-1, 1]$ and a unit vector w such that

$$J \;=\; \frac{1}{2}(1+\beta)I + \frac{1}{2}\begin{pmatrix} -\beta & w^T \\ w & -\beta w w^T \end{pmatrix}.$$

Let $\{\xi^{(k)}\} = \{(\xi_1^{(k)}, \xi_2^{(k)})\} \subseteq \Re \times \Re^{n-1}$ be defined by

$$\begin{aligned} \xi_1^{(k)} &:= \frac{\beta}{k+1}\left(1 - \frac{1}{k+1}\right) \\ \xi_2^{(k)} &:= \frac{w}{k+1}, \end{aligned}$$

and $\{\eta^{(k)}\} = \{(\eta_1^{(k)}, \eta_2^{(k)})\} \subseteq \Re \times \Re^{n-1}$ be defined by $(\eta_1^{(k)}, \eta_2^{(k)}) := (\xi_1^{(k)}, \xi_2^{(k)})/\|\xi_2^{(k)}\|$. Then, $\xi^{(k)} \in S_2$ for all k since $\xi_1^{(k)}/\|\xi_2^{(k)}\| = \beta(1 - 1/(k+1)) \in (-1, 1)$. Moreover, we have $\lim_{k\to\infty} \xi^{(k)} = 0$, $\lim_{k\to\infty} \eta_1^{(k)} = \beta$ and $\lim_{k\to\infty} \eta_2^{(k)} = w$. Hence, we have

$$\begin{aligned} \lim_{k\to\infty} \nabla\Pi_2(\xi^{(k)}) \;&=\; \lim_{k\to\infty} \left\{ \frac{1}{2}\left(1 + \eta_1^{(k)}\right) I + \frac{1}{2}\begin{pmatrix} -\eta_1^{(k)} & (\eta_2^{(k)})^T \\ \eta_2^{(k)} & -\eta_1^{(k)}\eta_2^{(k)}(\eta_2^{(k)})^T \end{pmatrix} \right\} \\ &= \frac{1}{2}(1+\beta)I + \frac{1}{2}\begin{pmatrix} -\beta & w^T \\ w & -\beta w w^T \end{pmatrix} \\ &= J. \end{aligned}$$

This implies $J \in \mathcal{T}$, that is, $\mathcal{T} \supseteq \mathcal{S}$. Thus, Clarke subdifferential of $P_{\mathcal{K}^n}$ at $z = 0$ is given by

$$\partial P_{\mathcal{K}^n}(z) \;=\; \mathrm{co}\,(\{O\} \cup \{I\} \cup \mathcal{S}).$$

Next, we show (D.2). Let λ_1 and λ_2 be the spectral values of z given by (2.2.4). Note that, if $z_2 \neq 0$, $\nabla P_\mu(z)$ is given by (3.3.9) with a_μ, b_μ and c_μ defined by (3.3.10). First we suppose $z_2 \neq 0$ and consider the limiting behavior of a_μ, b_μ and c_μ as $\mu \downarrow 0$. Since we have

$$a_\mu \;=\; \frac{\hat{g}(\lambda_2/\mu) - \hat{g}(\lambda_1/\mu)}{\lambda_2/\mu - \lambda_1/\mu} \;=\; \frac{\gamma_\mu(\lambda_2) - \gamma_\mu(\lambda_1)}{\lambda_2 - \lambda_1},$$

(3.4.8) implies that the limit $a_0 := \lim_{\mu\downarrow 0} a_\mu$ is given by

$$a_0 \;=\; \frac{\gamma_0(\lambda_2) - \gamma_0(\lambda_1)}{\lambda_2 - \lambda_1} \;=\; \begin{cases} 1 & (0 \le \lambda_1 < \lambda_2) \\ (1+r_1)/2 & (\lambda_1 < 0 < \lambda_2) \\ 0 & (\lambda_1 < \lambda_2 \le 0), \end{cases} \tag{D.5}$$

where the second case follows from $\lambda_2/(\lambda_2 - \lambda_1) = (z_1 + \|z_2\|)/(2\|z_2\|) = (1 + r_1)/2$. By (3.3.10) and (3.4.9), the limit $b_0 := \lim_{\mu\downarrow 0} b_\mu$ is given by

$$b_0 \;=\; \frac{1}{2}(\gamma_0^+(\lambda_2) + \gamma_0^+(\lambda_1)) \;=\; \begin{cases} 1 & (0 < \lambda_1 < \lambda_2) \\ (1 + \hat{g}'(0))/2 & (\lambda_1 = 0 < \lambda_2) \\ 1/2 & (\lambda_1 < 0 < \lambda_2) \\ \hat{g}'(0)/2 & (\lambda_1 < 0 = \lambda_2) \\ 0 & (\lambda_1 < \lambda_2 < 0), \end{cases} \tag{D.6}$$

and $c_0 := \lim_{\mu\downarrow 0} c_\mu$ is given by

$$c_0 \;=\; \frac{1}{2}(\gamma_0^+(\lambda_2) - \gamma_0^+(\lambda_1)) \;=\; \begin{cases} 0 & (0 < \lambda_1 < \lambda_2) \\ (1 - \hat{g}'(0))/2 & (\lambda_1 = 0 < \lambda_2) \\ 1/2 & (\lambda_1 < 0 < \lambda_2) \\ \hat{g}'(0)/2 & (\lambda_1 < 0 = \lambda_2) \\ 0 & (\lambda_1 < \lambda_2 < 0)\,. \end{cases} \tag{D.7}$$

