

### **Copyright Undertaking**

This thesis is protected by copyright, with all rights reserved.

#### By reading and using the thesis, the reader understands and agrees to the following terms:

- 1. The reader will abide by the rules and legal ordinances governing copyright regarding the use of the thesis.
- 2. The reader will use the thesis for the purpose of research or private study only and not for distribution or further reproduction or any other purpose.
- 3. The reader agrees to indemnify and hold the University harmless from and against any loss, damage, cost, liability or expenses arising from copyright infringement or unauthorized usage.

If you have reasons to believe that any materials in this thesis are deemed not suitable to be distributed in this form, or a copyright owner having difficulty with the material being included in our database, please contact <a href="https://www.lbsys@polyu.edu.hk">lbsys@polyu.edu.hk</a> providing details. The Library will look into your claim and consider taking remedial action upon receipt of the written requests.

Pao Yue-kong Library, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

http://www.lib.polyu.edu.hk

# Generalized Newton-Type Methods and Their Applications

by

CHEN LING

A thesis submitted in partial fulfilment of the requirements for the Degree of Doctor of Philosophy in Department of Applied Mathematics The Hong Kong Polytechnic University

> Supervisor: Chair Professor LI QUN QI Co-supervisor: Professor XIAO QI YANG

> > September 2005



## CERTIFICATE OF ORIGINALITY

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of a university or other institute of higher learning, except where due acknowledgment is made in the text.

CHEN LING

## Contents

1 Preview			1	
	1.1 Generalized Newton-Type Methods		e Methods	1
		1.1.1 Semismooth Newt	ton Method $\ldots$	2
		1.1.2 Smoothing Newto	on Method	7
	1.2	Semi-Infinite Programming Problems		10
		1.2.1 Optimality Condi	tions $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$	12
		1.2.2 Numerical Method	ds for SIP Problems	14
	1.3	Generalized Semi-Infinite	Programming Problems	19
	1.4	Option Price Interpolatio	on Problems	20
	1.5	Spectral Estimation Problems		21
	1.6	Notation		23
<b>2</b>	A Class of Integral Functions		<b>24</b>	
	2.1	Introduction and Motivat	tion $\ldots$	24
	2.2	Differentiability of Integr	al Functions	25
	2.3	Semismoothness of Integr	ral Functions	31

	2.4	Smoot	thing Approximation Functions	42
3	Nur	nerica	l Methods for SIP Problems	49
	3.1	Introd	luction	49
	3.2	A Smo	bothing SQP Algorithm	52
		3.2.1	Some Preliminaries	53
		3.2.2	Smoothing SQP Algorithm	54
		3.2.3	Convergence Analysis	58
		3.2.4	Preliminary Numerical Examples	61
	3.3	A Smo	bothing Projected Newton-Type Algorithm	63
		3.3.1	A Constrained Equation Reformulation of KKT System	63
		3.3.2	Smoothing Projected Newton-Type Algorithm	65
		3.3.3	Convergence Analysis	72
		3.3.4	Preliminary Numerical Examples	76
	3.4	A Smo	bothing Newton-Type Algorithm	81
		3.4.1	A Semismooth Equation Reformulation of KKT System	81
		3.4.2	Smoothing Newton-Type Algorithm	83
		3.4.3	Convergence Analysis	89
		3.4.4	Preliminary Numerical Examples	93
	3.5	Some	Comments	98
4	AN	/lethod	l for Solving Large Scale SIP Problems	100
	4.1	Introd	luction	100

	4.2	A Truncated Projected Newton-Type Algorithm	101
	4.3	B Convergence Analysis	
	4.4	4 Implementation and Numerical Tests	
		4.4.1 Implementation of Algorithm 4.2.1	117
		4.4.2 Numerical Results	118
	4.5	Some Comments	121
<b>5</b>	$\mathbf{A}$	A Smoothing Implicit Programming Approach for Solving a Class of	
	Stochastic Generalized Semi-Infinite Programming Problems		122
	5.1	Introduction	122
	5.2	A New Reformulation	124
	5.3	Smoothing Approximation for $\tilde{P}$	126
	5.4	Algorithm and Its Convergence Analysis	137
	5.5	Some Remarks	141
6	$\frac{4}{3}$ -O	$\frac{4}{2}$ -Order Convergence of the Generalized Newton Method for Solvin	
	the No-Arbitrage Option Price Interpolation Problem		144
	6.1	Introduction	144
	6.2	Convergence Analysis	147
	6.3	A Damped Version of the Generalized Newton Method	151
7	7 A Newton Type Method for $L_2$ Spectral Estimation		157
	7.1	Introduction	157
	7.2	Dual and Its Reformulation	161

8	Cor	clusions and Suggestions for Future Studies	183
	7.5	Numerical Results	174
	7.4	Globalized Newton-Type Method and Its Convergence	172
	7.3	Some Properties of $F$	163

## Acknowledgments

First and foremost, I wish to express my deep gratitude to my supervisor Professor Li Qun Qi for introducing me to the research topics of this thesis. During the past nearly four years, Prof. Qi, as a brilliant mathematician, has never ceased to inspire; as a kind teacher, has never lacked encouraging words; as a dedicated scholar, has never yielded to challenges. He has been an exemplary role model for my work and study in future.

I would like to express my sincere appreciation to my co-supervisor Professor Xiao Qi Yang for his friendship, constant encouragement and help. It has been a precious experience for me to study for my PhD degree under two professors's guidance in the Department of Applied Mathematics at The Hong Kong Polytechnic University.

I am greatly indebted to Professor A. Shapiro (Georgia Institute of Technology, USA), Professor M. Fukushima (Kyoto University, Japan) and Professor K. L. Teo (Curtin University of Technology, Australia), who gave me much help. I also wish to thank Professor Soonyi Wu (National Cheng-Kung University, Taiwan), Professor Xiaojun Chen (Hirosaki University, Japan) and Dr. Guanglu Zhou (University of Birming-ham, UK) for all their help and support during my time for my PhD degree. Especially, I had many fruitful and stimulating discussions on our papers with Prof. Wu and Dr. Zhou. I also thank Professor Qin Ni (Nanjing University of Aeronautics and Astronautics, China), Professor Xiaojiao Tong (Changsha University of Science and Technology, China) and Professor Hongxia Yin (Chinese Academy of Sciences, China) for their help during my PhD studies, and Mr. Kai Zhang for his help in studying computer programming. In addition, I would like to thank a number of editors and anonymous referees from several journals, who have given me valuable comments and suggestions which have contributed to a better presentation of the thesis. I wish to thank all my friends at The Hong Kong Polytechnic University. They made my stay at Hong Kong very enjoyable. In particular, to Dr. Donghui Li, Dr. Xinmin Yang, Dr. Shengjie Li, Dr. Xuexing Huang, Dr. Yiju Wang, Dr. Yuying Zhou, Mr. Fei Wang and Mrs. Vera Roshchina. I am grateful to them for their friendship. Thanks are also due to the Research Committee of The Hong Kong Polytechnic University for offering me financial support over the nearly four years. I sincerely thank Professor Shuihung Hou and Mrs. Fanny(The Hong Kong Polytechnic University) for their kindness and support. I shall always treasure the stimulating research environment and the excellent research facilities of the Department of Applied Mathematics of The Hong Kong Polytechnic University.

And finally, my deepest gratitude goes to my parents, my wife, Lingling Du, and my daughter, Xiaotong Ling, for their sacrifices and encouragement. Without their love and support this thesis would not be completed on time.

## Abstract

The main purposes of this thesis are to solve the semi-infinite programming (SIP) problems, the option price interpolation problems and the  $L_2$  spectral estimation problems by using some generalized Newton methods.

Our proposed methods have the following three features:

- (1) At each iteration, only a system of linear equations needs to be solved;
- (2) These methods have Global convergence;
- (3) These methods are shown to be locally superlinearly convergent.

We also present a smoothing implicit programming method to solve the generalized semi-infinite programming (GSIP) problem with uncertainty.

The main contributions of this thesis are as follows.

We introduce a class of integral functions which arises from many applications such as nonsmooth equation reformulations of the option price problems, the SIP problems and the  $L_2$  spectral estimation problems. We investigate the differentiability, semismoothness and smoothing approximation properties of this class of integral functions. This content is mainly based on the papers 1, 3 and 4 in Underlying Papers.

We introduce four kinds of algorithms for solving SIP problems. First, we present a smoothing sequential quadratic programming (SQP) algorithm. At each iteration of this algorithm, we only need to solve a quadratic program which is always feasible and solvable. The global convergence of the smoothing SQP algorithm is established under some mild conditions. Further, we present a smoothing projected Newton-type algorithm and prove its global and local superlinear convergence property. However, the accumulation point of an iterative sequence generated by these algorithms above may not be a stationary point of the original SIP problem. So, we propose the third method, say, smoothing Newton-type algorithm. For this algorithm, we not only prove its global and local superlinear convergence under some mild conditions, but also show that any accumulation point of an iterative sequence generated by it is a stationary point of the original SIP problem. Finally, based on the smoothing projected Newton-type algorithm, we develop a truncated projected Newton-type algorithm which can solve large scale SIP problems with 2000 decision variables. The feasibility for all algorithms is ensured by an integral function. For all these algorithms, numerical experiments are also given. These contents are mainly based on the papers 3-6 in Underlying Papers.

We discuss a generalized semi-infinite programming problem with uncertainty. We propose a reformulation of the considered problem by using the first order optimality conditions of the second stage optimization problem and present a smoothing implicit programming method to solve the problem with finite discrete distribution. Global convergence results are obtained. This content is mainly based on the paper 2 in Underlying Papers.

For option price interpolation problem, Wang, Yin and Qi (2004) presented a generalized Newton method for solving it and established its superlinear convergence rate. We show that the proposed method has at least  $\frac{4}{3}$ -order convergence rate, and then give conditions under which this method has  $\frac{3}{2}$ -order and quadratic convergence rate. And finally, we give a damped version of the generalized Newton method and show that it is globally convergent and the convergence order is at least  $\frac{4}{3}$ . This content is mainly based on the paper 1 in Underlying Papers.

A Newton method for solving power spectrum estimation problems is proposed in Chapter 7, and it is proved that the method is at least  $1 + \frac{1}{2m}$ -order convergent rate. We also produce a globalized Newton-type method for solving the problem, which has at least  $1 + \frac{1}{2m}$ -order convergence rate. This content is mainly based on the paper 7 in Underlying Papers.

## **Underlying Papers**

This thesis is based on the following papers written by the author during the period of stay in the Department Applied Mathematics, The Hong Kong Polytechnic University as a graduate student.

- Qi, L., Shapiro, A. and Ling, C., Differentiability and semismoothness properties of integral functions and their applications, Mathematical Programming, Ser.A, Vol.102, pp.223-248 (2005).
- 2. Ni, Q., Ling, C., Qi, L. and Teo, K.L., A truncated projected Newton-type algorithm for large scale semi-infinite programming, to appear in: SIAM Journal on Optimization.
- Yin, H.X., Ling, C. and Qi, L., Convergence rate of Newton's method for L<sub>2</sub> spectral estimation, Mathematical Programming, published online: December 30, 2005.
- Ling, C., Chen, X.J. Fukushima, M. and Qi, L., A smoothing implicit programming approach for solving a class of stochastic generalized semi-infinite programming problems, Pacific Journal of Optimization, Vol.1, pp.127-145 (2005).
- Ling, C., Qi, L., Zhou, G. L. and Wu, S. Y., Global convergence of a robust smoothing SQP method for semi-infinite programming, to appear in: Journal of Optimization Theory and Applications, Vol.129, No.1, (2006).
- 6. Qi, L., Ling, C., Tong, X. J. and Zhou, G. L., A smoothing projected Newton-type algorithm for solving semi-infinite programming, 2004, submitted.
- Ling, C., Ni, Q., Qi,L. and Wu, S.Y., A new smoothing Newton-type algorithm for solving semi-infinite programming, 2004, Technical report.

In addition, the following is a list of other papers written by the author during the period of his Ph.D study.

- Hu, Y.D. and Ling, C., The generalized optimality conditions of multiobjective programming in topological vector space, Journal of Mathematical Analysis and Applications, Vol.290, pp.363-372 (2004).
- Ling, C., Generalized tangent epiderivative and applications to set-valued map optimization, Journal of Nonlinear and Convex Analysis, Vol.3, No.3, pp.303-313 (2002).
- 3. Ling, C., Qi, L., Zhou, G. L. and Caccetta, L., *Properties of expected residual* functions arising from stochastic complementarity problems, submitted.
- Ling, C., Qi, L. and Yin, H.X., A smoothing Newton-type method for solving L<sub>2</sub> spectral estimation problem with upper-lower bounds, Technical report.
- Ling, C., Zhou, G. L. and Qi, L., On stochastic R<sub>0</sub>-type nonlinear complementarity problems, Technical report.
- Lopez, M. A., Wu, S.Y., Ling, C. and Qi, L., An infinite-dimensional mathematical programming approach to separation in L<sub>p</sub>(X, A, μ), in the working status.
- 7. Qi, L., Yin, H.X. and Ling, C., Smooth and semismooth Newton's methods for constrained approximation and estimation, in the working status.

## Chapter 1

### Preview

### 1.1 Generalized Newton-Type Methods

The classic Newton method has the following form

$$x^{k+1} = x^k - (\nabla^T G(x^k))^{-1} G(x^k)$$

where  $\nabla^T G(x)$  denotes the Jacobian of G at x, which is used to solve the smooth nonlinear equations

$$G(x) = 0,$$
 (1.1.1)

where  $G : \Re^n \to \Re^n$  is smooth (continuously differentiable) function. The Newton method is the prototype of many local, fast algorithms for solving smooth equations. Such algorithms have excellent convergence rates if the starting iterate point belongs to a suitably chosen neighborhood of the desired solution. In addition, the damped Newton and the damped Gauss-Newton methods were presented for improving the global convergence of algorithm [30, 108]. However, if G is nonsmooth, then the above classic Newton-type methods cannot be used. To solve nonsmooth nonlinear equations which arises from many applications such as nonlinear complementarity and variational inequality problems, a number of generalized Newton-type methods were proposed, see, for example [45, 63, 106, 109–112, 122, 131, 138, 188, 189]. In this section, we mainly review two classes of generalized Newton-type methods: semismooth Newton method and smoothing Newton method.

#### 1.1.1 Semismooth Newton Method

Suppose that  $G : \Re^n \to \Re^m$  is locally Lipschitz but not necessarily smooth. By Rademacher's Theorem, G is almost everywhere differentiable. Let

 $D_G := \{x \in \Re^n : G \text{ is differentiable at } x\}.$ 

Then the Clarke generalized Jacobian of G at x can be defined by

$$\partial G(x) = \operatorname{conv} \partial_B G(x),$$

where

$$\partial_B G(x) = \left\{ \lim_{x^j \to x, \ x^j \in D_G} \nabla^T G(x^j) \right\},$$

which is called the *B*-subdifferential of *G* at  $x \in \Re^n$ . The set  $\partial G(x)$  is nonempty, convex and compact for any fixed point x [23]. The nonemptyness of  $\partial G(x)$  clearly implies that  $\partial_B G(x)$  is nonempty too. Let  $G = (G_1, G_2)$  where  $G_1 : \Re^n \to \Re^{m_1}$  and  $G_2 : \Re^n \to \Re^{m_2}$ . It is easy to see that for any  $x \in \Re^n$ ,

$$\partial G(x) \subseteq \partial G_1(x) \times \partial G_2(x). \tag{1.1.2}$$

As a natural extension of the classic Newton method, Qi and Sun [131] proposed a generalized Newton method for solving the nonsmooth equations (1.1.1), in which a system of linear equations is solved at each step. This generalized Newton methods can be described as follows: Given the vector  $x^k$ , compute  $x^{k+1}$  by

$$x^{k+1} = x^k - V_k^{-1} G(x^k), (1.1.3)$$

where  $V_k \in \partial G(x^k)$ . It is clear that the iterative method (1.1.3) reduces to the classic Newton method for a system of equations if G is smooth. The classic Newton method has a favorable feature that the sequence  $\{x^k\}$  generated by (1.1.3) is locally superlinearly (quadratically) convergent to a solution  $x^*$  of (1.1.1) if  $\nabla G(x^*)$  is nonsingular (and  $\nabla G(\cdot)$  is Lipschitz continuous) [108]. However, the iterative method (1.1.3) is not convergent for nonsmooth equations (1.1.1) in general.

Superlinear convergence of the algorithm (1.1.3) was analyzed by Qi and Sun [131] based on a key concept of so-called semismoothness, and that is why this algorithm is called semismooth Newton method.

**Definition 1.1.1** Let  $G : \Re^n \to \Re^m$  be directionally differentiable at  $x \in \Re^n$ . The function G is said to be semismooth at x if

$$Qd - G'(x; d) = o(||d||), \quad d \to 0$$
 (1.1.4)

and G is said to be p-order semismooth at x if

$$Qd - G'(x; d) = O\left(\|d\|^{1+p}\right), \quad d \to 0,$$
 (1.1.5)

where  $Q \in \partial G(x+d)$  and 0 . Here, <math>o(||d||) stands for a vector function of d, which satisfies

$$\lim_{d \to 0} \frac{o(\|d\|)}{\|d\|} = 0,$$

while  $O(||d||^{1+p})$  stands for a vector function of d, which satisfies

$$||O(||d||^{1+p})|| \le M ||d||^{1+p}$$

for all d satisfying  $||d|| \leq \delta$ , and some M > 0 and  $\delta > 0$ . In particular, G is said to be strongly semismooth at x if (1.1.5) holds for p = 1.

Semismoothness was originally introduced by Mifflin [105] for functionals, which plays an important role in the global convergence theory of nonsmooth optimization, see Polak [117]. 16 years later, Qi and Sun [131] extended the concept of semismoothness to vector-valued functions.

The following two lemmas are direct results of [131] and [165].

**Lemma 1.1.1** Let  $G : \Re^n \to \Re^m$  be a locally Lipschitz function in a neighborhood of  $x \in \Re^n$ . Then the following statements are equivalent:

(i) G is semismooth at x;

(ii) for any  $Q \in \partial G(x+d), d \to 0$ ,

$$G(x+d) - G(x) - Qd = o(||d||);$$

(iii) for any  $x + d \in D_G$ ,  $d \to 0$ ,

$$G(x+d) - G(x) - \nabla^T G(x+d)d = o\left(\|d\|\right).$$

**Lemma 1.1.2** Let  $G : \Re^n \to \Re^m$  be a locally Lipschitz function in a neighborhood of  $x \in \Re^n$ . Then the following statements are equivalent:

- (i) G is p-semismooth at x;
- (ii) for any  $Q \in \partial G(x+d), d \to 0$ ,

$$G(x+d) - G(x) - Qd = O\left( ||d||^{1+p} \right);$$

(iii) for any  $x + d \in D_G$ ,  $d \to 0$ ,

$$G(x+d) - G(x) - \nabla^T G(x+d)d = O\left( ||d||^{1+p} \right).$$

Let  $G : \Re^n \to \Re^n$  be a locally Lipschitz function. The function G is said to be CD-regular at  $x \in \Re^n$  if all  $Q \in \partial G(x)$  are nonsingular. Using semismoothness, Qi and Sun [131] presented the following convergence theorem for the semismooth Newton method (1.1.3).

**Theorem 1.1.1** Suppose that  $x^*$  is a solution of G(x) = 0, G is semismooth at  $x^*$ and G is CD-regular at  $x^*$ . Then the iteration method (1.1.3) is well-defined and the sequence  $\{x^k\}$  generated by algorithm converges to  $x^*$  when  $x^0$  is chosen sufficiently close to  $x^*$ . Moreover, the convergence rate is Q-superlinear (it is called simply superlinear), i.e.,

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

If, in addition, G is p-order semismooth at  $x^*$ , then the convergence of (1.1.3) is of order 1 + p, i.e.,

$$\limsup_{k \to \infty} \frac{\left\| x^{k+1} - x^* \right\|}{\left\| x^k - x^* \right\|^{1+p}} < \infty.$$

**Remark 1.1.1** Kummer [94] independently presented a general analysis of superlinear convergence for this generalized Newton method for solving the nonsmooth equations under similar conditions used in Qi and Sun [131].

Note that the nonsingularity of  $\partial G(x^*)$  in the above theorem is somewhat restrictive in some cases. To overcome this drawback, Qi [122] presented a modified version of (1.1.3) which may be stated as follows

$$x^{k+1} = x^k - V_k^{-1} G(x^k), (1.1.6)$$

where  $V_k \in \partial_B G(x^k)$ . The difference of this version from (1.1.3) is that  $V_k$  is chosen from  $\partial_B G(x^k)$  rather than the convex hull of  $\partial_B G(x^k)$ .

To study the convergence property of the iterative method (1.1.6), the concept of so-called BD-regularity is needed. Let  $G : \Re^n \to \Re^n$  be a locally Lipschitz function. The function G is said to be BD-regular at  $x \in \Re^n$  if all  $Q \in \partial_B G(x)$  are nonsingular. Qi [122] and Pang and Qi [112] proved the following results, respectively.

**Proposition 1.1.1** Suppose that  $G : \Re^n \to \Re^n$  is locally Lipschitz continuous and G is BD-regular at  $x \in \Re^n$ . Then there exist a neighborhood N(x) of x and a constant C such that for any  $y \in N(x)$  and  $Q \in \partial_B G(y)$ , Q is nonsingular and  $||Q^{-1}|| \leq C$ .

**Proposition 1.1.2** Suppose that  $G : \Re^n \to \Re^n$  is locally Lipschitz continuous and G is BD-regular at a solution  $x^*$  of G(x) = 0. If G is semismooth at  $x^*$ , then there exist a neighborhood  $N(x^*)$  of  $x^*$  and a constant C such that for any  $x \in N(x^*)$ ,

$$||G(x)|| \ge C||x - x^*||.$$

Analogously to Theorem 1.1.1, Qi [122] established the following result.

**Theorem 1.1.2** Suppose that  $x^*$  is a solution of G(x) = 0, G is semismooth at  $x^*$ and G is BD-regular at  $x^*$ . Then the iteration method (1.1.6) is well-defined and the sequence  $\{x^k\}$  generated by the algorithm converges superlinearly to  $x^*$  when  $x^0$  is chosen sufficiently close to  $x^*$ . If, in addition, G is p-order semismooth at  $x^*$ , then the convergence of (1.1.6) is of order 1 + p.

There are also some inexact versions of (1.1.3) and (1.1.6) and their superlinear convergence theorems, see [39, 104] for details.

Pang and Qi [112] also generalized the superlinear convergence results of Dennis-Moré [29] for quasi-Newton methods for smooth equations. **Theorem 1.1.3** Assume that G is semismooth at  $x^*$  and that G is BD-regular at  $x^*$ . Let  $\{x^k\}$  be any sequence that converges to  $x^*$  with  $x^k \neq x^*$  for all k. Then  $\{x^k\}$  converges superlinearly to  $x^*$  and  $G(x^*) = 0$  if and only if

$$\lim_{k \to \infty} \frac{\left\| G(x^k) + V_k d^k \right\|}{\left\| d^k \right\|} = 0,$$

where  $V_k \in \partial_B G(x^k)$  and  $d^k = x^{k+1} - x^k$ .

Theorems 1.1.1, 1.1.2 and 1.1.3 are very important on theoretical aspect, since they generalize the convergence results of classical Newton method for smooth equations without assuming differentiability of G. Furthermore, in the last decade, the semi-smooth Newton method became a powerful tool for solving problems arising from some important mathematical programming problems such as large scale nonlinear complementarity, variational inequality and nonlinear programming problems. These are due to the nonsmooth equation reformulations of such original problems. In particular, the so-called Fischer-Burmeister function

$$\phi_{FB}(a,b) = \sqrt{a^2 + b^2} - a - b \tag{1.1.7}$$

is used to reformulate the nonlinear complementarity problem as a system of nonsmooth equations. It is well known that  $\phi_{FB}$  is not smooth, but it is strongly semismooth. It turns out that this system is semismooth, therefore, the semismooth Newton method and its convergence results can be applied to some important mathematical programming problems such as nonlinear complementarity problems, variational inequalities and KKT conditions of optimization. For example, see [112], [104], [47], [126], [134]. This may be seen in the book by Facchinei and Pang [40] and the abundant references in that book. In the recent five years, while there are still further research work on the semismooth Newton method for solving nonlinear complementarity and variational inequality problems, the semismooth Newton method has been further applied to semidefinite problems [169], operator equations [178], shape-preserving interpolation problems [33], [34], [35] and option price problems [179].

Note that (1.1.3) is only convergent locally under semismoothness assumption. A natural question is that whether (1.1.3) can be globalized similar to classic Newton's method for solving smooth equations or not. In general, the answer is negative because

the function  $\theta$  defined by

$$\theta(x) = \frac{1}{2} \left\| G(x) \right\|^2$$

is not smooth. Fortunately, in some especial but important cases,  $\theta$  can be smooth though G itself is not smooth. For example, if  $G(x) = \max(0, x), x \in \Re$  or  $G(x) = (G_1(x), \dots, G_n(x))^T$  with

$$G_i(x) = \phi_{FB}(x_i, F_i(x)), \quad i = 1, \cdots, n,$$

where  $F : \Re^n \to \Re^n$  is continuously differentiable. Obviously, G is not differentiable at x = 0, but  $\theta$  is smooth. By assuming that  $\theta$  is smooth, a damped semismooth Newton method was presented and a global convergence result was proved by De Luca et al., see [101] for details. In addition, see [83] for various globalized semismooth Newton methods.

#### 1.1.2 Smoothing Newton Method

As mentioned in Subsection 1.1.1, in the case when  $\theta$  is smooth, we can construct a global convergence algorithm for solving the system of nonsmooth equations (1.1.1). But, if  $\theta$  is not smooth, how to solve the system of nonsmooth equations? In this subsection, we review some existing smoothing methods for solving equations with nonsmooth  $\theta$ , these methods also overcome the difficulty of computation of  $\partial G(x^k)$  in semismooth Newton method.

The key idea of smoothing methods is to construct a smoothing approximation function  $\overline{G}: \Re_{++} \times \Re^n \to \Re^n$  of G such that for any  $\varepsilon > 0$  and  $x, \overline{G}(\varepsilon, \cdot)$  is continuously differentiable on  $\Re^n$  and satisfies

$$\left\|\bar{G}(\varepsilon, x) - G(x)\right\| \to 0 \quad \text{ as } \ \varepsilon \downarrow 0$$

and to find a solution of (1.1.1) by (inexactly) solving the following problems for a given positive sequence  $\{\varepsilon^k\}, k = 0, 1, 2, \cdots,$ 

$$\bar{G}(\varepsilon^k, x) = 0.$$

Let us denote the Jacobian of  $\overline{G}$  with respect to the second variable by  $\nabla_x^T \overline{G}(\varepsilon, x)$ .

Then a smoothing Newton method can be defined as follows:

$$x^{k+1} = x^{k} - t_{k} \left( \nabla_{x}^{T} \bar{G}(\varepsilon^{k}, x^{k}) \right)^{-1} G(x^{k}), \qquad (1.1.8)$$

where  $\varepsilon^k > 0$  and  $t_k > 0$  is the stepsize. Smoothing Newton method (1.1.8) for solving the nonsmooth equation (1.1.1) has been studied for decades in different areas (see [20] for references). The global and linear convergence of (1.1.8) has been discussed in [123, 125]. In [20], the authors defined a Jacobian consistency property and showed that the smoothing approximation functions in [18, 48] have this property. Let  $\partial_C G$  be defined as

 $\partial_C G(x) = \partial G_1(x) \times \partial G_2(x) \times \cdots \times \partial G_n(x),$ 

where  $G = (G_1, G_2, \dots, G_n)$  with  $G_i : \Re^n \to \Re$  for  $i = 1, 2, \dots, n$ .

**Definition 1.1.2** Let G be a Lipschitz continuous function in  $\Re^n$ . We call  $\overline{G}$  a smoothing approximation function of G if  $\overline{G}$  is continuously differentiable with respect to the second variable and there is a constant  $\mu > 0$  such that for any  $x \in \Re^n$  and  $\varepsilon \in \Re_{++}$ ,

$$\left\|\bar{G}(\varepsilon, x) - G(x)\right\| \le \mu\varepsilon.$$
(1.1.9)

Furthermore, if for any  $x \in \Re^n$ ,

$$\lim_{\varepsilon \downarrow 0} \operatorname{dist}(\nabla_x^T \bar{G}(\varepsilon, x), \partial_C G(x)) = 0,$$

then we say  $\overline{G}$  satisfies the Jacobian consistency property.

Under the assumption that  $\bar{G}$  satisfies the Jacobian consistency property, a smoothing Newton method was introduced by Chen, Qi and Sun [20], which was called Jacobian smoothing Newton method in [88].

For the Jacobian smoothing Newton method, under suitable conditions, Chen, Qi and Sun [20] proved that the sequence  $\{x^k\}$  generated by the algorithm is bounded and each accumulation point of  $\{x^k\}$  is a solution of (1.1.1). Furthermore, Chen, Qi and Sun [20] proved that if G is CD-regular at an accumulation point  $x^*$  of  $\{x^k\}$ and G is semismooth (strongly semismooth) at  $x^*$ , then  $\{x^k\}$  converges superlinearly (quadratically) to  $x^*$ . Note that the convergence analysis of the Jacobian smoothing Newton method strongly depends on the Jacobian consistency property. It was verified in [20] that many smoothing functions satisfy it. However, on the other hand, the smoothing functions based on normal maps [145], which only require the mapping to be defined on the feasible region instead of on  $\Re^n$ , do not satisfy this property. See [130, 166, 191] for the smoothing forms of normal maps. In addition, more smoothing functions which do not satisfy the Jacobian consistency property arise. In order to circumvent one or several of these difficulties, another class of smoothing Newton methods were introduced in [130], i.e., squared smoothing Newton method. For convenience, we suppose that for any  $\varepsilon < 0$  and  $x \in \Re^n$ ,  $\bar{G}(\varepsilon, x) = \bar{G}(-\varepsilon, x)$  and  $\bar{G}(0, x) = G(x)$ . Let

$$\Phi(\varepsilon, x) = \left(\begin{array}{c} \bar{G}(\varepsilon, x) \\ \varepsilon \end{array}\right),$$

where  $\overline{G}$  is continuously differentiable at any  $z := (\varepsilon, x) \in \Re_{++} \times \Re^n$  and satisfies (1.1.9). It is obvious that  $x^*$  is a solution of (1.1.1) if  $(\varepsilon^*, x^*)$  satisfies  $\Phi(\varepsilon^*, x^*) = 0$ .

The important characteristic of the squared smoothing Newton method is that the smoothing parameter  $\varepsilon$  is regarded as a variable just as the original variable x.

For the squared smoothing Newton method, under suitable assumption, Qi, Sun and Zhou [130] proved that each accumulation point  $z^*$  of sequence  $\{z^k\}$  generated by algorithm is a solution of  $\Phi(z) = 0$ . Furthermore, if  $\Phi$  is semismooth (strongly semismooth) at  $z^*$  and  $\Phi$  is CD-regular at  $z^*$ , then  $\{z^k\}$  converges superlinearly (quadratically) to  $z^*$ .

There are several modifications of smoothing Newton method. The corresponding convergence results were also proved. See [22,88] for details. We should point out that, in the last decade, smoothing techniques were used widely on solving nonsmooth optimization problems such as complementarity problems, variational inequality problems and mathematical programs with equilibrium constraints. These contents can be found in X. Chen, L. Qi and D. Sun [20], F. Facchinei, H. Jiang and L. Qi [38], X. Chen, Z. Nashed and L. Qi [19], D. Sun and L. Qi [167], D. Li, L. Qi, J. Tam and S. Y. Wu [98], Y. Yang, L. Qi [190], X. D. Chen, D. Sun and J. Sun [21], Z. H. Huang, J. Han and Z. Chen [73], L. Qi and D. Sun [129], H. D. Qi and L. Z. Liao [121], L. Qi and G. Zhou [137] and L. Qi, D. Sun and G. Zhou [130]. For more details, see also L. Qi and D. Sun [128].

### **1.2 Semi-Infinite Programming Problems**

Consider the optimization problem:

$$\min_{x} f(x)$$

$$s.t. \ x \in X,$$

$$(1.2.1)$$

where the feasible set X is given in the form

$$X = \{ x \in \Re^n : g(x, v) \le 0, \forall v \in V \}.$$

Here  $V \subset \Re^m$  is a set of parameters and, for most engineering problems, V is nonempty compact. The functions  $f : \Re^n \to \Re$  and  $g : \Re^n \times V \to \Re$  are twice continuously differentiable functions. In the case when the set V is infinite, (1.2.1) is called a semiinfinite programming (SIP) problem. We say that the SIP problem (1.2.1) is convex if  $f(\cdot)$  is convex and the function  $g(\cdot, v)$  is convex for any  $v \in V$ .

One of the reasons why researchers are focusing more attention on the SIP problem is that the SIP problem arises from various applications such as approximation theory [50, 58, 92], optimal control [71, 103, 115, 150, 171], filter design in signal processing [91], eigenvalue computation, mechanical stress of materials [70], pollution control [58, 70], and statistical design [157].

Since that the solutions of many practical application problems can be approximated by the optimal solutions of the related SIP problems, the theory and numerical methods for SIP problems are very important. The development of theory and numerical methods for SIP problems can be found in [44,51,52,70,115,117,142,143,164,174,176].

On the theoretical aspect, Krabs [92] obtained KKT optimality conditions for SIP problems under Slater's constraint qualification. Hettich and Zencke [67] and Nuernberger [107] discussed first-order sufficient conditions for SIP problems under a stronger KKT condition. Hassouni and Oettli [64] derived a sufficient and necessary condition for convex SIP problems under a regularity condition. Second-order necessary and sufficient optimality conditions for SIP problems were first derived via the so-called reduction method by Wetterling [183] and Hettich and Jongen [65]. Shapiro [154, 155] discussed the differentiability of the value function and the Lipschitzian stability of the solution set mapping for a parametric SIP problem. Colgen [25] discussed the compactness of the solution set under a kind of upper-continuity of the solution set mapping.

Bonnans and Shapiro [11] derived a zero duality gap of the convex SIP problem and its Lagrangian dual problem under Slater's constraint qualification condition.

On the numerical aspect, the existing different methods for solving the SIP problem can be divided into the following six categories:

- 1. discretization methods (by grids and cutting planes);
- 2. local reduction based methods;
- 3. exchange methods;
- 4. simplex-like methods;
- 5. descent methods;
- 6. generalized Newton methods.

In the first category, a sequence of relaxed problems with a finite number of constraints are solved according to a predefined or adaptively controlled grid generation scheme or some cutting plane scheme [60, 61, 67, 69, 140, 141, 174, 175, 185]. The local reduction method of second category replaces an SIP problem by a locally equivalent problem with a finite number of implicitly defined inequality constraints, or equivalently a system of nonlinear equations with finitely many unknowns, which is solved essentially by the Newton method and, hence the local reduction based methods have good local convergence properties [26, 53, 66, 78, 118, 170]. In the third category, typical exchange methods consist of two phases: the purification phase providing an extreme point and the pivoting phase generating a sequence of linked extreme point leading to an optimal solution [70, 95, 146, 151, 180, 184, 186]. Some of other methods for SIP problems can be found in [41, 42, 98, 99, 115, 117, 134, 158, 176, 177].

We now briefly review the optimality conditions and numerical methods for SIP problems.

#### **1.2.1** Optimality Conditions

Define

$$\Lambda(x) := \max_{v \in V} g(x, v),$$

then the SIP problem (1.2.1) becomes

$$\min_{x} f(x)$$
s.t.  $\Lambda(x) < 0.$ 
(1.2.2)

The fundamental difficulty, however, with the problem (1.2.2) is that the constraint function  $\Lambda(x)$  may not be differentiable. Thus a theory for nondifferentiable nonlinear programming is needed. Such a first order theory does exist (e.g. [49]) and was used to establish first order condition for (1.2.1), see e.g. [120]. In this subsection, we only recall briefly the first order optimality conditions for the SIP problem (1.2.1).

Let Y = C(V) be the normed space of continuous functions  $y : V \to \Re$ , equipped with the sup-norm

$$\|y\| = \sup_{v \in V} |y(v)|$$

and  $K \subset C(V)$  be the cone formed by nonpositive-valued continuous functions y(v). Let  $Y^*$  be the dual space of the Banach space Y, formed by continuous linear functionals on Y and equipped with the dual norm  $||y^*|| := \sup_{y \in U_Y} \langle y^*, y \rangle$ , where  $U_Y$  is the unit ball in Y, i.e.,  $U_Y := \{y \in Y : ||y|| \le 1\}$ , and  $\langle y^*, y \rangle = y^*(y), y^* \in Y^*, y \in Y$ . And let  $K^-$  be the polar (negative dual) cone of the cone K, i.e.,

$$K^{-} = \{y^* \in Y^* : \langle y^*, y \rangle \le 0 \text{ for all } y \in K\}.$$

Then  $Y^*$  is the space of finite signed measures on  $(V, \mathcal{B})$ , where  $\mathcal{B}$  is the Borel  $\sigma$ -algebra of V, with the norm given by the total variation of the corresponding measure. And it is not difficult to know that  $K^-$  is formed by the set of nonnegative Borel measures on V. See [154] for details.

Consider the mapping  $\Xi : \Re^n \to C(V)$  taking a point x into the function  $y = \Xi(x)$ ,  $y(\cdot) = g(x, \cdot)$ . Then the feasible set X of (1.2.1) can be written in the form

$$X = \left\{ x \in \mathbb{R}^n : \ \Xi(x) \in K \right\}.$$

Since the constraint function g(x, v) is differentiable in x and  $\nabla_x g(x, v)$  is continuous on  $\Re^n \times V$ , it follows that the mapping  $\Xi(x)$  is continuously differentiable and for  $h \in \Re^n$ 

$$[D\Xi(x)h](\cdot) = h^T \nabla_x g(x, \cdot),$$

where  $D\Xi(x)h := h \cdot \nabla \Xi(x)$ . We say that a point  $x_0 \in \Re^n$  is a regular point of the mapping  $\Xi(x)$ , in the sense of [144], if

$$0 \in \inf \left\{ \Xi(x_0) + D\Xi(x_0) \Re^n - K \right\}.$$
(1.2.3)

We assume the optimal value of (1.2.1) is finite, say  $\mu$ . We also assume that the set

$$S = \{x \in X : f(x) = \mu\},\$$

of optimal solutions of (1.2.1) is nonempty. Let

$$V(x) = \{ v \in V : g(x, v) = 0 \}.$$

A first-order necessary condition for the SIP problem (1.2.1) is well known (e.g., Pshenichnyi [120]), and it is called the Fritz-John necessary optimality condition.

**Theorem 1.2.1** Let  $x_0$  be an optimal solution of (1.2.1). Then there exist  $v^i \in V(x_0)$ ,  $i = 1, \dots, n$  and nonnegative multipliers  $u_0, u_1, \dots, u_n$ , not all of them zero, such that

$$u_0 \nabla f(x_0) + \sum_{i=1}^n u_i \nabla_x g(x_0, v^i) = 0.$$
 (1.2.4)

We should point out that the multiplier  $u_0$  in Theorem 1.2.1 may be zero, in this case, (1.2.4) is not very meaningful since it does not show any information about the objective function f. In order to obtain the Fritz-John necessary optimality condition with  $u_0 \neq 0$ , we need the following constraint qualifications which are straightforward extensions of their finite analogues for the SIP problem (1.2.1).

**Definition 1.2.1** (i) We say that the extended Mangasarian-Fromovitz constraint qualification (EMFCQ) holds at a point  $x_0 \in S$  if there exists a vector  $d \in \Re^n$  such that

$$\nabla_x^T g(x_0, v) d < 0 \tag{1.2.5}$$

for all  $v \in V(x_0)$ .

(ii) It is said that the Slater condition holds if there exists a point  $\bar{x}$  such that

$$g(\bar{x}, v) < 0 \tag{1.2.6}$$

for all  $v \in V$ .

It is not difficult to show that (1.2.5) is equivalent to the condition that the point  $x_0$  is a regular point of the mapping  $\Xi(x)$  with respect to the cone K [154]. Since the set V is compact, condition (1.2.6) implies existence of a positive number  $\varepsilon$  such that  $g(\bar{x}, v) < -\varepsilon$  for all  $v \in V$ . That is,  $\Xi(\bar{x})$  is an interior point of the cone K, which implies

$$0 \in \operatorname{int} \left\{ \Xi(\mathfrak{R}^n) - K \right\}, \tag{1.2.7}$$

where  $\Xi(\Re^n)$  is the range of the mapping  $\Xi$ , i.e.,  $\Xi(\Re^n)$  is the set  $\{\Xi(x): x \in \Re^n\}$ . Since  $\Xi$  is continuously differentiable, (1.2.7) is equivalent to (1.2.3).

**Theorem 1.2.2** Let  $x_0$  be an optimal solution of (1.2.1). If EMFCQ holds at point  $x_0$ , then there exist  $v^i \in V(x_0)$ ,  $i = 1, \dots, n$  and nonnegative multipliers  $u_1, \dots u_n$ , not all of them zero, such that

$$\nabla f(x_0) + \sum_{i=1}^n u_i \nabla_x g(x_0, v^i) = 0.$$

The Fritz John condition with  $u_0 \neq 0$  is a semi-infinite version of the KKT condition, e.g. [102], so we call it KKT condition for the SIP problem (1.2.1). Generally, the Fritz John condition is necessary but not sufficient for optimality, and if the SIP problem (1.2.1) is convex, then the KKT condition is sufficient but not necessary for optimality. Moreover, if a constraint qualification is assumed, then the KKT condition is both necessary and sufficient for optimality for (1.2.1). Under certain assumptions, there are some second order optimality conditions for the SIP problem (1.2.1), see e.g. [7], [152], [156] for details.

#### **1.2.2** Numerical Methods for SIP Problems

It is well known that the main difficulty for solving SIP problems is that it has infinite constraints, and the main effort of existing methods is to reduce the infinite set V to a finite one. The survey papers by Hettich [68], Gustafson and Kortanek [62], Polak [115], Fiacco and Ishizuka [43], and Hettich and Kortanek [70] discussed these methods in more detail. In this subsection, we will mainly review the discretization methods and the local reduction based methods.

#### **Discretization Methods**

One approach to the solution of an SIP problem is to minimize its objective function subject to only a finite subset of the infinite set of constraints. That is, the original SIP problem (1.2.1) is replaced by the following finite subproblem  $P[\bar{V}]$ :

$$\min_{x} f(x)$$

$$g(x,v) \le 0, \ v \in \overline{V},$$

$$(1.2.8)$$

where  $\bar{V} \subset V$  with cardinality  $|\bar{V}| < \infty$ . The set  $\bar{V}$  is typically called a grid. Unfortunately, for general problems, there does not exist a subset  $\bar{V}$  of V which yields identical solutions for (1.2.1) and (1.2.8). Therefore, one may possibly repeat the procedure for an enlarged set when higher precision is requested or when, from consideration of a sequence of such solutions, an estimate of their accuracy is to be obtained. But, if a finite sequence of finer and finer grids  $\bar{V}_i$  are used, it is not necessarily true that the sequence of solution points of (1.2.8) converge to the solution of (1.2.1). For these statements to be true, the grids must be chosen with care [70], [117]. More precisely, one may successively compute an "(approximate) solution" of the discretized SIP problem  $P[\bar{V}_i]$ for  $i = 0, 1, \cdots$  by an algorithm for solving finite optimization problem, where  $\{\bar{V}_i\}$  is a sequence of finite subsets of V such that  $\lim_{i\to\infty} \operatorname{dist}(\bar{V}_i, V) = 0$ . Here

$$\operatorname{dist}(\bar{V}_i, V) := \sup_{v \in V} \inf_{v' \in \bar{V}_i} \|v - v'\|_{\infty},$$

which is usually called a density of grid  $\bar{V}_i$  in V and  $\|\cdot\|_{\infty}$  is the usual  $l^{\infty}$ -norm in  $\Re^m$ . The conception of density of set  $\bar{V}_i$  in V is closely related to the theory of consistent approximations. The papers by Reemtsen [140] and Polak [117] give some conditions under which a solution of problem (1.2.8) is equivalent to (1.2.1) as successive grids are refined. A procedure of this type is denoted as a discretization method.

Discretization methods have the advantage to internally work with finite subsets of V only. In particular, feasibility with respect to the finite program  $P[\bar{V}_i]$  can normally be checked easily and accurately. Therefore, a discretization method is especially suited for problems with a solution  $x^*$  at which  $g(x^*, \cdot)$  is (almost) constant on V or on parts of V. Almost constancy is a phenomenon which, for instance, can occur for the constraint function at complex Chebyshev approximation problem [142].

Discretization methods, however, are very expansive in numerical computation since the number of the constraints of the subproblem solved in such methods may increase dramatically with the growing cardinality of the approximate subset at each iteration. The numerical costs for solving discretized SIP problems normally tend to infinity when the grid densities in V converge to zero. Therefore, in practice, only grids with a limited number of points can be used, and the grids obtained by typical methods have at most 50,000 to 100,000 points for problems with less than 100 variables [142]. Another characteristic of a discretization method stressed in the literature is that it normally produces outer approximations of a solution of the SIP problem, i.e., approximate solutions which are not feasible for (1.2.1). Observe that, for  $\bar{V}_i \subseteq V$ , a global solution  $x^{i*}$  of  $P[\bar{V}_i]$  that is feasible for P[V] solves P[V] since

$$f(x^{i*}) = \inf_{x \in F(\bar{V}_i)} f(x) \le \inf_{x \in F(V)} f(x) \le f(x^{i*}),$$

where

$$F(V) := \{ x \in \Re^n : g(x, v) \le 0, v \in V \}$$

and

$$F(\bar{V}_i) := \left\{ x \in \Re^n : g(x, v) \le 0, \ v \in \bar{V}_i \right\}$$

An approximate solution of an SIP problem which has been obtained by a discretization method may have to be improved by a method based on local reduction when the method becomes too inefficient. A possible difficulty connected with that is that the obtained solution may not be close enough to a solution of the SIP problem and hence not be in the convergence region of such method.

For the discretization methods for solving SIP problems, Polak [116] has developed a theory of consistent approximations that provides conditions under which (local) minimizers and certain stationary points of the discretized problems converge to (local) minimizers and related stationary points of the SIP problem. The theory includes conditions which imply convergence of the entire sequence of iterates generated by a discretization algorithm, and it contains condition on the rate of discretization which ensures that the entire sequence converges with the same rate as the algorithm used for the solution of the finite subproblem. The discretization methods have no superlinear convergence property.

#### Local Reduction Based Methods

One may use some different ways to form the approximate subset of V so that the number of the constraints of the finite nonlinear optimization problem increases not so quickly. Among these ways, one class of methods relies on the fact that the SIP problem (1.2.1) can be reduced locally to an optimization problem with finite number of constraints. The term local reduction comes from that

$$g(x,v) \le 0, \ \forall \ v \in V$$

is closely related to the following parametric optimization problem:

$$\max_{v \in V} g(x, v). \tag{1.2.9}$$

As mentioned in Subsection 1.2.1, for the SIP problem (1.2.1), if we let

$$\Lambda(x) := \max_{v \in V} g(x, v), \qquad (1.2.10)$$

then it becomes

$$\min_{x} f(x)$$
  
s.t.  $\Lambda(x) \le 0.$ 

The review by Polak [115] treats this general case. However, more progress can be made by treating  $\Lambda(x)$  locally. The goal is to represent  $\Lambda(x)$  locally near almost every  $x_* \in \Re^n$  by

$$\Lambda(x) = \max\left\{g^l(x): \ l \in L\right\}$$

with smooth function  $g^{l}(x)$ ,  $|L| < \infty$ , defined on some neighborhood of  $x_{*}$  [55]. This holds under some mild regularity assumptions.

Let  $\bar{x} \in \Re^n$ . Denote all the local solutions of (1.2.10) by  $v^l(\bar{x})$ ,  $l \in \bar{L}$ . If problem (1.2.10) is regular for  $\bar{x}$ , then  $|\bar{L}| < \infty$ , and there is a neighborhood  $U(\bar{x})$  of  $\bar{x}$  such that  $x \in U(\bar{x})$  is the feasible solution of the SIP problem if and only if  $x \in U(\bar{x})$  satisfies the finitely many inequality constraints

$$g^{l}(x) := g(x, v^{l}(x)) \le 0, \ l = 1, \cdots, \bar{L},$$
 (1.2.11)

where  $v^l(x)$  are the local solutions of (1.2.10) with parameter x. While the functions  $g^l$  themselves usually cannot be given explicitly, the function values  $g^l(x)$  obviously are computable for  $x \in U(\bar{x})$ . Methods, in which the infinite constraints  $g(x, v) \leq 0, v \in V$ ,

of the SIP problem (1.2.1) is locally replaced at  $\bar{x}$  by the finitely many constraints in (1.2.11), are called local reduction based methods. These assertions may lead directly to a conceptual reduction method [142]. The papers by Gramlich et al. [55] and Hettich and Kortanek [70] explain each step of algorithm and give guidelines on implementation. The most commonly used nonlinear programming algorithm at the main step in the conceptual reduction method is successive quadratic programming (SQP) algorithm.

The advantage of such methods certainly lies in the fact that they only deal with relatively small finite programs internally. These programs are convex for linear and convex SIP problems, since, in these cases, the functions  $g^l$  are strictly convex on U(x) [67], and they are usually nonlinear for all other SIP problems. The drawbacks of reduction based methods are connected with the fact that the set  $U(\bar{x})$  and the functions  $g^l$  are not known explicitly. In almost every iteration, a (continuous) global maximizer and hence all (continuous) local maximizers of  $g(x^i, \cdot)$  over V need to be computed exactly or inexactly. Up to now, however, there does not exist the algorithm that is able to detect a global maximizer of an arbitrary continuous function with certainty. Furthermore, local reduction based methods needs strong conditions to ensure its convergence.

In this thesis, we shall introduce four algorithms for solving SIP problems. To this end, we first study the differentiability, semismoothness and smoothing approximation properties of a class of integral functions in Chapter 2. Then, in Chapter 3 we introduce three algorithms for solving SIP problems, say, smoothing SQP algorithm, smoothing projected Newton-type algorithm and smoothing Newton-type algorithm. The later two algorithms have many advantages. For example, at each iteration of the algorithm, only a system of linear equations needs to be solved and they have global and local superlinear convergence. Furthermore, in Chapter 4, we extend the smoothing projected Newton-type algorithm to solving large scale SIP problems, say, truncated projected Newton-type algorithm. This algorithm enjoys the advantages mentioned above.

### 1.3 Generalized Semi-Infinite Programming Problems

Consider a generalized semi-infinite programming (GSIP) problem given as follows:

$$\min_{x} \quad f(x) 
s. t. \quad g(x, u) \le 0, \ \forall \ u \in T(x),$$
(1.3.1)

where  $T(x) = \{ u \in \Re^r \mid h(x, u) \leq 0 \}$ . Here,  $f : \Re^n \to \Re, g : \Re^n \times \Re^r \to \Re,$  $h : \Re^n \times \Re^r \to \Re^J, T : \Re^n \to 2^{\Re^r}$ , and  $2^{\Re^r}$  is the set of all subsets in  $\Re^r$ .

Recently, it was observed that many practical problems, such as those arising in the study of maneuverability problems in robotics, optimal control problems with terminal constraints, reverse Chebyshev problems in approximation theory, time-minimal heating or cooling of a ball and vector variational inequality problems, can be transformed into the GSIP problem. Consequently, the GSIP problem has become an active research topic in optimization. Jongen et al [86]and Stein [159] derived directly a first order necessary condition of the Fritz John type without assuming a constraint qualification. Rückmann and Shapiro [148] also obtained a first order necessary condition by using the boundedness of the upper and lower directional derivatives of the corresponding value function. Hettich and Still [72] investigated second order optimality conditions under the assumption that the feasible set can be described by means of a finite number of value functions. Still [162, 163] investigated how the numerical methods for SIP problems can be extended to GSIP problems. Kaplan and Tichatschke [89] derived numerical algorithm for a special class of GSIP problems.

On the other hand, since there are many uncertainty factors in many practical problems, stochastic programming is another important branch of mathematical programming. Of course, it is very significative to study the stochastic generalized semi-infinite programming (SGSIP) problem. The GSIP problem is a hard problem with an infinite constraint index set that may vary since it is correlated with decision variable x. Presence of an additional random variable makes the SGSIP problem even harder to solve than the GSIP problem.

We shall present a smoothing implicit programming method for solving a class of SGSIP problems and prove its global convergence result in Chapter 5.

### **1.4 Option Price Interpolation Problems**

In 1900, Bachelier [3] introduced a formula for the price of an option based upon a continuous stochastic process for the underlying asset. Although this model has some shortcomings, this work pushed option pricing into the realm of higher mathematics. In 1973, Black and Scholes [10] formulated the first modern option pricing solution, which overcomes some of the major shortcomings of earlier works. The explosion of option pricing papers has been phenomenal since 1973. Especially, after the equity market crash in 1987, some complicated models are proposed, such as local volatility function model of Dumas, Flemming and Whaley [37], stochastic volatility model of Hull and White [75], and jump-diffusion model of Bates [4], etc. In addition, the work on empirical tests and methods to solve option problems grew rapidly, and continues today at a great pace. We should point out that no matter what kind of process for the underlying asset, the Black-Scholes formula is always a basic tool.

In the last several years, the option price model reverse engineering problem has attracted intensive attention, since it is widely used in many important fields such as risk management and exotic option pricing. Basically, given very few assumptions about the underlying process, people try to back out the underlying process from the option price observation. In the option price model reverse engineering, the Black-Scholes formula is often used to calculate the option's risk sensitivities. In addition, in the option price model reverse engineering, European is often used as a standard option model, that is, if a model cannot correctly price European options then it cannot be used to price the exotic option. Many models have been proposed for the option price reverse engineering problems. For example, in the equity option modeling area, we have the continuous version local volatility function model of Andersen and Brotherton-Ratcliffe [2], Lattic version local volatility function model of Rubinstein [147], Derman, Kani and Chriss [31] and Markov functional interest rate term structure model of Hunt, Kennedy and Pelsser [76], etc.

Many option price reverse engineering models require a complete set of European option price observations. However, in a typical option market, one often observes the prices of 10 to 30 options with the same time-to-expiration, but different strike levels. This reality requires a good method to interpolate the option price as a continuous function of strike price. Therefore, we face the problem of how to interpolate the option price function. The market standard practice is to interpolate the implied volatilities, then to apply the interpolated volatility into the Black-Scholes formula. Such an indirect interpolation has some merits, such as accuracy and stability, see Andreasen [1]. Piecewise linear interpolation is a common method to interpolate market implied volatilities when pricing unquoted options. Cubic spline volatilities interpolation is the method used for almost all option price reverse engineering models.

Unfortunately, in some case, the interpolation method cannot be chosen freely. For example, the no-arbitrage principle determines that a call option price must be a monotonically decreasing and convex function of the strike price. Especially, this restriction violation in the option price reverse engineering problems will make the model break down. So, shape-preserving interpolation methods are very important in the option price reverse engineering problems. Rubinstein [147], Jackwerth and Rubinstein [79] presented two methods deal with this kind of problem, but their computation are intensive.

Wang, Yin and Qi [179] presented a generalized Newton algorithm for solving noarbitrage option price interpolation problem. And they proved the local superlinear convergence of this method, however they did not establish its quadratic convergence.

In Chapter 6 of this thesis, we shall prove that this algorithm has at least  $\frac{4}{3}$ -order convergence rate, by using the properties of integral function discussed in Chapter 2. We also give conditions under which this method has  $\frac{3}{2}$ -order and quadratic convergence rate. We will present a globalized algorithm for solving no-arbitrage option price interpolation problem and prove that its convergence rate is at least  $\frac{4}{3}$ .

### **1.5** Spectral Estimation Problems

Spectral estimation problems arise naturally in time series and signal processing, and have long received deep and fruitful attention from statisticians and engineers. Given a compact set X with finite measure  $\sigma$ , a set of continuous functions  $\{\phi_i\}_{i=1}^n$  linearly independent on X, and a data vector  $\vec{r} = (r_1, r_2, \dots, r_n)^T \in \Re^n$ , the problem is to find a function f satisfying the nonnegativity condition  $f \ge 0$  a.e., and the constraints

$$r_i = \int_X f(x)\phi_i(x)d\sigma(x), \quad i = 1, 2, \cdots, n.$$
 (1.5.1)

The nonnegativity condition arises naturally in many signal processing applications such as estimating a power spectrum or in interference spectroscopy where the problem is to estimate an intensity distribution of an electromagnetic source. It is assumed that f is band limited, i.e., f has support contained in a compact set X. The band limited constraint may be due to properties of the source, medium, sensors, or problem geometry [24]. The investigation of such problem raises two important questions. The first and fundamental question concerns the existence of such function f. This question was answered by characterizing the set of extendible correlation measurements, see [97, 100] for details. The second question raised is that if there is a unique f, and if not, how can a specific one be chosen? In fact, a unique f does not exist except in very special cases; the task of a spectral estimation method is the selection of one out of an ensemble of f satisfying the constraint (1.5.1), nonnegativity, and support constraint.

One popular method for selecting f is the maximum entropy method [14, 15, 100]. In this method, the usual form the solution takes is  $f^*(x) = \frac{1}{P(x)}$ , where P is a positive trigonometric polynomial. However, in some application, see examples in [6], a strictly positive solution  $\frac{1}{P(x)}$  fails to exist. Goodrich and Steinhardt [54] suggested an alternative way for selecting f(x) by considering an  $L_2$ -norm optimization problem, which is called  $L_2$  spectral estimation. In 1993, Cole and Goodrich [24] investigated the  $L_p$ special estimation with an  $L_{\infty}$ -upper bound, they compared the numerical performance of some solution methods and found Newton's method does the best job of fitting the solution to the data. Potter [119] also obtained similar numerical results.

An important problem in spectral estimation is the estimation of a power spectrum, a measure  $\mu$  on  $\Re^n$ , with a known support, given a finite collection of measured correlations. This problem has many applications in a wide range of settings such as geophysics, radio astronomy, radar, and sonar, see [15,90,96,97] and references therein. In many of these problems, the measure  $\mu$  has a density function f(x), i.e.,

$$\mu(\Omega) = \int_{\Omega} f(x) dx$$

for all Lebesgue measurable sets  $\Omega$  and  $\mu$  is band limited. Thus the estimation problem of power spectrum becomes the form of (1.5.1), see [6,54] for details.

In Chapter 7, we shall present a generalized Newton algorithm for solving the  $L_2$  spectral estimation problem and we prove the global and local  $1 + \frac{1}{2m}$ -order convergence of the algorithm.

### 1.6 Notation

The following notation will be used in this thesis.  $\Re^n$  denotes the real Euclidean space of column vectors of length n.  $\Re^n_+$  denotes the set of all vectors with nonnegative components, i.e.,  $\Re^n_+ = \{ u \in \Re^n | u \ge 0 \}$ , and  $\Re^n_{++}$  denotes the set of all vectors with positive components, i.e.,  $\Re_{++}^n = \{u \in \Re^n | u > 0\}$ . For  $u \in \Re^n$  and  $v \in \Re^m$ , (u, v)denotes the column vector  $(u^T, v^T)^T$  in  $\Re^{n+m}$ . If  $v_i \in \Re^n$  and  $L = \{1, 2, \dots, l\}, (v_i, i \in \Re^n)$ L) means the vector  $(v_1^T, v_2^T, \dots, v_l^T)^T$  in  $\Re^{nl}$ . For any  $x \in \Re$ ,  $[x]_+ := \max\{0, x\}$ . By  $\|\cdot\|$ , we mean the Euclidean norm. If  $x_1$  and  $x_2$  are two vectors with the same dimension, then  $x_1^T x_2$  denotes the inner products of these two vectors. I and O denote the identity and zero matrix with a suitable dimension, respectively, and U denotes the closed unit ball in an Euclidean space with a suitable dimension. For a differentiable function  $f: \Re^n \to \Re^p, \nabla f(x) \in \Re^{n \times p}$  denotes the gradient of f at x, whereas  $\nabla^T f(x) \in \Re^{p \times n}$ means the Jacobian of f at x, i.e.,  $\nabla^T f(x) = (\nabla f(x))^T$ . For a differentiable function  $f: \Re^n \times \Re^m \to \Re^p, \nabla_x f(x,y)$  and  $\nabla_y f(x,y)$  denote the gradients of f at (x,y) with respect to x and y, the  $n \times p$  and  $m \times p$  matrices, respectively. Whereas  $\nabla_x^T f(x, y)$  and  $\nabla_y^T f(x,y)$  mean the Jacobians of f at (x,y) with respect to x and y, respectively, i.e.,  $\nabla_x^T f(x,y) = (\nabla_x f(x,y))^T$  and  $\nabla_y^T f(x,y) = (\nabla_x f(x,y))^T$ . For a differentiable function  $f: \Re^n \times \Re \to \Re, \nabla^{(k)}_y f(x, y)$  means the k-th derivative of f at (x, y) with respect to y. For a locally Lipschitz function  $f: \Re^n \to \Re^p, \partial f(x)$  denotes the generalized Jacobian of f at x in the sense of Clarke. For a locally Lipschitz function  $f: \Re^n \times \Re^m \to \Re^p$ ,  $\partial_x f(x,y)$  and  $\partial_y f(x,y)$  denote the generalized Jacobian of f at (x,y) with respect to x and y, respectively. For a directionally differentiable function  $f: \Re^n \to \Re^p, f'(x; h)$ denotes the directional derivative of f at x in a direction h. For a closed set A and a point  $x \in \Re^n$ , dist(x, A) means the distance between x and A, i.e.,

$$dist(x, A) = min \{ ||x - x'|| : x' \in A \}.$$

If  $\delta$  is a small quantity,  $O(\delta)$  and  $o(\delta)$  mean the same order and higher order small quantity, respectively.
# Chapter 2

# A Class of Integral Functions

## 2.1 Introduction and Motivation

The integral function  $F: \Re^n \to \Re$ , defined by

$$F(x) := \int_{a}^{b} [g(x,v)]_{+} p(v) dv, \qquad (2.1.1)$$

where  $p(v) \ge 0$  for all  $v \in [a, b]$ , arises from nonsmooth equation reformulations of the shape-preserving interpolation problem [33], [34], the option price problem [179] and the spectral estimation problem which will be discussed in Chapter 7. It also arises from aggregate reformulation of SIP problems, see [173], [81], [172] and [124]. As mentioned in Subsection 1.1.1, semismoothness, p-order semismoothness and strong semismoothness are the key conditions for superlinear, (1+p)-order and quadratic convergence of the generalized Newton method for solving a system of nonsmooth equations, respectively. Therefore, convergence analyses of numerical methods designed for solving such problems above via their reformulations are highly related to differentiability properties of this integral function.