Next we describe the limit $\lim_{\mu\downarrow 0} P_\mu(z)$. We first consider the case where $z \in S_1$, that is, $0 < \lambda_1 \le \lambda_2$. When $z_2 = 0$, we have $0 < \lambda_1 = \lambda_2 = z_1$. Hence, by (3.3.9) and (3.4.9), we have

$$\begin{aligned} \lim_{\mu\downarrow 0} \nabla P_\mu(z) \;&=\; \lim_{\mu\downarrow 0} \hat{g}'(z_1/\mu) I \\ &=\; \gamma_0^+(z_1) I \\ &=\; I. \end{aligned}$$

When $z_2 \neq 0$, that is, $0 < \lambda_1 < \lambda_2$, we have $a_0 = b_0 = 1$ and $c_0 = 0$ by (3.4.14) – (3.4.16). This, together with (3.3.9), yields

$$\begin{aligned} \lim_{\mu\downarrow 0} \nabla P_\mu(z) \;&=\; \begin{pmatrix} b_0 & c_0 r_2^T \\ c_0 r_2 & a_0 I + (b_0 - a_0) r_2 r_2^T \end{pmatrix} \\ &=\; I. \end{aligned}$$

We thus have $J_P^o(z) = I$ for any $z \in S_1$. When $z \in S_3$, that is, $\lambda_1 \le \lambda_2 < 0$, we can show that $J_P^o(z) = O$ in a similar way. When $z \in S_2$, that is, $\lambda_1 < 0 < \lambda_2$ and $z_2 \neq 0$, we have $a_0 = (1 + r_1)/2$ and $b_0 = c_0 = 1/2$ from (3.4.14) – (3.4.16). It then follows from (3.3.9) that

$$J_P^o(z) \;=\; \frac{1}{2}(1 + r_1) I + \frac{1}{2} \begin{pmatrix} -r_1 & r_2^T \\ r_2 & -r_1 r_2 r_2^T \end{pmatrix}.$$

We can argue the case of $z \in S_4 \cup S_5$ in a similar way. When $z \in S_6$, we have $\lambda_1 = \lambda_2 = z_1 = 0$ and $z_2 = 0$. Therefore, by (3.3.9) and (3.4.9), we have $J_P^o(z) = g'(0) I$. This completes the proof. ■

Appendix E

Well-definedness of Algorithm 3.4.2

We show that Algorithm 3.4.2 is well-defined in the sense that the number of inner iterations Steps 2.0 – 2.4 is finite at each major iteration.

Proposition E.1 *For each k,* Step 2 *of Algorithm* 3.4.2 *finds a $w^{(k+1)} := v^{(j+1)}$ for some j.*

Proof. Fix k arbitrarily. Notice that we always have $\|H_{\rm NR}(w^{(k)})\| > 0$ in Step 2. First we show the existence of a nonnegative integer m satisfying (3.4.20). Without loss of generality, we assume $\Psi_{\mu,\varepsilon}(v^{(j)}) > 0$. Noticing that

$$\begin{aligned}\nabla\Psi_{\mu_k,\varepsilon_k}(v^{(j)})^T\hat{d}^{(j)} &= H_{\mu_k,\varepsilon_k}(v^{(j)})^T\nabla H_{\mu_k,\varepsilon_k}(v^{(j)})^T\hat{d}^{(j)}\\ &= -H_{\mu_k,\varepsilon_k}(v^{(j)})^T H_{\mu_k,\varepsilon_k}(v^{(j)})\\ &= -2\Psi_{\mu_k,\varepsilon_k}(v^{(j)}),\end{aligned} \tag{E.1}$$

we have

$$\begin{aligned}&\Psi_{\mu_k,\varepsilon_k}(v^{(j)}+\tau\hat{d}^{(j)}) - (1-2\sigma\tau)\Psi_{\mu_k,\varepsilon_k}(v^{(j)})\\ &= \left\{\Psi_{\mu_k,\varepsilon_k}(v^{(j)}) + \tau\nabla\Psi_{\mu_k,\varepsilon_k}(v^{(j)})^T\hat{d}^{(j)} + o(\tau)\right\} - (1-2\sigma\tau)\Psi_{\mu_k,\varepsilon_k}(v^{(j)})\\ &= -2\tau\Psi_{\mu_k,\varepsilon_k}(v^{(j)}) + 2\sigma\tau\Psi_{\mu_k,\varepsilon_k}(v^{(j)}) + o(\tau)\\ &= \tau\left\{-2(1-\sigma)\Psi_{\mu_k,\varepsilon_k}(v^{(j)}) + o(\tau)/\tau\right\}.\end{aligned} \tag{E.2}$$

Since $1-\sigma > 0$ and $\Psi_{\mu_k,\varepsilon_k}(v^{(j)}) > 0$, the right-hand side of (E.2) is negative for all $\tau > 0$ sufficiently small. Hence, (3.4.20) holds for some integer $m \geq 0$.

Next, for the contradiction purpose, we assume that the inner iterations of Step 2 never terminate, that is,

$$\|H_{\mu_k,\varepsilon_k}(v^{(j)})\| > \beta_k > 0 \tag{E.3}$$

for all j at Step 2.4. We consider two cases (i) $\liminf_{j\to\infty}\tau_j > 0$ and (ii) $\liminf_{j\to\infty}\tau_j = 0$. In case (i), there exists $\overline{\tau} \in (0, (2\sigma)^{-1})$ such that $\tau_j \geq \overline{\tau}$ for all j. Since

$$\begin{aligned}\Psi_{\mu_k,\varepsilon_k}(v^{(j)}) &= \Psi_{\mu_k,\varepsilon_k}(v^{(j-1)} + \tau_{j-1}\hat{d}^{(j-1)})\\ &\leq (1-2\sigma\tau_{j-1})\Psi_{\mu_k,\varepsilon_k}(v^{(j-1)}),\end{aligned}$$

we have

$$\begin{aligned}\left\|H_{\mu_k,\varepsilon_k}(v^{(j)})\right\| &\leq \sqrt{1-2\sigma\tau_{j-1}}\,\left\|H_{\mu_k,\varepsilon_k}(v^{(j-1)})\right\| \\ &= \left\|H_{\mu_k,\varepsilon_k}(v^{(0)})\right\| \prod_{i=0}^{j-1}\sqrt{1-2\sigma\tau_i} \\ &\leq \left\|H_{\mu_k,\varepsilon_k}(v^{(0)})\right\| (1-2\sigma\overline{\tau})^{j/2}.\end{aligned}$$

The right-hand side of this inequality tends to 0 as $j \to \infty$, which contradicts (E.3). In case (ii), there exists a subsequence such that $\{\tau_j\}_{j\in J} \to 0$. Note that Step 2.3 implies

$$\Psi_{\mu_k,\varepsilon_k}(v^{(j)} + \rho^{m_j-1}\hat{d}^{(j)}) \;>\; (1-2\sigma\rho^{m_j-1})\Psi_{\mu_k,\varepsilon_k}(v^{(j)})$$

for all j. Since $\rho^{m_j-1} = \tau_j\rho^{-1}$, we then have

$$\begin{aligned}-2\sigma\Psi_{\mu_k,\varepsilon_k}(v^{(j)}) &< \frac{\Psi_{\mu_k,\varepsilon_k}(v^{(j)} + \tau_j\rho^{-1}\hat{d}^{(j)}) - \Psi_{\mu_k,\varepsilon_k}(v^{(j)})}{\tau_j\rho^{-1}} \\ &= \nabla\Psi_{\mu_k,\varepsilon_k}(v^{(j)})^T\hat{d}^{(j)} + o(\tau_j)/\tau_j \\ &= -2\Psi_{\mu_k,\varepsilon_k}(v^{(j)}) + o(\tau_j)/\tau_j,\end{aligned}$$

that is,

$$2(1-\sigma)\Psi_{\mu_k,\varepsilon_k}(v^{(j)}) \;<\; o(\tau_j)/\tau_j. \tag{E.4}$$

However, (E.4) cannot hold when $j \in J$ is sufficiently large, since $1-\sigma > 0$ and $\Psi_{\mu_k,\varepsilon_k}(v^{(j)}) > \beta_k^2/2 > 0$. Thus, we have a contradiction. This completes the proof. ∎

Appendix F

A sufficient condition for the solvability of linear SOCCP

In the numerical experiments of Section 3.5, we generated linear SOCCP (3.5.1) whose solution set is nonempty, based on the following theorem which shows a sufficient condition for a linear SOCCP to be solvable. This result is a natural extension of the well-known result [24, Theorem 3.1.2] for the linear complementarity problem.

Theorem F.1 *Let $M \in \Re^{n\times n}$ be a positive semidefinite matrix, and $q \in \Re^n$ be a real-valued vector. If there exists a pair $(\overline{x}, \overline{y})$ such that $\overline{x} \in \operatorname{int}\mathcal{K}$, $\overline{y} \in \operatorname{int}\mathcal{K}$ and $\overline{y} = M\overline{x} + q$, then the SOCCP*

$$x \in \mathcal{K},\ y \in \mathcal{K},\ x^T y = 0,\ y = Mx + q \tag{F.1}$$

has at least one solution.