Differentiability properties and applications of the integral function (2.1.1) were discussed in recent literatures by Dontchev, Qi and Qi [33], [34], Qi [124], Qi and Tseng [132], Qi and Yin [136], and Wang, Yin and Qi [179]. In the applications of shape-preserving interpolation problems and option price problems, the integral function F, defined by (2.1.1), plays a central role. Its semismoothness was established in [33], and hence the superlinear convergence of a Newton-like method for solving the system of nonsmooth equations arising in the shape-preserving interpolation problem was proved, which was a conjecture for 15 years [77]. In [34], strong semismoothness of a particular form of the integral function for that shape-preserving interpolation problem was established. This, further, established quadratic convergence of the Newton-like method. In [136], this result was generalized to a class of integral functions, which are still a special case of (2.1.1).

The main aim of this chapter is to investigate the differentiability, semismoothness and p-order semismoothness of a general class of integral functions which includes functions of the form (2.1.1) as a particular case. Also, in this chapter, we study some smoothing approximation properties of a class of integral functions.

The outline of this chapter is as follows: in Section 2.2, some differentiability properties of a general class of integral functions are discussed; in Section 2.3, the semismoothness and p-order semismoothness of a general class of integral function are investigated; and finally, in Section 2.4, some smoothing approximation properties of integral function (2.1.1) are obtained.

## 2.2 Differentiability of Integral Functions

Consider an integral function  $F: \Re^n \to \Re$ , defined by

$$F(x) := \int_{V} f(x, v) d\mu(v), \qquad (2.2.1)$$

where  $f: X \times V \to \Re$ , X is an open subset of  $\Re^n$  and  $\mu$  is a finite measure defined on a measurable space  $(V, \mathcal{F})$ . The function defined by (2.2.1) includes functions of the form (2.1.1) as a particular case.

We assume that for every  $x \in X$ , the function  $f(x, \cdot)$  is  $\mathcal{F}$ -measurable and  $\mu$ integrable, i.e.,

$$\int_{V} |f(x,v)| d\mu(v) < +\infty.$$

This implies that the integral function F(x) is well-defined and finite valued. Denote  $f_v(\cdot) := f(\cdot, v)$  and let  $x \in X$  be fixed. We say that a property holds for almost every (a.e.)  $v \in V$  if it holds for all  $v \in V$  except on a set with  $\mu$ -measure zero.

The following result is a consequence of the Lebesgue Dominated Convergence Theorem (e.g., [11]).

**Proposition 2.2.1** Suppose that: (i) there exists an integrable function  $\kappa : V \to \Re_+$  such that

$$|f(x^1, v) - f(x^2, v)| \le \kappa(v) ||x^1 - x^2||$$
, for all  $x^1, x^2 \in X$ ; and a.e.  $v \in V$ , (2.2.2)

(ii) for a.e.  $v \in V$ ,  $f_v(\cdot)$  is directionally differentiable at a point  $x \in X$ . Then  $F(\cdot)$  is Lipschitz continuous on X, directionally differentiable at x and

$$F'(x;h) = \int_{V} f'_{v}(x;h) d\mu(v).$$
(2.2.3)

Condition (2.2.2) implies, of course, that for a.e.  $v \in V$  the function  $f(\cdot, v)$  is Lipschitz continuous on X. Note that the results of the above proposition have a local nature and the set X can be reduced to a neighborhood of a considered point x. Note also that for locally Lipschitz continuous functions the concepts of Fréchet and Gâteaux directional differentiability do coincide (e.g., [153]). Hence, under the conditions of Proposition 2.2.1, we may simply discuss directional differentiability (or differentiability) of  $F(\cdot)$  at x.

It immediately follows from (2.2.3) that F'(x;h) is linear in h, i.e.,  $F(\cdot)$  is differentiable at x, if  $f_v(\cdot)$  is differentiable at x for a.e.  $v \in V$ . Moreover, we have the following result (e.g., [149]).

**Proposition 2.2.2** Suppose that, in addition to the assumptions (i) and (ii) of Proposition 2.2.1,  $f'_v(x; \cdot)$  is convex for a.e.  $v \in V$ . Then  $F(\cdot)$  is differentiable at x if and only if  $f_v(\cdot)$  is differentiable at x for a.e.  $v \in V$ . In this case, it holds that

$$\nabla F(x) = \int_{V} \nabla f_{v}(x) d\mu(v). \qquad (2.2.4)$$

Suppose now that the function f(x, v) is given as the maximum of a family of smooth functions  $g_j : X \times V \to \Re, j \in J$ . That is,

$$f(x,v) := \sup_{j \in J} g_j(x,v).$$
(2.2.5)

We make the following assumptions.

(A1) V is a compact metric space and  $\mathcal{F}$  is its Borel sigma algebra.

(A2) For every  $v \in V$  and  $j \in J$ , the function  $g_{jv}(\cdot) := g_j(\cdot, v)$  is continuously differentiable on X.

(A3) The function  $(x, v, j) \mapsto G_{jv}(x) := \nabla g_{jv}(x)$  is continuous on  $X \times V \times J$ .

(A4) The set J is a compact metric space.

Of course, if the set J is finite, then the last assumption (A4) holds automatically.

By the Danskin theorem (e.g., [11]), it follows from assumptions (A2)–(A4) that the max-function  $f_v(\cdot)$ , defined in (2.2.5), is directionally differentiable at every point  $x \in X$  and

$$f'_{v}(x;h) = \sup_{j \in J^*_{v}(x)} h^T G_{jv}(x).$$
(2.2.6)

Here  $J_v^*(x)$  denotes the index set of active at  $x \in X$  constraints,

$$J_v^*(x) := \arg \max_{i \in J} g_i(x, v).$$
(2.2.7)

Note that since it is assumed that the set J is compact and the function  $g_j(x, v)$  is continuous in  $j \in J$ , the set  $J_v^*(x)$  is nonempty and compact.

Let  $U \subset X$  be a compact neighborhood of a point  $\bar{x} \in X$ . By the Mean Value Theorem and assumptions (A2) and (A3), we know that for all  $x^1, x^2 \in U$  and  $\kappa_j(v) := \sup_{x \in U} ||G_{jv}(x)||$ , the following holds

$$|g_j(x^1, v) - g_j(x^2, v)| \le \kappa_j(v) ||x^1 - x^2||.$$

It follows that  $f_v(\cdot)$  is Lipschitz continuous on U with the Lipschitz constant  $\kappa(v) := \sup_{j \in J} \kappa_j(v)$ . By the assumption (A3) and the fact that the sets V and J are compact, the function  $\kappa(v)$  is bounded on V, and hence is integrable.

By formula (2.2.6),  $f'_v(x; \cdot)$  is given by the maximum of linear functions and hence is convex. It also follows from (2.2.6) that  $f_v(\cdot)$  is differentiable at x iff  $G_{jv}(x)$  is the same for all  $j \in J^*_v(x)$ , say  $G_{jv}(x) = G_v(x)$  for all  $j \in J^*_v(x)$ , in which case  $\nabla f_v(x) = G_v(x)$ . Consider the set

$$\Upsilon(x) := \{ v \in V : \text{ there exist } i, j \in J_v^*(x) \text{ such that } G_{iv}(x) \neq G_{jv}(x) \}.$$
(2.2.8)

The set  $\Upsilon(x)$  is the set of those  $v \in V$  for which  $f_v(\cdot)$  is not differentiable at x. The above discussion together with Propositions 2.2.1 and 2.2.2 imply the following result.

**Theorem 2.2.1** Consider the integral function  $F(\cdot)$  defined in (2.2.1) with  $f(\cdot, \cdot)$  defined in (2.2.5). Suppose that the assumptions (A1)-(A4) are satisfied. Then  $F(\cdot)$  is locally Lipschitz continuous, directionally differentiable and formula (2.2.3) holds. Moreover,  $F(\cdot)$  is differentiable at a point  $x \in X$  if and only if the set  $\Upsilon(x)$  has  $\mu$ -measure zero, and in this case (2.2.4) holds.

Clearly, for  $x \in X$ , the set  $\Upsilon(x)$  is included in the set of such  $v \in V$  that  $J_v^*(x)$  is not a singleton. Therefore, it follows from the above theorem that if  $J_v^*(x)$  is a singleton for a.e.  $v \in V$ , then  $F(\cdot)$  is differentiable at x.

Denote the set of such  $x \in X$  that  $F(\cdot)$  is differentiable at x by  $X_F$ . Since  $F(\cdot)$  is locally Lipschitz continuous, we have by Rademacher's theorem that  $F(\cdot)$  is differentiable almost everywhere, i.e., the set  $X \setminus X_F$  has Lebesgue measure zero. We say that  $F(\cdot)$  is  $X_F$ -continuously differentiable at a point  $\bar{x} \in X$  if  $\bar{x} \in X_F$  and

$$\lim_{X_F \ni x \to \bar{x}} \nabla F(x) = \nabla F(\bar{x}).$$

Note that it is assumed in the above that  $F(\cdot)$  is differentiable at  $\bar{x}$ , but not necessarily at all x near  $\bar{x}$ .

**Proposition 2.2.3** Suppose that the set J is finite, assumptions (A1)-(A3) are satisfied and, for  $\bar{x} \in X$ , the set  $J_v^*(\bar{x})$  is a singleton for a.e.  $v \in V$ . Then  $F(\cdot)$  is  $X_F$ -continuously differentiable at  $\bar{x}$ .

**Proof.** By the above discussion we have that, under the assumptions (A1)–(A3) and since  $J_v^*(x) = \{j_v\}$  is a singleton for a.e.  $v \in V$ , the integral function  $F(\cdot)$  is differentiable at  $\bar{x}$ , i.e.,  $\bar{x} \in X_F$ . Also since  $G_{jv}(\cdot)$  are continuous and J is finite, we have that if  $J_v^*(\bar{x}) = \{j_v\}$  is a singleton for some  $v \in V$ , then  $J_v^*(x) = \{j_v\}$  for all x in a neighborhood (depending on v) of  $\bar{x}$ . For such x and v we have that  $\nabla f_v(x) = G_{j_vv}(x)$ . Since V is compact and for every  $j \in J$ ,  $G_{jv}(x)$  is continuous on  $X \times V$ , there exists a constant L > 0 such that  $||G_{jv}(x)|| \leq L$  for all  $v \in V$ , x in a neighborhood of a point  $\bar{x}$  and  $j \in J$ . Consequently, by the Lebesgue Dominated Convergence theorem, we can take the following limit inside the integral

$$\lim_{X_F \ni x \to \bar{x}} \nabla F(x) = \int_V \lim_{X_F \ni x \to \bar{x}} G_{j_v v}(x) d\mu(v).$$

Then,  $X_F$ -continuity of  $\nabla F(x)$  follows from the continuity of  $G_{jv}(\cdot)$ .

In the remainder of this section, we discuss the following particular case of integral functions which is important for applications considered in Chapters 3, 4, 6 and 7.

Let  $g: X \times V \to \Re$  and consider the integral function

$$F(x) := \int_{V} [g(x,v)]_{+} d\mu(v).$$
(2.2.9)

In particular, if V = [a, b] and  $d\mu(v) = p(v)dv$ , then the above integral function  $F(\cdot)$ reduces to the function defined in (2.1.1). Clearly, the function  $f_v(x) := [g(x, v)]_+$  can be written as the maximum of the function g(x, v) and the identically zero function. The corresponding assumptions (A2) and (A3) take here the following form:

- (A5) For every  $v \in V$  the function  $g_v(\cdot) := g(\cdot, v)$  is continuously differentiable.
- (A6) The function  $(x, v) \mapsto G_v(x) := \nabla g_v(x)$  is continuous on  $X \times V$ .

From the definition of  $f_v$ , we know that  $f_v(\cdot)$  is directionally differentiable and

$$f'_{v}(x;h) = \begin{cases} [h^{T}G_{v}(x)]_{+}, & \text{if } v \in V_{0}(x), \\ 0, & \text{if } v \in V_{-}(x), \\ h^{T}G_{v}(x), & \text{if } v \in V_{+}(x), \end{cases}$$
(2.2.10)

where

$$V_0(x) := \{ v \in V : g_v(x) = 0 \},$$
  

$$V_-(x) := \{ v \in V : g_v(x) < 0 \},$$
  

$$V_+(x) := \{ v \in V : g_v(x) > 0 \}.$$

By (2.2.10), we know that for given  $x \in X$  and any  $v \in V$ , the function  $f_v(\cdot)$  is differentiable at x iff either  $v \in V_-(x) \cup V_+(x)$  or  $v \in V_0(x)$  and  $G_v(x) = 0$ . Therefore, the following result is a consequence of Propositions 2.2.1 and 2.2.2 and Theorem 2.2.1.

**Corollary 2.2.1** Suppose that the assumptions (A1), (A5) and (A6) are satisfied. Then the integral function  $F(\cdot)$  defined in (2.2.9) is locally Lipschitz continuous, directionally differentiable, and formula (2.2.3) holds. Moreover,  $F(\cdot)$  is differentiable at a point  $x \in X$  if and only if

$$\mu\left(\{v \in V_0(x) : G_v(x) \neq 0\}\right) = 0, \tag{2.2.11}$$

in which case

$$\nabla F(x) = \int_{V_+(x)} G_v(x) d\mu(s).$$
 (2.2.12)

Note that because of the condition (2.2.11), the set  $V_+(x)$  can be replaced by the set  $V_+(x) \cup V_0(x)$  without changing the value of the integral in the right hand side of (2.2.12). We have by Corollary 2.2.1 that, under the specified assumptions, the set  $X_F$ is formed by such  $x \in X$  that condition (2.2.11) holds.

Denote the set of such  $v \in V$  that  $g_v(x+h)$  and  $g_v(x)$  have the same sign by  $V_1(x,h)$ , and the set of such  $v \in V$  that  $g_v(x+h)$  and  $g_v(x)$  have different signs by  $V_2(x,h) = V \setminus V_1(x,h)$ . (By definition, we say that  $g_v(x+h)$  and  $g_v(x)$  have the same sign if one of these numbers is zero.)

**Proposition 2.2.4** Suppose that the assumptions (A1), (A5) and (A6) hold and condition (2.2.11) is satisfied at a point  $\bar{x} \in X$ . Suppose, further, that the following condition holds

$$\lim_{h \to 0} \mu \left( V_2(\bar{x}, h) \right) = 0. \tag{2.2.13}$$

Then the integral function  $F(\cdot)$  defined in (2.2.9) is  $X_F$ -continuously differentiable at  $\bar{x}$ .

**Proof.** By Corollary 2.2.1, formula (2.2.12) holds for all  $x \in X_F$ . Therefore, for all  $x = \bar{x} + h \in X_F$  in a neighborhood of  $\bar{x}$ ,

$$\|\nabla F(\bar{x}+h) - \nabla F(\bar{x})\| \leq \int_{V} \|G_{v}(\bar{x}+h) - G_{v}(x)\| d\mu(v) + \int_{V_{2}(\bar{x},h)} (\|G_{v}(\bar{x}+h)\| + \|G_{v}(x)\|) d\mu(v).$$

$$(2.2.14)$$

By the Lebesgue Dominated Convergence theorem, we can take the following limit inside the integral, and hence

$$\lim_{x \to \bar{x}} \int_{V} \|G_{v}(x) - G_{v}(\bar{x})\| \, d\mu(v) = \int_{V} \lim_{x \to \bar{x}} \|G_{v}(x) - G_{v}(\bar{x})\| \, d\mu(v) = 0.$$
(2.2.15)

We also know that the second integral in (2.2.14) is bounded by  $2L\mu(V_2(\bar{x},h))$ , where L is a constant bounding  $||G_v(x)||$  for all  $v \in V$  and x in a neighborhood of  $\bar{x}$ . It follows then by (2.2.13) and (2.2.15) that

$$\lim_{X_F \ni x \to \bar{x}} \|\nabla F(x) - \nabla F(\bar{x})\| = 0, \qquad (2.2.16)$$

which proves that  $F(\cdot)$  is  $X_F$ -continuously differentiable at  $\bar{x}$ .

Note that condition (2.2.13) alone does not imply differentiability of  $F(\cdot)$ . Think, for example, about  $g(x, v) \equiv g(x)$  independent of v and such that  $g(\bar{x}) = 0$  while  $\nabla g(\bar{x}) \neq 0$ . In that case the set  $V_2(\bar{x}, h)$  is empty and hence condition (2.2.13) holds. On the other hand,  $F(\cdot) = \mu(V)([g(\cdot)]_+)$  is not differentiable at  $\bar{x}$ .

### 2.3 Semismoothness of Integral Functions

In this section, we discuss the semismoothness and p-semismoothness properties of a class of integral functions, these properties are very important for the convergence analysis of the generalized Newton methods designed for solving semi-infinite programming problems, option price problems and  $L_2$  spectral estimation problems, which will be further discussed in Chapters 3, 6 and 7, respectively.

Consider the integral function  $F(\cdot)$  defined in (2.2.1). Suppose that in addition to the assumptions (i) and (ii) of Proposition 2.2.1,  $f_v(\cdot)$  is semismooth for a.e.  $v \in V$ , that is,  $f_v(\cdot)$  is directionally differentiable and

$$|f'_{v}(x+h;h) - f'_{v}(x;h)| \le \varepsilon_{v}(h) \|h\|, \qquad (2.3.1)$$

where  $\varepsilon_v(h) \to 0$  as  $h \to 0$ . Then

$$|F'(x+h;h) - F'(x;h)| \leq \int_{V} |f'_{v}(x+h;h) - f'_{v}(x;h)| d\mu(v) \\ \leq ||h|| \int_{V} \varepsilon_{v}(h) d\mu(v).$$
(2.3.2)

Suppose, further, that  $\varepsilon_v(h)$  is dominated by an integrable function  $\gamma(v)$  for all h in a neighborhood U of  $0 \in \Re^n$ , i.e.,  $\sup_{h \in U} \varepsilon_v(h) \leq \gamma(v)$  for a.e.  $v \in V$  and  $\int_V \gamma(v) d\mu(v) < \infty$ . Then, by the Lebesgue Dominated Convergence theorem, we can take the limit

inside the integral, and hence

$$\lim_{h \to 0} \int_{V} \varepsilon_{v}(h) d\mu(v) = \int_{V} \lim_{h \to 0} \varepsilon_{v}(h) d\mu(v) = 0.$$
(2.3.3)

It follows from (2.3.2) and (2.3.3) that  $F(\cdot)$  is semismooth.

Now, we consider the case that  $d\mu(v) = dv$ . Let f(x, v) be continuous with respect to  $v \in V$  for each fixed  $x \in \Re^n$ , and be locally Lipschitz with respect to x uniformly in  $v \in V$ , i.e., there exist a neighborhood N of 0 and a positive constant C(x) such that

$$||f(x+h,v) - f(x,v)|| \le C(x) ||h||, \quad \forall h \in N, v \in V.$$

**Proposition 2.3.1** Suppose that  $\partial_x f(x, v)$ , viewed as a joint mapping of x and v, is upper semicontinuous, i.e., for every neighborhood N of  $\partial_x f(x, v)$ , there exists  $\delta > 0$ such that

$$\partial_x f(x',v') \subset N$$
, for all  $x' \in N_1(x,\delta), v' \in N_2(v,\delta)$ ,

where

$$N_1(x,\delta) = \{x': \|x' - x\| \le \delta\}$$

and

$$N_2(v, \delta) = \{v': \|v' - v\| \le \delta\} \cap V.$$

Then F is semismooth at  $\bar{x}$  if  $f(\cdot, v)$  is semismooth at  $\bar{x}$  for every  $v \in V$ .

**Proof.** It follows from Proposition 2.2.1 that F is directionally differentiable at  $\bar{x}$ . On the other hand, by Theorem 2.7.2 in [23], we obtain

$$\partial F(x) \subset \int_{V} \partial_x f(x, v) dv.$$
 (2.3.4)

This means that for any  $Q \in \partial F(x)$ , there exists a measurable mapping  $v \to Q_v$  from V to  $\Re^n$  with  $Q_v \in \partial_x f(x, v)$  a.e. such that for every  $h \in \Re^n$ ,

$$Qh = \int_V Q_v h dv.$$

Take any  $h \in \mathbb{R}^n$  and  $Q \in \partial F(\bar{x} + h)$ . We have

$$F(\bar{x}+h) - F(\bar{x}) - Q \ h = \int_{V} \left( f(\bar{x}+h,v) - f(\bar{x},v) - Q_{v}h \right) dv,$$

where  $Q_v \in \partial_x f(\bar{x} + h, v)$ , which implies

$$|F(\bar{x}+h) - F(\bar{x}) - Q h| \le \int_{V} |f(\bar{x}+h,v) - f(\bar{x},v) - Q_{v}h| dv.$$
(2.3.5)

To prove F is semismooth, it suffices to show that

$$\lim_{h \to 0} \frac{|F(\bar{x}+h) - F(\bar{x}) - Q h|}{\|h\|} = 0.$$
(2.3.6)

Since  $f(\cdot, v)$  is semismooth at  $\bar{x}$  for every fixed  $v \in V$ , we have

$$\lim_{h \to 0} \frac{|f(\bar{x}+h,v) - f(\bar{x},v) - Q_v h|}{\|h\|} = 0, \text{ for all } Q_v \in \partial_x f(\bar{x}+h,v).$$
(2.3.7)

If there exist a neighborhood N of 0 and C > 0 such that

$$\frac{|f(\bar{x}+h,v) - f(\bar{x},v) - Q_v h|}{\|h\|} \le C,$$
(2.3.8)

for all  $h \in N$ ,  $Q_v \in \partial_x f(\bar{x} + h, v)$  and  $v \in V$ , then by the Lebesgue Dominated Convergence theorem, (2.3.6) follows from (2.3.7).

Now we prove (2.3.8). Since f is locally Lipschitz continuous at  $\bar{x}$  uniformly in  $v \in V$ , there exist a neighborhood N of 0 and  $C(\bar{x}) > 0$  such that

$$\frac{|f(\bar{x}+h,v) - f(\bar{x},v)|}{\|h\|} \le C(\bar{x}), \ \forall \ h \in N, \ v \in V.$$
(2.3.9)

On the other hand, the upper semicontinuity of  $\partial_x f(x, v)$  implies that for any  $v \in V$ and neighborhood N(v) of  $\partial_x f(\bar{x}, v)$ , there exists  $\delta_v > 0$  such that

$$\partial_x f(\bar{x}+h, v') \subset N(v)$$
, for all  $h \in N_1(0, \delta_v), v' \in N_2(v, \delta_v)$ .

Obviously,

$$V \subset \bigcup_{v \in V} N_2(v, \delta_v).$$

By the compactness of V, there exist a finite number of neighborhoods, say  $N_2(v_j, \delta_{v_j})$ ,  $j = 1, 2, \dots, m$  such that

$$V \subset \bigcup_{j=1}^m N_2(v_j, \delta_{v_j}).$$

Let  $\bar{\delta} = \min\{\delta_{v_1}, \cdots, \delta_{v_m}\}$ . Then we have

$$\bigcup_{v' \in V} \partial_x f(\bar{x} + h, v') \subset \bigcup_{j=1}^m N(v_j), \text{ for all } h \in N_1(0, \bar{\delta}).$$

It is well known that every  $\partial_x f(\bar{x}, v_j)$  is compact,  $j = 1, 2, \dots, m$ . Consequently,  $\bigcup_{j=1}^m \partial_x f(\bar{x}, v_j)$  is compact and  $\bigcup_{j=1}^m N(v_j)$  can be taken a bounded set. Hence,

$$\bigcup_{v' \in V} \partial_x f(\bar{x} + h, v')$$

is bounded, which together with (2.3.9) implies (2.3.8) holds. We obtain the desired result and complete the proof of the proposition.

**Proposition 2.3.2** Suppose that f(x, v) is the max-function, defined in (2.2.5), and that the assumptions (A1)-(A4) hold. Then the integral function  $F(\cdot)$  is semismooth at every  $x \in X$ .

**Proof.** Assumptions (A2)–(A4) imply that for every  $v \in V$  the function  $f_v(\cdot)$  is semismooth [105]. Consider a point  $x \in X$ . By the above discussion, to prove the assertion, we only need to verify that  $\varepsilon_v(h)$  is dominated by an integrable function for all h in a neighborhood U of  $0 \in \Re^n$ . We have that the constant

$$K := \sup \{ \|G_{jv}(x+h)\| : j \in J, v \in V, h \in U \}$$

is finite, provided that the neighborhood U is compact. Also by (2.2.6) we have that  $|f'_v(x+h;h)| \leq K ||h||$  for all  $v \in V$ ,  $j \in J$ , and  $h \in U$ . It follows that  $\varepsilon_v(h)$  is dominated by the constant function  $\gamma(v) = 2K$ , and hence the proof is complete.

Suppose now that for every  $v \in V$ ,  $f_v(\cdot)$  is *p*-order semismooth [131], at a point  $x \in X$ , for  $0 . (Recall that 1-order semismoothness is called strongly semismooth, see Section 1.1.1). That is, <math>\limsup_{h\to 0} c_v(h) < \infty$ , where

$$c_v(h) := \frac{|f'_v(x+h;h) - f'_v(x;h)|}{\|h\|^{1+p}}, \ h \neq 0.$$

We have that

$$|F'(x+h;h) - F'(x;h)| \le ||h||^{1+p} \int_V c_v(h) d\mu(v).$$
(2.3.10)

Therefore, in order to show that  $F(\cdot)$  is p-order semismooth at x, we need to verify that

$$\limsup_{h \to 0} \int_{V} c_v(h) d\mu(v) < \infty.$$
(2.3.11)

As an example, consider the *p*-order semismoothness of the integral function  $F(\cdot)$  defined in (2.2.9). It is easy to know that  $c_v(h) \leq q_v(h)$  with

$$q_{v}(h) := \begin{cases} \|h\|^{-p} \|G_{v}(x+h) - G_{v}(x)\|, & \text{if } v \in V_{1}(x,h), \\ \|h\|^{-p} \max\{\|G_{v}(x+h)\|, \|G_{v}(x)\|\}, & \text{if } v \in V_{2}(x,h). \end{cases}$$
(2.3.12)

**Theorem 2.3.1** Suppose that the assumptions (A1), (A5) and (A6) hold. Then the integral function  $F(\cdot)$  defined in (2.2.9) is semismooth. Suppose, further, that the following two conditions hold: there exists an integrable function  $\eta: V \to \Re_+$  such that

$$\left\|G_{v}(x^{1}) - G_{v}(x^{2})\right\| \le \eta(v) \left\|x^{1} - x^{2}\right\|^{p}$$
, for all  $x^{1}, x^{2} \in X$  and a.e.  $v \in V$ , (2.3.13)

and

$$\mu(V_2(x,h)) = O(||h||^p).$$
(2.3.14)

Then  $F(\cdot)$  is p-order semismooth at x.

**Proof.** Semismoothness of  $F(\cdot)$  follows by Proposition 2.3.2. In order to show *p*-order semismoothness of  $F(\cdot)$  we need to verify (2.3.11). It is obvious that

$$\int_{V} c_{v}(h) d\mu(v) \leq \int_{V_{1}(x,h)} q_{v}(h) d\mu(v) + \int_{V_{2}(x,h)} q_{v}(h) d\mu(v).$$
(2.3.15)

By (2.3.12) and (2.3.13), one has

$$\int_{V_1(x,h)} q_v(h) d\mu(v) \le \int_V \eta(h) d\mu(v) < +\infty.$$

Since  $||G_v(x+h)||$  and  $||G_v(x)||$  are bounded for  $v \in V$ , we know by (2.3.12) and (2.3.14) that the second integral in the right hand side of (2.3.15) is also bounded and the assertion follows.

The condition (2.3.13) holds, in particular, if  $G_v(\cdot)$  is differentiable and  $\nabla G_v(x)$ is continuous on  $X \times V$ . Condition (2.3.14) is more delicate. It is clear that this condition implies condition (2.2.13). Later, we will analyze cases in which condition (2.3.14) holds. At this point, we remark that this condition for p = 1 may fail and the integral function F may not be strongly semismooth. An example of an integral function  $F(\cdot)$ , of the form (2.2.9), which is not strongly semismooth was given in Qi and Yin [136]. Ralph [139] gave a simplified example as follows: F is defined by (2.2.9),  $g(x,v) := v^2 - x$ ,  $v \in [0,1]$ ,  $x \in \Re$  and  $d\mu(v) = dv$ . Here, for x = 0 and  $h > 0, V_2(0, h) = (0, \sqrt{h})$ , and hence condition (2.3.14) does not hold. And, indeed, the integral function is not strongly semismooth in this example.

In the rest of this section, we study the *p*-order semismoothness of the integral function  $F : \Re^n \to \Re$  defined in (2.1.1). For the sake of convenience, we first recall the concept of tensor and discuss its properties.

We use  $A_n^{(k)}$  to denote a k-th order n-dimensional tensor and use  $A_{n,i_1\cdots i_k}^{(k)}$  to denote its elements. We assume  $i_l = 1, \dots, n$  for  $l = 1, \dots, k$ . We assume that  $A_n^{(k)}$  is totally symmetric, i.e.,

$$A_{n,i_1\cdots i_k}^{(k)} = A_{n,j_1\cdots j}^{(k)}$$

if  $\{j_1, \dots, j_k\}$  is any reordering of  $\{i_1, \dots, i_k\}$ . Let  $x \in \Re^n$ . Denote

$$A_n^{(k)} x^k := \sum_{i_1, \cdots, i_k=1}^n A_{n, i_1 \cdots i_k}^{(k)} x_{i_1} \cdots x_{i_k}.$$

Let  $\|\cdot\|$  be the *F*-norm of the tensor space, that is,

$$\left\|A_{n}^{(k)}\right\| = \sqrt{\sum_{i_{1},\cdots,i_{k}=1}^{n} (A_{n,i_{1}\cdots i_{k}}^{(k)})^{2}}.$$

Note that the above concept extends the F-norm concepts of matrices and vectors.

**Proposition 2.3.3** Let  $A_n^{(k)}$  be a k-th order n-dimensional tensor and  $x \in \Re^n$ . Then

$$\left\|A_{n}^{(k)}x^{k}\right\| \leq \left\|A_{n}^{(k)}\right\| \|x\|^{k}.$$
 (2.3.16)

**Proof.** We show inductively that (2.3.16) holds. It is obvious that (2.3.16) holds trivially for k = 1 because  $A_n^{(1)}$  is an *n*-dimensional vector. Let us assume that (2.3.16) holds for some k = l. Now, we show that (2.3.16) holds also for k = l + 1. By the definition of  $A^{(k)}x^k$ ,

$$\begin{aligned} \left| A_{n}^{(l+1)} x^{l+1} \right| &= \left| \sum_{i_{1}, \cdots, i_{l}, i_{l+1}=1}^{n} A_{n, i_{1} \cdots i_{l} i_{l+1}}^{(l+1)} x_{i_{1}} \cdots x_{i_{l}} x_{i_{l+1}} \right| \\ &= \left| x_{1} y_{1} + \cdots + x_{n} y_{n} \right| \\ &\leq \left\| x \right\| \left\| y \right\|, \end{aligned}$$

where  $y = (y_1, \dots, y_n)^T$ ,  $y_j = \sum_{i_1, \dots, i_l=1}^n A_{n, i_1 \dots i_l j}^{(l+1)} x_{i_1} \dots x_{i_l} = A_n^{(l)}(j) x^l$ , and  $A_n^{(l)}(j)$  is the *l*th order *n*-dimensional tensor produced by fixing the (l+1)th order subscript *j* for  $j = 1, \dots, n$ . By the assumption,

$$\left|A_{n}^{(l)}(j)x^{l}\right| \leq \left\|A_{n}^{(l)}(j)\right\| \|x\|^{l}.$$

Hence,

$$\|y\| = \sqrt{\sum_{j=1}^{n} y_j^2} \le \sqrt{\sum_{j=1}^{n} \|A_n^{(l)}(j)\|^2 (\|x\|^l)^2} = \|A_n^{(l+1)}\| \|x\|^l.$$

 $\operatorname{So}$ 

$$\left|A_{n}^{(l+1)}x^{l+1}\right| \leq \left\|A_{n}^{(l+1)}\right\| \left\|x\right\|^{l+1}.$$

By induction, (2.3.16) holds for any positive integer k. The proof is completed.

Since semismoothness of F at a point  $\bar{x}$  is a local property which depends on only the status of F near  $\bar{x}$ , in order to study p-order semismoothness of F defined by (2.1.1), we first establish the following lemma which characterizes the perturbation property of the root v(x) of  $g(x, \cdot) = 0$ , where  $g : \Re^{n+1} \to \Re$  has continuous m-order derivative. In this lemma we assume v(x) exists (but maybe not unique) for any  $x \in X$ , where X is a certain open convex set containing  $\bar{x}$ . Note that X can be a neighborhood of  $\bar{x}$ .

**Lemma 2.3.1** Let  $\bar{x} \in X$ , where  $X \subset \Re^n$  is an open convex set. Suppose that for any  $x \in X$ ,  $g(x, \cdot) = 0$  has at least one root on [a, b], denoted as v(x). Let  $\bar{v} = v(\bar{x})$  and assume that the following condition holds:

$$\begin{aligned} \nabla_{v}^{(k)}g(\bar{x},\bar{v}) &= 0, k = 1, 2, \cdots, m-1, \\ \left|\nabla_{v}^{(m)}g(x,v)\right| &> c, \forall \ x \in X, v \in [a,b], \end{aligned} \tag{2.3.17}$$

where m is a positive integer and c is a positive number. Then  $v(x) \to \bar{v}$  as  $x \to \bar{x}$ , and there exists a positive constant  $L(\bar{x}, \bar{v})$  such that

$$|v(x) - \bar{v}|^m \le L(\bar{x}, \bar{v}) ||x - \bar{x}||,$$

and  $\bar{v}$  is the unique root of  $g(\bar{x}, \cdot) = 0$ .

**Proof.** Let  $x_{n+1} = v, z = (x, x_{n+1}), \bar{z} = (\bar{x}, \bar{v})$  and

$$A_{n+1,i_1\cdots i_k}^{(k)}(z) = \left. \frac{\partial^k g}{\partial x_{i_1}\cdots \partial x_{i_k}} \right|_z, \quad i_l = 1, \cdots, n+1, \ l = 1, \cdots, k.$$
(2.3.18)

It is clear that  $A_{n+1}^{(k)}(z)$  is a totally symmetric tensor. By using (2.3.18), the standard Taylor theorem for multivariate functions [27] can be written as:

$$g(z) = g(\bar{z}) + A_{n+1}^{(1)}(\bar{z})\Delta z + \frac{A_{n+1}^{(2)}(\bar{z})\Delta z^2}{2!} + \dots + \frac{A_{n+1}^{(m-1)}(\bar{z})\Delta z^{m-1}}{(m-1)!} + \frac{A_{n+1}^{(m)}(z(\xi))\Delta z^m}{m!},$$
(2.3.19)

where  $\Delta z = z - \bar{z}$ , and  $z(\xi)$  is a point in the segment connecting z and  $\bar{z}$ . By direct computation, one has

$$\begin{aligned}
A_{n+1}^{(k)}(z)\Delta z^{k} &= \nabla_{v}^{(k)}g(z)\Delta v^{k} + C_{k}^{k-1}\left(B_{n,k}^{(1)}(z)\Delta x\right)\Delta v^{k-1} \\
&+ C_{k}^{k-2}\left(B_{n,k}^{(2)}(z)\Delta x^{2}\right)\Delta v^{k-2} + \cdots \\
&+ C_{k}^{1}\left(B_{n,k}^{(k-1)}(z)\Delta x^{k-1}\right)\Delta v \\
&+ C_{k}^{0}\left(B_{n,k}^{(k)}(z)\Delta x^{k}\right), \ k = 1, \cdots, m,
\end{aligned}$$
(2.3.20)

where

$$C_k^j = \frac{k!}{(k-j)!j!}, \quad j = 0, 1, \cdots, k$$

and  $B_{n,k}^{(l)}(z)$  is an *l*-th order *n*-dimensional tensor with components

$$B_{n,i_1\cdots i_l}^{(l)}(z) = \left. \frac{\partial^k g}{\partial x_{i_1}\cdots \partial x_{i_l} \partial v^{k-l}} \right|_z, l = 1, \cdots, k.$$

From (2.3.20) and the condition given, the formula (2.3.19) can be rewritten as follows:

$$\begin{split} -\frac{1}{m!} \nabla_{v}^{(m)} g(z(\xi)) \Delta v^{m} &= \begin{bmatrix} B_{n,1}^{(1)}(\bar{z}) \Delta x + \frac{1}{2!} C_{2}^{1} \left( B_{n,2}^{(1)}(\bar{z}) \Delta x \right) \Delta v + \cdots \\ &+ \frac{1}{(m-1)!} C_{m-1}^{m-2} \left( B_{n,m-1}^{(1)}(\bar{z}) \Delta x \right) \Delta v^{m-2} \\ &+ C_{m}^{m-1} \left( B_{n,m}^{(1)}(z(\xi)) \Delta x \right) \Delta v^{m-1} \end{bmatrix} \\ &+ \left[ \frac{1}{2!} C_{2}^{0} \left( B_{n,2}^{(2)}(\bar{z}) \Delta x^{2} \right) + \frac{1}{3!} C_{3}^{1} \left( B_{n,3}^{(2)}(\bar{z}) \Delta x^{2} \right) \Delta v + \cdots \\ &+ \frac{1}{(m-1)!} C_{m-1}^{m-3} \left( B_{n,m-1}^{(2)}(\bar{z}) \Delta x^{2} \right) \Delta v^{m-3} \\ &+ \frac{1}{m!} C_{m}^{m-2} \left( B_{n,m}^{(2)}(z(\xi)) \Delta x^{2} \right) \Delta v^{m-2} \end{bmatrix} \\ &+ \cdots \\ &+ \left[ \frac{1}{(m-1)!} C_{m-1}^{1} \left( B_{n,m-1}^{(m-1)}(\bar{z}) \Delta x^{m-1} \right) \\ &+ \frac{1}{m!} C_{m}^{1} \left( B_{n,m}^{(m-1)}(z(\xi)) \Delta x^{m-1} \right) \Delta v \right] \\ &+ \frac{1}{m!} C_{m}^{0} \left( B_{n,m}^{(m)}(z(\xi)) \Delta x^{m} \right). \end{split}$$

Since  $\left|\nabla_{v}^{(m)}g(x,v)\right| > c$ , by Proposition 2.3.3, we have that

$$\begin{aligned} |\Delta v|^{m} &\leq m! \left| \nabla_{v}^{(m)} g(z(\xi)) \right|^{-1} \left\{ \left[ \left\| B_{n,1}^{(1)}(\bar{z}) \right\| + \frac{1}{2!} C_{2}^{1} \left\| B_{n,2}^{(1)}(\bar{z}) \right\| \left| \Delta v \right| + \cdots \right. \right. \\ &+ \frac{1}{(m-1)!} C_{m-1}^{m-2} \left\| B_{n,m-1}^{(1)}(\bar{z}) \right\| \left| \Delta v \right|^{m-2} + C_{m}^{m-1} \left\| B_{n,m}^{(1)}(z(\xi)) \right\| \left| \Delta v \right|^{m-1} \right] \\ &+ \left[ \frac{1}{2!} C_{2}^{0} \left\| B_{n,2}^{(2)}(\bar{z}) \right\| + \cdots + \frac{1}{(m-1)!} C_{m-1}^{m-3} \left\| B_{n,m-1}^{(2)}(\bar{z}) \right\| \left| \Delta v \right|^{m-3} \end{aligned}$$

$$+ \frac{1}{m!} C_m^{m-2} \left\| B_{n,m}^{(2)}(z(\xi)) \right\| |\Delta v|^{m-2} \right] \|\Delta x\|$$

$$+ \cdots$$

$$+ \left[ \frac{1}{(m-1)!} C_{m-1}^1 \left\| B_{n,m-1}^{(m-1)}(\bar{z}) \right\| + \frac{1}{m!} C_m^1 \left\| B_{n,m}^{(m-1)}(z(\xi)) \right\| |\Delta v| \right] \|\Delta x\|^{m-2}$$

$$+ \frac{1}{m!} C_m^0 \left\| B_{n,m}^{(m)}(z(\xi)) \right\| \|\Delta x\|^{m-1} \right\} \|\Delta x\|$$

It is clear that all coefficients of  $\|\Delta x\|^k$  and  $|\Delta v|^k$ ,  $(k = 1, 2, \dots, m-1)$  are bounded. Consequently, from the fact that  $\left|\nabla_v^{(m)}g(z(\xi))\right|^{-1}$  is bounded, there exists a positive constant  $\bar{L}(\bar{x}, \bar{v})$  such that

$$|\Delta v|^m \le \bar{L}(\bar{x}, \bar{v}) \|\Delta x\|.$$

This shows that  $v(x) \to \overline{v}$  as  $x \to \overline{x}$ , and  $\overline{v}$  is the unique root of  $g(\overline{x}, \cdot) = 0$ . The proof is completed.

**Remark 2.3.1** The second item in (2.3.17), i.e.,  $|\nabla_v^{(m)}g(x,v)| > c$ , for all  $x \in X$ ,  $v \in [a,b]$  characterizes the uniform sharpness of the curve family  $\{g(x,\cdot): x \in X\}$  on interval [a,b] in a sense. For instance, in the case m = 2, since the value of  $\nabla_v^{(2)}g(x,v)$  characterizes the "convexity degree" of curve  $g(x,\cdot)$ , the second item in (2.3.17) shows the curve family  $\{g(x,\cdot): x \in X\}$  has at least "convexity degree" c on interval [a,b].

We now return to considering the  $F(\cdot)$  defined by (2.1.1). We give a sufficient condition under which F is *p*-order semismooth at a given point  $\bar{x}$ .

**Theorem 2.3.2** Consider the integral function  $F(\cdot)$  defined by (2.1.1) at a point  $\bar{x} \in \Re^n$ . Suppose that:

(i)  $g_v(x) = g(x, v)$  is m-order continuously differentiable, jointly in x and v, where m is a certain positive integer,

- (ii)  $V_0(\bar{x})$  is a singleton set, denoted as  $\{\bar{v}\}$ ,
- (iii)  $\bar{v}$  is an m-th order root of  $g(\bar{x}, \cdot) = 0$ , that is,

$$\begin{cases} \nabla_v^{(k)} g(\bar{x}, \bar{v}) = 0, \ k = 1, 2, \cdots, m - 1, \\ \nabla_v^{(m)} g(\bar{x}, \bar{v}) \neq 0, \end{cases}$$
(2.3.21)

(iv) there exists an integrable function  $\eta(v)$  such that

$$\|G_v(x+h) - G_v(x)\| \le \eta(v) \ \|h\|^{\frac{1}{m}}.$$
(2.3.22)

Then  $F(\cdot)$  is  $\frac{1}{m}$ -order semismooth at  $\bar{x}$ .

**Proof.** By Theorem 2.3.1, we only need to check that whether (2.3.14) holds. Since  $\nabla_v^{(m)}g(x,v)$  is continuous at  $(\bar{x},\bar{v})$  and  $\bar{d} := \nabla_v^{(m)}g(\bar{x},\bar{v}) \neq 0$ , there exist a neighborhood U of  $\bar{x}$  and a subinterval  $[\bar{v} - \delta, \bar{v} + \delta]$  such that

$$\min_{\in U, v \in [\bar{v} - \delta, \bar{v} + \delta]} \left| \nabla_v^{(m)} g(x, v) \right| > \left| \frac{\bar{d}}{2} \right|.$$

Take any  $h \in \Re^n$  with  $\bar{x} + h \in U$ . To prove the assertion, we consider the following four cases: (i)  $\bar{d} < 0$  and m is an even number; (ii)  $\bar{d} < 0$  and m is an odd number; (iii)  $\bar{d} > 0$  and m is an even number; and (iv)  $\bar{d} > 0$  and m is an odd number. We now only discuss cases (i) and (ii), the proof for the other two cases is similar.

For case (i), it is not difficult to know that

x

$$g(\bar{x}, v) \le g(\bar{x}, \bar{v}) = 0, \ \forall \ v \in [a, b].$$
 (2.3.23)

Without loss of generality, we assume that  $a < \bar{v} < b$ . If  $V_0(\bar{x}+h) = \emptyset$ , then  $V_-(\bar{x}+h) = [a, b]$ . Hence,  $V_2(\bar{x}+h) = \emptyset$ . Now, we assume that  $V_0(\bar{x}+h) \neq \emptyset$ . Let

$$\hat{v}(h) := \sup \{ v : v \in V_0(\bar{x} + h) \},\$$
  
 $\tilde{v}(h) := \inf \{ v : v \in V_0(\bar{x} + h) \}.$ 

By the continuity of g, we know that  $V_0(\bar{x} + h)$  is a closed set. Consequently,  $\hat{v}(h)$ ,  $\tilde{v}(h) \in V_0(\bar{x} + h)$ . We assume, shrinking U if necessary, that  $v(x) \in [\bar{v} - \delta, \bar{v} + \delta]$  for all  $v(x) \in V_0(x)$  and  $x \in U$ . From  $g(\bar{x}, b) < 0$ , we know that  $g(\bar{x} + h, b) < 0$  whenever ||h|| is small enough. We conclude that for any  $v \in (\hat{v}(h), b]$  or  $v \in [a, \tilde{v}(h))$ ,

$$g(\bar{x}+h,v) < 0.$$

In fact, if there exists, without loss of generality, a  $v' \in (\hat{v}(h), b]$  such that

$$g(\bar{x}+h,v') > 0,$$

then, by the Mean-Value Theorem, there exists a v'' on the open line segment from v' to b such that

$$g(\bar{x}+h,v'')=0,$$

which contradicts the definition of  $\hat{v}(h)$ . So,

$$V_2(\bar{x},h) \subseteq [\tilde{v}(h), \hat{v}(h)].$$

Further, since  $\hat{v}(h)$ ,  $\tilde{v}(h) \in [\bar{v} - \delta, \bar{v} + \delta]$ , applying Lemma 2.3.1 to the case that X = Uand  $[a, b] = [\bar{v} - \delta, \bar{v} + \delta]$ , we obtain

$$\mu(V_2(\bar{x},h)) \le \Delta \hat{v}(h) + \Delta \tilde{v}(h) = O(\|h\|^{\frac{1}{m}}),$$

where  $\Delta \hat{v}(h) = |\hat{v}(h) - \bar{v}|$  and  $\Delta \tilde{v}(h) = |\tilde{v}(h) - \bar{v}|$ .

For case (ii), one has

$$g(\bar{x}, v) > 0, \ \forall \ v \in [a, \bar{v}),$$
$$g(\bar{x}, v) < 0, \ \forall \ v \in (\bar{v}, b].$$

That is,  $V_+(\bar{x}) = [a, \bar{v})$  and  $V_-(\bar{x}) = (\bar{v}, b]$ . Hence, for any  $h \in \mathbb{R}^n$ ,  $V_0(\bar{x} + h) \neq \emptyset$ whenever ||h|| is small enough, and

$$\begin{split} g(\bar{x}+h,v) &> 0, \ \forall \ v \in [a,\tilde{v}(h)), \\ g(\bar{x}+h,v) &< 0, \ \forall \ v \in (\hat{v}(h),b]. \end{split}$$

So,

$$V_2(\bar{x},h) \subseteq [\tilde{v}(h),\bar{v}] \cup [\bar{v},\hat{v}(h)].$$

Similarly to case (i), we obtain also

$$\mu(V_2(\bar{x},h)) \le O(\|h\|^{\frac{1}{m}}).$$

The proof is completed.  $\blacksquare$ 

**Remark 2.3.2** In Theorem 2.3.2, the condition (ii) is not essential since the sum of a finite number of p-order semismooth functions is still a p-order semismooth function. Suppose the set  $V_0(\bar{x}) = \{v \in [a,b] : g_v(\bar{x}) = 0\}$  is finite and the highest order of roots is m, then by separating [a,b] into a certain number of subintervals such that every subinterval contains only a single root of  $g_v(\bar{x}) = 0$ , we know that F is the sum of the corresponding integral functions defined on these subintervals. By Theorem 2.3.2, these integral functions are at least  $\frac{1}{m}$ -order semismooth at  $\bar{x}$  and so is F. Note, condition (2.3.21) is easier to check than (2.3.14).

With the above results and the results of [132] at hand, we may identify whether a particular integral function F is differentiability, (*p*-order) semismooth, almost smooth [132] or none of the above.

### 2.4 Smoothing Approximation Functions

From Section 2.2, we know that the integral function F defined in (2.2.9) is locally Lipschitz continuous and directionally differentiable, however it is nonsmooth in general since the function  $f_v(x) = [g(x, v)]_+$  is nonsmooth. Therefore, it is very difficult to compute the generalized Jacobian of the integral function F. In this section, we introduce the smoothing approximation function  $\overline{F}$  for F and study two classes of properties with respect to  $\overline{F}$ . The main difference between the two classes of properties lies in the fact that the smoothing parameter is ordinary one in the first class of properties, whereas in the second class of properties the smoothing parameter is also regarded as variable just as the original variable x.

For the function  $f_v(x) = [g(x, v)]_+$ , we may introduce many smoothing approximation functions to it. The three most frequently used Gabriel-Moré type smoothing approximation functions in the literature are as follows:

$$\bar{g}_1(t, x, v) = t \ln(1 + e^{g(x, v)/t}),$$
(2.4.1)

$$\bar{g}_2(t,x,v) = \frac{\sqrt{(g(x,v))^2 + 4t^2} + g(x,v)}{2}$$
(2.4.2)

and

$$\bar{g}_{3}(t,x,v) = \begin{cases} 0, & \text{if } g(x,v) < -t^{2}; \\ (g(x,v)+t^{2})^{2}/4t^{2}, & \text{if } -t^{2} \leq g(x,v) \leq t^{2}; \\ g(x,v), & \text{if } g(x,v) > t^{2}, \end{cases}$$
(2.4.3)

where  $t \neq 0$  is smoothing approximation parameter. In some literatures,  $\bar{g}_1(t, x, v)$ ,  $\bar{g}_2(t, x, v)$  and  $\bar{g}_3(t, x, v)$  are called the neural networks smoothing plus function, the Chen-Harker-Kanzow-Smale smoothing function and the Zang smoothing plus function of  $[g(x, v)]_+$ , respectively, see [130].

Let

$$\bar{F}(t,x) = \int_{V} \bar{g}(t,x,v) d\mu(v),$$
 (2.4.4)

where  $\bar{g}(t, x, v)$  may be any one of  $\bar{g}_1(t, x, v)$ ,  $\bar{g}_2(t, x, v)$  and  $\bar{g}_3(t, x, v)$  mentioned above. Clearly, for any  $t \neq 0$ ,  $\bar{F}(t, \cdot)$  is continuously differentiable in  $\Re^n$  and

$$\nabla_x \bar{F}(t,x) = \int_V \nabla_x \bar{g}(t,x,v) d\mu(v). \qquad (2.4.5)$$

Now we study the first class of properties of  $\overline{F}$ . To this end, we first give a generalized Jacobian formula for F.

For  $\bar{x} \in \Re^n$ , let  $\Lambda_0$  be the set of all mappings  $\lambda : V_0(\bar{x}) \to [0, 1]$  such that for every  $\xi \in \Re^n$ , the function  $v \mapsto \lambda(v) G_v(\bar{x})^T \xi$  belongs to  $L_1(V, \Re)$ . We have the following result.

Lemma 2.4.1 Suppose that the assumptions (A1) and (A6) hold. Then we have

$$\partial F(\bar{x}) = \left\{ \int_{V_+(\bar{x})} G_v(\bar{x})^T d\mu(v) + \int_{V_0(\bar{x})} \lambda(v) G_v(\bar{x})^T d\mu(v) : \lambda \in \Lambda_0 \right\}.$$

**Proof.** Since  $\Re^n$  is a separable space and  $f_v(\cdot)$  is regular at  $\bar{x}$ , we obtain, by Theorem 2.7.2 in [23], that

$$\partial F(\bar{x}) = \int_{V} \partial f_{v}(\bar{x}) d\mu(v). \qquad (2.4.6)$$

Here, the interpretation of (2.4.6) is as follows: For every  $\zeta$  in  $\partial F(\bar{x})$ , there exists a mapping  $v \mapsto \zeta_v$  from V to  $\Re^n$  with

$$\zeta_v \in \partial f_v(\bar{x}) \quad a.e.,$$

and having the property that for every  $\xi \in \Re^n$ , the function  $v \mapsto \zeta_v \cdot \xi$  belongs to  $L_1(V, \Re)$  and

$$\zeta \cdot \xi = \int_V \zeta_v \cdot \xi d\mu(v).$$

It is readily shown that  $\partial f_v(\bar{x}) = \{0\}$  for  $v \in V_-(\bar{x}), \ \partial f_v(\bar{x}) = \{G_v(\bar{x})^T\}$  for  $v \in V_+(\bar{x})$ and  $\partial f_v(\bar{x}) = \{\lambda G_v(\bar{x})^T : \lambda \in [0,1]\}$  for any  $v \in V_0(\bar{x})$ . Thus, we obtain the desired result and complete the proof.  $\blacksquare$ 

**Theorem 2.4.1** Suppose that the assumptions (A1) and (A6) hold. Then

(i)  $\overline{F}$  is a smoothing approximation function for F, that is,  $\overline{F}$  is continuously differentiable with respect to the second variable for any  $t \neq 0$  and there is a constant C > 0such that for any  $x \in \Re^n$  and  $t \in \Re$ ,

$$|\bar{F}(t,x) - F(x)| \le C|t|.$$

(ii) We have

$$\lim_{x \to \bar{x}, t \to 0} \operatorname{dist}(\nabla_x^T \bar{F}(t, x), \partial F(\bar{x})) = 0.$$
(2.4.7)

**Proof.** For the sake of simplicity, we only consider the case that  $\bar{g}(t, x, v) = \bar{g}_3(t, x, v)$ , where  $\bar{g}_3(t, x, v)$  is defined by (2.4.3). The other two functions can be discussed in a similar way. It follows from (2.2.9), (2.4.3) and (2.4.4) that (i) holds.

Now we prove (2.4.7) holds. To this end, we only need to prove that for any sequence  $\{(t_k, x^k)\}$  with  $(t_k, x^k) \to (0, \bar{x})$  and  $t_k \neq 0$ , there exists a subsequence  $\{(t_{k_l}, x^{k_l})\}$  of  $\{(t_k, x^k)\}$  such that

$$\lim_{l \to \infty} \nabla_x^T \bar{F}(t_{k_l}, x^{k_l}) \in \partial F(\bar{x}).$$

By simple computation,

$$\nabla_x \bar{g}(t, x, v) = \begin{cases} 0, & \text{if } g(x, v) < -t^2; \\ \frac{g(x, v) + t^2}{2t^2} G_v(x), & \text{if } -t^2 \le g(x, v) \le t^2; \\ G_s(x), & \text{if } g(x, v) > t^2. \end{cases}$$

Then

$$\begin{aligned} \nabla_x \bar{F}(t,x) &= \int_{V_0(x,t)} \frac{g(x,v) + t^2}{2t^2} G_v(x) d\mu(v) + \int_{V_+(x,t)} G_v(x) d\mu(v) \\ &= \int_V \chi_{V_0(x,t)}(v) \frac{g(x,v) + t^2}{2t^2} G_v(x) d\mu(v) \\ &+ \int_V \chi_{V_+(x,t)}(v) G_v(x) d\mu(v), \end{aligned}$$

where  $V_0(x,t) = \{v \in V : -t^2 \le g(x,v) \le t^2\}, V_+(x,t) = \{v \in V : g(x,v) > t^2\}$ , and for a set  $A, \chi_A(\cdot)$  is defined by

$$\chi_A(s) = \begin{cases} 1, & s \in A, \\ 0, & \text{otherwise.} \end{cases}$$
(2.4.8)

Moreover, we obtain that

$$\nabla_{x}\bar{F}(t,x) = \int_{V_{+}(\bar{x})} \chi_{V_{0}(x,t)}(v) \frac{g(x,v)+t^{2}}{2t^{2}} G_{v}(x) d\mu(v) \\
+ \int_{V_{0}(\bar{x})} \chi_{V_{0}(x,t)}(v) \frac{g(x,v)+t^{2}}{2t^{2}} G_{v}(x) d\mu(v) \\
+ \int_{V_{-}(\bar{x})} \chi_{V_{0}(x,t)}(v) \frac{g(x,v)+t^{2}}{2t^{2}} G_{v}(x) d\mu(v) \\
+ \int_{V_{+}(\bar{x})} \chi_{V_{+}(x,t)}(v) G_{v}(x) d\mu(v) \\
+ \int_{V_{0}(\bar{x})} \chi_{V_{+}(x,t)}(v) G_{v}(x) d\mu(v) \\
+ \int_{V_{-}(\bar{x})} \chi_{V_{+}(x,t)}(v) G_{v}(x) d\mu(v).$$
(2.4.9)

For any  $v \in V_{-}(\bar{x})$ ,  $\chi_{V_{+}(x,t)}(v) = 0$  whenever (t,x) is enough close to  $(0,\bar{x})$ , which implies  $\lim_{k\to\infty} \chi_{V_{+}(x^{k},t_{k})}(v)G_{v}(x^{k}) = 0$  for any  $v \in V_{-}(\bar{x})$ . So, by Lebesgue Dominated Convergence Theorem, one has

$$\lim_{k \to \infty} \int_{V_{-}(\bar{x})} \chi_{V_{+}(x^{k}, t_{k})}(v) G_{v}(x^{k}) d\mu(v) = 0.$$
(2.4.10)

For any  $v \in V_+(\bar{x})$ ,  $\chi_{V_+(x,t)}(v) = 1$  whenever (t,x) is enough close to  $(0,\bar{x})$ , which implies  $\lim_{k\to\infty} \chi_{V_+(x^k,t_k)}(v)G_v(x^k) = G_v(\bar{x})$  for any  $v \in V_+(\bar{x})$ . So, by Lebesgue Dominated Convergence Theorem again, one has

$$\lim_{k \to \infty} \int_{V_+(\bar{x})} \chi_{V_+(x^k, t_k)}(v) G_v(x^k) d\mu(v) = \int_{V_+(\bar{x})} G_v(\bar{x}) d\mu(v).$$

In a similar way, one has

$$\lim_{k \to \infty} \int_{V_+(\bar{x})} \chi_{V_0(x^k, t_k)}(v) \frac{g(x^k, v) + t_k^2}{2t_k^2} G_v(x^k) d\mu(v) = 0$$

and

$$\lim_{k \to \infty} \int_{V_{-}(\bar{x})} \chi_{V_{0}(x^{k}, t_{k})}(v) \frac{g(x^{k}, v) + t_{k}^{2}}{2t_{k}^{2}} G_{v}(x^{k}) d\mu(v) = 0.$$

Denote

$$H_{v}(x,t) = \chi_{V_{+}(x,t)}(v) + \chi_{V_{0}(x,t)}(v)\frac{g(x,v) + t^{2}}{2t^{2}}$$

Obviously,  $0 \leq H_v(x,t) \leq 1$ . Then there exist a subsequence  $\{(t_{k_l}, x^{k_l})\}$  of  $\{(t_k, x^k)\}$ and a mapping  $\lambda : V_0(\bar{x}) \to [0, 1]$  such that

$$\lim_{l \to \infty} \int_{V_0(\bar{x})} H_v(x^{k_l}, t_{k_l}) G_v(x^{k_l}) d\mu(v) = \int_{V_0(\bar{x})} \lambda(v) G_v(\bar{x}) d\mu(v) d\mu(v$$

By summing the above discussion, we know that

$$\lim_{l \to \infty} \nabla_x^T \bar{F}(t_{k_l}, x^{k_l}) = \int_{V_+(\bar{x})} G_v(\bar{x})^T d\mu(v) + \int_{V_0(\bar{x})} \lambda(v) G_v(\bar{x})^T d\mu(v) \in \partial F(\bar{x}).$$

We obtain the desired result and complete the proof.  $\blacksquare$ 

It immediately follows from Theorem 2.4.1 that the following corollary holds.

**Corollary 2.4.1** Suppose that the assumptions (A1) and (A6) hold. Then  $\overline{F}$  satisfies the Jacobian consistency property, i.e., for any  $x \in \Re^n$ ,

$$\lim_{t \to 0} \operatorname{dist}(\nabla_x^T \bar{F}(t, x), \partial F(x)) = 0.$$

In the remainder of this section, we regard the smoothing approximation parameter t in  $\overline{F}$  as a variable together with the original variable x and discuss the semismoothness of  $\overline{F}(\cdot, \cdot)$  in the case when  $d\mu(v) = dv$ .

**Theorem 2.4.2** The function  $\overline{F}$  has the following properties:

- (i) It is twice continuously differentiable for any  $t \neq 0$ .
- (ii) The function  $\overline{F}(\cdot, \cdot)$  is semismooth.

**Proof.** It is obvious that (i) holds. Now we prove that (ii) holds.

By (i), we only need to show that (ii) holds on  $\bar{z} = (0, \bar{x})$ . Since the composition of semismooth functions is a semismooth function [46],  $\bar{g}(t, x, v)$  is semismooth with respect to (t, x) for any fixed  $v \in V$ . To prove the semismoothness of  $\bar{F}(t, x)$ , by Proposition 2.3.1, we only need to show that  $\partial_{(t,x)}\bar{g}(t, x, v)$  is upper semicontinuous with respect to (t, x, v) and  $\bar{g}(t, x, v)$  is locally Lipschitz with respect to (t, x) uniformly in  $v \in V$ . For any  $(x, v) \in \Re^n \times V$ , we denote g(x, v) and  $\nabla_x^T g(x, v)$  by g and  $\nabla_x^T g$ , respectively. By direct computation, we have the followings.

(i) If 
$$\bar{g}(t, x, v) = \bar{g}_1(t, x, v)$$
, where  $\bar{g}_1(t, x, v)$  is defined by (2.4.1), then  

$$\partial_{(t,x)}\bar{g}(t, x, v) = \begin{cases} \left\{ \left( \ln(1 + e^{\frac{g}{t}}) - \frac{ge^{\frac{g}{t}}}{t\left(1 + e^{\frac{g}{t}}\right)}, \frac{e^{\frac{g}{t}}}{1 + e^{\frac{g}{t}}} \nabla_x^T g \right) \right\}, & \text{if } t \neq 0 \\ \left\{ \left( 0, \lambda \nabla_x^T g \right) \ : \ \lambda \in [0, 1] \right\}, & \text{if } t = 0, \ g(x, v) \neq 0 \\ \left\{ \left( \lambda, \mu \nabla_x^T g \right) \ : \ 0 \leq \lambda \leq \phi(\mu), \mu \in [0, 1] \right\}, & \text{if } t = 0, \ g(x, v) = 0, \\ (2.4.11) \end{cases} \end{cases}$$

where  $\phi(\mu) = -[(1 - \mu)\ln(1 - \mu) + \mu\ln\mu].$ 

(ii) If  $\bar{g}(t, x, v) = \bar{g}_2(t, x, v)$ , where  $\bar{g}_2(t, x, v)$  is defined by (2.4.2), then

$$\partial_{(t,x)}\bar{g}(t,x,v) = \begin{cases} \left\{ \frac{1}{2} \left( \frac{4t}{\sqrt{g^2 + 4t^2}}, \left( 1 + \frac{g}{\sqrt{g^2 + 4t^2}} \right) \nabla_x^T g \right) \right\}, & \text{if } t \neq 0 \\ \left\{ \left( 0, \nabla_x^T g \right) \right\}, & \text{if } t = 0, \ g(x,v) > 0 \\ \left\{ \left( 0, 0 \right) \right\}, & \text{if } t = 0, \ g(x,v) < 0 \\ \left\{ \left( \lambda, \mu \nabla_x^T g \right) \ : \ \lambda^2 + (2\mu - 1)^2 \le 1 \right\}, & \text{if } t = 0, \ g(x,v) = 0. \end{cases}$$

$$(2.4.12)$$

(iii) If  $\bar{g}(t, x, v) = \bar{g}_3(t, x, v)$ , where  $\bar{g}_3(t, x, v)$  is defined by (2.4.3), then

$$\partial_{(t,x)}\bar{g}(t,x,v) = \begin{cases} \{(0,0)\}, & \text{if } t \neq 0, \ g(x,v) < -t^2 \\ \left\{ \left(\frac{t^4 - g^2}{2t^3}, \frac{t^2 + g}{2t^2} \nabla_x^T g\right) \right\}, & \text{if } t \neq 0, \ -t^2 \leq g(x,v) \leq t^2 \\ \left\{ (0, \nabla_x^T g) \right\}, & \text{if } t \neq 0, \ g(x,v) > t^2 \\ \{(0,0)\}, & \text{if } t = 0, \ g(x,v) < 0 \\ \left\{ (0, \lambda \nabla_x^T g) : \lambda \in [0,1] \right\}, & \text{if } t = 0, \ g(x,v) = 0 \\ \left\{ (0, \nabla_x^T g) \right\}, & \text{if } t = 0, \ g(x,v) > 0 \end{cases}$$
(2.4.13)

From (2.4.11), (2.4.12) and (2.4.13), it is easy to verify  $\partial_{(t,x)}\bar{g}(t,x,v)$  is upper semicontinuous with respect to (t,x,v) on  $\Re \times \Re^n \times V$  for  $\bar{g}$  defined by (2.4.1)-(2.4.3).

Now we verify that  $\overline{g}(t, x, v)$  is locally Lipschitz with respect to (t, x) uniformly in  $v \in V$ . Let z = (t, x). We now break up the verification into two cases.

First, if  $t \neq 0$ , then by the Mean-Value theorem, there exists a point  $\tilde{z}$  in the open segment connecting z and  $\bar{z}$  such that

$$\bar{g}(z,v) - \bar{g}(\bar{z},v) = \nabla_z \bar{g}(\tilde{z},v)^T (z-\bar{z}).$$

By (2.4.11), (2.4.12) and (2.4.13), it is easy to know that there exists C > 0 such that

$$|\bar{g}(z,v) - \bar{g}(\bar{z},v)| \le C ||z - \bar{z}||, \ \forall \ v \in V,$$
(2.4.14)

since g is continuously differentiable and V is compact.

The second case is that t = 0. One has

$$\begin{aligned} |\bar{g}(z,v) - \bar{g}(\bar{z},v)| &= |[g(x,v)]_{+} - [g(\bar{x},v)]_{+}| \\ &\leq 2 |g(x,v) - g(\bar{x},v)| \\ &\leq \|\nabla_{x}\bar{g}(\tilde{x},v)\| \|x - \bar{x}\| \\ &= \|\nabla_{x}\bar{g}(\tilde{x},v)\| \|z - \bar{z}\|, \end{aligned}$$
(2.4.15)

where  $\tilde{x}$  is in the open segment connecting x and  $\bar{x}$ , the first inequality comes from the fact that  $|[a]_+ - [b]_+| \leq 2 |a - b|$ . By (2.4.15) and the condition that g is continuously differentiable and V is compact, there exists C > 0 such that (2.4.14) holds. The proof is complete.

**Remark 2.4.1** It is not difficult to know that  $\bar{g}(\cdot, \cdot, v)$  is strongly semismooth for every  $v \in V$ . But, the following example shows that  $\bar{F}(\cdot, \cdot)$  may not be strongly semismooth in general.

**Example 2.4.1** Let  $g(x,v) = v^2 - x$ , V = [0,1] and  $\bar{g}(t,x,v) = \bar{g}_2(t,x,v)$ , where  $\bar{g}_2(t,x,v)$  is defined by (2.4.2). We consider the strongly semismoothness of  $\bar{F}$  at (0,0). Let  $\Delta z := (\Delta t, \Delta x) = (\Delta t, 0)$  and  $\Delta t > 0$ . By a direct computation, we have

$$\bar{F}(0,0) = \int_0^1 v^2 dv,$$

$$\bar{F}(\triangle t, 0) = \frac{1}{2} \int_0^1 \left( \sqrt{v^4 + 4(\triangle t)^2} + v^2 \right) dv$$

and

$$\partial \bar{F}(\Delta t, 0) = \left\{ \left( \frac{1}{2} \int_0^1 \frac{4\Delta t}{\sqrt{v^4 + 4(\Delta t)^2}} dv, \frac{1}{2} \int_0^1 \frac{-v^2}{\sqrt{v^4 + 4(\Delta t)^2}} dv \right) \right\}.$$

Consequently, for  $Q \in \partial \bar{F}(\triangle t, 0)$ ,

$$\frac{\left|\bar{F}(\Delta t,0) - \bar{F}(0,0) - Q \Delta z\right|}{\left\|\Delta z\right\|^{2}} = 2\int_{0}^{1} \frac{v^{2}}{\left(\sqrt{v^{4} + 4(\Delta t)^{2}} + v^{2}\right)\sqrt{v^{4} + 4(\Delta t)^{2}}}dv$$
$$\geq \int_{0}^{1} \frac{v^{2}}{v^{4} + 4(\Delta t)^{2}}dv$$
$$\geq \int_{0}^{1} \frac{v^{3}}{v^{4} + 4(\Delta t)^{2}}dv.$$

It is obvious that  $\int_0^1 \frac{v^3}{v^4 + 4(\Delta t)^2} dv \to \infty$  as  $\Delta t \to 0^+$ . This shows that  $\bar{F}$  is not strongly semismooth at (0,0).

# Chapter 3

# Numerical Methods for SIP Problems

## 3.1 Introduction

We consider the following SIP problem:

$$\min_{x} f(x)$$
s.t.  $g(x, v) \le 0, \quad v \in V,$ 

$$(3.1.1)$$

where  $f: \Re^n \to \Re$  and  $g: \Re^n \times \Re^m \to \Re$  are continuously differentiable and  $V \subset \Re^m$ is a compact set. As mentioned in Section 1.2, the SIP problem (3.1.1) has a strong practical background, that is why this problem received much attention in recent 30 years. There are many numerical methods for solving the SIP problems, see Section 1.2 for details. The main effort of existing methods is to reduce the infinite set Vto a finite one. At the turn of last decade, Teo and his colleagues [81,173], by using an integral aggregate technique, converted the SIP problem (3.1.1) into a nonlinear programming problem with one constraint, and then gave a computable algorithm by solving an approximation problem of (3.1.1). The solution obtained by the algorithm in [81,173] is an approximate solution of (3.1.1).

On the other hand, some generalized Newton Methods for solving the SIP problems were presented by Li, Qi, Tam and Wu [98] and Qi, Wu and Zhou [134]. The main idea of these methods may be described as follows. Recall

$$V(x) = \{ v \in V : g(x, v) = 0 \}.$$

From Subsection 1.2.1, we know that if x is a local minimizer of the SIP problem (3.1.1) and EMFCQ holds at x, then the following KKT system of (3.1.1) holds,

$$\begin{cases} \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i) = 0, \\ g(x, v) \le 0, \quad \forall \ v \in V, \\ u_i > 0, \quad g(x, v^i) = 0, \ i = 1, \cdots, p, \end{cases}$$
(3.1.2)

where  $v^i \in V(x)$  for  $i = 1, \dots, p$ . In this case, x is called a stationary point of the SIP problem, and  $u \equiv (u_1, \dots, u_p) \in \Re^p$  and  $v^i$  for  $i = 1, \dots, p$  are called its Lagrange multiplier and attainers, respectively.

Consider the case that

$$V = \{ v \in \Re^m : c(v) \le 0 \},\$$

where  $c: \Re^m \to \Re^q$  are twice continuously differentiable functions. By the definition of V(x) and the second constrained condition of (3.1.2),  $v^i \in V(x)$   $(i = 1, \dots, p)$  imply that  $v^i$   $(i = 1, \dots, p)$  are global maximizers of the nonlinear programming problem

$$\max_{v \in V} g(x, v). \tag{3.1.3}$$

It is well known that if a constraint qualification (CQ) for the problem (3.1.3) holds, then there are p auxiliary Lagrange multipliers  $w^i \equiv (w_1^i, \cdots, w_q^i) \in \Re^q$ ,  $(i = 1, \cdots, p)$ such that for  $i = 1, \cdots, p$ ,

$$\begin{cases} -\nabla_{v}g(x,v^{i}) + \sum_{j=1}^{q} w_{j}^{i}\nabla c_{j}(v^{i}) = 0, \\ w_{j}^{i} \geq 0, \ c_{j}(v^{i}) \leq 0, \\ w_{j}^{i} \ c_{j}(v^{i}) = 0, \ j = 1, \cdots, q. \end{cases}$$
(3.1.4)

It is well known that the traditional CQ's for nonlinear programming problem include the linear independence CQ (LICQ) [102], the Slater CQ (SLCQ) [102], the Mangasarian-Fromovitz CQ (MFCQ) [102], [148], the constant rank CQ (CRCQ) [80], [133] etc..

System (3.1.4) is the first order necessary condition for  $v^i$ ,  $i = 1, \dots, p$  to be local solutions of (3.1.3). If some second order sufficiency conditions hold for (3.1.3) at  $v^i$  for

 $i = 1, \dots, p$ , then  $v^i, i = 1, \dots, p$  are local solutions of (3.1.3). Thus, the system (3.1.2) and  $v^i \in V(x)$   $(i = 1, \dots, p)$  are transformed into the following system:

$$\begin{cases} \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i) = 0, \\ g(x, v) \le 0, \quad \forall \ v \in V, \\ u_i > 0, \quad g(x, v^i) = 0, \ i = 1, \cdots, p, \\ -\nabla_v g(x, v^i) + \sum_{j=1}^{q} w_j^i \nabla c_j(v^i) = 0, \\ w_j^i \ge 0, \ c_j(v^i) \le 0, \\ w_j^i \ c_j(v^i) = 0, \ i = 1, \cdots, p, j = 1, \cdots, q. \end{cases}$$
(3.1.5)

It is then desirable to develop numerical methods for solving (3.1.1) on the basis of (3.1.5). However, in order to possess the nonsingularity conditions required by the algorithms proposed in [98] and [134], the above system should be modified accordingly. Since  $u_i > 0$  for  $i = 1, 2, \dots, p$ , we may multiply the fourth equation in (3.1.5) by  $u_i$  and then further replace  $u_i w_j^i$  by  $w_j^i$  for  $i = 1, 2, \dots, p$ ;  $j = 1, 2, \dots, q$ . Thus system (3.1.5) is equivalent to the following:

$$\begin{cases} \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i) = 0, \\ g(x, v) \le 0, \quad \forall \ v \in V, \\ u_i > 0, \quad g(x, v^i) = 0, \ i = 1, \cdots, p, \\ -u_i \nabla_v g(x, v^i) + \sum_{j=1}^{q} w_j^i \nabla c_j(v^i) = 0, \\ w_j^i \ge 0, \ c_j(v^i) \le 0, \\ w_j^i \ c_j(v^i) = 0, \ i = 1, \cdots, p, \ j = 1, \cdots, q. \end{cases}$$
(3.1.6)

Based on (3.1.6) except the feasibility constraints, a semismooth Newton method and a smoothing Newton method were presented in [134] and [98], respectively. The advantage of these two methods proposed in [98, 134] is that, at each iteration, only a system of linear equations needs to be solved. Moreover, these methods enjoy global convergence and locally superlinear convergence rate. However, these two methods cannot ensure the feasibility of (3.1.1), since the second feasibility constraint in (3.1.6) is omitted. Quite recently, another iterative method for solving the KKT system of (3.1.1) was proposed in [187], in which the feasibility issue was considered. However, the method in [187] does not have locally superlinear convergence property. In this chapter, we will present three kinds of algorithms for solving SIP problems, say, smoothing SQP algorithm, smoothing projected Newton-type algorithm and smoothing Newton-type algorithm. We should point out that the smoothing projected Newton-type algorithm and smoothing Newton-type algorithm have the following two features:

- (1) At each iteration, only a system of linear equations needs to be solved;
- (2) These methods have global and local superlinear convergence property.

The following is the outline of this chapter. In Section 3.2, we first convert the SIP problem (3.1.1) into a nonsmooth programming problem with one constraint by using an integral function. Then, we present a smoothing SQP algorithm for solving the resulted programming problem. The global convergence of the smoothing SQP algorithm is established under some mild conditions. In Section 3.3, we first reformulate the KKT system of the SIP problem (3.1.1) as a system of constrained nonsmooth equations, then present a smoothing projected Newton-type algorithm for solving the resulted system. In Section 3.4, we further reformulate the KKT system of the SIP problem (3.1.1) as a system of unconstrained nonsmooth equations, and then present a smoothing Newton-type algorithm for solving the resulted system. We prove that the later two algorithms have global and local superlinear convergence under some standard conditions in Sections 3.3 and 3.4, respectively. For the three algorithms above, numerical experiments are given in the corresponding sections and some comments are made in the last section.

# 3.2 A Smoothing SQP Algorithm

Define  $\varphi : \Re^n \to \Re$  by

$$\varphi(x) := \int_{V} [g(x,v)]_{+} d\mu(v), \qquad (3.2.1)$$

where  $\mu$  is a finite measure defined on a measurable space  $(V, \mathcal{F})$ . For any given  $x \in \Re^n$ , clearly,  $\varphi(x) \ge 0$ . Then problem (3.1.1) can be converted into the following equivalent

nonlinear programming problem with only one inequality constraint:

$$\begin{array}{ll}
\min_{x} & f(x) \\
s.t. & \varphi(x) \le 0.
\end{array}$$
(3.2.2)

Unfortunately, the function  $\varphi$  is nonsmooth. Therefore, the existing gradient-based optimization methods cannot be used to solve (3.2.2) directly. In this section, we will present a smoothing SQP algorithm for solving the resulted nonsmooth nonlinear programming problem (3.2.2).

#### 3.2.1 Some Preliminaries

In this subsection, we give some preliminaries about (3.1.1) and (3.2.2).

**Definition 3.2.1** The point  $x \in \Re^n$  is said to be a generalized stationary point of (3.2.2) if there exists a constant  $\gamma$  such that the following generalized Karush-Kuhn-Tucker (GKKT) condition holds:

$$\begin{cases} 0 \in \nabla^T f(x) + \gamma \partial \varphi(x), \\ \gamma \varphi(x) = 0, \quad \varphi(x) \le 0, \quad \gamma \ge 0. \end{cases}$$
(3.2.3)

Let x be a generalized stationary point of (3.2.2). Since  $\varphi(x) \leq 0$ ,  $V_+(x) = \emptyset$ , we in turn, by Lemma 2.4.1, have

$$\partial \varphi(x) = \left\{ \int_{V_0(x)} \lambda(v) G_v(x)^T d\mu(v) : \lambda \in \Lambda_0 \right\},$$
(3.2.4)

where  $\Lambda_0$  is the set of all mappings  $\lambda : V_0(x) \to [0, 1]$ . From the first expression of (3.2.3), there exists a mapping  $\lambda \in \Lambda_0$  such that

$$abla f(x) + \gamma \int_{V_0(x)} \lambda(v) G_v(x) d\mu(v) = 0.$$

Suppose that  $\mu|_{V_0(x)}$  has a finite support (discrete measure) with  $p \leq n$ , that is,

$$\mu|_{V_0(x)} = \sum_{i=1}^p \sigma_i \delta(v^i)$$

where  $\delta(v)$  denotes a measure of mass one at the point  $v \in V_0(x)$  and  $\sigma_i > 0$ . Then

$$\nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i) = 0,$$

where  $u_i = \gamma \lambda_{\mu}(v^i) \sigma_i \ge 0$ . Moreover, if  $u_i > 0$ , then x is a stationary point of (3.1.1). Conversely, we have **Proposition 3.2.1** Suppose that x is a stationary point of (3.1.1). Then x is a generalized stationary point of (3.2.2).

**Proof.** Since x is a stationary point of (3.1.1), there exist a nonnegative integer  $p \le n$  and multipliers  $u_i$ , i = 1, ..., p such that (3.1.2) holds. That is,

$$\begin{cases} \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i) = 0\\ g(x, v) \le 0, \ \forall \ v \in V, \\ u_i > 0, \ g(x, v^i) = 0, \ i = 1, \dots, p, \end{cases}$$
(3.2.5)

where  $v^i \in V(x), i = 1, \dots, p$ . We design the finite discrete measure  $\mu$  on V as follows

$$\mu(A) = \sum_{v^i \in A} u_i, \quad \forall \ A \in \mathcal{F}.$$

It is clear that  $\varphi(x) = 0$  and

$$0 = \nabla^T f(x) + \sum_{i=1}^p u_i \nabla^T_x g(x, v^i)$$
  
$$= \nabla^T f(x) + \int_{V(x)} G_v(x)^T d\mu(v)$$
  
$$\in \nabla^T f(x) + \beta \ \partial\varphi(x),$$

where the last expression follows from  $\partial \varphi(x) = \left\{ \int_{V(x)} \lambda(v) G_v(x)^T d\mu(v) : \lambda \in \Lambda_0 \right\}$  and  $\beta = 1$ . Therefore, x is a generalized stationary point of (3.2.2).

From the discussion above, we can see that it is possible to obtain the stationary point of the SIP problem (3.1.1) by solving the generalized stationary point of (3.2.2). In next subsection, we will present a smoothing SQP algorithm for solving the generalized stationary point of (3.2.2).

#### 3.2.2 Smoothing SQP Algorithm

For given  $x \in \Re^n$ , t > 0 and r > 0, define a quadratic program QP(t, x, r) as follows:

$$\min_{\substack{d \in \mathbb{R}^n, \, \xi \in \mathbb{R} \\ s.t. \quad \bar{\varphi}(t,x) + \nabla_x^T \bar{\varphi}(t,x) d \leq \xi, \\ \xi \geq 0,} \nabla f(x)^T d + \frac{1}{2} d^T W d + r\xi$$
(3.2.6)

where W is a symmetric positive definite matrix and

$$\bar{\varphi}(t,x) = \int_{V} \bar{g}(t,x,v) d\mu(v), \qquad (3.2.7)$$

where  $\bar{g}(t, x, v)$  may be any one of the Gabried-Moré type smoothing approximation functions  $\bar{g}_i(t, x, v)$ , i = 1, 2, 3, in (2.4.1)-(2.4.3).

Since for any t > 0,  $\bar{\varphi}(t, \cdot)$  is continuously differentiable in  $\Re^n$  and

$$\nabla_x \bar{\varphi}(t,x) = \int_V \nabla_x \bar{g}(t,x,v) d\mu(v), \qquad (3.2.8)$$

we may compute easily the gradient  $\nabla_x \bar{\varphi}(t, x)$  of the function  $\bar{\varphi}$  with respect to variable x, whenever the evaluation of the integral function (3.2.8) is not very expensive. Furthermore, since  $\bar{\varphi}(t, x) \geq 0$  for all  $x \in \Re^n$  and  $t \geq 0$ , it is readily shown that (3.2.6) is always feasible and solvable.

Let  $(d, \xi)$  be a solution of (3.2.6). Then its KKT condition can be written as follows:

$$\begin{cases} \nabla f(x) + Wd + \lambda_{\bar{\varphi}} \nabla_x \bar{\varphi}(t, x) = 0, \\ r = \lambda_{\bar{\varphi}} + \lambda_{\xi}, \\ 0 \le \xi - \bar{\varphi}(t, x) - \nabla_x^T \bar{\varphi}(t, x) d \perp \lambda_{\bar{\varphi}} \ge 0, \\ 0 \le \xi \perp \lambda_{\xi} \ge 0, \end{cases}$$
(3.2.9)

where  $(\lambda_{\bar{\varphi}}, \lambda_{\xi})$  is the corresponding KKT multiplier.

Define a penalty merit function  $\Theta$  by

$$\Theta_{(r^{\theta}, t)}(x) = f(x) + r^{\theta} \bar{\varphi}(t, x). \qquad (3.2.10)$$

Now we present our smoothing SQP algorithm for (3.2.2).

#### Algorithm 3.2.1 (Smoothing SQP Algorithm)

- Step 0. (Initialization) Let  $r_{-1} > 0$ ,  $\delta > 0$ ,  $\tau \in (0, 1)$ ,  $\sigma \in (0, 1)$  and  $\beta \in (0, 1)$ . Choose  $x^0 \in \Re^n$ ,  $t_0 > 0$  and a symmetric positive definite matrix  $W_0 \in \Re^{n \times n}$ . Set k := 0.
- **Step 1.** (Search direction) Solve (3.2.6) with  $x = x^k$ ,  $t = t_k$ ,  $W = W_k$  and  $r = r_{k-1}$ . Let  $(d^k, \xi^k)$  be a solution of (3.2.6) and  $\lambda^k = (\lambda^k_{\bar{\varphi}}, \lambda^k_{\xi})$  be its corresponding multiplier.

- Step 2. (Termination check) If a stopping rule is satisfied, terminate. Otherwise, go to Step 3.
- **Step 3.** (Penalty update) Define  $r_k^{\theta} = r_{k-1}$  and let

$$r_{k} = \begin{cases} r_{k-1}, & \text{if } \xi^{k} = 0, \\ r_{k-1} + \delta, & \text{otherwise.} \end{cases}$$
(3.2.11)

Step 4. (Line search) Let  $s_k = \tau^{i_k}$ , where  $i_k$  is the smallest nonnegative integer i such that

$$\Theta_{(r_k^\theta, t_k)}(x^k + \tau^i d^k) - \Theta_{(r_k^\theta, t_k)}(x^k) \le -\sigma \tau^i d^{kT} W_k d^k.$$
(3.2.12)

Step 5. (Update) Let  $x^{k+1} := x^k + s_k d^k$ ,  $t_{k+1} := \beta t_k$  if  $||d^k|| \le t_k$ , otherwise  $t_{k+1} := t_k$ . Choose a symmetric positive definite matrix  $W_{k+1} \in \Re^{n \times n}$ . Set k := k+1 and go to Step 1.

**Remark 3.2.1** (i) The above smoothing SQP algorithm is a modified version of the explicit smooth SQP algorithm [85], and part of our convergence analysis presented in the next section is modelled after the paper [85].

(ii) At Step 2 of the algorithm, we do not specify a stopping rule. Usually, we can stop the iteration when  $||d^k||$  and  $t_k$  are all very small. In practice, we may use the following condition as a stopping rule:  $||d^k|| + t_k \leq \hat{\tau}$ , where  $\hat{\tau}$  is a given small positive number which characterizes the precision.

(iii) At each iteration of the above algorithm the quadratic program (3.2.6) needs to be solved, which is always feasible and solvable. Traditional SQP methods solve the following quadratic program at each iteration:

$$\min_{\substack{d\in\mathfrak{R}^n\\s.t.}} \nabla f(x)^T d + \frac{1}{2} d^T W d$$

$$(3.2.13)$$

(3.2.13) may not be feasible when  $\bar{\varphi}(t,x) > 0$  and  $\nabla_x \bar{\varphi}(t,x) = 0$ .

(iv) Proposition 3.2.2 below will show that the line search in Step 4 is well defined. Therefore, the smoothing SQP method is well defined when  $t_k > 0$  and  $W_k$  is symmetric positive definite at each iteration. **Remark 3.2.2** Since the infinite inequality constraints of (3.1.1) are aggregated into an inequality constraint, the smoothing SQP algorithm may be easy to implement, whenever the evaluation of the integral function is not very expensive. For the iterative method [187] for (3.1.1), at each iteration a KKT system is solved, which is more expensive than computing a quadratic program. Also, the smoothing SQP method presented in this subsection generates a sequence with the property that at least one limit point is a generalized stationary point of (3.2.2) under some mild conditions (which will be proved in Subsection 3.2.3), but semismooth and smoothing Newton methods [98, 134] do not have this property.

#### **Proposition 3.2.2** For any t > 0,

(i)  $\Theta_{(r^{\theta}, t)}(\cdot)$  is continuously differentiable at x. Furthermore, if  $(d, \xi)$  is a solution of (3.2.6),  $\lambda = (\lambda_{\bar{\varphi}}, \lambda_{\xi})$  is its corresponding multiplier and  $r^{\theta} = r$ , then

$$\Theta'_{(r^{\theta}, t)}(x; d) \le -d^T W d - (r - \lambda_{\bar{\varphi}}) \bar{\varphi}(t, x)$$
(3.2.14)

and

$$\Theta'_{(r^{\theta},t)}(x;d) \le -d^T W d. \tag{3.2.15}$$

(ii) Suppose W is a symmetric positive definite matrix. If  $(d,\xi)$  is a solution of (3.2.6) with  $d \neq 0$ , and  $r^{\theta} = r$ , then d is a descent direction of  $\Theta_{(r^{\theta}, t)}(\cdot)$  at x.

**Proof.** It is readily shown that  $\Theta_{(r^{\theta}, t)}(\cdot)$  is continuously differentiable at x. Then we obtain

$$\Theta'_{(r^{\theta}, t)}(x; d) = \nabla f(x)^T d + r^{\theta} \nabla^T_x \bar{\varphi}(t, x) d.$$
(3.2.16)

Since  $(d,\xi)$  is a solution of (3.2.6) and  $\lambda = (\lambda_{\bar{\varphi}}, \lambda_{\xi})$  is the corresponding multiplier, (3.2.9) holds. It follows from (3.2.16) and the third inequality of (3.2.9) that

$$\Theta'_{(r^{\theta},t)}(x;d) \le \nabla f(x)^T d + r^{\theta}(\xi - \bar{\varphi}(t,x)).$$
(3.2.17)

On the other hand, by (3.2.9),

$$\nabla f(x)^T d = -d^T W d - \lambda_{\bar{\varphi}} \nabla_x^T \bar{\varphi}(t, x) d \qquad (3.2.18)$$

$$= -d^T W d - \lambda_{\bar{\varphi}}(\xi - \bar{\varphi}(t, x)). \qquad (3.2.19)$$

By (3.2.17) and (3.2.19), we obtain

$$\Theta'_{(r^{\theta}, t)}(x; d) \le -d^T W d - \lambda_{\bar{\varphi}}(\xi - \bar{\varphi}(t, x)) + r^{\theta}(\xi - \bar{\varphi}(t, x)).$$

It follows from the second equality and fourth inequality of (3.2.9) that

$$r^{\theta}\xi = \lambda_{\bar{\varphi}}\xi.$$

Hence,

$$\Theta'_{(r^{\theta}, t)}(x; d) \leq -d^{T}Wd - (r^{\theta} - \lambda_{\bar{\varphi}})\bar{\varphi}(t, x)$$
  
$$\leq -d^{T}Wd. \qquad (3.2.20)$$

This shows that (i) holds. It follows from (i) and that W is symmetric positive definite that (ii) holds.  $\blacksquare$ 

#### 3.2.3 Convergence Analysis

In order to obtain the global convergence of the smoothing SQP algorithm presented in Subsection 3.2.2, we make the following standard assumptions:

(B1) There exist two positive numbers m and M satisfying m < M such that each of the symmetric matrices  $W_k$  used in smoothing SQP algorithm satisfies the following condition that for all vectors u of appropriate dimension:

$$m\|u\|^{2} \le u^{T}W_{k}u \le M\|u\|^{2}.$$
(3.2.21)

(B2) For all large  $k, r_k = r^* > 0$ .

**Theorem 3.2.1** Assume that (B1) and (B2) hold. Let  $\{x^k\}$  and  $\{t_k\}$  be the sequences generated by the smoothing SQP algorithm. Let  $K = \{k : ||d^k|| \le t_k\}$ . If  $\{x^k\}_{k \in K}$  has an accumulation point  $x^*$ , then  $x^*$  is a generalized stationary point of (3.2.2).

**Proof.** We assume, without loss of generality, that

$$\lim_{k \to \infty, k \in K} x^k = x^*$$

It follows from the penalty update rule in Step 3 of the algorithm and the assumption (B2) that  $\xi^k = 0$  and  $r_k = r^*$  for all sufficiently large k. Moreover, the second equality

of (3.2.9) implies that  $\{\lambda^k = (\lambda_{\bar{\varphi}}^k, \lambda_{\xi}^k)\}$  is bounded. The boundedness of  $\{d^k\}_{k \in K}$  is implied by (B1) and the first equality of (3.2.9). We assume, without loss of generality, that

$$\lim_{k \to \infty, k \in K} d^k = d^*, \quad \lim_{k \to \infty, k \in K} \lambda^k_{\xi} = \lambda^*_{\xi}, \quad \lim_{k \to \infty, k \in K} \lambda^k_{\bar{\varphi}} = \lambda^*_{\bar{\varphi}} \quad \text{and} \quad \lim_{k \to \infty, k \in K} W_k = W^*.$$

The existence of  $W^*$  follows from condition (B1). Furthermore,  $W^*$  is positive definite. It follows from the monotonically decreasing property of  $\{t_k\}$  that  $\lim_{k\to\infty} t_k = t^*$  exists. We may claim that  $t^* = 0$ . Otherwise, we assume, without loss of generality, that  $t_k = t_{k_0} > 0$  for all  $k \ge k_0$ . This implies  $||d^k|| > t_{k_0}$  for all  $k \ge k_0$ . In this case, our smoothing method reduces to the modified SQP method presented in Appendix of [84] for a smooth nonlinear program. By a common argumentation for SQP method for smooth nonlinear program, it follows that some subsequence of  $\{d^k\}_{k\in K}$  approaches 0 as  $k \to \infty$ , which implies that  $||d^k|| \le t_{k_0}$  will eventually happen, which is a contradiction. Therefore,  $t^* = 0$ . Consequently,  $d^* = 0$ .

It follows from (3.2.9) and Theorem 2.4.1 that

$$0 \in \nabla^T f(x^*) + \lambda^*_{\bar{\omega}} \partial \varphi(x^*), \qquad (3.2.22)$$

$$0 \le -\varphi(x^*) \perp \lambda_{\bar{\varphi}}^* \ge 0. \tag{3.2.23}$$

We obtain the desired result and complete the proof of the theorem.

Now we give some conditions under which (B2) holds.

(B3)  $\{x^k\}$  is bounded.

(B4) Let T be the set of all vectors V satisfying that there exists a subsequence  $\{k_l\}$  of  $\{1, 2, \dots\}$  such that

$$\lim_{l \to \infty} \nabla_x \bar{\varphi}(t_{k_l}, x^{k_l}) = V.$$

We assume that  $0 \notin T$ .

In [85], the global convergence results of the algorithms are obtained under a generalized Mangasarian-Fromovitz constraint qualification (GMFCQ). However, for the nonsmooth program (3.2.2), it is readily shown that GMFCQ does not hold at any generalized stationary point  $x^*$ , since  $0 \in \partial \varphi(x^*)$  by Lemma 2.4.1.
If  $(t_{k_l}, x^{k_l}) \to (0, x^*)$  as  $l \to +\infty$ , by Theorem 2.4.1 (ii)  $\lim_{l\to\infty} \nabla_x^T \bar{\varphi}(t_{k_l}, x^{k_l}) \in \partial \varphi(x^*)$ . Thus, if  $t_k \to 0$ , condition (B4) is weaker than the GMFCQ condition used in [85].

**Lemma 3.2.1** Assume the assumptions (B1), (B3) and (B4) hold. If  $\{r_k\} \to \infty$ , then

- (i)  $\{(d^k, \xi^k)\}$  is bounded;
- (ii)  $\{\lambda_{\bar{\omega}}^k/r_{k-1}\} \rightarrow 0$ ; and
- (iii)  $\xi^k = 0$  for all sufficiently large k.

**Proof.** (i) For a contradiction, let  $\{(x^k, W_k)\}_{k \in L}$  be a convergent subsequence with limit  $(x^*, W^*)$  such that  $\{\|(d^k, \xi^k)\|\} \to \infty$ . Without loss of generality, we suppose that

$$\lim_{k \to \infty, k \in L} \nabla_x \bar{\varphi}(t_k, x^k) = V^*.$$

By (B4),  $V^* \neq 0$ . Let  $\bar{d}$  be the vector such that  $\varphi(x^*) + V^{*T}\bar{d} < 0$ . It follows that for large enough  $k \in L$ ,  $\bar{\varphi}(t_k, x^k) + \nabla_x^T \bar{\varphi}(t_k, x^k) \bar{d} < 0$ , which implies that  $(\bar{d}, 0)$  is a feasible solution for the quadratic program  $QP(t_k, x^k, r_{k-1})$ . Hence, for any  $k \in L$ ,

$$\nabla f(x^{k})^{T} d^{k} + \frac{1}{2} d^{kT} W_{k} d^{k} \leq \nabla f(x^{k})^{T} d^{k} + \frac{1}{2} d^{kT} W_{k} d^{k} + r_{k-1} \xi_{k} \\ \leq \nabla f(x^{k})^{T} \bar{d} + \frac{1}{2} \bar{d}^{T} W_{k} \bar{d}.$$

Since  $\{\nabla f(x^k)^T \bar{d} + \frac{1}{2} \bar{d}^T W_k \bar{d}\}$  is bounded for  $k \in L$ , it follows from (B1) and (B3) that  $\{d^k\}_{k\in L}$  is bounded. Finally, it is obvious from optimality of  $(d^k, \xi^k)$  that  $\xi^k = [\bar{\varphi}(t_k, x^k) + \nabla_x^T \bar{\varphi}(t_k, x^k) d^k]_+$  for each k. Hence,  $\{\xi^k\}_{k\in L}$  is also bounded. This contradicts the unboundedness condition.

(ii) We have  $d^k/r_{k-1} \to 0$  from (i) and  $r_{k-1} \to \infty$  by hypothesis. For any limit point  $(x^*, d^*, \xi^*, \bar{\lambda}_{\bar{\varphi}}, \bar{\lambda}_{\xi})$  of  $(x^k, d^k, \xi^k, \lambda_{\bar{\varphi}}^k/r_{k-1}, \lambda_{\xi}^k/r_{k-1})$ , we have, by dividing the every expression in (3.2.9) by  $r_{k-1}$  and letting  $k \to \infty$ , that

$$\begin{cases} \bar{\lambda}_{\bar{\varphi}}V^* = 0, \\ 1 = \bar{\lambda}_{\bar{\varphi}} + \bar{\lambda}_{\xi}, \\ 0 \le \xi^* - \varphi(x^*) - V^{*T}d^* \perp \bar{\lambda}_{\bar{\varphi}} \ge 0, \\ 0 \le \xi^* \perp \bar{\lambda}_{\xi} \ge 0. \end{cases}$$
(3.2.24)

By (B4),  $\bar{\lambda}_{\bar{\varphi}} = 0$ , which implies  $\{r_{\bar{\varphi}}^k/r_{k-1}\} \to 0$ .

(iii) Since  $\bar{\lambda}_{\bar{\varphi}} = 0$ , we have, from the second equality of (3.2.24), that  $\bar{\lambda}_{\xi} = 1$ , which implies that  $\lambda_{\xi}^{k}$  is strictly positive for all large k. Hence, we know, by the last inequality of (3.2.9), that  $\xi^{k} = 0$  for all sufficiently large k.

**Theorem 3.2.2** Assume the assumptions (B1), (B3) and (B4) hold. Then (B2) also holds.

**Proof.** Suppose that (B2) does not hold, in which case we have  $\{r_k\} \to \infty$ . From Lemma 3.2.1 (iii),  $r_k = r_{k-1}$  for all large k. This contradiction shows the theorem holds.

From Theorems 3.2.1 and 3.2.2, we have

**Theorem 3.2.3** Assume the assumptions (B1), (B3) and (B4) hold. Let  $\{x^k\}$  and  $\{t_k\}$  be the sequences generated by the smoothing SQP algorithm. Then  $K = \{k : ||d^k|| \le t_k\}$  is an infinite set, and every accumulation point of  $\{x^k\}_{k \in K}$  is generalized stationary point of  $\{3.2.2\}$ .

## 3.2.4 Preliminary Numerical Examples

In this subsection, we report our preliminary numerical test results. We implemented the smoothing SQP algorithm described in Subsection 3.2.2 in MATLAB and the numerical experiments were done by using a Pentium III 450MHz workstation. We compared the performance of the smoothing SQP algorithm with **fseminf** that is a solver for SIP based on an implementation of the discretization SQP method in MATLAB toolbox. We tested 5 problems which are called Problems 3.2.1-3.2.5 and can be found in [134, 187].

Problem 3.2.1

min 
$$f(x) = 1.21 \exp(x_1) + \exp(x_2)$$
  
s.t.  $g(x, v) = v - \exp(x_1 + x_2) \le 0, \quad \forall v \in [-10, 1]$   
 $x^0 = (0, 0)^T.$ 

#### Problem 3.2.2

min 
$$f(x) = x_1^2 + x_2^2 + x_3^2$$
  
s.t.  $g(x, v) = x_1 + x_2 \exp(x_3 v) + \exp(2v) - 2\sin(4v) \le 0, \quad \forall v \in [0, 1].$   
 $x^0 = (1, 1, 1)^T.$ 

#### Problem 3.2.3

min 
$$f(x) = (x_1 - 2x_2 + 5x_2^2 - x_2^3 - 13)^2 + (x_1 - 14x_2 + x_2^2 + x_2^3 - 29)^2$$
  
s.t.  $g(x, v) = x_1^2 + 2x_2v^2 + \exp(x_1 + x_2) - \exp(v) \le 0, \quad \forall \ v \in [0, 50].$   
 $x^0 = (1, -1)^T.$ 

#### Problem 3.2.4

min 
$$f(x) = \frac{1}{3}x_1^2 + \frac{1}{2}x_1 + x_2^2$$
  
s.t.  $g(x,v) = (1 - x_1^2v^2)^2 - x_1v^2 - x_2^2 + x_2 \le 0, \quad \forall v \in [-1,1].$   
 $x^0 = (1,1)^T.$ 

#### Problem 3.2.5

min 
$$f(x) = \sum_{i=1}^{n} \exp(x_i)$$
  
s.t.  $g(x, v) = 1/(1 + v^2) - \sum_{i=1}^{n} x_i v^{i-1} \le 0, \quad \forall v \in [-1, 1],$   
where  $n = 20.$ 

For Problem 3.2.5, we chose the vector of ones as the starting point.

Throughout the computational experiments, we use the function defined in (2.4.2) as a smoothing approximation function of the function  $[g(x, v)]_+$ . The parameters used in the smoothing SQP algorithm are  $r_{-1} = 10^4$ ,  $\delta = 20$ ,  $\tau = 0.8$ ,  $\sigma = 0.5$ ,  $t_0 = 10^{-5}$  and  $\beta = 0.1$ . For each k, we let  $W_k = I$  and we solve the quadratic program (3.2.6) by using **qp** in MATLAB toolbox. The values of  $\bar{\varphi}(t_k, x^k)$  and  $\nabla_x \bar{\varphi}(t_k, x^k)$  are obtained by using **quad** in MATLAB toolbox. We use  $||d^k|| \leq 10^{-5}$  as the stopping criterion for the smoothing SQP algorithm.

The test results are summarized in Table 3.1, where **k** denotes the number of the iteration, **cpu** the CPU time in second for solving each problem,  $f(x^k)$  and  $\bar{\varphi}(t_k, x^k)$  the values of the objective function and the function  $\bar{\varphi}(t, x)$  at the final iteration, respectively.

	Smoothing SQP Algorithm				fseminf			
Problem	k	cpu	$f(x^k)$	$\bar{\varphi}(\varepsilon_k, x^k)$	k cpu		$f(x^k)$	
3.2.1	2	0.06	2.20000e+00	0.00e+00	3	0.18	2.19934e+00	
3.2.2	21	0.65	5.33502e+00	1.18e-10	24	0.58	5.33120e+00	
3.2.3	13	0.55	$9.71589e{+}01$	1.56e-08	8	2.36	9.71589e+01	
3.2.4	12	0.44	2.43054e+00	9.87e-09	6	0.13	2.43053e+00	
3.2.5	23	5.52	2.12249e+01	1.92e-08	11	7.94	2.12229e+01	

Table 3.1: Test Results for the smoothing SQP algorithm

The results reported in Table 3.1 show that the smoothing SQP algorithm performs well when the evaluation of the integral function is not very expensive. From the cpu columns of the table we can see that the smoothing SQP algorithm uses less CPU time than **fseminf** for Problems 3.2.1, 3.2.3 and 3.2.5. For the other two problems **fseminf** uses less CPU time than the smoothing SQP algorithm.

# 3.3 A Smoothing Projected Newton-Type Algorithm

The smoothing SQP algorithm presented in Section 3.2 has global convergence property and only a quadratic program needs to be solved at each iteration of the algorithm. In this section, we propose a smoothing projected Newton-type algorithm for solving the SIP problem (3.1.1), which has stronger convergence property than the method stated above, i.e., the latter method has local superlinear convergence property.

## 3.3.1 A Constrained Equation Reformulation of KKT System

We assume, in addition, that  $f : \Re^n \to \Re$  and  $g : \Re^n \times \Re^m \to \Re$  are twice continuously differentiable functions and V is a nonempty compact box with

$$V = \{ v \in \Re^m : a \le v \le b \},\$$

where  $a \in \Re^m$ ,  $b \in \Re^m$ , and a < b. Here, the inequality a < b means that  $a_i < b_i$  for all  $i = 1, 2, \dots, m$ .

Consider the KKT system (3.1.2) of the SIP problem (3.1.1). We know that  $v^i$   $(i = 1, \dots, p)$  are global minimizers of the following minimization problem:

$$\min_{\substack{x,v \\ s.t. \\ v \in V.}} -g(x,v)$$

$$(3.3.1)$$

The KKT system of (3.3.1) can be rewritten as

$$(v'-v)^T(-\nabla_v g(x,v)) \ge 0, \quad \forall \ v' \in V,$$

and it can be reformulated as a system of nonsmooth equations (see [20, 40] for details):

$$\phi(x, v) = 0. \tag{3.3.2}$$

Here,  $\phi(x, v)$  is defined as

$$\phi(x,v) := v - P(a, b, v + \nabla_v g(x, v)), \qquad (3.3.3)$$

where the function P is the mid-function defined for all  $j = 1, \dots, m$ , as

$$(P(c, d, w))_j = \begin{cases} c_j, & \text{if } w_j < c_j, \\ w_j, & \text{if } c_j \le w_j \le d_j, \\ d_j, & \text{if } d_j < w_j. \end{cases}$$

Then the KKT system of the SIP problem (3.1.1) can be reformulated as follows:

$$\begin{cases} \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i) = 0, \\ g(x, v) \le 0, \quad \forall \ v \in V, \\ u_i > 0, \quad g(x, v^i) = 0, \quad (i = 1, \cdots, p) \\ \phi(x, v^i) = 0 \quad (i = 1, \cdots, p). \end{cases}$$
(3.3.4)

Let

$$G(x) = \int_{V} [g(x, v)]_{+} dv, \qquad (3.3.5)$$

this function G(x) was given in [173]. Then (3.3.4) is equivalent to

$$\begin{cases} \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i) = 0, \\ G(x) = 0, \\ u_i > 0, \quad g(x, v^i) = 0, \quad (i = 1, \cdots, p) \\ \phi(x, v^i) = 0 \quad (i = 1, \cdots, p). \end{cases}$$
(3.3.6)

It is readily shown in Section 2.3 that G(x) is nonsmooth but semismooth.

Let

$$\mathbf{v} = (v^1, v^2, \cdots, v^p).$$

Define

$$F(x, u, \mathbf{v}) = \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i)$$
(3.3.7)

and

$$\mathbf{g}(x,\mathbf{v}) = \begin{pmatrix} g(x,v^1) \\ \vdots \\ g(x,v^p) \end{pmatrix}, \quad \hat{\phi}(x,\mathbf{v}) = \begin{pmatrix} \phi(x,v^1) \\ \vdots \\ \phi(x,v^p) \end{pmatrix}$$

By introducing an artificial variable  $s \in \Re$  and relaxing  $u_i > 0$  as  $u_i \ge 0$ , (3.3.6) can be written as the following system of nonsmooth equations with bounded constraints:

$$H(z) = 0,$$
  
 $u \ge 0, \quad s \ge 0,$ 
(3.3.8)

where  $z = (x, u, \mathbf{v}, s) \in \Re^n \times \Re^p \times \Re^{mp} \times \Re$ , and

$$H(z) = \begin{pmatrix} F(x, u, \mathbf{v}) \\ \mathbf{g}(x, \mathbf{v}) \\ G(x) + s \\ \hat{\phi}(x, \mathbf{v}) \end{pmatrix}.$$

Here, the introduction of s balance the number between equations and variables. In addition, it also can reduce the possible degeneration generated by the function G(x).

## 3.3.2 Smoothing Projected Newton-Type Algorithm

From the previous subsection, we see that the KKT system of the SIP problem (3.1.1) is equivalent to a system of nonsmooth equations with bounded constraints. This motivates us to obtain the stationary point of (3.1.1) by solving a system of constrained equations. But, since G(x) and  $\hat{\phi}(x, \mathbf{v})$  are not smooth, it is very difficult to compute their generalized Jacobian. We first introduce some smoothing techniques which deal with the corresponding functions.

Define  $\overline{G}: \Re \times \Re^n \to \Re$  by

$$\bar{G}(t,x) = \int_{V} \bar{g}(t,x,v)dv, \qquad (3.3.9)$$

where  $\bar{g}: \Re \times \Re^n \times \Re^m \to \Re$  is defined by

$$\bar{g}(t,x,v) = \frac{\sqrt{(g(x,v))^2 + 4t^2} + g(x,v)}{2}.$$
(3.3.10)

The function  $\bar{g}$  is the Chen-Harker-Kanzow-Smale smoothing function of  $[g(x, v)]_+$ , which was mentioned in Section 2.4. Of course, we also may choose other smoothing functions of  $[g(x, v)]_+$  as  $\bar{g}(t, x, v)$ . It is obvious that  $\bar{G}(0, x) = G(x)$ , and for any  $t \neq 0$ ,  $\bar{G}(t, x)$  is smooth with respect to variable x and

$$\nabla_x \bar{G}(t,x) = \int_V \nabla_x \bar{g}(t,x,v) dv. \qquad (3.3.11)$$

,

Define  $\varphi : \Re^4 \to \Re$  by

$$\varphi(t,c,d,w) = \frac{c + \sqrt{(c-w)^2 + 4t^2}}{2} + \frac{d - \sqrt{(d-w)^2 + 4t^2}}{2}$$

which is the Chen-Harker-Kanzow-Smale smoothing function for P(c, d, w). For  $a, b, v \in \Re^m$ , we define  $\bar{\phi} : \Re \times \Re^n \times \Re^m \to \Re^m$  by

$$(\bar{\phi}(t,x,v))_i = v_i - \varphi(t,a_i,b_i,v_i + (\nabla_v g(x,v))_i), \qquad (3.3.12)$$

where  $i = 1, \dots, m$ . It is clear that  $\overline{\phi}$  is smooth for  $t \neq 0$ .

From Theorem 3 in [130], Lemma 2.3 and Theorem 3.3 in [48], it is easy to prove the following results for  $\bar{\phi}$ .

**Proposition 3.3.1** The function  $\overline{\phi}$  defined in (3.3.12) has the following properties: (i) It is twice continuously differentiable for  $t \neq 0$ .

(ii) It is semismooth. Furthermore, if g is twice Lipschitz continuously differentiable, it is strongly semismooth.

(iii) There exists a constant C > 0 such that for any  $(x, v) \in \Re^{n+m}$  and  $t \in \Re$ ,

$$\left\|\bar{\phi}(t,x,v) - \phi(x,v)\right\| \le C|t|.$$

Denote  $w = (t, z) = (t, x, u, \mathbf{v}, s) \in \Re \times \Re^n \times \Re^p \times \Re^{mp} \times \Re$  and

$$\tilde{\phi}(t, x, \mathbf{v}) = \begin{pmatrix} \bar{\phi}(t, x, v^1) \\ \vdots \\ \bar{\phi}(t, x, v^p) \end{pmatrix}.$$

We define the following system of constrained equations:

$$\Phi(t, z) = 0 
u \ge 0, s \ge 0,$$
(3.3.13)

where

$$\Phi(t,z) = \begin{pmatrix} t \\ \bar{H}(t,z) \end{pmatrix}, \quad \bar{H}(t,z) = \begin{pmatrix} F(x,u,\mathbf{v}) \\ \mathbf{g}(x,\mathbf{v}) \\ \bar{G}(t,x) + s \\ \tilde{\phi}(t,x,\mathbf{v}) \end{pmatrix}.$$

It is obvious that if (t, z) is a solution of (3.3.13) then z is a solution to (3.3.8). By Theorems 2.4.1 and 2.4.2 and Proposition 3.3.1, we have the following result.

**Theorem 3.3.1**  $\overline{H}$  is a smoothing approximation function of H and it is semismooth at (0, z).

Motivated by the smoothing method in [130] for a system of unconstrained nonsmooth equations and the method in [168] for a system of constrained nonsmooth equations, in what follows, we present a smoothing projected Newton-type method for solving (3.3.8).

Let

$$W = \{ w = (t, x, u, \mathbf{v}, s) : u \ge 0, s \ge 0 \},\$$

and

$$Z = \{ (x, u, \mathbf{v}, s) \in \Re^n \times \Re^p \times \Re^{mp} \times \Re : u \ge 0, s \ge 0 \}.$$

Define a merit function of (3.3.13) by

$$\Psi(w) = \frac{1}{2} \|\Phi(w)\|^2.$$

Then solving (3.3.13) is equivalent to finding a global solution of the following minimization problem:

$$\min_{\substack{w \in \mathbb{N}, \\ w \in \mathbb{N},$$

And w is a stationary point of (3.3.14) if it satisfies

$$\|\bar{d}_G(1)\| = 0. \tag{3.3.15}$$

Here,

$$\bar{d}_G(1) = \Pi_W(w - \gamma \nabla \Psi(w)) - w = \begin{pmatrix} -\gamma \nabla_t \Psi(w) \\ \Pi_Z(z - \gamma \nabla_z \Psi(w)) - z \end{pmatrix}, \quad (3.3.16)$$

where  $\gamma > 0$  is a constant,  $\Pi_W(\cdot)$  is an orthogonal projection operator onto W.

Let  $\alpha \in (0,1)$  be a constant. For a sequence  $\{w^k\}_{k=0}^{\infty}$ , we define

$$\beta_0 = \beta(w^0) = \alpha \min\{1, \|\bar{d}_G^0(1)\|^2\},\$$

and

$$\beta_k = \beta(w^k) := \begin{cases} \beta_{k-1}, & \text{if } \alpha \min\{1, \|\vec{d}_G^k(1)\|^2\} > \beta_{k-1} \\ \alpha \min\{1, \|\vec{d}_G^k(1)\|^2\}, & \text{otherwise.} \end{cases}$$
(3.3.17)

Now we state our smoothing projected Newton-type algorithm for solving (3.3.14).

#### Algorithm 3.3.1 (Smoothing Projected Newton-Type Algorithm)

#### Step 0. (Initialization)

Choose constants  $\eta, \rho, \sigma \in (0, 1), p_1 > 0, p_2 > 2$  and  $\alpha > 0, \bar{t} > 0$  with  $\alpha \bar{t} < 1$ . 1. Let  $\bar{w} = (\bar{t}, 0, 0, 0, 0), t_0 = \bar{t}$  and  $w^0 = (t_0, x^0, u^0, \mathbf{v}^0, s^0)$  with  $u_i^0 \ge 0$   $(i = 1, \dots, p); s^0 \ge 0$ . Set k := 0.

Step 1. (Stop Test)

Let

$$\gamma_k = \min\left\{1, \frac{t_k}{|t_k + \nabla_t \bar{H}(w^k) | \bar{H}(w^k)|}, \frac{\eta \|\Phi(w^k)\|}{\|\nabla\Psi(w^k)\|}, \frac{\eta\Psi(w^k)}{\|\nabla\Psi(w^k)\|^2}\right\}, \qquad (3.3.18)$$

where  $\nabla_t \bar{H}(w^k)$  is the first row of  $\nabla \bar{H}(w^k)$ . Compute  $\bar{d}_G^k(1)$  by (3.3.16). If  $\|\bar{d}_G^k(1)\| = 0$ , stop. Otherwise, compute  $\beta_k$  by (3.3.17).

Step 2. (Compute Search Direction)

Compute  $d_G^k$  by

$$d_G^k = -\gamma_k \nabla \Psi(w^k) + \beta_k \bar{w}. \qquad (3.3.19)$$

Compute  $d_N^k$  by solving the following linear system:

$$\Phi(w^k) + \nabla^T \Phi(w^k) d_N^k = \beta_k \bar{w}.$$
(3.3.20)

If (3.3.20) has no solution or

$$-\nabla \Psi(w^k)^T d_N^k < p_1 \left\| d_N^k \right\|^{p_2},$$

then let  $d_N^k := d_G^k$ .

#### Step 3. (Line Search)

Let  $m_k$  be the smallest nonnegative integer m satisfying

$$\Psi(w^{k} + \bar{d}^{k}((\rho)^{m})) \leq \Psi(w^{k}) + \sigma \nabla \Psi(w^{k})^{T} \tilde{d}^{k}_{G}((\rho)^{m}), \qquad (3.3.21)$$

where for any  $\lambda \in [0, 1]$ ,

$$\bar{d}^k(\lambda) = \tau^*(\lambda)\tilde{d}^k_G(\lambda) + (1 - \tau^*(\lambda))\tilde{d}^k_N(\lambda).$$
(3.3.22)

Here

$$\tilde{d}_{G}^{k}(\lambda) := \Pi_{W}(w^{k} + \lambda d_{G}^{k}) - w^{k}, \quad \tilde{d}_{N}^{k}(\lambda) := \Pi_{W}(w^{k} + \lambda d_{N}^{k}) - w^{k}, \quad (3.3.23)$$

 $\tau^*(\lambda)$  is a solution of the following minimization problem:

$$\min_{\tau \in [0,1]} \frac{1}{2} \|\Phi(w^k) + \nabla^T \Phi(w^k) [\tau \tilde{d}_G^k(\lambda) + (1-\tau) \tilde{d}_N^k(\lambda)] \|^2$$
  
Let  $\lambda_k = (\rho)^{m_k}$  and  $w^{k+1} = w^k + \bar{d}^k(\lambda_k).$ 

Step 4. Set k := k + 1 and go to Step 1.

**Remark 3.3.1** (a) Algorithm 3.3.1 is an extension of the method for solving unconstrained nonsmooth equations presented in [130]. It is also a smoothing version of the algorithm proposed in [168]. In [168], it is required that the merit function  $\Psi$  must be smooth. In this section, we do not need this requirement. (b) By using a similar way to the proof of Lemma 3.1 [168], we can obtain the following result about  $\tau^*(\lambda)$ .

$$\tau^*(\lambda) = \max\{0, \min\{1, \tau(\lambda)\}\},$$
(3.3.24)

where  $\tau(\lambda)$  is defined as

$$\tau(\lambda) = \begin{cases} 0, & \text{if } \nabla^T \Phi(w^k) [\tilde{d}_G^k(\lambda) - \tilde{d}_N^k(\lambda)] = 0, \\ -\frac{[\Phi(w^k) + \nabla^T \Phi(w^k) \tilde{d}_N^k(\lambda)]^T \nabla^T \Phi(w^k) [\tilde{d}_G^k(\lambda) - \tilde{d}_N^k(\lambda)]}{\|\nabla^T \Phi(w^k) [\tilde{d}_G^k(\lambda) - \tilde{d}_N^k(\lambda)]\|^2}, & \text{otherwise.} \end{cases}$$

The following projection properties are used in our analysis (see [16]).

**Lemma 3.3.1** The projection operator  $\Pi_W(\cdot)$  with any convex set  $W \subset \Re^n$  satisfies (i) For any  $w \in W$ ,

$$[\Pi_W(w') - w']^T [\Pi_W(w') - w] \le 0 \quad for \ all \ w' \in \Re^n.$$

(ii)

$$|\Pi_W(w') - \Pi_W(w'')|| \le ||w' - w''|| \quad for \ all \ w', w'' \in \Re^n.$$

(iii) Given  $w, d \in \Re^n$ , the function  $\zeta$  defined by

$$\zeta(\lambda) = \|\Pi_W(w + \lambda d) - w\|/\lambda, \quad \lambda > 0$$

is non-increasing.

From the definition of  $\beta_k$ , the following proposition is obvious.

**Proposition 3.3.2**  $\{\beta_k\}$  defined in (3.3.17) has the following properties:

(i)  $\{\beta_k\}$  is a non-increasing sequence.