Proof. First we consider the SOCP :

$$\begin{aligned} \text{minimize} \quad & z^T(Mz+q) \\ \text{subject to} \quad & Mz+q \in \mathcal{K},\ z \in \mathcal{K}\,, \end{aligned} \tag{F.2}$$

which is strictly feasible by assumption. Since it holds that

$$\xi \in \mathcal{K},\ \eta \in \mathcal{K} \implies \xi^T\eta \geq 0, \tag{F.3}$$

the objective function of SOCP (F.2) is bounded below on the feasible region. Therefore, there exist an optimal solution $z^* \in \Re^n$ and a multiplier vector $\lambda^* \in \Re^n$ satisfying the following KKT conditions :

$$z^* \in \mathcal{K}, \tag{F.4}$$
$$(M+M^T)z^* + q - M^T\lambda^* \in \mathcal{K}, \tag{F.5}$$
$$(z^*)^T((M+M^T)z^* + q - M^T\lambda^*) = 0, \tag{F.6}$$
$$\lambda^* \in \mathcal{K}, \tag{F.7}$$
$$Mz^* + q \in \mathcal{K}, \tag{F.8}$$
$$(\lambda^*)^T(Mz^* + q) = 0. \tag{F.9}$$

From (F.6) we have

$$-(z^*)^T M^T (z^* - \lambda^*) \;=\; (z^*)^T (Mz^* + q) \;\geq\; 0, \tag{F.10}$$

where the inequality follows from (F.4), (F.8) and (F.3). Moreover, we have

$$\begin{aligned} 0 \;&\leq\; (\lambda^*)^T ((M + M^T) z^* + q - M^T \lambda^*) \\ &=\; (\lambda^*)^T (Mz^* + q) + (\lambda^*)^T M^T (z^* - \lambda^*) \\ &=\; (\lambda^*)^T M^T (z^* - \lambda^*), \end{aligned} \tag{F.11}$$

where the inequality follows from (F.7), (F.5) and (F.3), and the last equality follows from (F.9). Hence, by (F.10) and (F.11), we have

$$(z^* - \lambda^*)^T M^T (z^* - \lambda^*) \;\leq\; 0. \tag{F.12}$$

Noticing the positive semidefiniteness of M, the inequality in (F.12) is replaced by the equality. Hence,

$$(z^*)^T M^T (z^* - \lambda^*) \;=\; (\lambda^*)^T M^T (z^* - \lambda^*) \;\geq\; 0, \tag{F.13}$$

where the inequality is due to (F.11). By (F.10) and (F.13), we have

$$(z^*)^T (Mz^* + q) \;=\; -(z^*)^T M^T (z^* - \lambda^*) \;=\; 0,$$

which implies $(x, y) := (z^*, Mz^* + q)$ is a solution of SOCCP (F.1). ∎

Bibliography

[1] F. Alizadeh, *Interior point methods in semidefinite programming with application to combinatorial optimization*, SIAM Journal on Optimization, 5 (1995), pp. 13–51.

[2] F. Alizadeh and D. Goldfarb, *Second-order cone programming*, Mathematical Programming, 95 (2003), pp. 3–51.

[3] J.-P. Aubin, *Mathematical Methods of Game and Economic Theory*, vol. 7 of Studies in Mathematics and Its Applications, North-Holland Publishing Company, Amsterdam, 1979.

[4] A. Bakushinsky and A. Goncharsky, *Ill-posed Problem : Theory and Applications*, vol. 301 of Mathematics and Its Applications, Kluwer Academic Publishers, Dordrecht, 1994.

[5] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robustness*, in Handbook of Semidefinite Programming: Theory, Algorithms and Applications, H. Wolkowicz, R. Saigal, and L. Vandenberghe, eds., Kluwer Academic Publishers, 2000, pp. 139–162.

[6] A. Ben-Tal and A. Nemirovski, *Robust convex optimization*, Mathematics of Operations Research, 23 (1998), pp. 769–805.

[7] A. Ben-Tal and A. Nemirovski, *Robust solutions of uncertain linear programs*, Operations Research Letters, 25 (1999), pp. 1–13.

[8] R. S. Bipasha, K. Yamada, N. Yamashita, and M. Fukushima, *A linearly convergent descent method for strongly monotone variational inequality problems*, Journal of Nonlinear and Convex Analysis, 4 (2003), pp. 15–23.

[9] J. V. Burke and S. Xu, *A non-interior predictor-corrector path following algorithm for the monotone linear complementarity problem*, Mathematical Programming, 87 (2000), pp. 113–130.