(ii) For all k,  $\beta_k$  satisfies

$$\beta_k \le \alpha \min\{1, \|\vec{d}_G^k(1)\|^2\}.$$

**Proposition 3.3.3** Suppose that  $w^k = (t^k, z^k) \in W$  with  $t^k > 0$  is not a stationary point of (3.3.14). Then for any  $\lambda \in (0, 1]$ , it holds that

$$\nabla \Psi(w^k)^T \tilde{d}_G^k(\lambda) \le -\frac{\lambda}{\gamma_k} (1 - \alpha \bar{t}) \| \bar{d}_G^k(1) \|^2 < 0.$$
(3.3.25)

**Proof.** In this proof, for simplicity, we drop the superscript k. For any  $w = (t, z) \in W$  with t > 0, suppose that w is not a stationary point of (3.3.14). Then

$$\nabla\Psi(w) = \nabla\Phi(w)\Phi(w) = \begin{pmatrix} t + \nabla_t \bar{H}(w)\bar{H}(w) \\ \nabla_z \bar{H}(w)\bar{H}(w) \end{pmatrix} \equiv \begin{pmatrix} \nabla_t\Psi(w) \\ \nabla_z\Psi(w) \end{pmatrix},$$

where  $\nabla_t \bar{H}(w)$  is the first row of  $\nabla \bar{H}(w)$  and  $\nabla_z \bar{H}(w)$  is the submatrix of  $\nabla \bar{H}(w)$ obtained by just removing the first row of  $\nabla \bar{H}(w)$ . Obviously,  $\tilde{d}_G(\lambda)$  can be written as

$$\tilde{d}_G(\lambda) \equiv \begin{pmatrix} (\tilde{d}_G(\lambda))_t \\ (\tilde{d}_G(\lambda))_z \end{pmatrix} = \begin{pmatrix} -\lambda\gamma(t + \nabla_t \bar{H}(w)\bar{H}(w)) + \lambda\beta(w)\bar{t} \\ \Pi_Z(z - \lambda\gamma\nabla_z\Psi(w)) - z \end{pmatrix}.$$

Then we have

$$(t + \nabla_t \bar{H}(w)\bar{H}(w))^T [-\lambda\gamma(t + \nabla_t \bar{H}(w)\bar{H}(w)) + \lambda\beta(w)\bar{t}]$$

$$= -\lambda\gamma \left\| t + \nabla_t \bar{H}(w)\bar{H}(w) \right\|^2 + \lambda(t + \nabla_t \bar{H}(w)\bar{H}(w))^T\beta(w)\bar{t}$$

$$\leq -\frac{\lambda}{\gamma} \| - \gamma\nabla_t\Psi(w) \|^2 + \frac{\lambda}{\gamma} \| - \gamma\nabla_t\Psi(w) \|\beta(w)\bar{t}$$

$$\leq -\frac{\lambda}{\gamma} \| - \gamma\nabla_t\Psi(w) \|^2 + \frac{\lambda}{\gamma} \| - \gamma\nabla_t\Psi(w) \|(\alpha\bar{t}\,)\|\bar{d}_G(1)\|$$

$$\leq -\frac{\lambda}{\gamma} \| - \gamma\nabla_t\Psi(w) \|^2 + \alpha\bar{t}\frac{\lambda}{\gamma} \|\bar{d}_G(1)\|^2,$$
(3.3.26)

where the second inequality comes from Proposition 3.3.2 (ii) and the fact that  $\beta(w) \leq \alpha \|\bar{d}_G(1)\|$ , the last inequality is due to  $\|-\gamma \nabla_t \Psi(w)\| \leq \|\bar{d}_G(1)\|$  (see (3.3.16)). Thus,

$$\begin{aligned} \nabla_{z}\Psi(w)^{T}[\Pi_{Z}(z-\lambda\gamma\nabla_{z}\Psi(w))-z] \\ &= -\frac{1}{\lambda\gamma}[z-\lambda\gamma\nabla_{z}\Psi(w)-z]^{T}[\Pi_{Z}(z-\lambda\gamma\nabla_{z}\Psi(w))-z] \\ &= \frac{1}{\lambda\gamma}[\Pi_{Z}(z-\lambda\gamma\nabla_{z}\Psi(w))-(z-\lambda\gamma\nabla_{z}\Psi(w))]^{T}[\Pi_{Z}(z-\lambda\gamma\nabla_{z}\Psi(w))-z] \\ &\quad -\frac{1}{\lambda\gamma}\|\Pi_{Z}(z-\lambda\gamma\nabla_{z}\Psi(w))-z\|^{2} \\ &\leq -\frac{1}{\lambda\gamma}\|\Pi_{Z}(z-\lambda\gamma\nabla_{z}\Psi(w))-z\|^{2} \\ &\leq -\frac{\lambda}{\gamma}\|\Pi_{Z}(z-\gamma\nabla_{z}\Psi(w))-z\|^{2}, \end{aligned}$$
(3.3.27)

where the first and second inequalities come from Lemma 3.3.1 (i) and (iii), respectively. It follows from (3.3.26) and (3.3.27) that

$$\nabla \Psi(w)^T \tilde{d}_G(\lambda) = (t + \nabla_t \bar{H}(w) \bar{H}(w))^T [-\lambda \gamma (t + \nabla_t \bar{H}(w) \bar{H}(w)) + \lambda \beta(w) \bar{t}] + \nabla_z \Psi(w)^T [\Pi_Z(z - \lambda \gamma \nabla_z \Psi(w)) - z] \leq -\frac{\lambda}{\gamma} \left[ \| - \gamma \nabla_t \Psi(w) \|^2 + \| \Pi_z(z - \gamma \nabla_z \Psi(w)) - z \|^2 \right] + \alpha \bar{t} \frac{\lambda}{\gamma} \| \bar{d}_G(1) \|^2 = -\frac{\lambda}{\gamma} (1 - \alpha \bar{t}) \| \bar{d}_G(1) \|^2 < 0.$$

The proof is complete.  $\blacksquare$ 

Now we have the following conclusion which shows that Algorithm 3.3.1 is welldefined.

**Theorem 3.3.2** Suppose that  $w^k = (t^k, z^k) \in W$  with  $t^k > 0$  is not a stationary point of (3.3.14). Then there exists a constant  $\lambda' \in (0, 1]$  such that for any  $\lambda \in (0, \lambda']$ ,  $\bar{d}^k(\lambda)$ is a descent direction of  $\Psi(w^k)$  at  $w^k$  and

$$\Psi(w^k + \bar{d}^k(\lambda)) \le \Psi(w^k) + \sigma \nabla \Psi(w^k)^T \tilde{d}^k_G(\lambda).$$
(3.3.28)

**Proof.** By using Proposition 3.3.3, the conclusion can be proved in a similar way to the proof of Theorem 3.1 in [168], so we omit it. ■

### 3.3.3 Convergence Analysis

In this subsection we analyze the global and local convergence of Algorithm 3.3.1. The following proposition is a key result which shows that Algorithm 3.3.1 can keep  $t^k > 0$  at each iteration.

**Proposition 3.3.4** For each  $k, k = 0, 1, \dots, w^k = (t^k, z^k)$  satisfies

$$t^k \ge \beta_k \bar{t}.\tag{3.3.29}$$

Furthermore, if  $w^k$  is not a stationary point of (3.3.14), then

$$t^k > 0.$$
 (3.3.30)

**Proof.** We show inductively that (3.3.29) holds. From the choices of  $t^0$  and  $\beta_0$  in Algorithm 3.3.1, (3.3.29) holds trivially for k = 0. Let us assume that (3.3.29) holds for some k = l. Now, we prove that (3.3.29) holds for k = l + 1 as well. We denote

$$\vec{d}^l(\lambda_l) = \tau^*(\lambda_l) \vec{d}^l_G(\lambda_l) + (1 - \tau^*(\lambda_l)) \vec{d}^l_N(\lambda_l) = \begin{pmatrix} (\vec{d}^l(\lambda_l))_t \\ (\vec{d}^l(\lambda_l))_z \end{pmatrix},$$

where  $\lambda_l$  is the accepted step-length at *l*-th iteration. It follows from Algorithm 3.3.1 that

$$\begin{aligned} (\bar{d}^{l}(\lambda_{l}))_{t} &= \tau^{*}(\lambda_{l})\lambda_{l}[-\gamma_{l}(t^{l}+\nabla_{t}\bar{H}(w)\bar{H}(w))+\beta(w^{l})\bar{t} ] \\ &+(1-\tau^{*}(\lambda_{l}))\lambda_{l}[-t^{l}+\beta(w^{l})\bar{t} ] \\ &= -\lambda_{l}\gamma_{l}\tau^{*}(\lambda_{l})(t^{l}+\nabla_{t}\bar{H}(w)\bar{H}(w))-(1-\tau^{*}(\lambda_{l}))\lambda_{l}t^{l}+\lambda_{l}\beta(w^{l})\bar{t} \\ &\geq -\lambda_{l}\tau^{*}(\lambda_{l})t^{l}-(1-\tau^{*}(\lambda_{l}))\lambda_{l}t^{l}+\lambda_{l}\beta(w^{l})\bar{t} \\ &= -\lambda_{l}t^{l}+\lambda_{l}\beta(w^{l})\bar{t}, \end{aligned}$$

where the inequality comes from the definition of  $\gamma_l$  (see (3.3.18)). Then we have

$$t^{l+1} - \beta(w^{l+1})\overline{t} = t^l + (\overline{d}^l(\lambda_l))_t - \beta(w^{l+1})\overline{t}$$
  

$$\geq (1 - \lambda_l)t^l + \lambda_l\beta(w^l)\overline{t} - \beta(w^{l+1})\overline{t}$$
  

$$\geq (1 - \lambda_l)t^l + \lambda_l\beta(w^l)\overline{t} - \beta(w^l)\overline{t}$$
  

$$= (1 - \lambda_l)t^l - (1 - \lambda_l)\beta(w^l)\overline{t} \geq 0,$$
(3.3.31)

where the second inequality is due to the monotonicity property of  $\beta(w^l)$  in Proposition 3.3.2, and the last inequality comes from that  $t^l \geq \beta(w^l)\bar{t}$ . By induction, (3.3.29) holds for any nonnegative integer k. Furthermore, from (3.3.29) and that  $w^k$  is not a stationary point of (3.3.14), (3.3.30) holds. We complete the proof.

**Theorem 3.3.3** Let  $\{w^k\} \subset W$  be a sequence generated by Algorithm 3.3.1. Then any accumulation point of  $\{w^k\}$  is a stationary point of (3.3.14).

**Proof.** Proposition 3.3.4 shows that if our algorithm does not stop at a stationary point of (3.3.14), then  $t^k > 0$  for any k. This means that  $\Phi$  and  $\Psi$  are continuously differentiable at  $w^k$ . Hence, by using a similar way to the proof of Theorem 4.1 in [168], we can prove that the theorem holds. Here, we omit the detailed proof.

In the rest of this subsection, we analyze the local convergence of Algorithm 3.3.1. We make the following standard assumption:

(C1) Let  $w^* = (t^*, z^*) = (0, z^*)$  be an accumulation point of the sequence  $\{w^k\}$  generated by Algorithm 3.3.1. Suppose  $\lim_{k \in K} w^k = w^*$  for some subset  $K \subset \{1, 2, \cdots\}$ ,  $w^*$  is a solution of the system of equations (3.3.13) and  $\Phi$  is BD-regular at  $w^*$ .

From the BD-regularity condition and semismoothness of function  $\Phi$ , we have the following lemma by using Propositions 1.1.1 and 1.1.2.

**Lemma 3.3.2** There exist positive constants  $\kappa$  and  $\epsilon$  such that for every  $w^k$  satisfying  $||w^k - w^*|| \le \epsilon$ , (i)  $\nabla \Phi(w^k)$  is nonsingular and satisfies

$$\|\nabla\Phi(w^k)\| \le \kappa.$$

(ii)

$$\|\Phi(w^k)\| = \sqrt{2}\Psi(w^k)^{\frac{1}{2}} = O(\|w^k - w^*\|).$$

**Lemma 3.3.3** For all  $k \in K$  sufficiently large,

(i)

$$\beta(w^k) = O(\Psi(w^k)) = O(\|w^k - w^*\|^2).$$

(ii) and for any  $\lambda \in (0, 1]$ 

$$w^{k} + \lambda d_{N}^{k} = (1 - \lambda)w^{k} + \lambda w^{*} + \lambda o(\Psi(w^{k})^{\frac{1}{2}}).$$
(3.3.32)

**Proof.** From the definition of  $\beta(w^k)$ , the choice of  $\gamma_k$ , the projection property and Lemma 3.3.2, for  $w^k$  sufficiently close to  $w^*$ ,

$$\beta(w^k) \le \alpha \|\bar{d}_G^k(1)\|^2 \le \alpha \gamma_k^2 \|\nabla \Psi(w^k)\|^2 \le \alpha \eta \Psi(w^k) = \frac{\alpha \eta}{2} \|\Phi(w^k)\|^2 = O(\|w^k - w^*\|^2).$$

This shows (i) holds. It follows from (i) and Lemma 3.3.2 that

$$\begin{split} w^{k} + \lambda d_{N}^{k} &= w^{k} + \lambda \left( \nabla^{T} \Phi(w^{k}) \right)^{-1} \left[ -\Phi(w^{k}) + \beta(w^{k}) \bar{w} \right] \\ &= w^{k} - \lambda \left( \nabla^{T} \Phi(w^{k}) \right)^{-1} \left[ \Phi(w^{k}) - \Phi(w^{*}) - \nabla^{T} \Phi(w^{k}) (w^{k} - w^{*}) \right] \\ &- \lambda (w^{k} - w^{*}) + \lambda \left( \nabla^{T} \Phi(w^{k}) \right)^{-1} \beta(w^{k}) \bar{w} \\ &= (1 - \lambda) w^{k} + \lambda w^{*} + \lambda o(\|w^{k} - w^{*}\|) + \lambda O(\Psi(w^{k})) \\ &= (1 - \lambda) w^{k} + \lambda w^{*} + \lambda o(\Psi(w^{k})^{\frac{1}{2}}), \end{split}$$

where the third equality is due to the semismoothness of  $\Phi$  and (i). (ii) is proved. The proof is complete  $\blacksquare$ 

**Lemma 3.3.4** For  $k \in K$  large enough,

$$\tilde{d}_{N}^{k}(\lambda) = -\lambda(w^{k} - w^{*}) + \lambda o(\Psi(w^{k})^{\frac{1}{2}})$$
(3.3.33)

and

$$\nabla \Psi(w^k)^T \tilde{d}_N^k(\lambda) \le -\mu \lambda \Psi(w^k), \qquad (3.3.34)$$

where  $\mu$  is any constant in (0, 2).

**Proof.** From Lemma 3.3.3 and the property of a projector, we obtain that

$$\begin{split} \tilde{d}_{N}^{k}(\lambda) &= \Pi_{W}(w^{k} + \lambda d_{N}^{k}) - w^{k} \\ &= \Pi_{W}[(1 - \lambda)w^{k} + \lambda w^{*} + \lambda o(\Psi(w^{k})^{\frac{1}{2}})] - w^{k} \\ &= \Pi_{W}[(1 - \lambda)w^{k} + \lambda w^{*}] - w^{k} \\ &+ \left\{ \Pi_{W}[(1 - \lambda)w^{k} + \lambda w^{*} + \lambda o(\Psi(w^{k})^{\frac{1}{2}})] - \Pi_{W}[(1 - \lambda)w^{k} + \lambda w^{*}] \right\} \\ &= -\lambda(w^{k} - w^{*}) + \lambda o(\Psi(w^{k})^{\frac{1}{2}}), \end{split}$$

where the last equality comes from  $(1 - \lambda)w^k + \lambda w^* \in W$  and the projection property (see Lemma 3.3.1 (ii)). It follows from (3.3.33) that

$$\begin{aligned} \nabla \Psi(w^k)^T \tilde{d}_N^k(\lambda) &= -\lambda \Phi(w^k)^T \nabla^T \Phi(w^k) (w^k - w^*) + \lambda o(\Psi(w^k)) \\ &= -2\lambda \Psi(w^k) + \lambda \Phi(w^k)^T [\Phi(w^k) - \Phi(w^*) - \nabla^T \Phi(w^k) (w^k - w^*)] \\ &\quad + \lambda o(\Psi(w^k)) \\ &\leq -\mu \lambda \Psi(w^k), \end{aligned}$$

where the last inequality comes from the semismoothness of  $\Phi$  and Lemma 3.3.2. We complete the proof.

**Lemma 3.3.5** We have that for  $k \in K$  large enough, (i)

$$\tau^*(\lambda)_k \le o(1),$$
 (3.3.35)

where  $\tau^*(\lambda)_k$  is defined as in (3.3.24). (ii)

$$\bar{d}^{k}(\lambda) = -\lambda(w^{k} - w^{*}) + \lambda o(\Psi(w^{k})^{\frac{1}{2}}).$$
(3.3.36)

(iii)

$$\nabla \Psi(w^k)^T \bar{d}^k(\lambda) = -2\lambda \Psi(w^k) + \lambda o(\Psi(w^k)).$$
(3.3.37)

**Proof.** By using Lemma 3.3.4, this lemma can be proved in a similar way to the proof of Theorem 3.2 in [168]. We omit the detailed proof. ■

Now we prove that the convergence rate of Algorithm 3.3.1 is locally superlinear under the BD-regularity condition.

**Theorem 3.3.4** Suppose that  $\{w^k\}$  is a sequence generalized by Algorithm 3.3.1 and  $w^*$  is a point satisfying (C1). Then the whole sequence  $\{w^k\}$  converges to  $w^*$  superlinearly.

**Proof.** From Lemma 3.3.5, we have that for sufficiently large  $k \in K$ ,

$$\|w^{k} + \bar{d}^{k}(1) - w^{*}\| = o(\Psi(w^{k})^{\frac{1}{2}}) = o(\|\Phi(w^{k})\|) = o(\|w^{k} - w^{*}\|), \qquad (3.3.38)$$

and

$$\Psi(w^{k} + \vec{d}^{k}(1)) = \frac{1}{2} \|\Phi(w^{k} + \vec{d}^{k}(1))\|^{2}$$
  

$$= \frac{1}{2} \|\Phi(w^{k} + \vec{d}^{k}(1)) - \Phi(w^{*})\|^{2}$$
  

$$= O(\|w^{k} + \vec{d}^{k}(1) - w^{*}\|^{2})$$
  

$$= o(\Psi(w^{k})),$$
(3.3.39)

where the last equality is due to (3.3.38). Thus,

$$\begin{aligned}
-\nabla\Psi(w^{k})^{T}\tilde{d}_{G}^{k}(1) &\leq \|\nabla\Psi(w^{k})\|\|\tilde{d}_{G}^{k}(1)\| \\
&= \|\nabla\Psi(w^{k})\|\|\Pi_{W}(w^{k} - \gamma_{k}\nabla\Psi(w^{k}) + \beta(w^{k})\bar{w}) - w^{k}\| \\
&\leq \|\nabla\Psi(w^{k})\|[\|\gamma_{k}\nabla\Psi(w^{k})\| + O(\Psi(w^{k}))] \\
&\leq \eta\Psi(w^{k}) + o(\Psi(w^{k})),
\end{aligned} \tag{3.3.40}$$

where the second inequality is due to the property of  $\beta(w^k)$  and the projection property, and the last inequality comes from the choice of  $\gamma_k$ . It follows (3.3.39) and (3.3.40) that

$$\Psi(w^{k}) + \sigma \nabla \Psi(w^{k})^{T} \tilde{d}_{G}^{k}(1) \geq (1 - \sigma \eta) \Psi(w^{k}) + o(\Psi(w^{k}))$$
  
$$\geq o(\Psi(w^{k}))$$
  
$$= \Psi(w^{k} + \bar{d}^{k}(1)), \qquad (3.3.41)$$

which implies

$$w^{k+1} = w^k + \bar{d}^k(1)$$

for k sufficiently large. Moreover, from (3.3.38) we conclude that  $w^k$  converges to  $w^*$  superlinearly. We complete the proof.

#### 3.3.4 Preliminary Numerical Examples

In this subsection, we report our preliminary numerical test results. We implemented Algorithm 3.3.1 in MATLAB and the numerical experiments were done by using a Pentium III 733MHz computer with 256 MB of RAM. We tested 12 problems which are called Problems 3.3.1-3.3.12. Problems 3.3.1-3.3.3 and 3.3.7 are from [181]. Problem 3.3.4 comes from [174] with a revised region. Problem 3.3.5 is a problem modified

from [187], and Problem 3.3.6 is from [26]. Problems 3.3.8-3.3.12 are some problems in which the dimension of the parameter v is 2.

Throughout the computational experiments, we use  $\|\bar{d}_G^k(1)\| \leq 10^{-6}$  as the stopping criterion for Algorithm 3.3.1. The values of  $\bar{G}(t,x)$  and  $\nabla \bar{G}(t,x)$  were computed by using the function quad in MATLAB when V is an interval in  $\Re$  and the function dblquad when V is a box set in  $\Re^2$ . The parameters used in the algorithm are specified as follows:

$$\eta = 0.9, \ \rho = 0.5, \ \sigma = 0.001, \ \alpha = 0.5, \ \bar{t} = 0.9, \ p_1 = 1.0e - 10, \ p_2 = 2.1.$$

The starting point  $u^0$  and  $y^0$  for all problems are set  $t^0 = \bar{t}$ ,  $u^0 = 0.05e$ ,  $y^0 = 0.5$ , where e is the vector of ones. We compared Algorithm 3.3.1 with fseminf. For the solver fseminf, we use all the default values.

#### Problem 3.3.1

$$f(x) = 1.21\exp(x_1) + \exp(x_2), \quad g(x,v) = v - \exp(x_1 + x_2),$$
  
 $V = [-10, 1], p = 1, (x_0, v_0) = (1, 1, 1).$ 

#### Problem 3.3.2

$$f(x) = x_1^2 + x_2^2 + x_3^2, \quad g(x, v) = x_1 + x_2 \exp(x_3 v) + \exp(2v) - 2\sin(4v),$$
$$V = [0, 1], p = 1, (x_0, v_0) = (1, 1, 1, 1).$$

Problem 3.3.3

$$f(x) = \frac{1}{3}x_1^2 + \frac{1}{2}x_1 + x_2^2, \quad g(x,v) = (1 - x_1^2v^2)^2 - x_1v^2 - x_2^2 + x_2,$$
$$V = [-1,1], p = 1, (x_0, v_0) = (-1, -1, 1).$$

Problem 3.3.4

$$f(x) = x_1^2 + (x_2 - 3)^2, \quad g(x, v) = x_2 - 2 + x_1 \sin(v/x_2 - 0.5),$$
  
 $V = [0, 10], p = 1, (x_0, v_0) = (1, -1, 1).$ 

# Problem 3.3.5

$$f(x) = \frac{1}{2}x^{T}x, \quad g(x,v) = 3 + 4.5\sin(4.7\pi(v-1.23)/8) - \sum_{i=1}^{n} x_{i}v^{i-1},$$
$$V = [0,1], n = 10, p = 1, (x_{0}, v_{0}) = (0, 0, \dots, 0, 1).$$

# Problem 3.3.6

$$f(x) = (x_1 - 2x_2 + 5x_2^2 - x_2^3 - 13)^2 + (x_1 - 14x_2 + x_2^2 + x_2^3 - 29)^2,$$
  
$$g(x, v) = x_1^2 + 2x_2v^2 + \exp(x_1 + x_2) - \exp(v), V = [0, 1], p = 1, (x_0, v_0) = (1, -1, 1).$$

# Problem 3.3.7

$$f(x) = x_1^2 + x_2^2 + x_3^3,$$
  
$$g(x,v) = x_1(v_1 + v_2^2 + 1) + x_2(v_1v_2 - v_2^2) + x_3(v_1v_2 + v_2^2 + v_2) + 1,$$
  
$$V = [0,1] \times [0,1], p = 1, (x_0, v_0) = (1, 1, 1, 1, 0).$$

## Problem 3.3.8

$$f(x) = x_1^2 + x_2^2 + x_3^2, \quad g(x, v) = x_1 + x_2 \exp(x_3 v_1) + \exp(2v_2) - 2\sin(4v_1),$$
$$V = [0, 1] \times [0, 1], p = 2, (x_0, v_0) = (-1, -1, -1, 0, 1, 1, 0).$$

# Problem 3.3.9

$$f(x) = x_1^2 + x_2^2 + x_3^2, \quad g(x, v) = x_1 + x_2 \exp(x_3 v_1) - \exp(2x_1 v_2) + \sin(4v_1),$$
$$V = [0, 1] \times [0, 1], p = 2, (x_0, v_0) = (-0.2, -0.2, -0.2, 0, 1, 1, 0).$$

#### Problem 3.3.10

$$f(x) = \frac{1}{3}x_1^2 + \frac{1}{2}x_1 + x_2^2, \quad g(x,v) = (1 - x_1^2v_1^2)^2 - x_1v_2^2 - x_2^2 + x_2.$$
$$V = [0,2] \times [0,2], p = 2, (x_0,v_0) = (-0.2, -0.2, 1, 0, 0, 1).$$

### Problem 3.3.11

$$f(x) = \frac{1}{2}(x_1^2 + x_2^2 + x_3^2 + x_4^2), \quad g(x, v) = \sin(v_1 v_2) - x_1 - x_2 v_1 - x_3 v_2 - x_4 v_1 v_2,$$

$$V = [0,1] \times [0,1], p = 1, (x_0, v_0) = (-0.5, -0.5, -0.5, -0.5, 0, 1).$$

Problem 3.3.12

$$f(x) = \frac{1}{2}(x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 + x_6^2),$$
  

$$g(x, v) = \exp(v_1^2 + v_2^2) - (x_1 + x_2v_1 + x_3v_2 + x_4v_1^2 + x_5v_1v_2 + x_6v_2^2),$$
  

$$V = [0, 1] \times [0, 1], p = 1, (x_0, v_0) = (-2, -2, -2, -2, -2, -2, 1, 1).$$

In all above test problems, the values of p are estimated by using the following adaptive strategy. First, we let p = 1 and use Algorithm 3.3.1 to solve a test problem. If this test problem can be solved within 30 iterations, then we let p = 1 be the number of attainers at the solution. Otherwise, we let p = 2 and use Algorithm 3.3.1 to solve this test problem again. If this test problem can be solved within 30 iterations, then we let p = 2 be the number of attainers. If this fails again, then we let p = 3 and then do the above procedure until we find a number p ( $p \le n$ ) which is the estimated number of attainers. It is interesting that we get p = 1 for 9 of 12 test problems and p = 2 for other three test problems by the above method.

The test results are summarized in Tables 3.2 and 3.3. In Table 3.2,  $\bar{d}_G^k(1)$  is the value of the function  $\bar{d}_G(1)$  defined in (3.3.16) at the k-th iteration. In Table 3.3, **n.it** represents the number of the total iterations; **cpu** is the total cost time in seconds for solving the SIP problem;  $\Psi(w^k)$ ,  $f(x^k)$  and  $G(x^k)$  denote the values of the merit function  $\Psi(w)$  of (3.3.13), the objective function in the SIP problem and the function G(x) of (3.3.5) at the final iteration, respectively.

The results reported in Tables 3.2 and 3.3 show that Algorithm 3.3.1 performs well for these test problems. From Table 3.2, we can see that Algorithm 3.3.1 indeed has superlinear convergence property. From Table 3.3, we can see that Algorithm 3.3.1 uses less CPU time than **fseminf** for 9 test problems and **fseminf** uses less CPU time than Algorithm 3.3.1 for other 3 test problems. Moreover, it appears from Table 3.3 that Algorithm 3.3.1 indeed can ensure the feasibility of the test problems.

In addition, we notice that for Problems 3.3.1–3.3.7, 3.3.11 and 3.3.12, when  $p \ge 2$ , these test problems cannot be solved by Algorithm 3.3.1 within 30 iterations. For Problems 3.3.8–3.3.10, when p = 1, these three test problems cannot be solved by

Problem	k	$\bar{d}_G^k(1)$	Problem	k	$\bar{d}_G^k(1)$	Problem	k	$\bar{d}_G^k(1)$
3.3.1	4	0.0053	3.3.2	7	0.0141	3.3.3	5	0.0036
	5	2.5166e-5		8	4.9812e-4		6	9.671e-5
	6	3.6233e-10		9	4.5572e-7		7	3.8475e-9
3.3.4	7	2.0688e-6	3.3.5	2	0.0027	3.3.6	3	6.3414e-4
	8	1.2389e-6		3	3.5370e-5		4	3.2046e-5
	9	4.2285e-7		4	1.3002e-10		5	1.2032e-8
3.3.7	5	0.0228	3.3.8	5	3.7704e-5	3.3.9	8	0.0075
	6	7.3722e-4		6	1.0023e-6		9	4.0859e-5
	7	4.1271e-7		7	2.7485e-10		10	7.8563e-8
3.3.10	7	9.0029e-4	3.3.11	6	0.0086	3.3.12	3	9.5606e-4
	8	2.2744e-6		7	0.0016		4	1.3685e-7
	9	1.6448e-9		8	7.8146e-7		5	2.4486e-15

Table 3.2: The last three iterates generated by Algorithm 3.3.1

Table 3.3: Test results for Algorithm 3.3.1 and fseminf

	Algorithm 3.3.1					fseminf				
Problem	n.it	cpu	$\Psi(w^k)$	$f(x^k)$	$G(x^k)$	n.it	cpu	$f(x^k)$	$G(x^k)$	
3.3.1	6	0.09	9.161e-19	2.2	0	16	0.53	2.1989	4.282e-7	
3.3.2	9	0.17	7.436e-11	5.3347	0	23	0.42	5.3307	1.114e-5	
3.3.3	7	0.13	2.346e-14	0.1945	0	4	0.12	0.1945	3.641e-12	
3.3.4	9	0.31	7.195e-9	1	4.799e-6	10	0.66	1	9.070e-9	
3.3.5	4	0.30	4.034e-19	0.0657	0	2	0.42	0.0656	1.746e-7	
3.3.6	5	0.11	2.051e-11	97.1589	1.304e-9	8	0.17	97.1589	7.455e-14	
3.3.7	7	4.81	1.051e-13	1	0	7	4.91	1	0	
3.3.8	7	14.22	1.299e-16	27.4166	0	6	5.33	27.3065	6.192e-8	
3.3.9	10	2.72	1.368e-13	0	0	3	2.98	3.152e-5	0	
3.3.10	9	1.48	2.664e-13	0.382	5.839e-7	15	2.66	0.382	0	
3.3.11	8	2.28	1.305e-11	0.0885	0	1	2.00	0.0885	1.808e-21	
3.3.12	5	0.73	3.773e-29	4.5498	0	1	3.38	4.5498	0	

Algorithm 3.3.1 within 30 iterations. This means that it is important to choose a suitable number p when we use Algorithm 3.3.1 to solve the SIP problem. When the size of the SIP problem and the number p are large, the above method to determine the number p may be expensive in computation. As future work, we will work on how

to find a good way to determine a suitable number p in the KKT system of the SIP problem.

# 3.4 A Smoothing Newton-Type Algorithm

In previous section, we presented a smoothing projected Newton-type algorithm for solving the SIP problem (3.1.1), which has global and local superlinear convergence property. However, from Theorem 3.3.3, we see that each accumulation point of  $\{w^k\}$ generated by Algorithm 3.3.1 is only a stationary point of (3.3.14) but may not be a stationary point of the original SIP problem (3.1.1). In this section, we present a smoothing Newton-type algorithm for solving the SIP problem (3.1.1), which overcomes the drawback stated above.

### 3.4.1 A Semismooth Equation Reformulation of KKT System

We suppose that, in addition to the assumptions on functions f and g in Section 3.3, the V in (3.1.1) has the following form

$$V = \{ v \in \mathfrak{R}^m : c(v) \le 0 \},\$$

where  $c : \Re^m \to \Re^q$  are twice continuously differentiable functions. As mentioned in the introduction of this chapter, under the assumption stated above, the KKT system of the SIP problem (3.1.1) can be rewritten as follows

$$\begin{cases} \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i) = 0, \\ g(x, v) \le 0, \quad \forall \ v \in V, \\ u_i > 0, \quad g(x, v^i) = 0, \ i = 1, \cdots, p, \\ -u_i \nabla_v g(x, v^i) + \sum_{j=1}^{q} w_j^i \nabla c_j(v^i) = 0, \\ w_j^i \ge 0, \ c_j(v^i) \le 0, \\ w_j^i \ c_j(v^i) = 0, \ i = 1, \cdots, p, \ j = 1, \cdots, q. \end{cases}$$
(3.4.1)

In order to reformulate the system (3.4.1) as a system of semismooth equations, we recall the definition of NCP functions.

A function  $\phi : \Re^2 \to \Re$  is called an NCP function [40] if  $\phi(a, b)=0$  if and only if  $a \ge 0, b \ge 0$  and ab = 0. Two well-known NCP functions are the minimum function

$$\phi_{\min}(a,b) = \min\{a,b\}$$

and the Fischer-Burmeister function (see (1.1.7)). Both the minimum function and the Fischer-Burmeister function are not smooth, but they are semismooth.

By the use of the Fischer-Burmeister function  $\phi_{FB}$  defined by (1.1.7) and G defined by (3.3.5), we may reformulate (3.4.1) as a system of semismooth equations:

$$H(s,z) = 0. (3.4.2)$$

Here

$$H(s,z) = \left(\begin{array}{c} G(x) + s \\ P(z) \end{array}\right)$$

and

$$P(z) = \begin{pmatrix} \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i) \\ \phi_{FB}(u_1, -g(x, v^1)) \\ \vdots \\ \phi_{FB}(u_p, -g(x, v^1)) \\ -u_1 \nabla_v g(x, v^1) + \sum_{j=1}^{q} w_j^1 \nabla c_j(v^1) \\ \vdots \\ -u_p \nabla_v g(x, v^p) + \sum_{j=1}^{q} w_j^p \nabla c_j(v^p) \\ \phi_{FB}(w_1^1, -c_1(v^1)) \\ \vdots \\ \phi_{FB}(w_q^1, -c_q(v^1)) \\ \vdots \\ \phi_{FB}(w_q^1, -c_1(v^p)) \\ \vdots \\ \phi_{FB}(w_q^p, -c_q(v^p)) \end{pmatrix},$$

where  $(s, z) = (s, x, u, \mathbf{v}, \mathbf{w}) \in \Re^{1+n+p(m+q+1)}$ ,  $\mathbf{v} = (v^1, \dots, v^p) \in \Re^{pm}$ ,  $\mathbf{w} = (w^1, \dots, w^p) \in \Re^{pq}$ . Here,  $s \in \Re$  is an auxiliary variable which ensure the numbers of the variables in the system equal to the numbers of the equations.

Nonlinear equation (3.4.2) transforms the system (3.4.1) into a semismooth equation of dimension 1 + n + (m + q + 1)p. If there is an 1 + n + (m + q + 1)p dimensional vector satisfying (3.4.2) and s = 0, we may then drop the part indexed by *i* where  $u_i = 0$ . In this case, we get a solution of (3.4.1) which obviously satisfies (3.4.2). Hence, (3.4.1) and (3.4.2) are equivalent in this sense. The discussion above shows that we may obtain the solution *z* of (3.4.1) by solving the system (3.4.2). However, the nonsmoothness of *G* and  $\phi_{FB}$  in (3.4.2) results in the difficulty of the implementation of the algorithm for solving (3.4.2). To overcome this difficulty, we will develop a smoothing Newton-type algorithm for solving (3.4.2) in next subsection.

### 3.4.2 Smoothing Newton-Type Algorithm

We first recall the smoothing function of  $\phi_{FB}$ . For a smoothing parameter  $t \in \Re$ , it is well known that the smoothing approximation function of  $\phi_{FB}$  can be defined by

$$\bar{\phi}_{FB}(t,a,b) = \sqrt{a^2 + b^2 + t^2} - a - b.$$

Let  $w \in \Re$  and  $h : \Re^m \to \Re$  be continuously differentiable. Denote  $\tilde{\phi}_{FB} : \Re \times \Re \times \Re^m \to \Re$  as follows

$$\tilde{\phi}_{FB}(t,w,v) = \bar{\phi}_{FB}(t,w,h(v)). \tag{3.4.3}$$

From Theorem 3 in [130], Lemma 2.3 and Theorem 3.3 in [48] and Proposition 6.1 in [135], it is easy to prove the following results for  $\tilde{\phi}_{FB}$ .

**Proposition 3.4.1** The function  $\tilde{\phi}_{FB}$  defined in (3.4.3) has the following properties:

- (i) It is twice continuously differentiable for any  $t \neq 0$ .
- (ii) There exists a constant C > 0 such that for any  $(w, v) \in \Re \times \Re^m$

$$\left\|\tilde{\phi}_{FB}(t, w, v) - \phi_{FB}(w, h(v))\right\| \le C|t|.$$

(iii) The function  $\tilde{\phi}_{FB}$  is semismooth with respect to (t, w, v).

Denote  $y = (t, s, z) = (t, s, x, u, \mathbf{v}, \mathbf{w}) \in \Re^{2+n+p(m+q+1)}$ . By using the smoothing approximation function  $\bar{\phi}_{FB}$  of  $\phi_{FB}$  and the smoothing approximation function  $\bar{G}$  defined

by (3.3.9) of G, we introduce the following system of equations:

$$\Phi(y) = 0, \tag{3.4.4}$$

where

$$\Phi(y) = \begin{pmatrix} t \\ \bar{G}(t,x) + s \\ \bar{P}(t,z) \end{pmatrix}$$

and

$$\bar{P}(t,z) = \begin{pmatrix} \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i) \\ \bar{\phi}_{FB}(t, u_1, -g(x, v^1)) \\ \vdots \\ \bar{\phi}_{FB}(t, u_p, -g(x, v^p)) \\ -u_1 \nabla_v g(x, v^1) + \sum_{j=1}^{q} w_j^1 \nabla c_j(v^1) \\ \vdots \\ -u_p \nabla_v g(x, v^p) + \sum_{j=1}^{q} w_j^p \nabla c_j(v^p) \\ \bar{\phi}_{FB}(t, w_1^1, -c_1(v^1)) \\ \vdots \\ \bar{\phi}_{FB}(t, w_q^1, -c_q(v^1)) \\ \vdots \\ \bar{\phi}_{FB}(t, w_1^p, -c_1(v^p)) \\ \vdots \\ \bar{\phi}_{FB}(t, w_q^p, -c_q(v^p)) \end{pmatrix}.$$

It is obvious that if y = (t, s, z) with  $s \ge 0$  is a solution of (3.4.4) then t = 0, s = 0 and (0, z) is a solution of (3.4.2). It follows from Proposition 3.4.1 that  $\overline{P}$  is semismooth, and hence  $\Phi$  is semismooth too.

Define a merit function of (3.4.4) by

$$\theta(y) = \|\Phi(y)\|^2$$

and define  $\beta:\Re^2\times\Re^{n+p(m+q+1)}\to\Re_+$  by

$$\beta(y) = \gamma \min \left\{ 1, \theta(y) \right\}.$$

Choose  $(\bar{t},\bar{s}) \in \Re^2_{++}$  and  $\gamma \in (0,1)$  such that  $\gamma(\bar{t}^2 + \bar{s}^2) < 1$ . Let  $\bar{y} = (\bar{t},\bar{s},0) \in \Re^2 \times \Re^{n+p(m+q+1)}$ . And let

$$\Omega := \left\{ y = (t, s, z) \in \Re^2 \times \Re^{n + p(m+q+1)} : (t, s) \ge \beta(y)(\bar{t}, \bar{s}) \right\}.$$

Then, since  $\beta(y) \leq \gamma < 1$  for any  $y \in \Re^2 \times \Re^{n+p(m+q+1)}$ , it follows that for any  $z \in \Re^{n+p(m+q+1)}$ ,

$$(\bar{t}, \bar{s}, z) \in \Omega.$$

Proposition 3.4.2 The following relations hold:

$$\Phi(y) = 0 \Leftrightarrow \beta(y) = 0 \Leftrightarrow \Phi(y) = \beta(y)\bar{y}.$$

**Proof.** It follows from the definitions of  $\Phi(\cdot)$  and  $\beta(\cdot)$  that

$$\Phi(y) = 0 \Leftrightarrow \beta(y) = 0$$
 and  $\beta(y) = 0 \Rightarrow \Phi(y) = \beta(y)\bar{y}$ 

Then, we only need to prove

$$\Phi(y) = \beta(y)\bar{y} \Rightarrow \beta(y) = 0.$$

However, this is an easy task because from  $\Phi(y) = \beta(y)\bar{y}$  we have

$$\theta(y) = \|\Phi(y)\|^2 = \beta(y)^2 \left(\bar{t}^2 + \bar{s}^2\right) \le \gamma^2 \left(\bar{t}^2 + \bar{s}^2\right) < 1.$$

Therefore,

$$\beta(y) = \gamma \theta(y) = \gamma \beta(y)^2 \left(\bar{t}^2 + \bar{s}^2\right).$$
(3.4.5)

If  $\beta(y) \neq 0$ , it follows from (3.4.5) and the fact  $\beta(y) \leq \gamma$  that

$$1 = \gamma \beta(y) \left( \bar{t}^2 + \bar{s}^2 \right) \le \gamma^2 \left( \bar{t}^2 + \bar{s}^2 \right),$$

which contradicts the fact that  $\gamma^2 (\bar{t}^2 + \bar{s}^2) < 1$ . This contradiction completes our proof.

Now, we develop a smoothing Newton-type algorithm for solving (3.4.4).

#### Algorithm 3.4.1 (Smoothing Newton-Type Algorithm)

#### Step 0. (Initialization)

Choose constants  $\rho \in (0, 1)$  and  $\sigma \in (0, 1/2)$ . Let  $t^0 = \overline{t}$ ,  $s^0 = \overline{s}$ ,  $z^0 \in \Re^{n+p(m+q+1)}$ be an arbitrary point and  $y^0 = (t^0, s^0, z^0)$ . Set k := 0.

#### Step 1. (Stop Test)

If  $\Phi(y^k) = 0$  then stop. Otherwise, let  $\beta_k := \beta(y^k)$ .

#### Step 2. (Compute Search Direction)

Let

$$\pi^{k} = -\left(\nabla_{z}^{T}\bar{P}(t^{k}, z^{k})\right)^{-1} \left[\bar{P}(t^{k}, z^{k}) + \nabla_{t}^{T}\bar{P}(t^{k}, z^{k})\left(\beta_{k}\bar{t} - t^{k}\right)\right]$$
(3.4.6)

and

$$\delta_k = \bar{G}(t^k, x^k) + \nabla_t \bar{G}(t^k, x^k) \left(\beta_k \bar{t} - t^k\right) + \nabla_x^T \bar{G}(t^k, x^k) \left(\pi^k\right)_x, \qquad (3.4.7)$$

where  $(\pi^k)_x$  is the sub-vector constituted of the first *n* components of  $\pi^k$ . Compute  $\triangle y^k := (\triangle t^k, \triangle s^k, \triangle z^k) \in \Re^2 \times \Re^{n+p(m+q+1)}$  by

$$\Phi(y^k) + \nabla^T \Phi(y^k) \bigtriangleup y^k = \bar{y}^k, \qquad (3.4.8)$$

where

$$\bar{y}^k = \begin{pmatrix} \beta_k \bar{t} \\ \beta_k \bar{s} + \delta_k \\ 0 \end{pmatrix}.$$

#### Step 3. (Line Search)

Let  $m_k$  be the smallest nonnegative integer m satisfying

$$\theta(y^k + \rho^m \bigtriangleup y^k) \le \left[1 - 2\sigma \left(1 - \gamma \left(\bar{t}^2 + \bar{s}^2\right)\right)\rho^m\right]\theta(y^k).$$
(3.4.9)  
Let  $\lambda_k = \rho^{m_k}$  and  $y^{k+1} = y^k + \lambda_k \bigtriangleup y^k.$ 

Step 4. Set k := k + 1 and go to Step 1.

In the rest of this subsection, we discuss some properties of Algorithm 3.4.1.

**Lemma 3.4.1** For any  $y = (t, s, z) \in \Re^2_{++} \times \Re^{n+p(m+q+1)}$ . Suppose that  $\nabla \Phi(y)$  is nonsingular. Let  $\Delta y = (\Delta t, \Delta s, \Delta z)$  be the unique solution of the following equation

$$\Phi(y) + \nabla^T \Phi(y) \bigtriangleup y = \begin{pmatrix} \beta(y)\bar{t} \\ \beta(y)\bar{s} + \delta \\ 0 \end{pmatrix}.$$
 (3.4.10)

Here

$$\delta = \bar{G}(t,x) + \nabla_t \bar{G}(t,x) \left(\beta(y)\bar{t} - t\right) + \nabla_x^T \bar{G}(t,x) \left(\pi\right)_x$$
(3.4.11)

and

$$\pi = -\left(\nabla_{z}^{T}\bar{P}(t,z)\right)^{-1}\left[\bar{P}(t,z) + \nabla_{t}^{T}\bar{P}(t,z)\left(\beta(y)\bar{t}-t\right)\right],$$
(3.4.12)

where  $(\pi)_x$  is the sub-vector constituted of the first *n* components of  $\pi$ ,  $\nabla_t \bar{P}(t,z)$  is the first row of  $\nabla_{(t,z)}\bar{P}(t,z)$  and  $\nabla_z \bar{P}(t,z)$  is the submatrix of  $\nabla_{(t,z)}\bar{P}(t,z)$  obtained by just removing the first row of  $\nabla_{(t,z)}\bar{P}(t,z)$ . Then

$$t + \Delta t = \beta(y)\bar{t} \tag{3.4.13}$$

and

$$s + \Delta s = \beta(y)\bar{s}.\tag{3.4.14}$$

**Proof.** It is easy to see that for any t > 0,

$$\nabla \Phi(y) = \left( \begin{array}{ccc} 1 & 0 & 0 \\ \nabla_t \bar{G}(t,x) & 1 & \nabla_z^T \bar{G}(t,x) \\ \nabla_t^T \bar{P}(t,z) & 0 & \nabla_z^T \bar{P}(t,z) \end{array} \right)^T,$$

By the special structure of  $\nabla \Phi(y)$ , we know from (3.4.10) that

$$\begin{cases} t + \Delta t = \beta(y)\bar{t}, \\ \bar{G}(t,x) + s + \nabla_t \bar{G}(t,x) \Delta t + \Delta s + \nabla_z^T \bar{G}(t,x) \Delta z = \beta(y)\bar{s} + \delta, \\ \bar{P}(t,z) + \nabla_t^T \bar{P}(t,z) \Delta t + \nabla_z^T \bar{P}(t,z) \Delta z = 0. \end{cases}$$
(3.4.15)

From the first equality of (3.4.15), (3.4.13) holds. Furthermore, it follows from the first and last equality of (3.4.15) that  $\Delta z = \pi$ . Consequently, it follows from (3.4.11), (3.4.12) and the second equality of (3.4.15) that (3.4.14) holds. The proof is complete.

**Lemma 3.4.2** For any  $\tilde{y} = (\tilde{t}, \tilde{s}, \tilde{z}) \in \Re^2_{++} \times \Re^{n+p(m+q+1)}$ . Suppose that  $\nabla \Phi(\tilde{y})$  is nonsingular, then there exist a closed neighborhood  $\mathcal{N}(\tilde{y})$  of  $\tilde{y}$  and a positive number  $\tilde{\alpha} \in (0, 1]$  such that for any  $y = (t, s, z) \in \mathcal{N}(\tilde{y})$  and all  $\alpha \in (0, \tilde{\alpha}]$  we have  $(t, s) \in \Re^2_{++}$ ,  $\nabla \Phi(y)$  is nonsingular and

$$\theta(y + \alpha \bigtriangleup y) \le \left[1 - 2\sigma \left(1 - \gamma \left(\bar{t}^2 + \bar{s}^2\right)\right)\alpha\right]\theta(y).$$
(3.4.16)

**Proof.** Since  $\nabla \Phi(\tilde{y})$  is nonsingular and  $(\tilde{t}, \tilde{s}) \in \Re^2_{++}$ , there exists a closed neighborhood  $\mathcal{N}(\tilde{y})$  of  $\tilde{y}$  such that for any  $y = (t, s, z) \in \mathcal{N}(\tilde{y})$  we have  $(t, s) \in \Re^2_{++}$  and that  $\nabla \Phi(y)$  is nonsingular. For any  $y \in \mathcal{N}(\tilde{y})$ , let  $\Delta y = (\Delta t, \Delta s, \Delta z) \in \Re^2 \times \Re^{n+p(m+q+1)}$  be the unique solution of (3.4.10). By Lemma 3.4.1, for any  $y \in \mathcal{N}(\tilde{y})$ ,

$$t + \Delta t = \beta(y)\bar{t}$$

Then, for all  $\alpha \in [0, 1]$  and all  $y \in \mathcal{N}(\tilde{y})$ ,

$$t + \alpha \bigtriangleup t = (1 - \alpha)t + \alpha\beta(y)\overline{t} > 0.$$

By this, we know that  $\overline{G}(t, x)$  and  $\overline{P}(t, z)$  are continuously differentiable. By using a similar way to the proof of Lemma 5 in [130], we can prove that the lemma holds. Here, we omit the detailed proof.  $\blacksquare$ 

We can get directly the following result from Lemmas 3.4.1 and 3.4.2.

**Proposition 3.4.3** For any  $k \ge 0$ , if  $y^k \in \Re^2_{++} \times \Re^{n+p(m+q+1)}$  and  $\nabla \Phi(y^k)$  is nonsingular, then Algorithm 3.4.1 is well defined at k-th iteration and  $y^{k+1} \in \Re^2_{++} \times \Re^{n+p(m+q+1)}$ .

**Proposition 3.4.4** For each fixed  $k \ge 0$ , if  $(t^k, s^k) \in \Re^2_{++}$ ,  $y^k \in \Omega$  and  $\nabla \Phi(y^k)$  is nonsingular, then for any  $\alpha \in [0, 1]$  such that

$$\theta(y^k + \alpha \bigtriangleup y^k) \le \left[1 - 2\sigma \left(1 - \gamma \left(\overline{t}^2 + \overline{s}^2\right)\right) \alpha\right] \theta(y^k),$$

it holds that  $y^k + \alpha \bigtriangleup y^k \in \Omega$ .

**Proof.** We can prove it by using a similar way to the proof of Proposition 6 in [130]. ■

**Theorem 3.4.1** Suppose that for every  $k \ge 0$  with  $(t^k, s^k) \in \Re^2_{++}$  and  $y^k \in \Omega$  we have  $\nabla \Phi(y^k)$  is nonsingular. Then an infinite sequence  $\{y^k = (t^k, s^k, z^k)\}$  generated by Algorithm 3.4.1 satisfies that  $(t^k, s^k) \in \Re^2_{++}$  and  $y^k \in \Omega$ .

**Proof.** First, because  $y^0 = (\bar{t}, \bar{s}, z^0) \in \Omega$ , from Proposition 3.4.4, we have that  $y^1$  is well defined,  $(t^1, s^1) \in \Re^2_{++}$  and  $y^1 \in \Omega$ . Then, by repeatedly resorting to Proposition 3.4.4, we can prove that an infinite sequence  $\{y^k\}$  is generated,  $(t^k, s^k) \in \Re^2_{++}$  and  $y^k \in \Omega$ . The proof is complete.

# 3.4.3 Convergence Analysis

In this subsection, we prove the global and local superlinear convergence of Algorithm 3.4.1. To this end, we first discuss the CD-regularity of  $\Phi$ . We make the following assumptions.

(D1) There exists a  $\tilde{t} > 0$  such that matrix  $\nabla \Phi(y)$  is nonsingular for each  $y \in \Omega$  with  $t \in (0, \tilde{t})$ .

Assumption (D1) can be justified under further assumptions.

**Theorem 3.4.2** Let  $t \in \Re$ . Then  $\Phi$  is CD-regular at y = (t, s, z) if  $\overline{P}(t, \cdot)$  is CD-regular at z.

**Proof.** It is easy to see that  $\overline{P}$  is regular. It then follows from Proposition 2.3.15 in [23] that

$$\partial_{(t,z)}\bar{P}(t,z) \subseteq \partial_t \bar{P}(t,z) \times \partial_z \bar{P}(t,z).$$

Consequently, by this and (1.1.2), we see that every element Q in  $\partial \Phi(y)$  has the following form

$$Q = \begin{pmatrix} 1 & 0 & 0 \\ \zeta_1 & 1 & \zeta_2 \\ U_t & 0 & U_z \end{pmatrix}.$$

Here  $\zeta_1$  is the first component of  $\zeta$  and  $\zeta_2$  is the sub-vector of  $\zeta$  obtained by just removing the first component of  $\zeta$ , where  $\zeta \in \partial_{(t,z)}\overline{G}(t,x)$ ,  $U_t \in \partial_t \overline{P}(t,z)$  and  $U_z \in \partial_z \overline{P}(t,z)$ . It is obvious that Q is nonsingular if  $U_z$  is nonsingular. We obtain the desired result and complete the proof.

Recall

$$F(x, u, \mathbf{v}) = \nabla f(x) + \sum_{i=1}^{p} u_i \nabla_x g(x, v^i)$$

Let

$$l_i(z) = -u_i \nabla_v g(x, v^i) + \sum_{j=1}^q w_j^i \nabla c_j(v^i), \quad i = 1, \cdots, p$$

and

$$\nabla c(v^i) = \left(\nabla c_1(v^i), \cdots, \nabla c_q(v^i)\right), \quad i = 1, \cdots, p.$$

We make further the following assumptions.

(D2)  $\nabla_x F(x, u, \mathbf{v})$  is positive semidefinite. Moreover, it is positive definite in the null space of  $\operatorname{Span}(\nabla_x g(x, v))$ . That is,  $d^T \nabla_x F(x, u, \mathbf{v}) d > 0$  for all  $d \in \Re^n \setminus \{0\}$  satisfying  $\nabla_x g(x, v)^T d = 0$ .

(D3)  $\nabla_v l_i(z)$  is positive semidefinite. Moreover, it is positive definite in the null space of  $\text{Span}(\nabla c(v^i))$ . That is,  $d^T \nabla_v l_i(z) d > 0$  for all  $d \in \Re^n \setminus \{0\}$  satisfying  $\nabla c(v^i)^T d = 0$ .

The following theorem comes from [98], which shows that Assumptions (D2) and (D3) are sufficient for  $\nabla_z \bar{P}(t, z)$  to be nonsingular for every t > 0.

**Theorem 3.4.3** Let Assumptions (D2) and (D3) hold. Then  $\nabla_z \overline{P}(t, z)$  is nonsingular for every t > 0.

We also make the following assumptions.

- **(D4)**  $u_i > 0 \quad \forall \ i = 1, 2, \cdots, p.$
- (D5) The vectors  $\nabla_x g(x, v^i), i = 1, 2, \dots, p$  are linearly independent.

(D6) For each  $i = 1, 2, \dots, p$ , the vectors  $\nabla c_j(v^i), j \in I(v^i) := \{j : c_j(v^i) = 0\}$  are linearly independent.

- (D7)  $w_i^i c_j(v^i) \neq 0, \forall i = 1, 2, \dots, p \text{ and } j = 1, 2, \dots, q.$
- (D8) for all  $(d, \xi_1, \dots, \xi_p) \in S(x, v) \setminus \{0\}$ ,  $d^T \nabla_x F(x, u, \mathbf{v}) d + 2 \sum_{i=1}^p u_i d^T \nabla_{xv}^2 g(x, v^i) \xi_i - \sum_{i=1}^p \xi_i^T \nabla_v l_i(x, u_i, v^i, w^i) \xi_i > 0$ ,

where S(x, v) be the set of all  $(d, \xi_1, \dots, \xi_p) \in \Re^n \times \Re^{mp}$  satisfying

$$d^T \nabla_x g(x, v^i) + \xi_i^T \nabla_v g(x, v^i) = 0 \quad \text{for } i = 1, 2, \cdots, p,$$

and

$$\xi_i^T \nabla c_j(v^i) = 0 \quad \text{for} \quad i = 1, 2, \cdots, p, \quad j \in I(v^i)$$

The following theorem comes from [134], which shows that Assumptions (D4)-(D8) are sufficient for  $\bar{P}(0, \cdot)$  to be CD-regular at z.

**Theorem 3.4.4** Suppose that  $(0, z^*)$  is a solution of  $\overline{P}(t, z) = 0$  and Assumptions  $(D_4)$ - $(D_8)$  hold. Then  $\overline{P}(0, \cdot)$  is CD-regular at  $z^*$ .

**Remark 3.4.1** (a) By Theorems 3.4.3 and 3.4.2, we see that if Assumptions (D2) and (D3) hold, then  $\nabla \Phi(y)$  is nonsingular for all t > 0.

(b) By Theorems 3.4.4 and 3.4.2, we see that if Assumptions (D4)-(D8) hold, then  $\Phi$  is CD-regular at (0, 0, z).

**Theorem 3.4.5** Suppose that Assumption (D1) holds. Let  $\{y^k\}$  be the sequences generated by Algorithm 3.4.1. Then each accumulation point  $\tilde{y}$  of  $\{y^k\}$  is a solution of  $\Phi(y) = 0$ .

**Proof.** It follows from Proposition 3.4.4 and Assumption (D1) that an infinite sequence  $\{y^k\}$  is generated such that  $y^k \in \Omega$  for all  $k \ge 0$ . From the design of Algorithm 3.4.1,  $\theta\left(y^{k+1}\right) < \theta\left(y^k\right)$  for all  $k \ge 0$ . Hence, the two sequences  $\{\theta(y^k)\}$  and  $\{\beta(y^k)\}$  are monotonically decreasing. Since  $\theta(y^k)$ ,  $\beta(y^k) \ge 0$  ( $k \ge 0$ ), there exist  $\tilde{\theta}$ ,  $\tilde{\beta} \ge 0$  such that  $\theta(y^k) \to \tilde{\theta}$  and  $\beta(y^k) \to \tilde{\beta}$  as  $k \to \infty$ . If  $\tilde{\theta} = 0$  and  $\{y^k\}$  has an accumulation point  $\tilde{y}$ , then from the continuity of  $\theta(\cdot)$  and  $\beta(\cdot)$  we obtain  $\theta(\tilde{y}) = 0$  and  $\beta(\tilde{y}) = 0$ . Then, we obtain the desired result. Suppose that  $\tilde{\theta} > 0$  and  $\tilde{y} = (\tilde{t}, \tilde{s}, \tilde{z}) \in \Re^2 \times \Re^{n+p(m+q+1)}$  is an accumulation point of  $\{y^k\}$ . By taking a subsequence if necessary, we may assume that  $\{y^k\}$  converges to  $\tilde{y}$ . It is easy to see that  $\tilde{\theta} = \theta(\tilde{y})$ ,  $\tilde{\beta} = \beta(\tilde{y})$  and  $\tilde{y} \in \Omega$ . Thus, from  $\beta(\tilde{y}) = \gamma \min\{1, \theta(\tilde{y})\} > 0$  and  $\tilde{y} \in \Omega$ , we see that  $(\tilde{t}, \tilde{s}) \in \Re^2_{++}$ . Then, from Assumption (D1),  $\nabla \Phi(\tilde{y})$  exists and is nonsingular. Hence, from Lemma 3.4.2, there exist a closed neighborhood  $\mathcal{N}(\tilde{y})$  of  $\tilde{y}$  and a positive number  $\tilde{\alpha} \in (0, 1]$  such that for any  $y = (t, s, z) \in \mathcal{N}(\tilde{y})$  and all  $\alpha \in (0, \tilde{\alpha}]$  we have  $(t, s) \in \Re^2_{++}$ ,  $\nabla \Phi(y)$  is nonsingular and (3.4.16) holds. Therefore, for a nonnegative integer l such that  $\rho^l \in (0, \tilde{\alpha}]$ , we have

$$\theta(y^k + \rho^l \bigtriangleup y^k) \le \left[1 - 2\sigma \left(1 - \gamma \left(\bar{t}^2 + \bar{s}^2\right)\right)\rho^l\right]\theta(y^k)$$

for all sufficiently large k. Then, for every sufficiently large k, we see that  $m_k \leq l$  and hence  $\rho^{m_k} \geq \rho^l$ . Then

$$\theta(y^{k+1}) \le \left[1 - 2\sigma\left(1 - \gamma\left(\bar{t}^2 + \bar{s}^2\right)\right)\rho^{m_k}\right]\theta(y^k) \le \left[1 - 2\sigma\left(1 - \gamma\left(\bar{t}^2 + \bar{s}^2\right)\right)\rho^l\right]\theta(y^k)$$

for all sufficiently large k. This contradicts the fact that the sequence  $\{\theta(y^k)\}$  converges to  $\tilde{\theta} > 0$ . So, we complete our proof.

**Theorem 3.4.6** Suppose that Assumption (D1) holds and  $y^*$  is an accumulation point of the infinite sequence  $\{y^k\}$  generated by Algorithm 3.4.1. Suppose that  $\overline{P}$  is CD-regular at  $(t^*, z^*)$ . Then the whole sequence  $\{y^k\}$  converges to  $y^*$ , and

$$\left\|y^{k+1} - y^*\right\| = o(\left\|y^k - y^*\right\|).$$
 (3.4.17)

**Proof.** First, it follows from Theorem 3.4.5 that  $y^* = (t^*, s^*, z^*)$  is a solution of  $\Phi(y) = 0$ , which implies that  $t^* = 0$  and  $s^* = 0$ . Let

$$\Psi(t,z) = \left(\begin{array}{c} t\\ \bar{P}(t,z) \end{array}\right)$$

Then, from Proposition 1.1.1, for all (t, z) sufficiently close to  $(t^*, z^*)$ ,

$$\left\|\nabla\Psi(t,z)^{-1}\right\| = O(1).$$

Hence, from the special structure of  $\nabla \Phi(y)$ , the definition of semismoothness and Proposition 1.1.1, we have that for  $(t^k, z^k)$  sufficiently close to  $(0, z^*)$ ,

$$\begin{aligned} \left\| \left( t^{k}, z^{k} \right) + \left( \bigtriangleup t^{k}, \bigtriangleup z^{k} \right) - (0, z^{*}) \right\| \\ &= \left\| \left( t^{k}, z^{k} \right) + \nabla \Psi(t^{k}, z^{k})^{-1} \left[ -\Psi(t^{k}, z^{k}) + \beta_{k} \left( \bar{t}, 0 \right) \right] - (0, z^{*}) \right\| \\ &= O \left( \left\| \Psi(t^{k}, z^{k}) - \Psi(0, z^{*}) - \nabla \Psi(t^{k}, z^{k}) \left( \left( t^{k}, z^{k} \right) - (0, z^{*}) \right) \right\| + \beta_{k} \bar{t} \right) \\ &= O \left( \left\| \left( \left( t^{k}, z^{k} \right) - (0, z^{*}) \right) \right\| \right) + O \left( \theta(y^{k}) \right). \end{aligned}$$
(3.4.18)

Noting that  $\Phi$  is locally Lipschitz continuous at  $(0, 0, z^*)$ , we know that for all  $y^k$  sufficiently close to  $y^*$ ,

$$\theta(y^k) = \left\| \Phi(y^k) \right\|^2 = O\left( \left\| y^k - y^* \right\|^2 \right), \tag{3.4.19}$$

which implies, together with (3.4.14), that

$$|s^{k} + \Delta s^{k} - s^{*}| = \beta_{k}\bar{s} = O\left(||y^{k} - y^{*}||^{2}\right).$$
 (3.4.20)

Therefore, we know, from (3.4.18), (3.4.19) and (3.4.20), that for all  $y^k$  sufficiently close to  $y^*$ ,

$$\|y^{k} + \Delta y^{k} - y^{*}\| = o\left(\|y^{k} - y^{*}\|\right).$$
 (3.4.21)

On the other hand, by Proposition 1.1.2, we know that for all  $y^k$  sufficiently close to  $y^*$ ,

$$||y^{k} - y^{*}|| = O(||\Phi(y^{k}) - \Phi(y^{*})||).$$

Consequently,

$$\begin{aligned} \theta(y^{k} + \Delta y^{k}) &= \left\| \Phi \left( y^{k} + \Delta y^{k} \right) \right\|^{2} \\ &= O \left( \left\| y^{k} + \Delta y^{k} - y^{*} \right\|^{2} \right) \\ &= o \left( \left\| y^{k} - y^{*} \right\|^{2} \right) \\ &= o \left( \left\| \Phi \left( y^{k} \right) - \Phi \left( y^{*} \right) \right\|^{2} \right) \\ &= o \left( \theta(y^{k}) \right). \end{aligned}$$

Therefore, we have that for all  $y^k$  sufficiently close to  $y^*$ ,

$$y^{k+1} = y^k + \triangle y^k,$$

which implies, together with (3.4.21), that (3.4.17) holds. The proof is complete.

# 3.4.4 Preliminary Numerical Examples

In this subsection, we report our preliminary numerical test results. We tested 14 problems which we call Problems 3.4.1-3.4.14. Problems 3.4.1-3.4.4, 3.4.6 and 3.4.9 are Problems 3.3.1-3.3.4, 3.3.6 and 3.3.7, respectively. Problems 3.4.5 is an problem in which the dimension of the parameter v is 1, whereas Problems 3.4.7-3.4.8 and 3.4.10-3.4.12 are some problems in which the dimension of the parameter v is 2. Problems 3.4.13-3.4.14 are two problems with higher dimension decision variable.

#### Problem 3.4.1

$$f(x) = 1.21 \exp(x_1) + \exp(x_2), \quad g(x, v) = v - \exp(x_1 + x_2),$$
$$V = [0, 1], \ p = 1, (x^0, v^0) = (2, -2, 1).$$

Problem 3.4.2

$$f(x) = x_1^2 + x_2^2 + x_3^2, \quad g(x,v) = x_1 + x_2 \exp(x_3 v) + \exp(2v) - 2\sin(4v),$$
$$V = [0,1], \quad p = 1, (x^0, v^0) = (-2, 0, 4, 1).$$

Problem 3.4.3

$$f(x) = \frac{1}{3}x_1^2 + \frac{1}{2}x_1 + x_2^2, \quad g(x,v) = (1 - x_1^2v^2)^2 - x_1v^2 - x_2^2 + x_2,$$
$$V = [0,1], \quad p = 1, (x^0, v^0) = (-4, -1, 1).$$

Problem 3.4.4

$$f(x) = x_1^2 + (x_2 - 3)^2, \quad g(x, v) = x_2 - 2 + x_1 \sin(v/(x_2 - 0.5)),$$
  
 $V = [0, 3], \quad p = 1, (x^0, v^0) = (1, 6, 1).$ 

Problem 3.4.5

$$f(x) = 2x_1^2 + 2x_1x_3 + 4x_2^2 + x_3^2,$$
  

$$g(x, v) = x_1 + x_1^2 \sin(2v) + 3x_1x_2 + x_2^2 \cos(3v) + x_3^2 - v,$$
  

$$V = [0, 3\pi], \ p = 1, (x^0, v^0) = (2, 3, 4, 1).$$

# Problem 3.4.6

$$f(x) = (x_1 - 2x_2 + 5x_2^2 - x_2^3 - 13)^2 + (x_1 - 14x_2 + x_2^2 + x_2^3 - 29)^2,$$
  

$$g(x, v) = x_1^2 + 2x_2v + \exp(x_1 + x_2) - \exp(v),$$
  

$$V = [0, 1], \ p = 1, (x^0, v^0) = (1, -1, 1).$$

Problem 3.4.7  

$$f(x) = \frac{1}{3}x_1^2 + \frac{1}{2}x_1 + x_2^2, \quad g(x,v) = (1 - x_1^2v_1^2)^2 - x_1v_2^2 - x_2^2 + x_2,$$

$$V = [0,2] \times [0,1], \quad p = 2, \quad (x^0,v^0) = (-1,-1,0,0,0,1).$$

Problem 3.4.8

$$f(x) = (x_1 - 2)^2 + x_2^2, \quad g(x, v) = x_1^2 \cos(v_1) + x_2 \sin(v_2) - 4,$$
  
$$V = [0, \pi] \times [0, \pi], \quad p = 1, \quad (x^0, v^0) = (-1, -1, 1, 0).$$

Problem 3.4.9

$$f(x) = x_1^2 + x_2^2 + x_3^3,$$
  

$$g(x,v) = x_1(v_1 + v_2^2 + 1) + x_2(v_1v_2 - v_2^2) + x_3(v_1v_2 + v_2^2 + v_2) + 1,$$
  

$$V = [0,1] \times [0,1], \quad p = 1, \quad (x^0, v^0) = (1,1,1,1,1).$$

Problem 3.4.10

$$f(x) = x_1^2 + x_2^2 + x_3^2,$$
  

$$g(x, v) = x_1 + x_2 \exp(x_3 v_1) - \exp(2x_1 v_2) + \sin(4v_1),$$
  

$$V = [0, 1] \times [0, 1], \quad p = 2, \quad (x^0, v^0) = (1, 1, 1, 1, 1, 0, 1).$$

Problem 3.4.11

$$f(x) = (x_1 - 3)^2 + x_2^2 - x_2,$$
  

$$g(x, v) = x_1^2 v_1 \cos(v_1 v_2) + (x_2 - 1) v_1^2 \sin(v_2 x_1 - \frac{13}{9}\pi) - 4v_2 + x_1,$$
  

$$V = [0, 2] \times [1, 2], \quad p = 1, \quad (x_0, v_0) = (1, 1, 0, 0).$$

Problem 3.4.12

$$f(x) = \frac{1}{2}(x_1^2 + x_2^2 + x_3^2 + x_4^2),$$
  

$$g(x, v) = \sin(v_1 v_2) - x_1 - x_2 v_1 - x_3 v_2 - x_4 v_1 v_2,$$
  

$$V = [0, 1] \times [0, 1], \quad p = 1, \quad (x^0, v^0) = (2, 2, 2, 2, 1, 0).$$

Problem 3.4.13

$$f(x) = \frac{1}{2}x^{T}x, \quad g(x,v) = 3 + 4.5\sin(4.7\pi(v-1.23)/8) - \sum_{i=1}^{n} x_{i}v^{i-1},$$
$$V = [0,1], \quad p = 1, \ (x^{0},v^{0}) = (2,2,\cdots,2,1).$$

Problem 3.4.14

$$f(x) = \int_0^1 \left(\sum_{i=1}^n x_i t^{i-1} - \tan t\right)^2 dt, \quad g(x,v) = \tan v - \sum_{i=1}^n x_i v^{i-1},$$
$$V = [0,1], \quad p = 1.$$

We first implemented Algorithm 3.4.1 for Problems 3.4.1-3.4.12 in MATLAB and the numerical experiments were done by using a Pentium III 733MHz computer with 256 MB of RAM. We use  $\|\Phi(y^k)\| \leq 10^{-6}$  as the stopping criterion for Algorithm 3.4.1. The values of  $\overline{G}(t,x)$  and  $\nabla \overline{G}(t,x)$  were computed by using the function **quad** in MATLAB when V is an interval in  $\Re$  and the function **dblquad** when V is a box set in  $\Re^2$ . The parameters used in algorithm are specified as follows

$$\gamma = 0.5, \ \rho = 0.5, \ \sigma = 0.001, \ \bar{t} = \bar{s} = 0.5.$$

The starting points  $t^0$ ,  $s^0$  for all problems are set  $t^0 = \bar{t}$ ,  $s^0 = \bar{s}$ . The starting points  $u^0, w^0$  are equal to 1.0e, 1.0e for Problems 3.4.1-3.4.12, where e is the vector of ones. We compared Algorithm 3.4.1 with fseminf. For the solver fseminf, we use all the default values.
In the test of Problems 3.4.1-3.4.12, the values of p are estimated by a similar adaptive strategy to that used in the implement of Algorithm 3.3.1. By that adaptive strategy, we get p = 1 for 10 of 12 test problems and p = 2 for other two test problems.

The test results for Problems 3.4.1-3.4.12 are summarized in Tables 3.4 and 3.5. In Table 3.4,  $\Phi(y^k)$  is the value of the function  $\Phi(y)$  in (3.4.4) at the k-th iteration. In Table 3.5, **n.it** represents the number of the total iterations; **cpu** is the total cost time in seconds for solving the SIP problem;  $f(x^k)$  is the value of the objective function in the SIP problem at the final iteration; and  $G(x^k)$  is the value of the function G(x) of (3.3.5) at the final iteration.

Problem	k	$\left\ \Phi(y^k)\right\ $	Problem	k	$\left\ \Phi(y^k)\right\ $	Problem	k	$\ \Phi(y^k)\ $
3.4.1	5	1.3496e-2	3.4.2	7	2.9284e-3	3.4.3	8	8.6888e-3
	6	9.0575e-5		8	5.1470e-6		9	1.2333e-4
	7	4.0332e-9		9	2.4003e-11		10	2.0364e-8
3.4.4	17	3.6864e-6	3.4.5	7	1.8260e-2	3.4.6	18	9.7817e-3
	18	1.3323e-6		8	2.6251e-4		19	3.3488e-5
	19	4.7863e-7		9	7.7329e-8		20	5.1996e-10
3.4.7	14	2.5145e-4	3.4.8	6	3.4413e-3	3.4.9	8	2.6080e-2
	15	1.1943e-6		7	1.5087e-5		9	4.7719e-4
	16	1.9282e-8		8	3.7231e-9		10	1.2153e-7
3.4.10	7	4.6280e-3	3.4.11	6	4.7301e-3	3.4.12	8	8.2845e-4
	8	1.1212e-5		7	2.7131e-5		9	8.6175e-5
	9	1.5885e-10		8	1.0377e-7		10	8.8161e-7

Table 3.4: The last three iterates generated by Algorithm 3.4.1

The results reported in Tables 3.4 and 3.5 show that Algorithm 3.4.1 performs well for these test problems. From Table 3.4, we can see that Algorithm 3.4.1 indeed has superlinear convergence property. From Table 3.5, we can see that Algorithm 3.4.1 uses less CPU time than **fseminf** for 7 test problems and **fseminf** uses less CPU time than Algorithm 3.4.1 for other 5 test problems. Moreover, it appears from Table 3.5 that Algorithm 3.4.1 indeed can ensure the feasibility of the test problems.

We also implemented Algorithm 3.4.1 for Problems 3.4.13-3.4.14 in FORTRAN 77 by using a Pentium III 1133 MHz computer with 256 MB memory. The dimensions (n) of

	Algorithm 3.4.1						fseminf		
Problem	n.it	cpu	$f(x^k)$	$G(x^k)$	n.it	cpu	$f(x^k)$	$G(x^k)$	
3.4.1	7	0.05	2.2	0	7	0.17	2.1989	7.804e-8	
3.4.2	9	0.17	5.3347	3.456e-13	30	0.50	5.3242	7.467e-5	
3.4.3	10	0.13	0.1945	0	3	0.03	0.1945	0	
3.4.4	19	0.16	1	6.357e-9	10	0.14	1	2.568e-3	
3.4.5	9	0.33	0	0	7	0.06	0	0	
3.4.6	20	0.28	97.1589	0	8	0.19	97.1589	2.010e-24	
3.4.7	16	1.92	0.3820	2.054e-12	13	2.23	0.3820	1.221e-7	
3.4.8	8	0.91	0	0	1	1.67	0	0	
3.4.9	10	13.75	1	0	7	4.78	1	0	
3.4.10	9	3.64	0	0	6	4.75	0	0	
3.4.11	8	1.23	1.0191	0	5	2.78	1.0191	0	
3.4.12	10	1.58	0.0885	0	2	1.88	0.0885	1.611e-10	

Table 3.5: Test results for Algorithm 3.4.1 and fseminf

the two problems are both chosen by 20, 40, 60, 80, 100 and 200. All calculation within the driving programs, test problems and optimization code are carried out in double precision. In the test of the two problems, the termination condition is  $\|\Phi(y^k)\| \leq 10^{-5}$ , the starting points  $u^0$ ,  $w^0$  are set 0.5**e** and 0.5**e**, respectively, and other parameters are same to those in the test of Problems 3.4.1-3.4.12.

The test results for Problem 3.4.13 and 3.4.14 are given in Table 3.6 and 3.7, respectively.

n	ITK	CPU	$\bar{G}(t^k, x^k)$	$\theta(y^k)$	$\overline{f(x^k)}$
20	50	0.61	6.36e-7	3.61e-11	0.344
40	38	0.82	8.36e-11	2.17e-13	0.742
60	92	1.12	7.91e-10	5.28e-11	1.188
80	96	1.59	8.45e-9	6.16e-13	1.461
100	117	2.56	1.41e-8	5.28e-11	1.685
200	160	3.24	7.65e-8	2.86e-11	2.720

Table 3.6: Test Results of Problem 3.4.13

n	ITK	CPU	$\bar{G}(t^k, x^k)$	$\theta(y^k)$	$f(x^k)$		
20	38	0.23	8.46e-8	7.53e-11	1.09		
40	129	2.37	2.42e-9	8.71e-11	2.26		
60	138	6.18	1.51e-7	6.33e-11	2.71		
80	161	10.71	7.55e-9	4.65e-12	3.75		
100	192	19.10	1.58e-8	4.41e-11	2.98		
200	210	80.14	3.75e-9	1.80e-13	3.41		

Table 3.7: Test Results of Problem 3.4.14

Analogous to the smoothing projected Newton-type algorithm, we will work on how to find a good way to determine a suitable number p in the KKT system of the SIP problem.

# 3.5 Some Comments

In this chapter, we have presented three kinds of algorithms for solving SIP problems, which are called smoothing SQP algorithm, smoothing projected Newton-type algorithm and smoothing Newton-type algorithm. The feasibility is ensured via the aggregated constraint in all three algorithms. At each iteration of the smoothing SQP algorithm, we only need to solve a quadratic program which is always feasible and solvable, and for other two algorithms, we only need to solve a system of linear equations at each iteration of algorithm. However, the smoothing SQP algorithm has only global convergence property. Moreover, from Theorem 3.2.1, we see that the accumulation point of sequence generated by the smoothing SQP algorithm is only a generalized stationary point of an equivalent problem. Therefore, the discussion of the smoothing SQP algorithm is preliminary. For smoothing projected Newton-type algorithm and smoothing Newton-type algorithm, we have proved their global and local superlinear convergence properties under some mild conditions. The main difference between the smoothing projected Newton-type algorithm and the smoothing Newton-type algorithm lies in the fact that the accumulation point of sequence generated by first algorithm may not be the stationary point of the original SIP problem, whereas, each accumulation point of sequence generated by second algorithm is a stationary point of the original SIP problem. Preliminary numerical tests for the three kinds of algorithms show that these algorithms perform well whenever the evaluation of the corresponding integral function is not very expensive. It is wondered whether the methods developed in this chapter can be applied to SIP problem such that the dimension of the decision variable is large, which is a topic discussed in the next chapter.

# Chapter 4

# A Method for Solving Large Scale SIP Problems

## 4.1 Introduction

Some large-scale SIP problems arise from the modelling of optimal control and approximation (see [58, 150, 171]). In order to increase the control precision in optimal control problem, one should increase the number of switching points. That is, the larger the number of switching points is set, the higher the control precision is. If one sets a large number of switching points, the discretization of the control space will lead to large scale SIP problems. In approximation theory, if a function f(v) is approximated on the interval [a, b] by a polynomial

$$f_N(v) = \sum_{j=1}^N x_j v^{j-1}$$

and the approximation is in the Chebyshev norm, then we get a SIP problem. It is clear that the larger the order of polynomial is, the higher the approximation precision is. When a very high order polynomial is used to approximate f on [a, b], a large scale SIP problem is generated. However some efficient algorithms for small scale SIP problems do not directly translate into algorithms for large scale SIP problems. The smoothing projected Newton-type algorithm presented in previous chapter also cannot be used directly to solve large scale SIP problems. The facts stated above motivate us to find some efficient methods for solving large scale SIP problems.

In this chapter, we extend the smoothing projected Newton-type algorithm presented in Chapter 3 to solving large scale SIP problem. We modify this algorithm in three aspects. First, the dimension of system of linear equations in each iteration is decreased due to decomposition technique. Second, a truncated solution of the system is determined by an iterative method, in which the computation of the matrix-vector product, instead of the matrix factorization, is needed such that the implementation at each iteration is relatively simple and time-economic. Third, in order to guarantee the global convergence, a robust loss function [74] is chosen as a merit function and the projected gradient method inserted is used to decrease the merit function. This loss function uses a measure which does not weigh very large components of the variable heavily. Numerical results show that this loss function is a good merit function. This modified algorithm is called truncated projected Newton-type algorithm, and is suitable for handling large-scale SIP problems. The global convergence of this algorithm is proved and the superlinear convergence rate is analyzed. The detailed implementation is discussed and some numerical tests for solving large scale SIP problems, with examples up to 2000 decision variables, are reported in this chapter.

This chapter is organized as follows: we present a truncated projected Newton-type algorithm in Section 4.2; the convergence of the algorithm is analyzed in Section 4.3 and numerical tests are given in Section 4.4; we propose some comments in Section 4.5.

# 4.2 A Truncated Projected Newton-Type Algorithm

We still consider the SIP problem (3.1.1) with  $V = \{v \in \Re^m : a \leq v \leq b\}$  where  $a \in \Re^m$ ,  $b \in \Re^m$ , and a < b. From Section 3.3, we know that the KKT system of the SIP problem (3.1.1) can be reformulated as a equivalent system of constrained equations in the following

$$\Phi(w) = 0$$
  

$$u \ge 0, s \ge 0,$$
(4.2.1)

where  $w = (t, z) = (t, x, u, \mathbf{v}, s) \in \Re \times \Re^n \times \Re^p \times \Re^{mp} \times \Re$ , and

$$\Phi(w) = \begin{pmatrix} t \\ \bar{H}(w) \end{pmatrix}, \quad \bar{H}(w) = \begin{pmatrix} F(x, u, \mathbf{v}) \\ \mathbf{g}(x, \mathbf{v}) \\ \tilde{\phi}(t, x, \mathbf{v}) \\ \bar{G}(t, x) + s \end{pmatrix}.$$
(4.2.2)

Here, the meanings of F,  $\mathbf{g}$ ,  $\tilde{\phi}$  and  $\bar{G}$  are the same as those used in Section 3.3. In the smoothing projected Newton-type algorithm presented in Section 3.3, Newton direction is obtained by solving the following linear system

$$\Phi(w^k) + \nabla^T \Phi(w^k) \bigtriangleup w_k = \beta_k \bar{w}$$
(4.2.3)

where  $\Delta w_k = (\Delta t_k, \Delta x_k, \Delta u_k, \Delta \mathbf{v}_k, \Delta \mathbf{s}_k) \in \Re^{\tilde{n}}, \, \tilde{n} = n + 2 + (m+1)p, \, \bar{w} = (\bar{t}, 0), \, \bar{t} > 0$ and

$$\nabla^{T} \Phi(w) = \begin{pmatrix} 1 & 0_{1 \times n} & 0_{1 \times p} \\ 0_{n \times 1} & \nabla_{x}^{T} F(x, u, \mathbf{v}) & \nabla_{x} \mathbf{g}(x, \mathbf{v}) \\ 0 & \nabla_{x}^{T} g(x, v^{1}) & 0_{1 \times p} \\ \vdots & \vdots & \vdots \\ 0 & \nabla_{x}^{T} g(x, v^{p}) & 0_{1 \times p} \\ \nabla_{t}^{T} \overline{\phi}(t, x, v^{1}) & \nabla_{x}^{T} \overline{\phi}(t, x, v^{1}) & 0_{m \times p} \\ \vdots & \vdots & \vdots \\ \nabla_{t}^{T} \overline{\phi}(t, x, v^{p}) & \nabla_{x}^{T} \overline{\phi}(t, x, v^{p}) & 0_{m \times p} \\ \nabla_{t} \overline{G}(t, x) & \nabla_{x}^{T} \overline{G}(t, x) & 0_{1 \times p} \end{pmatrix}$$

$$(4.2.4)$$

$$\begin{pmatrix} 0_{1 \times m} & \cdots & 0_{1 \times m} & 0 \\ u_{1} \nabla_{v}^{T} (\nabla_{x} g(x, v^{1})) & \cdots & u_{p} \nabla_{v}^{T} (\nabla_{x} g(x, v^{p})) & 0_{n \times 1} \\ \nabla_{v}^{T} g(x, v^{1}) & \cdots & 0_{1 \times m} & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0_{1 \times m} & \cdots & \nabla_{v}^{T} p \overline{\phi}(t, x, v^{p}) & 0_{m \times 1} \\ \vdots & \ddots & \vdots & \vdots \\ 0_{m \times m} & \cdots & \nabla_{v}^{T} p \overline{\phi}(t, x, v^{p}) & 0_{m \times 1} \\ \vdots & \ddots & \vdots & \vdots \\ 0_{n \times m} & \cdots & \nabla_{v}^{T} p \overline{\phi}(t, x, v^{p}) & 0_{m \times 1} \\ 0_{1 \times m} & \cdots & 0_{1 \times m} & 1 \end{pmatrix}$$

In order to solve large-scale problem, we decompose the system (4.2.3) into

$$t^k + \Delta t_k = \beta_k \bar{t}, \tag{4.2.5}$$

$$\nabla_{x}^{T}F(x^{k}, u^{k}, \mathbf{v}^{k}) \bigtriangleup x_{k} + \sum_{j=1}^{p} \nabla_{x}g(x^{k}, v^{jk}) \bigtriangleup u_{k}^{j} + \sum_{j=1}^{p} S_{kj} \bigtriangleup v_{k}^{j} = -F(x^{k}, u^{k}, \mathbf{v}^{k}), \quad (4.2.6)$$

$$\begin{cases}
\nabla_{x}^{T}g(x^{k}, v^{1k}) \bigtriangleup x_{k} + \nabla_{v^{1}}^{T}g(x^{k}, v^{1k}) \bigtriangleup v_{k}^{1} = -g(x^{k}, v^{1k}), \\
\cdots \cdots \\
\nabla_{x}^{T}g(x^{k}, v^{pk}) \bigtriangleup x_{k} + \nabla_{v^{p}}^{T}g(x^{k}, v^{pk}) \bigtriangleup v_{k}^{p} = -g(x^{k}, v^{pk}), \\
\nabla_{t}^{T}\bar{\phi}(t^{k}, x^{k}, v^{1k}) \bigtriangleup t_{k} + \nabla_{x}^{T}\bar{\phi}(t^{k}, x^{k}, v^{1k}) \bigtriangleup x_{k} \\
+ \nabla_{v^{1}}^{T}\bar{\phi}(t^{k}, x^{k}, v^{1k}) \bigtriangleup t_{k} = -\bar{\phi}(t^{k}, x^{k}, v^{1k}), \\
\nabla_{t}^{T}\bar{\phi}(t^{k}, x^{k}, v^{pk}) \bigtriangleup t_{k} + \nabla_{x}^{T}\bar{\phi}(t^{k}, x^{k}, v^{pk}) \bigtriangleup x_{k} \\
+ \nabla_{v^{p}}^{T}\bar{\phi}(t^{k}, x^{k}, v^{pk}) \bigtriangleup x_{k} \\
+ \nabla_{v^{p}}^{T}\bar{\phi}(t^{k}, x^{k}, v^{pk}) \bigtriangleup v_{k}^{p} = -\bar{\phi}(t^{k}, x^{k}, v^{pk})
\end{cases}$$

and

$$\nabla_t \bar{G}(t^k, x^k) \bigtriangleup t_k + \nabla_x^T \bar{G}(t^k, x^k) \bigtriangleup x_k + \bigtriangleup s_k = -\bar{G}(t^k, x^k) - s^k, \qquad (4.2.9)$$

where  $v^{jk}$   $(j = 1, 2, \dots, p)$  means the value of vector  $v^j$  at k-th iteration,  $S_{kj} = u_j^k \nabla_{v^j}^T \left( \nabla_x g(x^k, v^{jk}) \right) \in \Re^{n \times m}, \ j = 1, 2, \dots, p$ . Let

$$J_k = \left\{ j, 1 \le j \le p : \left| \det \left( \nabla_{v^j}^T \bar{\phi}(t^k, x^k, v^{jk}) \right) \right| \ge \varepsilon \right\},\$$

where  $\varepsilon > 0$  is given constant,  $K_k = \{1, 2, \dots, p\} \setminus J_k$ .  $p_k$  is the cardinal number of the index set  $K_k$ , i.e.,  $p_k = |K_k|$ .

From (4.2.5) and (4.2.8), it follows that

$$\Delta t_k = \beta_k \bar{t} - t^k \tag{4.2.10}$$

and

$$\Delta v_k^j = \tilde{v}^{jk} + M_{kj} \Delta x_k, \ j \in J_k, \tag{4.2.11}$$

where

$$\begin{split} \tilde{v}^{jk} &= -\left(\nabla_{v^{j}}^{T}\bar{\phi}(t^{k}, x^{k}, v^{jk})\right)^{-1} \left(\bar{\phi}(t^{k}, x^{k}, v^{jk}) + \nabla_{t}^{T}\bar{\phi}(t^{k}, x^{k}, v^{jk}) \left(\beta_{k}\bar{t} - t^{k}\right)\right), \\ M_{kj} &= -\left(\nabla_{v^{j}}^{T}\bar{\phi}(t^{k}, x^{k}, v^{jk})\right)^{-1} \nabla_{x}^{T}\bar{\phi}(t^{k}, x^{k}, v^{jk}). \end{split}$$

Substituting these equalities in (4.2.6)-(4.2.8), we obtain the following linear system

$$\begin{pmatrix} A_k & G_k & B_k \\ \tilde{G}_k & 0 & \tilde{E}_k \\ Z_k & \tilde{Z}_k \end{pmatrix} \begin{pmatrix} \Delta x_k \\ \Delta u_k \\ \Delta \bar{\mathbf{v}}_k \end{pmatrix} = \begin{pmatrix} b_{k1} \\ b_{k2} \\ b_{k3} \end{pmatrix}, \qquad (4.2.12)$$

where

$$A_{k} = \nabla_{x}^{T} F(x^{k}, u^{k}, \mathbf{v}^{k}) + \sum_{j \in J_{k}} S_{kj} M_{kj}, \quad G_{k} = \left(\nabla_{x} g(x^{k}, v^{1k}), \cdots, \nabla_{x} g(x^{k}, v^{pk})\right), \quad (4.2.13)$$

$$B_k = (S_{kj}, \quad j \in K_k) \in \Re^{n \times mp_k}, \tag{4.2.14}$$

$$\tilde{G}_{k} = G_{k}^{T} + E_{k}, \ e_{j}^{T} E_{k} = \begin{cases} \nabla_{v^{j}}^{T} g(x^{k}, v^{jk}) M_{kj}, & j \in J_{k}, \\ 0, & j \in K_{k}, \end{cases}$$
(4.2.15)

$$\tilde{E}_k \in R^{p \times mp_k}, \ e_j^T \tilde{E}_k = \begin{cases} 0, & j \in J_k, \\ \left(0, \cdots, \nabla_{v^j}^T g(x^k, v^{jk}), \cdots, 0\right), & j \in K_k, \end{cases}$$
(4.2.16)

$$Z_k = \left(\nabla_x \bar{\phi}(t^k, x^k, v^{jk}), \quad j \in K_k\right)^T \in \Re^{mp_k \times n}, \tag{4.2.17}$$

$$\tilde{Z}_k = \operatorname{diag}\left(\nabla_{v^j}^T \bar{\phi}(t^k, x^k, v^{jk}), \quad j \in K_k\right) \in \Re^{mp_k \times mp_k}, \tag{4.2.18}$$

$$b_{k1} = -F(x^k, u^k, \mathbf{v}^k) - \sum_{j \in J_k} S_{kj} \tilde{v}^{jk}, \qquad (4.2.19)$$

$$b_{k2} \in R^p, \ (b_{k2})_j = \begin{cases} -g(x^k, v^{jk}), & j \in K_k, \\ -g(x^k, v^{jk}) - \nabla_{v^j}^T g(x^k, v^{jk}) \tilde{v}^{jk}, & j \in J_k, \end{cases}$$
(4.2.20)

$$b_{k3} = \left(-\bar{\phi}(t^k, x^k, v^{jk}) - \nabla_t^T \bar{\phi}(t^k, x^k, v^{jk}) \left(\beta_k \bar{t} - t_k\right), \ j \in K_k\right) \in \Re^{mp_k},$$
(4.2.21)

and

$$riangle \overline{\mathbf{v}}_k = \left( \left( riangle v_k^j \right), \quad j \in K_k \right) \in \Re^{mp_k}.$$

It is remarked that the dimension of system of (4.2.12) is  $n + p + mp_k$  which is smaller than  $\tilde{n}$ .

The vector  $(\Delta x_k, \Delta u_k, \Delta \bar{\mathbf{v}}_k)$  is called a truncated solution of (4.2.12) if

$$\left\| \begin{pmatrix} A_k & G_k & B_k \\ \tilde{G}_k & 0 & \tilde{E}_k \\ Z_k & \tilde{Z}_k \end{pmatrix} \begin{pmatrix} \Delta x_k \\ \Delta u_k \\ \Delta \bar{\mathbf{v}}_k \end{pmatrix} - \begin{pmatrix} b_{k1} \\ b_{k2} \\ b_{k3} \end{pmatrix} \right\| \le \eta_k$$
(4.2.22)

for  $\eta_k > 0$ . After  $(\Delta x_k, \Delta u_k, \Delta \bar{\mathbf{v}}_k)$  is determinated, we calculate  $\Delta y_k$ ,  $\Delta t_k$  and  $\Delta v_k^j$ ,  $j \in J_k$  by (4.2.9)-(4.2.11), denote  $d_{tN}^k = \Delta w_k$ , and call  $d_{tN}^k$  a truncated solution of (4.2.3).

In Section 3.3, a simple merit function

$$\Psi(w) = \frac{1}{2} \sum_{j=1}^{\tilde{n}} \Phi_j^2(w)$$

is chosen and its gradient is

$$\nabla\Psi(w) = \nabla\Phi(w)\Phi(w)$$

In order to solve the large scale SIP problem, we consider the following function

$$\Psi_h(w) = \sum_{j=1}^{\tilde{n}} \rho_{h_j} \left( \Phi_j(w) \right), \qquad (4.2.23)$$

where

$$\rho_{h_j}\left(\xi\right) = \begin{cases} \xi^2/2 & \text{if } |\xi| \le h_j, \\ h_j \xi - h_j^2/2 & \text{otherwise,} \end{cases}$$

 $h_j, j = 1, 2, \dots, \tilde{n}$  are positive constants, and  $\rho_{h_j}(\xi)$  is linear in  $\xi$  for  $|\xi| > h_j$ . This function was proposed by Huber and Dutter (see [74] and [28]) for solving the least squares problems. The measure  $\rho(\xi)$  in this function does not weigh very large components of  $\xi$  heavily.

We use the function (4.2.23) as the merit function. The gradient of this function  $\Psi_h(w)$  is

$$\nabla \Psi_h(w) = \nabla \Phi(w) \Phi_h(w),$$

where

$$\Phi_{h}(w) = \sum_{j \in J_{h}} \Phi_{j}(w)e_{j} + \sum_{j \in K_{h}} \operatorname{sign}(\Phi_{j}(w))h_{j}e_{j}, \qquad (4.2.24)$$
$$J_{h} = \{j: \ 1 \le j \le \tilde{n}, \ |\Phi_{j}(w)| \le h_{j}\}, \quad K_{h} = \{1, 2, \cdots, \tilde{n}\} \setminus J_{h}.$$

The problem (4.2.1) is equivalent to finding a global solution of the following minimization problem:

$$\min_{\substack{w \in \mathbb{N}, \\ w \in \mathbb{N},$$

We call w a stationary point of (4.2.25) if it satisfies

$$\|\bar{d}_G(1)\| = 0, \tag{4.2.26}$$

where

$$\bar{d}_G(1) = \Pi_W(w - \gamma \nabla \Psi_h(w)) - w = \begin{pmatrix} -\gamma \nabla_t \Psi_h(w) \\ \Pi_Z(z - \gamma \nabla_z \Psi_h(w)) - z \end{pmatrix}, \quad (4.2.27)$$

where  $\gamma > 0$  is a constant, and W, Z and  $\Pi_W(\cdot)$  are the same as those defined in Section 3.3.

Let  $\alpha \in (0,1)$  be a constant. For a sequence  $\{w^k\}_{k=0}^{\infty}$ , we define

$$\beta_0 = \beta(w^0) = \alpha \min\{1, \|\bar{d}_G^0(1)\|^2\},\$$

and

$$\beta_{k} = \beta(w^{k}) := \begin{cases} \beta_{k-1}, & \text{if } \alpha \min\{1, \|\bar{d}_{G}^{k}(1)\|^{2}\} > \beta_{k-1} \\ \alpha \min\{1, \|\bar{d}_{G}^{k}(1)\|^{2}\}, & \text{otherwise.} \end{cases}$$
(4.2.28)

Now we state our truncated projected Newton-type algorithm for solving (4.2.1) below.

# Algorithm 4.2.1 (Truncated Projected Newton-Type Algorithm)

Step 0. (Initialization)

Choose constants  $\eta, \rho, \sigma \in (0, 1)$  with  $\sigma\eta < 1$ ,  $p_1 > 0$ ,  $p_2 > 2$  and  $\alpha > 0$ ,  $\bar{t} > 0$ with  $\alpha \bar{t} < 1$ ,  $h_j \ge 1$ ,  $j = 1, 2, \dots, \tilde{n}$ . Let  $\bar{w} = (\bar{t}, 0, 0, 0, 0)$ ,  $t_0 = \bar{t}$  and  $w^0 = (t_0, x^0, u^0, \mathbf{v}^0, \mathbf{s}^0)$  with  $u_i^0 \ge 0$   $(i = 1, \dots, p)$ ;  $s^0 \ge 0$ . Set k := 0.

Step 1. (Termination Test)

Compute

$$\xi_{k} = \min\left\{1, \frac{t_{k}}{|t_{k} + \nabla_{t}\bar{H}(w^{k})|\bar{H}_{h}(w^{k})|}, \frac{\eta \|\Phi(w^{k})\|}{\|\nabla\Psi_{h}(w^{k})\|}\right\},$$
(4.2.29)

where  $\nabla_t \bar{H}(w^k)$  is the first row of  $\nabla \bar{H}(w^k)$  and  $\bar{H}_h(w^k)$  is obtained by just removing the first row of  $\Phi_h(w^k)$  (see (4.2.24)).

$$\gamma_k = \begin{cases} \min\left\{\xi_k, \frac{\eta \Psi_h(w^k)}{\|\nabla \Psi_h(w^k)\|^2}\right\} & \text{if } \left|\Phi_j(w^k)\right| \le h_j, j = 1, 2, \cdots, \tilde{n}, \\ \xi_k & \text{otherwise.} \end{cases}, \quad (4.2.30)$$

If  $\|\vec{d}_G^k(1)\| = 0$ , then stop. Otherwise, compute  $\beta_k$  by (4.2.28), go to step 2.

#### Step 2. (Search Directions)

2.1) Compute negative gradient direction. Compute

$$d_G^k = -\gamma_k \nabla \Psi_h(w^k) + \beta_k \bar{w}. \tag{4.2.31}$$

106Pao Yue-kong Library PolyU · Hong Kong

$$\left| \Phi_j(w^k) \right| \le h_j, j = 1, 2, \cdots, \tilde{n},$$
 (4.2.32)

then go to 2.2), otherwise set  $d_{tN}^k = d_G^k$ , go to Step 3.

2.2) Compute truncated Newton direction. Determine  $(\Delta x_k, \Delta u_k, \Delta \bar{\mathbf{v}}_k)$  which satisfies

$$\left\| \begin{pmatrix} A_k & G_k & B_k \\ \tilde{G}_k & 0 & \tilde{E}_k \\ Z_k & \tilde{Z}_k \end{pmatrix} \begin{pmatrix} \Delta x_k \\ \Delta u_k \\ \Delta \bar{\mathbf{v}}_k \end{pmatrix} - \begin{pmatrix} b_{k1} \\ b_{k2} \\ b_{k3} \end{pmatrix} \right\| = o\left(\Psi_h(w^k)\right)$$
(4.2.33)

where  $A_k$ ,  $G_k$ ,  $B_k$ ,  $\tilde{G}_k$ ,  $\tilde{E}_k$ ,  $Z_k$ ,  $\tilde{Z}_k$ ,  $b_{k1}$ ,  $b_{k2}$ ,  $b_{k3}$  are given in (4.2.13)-(4.2.21). Compute  $\Delta t_k$ ,  $\Delta v_k^j$ ,  $j \in J_k$  and  $\Delta s_k$  by the formulas (4.2.9)-(4.2.11). Set  $d_{tN}^k = (\Delta t_k, \Delta x_k, \Delta u_k, \Delta \mathbf{v}_k, \Delta s_k)$ .

#### Step 3. (Line Search)

Let  $m_k$  be the smallest nonnegative integer m satisfying

$$\Psi_h(w^k + \bar{d}^k((\rho)^m)) \le \Psi_h(w^k) + \sigma \nabla \Psi_h(w^k)^T \tilde{d}^k_G((\rho)^m), \qquad (4.2.34)$$

where for any  $\lambda \in [0, 1]$ ,

$$\bar{d}^k(\lambda) = \tau^*(\lambda)\tilde{d}^k_G(\lambda) + (1 - \tau^*(\lambda))\tilde{d}^k_{tN}(\lambda).$$
(4.2.35)

Here

$$\tilde{d}_{G}^{k}(\lambda) := \Pi_{W}(w^{k} + \lambda d_{G}^{k}) - w^{k}, \quad \tilde{d}_{tN}^{k}(\lambda) := \Pi_{W}(w^{k} + \lambda d_{tN}^{k}) - w^{k}, \quad (4.2.36)$$

 $\tau^*(\lambda)$  is a solution of the following minimization problem:

$$\min_{\tau \in [0,1]} \frac{1}{2} \| \Phi(w^k) + \nabla^T \Phi(w^k) [\tau \tilde{d}_G^k(\lambda) + (1-\tau) \tilde{d}_{tN}^k(\lambda)] \|^2$$

Let  $\lambda_k = (\rho)^{m_k}$  and  $w^{k+1} = w^k + \overline{d}^k(\lambda_k)$ .

Step 4. Set k := k + 1 and go to Step 1.

**Remark 4.2.1** (1) Algorithm 4.2.1 is able to handle the sparse large scale SIP problems. In Step 2.2 of the algorithm, a truncated solution of the problem (4.2.12) is determined by using conjugate gradient method. Hence, the matrix factorizations are avoided, because this iterative algorithm requires computing only matrix-vector products. Since the SIP problem possesses the sparse date structure, the computation of the matrix,  $A_k$ , can take advantage of the sparsity of  $\nabla_x^T F(x^k, u^k, \mathbf{v}^k)$ . Therefore Algorithm 4.2.1 is applicable to the sparse large scale SIP problem.

(2) If the condition (4.2.32) is not satisfied, then only projected negative gradient direction is generated in the iteration, otherwise Step 2.2 is carried out and mixed projected directions are generated. In addition, if (4.2.32) is satisfied, then  $\Psi_h(w^k) = \Psi(w^k)$  holds.

(3) The condition 4.2.33 guarantees the convergence of Algorithm 4.2.1 which is discussed in next section. In the implementation of algorithm, one kind of choice of right side in (4.2.33) is  $\frac{1}{k+1} \min \{1, \Psi_h(w^k)\}$ .

(4)  $\tau^*(\lambda)$  is easily obtained and it is similar to that in Section 3.3.

(5) Another line search technique in Step 3 can be used if only projected negative gradient is search direction. Although it does not affect the convergence and its proof, it can decrease the number of inner iterations. In Section 4.4 we give a detailed description.

# 4.3 Convergence Analysis

In this section we discuss the convergence property of Algorithm 4.2.1. From the definition of  $\beta_k$ , the following lemma is obvious.

**Lemma 4.3.1**  $\{\beta_k\}$  defined in (4.2.28) has the following properties: (i)  $\{\beta_k\}$  is a non-increasing sequence; (ii) For all k,  $\beta_k$  satisfies

$$\beta_k \le \alpha \min\{1, \|\bar{d}_G^k(1)\|^2\}.$$

With the similar way to the proof of Proposition 3.3.3, we have the following descent property of  $\tilde{d}_G^k(\lambda)$  in Algorithm 4.2.1 and omit its proof.

**Lemma 4.3.2** Suppose that  $w^k = (t^k, z^k) \in W$  with  $t^k > 0$  is not a stationary point of (4.2.25). Then for any  $\lambda \in (0, 1]$ , it holds that

$$\nabla \Psi_h(w^k)^T \tilde{d}_G^k(\lambda) \le -\frac{\lambda}{\xi_k} (1 - \alpha \bar{t}) \|\bar{d}_G^k(1)\|^2 < 0.$$
(4.3.1)

**Remark 4.3.1** If (4.2.32) is not satisfied and  $d_G^k(1) \neq 0$ , then from Step 2.1 we know that only projected negative gradient is chosen as a search direction. Hence, Lemma 4.3.2 shows that this is a descent direction which implies that after number of iterations (4.2.32) is always satisfied.

Now we discuss the perturbed property of truncated solution generated in Step 2.2 of Algorithm 4.2.1 in the following lemma.

**Lemma 4.3.3** Let  $(\triangle x_k, \triangle u_k, \triangle \bar{\mathbf{v}}_k)$  be a truncated solution of (4.2.12), i.e.,

$$\begin{pmatrix} A_k & G_k & B_k \\ \tilde{G}_k & 0 & \tilde{E}_k \\ Z_k & \tilde{Z}_k \end{pmatrix} \begin{pmatrix} \Delta x_k \\ \Delta u_k \\ \Delta \bar{\mathbf{v}}_k \end{pmatrix} - \begin{pmatrix} b_{k1} \\ b_{k2} \\ b_{k3} \end{pmatrix} - \begin{pmatrix} r_{k1} \\ r_{k2} \\ r_{k3} \end{pmatrix} = 0, \quad (4.3.2)$$

where  $r_{k3} = \left( \left( r_j^{(k3)} \right), j \in K_k \right) \in \Re^{mp_k}, r_j^{(k3)} \in \Re^m, let \bigtriangleup s_k, \bigtriangleup t_k and \bigtriangleup v_k^j, j \in J_k$  be calculated by (4.2.9)-(4.2.11). Define  $\theta_k \in \Re^{\tilde{n}}$  such that

$$\Phi(w^k) + \nabla^T \Phi(w^k) \bigtriangleup w^k - \beta_k \bar{w} - \theta_k = 0, \qquad (4.3.3)$$

where  $\theta_k = \left(\theta_1^{(k)}, \cdots, \theta_{n+p+1}^{(k)}, \theta_{p1}^{(k)}, \cdots, \theta_{pp}^{(k)}, \theta_{\tilde{n}}^{(k)}\right)^T$ ,  $\theta_{pj}^{(k)} \in \Re^m$ ,  $j = 1, 2, \cdots, p$ . Then  $\theta_k$  satisfies

$$\theta_1^{(k)} = 0, \ \theta_{1+j}^{(k)} = r_j^{(k1)}, \ j = 1, 2, \cdots, n; \ \theta_{n+1+j}^{(k)} = r_j^{(k2)}, \ j = 1, 2, \cdots, p; \\ \theta_{pj}^{(k)} = \begin{cases} r_j^{(k3)}, & j \in K_k, \\ 0, & j \in J_k, \end{cases} \quad j = 1, 2, \cdots, p; \ \theta_{\tilde{n}}^{(k)} = 0, \end{cases}$$

*i.e.*,  $||r_k|| = ||\theta_k||$ , where  $r_k = (r_{k1}^T, r_{k2}^T, r_{k3}^T)^T$ .

**Proof.** Because  $\Delta t_k$  is calculated by (4.2.10),  $\theta_1^{(k)} = 0$  holds. From (4.3.2), (4.2.13)-(4.2.14) and (4.2.19), we have

$$\begin{aligned} r_{k1} &= A_k \bigtriangleup x_k + G_k \bigtriangleup u_k + B_k \bigtriangleup \overline{\mathbf{v}}_k - b_{k1} \\ &= \nabla_x^T F(x^k, u^k, \mathbf{v}^k) \bigtriangleup x_k + \sum_{j \in J_k} S_{kj} M_{kj} \bigtriangleup x_k + G_k \bigtriangleup u_k \\ &+ \sum_{j \in K_k} S_{kj} \bigtriangleup v_k^j + F(x^k, u^k, \mathbf{v}^k) + \sum_{j \in J_k} S_{kj} \tilde{v}^{jk} \\ &= \nabla_x^T F(x^k, u^k, \mathbf{v}^k) \bigtriangleup x_k + G_k \bigtriangleup u_k + \sum_{j=1}^p S_{kj} \bigtriangleup v_k^j + F(x^k, u^k, \mathbf{v}^k), \end{aligned}$$

where the third equality is due to (4.2.11). Then from (4.3.3), (4.2.2) and (4.2.4) we have that

$$\theta_{1+j}^{(k)} = r_j^{(k1)}, \ j = 1, 2, \cdots, n.$$

By (4.3.2),

$$r_{k2} = \tilde{G}_k \bigtriangleup x_k + \tilde{E}_k \bigtriangleup \bar{\mathbf{v}}_k - b_{k2}$$

From (4.2.15), (4.2.16) and (4.2.20), we get that for all  $j \in K_k$ ,

$$r_{j}^{(k2)} = g(x^{k}, v^{jk}) + \nabla_{x}^{T} g(x^{k}, v^{jk}) \bigtriangleup x_{k} + \nabla_{v^{j}}^{T} g(x^{k}, v^{jk}) \bigtriangleup v_{k}^{j}$$

$$= \theta_{n+1+j}^{(k)},$$

$$(4.3.4)$$

and for all  $j \in J_k$ ,

where the second equality in (4.3.4) and the third equality in (4.3.5) come from (4.3.3), (4.2.2) and (4.2.4), the second equality in (4.3.5) is due to (4.2.11).

By (4.3.2),

$$r_{k3} = Z_k \bigtriangleup x_k + \tilde{Z}_k \bigtriangleup \bar{\mathbf{v}}_k - b_{k3}$$

From (4.2.17), (4.2.18), (4.2.21) and (4.2.11), it follows that for all  $j \in K_k$ ,

$$r_{j}^{(k3)} = \phi(t^{k}, x^{k}, v^{jk}) + \nabla_{t}\phi(t^{k}, x^{k}, v^{jk}) \bigtriangleup t_{k}$$
$$+ \nabla_{x}^{T}\bar{\phi}(t^{k}, x^{k}, v^{jk}) \bigtriangleup x_{k} + \nabla_{v^{j}}^{T}\bar{\phi}(t^{k}, x^{k}, v^{jk}) \bigtriangleup v_{k}^{j}$$
$$= \theta_{pj}^{(k)},$$

where the second equality comes from (4.3.3), (4.2.2) and (4.2.4). Because  $\Delta v_k^j$ ,  $j \in J_k$  is calculated by (4.2.11), we have from (4.3.3), (4.2.2) and (4.2.4) that  $\theta_{pj}^{(k)} = 0$ , for all

 $j \in J_k$ . In addition,  $\theta_{\tilde{n}}^{(k)} = 0$ , for  $\Delta s_k$  is calculated by (4.2.9). We complete the proof.

Before we give the global convergence of Algorithm 4.2.1, we need the following lemma which shows that Algorithm 4.2.1 can keep  $t^k > 0$  at each iteration.

**Lemma 4.3.4** Let  $\{w^k\}$  be a sequence generated by Algorithm 4.2.1. Then for each k,  $k = 0, 1, \dots, w^k = (t^k, z^k)$  satisfies

$$t^k \ge \beta_k \bar{t}. \tag{4.3.6}$$

Furthermore, if  $w^k$  is not a stationary point of (4.2.25), then

$$t^k > 0.$$

**Proof.** We prove this lemma by induction. From the choices of  $t^0$  and  $\beta_0$  in Algorithm 4.2.1, it is obvious that (4.3.6) holds for k = 0. Suppose that for any integer l,  $w^l = (t^l, z^l)$  satisfies (4.3.6). Now we prove that  $w^{l+1} = (t^{l+1}, z^{l+1})$  satisfies (4.3.6) as well.

If the condition (4.2.32) is not satisfied for k = l, we have

$$\vec{d}^l(\lambda_l) = \tilde{d}^l_G(\lambda_l) = \Pi_W(w^l + \lambda_l d^l_G) - w^l, \ d^l_G = -\xi_l \nabla \Psi_h(w^l) + \beta_l \bar{w},$$

where  $\lambda_l$  is the accepted step-length at *l*-th iteration. It follows from Algorithm 4.2.1 that

$$(\bar{d}^{l}(\lambda_{l}))_{t} = \lambda_{l} \left[ -\xi_{l} \left( t + \nabla_{t} \bar{H}(w) \bar{H}_{h}(w) \right) + \beta(w^{l}) \bar{t} \right]$$
  

$$\geq -\lambda_{l} t^{l} + \lambda_{l} \beta(w^{l}) \bar{t} \quad (\text{ see } (4.2.29)),$$

where  $(\overline{d}^{l}(\lambda_{l}))_{t}$  is the first element of  $\overline{d}^{l}(\lambda_{l})$ . Then we have

$$t^{l+1} - \beta(w^{l+1})\overline{t} = t^l + (\overline{d}^l(\lambda_l))_t - \beta(w^{l+1})\overline{t}$$
  

$$\geq (1 - \lambda_l)t^l + \lambda_l\beta(w^l)\overline{t} - \beta(w^{l+1})\overline{t}$$
  

$$\geq (1 - \lambda_l)t^l + \lambda_l\beta(w^l)\overline{t} - \beta(w^l)\overline{t}$$
  

$$= (1 - \lambda_l)t^l - (1 - \lambda_l)\beta(w^l)\overline{t} \geq 0,$$

where the second and third inequalities are due to the monotonicity property of  $\beta(w^l)$ in Lemma 4.3.1 and  $t^l \ge \beta(w^l)\bar{t}$ .

If the condition (4.2.32) is satisfied for k = l, then we have

$$(\vec{d}^l(\lambda_l))_t = \left(\tau^*(\lambda_l)\vec{d}^l_G(\lambda_l) + (1-\tau^*(\lambda_l))\vec{d}^l_{tN}(\lambda_l)\right)_t.$$

By the similar way, we can obtain that  $t^{l+1} - \beta(w^{l+1})\bar{t} \ge 0$ .

Therefore, (4.3.6) holds for any nonnegative integer k. Furthermore, from (4.3.6) and that  $w^k$  is not a stationary point of (4.2.25),  $t^k > 0$  holds. We complete the proof.

**Theorem 4.3.1** Let  $\{w^k\} \subset W$  be a sequence generated by Algorithm 4.2.1. Then any accumulation point of  $\{w^k\}$  is a stationary point of (4.2.25).

**Proof.** Lemma 4.3.4 shows that if Algorithm 4.2.1 does not stop at a stationary point of (4.2.25), then  $t^k > 0$  for any k. This means that  $\Psi$  and  $\Psi_h$  are continuously differentiable at  $w^k$ . Remark of Lemma 4.3.2 means that for k sufficiently large, the condition (4.2.32) is always satisfied and  $\Psi_h(w) = \Psi(w)$  (see Remark (2) of Algorithm 4.2.1). Hence, by using a similar way to the proof of Theorem 4.1 [168], we can prove that the theorem holds.

In the rest of this section, we analyze the local convergence of Algorithm 4.2.1. We make the following standard assumption:

(E1) Let  $w^* = (t^*, z^*) = (0, z^*)$  be an accumulation point of the sequence  $\{w^k\}$  generated by Algorithm 4.2.1. Suppose  $\lim_{k \in K} w^k = w^*$  for some subset  $K \subset \{1, 2, \cdots\}$ ,  $w^*$  is a solution of the system of equations (4.2.1) and  $\Phi$  is BD-regular at  $w^*$ .

BD-regularity can be satisfied without special difficulty. Before giving a sufficient condition for BD-regularity to hold, we need the following assumptions:

(E2) The vectors  $\nabla_x g(x, v^j), j = 1, \dots, p$  are linearly independent.

(E3) The matrix  $\nabla_x^T F(x, u, \mathbf{v})$  is positive definite, and for every j = 1, 2, ..., p, the matrix  $(\nabla_v^2 g(x, v^j))_M$  is negative definite whenever  $J_M(x, v^j) \neq \emptyset$ , where

$$J_M(x,v) = \{i \mid a_i < v_i + (\nabla_v g(x,v))_i < b_i\},\$$

 $(\nabla_v^2 g(x, v))_M$  is a principal square submatrix of  $\nabla_v^2 g(x, v)$ , which is determined by the columns and rows with the index  $i \in J_M(x, v)$ .

(E4) For every j = 1, 2, ..., p,  $\{i \mid v_i + (\nabla_v g(x, v))_i = a_i \text{ or } v_i + (\nabla_v g(x, v))_i = b_i\}$ is an empty set. In addition, for any  $(x, v) \in \mathbb{R}^n \times \mathbb{R}^m$ , we denote

$$J_L(x,v) = \{i \mid v_i + (\nabla_v g(x,v))_i < a_i\}, \ J_R(x,v) = \{i \mid b_i < v_i + (\nabla_v g(x,v))_i\}.$$

We now state and prove a lemma in the following.

Lemma 4.3.5 Let

$$T = \left(\begin{array}{ccc} A & B & DC \\ B^T & 0 & 0 \\ D^T & 0 & F \end{array}\right)$$

where  $A \in \Re^{p \times p}, B \in \Re^{p \times q}, C \in \Re^{r \times r}, D \in \Re^{p \times r}$  and  $F \in \Re^{r \times r}$ . Suppose that A and  $C^T F$  are positive definite and negative semidefinite, respectively. If the column rank of B and F are q and r, respectively, then T is nonsingular.

**Proof.** Let Td = 0, where  $d = (d_1, d_2, d_3)$  is a suitable partitioned vector. Then

$$Ad_1 + Bd_2 + DCd_3 = 0 (4.3.7)$$

$$B^T d_1 = 0 (4.3.8)$$

$$D^T d_1 + F d_3 = 0 (4.3.9)$$

Multiplication (4.3.7) with  $d_1^T$  yields

$$d_1^T A d_1 + d_1^T B d_2 + d_1^T D C d_3 = 0,$$

which, together with (4.3.8) and (4.3.9), implies

$$d_1^T A d_1 + d_3^T (-C^T F) d_3 = 0.$$

From the property of A and  $C^T F$ , we have that  $d_1 = 0$ . Then it follows from (4.3.9) and the property of F that  $d_3 = 0$ . Because of (4.3.7) and the property of B,  $d_2 = 0$  holds. The proof is complete.

**Theorem 4.3.2** Suppose that  $w^* = (t^*, z^*) = (t^*, x^*, u^*, \mathbf{v}^*, y^*)$  is a solution of (4.2.1) and satisfies (E2)-(E4). Then  $\Phi$  is BD-regular at  $w^*$ .

**Proof.** Without loss of generality, by (E4), we assume

$$J_L(x^*, v^{j*}) = \{1, 2, \cdots, k_1^j\},\$$
  
$$J_M(x^*, v^{j*}) = \{k_1^j + 1, \cdots, k_2^j\},\$$
  
$$J_R(x^*, v^{j*}) = \{k_2^j + 1, \cdots, m\},\$$

where  $1 \le k_1^j \le k_2^j \le m$ . Because  $w^* = (t^*, z^*)$  is a solution of (4.2.1),  $t^* = 0$ . Moreover, we have, by  $\phi(0, x^*, v^{j^*}) = 0$ , that

$$v^{j*} - \operatorname{mid}(a, b, v^{j*} + \nabla_v g(x^*, v^{j*})) = 0, \quad j = 1, \cdots, p.$$
 (4.3.10)

By (4.3.10) and the definition of the mid function, we have that for  $j = 1, \dots, p$  and  $i \in J_M(x^*, v^{j^*})$ ,

$$(\nabla_{v^j} g(x^*, v^{j^*}))_i = 0. \tag{4.3.11}$$

By direct computation, we obtain that for any  $Q \in \partial_B \Phi(w^*)$ ,

$$Q = \begin{pmatrix} 1 & 0_{1 \times n} & 0_{1 \times p} & 0_{1 \times m} & \cdots & 0_{1 \times m} & 0 \\ 0_{n \times 1} & \nabla_x^T F(x^*, u^*, \mathbf{v}^*) & \nabla_x \mathbf{g}(x^*, \mathbf{v}^*) & u_1^* D_1 & \cdots & u_p^* D_p & 0_{n \times 1} \\ 0 & \nabla_x^T g(x^*, v^{1*}) & 0_{1 \times p} & \nabla_{v_1}^T g(x^*, v^{1*}) & \cdots & 0_{1 \times m} & 0 \\ 0 & \nabla_x^T g(x^*, v^{2*}) & 0_{1 \times p} & 0_{1 \times m} & \cdots & 0_{1 \times m} & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \nabla_x^T g(x^*, v^{p*}) & 0_{1 \times p} & 0_{1 \times m} & \cdots & \nabla_{v_p}^T g(x^*, v^{p*}) & 0 \\ Q_1 & C_1 D_1^T & 0_{m \times p} & E_1 + C_1 F_1 & \cdots & 0_{m \times m} & 0_{m \times 1} \\ Q_2 & C_2 D_2^T & 0_{m \times p} & 0_{m \times m} & \cdots & 0_{m \times m} & 0_{m \times 1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ Q_p & C_p D_p^T & 0_{m \times p} & 0_{m \times m} & \cdots & E_p + C_p F_p & 0_{m \times 1} \\ U_1 & U_2 & 0_{1 \times p} & 0_{1 \times m} & \cdots & 0_{1 \times m} & 1 \end{pmatrix}$$

$$(4.3.12)$$

where  $U_1 \in \partial_t G(0, x^*)$ ,  $U_2 \in \partial_x G(0, x^*)$  and for  $j = 1, \dots, p$ ,

$$Q_{j} \in \partial_{t} \bar{\phi}(0, x^{*}, v^{j^{*}}), \ D_{j} = \nabla_{v^{j}}^{T} (\nabla_{x} g(x^{*}, v^{j^{*}})), \ F_{j} = \nabla_{v^{j}}^{T} (\nabla_{v^{j}} g(x^{*}, v^{j^{*}})),$$
$$C_{j} = \operatorname{diag}(0_{j_{1}}, -I_{j_{2}}, 0_{j_{3}}), \ E_{j} = \operatorname{diag}(I_{j_{1}}, 0_{j_{2}}, I_{j_{3}}).$$
(4.3.13)

where  $0_{j_1}, 0_{j_2}, 0_{j_3}$  are zero square matrices with  $k_1^j$ ,  $(k_2^j - k_1^j)$  and  $(m - k_2^j)$  order respectively,  $I_{j_1}, I_{j_2}, I_{j_3}$  are identity matrices with  $k_1^j$ ,  $(k_2^j - k_1^j)$ , and  $(m - k_2^j)$  order respectively.

By (4.3.12), it is easy to see that the matrix Q is also nonsingular as the matrix

$$\tilde{Q} = \begin{pmatrix} \nabla_x^T F(x^*, u^*, \mathbf{v}^*) & \nabla_x \mathbf{g}(x^*, \mathbf{v}^*) & u_1^* D_1 & \cdots & u_p^* D_p \\ \nabla_x^T g(x^*, v^{1*}) & 0_{1 \times p} & \nabla_{v^1}^T g(x^*, v^{1*}) & \cdots & 0_{1 \times m} \\ \nabla_x^T g(x^*, v^{2*}) & 0_{1 \times p} & 0_{1 \times m} & \cdots & 0_{1 \times m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \nabla_x^T g(x^*, v^{p*}) & 0_{1 \times p} & 0_{1 \times m} & \cdots & \nabla_{v^p}^T g(x^*, v^{p*}) \\ C_1 D_1^T & 0_{m \times p} & E_1 + C_1 F_1 & \cdots & 0_{m \times m} \\ C_2 D_2^T & 0_{m \times p} & 0_{m \times m} & \cdots & 0_{m \times m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_p D_p^T & 0_{m \times p} & 0_{m \times m} & \cdots & E_p + C_p F_p \end{pmatrix}$$

We denote by  $(D_j)_{ML}$  a submatrix of  $D_j$  constituted of the columns with the index  $i \in J_M(x, v^j)$ , and by  $(F_j)_M$  a principal square submatrix of  $F_j$ , which is determined by the columns and rows with the index  $i \in J_M(x, v^j)$ . Then from special forms of  $C_j$  and  $E_j$  we have

$$C_j D_j^T = \begin{pmatrix} 0 \\ -(D_j)_{ML}^T \\ 0 \end{pmatrix}, \ E_j + C_j F_j = \begin{pmatrix} I_{j_1} & 0 & 0 \\ * & -(F_j)_M & * \\ 0 & 0 & I_{j_2} \end{pmatrix}$$

where two \* are some proper partitioned matrices. Hence the nonzero elements of  $\nabla_{v^j}^T g(x^*, v^{j*})$  and the matrix \* are deleted by the some proper row transformations. Hence it is not difficult to know that the matrix  $\tilde{Q}$  is also nonsingular as the

$$Q^* = \begin{pmatrix} \nabla_x^T F(x^*, u^*, \mathbf{v}^*) & \nabla_x \mathbf{g}(x^*, \mathbf{v}^*) & DU \\ \nabla_x^T \mathbf{g}(x^*, \mathbf{v}^*) & 0 & 0 \\ D^T & 0 & F \end{pmatrix}, \qquad (4.3.14)$$

where  $D = ((D_1)_{ML}, \dots, (D_p)_{ML})$ ,  $F = \text{diag}((F_1)_M, \dots, (F_p)_M)$ ,  $U = \text{diag}(u_1^*I_1, \dots, u_p^*I_p)$ and  $I_j, j = 1, \dots, p$ , are some proper identity matrices. It is clear that  $U^T F$  is negative definite, from (E2) and (E3) it follows that all other conditions in Lemma 4.3.5 are satisfied. Hence from Lemma 4.3.5 we know that  $Q^*$  is nonsingular and complete the proof.

The following lemma is the same as Lemma 3.3.2, its proof is omitted.