[10] B. Chen and P. T. Harker, *A non-interior-point continuation method for linear complementarity problems*, SIAM Journal on Matrix Analysis and Applications, 14 (1993), pp. 1168–1190.

[11] B. Chen and P. T. Harker, *Smooth approximations to nonlinear complementarity problems*, SIAM Journal on Optimization, 7 (1997), pp. 403–420.

[12] C. Chen and O. L. Mangasarian, *Smoothing methods for convex inequalities and linear complementarity problems*, Mathematical Programming, 71 (1995), pp. 51–69.

[13] C. Chen and O. L. Mangasarian, *A class of smoothing functions for nonlinear and mixed complementarity problems*, Computational Optimization and Applications, 5 (1996), pp. 97–138.

[14] J.-S. Chen, *Alternative proofs for some results of vector-valued functions associated with second-order cone*, report, Mathematics Department, National Taiwan Normal University, Taiwan, 2004.

[15] J.-S. Chen, X. Chen, and P. Tseng, *Analysis of nonsmooth vector-valued functions associated with second-order cones*, Mathematical Programming, 101 (2004), pp. 95–117.

[16] X. Chen, *Smoothing methods for complementarity problems and their applications: A survey*, Journal of the Operations Research Society of Japan, 43 (2000), pp. 32–47.

[17] X. Chen, L. Qi, and D. Sun, *Global and superlinear convergence of the smoothing Newton method and its application to general box constrained variational inequalities*, Mathematics of Computation, 67 (1998), pp. 519–540.

[18] X. Chen and Y. Ye, *On homotopy-smoothing methods for box-constrained variational inequalities*, SIAM Journal on Control and Optimization, 37 (1999), pp. 589–616.

[19] X. D. Chen, D. Sun, and J. Sun, *Complementarity functions and numerical experiments for second-order-cone complementarity problems*, Computational Optimization and Applications, 25 (2003), pp. 39–56.

[20] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Wiley, New York, 1983.

[21] M. Coelho, L. Fernandes, J. Judice, and J. Patricio, *On the solution of a spatial equilibrium problem*, Investigacao Operacional, 13 (1993), pp. 119–134.

[22] A. R. Conn, N. I. M. Gould, and P. L. Toint, *Trust-Region Methods*, Society for Industrial and Applied Mathematics, Philadelphia, 2000.

[23] R. W. Cottle and G. B. Dantzig, *Complementarity pivot theory of mathematical programming*, Linear Algebra and Its Applications, 1 (1968), pp. 103–125.

[24] R. W. Cottle, J.-S. Pang, and R. E. Stone, *The Linear Complementarity Problem*, Academic Press, San Diego, 1992.

[25] J.-P. Crouzeix, *Pseudomonotone variational inequality problems: Existence of solutions*, Mathematical Programming, 78 (1997), pp. 305–314.

[26] H. Dan, N. Yamashita, and M. Fukushima, *A superlinearly convergent algorithm for the monotone nonlinear complementarity problem without uniqueness and nondegeneracy conditions*, Mathematics of Operations Research, 27 (2002), pp. 743–753.

[27] G. B. Dantzig and R. W. Cottle, *Positive (semi-definite) matrices and mathematical programming*, Report ORC 63-18 (RR)13, University of California, May 1963.

[28] E. De Klerk, *Aspects of Semidefinite Programming: Interior Point Algorithms and Selected Applications*, vol. 65 of Applied Optimization Series, Kluwer Academic Publishers, Dordrecht, 2002.

[29] R. De Leone and O. L. Mangasarian, *Asynchronous parallel successive overrelaxation for the symmetric linear complementarity problem*, Mathematical Programming, 42 (1988), pp. 347–361.

[30] A. L. Dontchev and T. Zolezzi, *Well-Posed Optimization Problems*, Lecture Notes in Mathematics 1543, Springer-Verlag, Berlin, 1993.

[31] L. El Ghaoui and H. Lebret, *Robust solutions to least-squares problem with uncertain data*, SIAM Journal on Matrix Analysis and Applications, 18 (1997), pp. 1035–1064.

[32] F. Facchinei and C. Kanzow, *Beyond monotonicity in regularization methods for non-linear complementarity problems*, SIAM Journal on Control and Optimization, 37 (1999), pp. 1150–1161.

[33] F. Facchinei and J.-S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer-Verlag, New York, 2003.

[34] J. Faraut and A. Korányi, *Analysis on Symmetric Cones*, Clarendon Press, New York, 1994.

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

[36] A. Fischer, *A special Newton-type optimization method*, Optimization, 24 (1992), pp. 269–284.

[37] A. Fischer, *An NCP-function and its use for the solution of complementarity problems*, in Recent Advances in Nonsmooth Optimization, D. Du, L. Qi, and R. Womersley, eds., World Scientific, River Edge, NJ, 1995, pp. 88–105.

[38] D. Fudenberg and J. Tirole, *Game Theory*, MIT Press, Cambridge, MA, 1991.