**Lemma 4.3.6** There exist positive constants  $\kappa$  and  $\epsilon$  such that for every  $w^k$  satisfying  $||w^k - w^*|| \leq \epsilon$ , (i)  $\nabla \Phi(w^k)$  is nonsingular and satisfies

$$\|\nabla\Phi(w^k)\| \le \kappa.$$

(ii)

$$\|\Phi(w^k)\| = \sqrt{2}\Psi(w^k)^{\frac{1}{2}} = O(\|w^k - w^*\|).$$

**Lemma 4.3.7** Let  $\{w^k\}$  be a sequence generated by Algorithm 4.2.1. Then for all  $k \in K$  sufficiently large, we have

$$\beta(w^k) = O(\Psi(w^k)) = O(\|w^k - w^*\|^2); \qquad (4.3.15)$$

and

$$w^{k} + \lambda d_{tN}^{k} = (1 - \lambda)w^{k} + \lambda w^{*} + \lambda o(\Psi(w^{k})^{\frac{1}{2}}).$$
(4.3.16)

for any  $\lambda \in (0, 1]$ 

**Proof.** From the definition of  $\beta(w^k)$  (see (4.2.28)), the choice of  $\gamma_k$  (see (4.2.30)), the projection property and Lemma 4.3.6, it follows that for  $w^k$  sufficiently close to  $w^*$ ,  $\Psi_h(w^k) = \Psi(w^k)$ ,

$$\beta(w^k) \le \alpha \|\bar{d}_G^k(1)\|^2 \le \alpha \gamma_k^2 \|\nabla \Psi(w^k)\|^2 \le \alpha \eta \Psi(w^k) = \frac{\alpha \eta}{2} \|\Phi(w^k)\|^2 = O(\|w^k - w^*\|^2).$$

This shows (4.3.15) holds. From (4.2.33), (4.3.2), (4.3.3) and Lemma 4.3.3, we have that for  $w^k$  sufficiently close to  $w^*$ ,

$$\Phi(w) - \beta_k \bar{w} - \theta_k + \nabla^T \Phi(w^k) d_{tN}^k = 0, \ \|\theta_k\| = \|r_k\| = o\left(\Psi_h(w^k)\right) = o\left(\Psi(w^k)\right), \ (4.3.17)$$

which imply

$$\begin{split} w^{k} + \lambda d_{tN}^{k} &= w^{k} + \lambda \nabla^{T} \Phi(w^{k})^{-1} [-\Phi(w^{k}) + \beta(w^{k})\bar{w} + \theta_{k}] \\ &= w^{k} - \lambda \nabla^{T} \Phi(w^{k})^{-1} [\Phi(w^{k}) - \Phi(w^{*}) - \nabla^{T} \Phi(w^{k})(w^{k} - w^{*})] \\ &- \lambda (w^{k} - w^{*}) + \lambda \nabla^{T} \Phi(w^{k})^{-1} \left(\beta(w^{k})\bar{w} + \theta_{k}\right) \\ &= (1 - \lambda)w^{k} + \lambda w^{*} + \lambda o(||w^{k} - w^{*}||) + \lambda O(\Psi(w^{k})) \\ &= (1 - \lambda)w^{k} + \lambda w^{*} + \lambda o(\Psi(w^{k})^{\frac{1}{2}}), \end{split}$$

where the third equality is due to the semismoothness of  $\Phi$ , (4.3.15) and (4.3.17). The proof is complete  $\blacksquare$ 

By a similar way to the proof of Theorem 3.3.4, we obtain the following theorem.

**Theorem 4.3.3** Suppose that  $\{w^k\}$  is a sequence generalized by Algorithm 4.2.1 and  $w^*$  is a point satisfying (E1). Then the whole sequence  $\{w^k\}$  superlinearly converges to  $w^*$ .

# 4.4 Implementation and Numerical Tests

In this section, we discuss some detailed implementation of Algorithm 4.2.1 and give some numerical results for medium-sized and large scale SIP problems.

#### 4.4.1 Implementation of Algorithm 4.2.1

In order to decrease the number of inner iterations, we use another line search technique if only projected gradient direction is search direction. In this case, the initial value of  $\lambda$  is set to

$$\min\left\{1, \frac{1}{\left\|d_{G}^{k}\right\|}, \frac{0.2\Psi_{h}(w^{k})}{-\nabla\Psi_{h}(w^{k})^{T}d_{G}^{k}}, \frac{t^{k}}{\left|t^{k}+\nabla_{t}\bar{H}(w^{k})\bar{H}_{h}(w^{k})\right|}\right\}$$

and  $\lambda$  is updated by quadratic interpolation technique.

In order to guarantee the numerical stability, we determine the truncated solution  $(\triangle x_k, \triangle u_k, \triangle \bar{\mathbf{v}}_k)$  by solving the problem

$$\left(\frac{M_k^T M_k}{\|M_k\|^2} + \omega I\right) \begin{pmatrix} \bigtriangleup x_k \\ \bigtriangleup u_k \\ \bigtriangleup \bar{\mathbf{v}}_k \end{pmatrix} = \frac{M_k^T}{\|M_k\|^2} \begin{pmatrix} b_{k1} \\ b_{k2} \\ b_{k3} \end{pmatrix}$$
(4.4.1)

instead of the problem (4.2.33), where

$$M_k = \begin{pmatrix} A_k & G_k & B_k \\ \tilde{G}_k & 0 & \tilde{E}_k \\ Z_k & & \tilde{Z}_k \end{pmatrix},$$

 $\omega$  is a damping factor. If  $M_k$  is nonsingular and  $\omega = 0$ , then the problem (4.4.1) is equivalent to the problem (4.2.33). At first, (4.4.1) is solved with  $\omega = 0$ . If the truncated solution does not generate a good descent direction, then (4.4.1) is solved with  $\omega = \frac{1}{(n+p+mp_k)^2}$  in next iteration. In Algorithm 4.2.1, we choose the suitable values of parameters (see Step 0) by

$$\eta = 0.9, \ \rho = 0.5, \ \sigma = 0.0005, \ \alpha = 0.5, \ \bar{t} = 0.9, \ p_1 = 1.0e - 10, \ p_2 = 2.1,$$

and

$$h_j = \max\left\{2.5, \ 10^{-3} \left|\Psi(w^0)\right|, \ j = 1, 2, \cdots, \tilde{n}\right\}.$$

The starting points  $t^0$ ,  $u^0$  and  $s^0$  for all problems are set  $t^0 = \bar{t}$ ,  $u^0 = 0.05 e$ ,  $s^0 = 0.5$ , where e is the vector of ones.

#### 4.4.2 Numerical Results

Now we discuss the implementation of Algorithm 4.2.1, which has been implemented in FORTRAN 77. All calculation within the driving programs, test problems and optimization code are carried out in double precision. The problem is solved on a personal computer (Pentium III 1133 MHz, 256 MB memory).

Although a lot of large SIP type problems arise from optimal control and approximation theory, it is difficult to find large-scale SIP problems in the literature suitable for use as test problems. In order to evaluate for large scale SIP problems, we enlarge 3 test problems where the first problem is the same as Problem 3.3.5 and the second problem is from [78], another is generated from optimal control problem. We list the three SIP problem in the following.

#### Problem 4.4.1

$$f(x) = \frac{1}{2}x^T x, \quad g(x,v) = 3 + 4.5\sin(4.7\pi(v-1.23)/8) - \sum_{i=1}^n x_i v^{i-1},$$
$$V = [0,b], \quad p = 1. \quad \text{if} \quad n \le 60, b = 100; \text{ otherwise } b = 1.$$

Problem 4.4.2

$$f(x) = \int_{0}^{1} \left( \sum_{i=1}^{n} x_{i} t^{i-1} - \tan t \right)^{2} dt, \quad g(x,v) = \tan v - \sum_{i=1}^{n} x_{i} v^{i-1}, V = [0,1], \quad p = 1.$$

Problem 4.4.3

$$\min \qquad p(g)h^T h \\ s.t. \quad g^T A(v_1, v_2)h \le r(v_1, v_2),$$

where  $v_1 \in [-\pi, \pi]$ ,  $v_2 \in [0, 2\pi]$ ,  $p(g) = g^T B g$ ,  $h \in \Re^{n_1}$ ,  $g \in \Re^{n_2}$ ,  $B \in \Re^{n_2 \times n_2}$ ,  $A(v_1, v_2) \in \Re^{n_2 \times n_1}$ ,  $n_2 = n_1$  and

 $A(v_1, v_2) =$ 

1	$\sin bv_2$	$\cos cv_1$						·	
$\sin a v_1$	1	$\sin bv_2$	$\cos cv_1$						
$\cos dv_2$	$\sin a v_1$	1	$\sin bv_2$	$\cos cv_1$					
	·	·	·	·	·				
		·	·	·	·	·			
			·	·	·	·	۰.		
				$\cos dv_2$	$\sin a v_1$	1	$\sin bv_2$	$\cos cv_1$	
					$\cos dv_2$	$\sin a v_1$	1	$\sin bv_2$	
						$\cos dv_2$	$\sin a v_1$	1	)

We use Algorithm 4.2.1 to solve these problems where the termination condition is that the  $l_2$  norm of the projected gradient,  $\|\bar{d}_G^k(1)\|$  is reduced below  $10^{-5}$ . The dimensions (n) of these problems are chosen by 10, 20, 40, 60, 80, 100, 200, 400, 1000 and 2000. The results of the test are given in Tables 4.1, 4.2 and 4.3. The number of iteration (ITK), the norm of projected gradient ( $\|\bar{d}_G^k(1)\|$ ) and the merit function value  $\Psi(w^k)$  and the objective function value  $f(x^k)$  are shown in these tables.

Table 4.1 shows that Algorithm 4.2.1 performs very well for solving Problem 4.4.1 with the different dimension. There is some difference among different dimensions. When  $n \ge 100$ , there is a slight increase in iteration number.

Problem 4.4.2 is dense, i.e., its Hessian of Lagrangian function  $\nabla_x^T F(x, u, \mathbf{v})$  is not sparse. Although the Hessian can be not stored for its special structure, the computation in each iteration can be not decreased. Hence, Algorithm 4.2.1 is used for solving

n	ITK	$\left\  \vec{d}_G^k(1) \right\ $	$\Psi(w^k)$	$f(x^k)$
10	64	8.71e-11	1.44e-17	0.08246
20	55	6.36e-6	2.31e-7	0.04408
40	16	8.36e-11	1.17e-15	3.1826
60	20	1.41e-11	2.023e-17	5.5918
100	183	2.65e-7	2.75e-12	2.3862
400	107	8.448e-6	1.47e-9	4.605
1000	98	5.05e-6	8.52e-9	8.2726
2000	252	7.77e-6	2.51e-4	16.96

Table 4.1: Test Result of Problem 4.4.1 for Algorithm 4.2.1

Problem 4.4.2, the dimensions of which range from 10 to 200. Table 4.2 shows that Algorithm 4.2.1 performs well for solving some medium dense SIP problems.

n	ITK	$\left  \vec{d}_{G}^{k}(1) \right $	$\Psi(w^k)$	$f(x^k)$
10	69	4.29e-6	1.07e-8	0.3147
20	66	4.68e-6	1.98e-9	0.6717
40	80	8.32e-6	3.573e-10	0.5803
80	84	5.68e-6	1.45e-10	1.424
100	85	7.62e-6	7.28e-11	1.069
200	75	5.35e-6	1.67e-10	1.323

Table 4.2: Test Result of Problem 4.4.2 for Algorithm 4.2.1

Problem 4.4.3 is a somewhat complicated SIP problem which often arises from optimal control field. In this problem,  $v \in \Re^2$ , while in problems 4.4.1 and 4.4.2,  $v \in \Re$ . Its Hessian of Lagrangian function is sparse, however the computation of elements is not simple due to some trigonometric functions. Numerical results of this problem is given in Table 4.3 which show that Algorithm 4.2.1 can solve some large scale sparse SIP problems. It is interesting that the outer iteration number does not increase and inner iteration numbers decrease as the dimensions increase.

			<u> </u>	
n	ITK	$\left  \vec{d}_G^k(1) \right $	$\Psi(w^k)$	$f(x^k)$
20	225	6.18e-7	5.97e-12	18.04
60	212	2.53e-7	1.54e-12	21.83
100	258	3.29e-6	3.69e-10	20.36
200	170	6.13e-6	1.26e-9	17.06
600	188	8.25e-6	2.23e-9	13.74
1000	135	5.58e-6	1.48e-6	13.85
2000	151	6.95e-6	4.72e-9	13.85

 Table 4.3: Test Result of Problem 4.4.3 for Algorithm 4.2.1

### 4.5 Some Comments

Although the development of the code for Algorithm 4.2.1 is still at its primary stage, the numerical results have indicated that Algorithm 4.2.1 is capable of processing large scale SIP problems. However, there are some issues which may be addressed in further research.

Because "large scale" here only refers to the decision variables, it is hoped that an improved version of Algorithm 4.2.1 may also be capable of handling high dimensional index sets. In addition, our method works on the KKT system of SIP, i.e. it does not minimize the original objective function f. Sometimes this may limit the applicability of this method to a special class of SIP problems.

By Algorithm 4.2.1 we can obtain stationary points of (4.2.25), it is possible that some of them may not be stationary points of (3.1.1). If V in (3.1.1) is a nonpolyhedral index set, then our method cannot be used directly.

We hope that with further research more efficient methods can be obtained for solving general SIP problem with many decision variables and high dimensional index sets.

# Chapter 5

# A Smoothing Implicit Programming Approach for Solving a Class of Stochastic Generalized Semi-Infinite Programming Problems

## 5.1 Introduction

A generalized semi-infinite programming (GSIP) problem is a constrained optimization problem in which the constraints are given by a possibly infinite index set that depends upon the decision variable x:

$$\min_{x} \quad f(x)$$
s. t.  $g(x, u) \le 0, \forall u \in T(x),$ 

$$(5.1.1)$$

where  $T(x) = \{ u \in \Re^r \mid h(x, u) \leq 0 \}$ . Here,  $f : \Re^n \to \Re, g : \Re^n \times \Re^r \to \Re, h : \Re^n \times \Re^r \to \Re^J, T : \Re^n \to 2^{\Re^r}$ , and  $2^{\Re^r}$  is the set of all subsets in  $\Re^r$ .

When the set-valued mapping T is constant, the GSIP problem reduces to a standard semi-infinite programming problem and will be abbreviated by SIP. Moreover, if T is a finite set, then SIP reduces to an ordinary nonlinear programming problem.

Recently, the GSIP problem becomes an active research topic in applied mathematics, as it arises in various fields of engineering such as the design problem, the problem of maneuverability of robots, and the reverse Chebyshev approximation problem, see, e.g., [56, 71, 93]. The first-order and second-order optimality conditions for the GSIP problem are studied in [72, 86, 148, 160]. Some numerical aspects of the GSIP problem are discussed in [162, 163].

Stochastic programming is another important branch of mathematical programming in which optimal decisions are sought under uncertainty. Modeling the uncertainty by random objects may lead to diverse stochastic programming problems. Various numerical methods for solving stochastic programming have been studied extensively, see [8,9,182].

In this chapter, we consider the following stochastic version of the GSIP problem (5.1.1):

$$\min_{x} E_{\omega}[f(x,\omega)]$$
s. t.  $g(x,u,\omega) \le 0$ , (5.1.2)  
 $u \in T(x,\omega), \ \omega \in \Omega, \ a.s.,$ 

where  $\Omega$  is a sample space,  $T(x, \omega) = \{u \in \Re^r \mid h(x, u, \omega) \leq 0\}$  is a constraint index set correlated with a decision variable x and a random variable  $\omega \in \Omega$ , the abbreviation a.s. means that the constraints hold almost surely, i.e., for all  $\omega \in \Omega$  except for a set with zero probability. We assume that  $f : \Re^n \times \Omega \to \Re$ ,  $g : \Re^n \times \Re^r \times \Omega \to \Re$ ,  $h : \Re^n \times \Re^r \times \Omega \to \Re^J$  are continuous,  $T : \Re^n \times \Omega \to 2^{\Re^r}$  and  $\Omega$  is a compact set in  $\Re^s$ . We call problem (5.1.2) a stochastic generalized semi-infinite programming (SGSIP) problem. Obviously, if  $\Omega$  is a singleton, then the problem (5.1.2) reduces to an ordinary GSIP problem. For each fixed  $\omega \in \Omega$ , the problem (5.1.2) is a GSIP problem, which can be reformulated as

$$\min_{x} \quad E_{\omega}[f(x,\omega)] 
s. t. \quad v(x,\omega) \le 0, \ \omega \in \Omega, a.s.,$$
(5.1.3)

where  $v(x, \omega)$  is defined as

$$v(x,\omega) = \sup_{u} \{ g(x,u,\omega) \mid u \in T(x,\omega) \}.$$

In this chapter, we apply the expected value approach to the constraints of (5.1.3) and propose a deterministic version of SGSIP problem as follows:

$$\min_{x} \quad E_{\omega}[f(x,\omega)] 
s. t. \quad E_{\omega}[v(x,\omega)] \le 0.$$
(5.1.4)

The expected value approach has been studied for stochastic variational inequality problems by Gürkan, Özge and Robinson [59]. The GSIP problem is a hard problem with an infinite constraint index set that may vary since it is correlated with decision variable x. Presence of an additional random variable makes the SGSIP problem even harder to solve than the GSIP problem.

Recently Stein and Still [161] studied interior point techniques for solving the GSIP problem. Under the reduction assumption (the LICQ holds, and both the strict complementary slackness (SCS) condition and the second-order sufficiency condition are valid), Stein and Still presented a similar algorithm for the GSIP problem and proved the convergence of the algorithm to Fritz John points and global optimal solutions. The main difference between the present paper and [161] is that here also a deterministic version of a stochastic GSIP model is presented and that the techniques for the proofs are completely different. Moreover, our approach does not use the SCS condition (in the parametric programming problem  $Q(x, \omega)$  defined later on).

The rest of this chapter is organized as follows. In Section 5.2, we reformulate problem (5.1.4) as a mathematical programming problem with complementarity constraints. In Section 5.3, we establish some properties of certain parametric smoothing approximations for the reformulated problem. In Section 5.4, we present global convergence analysis of a smoothing implicit programming algorithm for solving the problem with finite discrete distribution. Some remarks are given in Section 5.5.

# 5.2 A New Reformulation

In this section, we present a new reformulation of problem (5.1.4). Our main idea is to regard (5.1.4) as a two-stage optimization problem and use the first order optimality condition of the second stage optimization problem to deal with the constraints of (5.1.4). **Assumption F1.** For any  $x \in \Re^n$  and  $\omega \in \Omega$ ,  $g(x, \cdot, \omega)$  is twice continuously differentiable and pseudo-concave,  $h(x, \cdot, \omega)$  is twice continuously differentiable and  $y^T h(x, \cdot, \omega)$  is quasi-convex for any  $y \in \Re^J_+$ .

For any  $(x, \omega) \in \Re^n \times \Omega$ , we define a parametric programming problem

$$Q(x,\omega): \max_{u} g(x,u,\omega)$$
  
s.t.  $u \in T(x,\omega).$ 

The first-order optimality conditions for problem  $Q(x, \omega)$  are given by

$$\nabla_{u}g(x, u, \omega) - \nabla_{u}h(x, u, \omega)y = 0,$$
  

$$y^{T}h(x, u, \omega) = 0,$$
  

$$h(x, u, \omega) \leq 0,$$
  

$$y \geq 0.$$
  
(5.2.5)

**Definition 5.2.1** We say that the linear independence constraint qualification (LICQ) is satisfied at  $\bar{u}$  for problem  $Q(x, \omega)$ , if the vectors

$$\nabla_u h_j(x, \bar{u}, \omega), \quad j \in \mathcal{I}_h(x, \bar{u}, \omega)$$

are linearly independent, where  $\mathcal{I}_h(x, \bar{u}, \omega)$  is the index set of active constraints

$$\mathcal{I}_h(x,\bar{u},\omega) = \{ j \mid h_j(x,\bar{u},\omega) = 0 \}.$$

We say that the Mangasarian-Fromovitz constraint qualification (MFCQ) [148] is satisfied at  $\bar{u}$  for problem  $Q(x, \omega)$ , if there exists a vector  $\gamma_0 \in \Re^r$  such that

$$\nabla_u h_j(x,\bar{u},\omega)^T \gamma_0 < 0, \ j \in \mathcal{I}_h(x,\bar{u},\omega).$$

Assumption F2. For any  $x \in \Re^n$  and  $\omega \in \Omega$ , problem  $Q(x, \omega)$  has a unique solution, which we denote  $u(x, \omega)$ . Moreover, the MFCQ is satisfied at  $u(x, \omega)$  for problem  $Q(x, \omega)$ .

Under Assumptions F1 and F2, we show that problem (5.1.4) is equivalent to the following problem.

$$\begin{split} \tilde{P}: & \min_{x} \quad E_{\omega}[f(x,\omega)] \\ & \text{s. t. } \quad E_{\omega}[g(x,u(x,\omega),\omega)] \leq 0, \end{split}$$

where  $u(x, \omega)$ , together with a vector  $y(x, \omega) \in \mathbb{R}^J$ , satisfies the following first-order optimality conditions for problem  $Q(x, \omega)$ :

$$\nabla_u g(x, u(x, \omega), \omega) - \nabla_u h(x, u(x, \omega), \omega) y(x, \omega) = 0,$$
  

$$\min(y(x, \omega), -h(x, u(x, \omega), \omega)) = 0.$$
(5.2.6)

**Lemma 5.2.1** Suppose that Assumptions F1 and F2 hold. Then,  $\tilde{x}$  is a feasible solution of problem (5.1.4) if and only if  $\tilde{x}$  is a feasible solution of problem  $\tilde{P}$ .

**Proof.** Let  $\tilde{x}$  be a feasible solution of problem (5.1.4), that is,  $E_{\omega}[g(\tilde{x}, u(\tilde{x}, \omega), \omega)] \leq 0$ , where  $u(\tilde{x}, \omega)$  is the unique solution of  $Q(\tilde{x}, \omega)$ . By Assumption F2, for every  $\omega \in \Omega$ , there exists a vector y such that  $(u(\tilde{x}, \omega), y)$  satisfies the first-order optimality conditions of  $Q(\tilde{x}, \omega)$  at  $u(\tilde{x}, \omega)$ , which implies that  $\tilde{x}$  is a feasible solution of problem  $\tilde{P}$ .

Conversely, let  $\tilde{x}$  be a feasible solution of problem  $\tilde{P}$ , that is, there exists a pair  $(u(\tilde{x}, \omega), y)$  such that  $(\tilde{x}, u(\tilde{x}, \omega), y)$  satisfies the constraints of  $\tilde{P}$ . From Assumption F1, the first-order optimality conditions imply

$$g(\tilde{x}, u(\tilde{x}, \omega), \omega) = v(\tilde{x}, \omega).$$

Hence,  $E_{\omega}[v(\tilde{x}, \omega)] \leq 0$ , that is,  $\tilde{x}$  is a feasible solution of (5.1.4). The proof is complete.

From Lemma 5.2.1, we readily obtain the following theorem. The proof is omitted.

**Theorem 5.2.1** Suppose that Assumptions F1 and F2 hold. Then  $\tilde{x}$  is a global (local) optimal solution of problem (5.1.4) if and only if  $\tilde{x}$  is a global (local) optimal solution of problem  $\tilde{P}$ .

# 5.3 Smoothing Approximation for $\tilde{P}$

In this section, we study a smoothing approach for solving problem  $\tilde{P}$ .

Let  $\varepsilon \in \Re_+$  be a smoothing parameter. Define a function  $\phi_{\varepsilon} : \Re^2 \to \Re$  by

$$\phi_{\varepsilon}(s,t) = \frac{1}{2}(s+t-\sqrt{(s-t)^2+4\varepsilon^2})$$

which is called the Chen-Harker-Kanzow-Smale smoothing function for the function  $\min(s, t)$ .

**Proposition 5.3.1** [87] For any  $\varepsilon \in \Re_+$ , we have

(i) 
$$|\phi_{\varepsilon}(s,t) - \min(s,t)| \leq \varepsilon$$
.

- (*ii*)  $\phi_{\varepsilon}(s,t) = 0 \iff s \ge 0, t \ge 0, st = \varepsilon^2$ .
- (iii)  $\phi_{\varepsilon}(s,t)$  is a  $C^{\infty}$  function of (s,t) for a fixed  $\varepsilon > 0$ .

Let us define the function  $\Psi: \Re_+ \times \Re^n \times \Re^r \times \Re^J \times \Omega \to \Re^{r+J}$  by

$$\Psi(\varepsilon, x, u, y, \omega) = \begin{pmatrix} \nabla_u g(x, u, \omega) - \nabla_u h(x, u, \omega) y \\ \phi_{\varepsilon}(y_1, -h_1(x, u, \omega)) \\ \vdots \\ \phi_{\varepsilon}(y_J, -h_J(x, u, \omega)) \end{pmatrix}.$$

Then, a parametric smooth approximation to problem  $\tilde{P}$  can be formulated as

$$\begin{split} \tilde{P}(\varepsilon,\delta) : & \min_{x} \quad E_{\omega}[f(x,\omega)] \\ & \text{s. t. } \quad E_{\omega}[g(x,u(\varepsilon,x,\omega),\omega)] \leq \delta, \end{split}$$

where  $\varepsilon$ ,  $\delta > 0$  are parameters, and  $u(\varepsilon, x, \omega)$ , together with a vector  $y(\varepsilon, x, \omega) \in \Re^J$ , satisfies

$$\Psi(\varepsilon, x, u(\varepsilon, x, \omega), y(\varepsilon, x, \omega), \omega) = 0.$$

We denote the feasible regions of  $\tilde{P}(\varepsilon, \delta)$  and  $\tilde{P}$  by  $\mathcal{F}(\varepsilon, \delta)$  and  $\tilde{\mathcal{F}}$ , respectively. It is clear that if  $(\varepsilon, \delta) = 0$  then  $\tilde{P}(\varepsilon, \delta)$  coincides with  $\tilde{P}$ , and hence  $\mathcal{F}(\varepsilon, \delta)$  is identical to  $\tilde{\mathcal{F}}$ . In the next section, we will present an algorithm for solving problem  $\tilde{P}$  by solving a sequence of problems  $\tilde{P}(\varepsilon, \delta)$ . In the rest of this section, we concentrate on establishing some properties of  $\tilde{P}(\varepsilon, \delta)$ . To this end, we state two lemmas at first. Their proofs are omitted since they can be found in some text books on matrix analysis.

Lemma 5.3.1 Let

$$T = \left(\begin{array}{cc} A & B \\ B^T & C \end{array}\right),$$

where  $A \in \Re^{p \times p}$ ,  $B \in \Re^{p \times q}$ ,  $C \in \Re^{q \times q}$ ,  $p \le q$ . Then the following two statements are true:

(i) If A is negative semidefinite, C is positive definite and the row rank of B is p, then T is nonsingular.

(ii) If A is negative definite and C is positive definite, then T is nonsingular.

Lemma 5.3.2 Let

$$T = \left( \begin{array}{ccc} A & BC & D \\ B^T & I - C & O \\ D^T & O & O \end{array} \right),$$

where  $A \in \Re^{p \times p}$ ,  $B \in \Re^{p \times q}$ ,  $C \in \Re^{q \times q}$ ,  $D \in \Re^{p \times s}$ . If A and  $C^T - C^T C$  are negative definite and positive semidefinite, respectively, and the column rank of (B, D) is q + s, then T is nonsingular.

Using Lemmas 5.3.1 and 5.3.2, we can investigate the nonsingularity of (generalized) Jacobian of  $\Phi$  with respect to the variable (u, y), which plays an important role in the rest of this section and the convergence analysis of the algorithm presented in Section 5.4. We have the following results.

**Proposition 5.3.2** The function  $\Psi$  is locally Lipschitz and regular.

**Proof.** It is similar to Lemma 1 in [38].  $\blacksquare$ 

**Proposition 5.3.3** Let  $\Psi(\bar{\varepsilon}, \bar{x}, \bar{u}, \bar{y}, \bar{\omega}) = 0$ . Suppose that

$$\bar{A}(\bar{\omega}) = \nabla_{uu}^2 g(\bar{x}, \bar{u}, \bar{\omega}) - \sum_{j=1}^J \bar{y}_j \nabla_{uu}^2 h_j(\bar{x}, \bar{u}, \bar{\omega})$$

is negative definite. In addition, suppose that the LICQ is satisfied at  $\bar{u}$  for problem  $Q(\bar{x},\bar{\omega})$  if  $\bar{\varepsilon} = 0$ . Then all matrices in  $\partial_{(u,y)}\Psi(\bar{\varepsilon},\bar{x},\bar{u},\bar{y},\bar{\omega})$  are nonsingular.

**Proof.** We only show the conclusion in the case where  $\bar{\varepsilon} = 0$ . The conclusion in the case where  $\bar{\varepsilon} > 0$  can be shown similarly by using Lemma 5.3.1. We assume without

loss of generality that

$$\begin{cases} -h_j(\bar{x}, \bar{u}, \bar{\omega}) > \bar{y}_j, & j = 1, \cdots, J_1, \\ -h_j(\bar{x}, \bar{u}, \bar{\omega}) = \bar{y}_j, & j = J_1 + 1, \cdots, J_2, \\ -h_j(\bar{x}, \bar{u}, \bar{\omega}) < \bar{y}_j, & j = J_2 + 1, \cdots, J, \end{cases}$$

and write

$$\begin{cases} \bar{B}_1 = [\nabla_u h_1(\bar{x}, \bar{u}, \bar{\omega}), \cdots, \nabla_u h_{J_1}(\bar{x}, \bar{u}, \bar{\omega})], \\ \bar{B}_2 = [\nabla_u h_{J_1+1}(\bar{x}, \bar{u}, \bar{\omega}), \cdots, \nabla_u h_{J_2}(\bar{x}, \bar{u}, \bar{\omega})], \\ \bar{B}_3 = [\nabla_u h_{J_2+1}(\bar{x}, \bar{u}, \bar{\omega}), \cdots, \nabla_u h_J(\bar{x}, \bar{u}, \bar{\omega})], \\ \bar{C}_2 = \text{diag}[\bar{c}_1, \cdots, \bar{c}_{J_2-J_1}], \ 0 \le \bar{c}_j \le 1, \ j = 1, \cdots, J_2 - J_1. \end{cases}$$

Then, from the definition of the generalized Jacobian, it is not difficult to obtain, by direct calculation, that

$$\begin{aligned} \partial_{(u,y)} \Psi(0,\bar{x},\bar{u},\bar{y},\bar{\omega}) &= \\ \left\{ \begin{pmatrix} \bar{A} & -\bar{B}_1 & -\bar{B}_2 & -\bar{B}_3 \\ O & I & O & O \\ -\bar{C}_2 \bar{B}_2^T & O & I - \bar{C}_2 & O \\ -\bar{B}_3^T & O & O & O \end{pmatrix} \middle| \begin{array}{c} 0 \leq \bar{c}_j \leq 1, \\ j = 1, \cdots, J_2 - J_1 \\ \end{array} \right\}. \end{aligned}$$

It is easy to see that the matrix

$$\begin{pmatrix} \bar{A} & -\bar{B}_1 & -\bar{B}_2 & -\bar{B}_3 \\ O & I & O & O \\ -\bar{C}_2\bar{B}_2^T & O & I - \bar{C}_2 & O \\ -\bar{B}_3^T & O & O & O \end{pmatrix}$$

is also nonsingular as the matrix

$$\begin{pmatrix} \bar{A} & \bar{B}_2 & \bar{B}_3 \\ \bar{C}_2 \bar{B}_2^T & I - \bar{C}_2 & O \\ \bar{B}_3^T & O & O \end{pmatrix}.$$

It is obvious that  $\bar{C}_2 - \bar{C}_2^T \bar{C}_2$  is positive semidefinite, since  $0 \leq \bar{c}_j \leq 1$  for  $j = 1, \dots, J_2 - J_1$ . Hence, by the given conditions and Lemma 5.3.2, all matrices in  $\partial_{(u,y)} \Psi(0, \bar{x}, \bar{u}, \bar{y}, \bar{\omega})$  are nonsingular. The proof is complete.

**Remark 5.3.1** In [161], the authors proved that Jacobian of the first two equalities in (5.2.5) with respect to (u, y) is nonsingular under the strict complementarity slackness

(SCS) condition. Note that the SCS condition implies that the problem is smooth at (u, y). Proposition 5.3.3 proves the nonsingularity of the generalized Jacobian at (u, y) without the SCS condition.

We now focus our discussion on problem  $\tilde{P}$  where  $\Omega$  is a finite discrete set. Specifically, let  $\Omega = \{\omega_1, \omega_2, \dots, \omega_L\}$ . For every  $\omega_l, l = 1, 2, \dots, L$ , we denote

$$f^{l}(\cdot) = f(\cdot, \omega_{l}), \quad g^{l}(\cdot, \cdot) = g(\cdot, \cdot, \omega_{l}), \quad h^{l}(\cdot, \cdot) = h(\cdot, \cdot, \omega_{l}).$$

Throughout the rest of this chapter, we let  $u_l$  and  $y_l$  denote the variables  $u(x, \omega_l)$ and  $y(x, \omega_l)$  in  $\tilde{P}$ , respectively. Then, problem  $\tilde{P}$  can be rewritten as

$$\min_{x} \quad f(x) 
s. t. \quad G(x, \mathbf{u}) \le 0,$$
(5.3.7)

where 
$$f(x) = \sum_{l=1}^{L} p_l f^l(x), \ G(x, \mathbf{u}) = \sum_{l=1}^{L} p_l g^l(x, u_l), \ p_l \ge 0, \ \sum_{l=1}^{L} p_l = 1, \text{ and}$$
$$\mathbf{u} = \begin{pmatrix} u_1 \\ \vdots \\ u_L \end{pmatrix} \in \Re^{rL}.$$

Here  $u_l \in \Re^r$ , together with a vector  $y_l \in \Re^J$ , satisfies

$$\begin{cases} \nabla_{u_l} g^l(x, u_l) - \nabla_{u_l} h^l(x, u_l) y_l = 0, \\ \min(y_l, -h^l(x, u_l)) = 0, \end{cases}$$

which constitutes the first-order optimality conditions for the problem

$$Q_l(x): \max_{x} g^l(x, u)$$
  
s. t.  $h^l(x, u) \le 0.$ 

On the other hand, problem  $\tilde{P}(\varepsilon, \delta)$  can be rewritten as

$$\min_{x} \quad f(x)$$
s. t.  $G(x, \mathbf{u}(\varepsilon, x)) \le \delta,$ 

$$(5.3.8)$$

where  $G(x, \mathbf{u}(\varepsilon, x)) = \sum_{l=1}^{L} p_l g^l(x, u_l(\varepsilon, x))$  and  $\mathbf{u}(\varepsilon, x) = \begin{pmatrix} u_1(\varepsilon, x) \\ \vdots \\ u_L(\varepsilon, x) \end{pmatrix}.$  Here,  $u_l(\varepsilon, x)$ , together with  $y_l(\varepsilon, x)$ , satisfies the system

$$\Phi_{l}(\varepsilon, x, u_{l}(\varepsilon, x), y_{l}(\varepsilon, x)) := \begin{pmatrix} \nabla_{u_{l}}g^{l}(x, u_{l}(\varepsilon, x)) - \nabla_{u_{l}}h^{l}(x, u_{l}(\varepsilon, x))y_{l}(\varepsilon, x) \\ \phi_{\varepsilon}((y_{l}(\varepsilon, x))_{1}, -h_{1}^{l}(x, u_{l}(\varepsilon, x))) \\ \vdots \\ \phi_{\varepsilon}((y_{l}(\varepsilon, x))_{J}, -h_{J}^{l}(x, u_{l}(\varepsilon, x))) \end{pmatrix} = 0$$

$$(5.3.9)$$

for  $l = 1, 2, \dots, L$ . Moreover, we set

$$\mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_L \end{pmatrix} \in \Re^{JL}$$

and define a nonlinear operator  $\Phi: \Re_+ \times \Re^n \times \Re^{(r+J)L} \to \Re^{(r+J)L}$  by

$$\Phi(\varepsilon, x, \mathbf{u}, \mathbf{y}) = \begin{pmatrix} \Phi_1(\varepsilon, x, u_1, y_1) \\ \vdots \\ \Phi_L(\varepsilon, x, u_L, y_L) \end{pmatrix}.$$
 (5.3.10)

**Proposition 5.3.4** Let  $\bar{\varepsilon} \in \Re_+$  and  $\Phi(\bar{\varepsilon}, \bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) = 0$ . Suppose that

$$\bar{A}_{l} = \nabla^{2}_{u_{l}u_{l}}g^{l}(\bar{x}, \bar{u}_{l}) - \sum_{j=1}^{J}(\bar{y}_{l})_{j}\nabla^{2}_{u_{l}u_{l}}h^{l}_{j}(\bar{x}, \bar{u}_{l})$$

is negative definite for each  $l = 1, 2, \dots, L$ , and the LICQ is satisfied at  $\bar{u}_l$  for problem  $Q_l(\bar{x})$ . Then there exist a neighborhood  $(\bar{\varepsilon} - \varepsilon', \bar{\varepsilon} + \varepsilon') \times N(\bar{x})$  of  $(\bar{\varepsilon}, \bar{x})$  and a continuous function  $(\mathbf{u}(\cdot, \cdot), \mathbf{y}(\cdot, \cdot))$ :  $\{(\bar{\varepsilon} - \varepsilon', \bar{\varepsilon} + \varepsilon') \cap \Re_+\} \times N(\bar{x}) \to \Re^{(r+J)L}$  such that for each  $(\varepsilon, x) \in \{(\bar{\varepsilon} - \varepsilon', \bar{\varepsilon} + \varepsilon') \cap \Re_+\} \times N(\bar{x}),$ 

$$\Phi(\varepsilon, x, \mathbf{u}(\varepsilon, x), \mathbf{y}(\varepsilon, x)) = 0.$$
(5.3.11)

**Proof.** According to the corollary of Theorem 7.1.1 in [23], it suffices to check that the projection  $\Pi_{(\mathbf{u},\mathbf{y})}\partial\Phi(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}})$  of  $\partial\Phi(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}})$  on the space of the variable  $(\mathbf{u},\mathbf{y})$ is comprised of nonsingular matrices. We only show the conclusion in the case where  $\bar{\varepsilon} = 0$ . The conclusion in the case where  $\bar{\varepsilon} > 0$  can be shown similarly. By Proposition
2.6.2 (e) in [23] and the definition of the projection operator, we have

$$\Pi_{(\mathbf{u},\mathbf{y})}\partial\Phi(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}}) \subseteq \Pi_{(\mathbf{u},\mathbf{y})} \begin{pmatrix} \partial\Phi_{1}(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}}) \\ \vdots \\ \partial\Phi_{s}(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}}) \end{pmatrix} \\ \subseteq \begin{pmatrix} \Pi_{(\mathbf{u},\mathbf{y})}[\partial\Phi_{1}(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}})] \\ \vdots \\ \Pi_{(\mathbf{u},\mathbf{y})}[\partial\Phi_{s}(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}})] \end{pmatrix},$$
(5.3.12)

where s = (r + J)L. Recall that  $\Phi$  is regular by Proposition 5.3.2. It then follows from Proposition 2.3.15 in [23] that

$$\partial \Phi_i(\bar{\varepsilon}, \bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \subseteq \partial_{(\varepsilon, x)} \Phi_i(\bar{\varepsilon}, \bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \times \partial_{(\mathbf{u}, \mathbf{y})} \Phi_i(\bar{\varepsilon}, \bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}), \quad i = 1, \cdots, s,$$

and hence

$$\begin{pmatrix}
\Pi_{(\mathbf{u},\mathbf{y})}[\partial\Phi_{1}(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}})] \\
\vdots \\
\Pi_{(\mathbf{u},\mathbf{y})}[\partial\Phi_{s}(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}})]
\end{pmatrix} \subseteq \begin{pmatrix}
\partial_{(\mathbf{u},\mathbf{y})}\Phi_{1}(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}}) \\
\vdots \\
\partial_{(\mathbf{u},\mathbf{y})}\Phi_{s}(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}})
\end{pmatrix}.$$
(5.3.13)

On the other hand, from the very special structure of the function  $\phi_{\varepsilon}$ , we have

$$\begin{pmatrix} \partial_{(\mathbf{u},\mathbf{y})} \Phi_1(\bar{\varepsilon}, \bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \\ \vdots \\ \partial_{(\mathbf{u},\mathbf{y})} \Phi_s(\bar{\varepsilon}, \bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \end{pmatrix} = \partial_{(\mathbf{u},\mathbf{y})} \Phi(\bar{\varepsilon}, \bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}),$$

see [23]. The above formula, together with (5.3.12) and (5.3.13), implies

$$\Pi_{(\mathbf{u},\mathbf{y})}\partial\Phi(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}})\subseteq\partial_{(\mathbf{u},\mathbf{y})}\Phi(\bar{\varepsilon},\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}}).$$

Hence, we obtain, by Proposition 5.3.3, that  $\Pi_{(\mathbf{u},\mathbf{y})}\partial\Phi(\bar{\varepsilon},\bar{x},\mathbf{\bar{u}},\mathbf{\bar{y}})$  is comprised of nonsingular matrices. The proof is complete.  $\blacksquare$ 

Let S denote the set of all points  $(x, \mathbf{u}, \mathbf{y})$  satisfying  $\Phi(0, x, \mathbf{u}, \mathbf{y}) = 0$  and  $G(x, \mathbf{u}) \leq 0$ , that is,

$$\mathcal{S} := \left\{ (x, \mathbf{u}, \mathbf{y}) \in \Re^{n + (r+J)L} \mid \Phi(0, x, \mathbf{u}, \mathbf{y}) = 0, \ G(x, \mathbf{u}) \le 0 \right\}.$$
 (5.3.14)

**Proposition 5.3.5** Let  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \in S$ . Suppose that for every  $l = 1, 2, \dots, L$ ,  $\bar{A}_l$  is negative definite, and the LICQ is satisfied at  $\bar{u}_l$  for problem  $Q_l(\bar{x})$ . Then, there exist

two positive numbers  $\bar{\varepsilon}$  and  $\bar{\tau}$ , a neighborhood  $N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$  of  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$ , and a continuous function  $(\mathbf{u}(\cdot, \cdot), \mathbf{y}(\cdot, \cdot))$ :  $[0, \bar{\varepsilon}) \times \prod_x N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \to \Re^{(r+J)L}$ , such that for any  $(\varepsilon, x, \mathbf{u}, \mathbf{y}) \in (0, \bar{\varepsilon}) \times (N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \cap S)$ ,

$$\Phi(\varepsilon, x, \mathbf{u}(\varepsilon, x), \mathbf{y}(\varepsilon, x)) = 0$$

and

$$||\mathbf{u}(\varepsilon, x) - \mathbf{u}|| \le 2\sqrt{LJ}\overline{\tau}\varepsilon, \quad ||\mathbf{y}(\varepsilon, x) - \mathbf{y}|| \le 2\sqrt{LJ}\overline{\tau}\varepsilon.$$
 (5.3.15)

**Proof.** Firstly, by Proposition 5.3.4, there exist a positive number  $\hat{\varepsilon}$ , a neighborhood  $N(\bar{x})$  of  $\bar{x}$  and a continuous function  $(\mathbf{u}(\cdot, \cdot), \mathbf{y}(\cdot, \cdot)) : [0, \hat{\varepsilon}) \times N(\bar{x}) \to R^{(r+J)L}$ , such that for any  $(\varepsilon, x) \in (0, \bar{\varepsilon}) \times N(\bar{x})$ ,

$$\Phi(\varepsilon, x, \mathbf{u}(\varepsilon, x), \mathbf{y}(\varepsilon, x)) = 0.$$
(5.3.16)

Secondly, it is not difficult to see that  $\Phi(\varepsilon, x, \mathbf{u}, \mathbf{y})$  is smooth and  $\nabla_{(\mathbf{u}, \mathbf{y})} \Phi(\varepsilon, x, \mathbf{u}, \mathbf{y})$ is nonsingular for any  $\varepsilon > 0$  and  $(x, \mathbf{u}, \mathbf{y})$  close enough to  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$ . We now show that there exist a neighborhood  $N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$  with  $\Pi_x N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \subset N(\bar{x})$  and a positive number  $\bar{\varepsilon} \in (0, \hat{\varepsilon})$  such that (5.3.15) holds for any  $(\varepsilon, x, \mathbf{u}, \mathbf{y}) \in (0, \bar{\varepsilon}) \times (N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \cap \mathcal{S})$ . To this end, we show that there exist a positive number  $\bar{\varepsilon}$ , a neighborhood  $N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \cap \mathcal{S})$ ,  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$  and a positive constant  $\bar{\tau}$  such that for any  $(\varepsilon, x, \mathbf{u}, \mathbf{y}) \in (0, \bar{\varepsilon}) \times (N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \cap \mathcal{S})$ ,

$$||\nabla_{(\mathbf{u},\mathbf{y})}\Phi(\tilde{\varepsilon}, x, \tilde{\mathbf{u}}, \tilde{\mathbf{y}})^{-1}|| \le \bar{\tau} , \qquad (5.3.17)$$

where  $0 < \tilde{\varepsilon} < \varepsilon$ ,  $\mathbf{u} < \tilde{\mathbf{u}} < \mathbf{u}(\varepsilon, x)$ , which means that every component of  $\tilde{\mathbf{u}}$  is in the open segment connecting the corresponding component of  $\mathbf{u}$  and  $\mathbf{u}(\varepsilon, x)$ , and  $\mathbf{y} < \tilde{\mathbf{y}} < \mathbf{y}(\varepsilon, x)$ . Here, in different rows of  $\nabla_{(\mathbf{u},\mathbf{y})} \Phi(\tilde{\varepsilon}, x, \tilde{\mathbf{u}}, \tilde{\mathbf{y}})$ , the values of  $\tilde{\varepsilon}$ ,  $\tilde{\mathbf{u}}$  and  $\tilde{\mathbf{y}}$  may not be the same, but for the sake of simplicity, they are still written as  $\tilde{\varepsilon}$ ,  $\tilde{\mathbf{u}}$  and  $\tilde{\mathbf{y}}$ . Suppose on the contrary that (5.3.17) does not hold, then there exist a sequence  $\{\varepsilon_k\}$  with  $\varepsilon_k \downarrow 0$ and  $\{(x^k, \mathbf{u}^k, \mathbf{y}^k)\}$  with  $(x^k, \mathbf{u}^k, \mathbf{y}^k) \to (\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$ , such that

$$||\nabla_{(\mathbf{u},\mathbf{y})}\Phi(\tilde{\varepsilon}_k, x^k, \tilde{\mathbf{u}}^k, \tilde{\mathbf{y}}^k)^{-1}|| \to \infty$$
(5.3.18)

for some  $0 < \tilde{\varepsilon}_k < \varepsilon_k$ ,  $\mathbf{u}^k < \tilde{\mathbf{u}}^k < \mathbf{u}(\varepsilon_k, x^k)$  and  $\mathbf{y}^k < \tilde{\mathbf{y}}^k < \mathbf{y}(\varepsilon_k, x^k)$ . Since  $(\mathbf{u}(\cdot, \cdot), \mathbf{y}(\cdot, \cdot))$  is a continuous function and  $(x^k, \mathbf{u}^k, \mathbf{y}^k) \to (\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$  and  $\varepsilon_k \downarrow 0$ , it follows that  $\tilde{\varepsilon}_k \to 0$  and  $(\tilde{\mathbf{u}}^k, \tilde{\mathbf{y}}^k) \to (\bar{\mathbf{u}}, \bar{\mathbf{y}})$ . On the other hand, by direct computation, we

obtain

$$\nabla_{(\mathbf{u},\mathbf{y})} \Phi(\tilde{\varepsilon}_k, x^k, \tilde{\mathbf{u}}^k, \tilde{\mathbf{y}}^k)^T = \begin{pmatrix} \tilde{A}_1^k & -\tilde{B}_1^k & & \\ (\tilde{C}_1^k + I)\tilde{B}_1^{kT} & \tilde{C}_1^k - I & & \\ & \ddots & & \\ & & \tilde{A}_L^k & -\tilde{B}_L^k \\ & & & (\tilde{C}_L^k + I)\tilde{B}_L^{kT} & \tilde{C}_L^k - I \end{pmatrix},$$

where 
$$\tilde{A}_{l}^{k}$$
,  $\tilde{B}_{l}^{k}$ ,  $\tilde{C}_{l}^{k}$ ,  $l = 1, 2, \cdots, L$ , are matrices given by  

$$\begin{cases}
\tilde{A}_{l}^{k} = \nabla_{u_{l}u_{l}}^{2}g_{l}(x^{k}, \tilde{u}_{l}^{k}) - \sum_{j=1}^{J}(\tilde{y}_{l}^{k})_{j}\nabla_{u_{l}u_{l}}^{2}h_{j}^{l}(x^{k}, \tilde{u}_{l}^{k}), \\
\tilde{B}_{l}^{k} = [\nabla_{u_{l}}h_{1}^{l}(x^{k}, \tilde{u}_{l}^{k}), \cdots, \nabla_{u_{l}}h_{J}^{l}(x^{k}, \tilde{u}_{l}^{k})], \\
\tilde{C}_{l}^{k} = \operatorname{diag}[\tilde{c}_{1}^{k}(l), \cdots, \tilde{c}_{J}^{k}(l)], \quad \tilde{c}_{j}^{k}(l) = \frac{(\tilde{y}_{l}^{k})_{j} + h_{j}^{l}(x^{k}, \tilde{u}_{l}^{k})}{\sqrt{((\tilde{y}_{l}^{k})_{j} + h_{j}^{l}(x^{k}, \tilde{u}_{l}^{k}))^{2} + 4\tilde{\varepsilon}_{k}^{2}}}, \\
l = 1, 2, \cdots, L.$$
Since  $\tilde{\varepsilon}_{k} \to 0$  and  $(x^{k}, \tilde{\mathbf{u}}^{k}, \tilde{\mathbf{y}}^{k}) \to (\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$ , it follows that

$$\lim_{k\to\infty}\nabla_{(\mathbf{u},\mathbf{y})}\Phi(\tilde{\varepsilon}_k,x^k,\tilde{\mathbf{u}}^k,\tilde{\mathbf{y}}^k)^T\in\partial_{(\mathbf{u},\mathbf{y})}\Phi(0,\bar{x},\bar{\mathbf{u}},\bar{\mathbf{y}})$$

By Proposition 5.3.3, all matrices in  $\partial_{(\mathbf{u},\mathbf{y})} \Phi(0, \bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$  are nonsingular. Hence, there exists a positive constant  $\bar{\tau}$  such that

$$||\nabla_{(\mathbf{u},\mathbf{y})}\Phi(\tilde{\varepsilon}_k,x^k,\tilde{\mathbf{u}}^k,\tilde{\mathbf{y}}^k)^{-1}|| \leq \bar{\tau},$$

for k large enough, which contradicts (5.3.18). Therefore, (5.3.17) holds.

We assume, without loss of generality, that  $(0, \bar{\varepsilon}) \times \Pi_x N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \subset (0, \hat{\varepsilon}) \times N(\bar{x})$ . Take any  $(\varepsilon, x, \mathbf{u}, \mathbf{y}) \in (0, \overline{\varepsilon}) \times (N(\overline{x}, \overline{\mathbf{u}}, \overline{\mathbf{y}}) \cap \mathcal{S})$ . Since  $\Phi(0, x, \mathbf{u}, \mathbf{y}) = 0$ , we have, by (5.3.16) and the mean value theorem, that

$$0 = \Phi(\varepsilon, x, \mathbf{u}(\varepsilon, x), \mathbf{y}(\varepsilon, x)) - \Phi(0, x, \mathbf{u}, \mathbf{y})$$
  
$$= \nabla_{(\mathbf{u}, \mathbf{y})} \Phi(\tilde{\varepsilon}, x, \tilde{\mathbf{u}}, \tilde{\mathbf{y}})^T \begin{pmatrix} \mathbf{u}(\varepsilon, x) - \mathbf{u} \\ \mathbf{y}(\varepsilon, x) - \mathbf{y} \end{pmatrix} + \varepsilon \begin{pmatrix} \mathbf{0} \\ \tilde{\theta}^{\mathbf{1}} \\ \vdots \\ \mathbf{0} \\ \tilde{\theta}^{\mathbf{L}} \end{pmatrix}, \qquad (5.3.19)$$

where  $\mathbf{u} < \tilde{\mathbf{u}} < \mathbf{u}(\varepsilon, x), \, \mathbf{y} < \tilde{\mathbf{y}} < \mathbf{y}(\varepsilon, x), \, 0 < \tilde{\varepsilon} < \varepsilon, \, \mathbf{0}$  is the *r*-dimensional zero vector and  $\tilde{\theta}^{\mathbf{l}} = (\tilde{\theta}_{1}^{\mathbf{l}}, \cdots, \tilde{\theta}_{\mathbf{J}}^{\mathbf{l}})^{\mathbf{T}}$ , where

$$\tilde{\theta}_j^l = \frac{4\tilde{\varepsilon}}{\sqrt{((\tilde{y}_l)_j + h_j^l(x, \tilde{u}_l))^2 + 4\tilde{\varepsilon}^2}}, \quad j = 1, 2, \cdots, J, \ l = 1, 2, \cdots, L.$$

It is clear that  $0 < \tilde{\theta}_j^l < 2$ . Note that (5.3.17) holds even if the values of  $\tilde{\varepsilon}$ ,  $\tilde{\mathbf{u}}$  and  $\tilde{\mathbf{y}}$  in different rows of  $\nabla_{(\mathbf{u},\mathbf{y})} \Phi(\tilde{\varepsilon}, x, \tilde{\mathbf{u}}, \tilde{\mathbf{y}})$  are different, we have, by (5.3.19) and (5.3.17), that

$$\left\| \begin{pmatrix} \mathbf{u}(\varepsilon, x) - \mathbf{u} \\ \mathbf{y}(\varepsilon, x) - \mathbf{y} \end{pmatrix} \right\| \leq \varepsilon \| \nabla_{(\mathbf{u}, \mathbf{y})} \Phi(\tilde{\varepsilon}, x, \tilde{\mathbf{u}}, \tilde{\mathbf{y}})^{-1} \| \left\| \begin{pmatrix} 0 \\ \tilde{\theta}^{\mathbf{1}} \\ \vdots \\ 0 \\ \tilde{\theta}^{\mathbf{L}} \end{pmatrix} \right\| \leq 2\sqrt{LJ} \bar{\tau} \varepsilon,$$

where the last inequality follows from the fact that  $||\tilde{\theta}^l|| \leq 2\sqrt{J}$  for  $l = 1, 2, \dots, L$ . This immediately yields the desired result. The proof is complete.

**Proposition 5.3.6** Suppose that there exists a vector  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \in S$  such that for every  $l = 1, 2, \dots, L$ ,  $\bar{A}_l$  is negative definite and the LICQ is satisfied at  $\bar{u}_l$  for problem  $Q_l(\bar{x})$ . Then there exists an  $\bar{\varepsilon} > 0$  such that the feasible set  $\mathcal{F}(\varepsilon^2, \delta)$  of problem  $\tilde{P}(\varepsilon^2, \delta)$  is nonempty for any  $0 < \varepsilon < \bar{\varepsilon}$  and  $\delta = \varepsilon$ .

**Proof.** By Proposition 5.3.5, there exist two positive numbers  $\bar{\varepsilon}$  and  $\bar{\tau}$  and a continuous function  $(\mathbf{u}(\cdot, \bar{x}), \mathbf{y}(\cdot, \bar{x})) : [0, \bar{\varepsilon}) \to \Re^r \times \Re^J$  such that for any  $0 < \varepsilon < \bar{\varepsilon}$ ,

$$\Phi(\varepsilon^2, \bar{x}, \mathbf{u}(\varepsilon^2, \bar{x}), \mathbf{y}(\varepsilon^2, \bar{x})) = 0$$
(5.3.20)

and

$$||\mathbf{u}(\varepsilon^2, \bar{x}) - \bar{\mathbf{u}}|| \le 2\sqrt{LJ}\bar{\tau}\varepsilon^2.$$

Hence,

$$||u_l(\varepsilon^2, \bar{x}) - \bar{u}_l|| \le 2\sqrt{LJ}\bar{\tau}\varepsilon^2, \ l = 1, 2, \cdots, L.$$
(5.3.21)

Since  $g^l(\bar{x}, u_l)$  has continuous first-derivatives  $\nabla_{u_l} g^l(\bar{x}, u_l)$  for every  $l = 1, 2, \dots, L$ , it is clear that  $g^l(\bar{x}, u_l)$  is locally Lipschitz with respect to the variable  $u_l$ . Therefore, there exists a positive constant  $\bar{M}$  such that for  $l = 1, 2, \dots, L$ ,

$$|g^l(\bar{x}, u_l(\varepsilon^2, \bar{x})) - g^l(\bar{x}, \bar{u}_l)| \le \bar{M} ||u_l(\varepsilon^2, \bar{x}) - \bar{u}_l||,$$

which in turn implies

$$G(\bar{x}, \mathbf{u}(\varepsilon^{2}, \bar{x})) \leq G(\bar{x}, \bar{\mathbf{u}}) + \bar{M} \sum_{l=1}^{L} p_{l} ||u_{l}(\varepsilon^{2}, \bar{x}) - \bar{u}_{l}||$$

$$\leq 2\bar{M}\sqrt{LJ}\bar{\tau}\varepsilon^{2},$$
(5.3.22)

where the last inequality follows from (5.3.21) and the fact that  $G(\bar{x}, \bar{\mathbf{u}}) \leq 0$ . By (5.3.22), we can take  $\bar{\varepsilon}$  small enough such that  $2\bar{M}\sqrt{LJ}\bar{\tau}\bar{\varepsilon} \leq 1$ . Hence, for any  $0 < \varepsilon < \bar{\varepsilon}$ , we have

$$G(\bar{x}, \mathbf{u}(\varepsilon^2, \bar{x})) \le \varepsilon.$$

This formula, together with (5.3.20), implies  $\bar{x} \in \mathcal{F}(\varepsilon^2, \varepsilon)$  for  $0 < \varepsilon < \bar{\varepsilon}$ . We obtain the desired result and complete the proof of the proposition.

**Assumption F3.** For every  $\varepsilon > 0$  and  $l = 1, 2, \dots, L$ , there are vectors  $u_l(\varepsilon^2, x)$ and  $y_l(\varepsilon^2, x)$  such that

$$\Phi_l(\varepsilon^2, x, u_l(\varepsilon^2, x), y_l(\varepsilon^2, x)) = 0.$$

The vector  $u_l(\varepsilon^2, x)$  is unique and continuous with respect to x for every  $l = 1, 2, \dots, L$ .

**Theorem 5.3.1** Suppose that for any  $x \in \Re^n$  and every  $l = 1, 2, \dots, L$ , the LICQ is satisfied at every  $u_l \in P_l(x)$  for problem  $Q_l(x)$ , where

$$P_l(x) := \{ u \in \Re^r \mid \exists \ j \in \{1, 2, \cdots, J\} \ s.t. \ h_j^l(x, u) = 0 \}.$$

Suppose further that there exists a vector  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$  such that the conditions in Proposition 5.3.6 hold and the level set

$$\{x \in \Re^n \mid f(x) \le f(\bar{x})\}\$$

is bounded. Then, under Assumption F3, there exists a positive number  $\bar{\varepsilon}$  such that problem  $\tilde{P}(\varepsilon^2, \varepsilon)$  is solvable for any  $0 < \varepsilon < \bar{\varepsilon}$ .

**Proof.** Firstly, by Propositions 5.3.6, there exists an  $\bar{\varepsilon} > 0$  such that  $\bar{x} \in \mathcal{F}(\varepsilon^2, \varepsilon)$  for any  $0 < \varepsilon < \bar{\varepsilon}$ , which implies  $\mathcal{F}(\varepsilon^2, \varepsilon)$  is nonempty. Furthermore, we may prove that for any fixed  $\varepsilon \in (0, \bar{\varepsilon})$ ,  $\mathcal{F}(\varepsilon^2, \varepsilon)$  is closed from Assumption F3 and the given condition that  $g^l(\cdot, \cdot)$ ,  $h^l(\cdot, \cdot)$ ,  $l = 1, 2, \dots, L$ , are continuous. In fact, for a sequence of feasible points  $\{x^k\} \subset \mathcal{F}(\varepsilon^2, \varepsilon)$  with a limit point  $\hat{x}$ , there exist  $\mathbf{u}(\varepsilon^2, x^k)$  and  $\mathbf{y}(\varepsilon^2, x^k)$  such that  $\Phi(\varepsilon^2, x^k, \mathbf{u}(\varepsilon^2, x^k), \mathbf{y}(\varepsilon^2, x^k)) = 0$  and  $G(x^k, \mathbf{u}(\varepsilon^2, x^k)) \leq \varepsilon$ . Since  $\Phi(\varepsilon^2, x^k, \mathbf{u}(\varepsilon^2, x^k)) = 0$ , we have that for every  $l = 1, 2, \dots, L$ ,

$$\begin{cases} \nabla_{u_l} g^l(x^k, u_l(\varepsilon^2, x^k)) - \nabla_{u_l} h^l(x^k, u_l(\varepsilon^2, x^k)) y_l(\varepsilon^2, x^k) = 0, \\ \phi_{\varepsilon^2}((y_l(\varepsilon^2, x^k))_j, -h_j^l(x^k, u_l(\varepsilon^2, x^k))) = 0, \ j = 1, 2, \cdots, J. \end{cases}$$
(5.3.23)

We claim that  $\{\mathbf{y}(\varepsilon^2, x^k)\}$  is bounded. Otherwise, there exists an index  $l_0$  such that  $||y_{l_0}(\varepsilon^2, x^k)|| \to \infty$ . Then, by dividing every equality for index  $l_0$  in (5.3.23) by  $||y_{l_0}(\varepsilon^2, x^k)||$  and letting  $k \to \infty$ , we obtain

$$\begin{cases} \nabla_{u_{l_0}} h^{l_0}(\hat{x}, \hat{u}_{l_0}) \hat{y}_{l_0} = 0, \\ (\hat{y}_{l_0})_j \ge 0, h_j^{l_0}(\hat{x}, \hat{u}_{l_0}) \le 0, \\ (\hat{y}_{l_0})_j h_j^{l_0}(\hat{x}, \hat{u}_{l_0}) = 0, \quad j = 1, 2, \cdots, J, \end{cases}$$
(5.3.24)

where  $\hat{u}_{l_0} = u_{l_0}(\varepsilon^2, \hat{x})$  and  $0 \neq \hat{y}_{l_0} = ((\hat{y}_{l_0})_1, (\hat{y}_{l_0})_2, \cdots, (\hat{y}_{l_0})_J)^T \in U$ . From the last equality in (5.3.24), we have  $(\hat{y}_{l_0})_j = 0$  for all j such that  $h_j^{l_0}(\hat{x}, \hat{u}_{l_0}) \neq 0$ . Hence, the first equality in (5.3.24) can be rewritten as

$$\sum_{j \in \mathcal{I}_{h^{l_0}}(\hat{x}, \hat{u}_{l_0})} (\hat{y}_{l_0})_j \nabla_{u_{l_0}} h_j^{l_0}(\hat{x}, \hat{u}_{l_0}) = 0.$$
(5.3.25)

It is clear that the set  $\{(\hat{y}_{l_0})_j \mid j \in \mathcal{I}_{h^{l_0}}(\hat{x}, \hat{u}_{l_0})\}$  contains a non-zero element and  $\hat{u}_{l_0} \in P_{l_0}(\hat{x})$ . Hence, by the given condition that the LICQ is satisfied at  $\hat{u}_{l_0}$  for problem  $Q_{l_0}(\hat{x})$ , we deduce that

$$\nabla_{u_{l_0}} h_j^{l_0}(\hat{x}, \hat{u}_{l_0}), \quad j \in \mathcal{I}_{h^{l_0}}(\hat{x}, \hat{u}_{l_0})$$

are linearly independent. This contradicts (5.3.25). Since  $\{\mathbf{y}(\varepsilon^2, x^k)\}$  is bounded, without loss of generality, we assume that  $\mathbf{y}(\varepsilon^2, x^k) \to \hat{\mathbf{y}}$ . Since  $\Phi(\varepsilon^2, x^k, \mathbf{u}(\varepsilon^2, x^k), \mathbf{y}(\varepsilon^2, x^k))$ = 0 and  $G(x^k, \mathbf{u}(\varepsilon^2, x^k)) \leq \varepsilon$ , letting  $k \to \infty$  yields  $\Phi(\varepsilon^2, \hat{x}, \mathbf{u}(\varepsilon^2, \hat{x}), \hat{\mathbf{y}}) = 0$  and  $G(\hat{x}, \mathbf{u}(\varepsilon^2, \hat{x})) \leq \varepsilon$ , which implies  $\hat{x} \in \mathcal{F}(\varepsilon^2, \varepsilon)$ . Hence, we obtain the desired result from the continuity of f. The proof is complete.

### 5.4 Algorithm and Its Convergence Analysis

In this section, we further consider problem  $\tilde{P}$  in the case where  $\Omega$  is a finite discrete set. From the discussion in the previous sections, problem (5.1.4) is equivalent to problem  $\tilde{P}$ . Furthermore, if  $\tilde{P}$  is solvable, then there exists a positive number  $\bar{\varepsilon}$  such that problem  $P(\varepsilon^2, \varepsilon)$  is solvable for any  $0 < \varepsilon < \bar{\varepsilon}$  under suitable conditions. Since  $\tilde{P}(\varepsilon^2, \varepsilon)$  is a smooth approximation to the nonsmooth problem  $\tilde{P}$ , we may obtain a solution of problem  $\tilde{P}$  by solving a sequence of smooth problems  $\tilde{P}(\varepsilon^2, \varepsilon)$ . Now we present a smoothing implicit programming approach for solving problem  $\tilde{P}$ : **Algorithm 5.4.1** Let  $\{\varepsilon_k\}$  be a sequence of positive numbers such that  $\varepsilon_k \downarrow 0$ . For  $k = 1, 2, \dots$ , find a global solution  $x^k$  of the problem

$$\min_{x} f(x)$$
(5.4.26)
  
s. t.  $G(x, \mathbf{u}(\varepsilon_k^2, x)) \le \varepsilon_k,$ 

where  $\mathbf{u}(\varepsilon_k^2, x) = (u_1(\varepsilon_k^2, x), \cdots, u_L(\varepsilon_k^2, x))$ , together with  $\mathbf{y}(\varepsilon_k^2, x) = (y_1(\varepsilon_k^2, x), \cdots, y_L(\varepsilon_k^2, x))$ , satisfies the system

$$\Phi_l(\varepsilon_k^2, x, u_l(\varepsilon_k^2, x), y_l(\varepsilon_k^2, x)) = 0, \qquad l = 1, 2, \cdots L.$$
(5.4.27)

Let 
$$u_l^k = u_l(\varepsilon_k^2, x^k), \ y_l^k = y_l(\varepsilon_k^2, x^k), \ l = 1, 2, \cdots L$$
, and  
 $\mathbf{u}^k = \begin{pmatrix} u_1^k \\ \vdots \\ u_L^k \end{pmatrix}, \quad \mathbf{y}^k = \begin{pmatrix} y_1^k \\ \vdots \\ y_L^k \end{pmatrix}.$ 

Note that problem (5.4.26) is a smooth optimization problem. Under Assumption F3, Algorithm 5.4.1 is well-defined. Now we investigate the limiting behavior of a sequence of optimal solutions of (5.4.26). To this end, we make the following assumption in addition to Assumption F3, throughout the rest of this section.

Assumption F4. The sequence  $\{(x^k, \mathbf{u}^k, \mathbf{y}^k)\}$  generated by Algorithm 5.4.1 is convergent to a point  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$ .

Recall that  $\tilde{F}$  denotes the feasible region of problem  $\tilde{P}$  and the set S is defined by (5.3.14). We define the set-valued mapping  $S : \tilde{\mathcal{F}} \to \Re^{(r+J)L}$  by

$$S(x) := \left\{ (\mathbf{u}, \mathbf{y}) \in \Re^{(r+J)L} \mid (x, \mathbf{u}, \mathbf{y}) \in \mathcal{S} \right\}.$$

**Definition 5.4.1** Let  $\bar{x} \in \tilde{\mathcal{F}}$  and  $(\bar{\mathbf{u}}, \bar{\mathbf{y}}) \in S(\bar{x})$ . We say that the set-valued mapping S is stable at  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$  if, for any neighborhood  $N(\bar{\mathbf{u}}, \bar{\mathbf{y}})$  of  $(\bar{\mathbf{u}}, \bar{\mathbf{y}})$ , there exists a neighborhood  $N(\bar{x})$  of  $\bar{x}$  such that  $S(x) \cap N(\bar{\mathbf{u}}, \bar{\mathbf{y}}) \neq \emptyset$  for any  $x \in N(\bar{x}) \cap \tilde{\mathcal{F}}$ .

**Theorem 5.4.1** Let  $\{(x^k, \mathbf{u}^k, \mathbf{y}^k)\}$  be a sequence generated by Algorithm 5.4.1. Then the limit point  $\bar{x}$  of  $\{x^k\}$  lies in  $\tilde{\mathcal{F}}$ . Moreover, suppose that for every  $l = 1, 2, \dots, L$ ,  $\bar{A}_l$ is negative definite, and the LICQ is satisfied at  $\bar{u}_l$  for problem  $Q_l(\bar{x})$ , and the set-valued mapping S is stable at  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$ . Then  $\bar{x}$  is a local optimal solution of problem  $\tilde{P}$ . **Proof.** First note that

$$G(x^k, \mathbf{u}^k) \le \varepsilon_k,$$
  
$$\Phi(\varepsilon_k^2, x^k, \mathbf{u}^k, \mathbf{y}^k) = 0$$

hold for all k. Letting  $k \to \infty$ , we have  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \in \mathcal{S}$ , which implies that  $\bar{x}$  is a feasible solution of  $\tilde{P}$ . Moreover, by Proposition 5.3.5, there exist a positive number  $\bar{\varepsilon}$ , a neighborhood  $N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$  of  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$ , a continuous function  $(\mathbf{u}(\cdot, \cdot), \mathbf{y}(\cdot, \cdot)) : [0, \bar{\varepsilon}) \times$  $\Pi_x N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \to \Re^{(r+J)L}$ , and a positive constant  $\bar{\tau}$  such that, for any  $(\varepsilon_k, x, \mathbf{u}, \mathbf{y}) \in$  $(0, \bar{\varepsilon}) \times (N(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}}) \cap \mathcal{S})$ ,

$$\Phi(\varepsilon_k^2, x, \mathbf{u}(\varepsilon_k^2, x), \mathbf{y}(\varepsilon_k^2, x)) = 0$$
(5.4.28)

and

$$||\mathbf{u}(\varepsilon_k^2, x) - \mathbf{u}|| \le 2\sqrt{LJ}\bar{\tau}\varepsilon_k^2, \quad ||\mathbf{y}(\varepsilon_k^2, x) - \mathbf{y}|| \le 2\sqrt{LJ}\bar{\tau}\varepsilon_k^2.$$
(5.4.29)

Then, in a similar way to the proof of Proposition 5.3.6, we can show that there exists a positive constant  $\overline{M}$  such that

$$G(x, \mathbf{u}(\varepsilon_k^2, x)) \leq G(x, \mathbf{u}) + \bar{M} ||\mathbf{u}(\varepsilon_k^2, x) - \mathbf{u}||$$
  
$$\leq 2\bar{M}\sqrt{LJ}\bar{\tau}\varepsilon_k^2 \leq \varepsilon_k,$$

for all k large enough. The above discussion shows that there exists a neighborhood  $N(\bar{x})$  of  $\bar{x}$  such that for any  $x \in N(\bar{x}) \cap \tilde{\mathcal{F}}$ , x is a feasible solution of (5.4.26) whenever k is large enough, since the set-valued mapping S is stable at  $(\bar{x}, \bar{\mathbf{u}}, \bar{\mathbf{y}})$  and (5.4.28) holds at x for k large enough. Therefore, for any  $x \in N(\bar{x}) \cap \tilde{F}$ , the inequality

$$f(x^k) \le f(x)$$

holds for all k large enough. Letting  $k \to \infty$ , we have

$$f(\bar{x}) \le f(x),$$

which implies that  $\bar{x}$  is a local optimal solution of problem  $\tilde{P}$ . The proof is complete.

**Theorem 5.4.2** Let  $\{(x^k, \mathbf{u}^k, \mathbf{y}^k)\}$  be a sequence generated by Algorithm 5.4.1. Suppose that, for every  $(x, \mathbf{u}, \mathbf{y}) \in S$  and every  $l = 1, 2, \dots, L$ ,  $A_l$  is negative definite, and the LICQ is satisfied at  $u_l$  for problem  $Q_l(x)$ , and the set-valued mapping S is stable at every  $(x, \mathbf{u}, \mathbf{y}) \in S$ . Then the limit point  $\bar{x}$  of  $\{x^k\}$  is a global optimal solution of problem  $\tilde{P}$ . **Proof.** Recall that  $\bar{x}$  is a feasible solution of  $\tilde{P}$ . For an arbitrary positive number  $\eta$ , we define the set  $\tilde{\mathcal{F}}_{\eta}$  by

$$\tilde{\mathcal{F}}_{\eta} = \left\{ x \in \tilde{\mathcal{F}} \mid ||x - \bar{x}|| \le \eta \right\}.$$

It is clear that  $\tilde{\mathcal{F}}_{\eta}$  is a nonempty compact set. For any  $\hat{x} \in \tilde{\mathcal{F}}_{\eta}$ , there exists  $(\hat{\mathbf{u}}, \hat{\mathbf{y}})$ such that  $(\hat{x}, \hat{\mathbf{u}}, \hat{\mathbf{y}}) \in \mathcal{S}$ . Since the conditions in Proposition 5.3.5 are satisfied at  $(\hat{x}, \hat{\mathbf{u}}, \hat{\mathbf{y}})$ , in a similar way to the proof of Theorem 5.4.1, we can show that there exist a neighborhood  $N(\hat{x}, \hat{\mathbf{u}}, \hat{\mathbf{y}})$  of  $(\hat{x}, \hat{\mathbf{u}}, \hat{\mathbf{y}})$ , two positive numbers  $\hat{\varepsilon} = \hat{\varepsilon}(\hat{x})$  and  $\hat{\tau} = \hat{\tau}(\hat{x})$ , and a continuous function  $(\mathbf{u}(\cdot, \cdot), \mathbf{y}(\cdot, \cdot)) : [0, \hat{\varepsilon}) \times \prod_x N(\hat{x}, \hat{\mathbf{u}}, \hat{\mathbf{y}}) \to \Re^{(r+J)L}$  such that, for any  $(\varepsilon, x, \mathbf{u}, \mathbf{y}) \in (0, \hat{\varepsilon}) \times (N(\hat{x}, \hat{\mathbf{u}}, \hat{\mathbf{y}}) \cap \mathcal{S})$ ,

$$\Phi(\varepsilon^2, x, \mathbf{u}(\varepsilon^2, x), \mathbf{y}(\varepsilon^2, x)) = 0$$
(5.4.30)

and

$$G(x, \mathbf{u}(\varepsilon^2, x)) \le 2\hat{M}\sqrt{LJ}\hat{\tau}\varepsilon^2, \qquad (5.4.31)$$

where  $\hat{M}$  is given by

$$\hat{M} = \max{\{\hat{M}^1(\hat{x}, \hat{u}_1), \cdots, \hat{M}^L(\hat{x}, \hat{u}_L)\}},$$

and  $\hat{M}^{l}(\hat{x}, \hat{u}_{l})$  is a local Lipschitz constant of the function  $g^{l}(\hat{x}, \cdot)$  at  $\hat{u}_{l}$  for each  $l = 1, 2, \dots, L$ . Moreover, there exists a neighborhood  $N(\hat{x})$  of  $\hat{x}$  such that (5.4.30) and (5.4.31) hold for any  $(\varepsilon, x) \in (0, \hat{\varepsilon}) \times N(\hat{x})$ , since the set-valued mapping S is stable at  $(\hat{x}, \hat{\mathbf{u}}, \hat{\mathbf{y}})$ . Since the family of neighborhoods

$$\mathcal{N} = \{ N(\hat{x}) \mid \hat{x} \in \tilde{\mathcal{F}}_{\eta} \}$$

is an open covering of  $\tilde{\mathcal{F}}_{\eta}$ , there is a finite number of neighborhoods, say  $N_1, N_2, \dots, N_s$ , in  $\mathcal{N}$  such that  $\{N_1, N_2, \dots, N_s\}$  constitutes a covering of  $\tilde{\mathcal{F}}_{\eta}$ . Accordingly, there exist constants  $\hat{\varepsilon}_1, \hat{\varepsilon}_2, \dots, \hat{\varepsilon}_s, \hat{\tau}_1, \hat{\tau}_2, \dots, \hat{\tau}_s$  and  $\hat{M}_1, \hat{M}_2, \dots, \hat{M}_s$ , respectively. Thus, by setting

$$\begin{aligned} \varepsilon^* &= \min\{\hat{\varepsilon}_1, \hat{\varepsilon}_2, \cdots, \hat{\varepsilon}_s\},\\ \tau^* &= \max\{\hat{\tau}_1, \hat{\tau}_2, \cdots, \hat{\tau}_s\},\\ M^* &= \max\{\hat{M}_1, \hat{M}_2, \cdots, \hat{M}_s\}, \end{aligned}$$

we have  $\varepsilon_k \leq \varepsilon^*$  and  $2M^* \sqrt{LJ} \tau^* \varepsilon_k \leq 1$  for all k large enough, and hence, for every  $x \in \tilde{\mathcal{F}}_{\eta}$ ,

$$\Phi(\varepsilon_k^2, x, \mathbf{u}(\varepsilon_k^2, x), \mathbf{y}(\varepsilon_k^2, x)) = 0$$

and

$$G(x, \mathbf{u}(\varepsilon_k^2, x)) \le \varepsilon_k.$$

This shows that for every  $x \in \tilde{\mathcal{F}}_{\eta}$ , x is a feasible solution of (5.4.26) whenever k is large enough. Therefore, by using similar arguments to the proof of Theorem 5.4.1, we can show that  $\bar{x}$  is an optimal solution of the problem

$$\min_{x} \quad f(x) \\ \text{s. t.} \quad x \in \tilde{\mathcal{F}}_{\eta}$$

Since  $\eta$  is arbitrary,  $\bar{x}$  is actually a global optimal solution of  $\tilde{P}$ . The proof is complete.

## 5.5 Some Remarks

In this chapter, we have reformulated the SGSIP problem as a nonlinear programming problem with stochastic complementarity constraints, and established some properties of smoothing approximations for the reformulated problem. Furthermore, we have presented a smoothing implicit programming algorithm (Algorithm 5.4.1) for solving the problem with finite discrete distribution. Unlike other numerical methods for semiinfinite programming, our approach does not discretize the index set, but we take advantage of the fact that the lower level programs can be characterized by its first order optimality condition. Because of the special structure of  $\Phi$  (see (5.3.10)), our approach is numerical tractable under some mild assumptions.

To illustrate the assumptions and the theorems in this chapter, we consider the following example.

Example 5.5.1 Let

$$\begin{cases} g^{l}(x, u_{l}) = \frac{1}{2}u_{l}^{T}B_{l}(x)u_{l} + C_{l}(x)^{T}u_{l} + d_{l}(x), \\ h_{j}^{l}(x, u_{l}) = p_{j,l}(x)^{T}u_{l} + q_{j,l}(x), \\ j = 1, 2, \cdots, J, \ l = 1, 2, \cdots, L, \end{cases}$$

where  $B_l(x): \Re^n \to \Re^{r \times r}, l = 1, 2, \dots, L$ , are  $r \times r$  continuous negative definite symmetric matrix-valued functions,  $C_l(x), p_{j,l}(x): \Re^n \to \Re^r, l = 1, 2, \dots, L, j = 1, 2, \dots, J$ ,

are continuous vector-valued functions,  $d_l(x)$ ,  $q_{j,l}(x) : \Re^n \to \Re$ ,  $l = 1, 2, \dots, L$ ,  $j = 1, 2, \dots, J$ , are continuous real-valued functions. Obviously, Assumption F1 holds. It is clear that the equation  $\Phi_l(\varepsilon^2, x, u_l, y_l) = 0$  can be written as

$$\begin{cases} B_l(x)u_l + C_l(x) - P_l(x)y_l = 0, \\ \phi_{\varepsilon^2}((y_l)_1, -h_1^l(x, u_l)) = 0, \\ \vdots \\ \phi_{\varepsilon^2}((y_l)_J, -h_J^l(x, u_l)) = 0, \end{cases}$$

where  $P_l(x) = [p_{1,l}(x), \cdots, p_{J,l}(x)]$ . Furthermore, we obtain

$$\begin{cases} \phi_{\varepsilon^{2}}\left((y_{l})_{1}, -p_{1,l}(x)^{T}B_{l}(x)^{-1}(P_{l}(x)y_{l} - C_{l}(x)) - q_{1,l}(x)\right) = 0, \\ \vdots \\ \phi_{\varepsilon^{2}}\left((y_{l})_{J}, -p_{J,l}(x)^{T}B_{l}(x)^{-1}(P_{l}(x)y_{l} - C_{l}(x)) - q_{J,l}(x)\right) = 0. \end{cases}$$

$$(5.5.32)$$

Write

$$w^l(x) = M_l(x)y_l + z^l(x),$$

where

$$M_l(x) = -P_l(x)^T B_l(x)^{-1} P_l(x), \quad z^l(x) = P_l(x)^T B_l(x)^{-1} C_l(x) - q_l(x)$$

and

$$q_l(x) = (q_{1,l}(x), \cdots, q_{J,l}(x))^T.$$

Then (5.5.32) can be further rewritten as

$$\begin{cases} M_{l}(x)y_{l} + z^{l}(x) - w^{l}(x) = 0, \\ \phi_{\varepsilon^{2}}((y_{l})_{1}, w_{1}^{l}(x)) = 0, \\ \vdots \\ \phi_{\varepsilon^{2}}((y_{l})_{J}, w_{J}^{l}(x)) = 0. \end{cases}$$

We discuss Assumption F3 in the following two cases.

(1) If  $P_l(x)$  is nonsingular for any x, then  $M_l(x)$  is a positive definite matrix. Hence, the equation  $\Phi_l(\varepsilon^2, x, u_l, y_l) = 0$  has a unique solution

$$\begin{cases} y_l(\varepsilon^2, x) = ((y_l(\varepsilon^2, x))_1, \cdots, (y_l(\varepsilon^2, x))_J)^T, \\ u_l(\varepsilon^2, x) = B_l(x)^{-1} (P_l(x)y_l(\varepsilon^2, x) - C_l(x)). \end{cases}$$

In particular, if

$$P_l(x)^T B_l(x)^{-1} P_l(x) = \operatorname{diag}(\lambda_{1,l}(x), \cdots, \lambda_{J,l}(x)),$$

where  $\lambda_{j,l}(x) < 0$  for  $j = 1, 2, \dots, J$ ,  $l = 1, 2, \dots, L$ , then, for every  $l = 1, 2, \dots, L$ , the unique solution of equation  $\Phi_l(\varepsilon^2, x, u_l, y_l) = 0$  is given by

$$\begin{cases} y_l(\varepsilon^2, x) = ((y_l(\varepsilon^2, x))_1, \cdots, (y_l(\varepsilon^2, x))_J)^T, \\ u_l(\varepsilon^2, x) = B_l(x)^{-1} (P_l(x)y_l(\varepsilon^2, x) - C_l(x)), \end{cases}$$

where

$$(y_l(\varepsilon^2, x))_j = \frac{-\bar{q}_{j,l}(x) - \sqrt{(\bar{q}_{j,l}(x))^2 - 4\lambda_{j,l}(x)\varepsilon^4}}{2\lambda_{j,l}(x)}$$

and

$$\bar{q}_{j,l}(x) = q_{1,l}(x) - p_{j,l}(x)^T B_l(x)^{-1} C_l(x), \quad l = 1, 2, \cdots, L.$$

(2) In addition, suppose  $M_l(x)$  is an R<sub>0</sub>-matrix if  $P_l(x)$  is singular. Since  $M_l(x)$  is a positive semidefinite matrix, by Corollary 3.9 in [87], the equation  $\Phi_l(\varepsilon^2, x, u_l, y_l) = 0$  also has a unique solution. On the other hand, it is clear that

$$A_l(x) = \nabla^2_{u_l u_l} g^l(x, u_l) - \sum_{j=1}^J (y_l)_j \nabla^2_{u_l u_l} h^l_j(x, u_l) = B_l(x), \quad l = 1, 2, \cdots, L,$$

are negative definite. Consequently, by Lemma 5.3.1 (2),  $\nabla_{(u_l,y_l)} \Phi_l(\varepsilon^2, x, u_l, y_l)$  is nonsingular for any  $\varepsilon > 0$  and  $(x, u_l, y_l)$ . Furthermore, since  $\Phi_l(\varepsilon^2, x, u_l, y_l)$  is continuously differentiable with respect to  $(u_l, y_l)$  for any  $\varepsilon > 0$ , it follows from the Implicit Function theorem [108, Theorem 5.2.4] that  $y_l(\varepsilon^2, x)$  and  $u_l(\varepsilon^2, x)$  are continuously differentiable. Therefore, Assumption F3 is satisfied. Furthermore, under certain conditions, Assumption F2 can be satisfied.

# Chapter 6

# <sup>4</sup>/<sub>3</sub>-Order Convergence of the Generalized Newton Method for Solving the No-Arbitrage Option Price Interpolation Problem

### 6.1 Introduction

Recently, Wang, Yin and Qi [179] developed a no-arbitrage interpolation method to preserve the shape of the option price function. The interpolation is optimal in terms of minimizing the distance between the implied risk-neutral density and a prior approximation function in  $L^2$ -norm, which is very important when only a few observations are available.

Since the seminal paper of Black-Scholes [10], numerous theoretical and empirical studies have been done on the no-arbitrage pricing theory, see Duffie [36] and the references therein. If the uncertainty of nature can be described by a stochastic process  $q_t$ , then the absence of arbitrage opportunities implies that there exists a state-price density (SPD) or risk-neutral density, which is denoted by  $p(q_{t_2}|F_{t_1})$ , where  $t_2$  is any time after time  $t_1$ ,  $F_{t_1}$  is all the information available at time  $t_1$ . The price of any financial security can be expressed as the expected net present value of future payoffs, where the expectation is taken with respect to the risk-neutral density. In the call option pricing case, the underlying asset price  $S_t$  can be used as the state variable, the risk-free rate is considered as a constant. So the price at time t is

$$C(S_t, s, \tau, r_{t,\tau}) = e^{-r_{t,\tau}\tau} \int_0^\infty (S_T - s)_+ p(S_T | S_t, \tau, r_{t,\tau}) dS_T,$$
(6.1.1)

where  $S_t$  is the underlying asset price at time t, s is the strike price of the option contract,  $\tau$  is the time-to-expiration,  $T = t + \tau$  is the expiration time,  $r_{t,\tau}$  is the risk free rate from time t to  $T = t + \tau$ . No matter what kind of process of the underlying asset price  $S_t$  is, and whether the market is complete or not, the equation above always holds. Wang, Yin and Qi [179] proved that the option price function is convex with respect to the strike price s. Assuming that  $0 < a = s_0 < s_1 < \cdots < s_{n+2} = b < \infty$ , they formulated the following constrained no-arbitrage interpolation problem:

Min 
$$||f''(s) - h(s)||_2$$
  
s.t.  $f(s_i) = y_i, i = 1, 2, \cdots, n+2,$   
 $f''(s) \ge 0$  for a.e.  $s \in [a, b], f \in W_2^2[a, b]$  (6.1.2)

where

$$h(s) = e^{-r_{t,\tau}\tau} \frac{1}{s\sigma\sqrt{2\pi\tau}} \exp\left\{-\frac{\left(\log s - \log S_t - r_{t,\tau}\tau + \sigma^2\tau/2\right)^2}{2\sigma^2\tau}\right\}.$$
 (6.1.3)

By using the duality theory and Lagrange multipliers as well as the normalized *B*-splines  $B_i$  of order two associated with  $(s_i, y_i)$ , Wang, Yin and Qi [179] converted the minimization problem (6.1.2) to a system of nonsmooth equations

$$F(x) = d, (6.1.4)$$

where  $d = (d_1, d_2, \dots, d_n)^T$  are the second divided differences,  $F = (F_1, F_2, \dots, F_n)^T$ :  $\Re^n \to \Re^n$  and the *i*-th component of F is defined by

$$F_i(x) = \int_a^b \left( \sum_{l=1}^n x_l B_l(s) + h(s) \right)_+ B_i(s) ds.$$
 (6.1.5)

We may see that  $F_i$  belongs to a class of integral functions defined in (2.1.1). Wang, Yin and Qi [179] applied the following generalized Newton method to solve (6.1.4):

$$M(x^k)(x^{k+1} - x^k) = -F(x^k) + d, \ k = 1, 2, \cdots,$$
(6.1.6)

where  $M(x) \in \Re^{n \times n}$  with components

$$M_{ij}(x) = \int_{a}^{b} \mathbf{1}_{(0,\infty)} \left( \sum_{l=1}^{n} x_{l} B_{l}(s) + h(s) \right) B_{i}(s) B_{j}(s) ds,$$

where  $\mathbf{1}_{(0,\infty)}(\cdot)$  is the characteristic function of the set  $(0, +\infty)$ , i.e.,  $\mathbf{1}_{(0,\infty)}(x) = 1$  for x > 0 and  $\mathbf{1}_{(0,\infty)}(x) = 0$  for  $x \le 0$ . It is not difficult to see that (6.1.6) can be written as:

$$M(x^{k})x^{k+1} = d - \int_{a}^{b} h(s)B(s)ds, \ k = 1, 2, \cdots,$$
(6.1.7)

where  $B(s) = (B_1(s), B_2(s), \dots, B_n(s))^T$ .

By applying the results of [33], Wang, Yin and Qi [179] proved semismoothness of the integral function F defined by (6.1.5), and hence established superlinear convergence of the generalized Newton method (6.1.7). However, Wang, Yin and Qi [179] has not proved strong semismoothness of the integral function F defined by (6.1.5), and hence has not established quadratic convergence of (6.1.7). Actually, they give a counter example that the integral function defined by (2.1.1) may not be strongly semismooth, also see [136].

This raises two questions:

1. When does the generalized Newton method (6.1.7) have quadratic convergence?

2. If in some cases the generalized Newton method (6.1.7) has no quadratic convergence, what is its convergence rate?

In this chapter, we first give the answer to these two questions for the generalized Newton method (6.1.7). And then we present a globalized algorithm for solving the no-arbitrage option price interpolation problem (6.1.2).

This chapter is organized as follows. In Section 6.2, by using the *p*-order semismoothness results of the integral function (2.1.1), we show that the generalized Newton method (6.1.7) has at least  $\frac{4}{3}$ -order convergence rate and we give conditions when this method has  $\frac{3}{2}$ -order and quadratic convergence rate. In Section 6.3, we propose a damped version of the generalized Newton method and show that it is globally convergent and the convergence order is at least  $\frac{4}{3}$ .

# 6.2 Convergence Analysis

In this section, we first apply Theorem 2.3.2 to the integral function  $F(\cdot)$  defined by (6.1.5), then we show that the generalized Newton method (6.1.7) has at least  $\frac{4}{3}$ -order convergence. Some conditions when this method has  $\frac{3}{2}$ -order and quadratic convergence are also given.

Recall that the *B*-spline  $B_i$  is given by

$$B_{i}(s) = \begin{cases} \alpha_{i}(s-s_{i}), & \text{for } s \in [s_{i}, s_{i+1}], \\ \bar{\alpha}_{i}(s_{i+2}-s), & \text{for } s \in [s_{i+1}, s_{i+2}], \\ 0, & \text{otherwise}, \end{cases}$$

where

$$\alpha_i = 2/((s_{i+2} - s_i)(s_{i+1} - s_i)), \ \bar{\alpha}_i = 2/((s_{i+2} - s_i)(s_{i+2} - s_{i+1})).$$

In the sequel, we study the following functions:

$$\Phi_{1}(x_{1}) = \int_{s_{1}}^{s_{2}} (x_{1}B_{1}(s) + h(s))_{+} B_{1}(s)ds,$$
  

$$\Phi_{2}(x_{n}) = \int_{s_{n+1}}^{s_{n+2}} (x_{n}B_{n}(s) + h(s))_{+} B_{n}(s)ds,$$
  

$$\Gamma_{i}(x_{i-1}, x_{i}) = \int_{s_{i}}^{s_{i+1}} (x_{i-1}B_{i-1}(s) + x_{i}B_{i}(s) + h(s))_{+} B_{i}(s)ds, \ i = 2, \cdots, n,$$
  

$$\Psi_{i}(x_{i}, x_{i+1}) = \int_{s_{i+1}}^{s_{i+2}} (x_{i}B_{i}(s) + x_{i+1}B_{i+1}(s) + h(s))_{+} B_{i}(s)ds, \ i = 1, \cdots, n-1.$$

Then

$$F_1(x) = \Phi_1(x_1) + \Psi_1(x_1, x_2),$$
  

$$F_i(x) = \Gamma_i(x_{i-1}, x_i) + \Psi_i(x_i, x_{i+1}), i = 2, \cdots, n-1,$$
  

$$F_n(x) = \Gamma_n(x_{n-1}, x_n) + \Phi_2(x_n).$$

We may use Theorem 2.3.2 to establish *p*-order semismoothness of  $\Phi_i$ ,  $\Gamma_i$  and  $\Psi_i$ . This implies *p*-order semismoothness of *F*. Since the  $h(\cdot)$  possesses a very special structure, it has at most two inflection points. For any  $\bar{x} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)^T \in \Re^n$ , we assume, separating [a, b] more finely if necessary, that  $\sum_{l=1}^n \bar{x}_l B_l(s) + h(s) = 0$  has at most a root on  $[s_i, s_{i+1}]$ . So, the position relation between the line segment  $y = \sum_{l=1}^n \bar{x}_l B_l(s)$  on  $[s_i, s_{i+1}]$  and the curve y = -h(s) has four cases: (1) not intersected, (2) intersected but not tangent, (3) tangent at a convex or concave arc point of the curve y = -h(s), and (4) tangent at an inflection point of the curve y = -h(s). On  $[s_i, s_{i+1}]$ , the first two cases happen if and only if the root of  $\sum_{l=1}^n \bar{x}_l B_l(s) + h(s) = 0$  is simple on this subinterval. The third case happens if and only if  $\sum_{l=1}^{n} \bar{x}_l B_l(s) + h(s) = 0$  has a 2-order root on this subinterval. The fourth case happens if and only if the order of root of  $\sum_{l=1}^{n} \bar{x}_l B_l(s) + h(s) = 0$  on this subinterval is 3. Therefore, the following result immediately follows Theorem 2.3.2, which strengthens Theorem 4.6 in [179].

**Theorem 6.2.1** Consider the integral function  $F(\cdot)$  defined by (6.1.5). For any  $\bar{x} \in \Re^n$ , there exist exactly the following three cases:

(1) If the roots of  $\sum_{l=1}^{n} \bar{x}_l B_l(s) + h(s) = 0$  are simple on every subinterval  $[s_i, s_{i+1}] \subset [a, b], i = 1, 2, \dots, n+1$ , then F is 1-order (strongly) semismooth at  $\bar{x}$ .

(2) If there exists a certain subinterval  $[s_i, s_{i+1}] \subset [a, b]$  such that the highest order of roots of  $\sum_{l=1}^{n} \bar{x}_l B_l(s) + h(s) = 0$  on  $[s_i, s_{i+1}]$  is 2, then F is  $\frac{1}{2}$ -order semismooth at  $\bar{x}$ .

(3) If there exists a certain subinterval  $[s_i, s_{i+1}] \subset [a, b]$  such that the highest order of roots of  $\sum_{l=1}^{n} \bar{x}_l B_l(s) + h(s) = 0$  on  $[s_i, s_{i+1}]$  is 3, then F is  $\frac{1}{3}$ -order semismooth at  $\bar{x}$ .

Let the matrix function  $M = (M_{ij})_{n \times n} : \Re^n \to \Re^{n \times n}$  be defined by

$$M_{ij}(x) = \int_{a}^{b} \mathbf{1}_{(0,\infty)} \left( \sum_{l=1}^{n} x_{l} B_{l}(s) + h(s) \right) B_{i}(s) B_{j}(s) ds, \ i, j = 1, 2, \cdots, n.$$

In order to establish the (1+p)-order convergence of the generalized Newton method (6.1.7), we give the following lemma at first.

**Lemma 6.2.1** For any  $\bar{x} \in \Re^n$ , function  $M(\cdot)$  is continuous at  $\bar{x}$ .

**Proof.** Take any  $h \in \Re^n$ . We only need to prove  $M_{ij}(\bar{x} + h) \to M_{ij}(\bar{x})$  as  $h \to 0$  for  $i, j = 1, 2, \dots, n$ . It is easy to know that

$$|M_{ij}(\bar{x}+h) - M_{ij}(\bar{x})| \le \int_{\Omega_2(\bar{x},h)} B_i(s) B_j(s) ds.$$

By using the same proving technique in the proof of Theorem 2.3.2, we know that  $\mu(\Omega_2(\bar{x},h)) \leq O\left(\|h\|^{\frac{1}{3}}\right)$  since  $\sum_{l=1}^n \bar{x}_l B_l(s) + h(s) = 0$  has only a finite number of roots. Hence,  $|M_{ij}(\bar{x}+h) - M_{ij}(\bar{x})| \leq O\left(\|h\|^{\frac{1}{3}}\right)$  because  $B_i(s)B_j(s)$  is bounded on [a, b]. By this, we obtain the desired result.

Let  $x^*$  be the solution of (6.1.4), then  $M(x^*)$  is positive definite, see [179]. Therefore, we may obtain, from the above lemma and Theorem 6.2.1, the following result.

**Lemma 6.2.2** (i) Suppose that  $x^*$  is a solution of (6.1.4). Then there exist constant  $c_1$  and  $\delta_1 > 0$  such that any matrix M(x) is nonsingular and

$$\max\{\|M(x)\|, \|M(x)^{-1}\|\} \le c_1$$

for all x with  $||x - x^*|| \leq \delta_1$ .

(ii) There exist constants  $c_2$  and  $\delta_2$  such that

$$c_2 ||x - x^*|| \le ||F(x) - F(x^*)|| \le ||x - x^*|| / c_2$$

for all x with  $||x - x^*|| \leq \delta_2$ .

Now we state and prove a convergence property of the generalized Newton method for solving the no-arbitrage option price interpolation problem.

**Theorem 6.2.2** Let  $x^*$  be a solution of (6.1.4). Then there exist exactly the following three cases:

(1) If the roots of  $\sum_{l=1}^{n} x_l^* B_l(s) + h(s) = 0$  are simple on every subinterval  $[s_i, s_{i+1}] \subset [a, b], i = 1, 2, \dots, n+1$ , then any sequence  $\{x^k\}$  generated by the generalized Newton method (6.1.7) converges quadratically to  $x^*$  provided that the initial point  $x^0$  is sufficiently close to  $x^*$ .

(2) If there exists a certain subinterval  $[s_i, s_{i+1}] \subset [a, b]$  such that the highest order of roots of  $\sum_{l=1}^{n} x_l^* B_l(s) + h(s) = 0$  on  $[s_i, s_{i+1}]$  is 2, then the convergence of (6.1.7) is of order  $\frac{3}{2}$  provided that the initial point  $x^0$  is sufficiently close to  $x^*$ .

(3) If there exists a certain subinterval  $[s_i, s_{i+1}] \subset [a, b]$  such that the highest order of roots of  $\sum_{l=1}^{n} x_l^* B_l(s) + h(s) = 0$  on  $[s_i, s_{i+1}]$  is 3, then the convergence of (6.1.7) is of order  $\frac{4}{3}$  provide that the initial point  $x^0$  is sufficiently close to  $x^*$ .

**Proof.** According to Theorem 1.1.1, we need to verify the following three conditions:

(1)  $M(x) \in \partial F(x);$ 

(2) There exists  $\delta_1 > 0$  such that matrix M(x) is nonsingular for all x with  $||x - x^*|| \le \delta_1$ ;

(3) F is strongly semismooth,  $\frac{1}{2}$ ,  $\frac{1}{3}$ -order semismooth at  $x^*$ , respectively, in corresponding three cases.

The first property is proved in [179]. The second property is proved in Lemma 6.2.2 (i). The property (3) is established in Theorem 6.2.1. Hence, we obtain the desired results. ■

In the rest of this section, we prove a local (1 + p)-order decrease in the residual function  $||F(\cdot) - d||$ . For the sake of conciseness, We only discuss the worst case that F is  $\frac{1}{3}$ -order semismooth at  $x^*$ . In fact, we have also corresponding results in other two cases.

**Theorem 6.2.3** Let  $x^*$  be a solution of (6.1.4). If there exists a certain subinterval  $[s_i, s_{i+1}] \subset [a, b]$  such that the highest order of roots of  $\sum_{l=1}^n x_l^* B_l(s) + h(s) = 0$  on  $[s_i, s_{i+1}]$  is 3. Then there exists constants c such that for the sequence  $\{x^k\}$  obtained from (6.1.7)

$$\left\|F(x^{k+1}) - d\right\| \le c \left\|F(x^k) - d\right\|^{\frac{4}{3}},$$

provided that the initial point  $x^0$  is sufficiently close to  $x^*$ .

**Proof.** Without loss of generality, let  $\{x^k\}$  be generated by (6.1.7) with an initial point  $x^0$  sufficiently close to  $x^*$  so that Theorem 6.2.2 and Lemma 6.2.2 hold. By Theorem 6.2.2 (3), there exists a constant  $c_3 > 0$  such that

$$\left\|x^{k+1} - x^*\right\| \le c_3 \left\|x^k - x^*\right\|^{\frac{4}{3}}$$
(6.2.1)

 $\frac{4}{3}$ 

for all k. Then we have, by Lemma 6.2.2, for all  $x^k$ ,

$$\begin{aligned} \left\| F(x^{k+1}) - d \right\| &= \left\| F(x^{k+1}) - F(x^*) \right\| \\ &\leq \left\| x^{k+1} - x^* \right\| / c_2 \\ &\leq \left( c_3 / c_2 \right) \left\| x^k - x^* \right\|^{\frac{4}{3}} \\ &\leq \left( c_3 / \left( c_2^2 \right) c \right) \left\| F(x^k) - F(x^*) \right\| \\ &= \left( c_3 / \left( c_2^2 \right) c \right) \left\| F(x^k) - d \right\|^{\frac{4}{3}}. \end{aligned}$$

We obtain the desired result.  $\blacksquare$ 

# 6.3 A Damped Version of the Generalized Newton Method

Wang, Yin and Qi [179] also proved, by a series of theoretical analysis, that the dual problem of (6.1.2) is

$$\max_{x \in \mathbb{R}^n} -\frac{1}{2} \int_a^b \left( \sum_{l=1}^n x_l B_l(s) + h(s) \right)_+^2 ds + \frac{1}{2} \int_a^b (h(s))^2 ds + \sum_{l=1}^n x_l d_l.$$
(6.3.1)

By deleting the constant item  $\frac{1}{2} \int_a^b (h(s))^2 ds$  which does not change the problem, (6.3.1) can be written as follows

$$\max_{x \in R^n} -\frac{1}{2} \int_a^b \left( \sum_{l=1}^n x_l B_l(s) + h(s) \right)_+^2 ds + \sum_{l=1}^n x_l d_l$$
(6.3.2)

In the remaining part of this section, we study a globalized version of the Newton method applied to the negative counterpart of the dual function (6.3.2)

$$L(x) = \frac{1}{2} \int_{a}^{b} \left( \sum_{l=1}^{n} x_{l} B_{l}(s) + h(s) \right)_{+}^{2} ds - \sum_{l=1}^{n} x_{l} d_{l}.$$
(6.3.3)

Note that  $\nabla L(x) = F(x) - d$ . We first have the following result which shows that L is coercive, i.e.,  $L(x) \to +\infty$  as  $||x|| \to \infty$ . This result extends Lemma 2.1 in [32] to the case  $h(x) \neq 0$ . Thus, any method that produces a minimizing sequence for (6.3.3) is convergent since that L(x) is (strictly) convex too.

**Lemma 6.3.1** The level set  $Lev(c) = \{x \in \Re^n : L(x) \le c\}$  is compact for any  $c \in \Re$ .

**Proof.** Assume on the contrary that  $Lev(c_0)$  is unbounded for some  $c_0$  and let, without loss of generality, that  $c_0 > \frac{1}{2} \int_a^b (h(s))^2 ds$ . We show first that there is a vector  $\bar{x} \in$  $\Re^n \setminus \{0\}$  such that  $t\bar{x} \in Lev(c_0)$  for every  $t \ge 0$ . Suppose that for every  $x \in \Re^n \setminus \{0\}$ there exists  $t(x) \ge 0$  such that  $t(x)x \notin Lev(c_0)$  Since L(x) is a convex function, Lev(c)is convex for any  $c \in R$ . From the convexity of  $Lev(c_0)$  and  $0 \in Lev(c_0)$ , it follows that  $tx \notin Lev(c_0)$  whenever  $t \ge t(x)$ . Let

$$T(x) = \sup \{t : t \ge 0, tx \in Lev(c_0)\}.$$

Then  $T(x) < +\infty$ . Since  $Lev(c_0)$  is closed and convex, it is easy to prove that  $T(\cdot)$  is upper semicontinuous function over  $\Re^n \setminus \{0\}$ . Then,

$$T^* = \sup \{T(x) : \|x\| = 1\} < +\infty.$$

Hence,  $Lev(c_0)$  is contained in a ball centered at the origin with radius  $T^*$ , a contradiction.

Define

$$w(t) = \frac{1}{2} \int_{a}^{b} \left( t \sum_{l=1}^{n} \bar{x}_{l} B_{l}(s) + h(s) \right)_{+}^{2} ds - t \sum_{l=1}^{n} \bar{x}_{l} d_{l}.$$

We obtain that  $w(t) \leq c_0$  whenever  $t \geq 0$ . If  $\bar{x}_l \leq 0$  for all  $l = 1, 2, \dots, n$ , then

$$w(t) \ge -t \sum_{l=1}^{n} \bar{x}_l d_l > c_0$$

for t large enough. Hence there exists an index  $\bar{l}$  such that  $\bar{x}_{\bar{l}} > 0$ . Then

$$R := \frac{1}{2} \int_{a}^{b} \left( t \sum_{l=1}^{n} \bar{x}_{l} B_{l}(s) + h(s) \right)_{+}^{2} ds$$
  
$$\geq \frac{1}{2} \int_{a}^{b} \left( \sum_{l=1}^{n} \bar{x}_{l} B_{l}(s) + h(s) \right)_{+}^{2} ds > 0$$

and  $w(t) = t^2 R - t \sum_{l=1}^{n} \bar{x}_l d_l \to +\infty$  as  $t \to +\infty$ , a contradiction. The proof is complete.

The following algorithm is a "damped" globalization of Newton's method based on regularization controlled by the residual.

#### Algorithm 6.3.1 (Damped Newton method)

- **Step 0.** (Initialization) Choose  $x^0 \in \Re^n$ ,  $\rho \in (0, 1)$ ,  $\sigma \in (0, \frac{1}{2})$ , and tolerance tol> 0. Set k := 0.
- Step 1. (Termination criterion) If  $\varepsilon_k = \|F(x^k) d\| \le tol$  then stop. Otherwise, go to Step 2.
- **Step 2.** (Direction generation) Let  $s^k$  be a solution of the following linear system

$$(M(x^k) + \varepsilon_k I)s = -\nabla L(x^k). \tag{6.3.4}$$

**Step 3.** (Line search) choose  $m_k$  as the smallest nonnegative integer m satisfying

$$L(x^{k} + \rho^{m} s^{k}) - L(x^{k}) \le \sigma \rho^{m} \nabla L(x^{k})^{T} s^{k}.$$
(6.3.5)

**Step 4.** (Update) Set  $x^{k+1} = x^k + \rho^{m_k} s^k$ , k := k + 1, return to Step 1.

If tol= 0, then Algorithm 6.3.1 will produces a infinite sequence of iterates  $\{x^k\}$ in general, otherwise, some iterate  $x^k$  is the solution of (6.1.4). In what follows, we assume that for any k,  $x^k$  is not the solution of (6.1.4). Since  $M(x^k)$  is always positive semidefinite, see [179],  $M(x^k) + \varepsilon_k I$  is always positive definite for any  $\varepsilon_k > 0$ . Therefore, the linear system (6.3.4) is uniquely solvable and  $s^k \neq 0$ . If there are no nonnegative integers satisfying (6.3.5), then we have

$$\lim_{m \to \infty} \frac{L(x^k + \rho^m s^k) - L(x^k)}{\rho^m} \ge \sigma \left(\nabla L(x^k)\right)^T s^k,$$

i.e.,

$$0 \le (1-\sigma)(\nabla L(x^k))^T s^k \le -(1-\sigma)\varepsilon_k \left\|s^k\right\|^2 < 0$$

The contradiction means that there is always an  $m_k$  satisfying (6.3.5), i.e.,  $x^{k+1}$  can be calculated from  $x^k$ . We formally state this result as follows.

**Proposition 6.3.1** Let tol= 0. For every starting point  $x^0$ , Algorithm 6.3.1 generates an infinite sequence  $x^k$ .

In the proof of convergence of Algorithm 6.3.1, we use the following technical lemma which shows that the unit stepsize is attained eventually.

**Lemma 6.3.2** Let  $x^*$  be a solution of (6.1.4). For every  $\delta \in (0,1)$ , there exists a neighborhood U of  $x^*$  and a scalar  $\bar{\varepsilon} > 0$  such that, for any  $x \in U$  and  $\varepsilon \in [0, \bar{\varepsilon}]$ , the unique solution  $s_x$  of the linear system

$$(M(x) + \varepsilon I)s = -\nabla L(x) \tag{6.3.6}$$

satisfies  $x + s_x \in U$  and

$$L(x + s_x) - L(x) - \frac{1}{2} \nabla L(x)^T s_x \le \delta ||s_x||^2.$$

**Proof.** By Lemma 6.2.2, there exists a neighborhood U of  $x^*$  such that M(x) is nonsingular for any  $x \in U$ . Hence the linear equation (6.3.6) has a unique solution for all  $\varepsilon \geq 0$ . Since L has a semismooth gradient, the desired result follows from Lemmas 3.1 and 3.3 in [113].

We now state and prove a convergence property of Algorithm 6.3.1, which shows that it is globally convergent and the convergence order is at least  $\frac{4}{3}$ .

**Theorem 6.3.1** Let  $x^0 \in \Re^n$  and  $\{x^k\}$  be generated by Algorithm 6.3.1. Then the whole sequence  $\{x^k\}$  converges to the solution  $x^*$  of (6.1.4), and the convergence is at least of order  $\frac{4}{3}$ .

**Proof.** Since the sequence  $L(x^k)$  is decreasing and the function L is coercive, see Lemma 6.3.1, the sequence of iterates  $\{x^k\}$  is bounded. Let  $x^{**}$  be a limit of a subsequence  $\{x^k\}_k, k \in K$ . We will prove that  $x^{**} = x^*$  and hence  $x^k$  is convergent to  $x^*$ . Assume on the contrary that  $x^{**} \neq x^*$ . Then  $\varepsilon_* := ||F(x^{**}) - d|| > 0$ , since we know that, from the (strictly) convex of L(x), the solution  $x^*$  of (6.1.4) is unique. By Lemma 6.2.1,  $M(x^k) \to M(x^{**}) \in \Re^{n \times n}$ . It is clear that  $M(x^{**})$  is positive semidefinite.

Let  $s^* \in \Re^n$  be the solution of the linear system:

$$(M(x^{**}) + \varepsilon_*I)s = -F(x^{**}) + d.$$

Under the above assumptions, we have that  $s^* \neq 0$  and  $s^k \to s^*$  as  $k \in K$ ,  $k \to \infty$ . Then for  $k \in K$  sufficiently large, we have

$$L(x^{k} + ts^{k}) - L(x^{k}) - \sigma t(\nabla L(x^{k}))^{T}s^{k} = (1 - \sigma)t(\nabla L(x^{k}))^{T}s^{k} + o(t)$$
  
=  $(1 - \sigma)t(\nabla L(x^{**}))^{T}s^{*} + o(t) + (1 - \sigma)t\alpha_{k}$ ,

where  $\alpha_k = (\nabla L(x^k))^T s^k - (\nabla L(x^{**}))^T s^*$ . It is obvious that  $\alpha_k \to 0$  as  $k \in K, k \to \infty$ . Note that  $(\nabla L(x^{**}))^T s^* = -(s^*)^T (M(x^{**}) + \varepsilon_* I) s^* < 0$ ; then there exists a nonnegative integer  $m_*$  such that for all  $k \in K$  sufficiently large and all  $m \ge m_*$ ,

$$L(x^k + \rho^m s^k) - L(x^k) - \sigma \rho^m (\nabla L(x^k))^T s^k \le 0.$$

For all  $k \in K$  sufficiently large this inequality implies that  $m_k \leq m_*$  and

$$L(x^{k+1}) - L(x^{k}) \le \sigma \rho^{m_{k}} (\nabla L(x^{k}))^{T} s^{k} \le \frac{1}{2} \sigma \rho^{m_{*}} (\nabla L(x^{**}))^{T} s^{*}$$

Hence,  $L(x^k) \to -\infty$ , contradicting the boundedness of  $\{L(x^k)\}$ . Therefore,  $x^{**} = x^*$ . Thus, we have  $x^k \to x^*$  and  $\varepsilon_k \to 0$ . Since  $M(x^*)$  is positive definite, from the continuity of  $M(\cdot)$  it follows that every  $M(x^k)$  is also positive definite for all  $x^k$  sufficiently close to  $x^*$ . Then there exists a constant  $\overline{\delta} > 0$  such that

$$(s^k)^T (M(x^k) + \varepsilon_k I) s^k \ge \bar{\delta} \left\| s^k \right\|^2$$
(6.3.7)

for all  $x^k$  sufficiently close to  $x^*$ . Choose  $\delta$  in Lemma 6.3.2 such that  $\delta < (\frac{1}{2} - \sigma)\overline{\delta}$ . Noticing that  $\delta < \frac{1}{2}$ , for all k sufficiently large, Lemma 6.3.2 and (6.3.7) yield

$$\begin{split} L(x^{k} + s^{k}) &- L(x^{k}) - \sigma(\nabla L(x^{k}))^{T} s^{k} \\ &= L(x^{k} + s^{k}) - L(x^{k}) - \frac{1}{2} (\nabla L(x^{k}))^{T} s^{k} + (\frac{1}{2} - \sigma) (\nabla L(x^{k}))^{T} s^{k} \\ &\leq \delta \left\| s^{k} \right\|^{2} - (\frac{1}{2} - \sigma) (s^{k})^{T} (M(x^{k}) + \varepsilon_{k} I) s^{k} \\ &\leq -((\frac{1}{2} - \sigma) \overline{\delta} - \delta) \left\| s^{k} \right\|^{2}. \end{split}$$

That is, eventually  $m_k = 0$ . Hence  $x^{k+1} = x^k + s^k$  for all k sufficiently large. We note that  $\varepsilon_k \to 0$  and  $(M(x^k))^{-1}$  is uniformly bounded in a small neighborhood of  $x^*$ , see Lemma 6.2.2 (i). Hence, for all k sufficiently large

$$(I + \varepsilon_k (M(x^k))^{-1})^{-1} = I - \varepsilon_k (M(x^k))^{-1} + o(\varepsilon_k).$$

If  $\Delta x^k$  is Newton's direction generated by (6.1.6) at  $x^k$ , then we have

$$s^{k} = (M(x^{k}) + \varepsilon_{k}I)^{-1}(-F(x^{k}) + d)$$
  
=  $(I + \varepsilon_{k}(M(x^{k}))^{-1})^{-1}(M(x^{k}))^{-1}(-F(x^{k}) + d)$  (6.3.8)  
=  $\Delta x^{k} - \varepsilon_{k}(M(x^{k}))^{-1}\Delta x^{k} + o(\varepsilon_{k})\Delta x^{k}$ 

By Theorem 6.2.2, we obtain

$$\|x^{k} + \Delta x^{k} - x^{*}\| = O\left(\|x^{k} - x^{*}\|^{\frac{4}{3}}\right).$$
 (6.3.9)

Noticing that  $F(x^*) = d$ , Lemma 6.2.2 implies that  $\varepsilon_k = O(||x^k - x^*||)$ . From (6.3.9) we obtain  $||\Delta x^k|| = O(||x^k - x^*||)$ . Then it follows from (6.3.8) that

$$s^{k} = \Delta x^{k} + O\left(\left\|x^{k} - x^{*}\right\|^{2}\right).$$

Thus we have

$$\begin{aligned} \|x^{k+1} - x^*\| &= \|x^k + s^k - x^*\| \\ &\leq \|x^k + \Delta x^k - x^*\| + O\left(\|x^k - x^*\|^2\right) \\ &= O\left(\|x^k - x^*\|^{\frac{4}{3}}\right) \end{aligned}$$

In the last equality we employ (6.3.9). Hence, the rate of sequence  $\{x^k\}$  converging to the solution  $x^*$  of (6.1.4) is at least of order  $\frac{4}{3}$ .

# Chapter 7

# A Newton Type Method for $L_2$ Spectral Estimation

## 7.1 Introduction

A basic problem in spectral estimation is the estimation of a *power spectrum*, a measure  $\mu$  on  $\Re^n$ , with a known support, given a finite collection of measured correlations. This problem has many applications in a wide range of settings such as geophysics, radio astronomy, radar, sonar, and interference spectroscopy, see [5, 6, 15, 17, 90, 96, 97] and references therein. In many of these problems, the power spectrum  $\mu$  is represented by a density. Let  $K \subseteq \Re^n$  be a measure space with  $\sigma$ -finite positive measure dx, and let  $\mu$  be absolutely continuous with respect to dx with density

$$s(x) = \frac{d\mu}{dx}.$$

Then the problem becomes to find a nonnegative integrable function s(x) on K which vanishes on the complement of K, and exactly matches the observed correlations

$$r_k = \int_K s(x)e^{jkx}dx, \ k \in \Delta, \ j := \sqrt{-1},$$

where  $\Delta$  is a finite subset of  $\Re^n$  with  $0 \in \Delta$ ,  $\Delta = -\Delta$  and  $r_{-k} = \bar{r}_k$ , the complex conjugate of  $r_k$ , for  $k \in \Delta$ . Even if the problem described above is feasible, it may not have a unique solution. The task of spectral estimation method is to select one s(x) out of the ensemble of spectra satisfying the correlation matching, positivity, and spectral support constraints. This selection is usually done by optimizing some convex functionals, see Lang and McClellan [97]. One popular method used in spectral estimation is the maximum entropy method, for example see [14] and [15]. In the method one attempts to find a solution s(x) satisfying

$$\max \int_{K} \ln s(x) dx$$
  
s.t.  $r_{k} = \int_{K} s(x) e^{jkx} dx, k \in \Delta,$   
 $s(x) \ge 0.$  (7.1.1)

The usual form the solution takes is  $s^*(x) = \frac{1}{P(x)}$ , where *P* is a positive trigonometric polynomial. However, in some applications, see examples in [6], a strictly positive solution of the form  $\frac{1}{P(x)}$  fails to exist. Goodrich and Steinhardt [54] suggested an alternative way for selecting s(x) by formulating the following optimization problem in  $L_2$  norm, which is called  $L_2$  spectral estimation,

$$\min \int_{K} (s(x))^2 dx$$

$$s.t. \quad r_k = \int_{K} s(x) e^{jkx} dx, k \in \Delta,$$

$$s(x) \ge 0.$$

$$(7.1.2)$$

Under appropriate conditions, it is shown that the optimal solution of (7.1.2) has the form:  $s^*(x) = \max(0, P(x))$ , where P(x) is a trigonometric polynomial. Ben-Tal, Borwain and Teboulle [6] developed a duality theory for multi-dimensional  $L_p$  (1problem under the so-called Borwein-Wolkowicz constraint qualification (BWCQ), seealso Borwein and Lewis [12] and [13]. The dual problem obtained is a finite dimensionalconcave program and the optimal dual variables are exactly the parameters of the op $timal spectral density <math>s^*(x)$ . The authors of [6] indicated "the simple unconstrained nature of the dual problem seems appropriate for computational purposes and may lead to the construction of reliable algorithms for computing the  $L_p$  optimal spectral estimate". In that paper, they discretized the support K by a finite number of points and approximated the dual problem by a nonsmooth optimization problem which can be solved by existing nonsmooth optimization algorithms.

In this chapter, we study the  $L_2$  spectral estimation problem by Newton-type method. For simplicity, we focus our attention on the time series case, an important subclass of the spectral estimation problem. The results of the chapter can be extended to the multidimensional case, but would be more complicated. Suppose that  $K = [-\pi, \pi]$  and  $\Delta = \{-m, \dots, -1, 0, 1, \dots, m\}$ , where *m* is an integer. Given a finite sequence  $\{r_k\}$  with  $k \in \Delta$ , the problem is now to find a nonnegative finite integrable function  $s \ge 0$  which satisfies

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} s(x) e^{jkx} dx = r_k, \ k \in \Delta.$$
(7.1.3)

This is the so-called trigonometric moment problem, see [6, 54, 57, 97]. We can rewrite problem (7.1.2) in the time series case (7.1.3) in the form of real constraints, that is

$$\min \quad \frac{1}{2} \int_{-\pi}^{\pi} (s(x))^2 dx$$
  
s.t. 
$$\frac{1}{2\pi} \int_{-\pi}^{\pi} s(x) \cos kx dx = \operatorname{Re}(r_k), \quad k = 0, 1, \cdots, m,$$
  
$$\frac{1}{2\pi} \int_{-\pi}^{\pi} s(x) \sin kx dx = \operatorname{Im}(r_k), \quad k = 1, \cdots, m,$$
  
$$s(x) \ge 0, \quad s(x) \in L_2[-\pi, \pi].$$
 (7.1.4)

Here, for convenience, we multiply the objective function of (7.1.2) by 1/2 without changing the problem. In 1993, Cole and Goodrich [24] investigated the  $L_p$ -spectral estimation with an  $L_{\infty}$ -upper bound, they compare the numerical performance of Least Squares methods for (7.1.4) (they take K = [0, 1] in the paper), Newton's method for solving the dual problem in the form of (7.2.3) and Newton's method for the system of nonlinear equations generated from the dual problem. They find the last method does the best job of fitting the solution to the data. Potter [119] also obtained similar numerical results. However, there is no convergence results of the algorithm for the spectral estimation problem in the paper.

It is obvious that problem (7.1.4) is a convex optimization problem. The problem has a unique solution provided that the feasible set is nonempty. In order to derive the optimality conditions, we need certain constraint qualification. Since the nonnegative cone of  $L_2[-\pi,\pi]$  has empty interior, the commonly used Slater constraint qualification does not hold. However, by [6], the BWCQ (Borwein-Wolkowicz Constraint Qualification) holds if and only if the Toeplitz matrix M, defined by

$$M = [r_{l-k}]_{l,k=0}^m \tag{7.1.5}$$

is positive definite, see page 993 in [6]. We take it as a blank assumption in the sequel of this chapter.

For convenience of expression, we take

$$B_{i}(x) = \begin{cases} 1, & \text{for } i = 1 \\ 2\cos kx & \text{for } k = 1, 2, \cdots, m, \text{ and } i = 2k, \\ 2\sin kx & \text{for } k = 1, 2, \cdots, m, \text{ and } i = 2k + 1 \end{cases}$$
(7.1.6)

and

$$d_{i} = \begin{cases} 2\pi r_{0}, & \text{for } i = 1\\ 4\pi \operatorname{Re}(r_{k}) & \text{for } k = 1, 2, \cdots, m, \text{ and } i = 2k, \\ 4\pi \operatorname{Im}(r_{k}) & \text{for } k = 1, 2, \cdots, m, \text{ and } i = 2k + 1. \end{cases}$$
(7.1.7)

Then (7.1.4) can be written as

$$\min \int_{-\pi}^{\pi} (s(x))^2 dx$$
  
s.t. 
$$\int_{-\pi}^{\pi} s(x)B_i(x)dx = d_i, \qquad i = 1, 2, \cdots, 2m + 1, \qquad (7.1.8)$$
  
$$s(x) \ge 0, \quad s(x) \in L_2[-\pi, \pi].$$

The functions  $\{B_i(x) : i = 1, 2, \dots, 2m + 1\}$  defined by (7.1.6) are called the trigonometric basis functions on  $[-\pi, \pi]$ , m is called the order of the trigonometric basis  $\{B_i(x)\}_{i=1}^{2m+1}$ .