[39] M. Fukushima, *Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems*, Mathematical Programming, 53 (1992), pp. 99–110.

[40] M. Fukushima, *Merit functions for variational inequality and complementarity problems*, in Nonlinear Optimization and Applications, G. Di Pillo and F. Giannessi, eds., Plenum Press, 1996, pp. 155–170.

[41] M. Fukushima, *The primal Douglas-Rachford splitting algorithm for a class of monotone mappings with application to the traffic equilibrium problem*, Mathematical Programming, 72 (1996), pp. 1–15.

[42] M. Fukushima and T. Itoh, *A dual approach to asymmetric traffic equilibrium problems*, Mathematica Japonica, 32 (1987), pp. 701–721.

[43] M. Fukushima, Z.-Q. Luo, and P. Tseng, *Smoothing functions for second-order cone complementarity problems*, SIAM Journal on Optimization, 12 (2001), pp. 436–460.

[44] D. C. Gazis, *Traffic Theory*, Kluwer Academic Publishers, Boston, 2002.

[45] R. Gibbons, *Game Theory for Applied Economists*, Princeton University Press, Princeton, NJ, 1992.

[46] G. H. Golub and J. M. Ortega, *Scientific Computing: An Introduction with Parallel Computing*, Academic Press, San Diego, 1993.

[47] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, 1996.

[48] M. S. Gowda, *On the continuity of solution map in linear complementarity problems*, SIAM Journal on Optimization, 2 (1992), pp. 619–634.

[49] M. S. Gowda and Y. Song, *On semidefinite linear complementarity problems*, 88 (2000), pp. 575–587.

[50] M. S. Gowda and Y. Song, *Some new results for the semidefinite linear complementarity problem*, SIAM Journal on Matrix Analysis and Applications, 24 (2002), pp. 25–39.

[51] A. Hadjidimos, *Accelerated overrelaxation method*, Mathematics of Computation, 32 (1978), pp. 149–157.

[52] 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), pp. 161–220.

[53] J. C. Harsanyi, *Games with incomplete information played by "Bayesian" players. Part I*, Management Science, 14 (1967), pp. 159–182.

[54] J. C. Harsanyi, *Games with incomplete information played by "Bayesian" players. Part II*, Management Science, 14 (1968), pp. 320–334.

[55] J. C. Harsanyi, *Games with incomplete information played by "Bayesian" players. Part III*, Management Science, 14 (1968), pp. 486–502.

[56] S. Hayashi, T. Yamaguchi, N. Yamashita, and M. Fukushima, *A matrix splitting method for symmetric affine second-order cone complementarity problems*, Journal of Computational and Applied Mathematics, 175 (2005), pp. 335–353.

[57] S. Hayashi, N. Yamashita, and M. Fukushima, *A combined smoothing and regularization method for monotone second-order cone complementarity problems*, SIAM Journal on Optimization, to appear.

[58] S. Hayashi, N. Yamashita, and M. Fukushima, *Robust Nash equilibria and second-order cone complementarity problems*, Journal of Nonlinear and Convex Analysis, to appear.

[59] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, 1985.

[60] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, 1991.

[61] Z. Huang, L. Qi, and D. Sun, *Sub-quadratic convergence of a smoothing Newton algorithm for the P_0– and monotone LCP*, Mathematical Programming, 99 (2004), pp. 423–441.

[62] T. Itoh, M. Fukushima, and T. Ibaraki, *An iterative method for variational inequalities with application to traffic equilibrium problems*, Journal of the Operations Research Society of Japan, 31 (1988), pp. 82–103.

[63] A. N. Iusem, *On the convergence of iterative methods for symmetric linear complementarity problems*, Mathematical Programming, 59 (1993), pp. 33–48.

[64] H. Jiang and L. Qi, *A new nonsmooth equations approach to nonlinear complementarity problems*, SIAM Journal on Optimization, 35 (1997), pp. 178–193.

[65] A. Kajii and S. Morris, *The robustness of equilibria to incomplete information*, Econometrica, 65 (1997), pp. 1283–1309.

[66] Y. Kanno, J. A. C. Martins, and A. Pinto da Costa, *Three-dimensional quasi-static frictional contact by using second-order cone linear complementarity problem*, BGE Research Report 04-01, Department of Urban and Environmental Engineering, Kyoto University, February 2004.

[67] C. Kanzow, *Some noninterior continuation methods for linear complementarity problems*, SIAM Journal on Matrix Analysis and Applications, 17 (1996), pp. 851–868.

[68] C. Kanzow and M. Fukushima, *Solving box constrained variational inequalities by using the natural residual with D-gap function globalization*, Operations Research Letters, 23 (1998), pp. 45–51.

[69] C. Kanzow and H. Pieper, *Jacobian smoothing methods for general complementarity problems*, SIAM Journal on Optimization, 9 (1999), pp. 342–373.

[70] M. Kojima, N. Megiddo, and T. Noma, *Homotopy continuation methods for nonlinear complementarity problems*, Mathematics of Operations Research, 16 (1991), pp. 754–774.

[71] M. Kojima, M. Shida, and S. Shindoh, *A predictor-corrector interior-point algorithm for the semidefinite linear complementarity problem using the Alizadeh-Haeberly-Overton search direction*, SIAM Journal on Optimization, 9 (1999), pp. 444–465.

[72] H. J. Kushner, *Heavy Traffic Analysis of Controlled Queueing and Communication Networks*, Springer, New York, 2001.

[73] C. E. Lemke, *Bimatrix equilibrium points and mathematical programming*, Management Science, 11 (1965), pp. 681–689.

[74] C. E. Lemke and J. Howson, *Equilibrium points of bimatrix games*, SIAM Journal on Applied Mathematics, 12 (1964), pp. 413–423.

[75] D. Li and M. Fukushima, *Smoothing Newton and quasi-Newton methods for mixed complementarity problems*, Computational Optimization and Applications, 17 (2000), pp. 203–230.

[76] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret, *Applications of second-order cone programming*, Linear Algebra and Its Applications, 284 (1998), pp. 193–228.

[77] Z.-Q. Luo and P. Tseng, *Error bound and convergence analysis of matrix splitting algorithms for the affine variational inequality problem*, SIAM Journal on Optimization, 2 (1992), pp. 43–54.

[78] N. Machida, M. Fukushima, and T. Ibaraki, *A multisplitting method for symmetric linear complementarity problems*, Journal of Computational and Applied Mathematics, 62 (1995), pp. 217–227.

[79] O. L. Mangasarian and R. De Leone, *Parallel successive overrelaxation methods for symmetric linear complementarity problems*, Journal of Optimization Theory and Applications, 54 (1987), pp. 437–446.

[80] R. Mifflin, *Semismooth and semiconvex functions in constrained optimization*, SIAM Journal on Control and Optimization, 15 (1977), pp. 959–972.

[81] D. Monderer and L. S. Shapley, *Potential games*, Games and Economic Behavior, 14 (1996), pp. 124–143.

[82] R. D. C. Monteiro and T. Tsuchiya, *Polynomial convergence of primal-dual algorithms for the second-order cone program based on the MZ-family of directions*, Mathematical Programming, 88 (2000), pp. 61–83.

[83] S. Morris, *Potential methods in interaction games*, working paper, Yale University, 1999.

[84] S. Morris and T. Ui, *Generalized potentials and robust sets of equilibria*, Journal of Economic Theory, to appear.

[85] M. Muramatsu and T. Suzuki, *A new second-order cone programming relaxation for MAX-CUT problems*, Journal of the Operations Research Society of Japan, 46 (2003), pp. 164–177.

[86] G. S. R. Murthy, T. Parthasarathy, and B. Sriparna, *Some recent results on the linear complementarity problem*, 19 (1998), pp. 898–905.

[87] K. G. Murty, *Linear Complementarity, Linear and Nonlinear Programming*, Heldermann Verlag, Berlin, 1988.

[88] R. B. Myerson, *Game Theory: Analysis of Conflict*, Harvard University Press, Cambridge, MA, 1991.

[89] M. J. Osborne and A. Rubinstein, *A Course in Game Theory*, MIT Press, Cambridge, MA, 1994.

[90] R. S. Palais and C.-L. Terng, *Critical Point Theory and Submanifold Geometry*, Lecture Notes in Mathematics 1353, Springer-Verlag, Berlin, 1988.

[91] J.-S. Pang and D. Ralph, *Piecewise smoothness, local invertibility, and parametric analysis of normal maps*, Mathematics of Operations Research, 21 (1996), pp. 401–426.

[92] J.-S. Pang, D. Sun, and J. Sun, *Semismooth homeomorphisms and strong stability of semidefinite and lorentz cone complementarity problems*, Mathematics of Operations Research, 28 (2003), pp. 39–63.

[93] J. M. Peng and M. Fukushima, *A hybrid Newton method for solving the variational inequality problem via the D-gap function*, Mathematical Programming, 86 (1999), pp. 367–386.

[94] L. Portugal and J. Júdice, *A hybrid algorithm for the solution of a single commodity spatial equilibrium model*, Computers and Operations Research, 23 (1996), pp. 623–639.

[95] H.-D. Qi, *A regularized smoothing Newton method for box constrained variational inequality problems with P_0-functions*, SIAM Journal on Optimization, 10 (1999), pp. 315–330.

[96] L. Qi, *Convergence analysis of some algorithms for solving nonsmooth equations*, Mathematics of Operations Research, 18 (1993), pp. 227–244.

[97] L. Qi and X. Chen, *A globally convergent successive approximation method for severely nonsmooth equations*, SIAM Journal on Control and Optimization, 33 (1995), pp. 402–418.

[98] L. Qi and D. Sun, *A nonsmooth version of Newton's method*, Mathematical Programming, 58 (1993), pp. 353–367.

[99] L. Qi and D. Sun, *A survey of some nonsmooth equations and smoothing Newton methods*, in Progress in Optimization: Contributions from Australasia, A. Eberhard, B. Glover, R. Hill, and D. Ralph, eds., Kluwer Academic Publisher, Boston, 1999, pp. 121–146.

[100] L. Qi and D. Sun, *Nonsmooth and smoothing methods for nonlinear complementary problems and variational inequalities*, in Encyclopedia of Optimization, C. A. Floudas and P. M. Pardalos, eds., vol. 4, Kluwer Academic Publisher, Nowell, MA, 2001, pp. 100–104.

[101] L. Qi and D. Sun, *Smoothing functions and a smoothing Newton method for complementarity and variational inequality problems*, Journal of Optimization Theory and Applications, 113 (2002), pp. 121–147.

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

[103] A. Shapiro, *On a class of nonsmooth composite functions*, Mathematics of Operations Research, 28 (2003), pp. 677–692.

[104] M. Shibata, N. Yamashita, and M. Fukushima, *The extended semidefinite linear complementarity problem: A reformulation approach*, in Nonlinear Analysis and Convex Analysis, W. Takahashi and T. Tanaka, eds., World Scientific, Singapore, 1999, pp. 326–332.

[105] D. Sun, *A regularization Newton method for solving nonlinear complementarity problems*, Applied Mathematics and Optimization, 40 (1999), pp. 315–339.

[106] D. Sun and L. Qi, *On NCP-functions*, Computational Optimization and Applications, 13 (1999), pp. 201–220.

[107] K. Taji and M. Fukushima, *A new merit function and a successive quadratic programming algorithm for variational inequality problems*, SIAM Journal on Optimization, 6 (1996), pp. 704–713.

[108] M. J. Todd, *Semidefinite optimization*, Acta Numerica, 10 (2001), pp. 515–560.

[109] K. C. Toh, R. H. Tütüncü, and M. J. Todd, *SDPT3 version 3.02 – a matlab software for semidefinite-quadratic-linear programming*, updated in December 2002. `http://www.math.nus.edu.sg/~mattohkc/sdpt3.html`.

[110] P. Tseng, *Growth behavior of a class of merit functions for the nonlinear complementarity problem*, Journal of Optimization Theory and Applications, 89 (1996), pp. 69–94.

[111] P. Tseng, *An infeasible path-following method for monotone complementarity problems*, SIAM Journal on Optimization, 7 (1997), pp. 386–402.

[112] T. Tsuchiya, *A convergence analysis of the scaling-invariant primal-dual path-following algorithms for second-order cone programming*, Optimization Methods and Software, 11&12 (1999), pp. 141–182.

[113] R. H. Tütüncü, K. C. Toh, and M. J. Todd, *Solving semidefinite-quadratic-linear programs using SDPT3*, Mathematical Programming, 95 (2003), pp. 189–217.

[114] T. Ui, *Robust equilibria of potential games*, Econometrica, 69 (2001), pp. 1373–1380.

[115] L. Vandenberghe and S. Boyd, *A primal-dual potential reduction method for problems involving matrix inequalities*, Mathematical Programming, 69 (1995), pp. 205–236.

[116] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Review, 38 (1996), pp. 49–95.

[117] H. Wolkowicz, R. Saigal, and L. Vandenberghe, eds., *Handbook of Semidefinite Programming: Theory, Algorithms and Applications*, Kluwer Academic Publishers, 2000.

[118] N. Yamashita and M. Fukushima, *On stationary points of the implicit Lagrangian for nonlinear complementarity problems*, Journal of Optimization Theory and Applications, 84 (1995), pp. 653–663.

[119] N. Yamashita and M. Fukushima, *Modified Newton methods for solving a semismooth reformulation of monotone complementarity problems*, Mathematical Programming, 76 (1997), pp. 469–491.

[120] N. Yamashita and M. Fukushima, *A new merit function and a descent method for semidefinite complementarity problems*, in Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers B.V., 1998, pp. 405–420.

[121] N. Yamashita and M. Fukushima, *On the level-boundedness of the natural residual function for variational inequality problems*, Tech. Rep. 2002-005, Department of Applied Mathematics and Physics, Kyoto University, March 2002.

[122] N. Yamashita and Z.-Q. Luo, *A nonlinear complementarity approach to multiuser power control for digital subscriber lines*, Optimization Methods and Software, 19 (2004), pp. 633–652.

[123] J. P. Zeng, D. H. Li, and M. Fukushima, *Weighted max-norm estimate of additive Schwarz iteration scheme for solving linear complementarity problems*, Journal of Computational and Applied Mathematics, 131 (2001), pp. 1–14.