By using the Lagrange dual technique, we transform the problem (7.1.8) into a finite dimension maximization problem. Further, the problem is reformulated as a system of nonsmooth equations F(x) = d (see (7.2.7) in Section 7.2). We establish the differentiability properties of the generated nonlinear equations and introduce the Newton-type method for solving the problem. We prove that the order of convergence of the method is at least  $1 + \frac{1}{2m}$ . This result provides a theoretical justification for the numerical observations in [24] and [119]. Moreover, based on an observation that the dual problem of (7.1.4) is  $SC^1$ , i.e., the objective function is smooth with its gradient function semismooth, we produce a globally convergent damped generalized Newton method for solving the problem. Some preliminary numerical tests are implemented for illustrating the efficiency and robustness of the method.

The outline of this chapter is as follows. Section 7.2 contains the dual problem of (7.1.8) and its reformulation. In Section 7.3, we investigate the differentiability of function F generated from the reformulation (see (7.2.8) in Section 7.2) and introduce the Newton-type method for solving the problem. Then we prove the convergence of the method and study the rate of convergence. In Section 7.4, we give a damped Newtontype algorithm for solving the problem, which is globally convergent. Preliminary numerical test results are listed in Section 7.5.

### 7.2 Dual and Its Reformulation

In this section we establish the dual of problem (7.1.8), then the solving of the spectral estimation problem (7.1.2), which is an infinite dimensional moment problem, is transformed into solving a finite dimensional unconstrained maximization problem with a concave objective function. Furthermore, the unique solution of the dual problem is the solution of a system of nonlinear equations.

First, according to the Lagrange multiplier rule,  $s^*$  is the unique solution of (7.1.8) if and only if there exist numbers  $\lambda_i^*$ ,  $i = 1, 2, \dots, 2m + 1$  such that  $s^*$  is the solution of the problem

min 
$$L(s,\lambda^*) = \frac{1}{2} \int_{-\pi}^{\pi} (s(x))^2 dx - \sum_{i=1}^{2m+1} \lambda_i^* \left( \int_{-\pi}^{\pi} s(x) B_i(x) dx - d_i \right)$$
  
s.t.  $s \ge 0, \ s \in L_2[-\pi,\pi].$ 
(7.2.1)

Hence the solution of (7.1.8) has the form

$$s^*(x) = \left(\sum_{i=1}^{2m+1} \lambda_i^* B_i(x)\right)_+, \qquad (7.2.2)$$

where  $a_+ := \max\{a, 0\}, \lambda_i^*, i = 1, 2, \dots, 2m + 1$ , are Lagrange multipliers. By dual theorem, substituting (7.2.2) into (7.2.1), we obtain that the value of the Lagrange multiplier vector  $\lambda^* = (\lambda_1^*, \lambda_2^*, \dots, \lambda_{2m+1}^*)^T \in \Re^{2m+1}$  is a solution of the following dual problem

$$\max_{\lambda \in \Re^{2m+1}} -\frac{1}{2} \int_{-\pi}^{\pi} \left( \sum_{i=1}^{2m+1} \lambda_i B_i(x) \right)_{+}^2 dx + \sum_{i=1}^{2m+1} \lambda_i d_i.$$
(7.2.3)

Let  $\Lambda_0 = \lambda_1$ ,  $\Lambda_k = \lambda_{2k} + j\lambda_{2k+1}$  for  $k = 1, 2, \dots, m$ . Analogous to the discussion above, we have, in the complex form, the dual problem of (7.1.8) is

$$\max_{\Lambda} \left\{ 2\pi \sum_{k=-m}^{m} \bar{r}_k \Lambda_k - \frac{1}{2} \int_{-\pi}^{\pi} \left( \sum_{k=-m}^{m} \Lambda_k e^{-jkx} \right)_+^2 dx \right\},$$
(7.2.4)

where  $\Lambda = (\Lambda_{-m}, \dots, \Lambda_{-1}, \Lambda_0, \Lambda_1, \dots, \Lambda_m)^T$  is a complex vector and  $\Lambda_{-k} = \overline{\Lambda}_k (k = 1, 2, \dots, m)$ . Moreover, we have

$$2\pi \sum_{k=-m}^{m} \bar{r}_k \Lambda_k = 2\pi \left[ \lambda_1 r_0 + 2 \sum_{k=1}^{m} \left( \lambda_{2k} \operatorname{Re}(r_k) + \lambda_{2k+1} \operatorname{Im}(r_k) \right) \right] = \lambda^T d$$
(7.2.5)

and

$$\sum_{k=-m}^{m} \Lambda_k e^{-jkx} = \lambda_1 + 2\sum_{k=1}^{m} (\lambda_{2k} \cos kx + \lambda_{2k+1} \sin kx) = \lambda^T B(x), \quad (7.2.6)$$

where  $d = (d_1, d_2, \dots, d_{2m+1})^T$  and  $B(x) = (B_1(x), B_2(x), \dots, B_{2m+1}(x))^T$ . Therefore, problem (7.2.3) and (7.2.4) are equivalent.

From the analysis above we have the following result, which is a direct deduction of Theorem 5.1 in [6].

**Theorem 7.2.1** For given K,  $\Delta$ , and  $r_k$ , if the Toeplitz matrix M, defined by (7.1.5) is positive definite, then problem (7.1.8) has a unique solution  $s^*$ , its dual problem (7.2.3) has a unique solution  $\lambda^*$ , and the optimal values of (7.1.8) and (7.2.3) are equal.

Notice that (7.2.3) is a finite dimensional problem, then by dual technique, the infinite dimensional optimization problem (7.1.8) can be solved via solving an unconstrained finite dimensional concave optimization problem (7.2.3). Further, by the first order optimality condition and concavity of (7.2.3), the dual problem can be reformulated as the following equation system:

$$F(\lambda) = d, \tag{7.2.7}$$

where

$$F = \int_{-\pi}^{\pi} \left( \sum_{i=1}^{2m+1} \lambda_i B_i(x) \right)_+ B(x) dx$$
 (7.2.8)

with

$$F_i(\lambda) = \int_{-\pi}^{\pi} \left( \sum_{l=1}^{2m+1} \lambda_l B_l(x) \right)_+ B_i(x) dx, \ i = 1, 2, \cdots, 2m+1.$$
(7.2.9)

More precisely,

$$F_{0}(\lambda) = \int_{-\pi}^{\pi} \left( \sum_{i=1}^{2m+1} \lambda_{i} B_{i}(x) \right)_{+} dx,$$
  

$$F_{2k}(\lambda) = 2 \int_{-\pi}^{\pi} \left( \sum_{i=1}^{2m+1} \lambda_{i} B_{i}(x) \right)_{+} \cos kx dx, \ k = 1, 2, \cdots, m,$$
  

$$F_{2k+1}(\lambda) = 2 \int_{-\pi}^{\pi} \left( \sum_{i=1}^{2m+1} \lambda_{i} B_{i}(x) \right)_{+} \sin kx dx, \ k = 1, 2, \cdots, m.$$
  
(7.2.10)

We can see that function F in (7.2.7) may be not smooth. In the next section we study the differentiability of F and introduce a Newton algorithm for solving (7.2.7).

# 7.3 Some Properties of F

In this section we study the differentiability properties of  $F(\lambda)$  given by (7.2.8) with its components given by (7.2.9) for  $\lambda \in \Re^{2m+1}$ , which are crucial for the establishment and convergence analysis of the algorithm for solving (7.1.8). We denote

$$K_0(\lambda) = \left\{ x \in [-\pi, \pi] : B(x)^T \lambda = 0 \right\},$$
  

$$K_+(\lambda) = \left\{ x \in [-\pi, \pi] : B(x)^T \lambda > 0 \right\},$$
  

$$K_-(\lambda) = \left\{ x \in [-\pi, \pi] : B(x)^T \lambda < 0 \right\}.$$

Then we have the following properties of function F in (7.2.7).

**Proposition 7.3.1** (i) For any  $\lambda \in \Re^{2m+1}$ , the function F is locally Lipschitz continuous;

(ii) F is semismooth at  $\hat{\lambda} = 0$ ;

(iii) F is continuously differentiable at any  $\hat{\lambda} \in \Re^{2m+1} \setminus \{0\}$ , and

$$\nabla F(\hat{\lambda}) = \int_{K_{+}(\hat{\lambda})} A(x) dx, \qquad (7.3.1)$$

where

$$A(x) = B(x)B(x)^T$$

$$= \begin{pmatrix} 1 & 2\cos x & \cdots & 2\cos mx & 2\sin mx \\ 2\cos x & 4\cos^2 x & \cdots & 4\cos x\cos mx & 4\cos x\sin mx \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 2\cos mx & 4\cos x\cos mx & \cdots & 4\cos^2 mx & 4\cos mx\sin mx \\ 2\sin mx & 4\cos x\sin mx & \cdots & 4\cos mx\sin mx & 4\sin^2 mx \end{pmatrix}.$$
(7.3.2)

**Proof.** For convenience of expression, we denote

$$g(\lambda, x) = \lambda^T B(x)$$

and

$$f(\lambda, x) = (g(\lambda, x))_+.$$

Then for every  $x \in [-\pi, \pi]$ ,  $g(\lambda, x)$  is continuously differentiable with respect to  $\lambda \in \Re^{2m+1}$ .  $G_x(\lambda) := \nabla_{\lambda} g(\lambda, x) = B(x)$  is continuous on  $[-\pi, \pi] \times \Re^{2m+1}$ . Then by Theorem 2.2.1 for any  $i = 1, 2, \dots, 2m + 1$ ,  $F_i$  is locally Lipschitz continuous, directionally differentiable and

$$F'_{i}(\lambda;h) = \int_{-\pi}^{\pi} (g(\lambda,x))^{0}_{+}g(h,x)B_{i}(x)dx, \text{ for } i = 1, 2, \cdots, 2m+1,$$

where

$$(a)^{0}_{+} = \begin{cases} 1 & \text{if } a > 0, \\ 0 & \text{otherwies} \end{cases}$$

From Proposition 2.3.2, we have that the integral functions  $F_i(\lambda), i = 1, 2, \dots, 2m + 1$ , are semismooth at  $\hat{\lambda} = 0$ . It is known from [126] that if each component of F is semismooth, then F itself is semismooth.

Since the basis functions  $\{1, 2\cos x, 2\sin x, \dots, 2\cos mx, 2\sin mx\}$  are linearly independent on  $[-\pi, \pi]$ ,  $B(x) = (1, 2\cos x, 2\sin x, \dots, 2\cos mx, 2\sin mx)^T \neq 0$  for any integer m and any  $x \in [-\pi, \pi]$ , there exists a neighborhood  $U(\hat{\lambda})$  of  $\hat{\lambda}$  such that  $\sigma(K_0(\lambda)) = 0$  for  $\lambda \in U(\hat{\lambda})$ , where  $\sigma(W)$  is the measure of set W. This implies that  $\sigma(\{x \in K_0(\lambda) : G_x(\lambda) \neq 0\}) = 0$ . By Corollary 2.2.1, differentiable at  $\lambda \in U(\hat{\lambda})$ and

$$\nabla F_i(\lambda) = \int_{-\pi}^{\pi} (g(\lambda, x))^0_+ B(x) B_i(x) dx$$
  
= 
$$\int_{K_+(\lambda)}^{\pi} B(x) B_i(x) dx.$$
 (7.3.3)

Precisely,

$$\nabla F_1(\lambda) = \int_{K_+(\lambda)} B(x) dx, \qquad (7.3.4)$$

$$\nabla F_{2k}(\lambda) = 2 \int_{K_+(\lambda)} B(x) \cos kx dx, \qquad (7.3.5)$$

$$\nabla F_{2k+1}(\lambda) = 2 \int_{K_+(\lambda)} B(x) \sin kx dx, \qquad (7.3.6)$$

for  $k = 1, 2, \dots, m$ . By direct computation, and noticing that  $A(x) = B(x)B(x)^T$ , we have (7.3.1). Moreover, it is readily to prove that  $\chi_{K_+(\lambda)}(x) \to \chi_{K_+(\hat{\lambda})}(x)$  as  $\lambda \to \hat{\lambda}$  for a.e. x and  $|\chi_{K_+(\lambda)}(x)B(x)B_i(x)| \leq |B(x)B_i(x)|$ . By Lebesgue Dominated Convergence theorem, we obtain the continuous differentiability of F.

It is not difficult to have the following result.

**Proposition 7.3.2** For any  $\lambda \in \Re^{2m+1}$ , any  $V \in \partial F(\lambda)$  is positive semidefinite.

**Proof.** Define  $\theta : \Re^{2m+1} \to \Re$  as

$$\theta(\lambda) = \frac{1}{2} \int_{-\pi}^{\pi} (\lambda^T B(x))_+^2 dx.$$
(7.3.7)

 $\theta(\lambda)$  is a continuously defferentiable convex function. Its gradient is  $F(\lambda)$ . From Proposition 2.3 (a) in [82], any matrix in the generalized Jacobian of the gradient of a convex function must be positive semidefinite.

Now we prove the positive definiteness of  $\nabla F(\cdot)$  at the solution point  $\lambda^*$ .

**Theorem 7.3.1** Suppose that  $\lambda^* \in \Re^{2m+1}$  is a solution of (6.1.4). Then  $\lambda^* \neq 0$ , and  $\nabla F(\cdot)$  is positive definite at  $\lambda^*$ .

**Proof.** Suppose that  $\lambda^* = \left(\lambda_1^*, \lambda_2^*, \lambda_3^*, \cdots, \lambda_{2m}^*, \lambda_{2m+1}^*\right)^T$  is a solution of (6.1.4). Then  $\int_{-\pi}^{\pi} \left(B(x)^T \lambda^*\right)_+ dx = d_1,$   $2 \int_{-\pi}^{\pi} \left(B(x)^T \lambda^*\right)_+ \cos kx dx = d_{2k},$   $2 \int_{-\pi}^{\pi} \left(B(x)^T \lambda^*\right)_+ \sin kx dx = d_{2k+1}$ 

for  $k = 1, 2, \dots, m$ . Consequently,

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} \left( B(x)^T \lambda^* \right)_+ dx = r_0, 
\frac{1}{2\pi} \int_{-\pi}^{\pi} \left( B(x)^T \lambda^* \right)_+ e^{jkx} dx = r_k, 
\frac{1}{2\pi} \int_{-\pi}^{\pi} \left( B(x)^T \lambda^* \right)_+ e^{-jkx} dx = r_{-k}$$
(7.3.8)

for  $k = 1, 2, \dots, m$ . We prove that  $\lambda^* \neq 0$  by contradiction. Assume that  $\lambda^* = 0$ . Then from (7.3.8) we have  $r_i = 0$  for all  $i = -m, \dots, -1, 0, 1, \dots, m$ . Therefore,  $M = [r_{l-h}]_{l,h=0}^m = 0$ , which contradicts the blank assumption that M is positive definite. Therefore, we obtain that  $\lambda^* \in \Re^{2m+1} \setminus \{0\}$ .

Now we prove the last part of the theorem. Since  $\{1, 2 \cos x, 2 \sin x, \dots, 2 \cos mx, 2 \sin mx\}$ are linearly independent on  $[-\pi, \pi]$ ,  $A(x) = B(x)B(x)^T$  is positive semidefinite for any  $x \in [-\pi, \pi]$ . Therefore, for any  $\lambda \in \Re^{2m+1} \setminus \{0\}$ ,  $\lambda^T A(x)\lambda \ge 0$  and it does not vanish on any interval of  $[-\pi, \pi]$ . Consequently,

$$\int_{W} \lambda^{T} A(x) \lambda dx > 0$$

for any set  $W \subseteq [-\pi, \pi]$  with  $\sigma(W) > 0$  and any  $\lambda \in \Re^{2m+1} \setminus \{0\}$ . Therefore, from

$$\lambda^T \nabla F(\lambda^*) \lambda = \int_{K_+(\lambda^*)} \lambda^T A(x) \lambda dx,$$

we know that  $\nabla F(\lambda^*)$  is positive definite if and only if  $\sigma(K_+(\lambda^*)) > 0$ .

For convenience of expression, let  $\Lambda_0^* = \lambda_1^*$ ,  $\Lambda_k^* = \lambda_{2k}^* + j\lambda_{2k+1}^*$ ,  $k = 1, 2, \dots, m$ and  $\Lambda^* = (\Lambda_0^*, \Lambda_1^*, \dots, \Lambda_m^*)^T$ . Then  $\Lambda^* \neq 0$  and  $\bar{\Lambda}^{*T} M \Lambda^* > 0$ , where  $\bar{\Lambda}^*$  is the complex conjugate of  $\Lambda^*$ . From (7.3.8) we have

$$\bar{\Lambda}^{*T} M \Lambda^* = \sum_{l,h=0}^m \bar{\Lambda}_l^* \Lambda_h^* r_{l-h}$$
$$= \frac{1}{2\pi} \int_{-\pi}^{\pi} \left( B(x)^T \lambda^* \right)_+ \left( \bar{\Lambda}_0^* + \sum_{k=1}^m \bar{\Lambda}_k^* e^{-jkx} \right) \left( \Lambda_0^* + \sum_{k=1}^m \Lambda_k^* e^{jkx} \right) dx.$$

Let  $z(x) = \Lambda_0^* + \sum_{k=1}^m \overline{\Lambda}_k^* e^{-jkx}$ , and  $\overline{z(x)}$  be its conjugate. Then

$$\bar{\Lambda}^{*T}M\Lambda^* = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left( B(x)^T \lambda^* \right)_+ z(x)\overline{z(x)} dx$$
  
$$= \frac{1}{2\pi} \int_{K_+(\lambda^*)}^{\pi} \left( B(x)^T \lambda^* \right)_+ z(x)\overline{z(x)} dx.$$
(7.3.9)

Because  $z(x)\overline{z(x)} \ge 0$  for any  $x \in \Re$ , the above integrand is nonnegative. Since  $\bar{\Lambda}^{*T}M\Lambda^* > 0$ , (7.3.9) implies that the measure of set  $K_+(\lambda^*)$  is nonzero, i.e.,  $\sigma(K_+(\lambda^*)) > 0$ . By this, we have that  $\nabla F(\lambda^*)$  is positive definite.

**Remark 7.3.1** From Theorem 7.3.1 and Proposition 3.1 in [131], we know that there is a neighborhood  $\mathcal{N}(\lambda^*)$  of  $\lambda^*$  such that for any  $\lambda \in \mathcal{N}(\lambda^*)$ ,  $\nabla F(\lambda)$  is a nonsingular matrix. Now we study the Hölder continuous property of function  $\nabla F(\lambda)$  in (7.3.1) for  $\lambda \in \Re^{2m+1} \setminus \{0\}$ . It is crucial for the convergence analysis of the Newton method for solving (7.1.8).

**Proposition 7.3.3** Suppose  $\hat{\lambda} \neq 0$ . Then  $\nabla F(\cdot)$  defined by (7.3.1) is at least  $\frac{1}{2m}$ -order Hölder continuous at  $\hat{\lambda}$ .

**Proof.** Since  $\{1, 2\cos x, 2\sin x, \dots, 2\cos mx, 2\sin mx\}$  are linearly independent and  $\hat{\lambda} \neq 0$ , the set  $K_0(\hat{\lambda})$  consists of a finite number of points, say  $\{\hat{x}_1, \dots, \hat{x}_s\}$ .

For  $k = 1, 2, \dots, m$ , denote  $\beta_k(x)^T = 2(\cos kx, \sin kx)$  and

$$A_k(x) = 2 \begin{pmatrix} -k\sin kx & k\cos kx \\ -k^2\cos kx & -k^2\sin kx \end{pmatrix}.$$

Then

$$G_B(x) := (B(x), B^{(1)}(x), \cdots, B^{(2m)}(x))^T$$

$$= \begin{pmatrix} 1 & \beta_1(x)^T & \beta_2(x)^T & \cdots & \beta_m(x)^T \\ 0 & A_1(x) & A_2(x) & \cdots & A_m(x) \\ 0 & -A_1(x) & -2^2 A_2(x) & \cdots & -m^2 A_m(x) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & (-1)^{m-1} A_1(x) & (-2^2)^{m-1} A_2(x) & \cdots & (-m^2)^{m-1} A_m(x) \end{pmatrix}.$$
 (7.3.10)

Let

$$Q(x) = \begin{pmatrix} A_1(x) & A_2(x) & \cdots & A_m(x) \\ -A_1(x) & -2^2 A_2(x) & \cdots & -m^2 A_m(x) \\ \vdots & \vdots & \ddots & \vdots \\ (-1)^{m-1} A_1(x) & (-2^2)^{m-1} A_2(x) & \cdots & (-m^2)^{m-1} A_m(x) \end{pmatrix}.$$

It is not difficult to decompose Q(x) as follows

$$Q(x) = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ -1 & -2^2 & \cdots & -m^2 \\ \vdots & \vdots & \ddots & \vdots \\ (-1)^{m-1} & (-2^2)^{m-1} & \cdots & (-m^2)^{m-1} \end{pmatrix} \cdot \begin{pmatrix} A_1(x) & 0 & \cdots & 0 \\ 0 & A_2(x) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_m(x) \end{pmatrix}.$$
Hence,

$$\det(Q(x)) = \prod_{l,h=1,l< h}^{m} (l^2 - h^2) \cdot \prod_{k=1}^{m} \det(A_k(x))$$

Since  $\det(A_k(x)) = 4k^3 \neq 0$  for any positive integer k and  $x \in [-\pi, \pi]$ , we have that  $\det(Q(x)) \neq 0$ , which implies that  $\det(G_B(x)) \neq 0$ . Therefore, for any  $\hat{x}_l, \ l = 1, 2, \dots, s$ , there exists  $k_l \leq 2m$  such that

$$\sum_{\substack{i=1\\2m+1\\i=1}}^{2m+1} \hat{\lambda}_i B_i^{(j)}(\hat{x}_l) = 0 \text{ for } j = 1, 2, \cdots, k_l - 1,$$

$$\vdots$$

$$\sum_{i=1}^{2m+1} \hat{\lambda}_i B_i^{(k_l)}(\hat{x}_l) \neq 0.$$
(7.3.11)

That is,

$$\nabla_x^{(i)} g(\hat{\lambda}, \hat{x}_l) = 0, \quad i = 0, 1, \cdots, k_l - 1,$$
  
 
$$\nabla_x^{(k_l)} g(\hat{\lambda}, \hat{x}_l) \neq 0.$$

Let  $\hat{K}(\hat{\lambda}, h)$  be the set of such  $x \in K$  that  $g(\hat{\lambda}, x)$  and  $g(\hat{\lambda} + h, x)$  have different signs. From the proof of Theorem 2.3.2, we have that  $\sigma(\hat{K}(\hat{\lambda}, h)) = O(||h||^{\frac{1}{2m}})$ , where  $\sigma(\Omega)$  is the measure of set  $\Omega$ . Noticing that  $K_{-}(\hat{\lambda}) \cap K_{+}(\hat{\lambda} + h) \subseteq \hat{K}(\hat{\lambda}, h)$  and  $K_{+}(\hat{\lambda}) \cap K_{-}(\hat{\lambda} + h) \subseteq \hat{K}(\hat{\lambda}, h)$ , we obtain that, for all  $h \in \Re^{2m+1}$  small enough,

$$\begin{aligned} \left\| \nabla F(\hat{\lambda}+h) - \nabla F(\hat{\lambda}) \right\| &= \left\| \int_{K_{+}(\hat{\lambda}+h)} A(x) dx - \int_{K_{+}(\hat{\lambda})} A(x) dx \right\| \\ &= \left\| \int_{K_{+}(\hat{\lambda}+h) \cap K_{-}(\hat{\lambda})} A(x) dx - \int_{K_{-}(\hat{\lambda}+h) \cap K_{+}(\hat{\lambda})} A(x) dx \right\| \\ &\leq \max_{x \in K} \|A(x)\| \cdot \sigma(\hat{K}(\hat{\lambda},h)). \end{aligned}$$

$$(7.3.12)$$

Therefore, there exists a constant L > 0 such that

$$\left\|\nabla F(\hat{\lambda}+h) - \nabla F(\hat{\lambda})\right\| \le L\left(\|h\|^{\frac{1}{2m}}\right),$$

which means that  $\nabla F(\cdot)$  in (7.3.1) is Hölder continuous at  $\hat{\lambda}$ . The proof is complete.

The following example shows that function  $F(\cdot)$  defined by (7.2.8) may not be strongly semismooth.

Example 7.3.1 Take  $\hat{\lambda} = (1, -\frac{1}{2}, 0, \dots, 0)^T$ ,  $h = (0, -\frac{\delta}{2}, 0, \dots, 0)^T$  with  $\delta > 0$ . Then  $F_1(\hat{\lambda}) = \int_{-\pi}^{\pi} (1 - \cos x)_+ dx = \int_{-\pi}^{\pi} (1 - \cos x) dx = 2\pi$  and

$$F_1(\hat{\lambda} + h) = \int_{-\pi}^{\pi} (1 - (1 + \delta) \cos x)_+ dx$$
  
=  $2 \int_{x_{\delta}}^{\pi} (1 - (1 + \delta) \cos x) dx$   
=  $2(\pi - x_{\delta} + (1 + \delta) \sin x_{\delta}),$ 

where  $x_{\delta}$  is the point satisfying  $1 - (1 + \delta) \cos x_{\delta} = 0$  in  $[0, \pi]$ . Furthermore, we have

$$\nabla^T F_1(\hat{\lambda} + h) = \int_{-\pi}^{\pi} (1 - (1 + \delta) \cos x)^0_+ B(x)^T dx$$
$$= \left( \int_{-\pi}^{-x_{\delta}} B(x)^T dx + \int_{x_{\delta}}^{\pi} B(x)^T dx \right).$$

Therefore,

$$F_1(\hat{\lambda} + h) - F_1(\hat{\lambda}) - \nabla^T F_1(\hat{\lambda} + h)h$$
  
=  $2(\pi - x_{\delta} + (1 + \delta) \sin x_{\delta}) - 2\pi + 2\delta \int_{x_{\delta}}^{\pi} \cos x dx$   
=  $2(\sin x_{\delta} - x_{\delta}) = O(x_{\delta}^3).$ 

On the other hand, from  $1 - \cos x_{\delta} = \frac{\delta}{1+\delta}$  and  $1 - \cos x_{\delta} \sim \frac{1}{2}x_{\delta}^2$ , we have

$$x_{\delta} = O\left(\delta^{\frac{1}{2}}\right).$$

Consequently,

$$F_1(\hat{\lambda} + h) - F_1(\hat{\lambda}) - \nabla^T F_1(\hat{\lambda} + h)h = O\left(\delta^{\frac{3}{2}}\right) = O\left(\|h\|^{\frac{3}{2}}\right),$$

which implies that  $F_1(\cdot)$  is not strongly semismooth and hence  $F(\cdot)$  is not strongly semismooth.

Now we investigate the strongly semismooth property of F at the origin  $\hat{\lambda} = 0$ .

**Proposition 7.3.4** *F* is not differentiable and not piecewise smooth at the origin  $\hat{\lambda} = 0$ , but it is strongly semismooth at  $\hat{\lambda} = 0$ . Moreover, for any  $\lambda \in \Re^{2m+1} \setminus \{0\}$ , if  $\sigma(K_+(\lambda)) > 0$ , then

$$V = \int_{K_+(\lambda)} A(x) dx \in \partial_B F(0).$$

In special, 0,  $2\pi D \in \partial_B F(0)$ , where  $D = \begin{pmatrix} 1 & 0 \\ 0 & 2I \end{pmatrix}$  and I is the identity matrix in  $\Re^{2m \times 2m}$ .

**Proof.** Suppose that  $\lambda \in \Re^{2m+1} \setminus \{0\}$  and  $\sigma(K_+(\lambda)) > 0$ . Then  $\tilde{\lambda} = \frac{1}{n}\lambda \neq 0$  and  $\tilde{\lambda} \to 0$  as  $n \to \infty$ . Since  $\lambda^T B(x) > 0$  if and only if  $(\frac{\lambda}{n})^T B(x) > 0$ , we have  $\sigma(K_+(\frac{\lambda}{n})) = \sigma(K_+(\lambda))$ . By Proposition 7.3.1, F is differentiable at  $\tilde{\lambda}$  and

$$\nabla^T F(\frac{\lambda}{n}) = \int_{K_+(\lambda)} A(x) dx.$$

Therefore, from the definition of Clarke generalized Jacobian, we obtain that

$$V = \lim_{n \to \infty} \nabla^T F(\frac{\lambda}{n}) = \int_{K_+(\lambda)} A(x) dx \in \partial_B F(0).$$

From the arbitrariness of  $\lambda$ ,  $\partial_B F(0)$  contains infinite many elements. In Pang and Ralph [114] Lemma 2, it is said that if a function F is piecewise smooth then  $\partial_B F(\lambda)$ contains finitely many elements. Therefore, F is not differentiable and not piecewise smooth at the origin  $\hat{\lambda} = 0$ .

Especially, by taking  $\tilde{\lambda} = (\frac{1}{n}, 0, 0, \dots, 0)^T$ , we have  $\tilde{\lambda} \neq 0$ ,  $\tilde{\lambda} \to 0$ . Moreover,  $B(x)^T \tilde{\lambda} > 0$  for any  $x \in [-\pi, \pi]$ , which implies  $K_+(\tilde{\lambda}) = [-\pi, \pi]$ . Therefore, by Proposition 7.3.1 (iii),

$$\nabla^T F(\tilde{\lambda}) = \int_{-\pi}^{\pi} \left( B(x)^T \lambda^n \right)_+^0 A(x) dx$$
$$= \int_{-\pi}^{\pi} A(x) dx$$
$$= 2\pi D.$$

From the definition of  $\partial F(\cdot)$ , we know that  $2\pi D \in \partial_B F(0)$ . Similarly, by taking  $\tilde{\lambda} = (-\frac{1}{n}, 0, 0, \cdots, 0)^T$ , we have  $0 \in \partial_B F(0)$ .

Finally, we prove the strongly semismoothness of F at the origin  $\hat{\lambda} = 0$ . For any  $i = 1, 2, \dots, 2m + 1$ , and any  $h \in \Re^{2m+1}, h \neq 0$ , by Proposition 7.3.1,  $F_i(\cdot)$  is differentiable at  $\lambda = 0 + h$  and

$$\nabla^T F_i(0+h) = \int_{-\pi}^{\pi} \left( h^T B(x) \right)_{+}^0 B(x)^T B_i(x) dx.$$

Therefore,

$$F_{i}(0+h) - F_{i}(0) - \nabla^{T}F_{i}(0+h)h$$
  
=  $\int_{-\pi}^{\pi} (h^{T}B(x))_{+} B_{i}(x)dx - h^{T}\int_{-\pi}^{\pi} (h^{T}B(x))_{+}^{0} B(x)B_{i}(x)dx$   
=  $\int_{-\pi}^{\pi} (h^{T}B(x))_{+} B_{i}(x)dx - \int_{-\pi}^{\pi} (h^{T}B(x))_{+}^{0} (h^{T}B(x)) B_{i}(x)dx$   
=  $0$ 

 $= O(||h||^2).$ 

By Lemma 1.1.2,  $F_i(\cdot)$  is strongly semismooth at  $\hat{\lambda} = 0$  for any  $i = 1, 2, \dots, 2m + 1$ , which means that function F is strongly semismooth at  $\hat{\lambda} = 0$ .

The standard Newton iteration for solving (7.2.7) is

$$\lambda^{l+1} = \lambda^l - (\nabla^T F(\lambda^l))^{-1} (F(\lambda^l) - d), \quad l = 0, 1, 2, \cdots.$$
 (7.3.13)

From the homogeneity of function F defined by (7.2.8), we have  $\nabla^T F(\lambda^l) \lambda^l = F(\lambda^l)$ . Then the Newton iteration (7.3.13) reduces to the following simple form

$$\nabla^T F(\lambda^l) \lambda^{l+1} = d, \quad l = 0, 1, 2, \cdots$$
 (7.3.14)

Hence, Newton's method is easy to implement for  $L_2$  spectral estimation problem. Further, we have the convergence rate of Newton's method for the problem.

**Theorem 7.3.2** Suppose that  $\lambda^*$  is a solution of (7.2.7). Then the iterative sequence generated by Newton's iteration (7.3.14) converges to  $\lambda^*$  if the initial point  $\lambda^0$  is close to  $\lambda^*$ . The rate of the convergence is at least of order  $1 + \frac{1}{2m}$ .

**Proof.** From Propositions 7.3.1 and 7.3.3, for any  $\lambda \in \Re^{2m+1} \setminus \{0\}$ , function F in (7.2.7) is smooth, and its Jacobian  $\nabla F$  is at least  $\frac{1}{2m}$ -order Hölder continuous. By Theorem 7.3.1, 0 is not the solution of equations (7.2.7), and  $\nabla F$  is nonsingular at the solution of (7.2.7). Therefore, from the result on page 312 of [108], we have what we expected.

At the end of this section, we should mention that after we obtained the optimal dual solution  $\lambda^*$ , by (7.2.2) we can have the solution to the primal problem (7.1.8).

#### 7.4 Globalized Newton-Type Method and Its Convergence

In this section, we introduce a damped version of the Newton method for solving the dual problem (7.2.3). To this end, we let

$$L(\lambda) = \frac{1}{2} \int_{-\pi}^{\pi} \left( \sum_{i=1}^{2m+1} \lambda_i B_i(x) \right)_{+}^2 dx - \sum_{i=1}^{2m+1} \lambda_i d_i.$$
(7.4.1)

Then the maximization problem (7.2.3) is equivalent to the following minimization problem

$$\min_{\lambda \in \Re^{2m+1}} L(\lambda) = \frac{1}{2} \int_{-\pi}^{\pi} \left( \sum_{i=1}^{2m+1} \lambda_i B_i(x) \right)_{+}^2 dx - \sum_{i=1}^{2m+1} \lambda_i d_i.$$
(7.4.2)

This is a finite dimension convex problem. Note that  $L(\lambda)$  is smooth,  $\partial L(\lambda) = F(\lambda) - d$ and F is semismooth for any  $\lambda \in \Re^{2m+1}$ . Hence problem (7.4.2) is an  $SC^1$  convex programming problem.

Dontchev and Kalchev proved that  $L(\lambda)$  is coercive when B(x) is the basis of cubic *B*-spline (see [32] Lemma 2.1), i.e.,  $L(\lambda) \to +\infty$  as  $||\lambda|| \to \infty$ . It can be seen from the proof of Lemma 2.1 in [32] that the result can be extended to the case that B(x) is the trigonometric basis in this chapter. Thus we have

**Lemma 7.4.1** Function  $L(\lambda)$  defined by (7.4.1) is coercive. The problem (7.4.2) is well-posed in the sense of Tykhonov.

Lemma 7.4.1 implies that any algorithm that produces a minimizing sequence of (7.4.2) converges to its unique solution. The following damped Newton algorithm is a globalized Newton-type methods, which is globally convergent and also keep the fast local convergence of Newton's method.

Algorithm 7.4.1 (Damped Newton-Type Method)

Step 0 (Initialization) Choose  $\lambda^0 \in \Re^{2m+1}, \lambda^0 \neq 0, \rho \in (0,1), \theta \in (0,1/2)$ , and tolerance  $\epsilon > 0$ . Set k := 0.

**Step 1** (Termination criterion) If  $\epsilon_k = ||F(\lambda^k) - d|| \le \epsilon$ , then stop.

Step 2 (Direction generation) Take

$$V(\lambda^k) \in \partial F(\lambda^k). \tag{7.4.3}$$

Let  $t_k$  be a solution of the following linear system

$$(V(\lambda^k) + \epsilon_k I)t = d - F(\lambda^k).$$
(7.4.4)

**Step 3** (Line search) Choose  $m_k$  as the smallest nonnegative integer m satisfying

$$L(\lambda^k + \rho^m t_k) - L(\lambda^k) \le \theta \rho^m \nabla L(\lambda^k)^T t_k.$$
(7.4.5)

**Step 4** (Update) Set  $\lambda^{k+1} = \lambda^k + \rho^{m_k} t_k$ , k := k + 1, return to Step 1.

**Remark 7.4.1** The matrix  $V(\lambda^k)$  in (7.4.3) is always positive semidefinite for any  $\lambda^k \in \Re^{2m+1}$ . If  $\lambda^k \neq 0$ , we may take the elements of  $V(\lambda^k)$  as

$$V_{ij}(\lambda^k) = \int_{-\pi}^{\pi} \left( B(x)^T \lambda^k \right)_{+}^0 B_i(x) B_j(x) dx; \quad i, j = 1, 2, \cdots, 2m + 1.$$

Hence, the matrix  $V(\lambda^k) + \epsilon_k I$  is always positive definite for  $\epsilon_k > 0$  and therefore, the linear system (7.4.4) is uniquely solvable and  $t_k \neq 0$  if  $F(\lambda^k) - d \neq 0$ .

**Remark 7.4.2** The line search (7.4.5) in Step 3 is well-defined in the sense that there always exists a nonnegative integer  $m_k$  satisfying (7.4.5). Suppose by contradictory that

$$L(\lambda^k + \rho^m t_k) - L(\lambda^k) > \theta \rho^m \nabla L(\lambda^k)^T t_k$$

holds for any nonnegative integer m, then we have

$$\nabla L(\lambda^k)^T t_k = \lim_{m \to \infty} \frac{L(\lambda^k + \rho^m t_k) - L(\lambda^k)}{\rho^m} \ge \theta \nabla L(\lambda^k)^T t_k.$$

Hence, by  $\nabla L(\lambda^k) = F(\lambda^k) - d$ ,  $t_k$  is the solution of (7.4.4) and the positive semidefiniteness of  $V(\lambda^k)$ , we have

$$0 \le (1-\theta)\nabla L(\lambda^k)^T t_k = -(1-\theta)t_k(V(\lambda^k) + \epsilon_k I)t_k \le -(1-\theta)\epsilon_k ||t_k||^2 < 0.$$

This is a contradiction. Therefore, if  $\epsilon = 0$ , then for any initial point  $\lambda^0 \in \Re^{2m+1}$ , Algorithm 7.4.1 generates an infinite sequence  $\{\lambda^k\}$ . **Remark 7.4.3** Let  $\lambda^*$  be the solution of dual problem (7.4.2). Then for any  $\delta \in (0, 1)$ there exists a neighborhood  $U(\lambda^*)$  of  $\lambda^*$  and a scalar  $\bar{\epsilon} > 0$  such that for any  $\lambda \in U(\lambda^*)$ and  $\hat{\epsilon} \in (0, \bar{\epsilon}), V(\lambda) = \nabla F(\lambda)$  is nonsingular. Therefore, the linear system

$$(V(\lambda) + \hat{\epsilon}I)t_{\lambda} = d - F(\lambda)$$

has unique solution  $t_{\lambda}$ . Moreover, since  $\nabla F(\lambda)$  is at least  $\frac{1}{2m}$ -order Hölder continuous, so is  $\nabla^2 L(\lambda) = \nabla F(\lambda)$ . Hence, we have

$$L(\lambda + t_{\lambda}) - L(\lambda) - \frac{1}{2}\nabla L(\lambda)^{T} t_{\lambda} \leq \delta ||t_{\lambda}||^{1 + \frac{1}{2m}}$$

and

$$\lambda + t_{\lambda} \in U(\lambda^*).$$

From the analysis above, and by using a common argumentation technique, we can establish the convergence result of Algorithm 7.4.1 as follows, we omit the proof here.

**Theorem 7.4.1** Let  $\epsilon = 0$ ,  $\lambda^0 \in \Re^{2m+1}$  be any initial point,  $\{\lambda^k\}$  be generated by Algorithm 7.4.1. Then the whole sequence  $\{\lambda^k\}$  converges to the solution of (7.4.2), and the order of the convergence is at least  $1 + \frac{1}{2m}$ .

#### 7.5 Numerical Results

In this section, we report our numerical test results on applying Algorithm 7.4.1 to compute the spectral estimation on a variety of functions. For all testing problems in this section, we take  $K = [-\pi, \pi]$  and generate data  $r_k$  by taking a function s(x) and computing

$$r_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} s(x) e^{jkx} dx$$
 for  $k = 0, 1, \cdots, m$ .

Problem 7.5.1-7.5.3 are generated from the examples in [24] but with the difference that the functions in [24] are defined on [0, 1] whereas the functions in this chapter are defined on  $[-\pi, \pi]$ . Problem 7.5.4-7.5.5 are first presented in this chapter.

We implemented Algorithm 7.4.1 in Matlab 6.5 on a personal computer Pentium III 601 MHz with 256 MB of memory. In the tables below, Time is the processing time of

the algorithm in second, the computing error is defined as  $\text{Error} = ||F(\lambda) - d||$ , Nit is the number of the iterations.

Problem 7.5.1 Ideal Low-Pass Filter [24].

$$s(x) = \begin{cases} 1, & -\pi \le x \le 0, \\ 0, & 0 < x \le \pi. \end{cases}$$
(7.5.1)

For this problem, we compare our results to that of Cole and Goodrich in [24]. Cole and Goodrich considered the Ideal Low-Pass Filter problem on K = [0, 1]. Correspondingly, the basis functions in [24] are defined as

$$\phi_i(x) = \begin{cases} \sqrt{2}\cos(i-1)\pi x, & \text{if } i \text{ is odd,} \\ \sqrt{2}\sin(i\pi x), & \text{if } i \text{ is even.} \end{cases}$$

And the data is generated by

$$r_i = \int_K s(x)\phi_i(x)dx, i = 1, 2, \cdots, n$$

The error defined in [24] is also slightly different with ours, it is

$$\operatorname{Error}_{CG} = \left(\frac{1}{n}\sum_{i=1}^{n} \left(r_i - \int_K f(x)\phi_i(x)dx\right)^2\right)^{1/2},$$

where f(x) is the estimate of s(x). Comparing the basis  $\{\phi_i(x)\}$  in [24] and the basis  $\{B_i(x)\}$  (see (7.1.6) in Section 7.1) in our thesis, it is easy to see that n = 2m + 1, and  $\operatorname{Error}_{CG} \leq c$  Error, where 0 < c < 1 is a constant.

In [24], Cole and Goodrich used three algorithms for solving the spectral estimation problems. These three algorithms are the nonlinear least-squares algorithm for the primal problem, Newton's method for the dual problem (denoted as (DA)), and the hybrid Powell method (which is a Newton-type method) for the nonlinear system of equations generated from the dual problem (denoted as (NLSA)). They implemented their algorithms in Fortran and run on VAX 8530. They claimed that the nonlinear least-squares algorithm is both the slowest and least proficient at matching the data vector r, and (NLSA) does the best job of fitting the solution to the data.

In Table 7.1, we list the numerical results generated by the Damped Generalized Newton method (DGNM), i.e., Algorithm 7.4.1 in our paper, and the results of Cole

and Goodrich, generated by (DA) and (NLSA) [24]. We implemented (DGNM) for  $m = 1, 2, \dots, 12$ . Cole and Goodrich implemented their algorithms for  $m = 1, 2, \dots, 5$ . We can see from Table 7.1 that (DGNM) and (NLSA) have much better performance than (DA). For  $m = 1, 2, \dots, 5$ , (DGNM) and (NLSA) have similar computing precision except (NLSA) is not convergent for m = 4 (see [24] for the details).

For the sake of simplicity, in Fig. 7.1 we only illustrate the estimates to s(x) with m = 5 and m = 11. The solid line is the original function s(x) in (7.5.1), the dotted curve is the estimate to s(x) with m = 11, and the dashed curve is the estimate to s(x) with m = 5.

		DGNM			DA			NLSA
m	Time	Error	Nit	Time	$\operatorname{Error}_{CG}$	Nit	Time	$\operatorname{Error}_{CG}$
1	0.772	$1.1876 \times 10^{-13}$	9	17	$1.63\times 10^{-8}$	3	20	$1.09\times10^{-13}$
2	1.322	$5.7812 \times 10^{-13}$	14	86	$1.74 \times 10^{-8}$	6	128	$5.24\times10^{-15}$
3	1.531	$7.4541 \times 10^{-15}$	25	221	$2.26\times 10^{-7}$	7	430	$2.45\times10^{-12}$
4	6.319	$4.2373 \times 10^{-15}$	59	652	$7.97\times10^{-8}$	11	271	$9.91 \times 10^{-3}$
5	9.219	$6.2362 \times 10^{-14}$	185	19557	$1.50 \times 10^{-7}$	200	4072	$4.95 \times 10^{-15}$
6	6.760	$9.9672 \times 10^{-5}$	42					
7	5.468	$9.8243\times10^{-5}$	73					
8	9.704	$9.4071\times10^{-5}$	39					
9	14.931	$8.8296 \times 10^{-5}$	65					
10	3.266	$6.7856 \times 10^{-4}$	25					
11	19.458	$9.0704 \times 10^{-5}$	64					
12	3.406	$7.0174 \times 10^{-4}$	21					

Table 7.1: Iteration results for Problem 7.5.1

We can also see from Table 7.1 that, when m is small, the compute precision can be very high, but when m gets large, the compute precision is decreased. For example, when we take m = 5, the computing precision is  $Error = 6.2362 \times 10^{-15}$ , which is much higher than  $Error = 9.0704 \times 10^{-5}$  in the case of m = 11. However, from Fig. 7.1 we can see that, comparing to the estimation curve of s(x) with m = 5 (the dashed curve in Fig. 7.1), the estimation curve of s(x) with m = 11 (the dotted curve in Fig. 7.1) is much closer to that of the original filter s(x) (the solid line in Fig. 7.1).



Figure 7.1: The estimations with m = 5 and m = 11.

Therefore, we may say that, the bigger value is taken for the order of trigonometric bases m, the better estimation may be obtained to the original function s(x), although the computing precision Error = ||F(x) - d|| does not increase, or even decrease. The estimation results for other functions in this section also support this observation.

Problem 7.5.2 Nonideal Low-Pass Filter.

$$s(x) = \begin{cases} 0.75, & -\pi \le x \le -\pi/6, \\ 0.75 \exp(-\tan^2(1.5x + \pi/4)), & -\pi/6 < x \le \pi/6, \\ 0, & \pi/6 < x \le \pi. \end{cases}$$
(7.5.2)

This problem is similar to Example 7.2. in [24]. Cole and Goodrich implemented their algorithms on Example 7.2. [24] for  $m = 1, \dots, 5$ , and illustrated their estimation results graphically for m = 3, (i.e., n = 7) in their paper, see page 350 in [24].

We applied Algorithm 7.4.1 to Problem 7.5.2 for  $m = 4, 5, \dots, 12$  respectively. The estimation results are listed in Table 6.2. In Fig. 7.2 we illustrate the estimation results for s(x) in (7.5.2) with m = 5 and m = 12. The solid line in Fig. 7.2 is the original filter s(x) (so are the solid lines in the figures for other problems in this section).

m	Time	Error	Nit	
4	1.598	$5.0859 \times 10^{-15}$	41	
5	4.406	$2.4268 \times 10^{-14}$	85	
6	9.844	$9.3241 \times 10^{-6}$	161	
7	14.906	$9.9700 \times 10^{-6}$	204	
8	4.859	$9.5933 \times 10^{-5}$	55	
9	8.000	$9.0282 \times 10^{-5}$	79	
10	5.828	$9.7060 \times 10^{-5}$	50	
11	8.609	$9.8194 \times 10^{-5}$	64	
12	9.532	$8.0141 \times 10^{-5}$	62	

Table 7.2: Iteration results for Problem 7.5.2



Figure 7.2: The estimations with m = 5 and m = 12.

From Fig. 7.1 and Fig. 7.2 we can see that the estimation to Problem 7.5.2 is better than that to Problem 7.5.1. The reason is that s(x) in Problem 7.5.2 is continuous in  $(-\pi, \pi)$ , but the function s(x) in Problem 7.5.1 has a jump at x = 0. It is always difficult to estimate a jump function by a continuous function, especially at the jump point.

Problem 7.5.3 Two Steps with Disjoint Support Filter.

$$s(x) = \begin{cases} 0.5, & -\pi \le x \le -\pi/2, \\ 0, & -\pi/2 < x \le \pi/2, \\ 1, & \pi/2 < x \le \pi. \end{cases}$$
(7.5.3)

A similar example was also done by Cole and Goodrich [24] for  $m = 2, \dots, 5$ , and the estimation result for m = 3 was illustrated graphically, see Example 7.4 in [24]. The iteration results of Algorithm 7.4.1 for Problem 7.5.3 are listed in Table 7.3 and the estimation results are illustrated in Fig. 7.3 for m = 5 and m = 12.

m	Time	Error	Nit
3	2.924	$2.9472 \times 10^{-14}$	31
4	6.489	$5.1456 \times 10^{-15}$	61
5	2.693	$9.6428 \times 10^{-5}$	17
6	10.144	$9.8538 \times 10^{-5}$	71
7	10.125	$9.8939 \times 10^{-5}$	59
8	8.031	$9.8696 \times 10^{-5}$	31
9	9.624	$9.0769 \times 10^{-5}$	43
10	17.094	$8.7689 \times 10^{-5}$	58
11	12.638	$9.6168 \times 10^{-5}$	38
12	16.694	$9.0477 \times 10^{-5}$	54

Table 7.3: Iteration results for Problem 7.5.3



Figure 7.3: The estimations with m = 5 and m = 12.

We also applied our algorithm to estimate the following two functions. We take  $m = 4, 5, \dots, 12$  in the tests. The iteration results are listed in Table 7.4 and Table 7.5, and the estimation results for the filters with m = 6 and m = 12 are illustrated in Fig.

7.4 and Fig. 7.5, respectively. For other values of m, the resulting estimates also match the corresponding original filter quite well.

Problem 7.5.4 Three Steps with Disjoint Support Filter.

$$s(x) = \begin{cases} 2/3, & -\pi \le x \le -\pi/2, \\ 0, & -\pi/2 < x \le 0, \\ 1/3, & 0 < x \le \pi/2, \\ 1, & \pi/2 < x \le \pi. \end{cases}$$
(7.5.4)

Problem 7.5.4 is more complicated than Problem 7.5.3.

m	Time	Error	Nit
4	2.464	$5.3682 \times 10^{-15}$	21
5	2.063	$5.0245 \times 10^{-15}$	13
6	4.306	$4.7050 \times 10^{-15}$	26
7	5.468	$5.9164 \times 10^{-15}$	29
8	3.475	$6.8438 \times 10^{-15}$	15
9	4.276	$2.6534 \times 10^{-15}$	15
10	6.019	$9.9292 \times 10^{-15}$	19
11	4.767	$5.0729 \times 10^{-15}$	16
12	15.022	$2.9448 \times 10^{-15}$	27

Table 7.4: Iteration results for Problem 7.5.4

From Table 7.4 we can see that the computing precision of Algorithm 7.4.1 for Problem 7.5.4 is very high. But from Fig. 7.4 we find that the estimation for the function s(x) in Problem 7.5.4 is not as good as the estimation for the functions in the previous problems. That is because the function (7.5.4) has too many jumps on  $[-\pi, \pi]$ , which lead to big difficulty for the continuous estimation for the function. Moreover, we can also see from Fig. 7.4 that, with the increase of the order of the trigonometric bases, say, from m = 6 to m = 12, the estimation curve is getting closer to that of the original function s(x) (comparing the dotted curve with m = 12 to the dashed curve with m = 6). Therefore, in order to get a good estimation for a jump function, we usually need to choose a big m, which means the dimension of the problem will be high.



Figure 7.4: The estimations with m = 6 and m = 12.

Problem 7.5.5 Parallel Filter.

$$s(x) = \begin{cases} x/\pi + 1, & -\pi \le x \le 0, \\ x/\pi, & 0 < x \le \pi. \end{cases}$$
(7.5.5)

m	Time	Error	Nit
4	1.222	$2.8165 \times 10^{-13}$	9
5	1.392	$1.7227 \times 10^{-11}$	9
6	1.763	$8.6088 \times 10^{-15}$	10
7	1.952	$7.1300 \times 10^{-14}$	10
8	2.443	$6.3625 \times 10^{-15}$	11
9	2.824	$7.1319 \times 10^{-15}$	11
10	3.024	$2.4748 \times 10^{-11}$	11
11	3.175	$2.5827 \times 10^{-11}$	12
12	4.787	$4.6952 \times 10^{-15}$	14

Table 7.5: Iteration results for Problem 7.5.5

For Problem 7.5.5, we can see from Fig. 7.5 that, the estimation for s(x) on  $(-\pi, 0)$  and  $(0, \pi)$  is very precise. With the increase of value m, the estimation for s(x) on the whole  $[-\pi, \pi]$  is getting better. Moreover, from Table 7.4 and Table 7.5 we can also



Figure 7.5: The estimations with m = 6 and m = 12.

see that the iteration number and processing time of the algorithm do not change a lot with the increase of the order of the trigonometric bases m.

From the preliminary numerical test results in this section and the numerical observation by Cole and Goodrich [24] and Potter [119], we can say that applying Newtontype methods to the nonlinear equations generated from the dual of the spectral estimation problem is an efficient way to solve the problem. Moreover, from the implementation process of Algorithm 7.4.1 we also noticed that Algorithm 7.4.1 is not sensitive with the choice of the starting point  $\lambda^0 \in \Re^{2m+1}$ . Hence, Algorithm 7.4.1 is efficient and stable.

### Chapter 8

# Conclusions and Suggestions for Future Studies

In this thesis, we developed some generalized Newton methods for solving a class of SIP problems, a class of option price interpolation problems and a class of  $L_2$  spectral estimation problems. We also developed a method for solving a class of stochastic generalized SIP problems.

In Chapter 2, we introduced a general class of integral functions which includes the particular integral functions arising from many application problem. We investigated the differentiability, semismoothness and smoothing approximation properties of this class of integral functions. These properties play a very important role in the convergence analysis of the algorithms mentioned above.

Based on the investigation of the integral function, in Chapters 3 and 4, we presented four generalized Newton methods for solving SIP problems. In Section 3.2, we first presented a smoothing SQP algorithm. At each iteration of the algorithm, we only need to solve a quadratic program which is always feasible and solvable. The global convergence of the smoothing SQP algorithm was established under some mild conditions. However, this algorithm has two drawbacks: (1) it has no local superlinear convergence; (2) the accumulation point of a sequence generated by it may not be a stationary point of the original SIP problem and is only a generalized stationary point of an equivalent programming problem. To overcome the first drawback, we presented a smoothing projected Newton-type algorithm in Section 3.3. We proved that this algorithm has global and local superlinear convergence under some mild conditions. Furthermore, based on the smoothing projected Newton-type algorithm, we also constructed a truncated projected Newton-type algorithm in Chapter 4, which not only has global and local superlinear convergence property but also can solve the large scale SIP problems with 2000 decision variables. This algorithm is significant since many large-scale SIP problems arise in various real fields. Considering the second drawback mentioned in the smoothing SQP algorithm still exists, in Section 3.4, we presented a new method for solving SIP problems, say, smoothing Newton-type algorithm, which overcome the two drawbacks stated above. In addition, for the four algorithms above, the feasibility of the accumulation point of a sequence generated by it was ensured by an integral function. Numerical test examples show that each algorithm performs well.

In Chapter 5, we discussed a generalized semi-infinite programming problem with uncertainty. The expected value approach was applied to define a deterministic version of the problem. We proposed a new reformulation by using the first order optimality conditions of the second stage optimization problem. Then, we presented a smoothing implicit programming method to solve the problem with finite discrete distribution. Global convergence results was obtained under some mild conditions.

In Chapter 6, we showed that the generalized Newton method presented by Wang, Yin and Qi has at least  $\frac{4}{3}$ -order convergence rate. We gave conditions under which this method has  $\frac{3}{2}$ -order and quadratic convergence rate. We also gave a damped version of the generalized Newton method and showed that it is globally convergent and the convergence order is at least  $\frac{4}{3}$ .

In Chapter 7, a Newton method for solving power spectrum estimation problems was proposed, and it was proved that the method is at least  $1 + \frac{1}{2m}$ -order convergent. We also developed a globalized Newton-type method for solving the problem, this method has at least  $1 + \frac{1}{2m}$ -order convergence.

The following is a list of some interesting and challenging problems for future research.

1. The smoothing SQP algorithm proposed in Section 3.2 has global convergence and

performs well in numerical test. However, It has no very good local convergence property such as superlinear or quadratic convergence. In addition, some assumptions (for instance, (B4)) needed in the convergence analysis of the algorithm are somewhat restrictive in theoretical aspect. Thus, it is of great significance to develop a smoothing SQP algorithm for solving SIP problems such that this algorithm has good local convergence property under some mild conditions.

2. The smoothing projected Newton-type algorithm, the smoothing Newton-type algorithm and the truncated projected Newton-type algorithm proposed in Chapters 3 and 4, are computationally efficient. However, the same problem appeared in the three algorithms is that the parameter p, the numbers of the local maximizers of the nonlinear programming problem

$$\max_{v \in V} g(x, v)$$

need to be guessed previously. Thus, it is very interesting to develop an efficient algorithm for solving SIP problems, such that the parameter p can be obtained in the implementation process of the algorithm, but is not guessed previously. In addition, further experience with testing and with actual applications will be necessary.

- 3. In Chapter 5, we proposed a method for solving the stochastic generalized SIP problems with finite discrete distribution. Of course, it is interesting to construct an algorithm for solving the stochastic generalized SIP problems with continuous distribution.
- 4. Two research topics related to Chapter 7 are: (1) How to construct an efficient algorithm for solving the high dimension  $L_2$  spectral estimation problems? (2) How to construct an efficient algorithm for solving the  $L_2$  spectral estimation problems with  $L_{\infty}$  upper bound? These works are very interesting, since the problems mentioned arise from many applications.

We will continue work on these topics.

## Bibliography

- Andreasen, L., Implied modeling: stable implementation, hedging, and duality. Research report, University of Aarhus.
- [2] Andreasen, L. and Brotherton-Ratcliffe, R., The equity option volatility smile: An implicit finite difference approach. *Journal of Computational Finance*, Vol.1, pp.5-38 (1998).
- [3] Bachelier, L., Théorie de la spéculation. Annales de l'Ecole Superieure, Vol.17, pp.21-86 (1900).
- [4] Bates, D., The crash of 87, was it expected? The evidence from options markets. Journal of Finance, Vol.46, pp.1009-1044 (1996).
- [5] Bell, R.J., Introductory Fourier Thansform Spectroscopy, Academic Press, New York, 383 pp.(1972).
- [6] Ben-Tal, A., Borwein, J.M. and Teboulle, M., A dual approach to multidimension L<sub>p</sub> spectral estimation problems. SIAM Journal on Control and Optimization, Vol.26, pp.985-996 (1988).
- [7] Ben-Tal, A., Teboulle, M. and Zowe, J., Second order necessary optimality conditions for semi-infinite programming problems. In: Hettich, R. eds. Semi-Infinite Programming, Lecture Notes in Control and Information Sciences, Vol.15, Springer-Verlag, New York, pp.17-30 (1978).
- [8] Birge, J.R. and Louveaux, F., Introduction to Stochastic Programming, Springer, New York, 421 pp.(1997).
- [9] Birge, J.R. and Qi, L., Continuous approximation schemes for solving stochastic programs. Annals of Operations Research, Vol.56, pp.15-38 (1995).

- [10] Black, F. and Scholes, M., The pricing of options and corporate liabilities. *Journal of Political Economy*, Vol.81, pp.637-659 (1973).
- [11] Bonnans, J.F. and Shapiro, A., Perturbation Analysis of Optimization Problems, Springer, New York, 601 pp.(2000).
- [12] Borwein, J.M. and Lewis, A.S., Dual relationships for entropy-like minimization problems. SIAM Journal Control Optimization, Vol.29, pp.325-333 (1991).
- [13] Borwein, J.M. and Lewis, A.S., Partially finite convex programming I: Quasi relative interiors and duality theory; II: Explicit lattice models. *Mathematical Programming*, Vol.57, pp.15-83 (1992).
- [14] Burg, J.P., Maxmum entropy spectral analysis. Proceedings, The 37-th international meeting of the society of exploratory geophysics, Oklahoma City, OK (1968).
- Burg, J.P., Maxmum Entropy Spectral Analysis, Ph.D. dissertation, Stanford Univ., CA, (1975).
- [16] Calamai, P.H. and Moré, J.J., Projected gradient methods for linear constrained problems. *Mathematical Programming*, Vol.39, pp.93-116 (1987).
- [17] Chamberlain, J.E., The Principles of Interferometric Spectroscopy, Wiley, New York, 347 pp.(1979).
- [18] Chen, C. and Mangasarian, O.L., A class of smoothing functions for nonlinear and mixed complementarity problems. *Computational Optimization and Applications*, Vol.5, pp.97-138 (1996).
- [19] Chen, X., Nashed, Z. and Qi, L., Smoothing methods and semismooth methods for nondifferentiable operator equations. SIAM Journal on Numerical Analysis, Vol.38, pp.1200-1216 (2000).
- [20] Chen, X., Qi, L. and Sun, D., Global and superlinear convergence of the smoothing Newton method and its application to general box constrained variational inequalities. *Mathematics of Computation*, Vol.67, pp.519-540 (1998).
- [21] Chen, X. D., Sun, D. and Sun, J., Complementarity functions and numerical experiments on some smoothing Newton methods for second-order-cone complemen-

tarity problems. *Computational Optimization and Applications*, Vol.25, pp.39-56 (2003).

- [22] Chen, X. and Ye, Y., On homotopy-smoothing methods for box-constrained variational inequalities. SIAM Journal Control Optimization, Vol.37, pp.589-616 (1999).
- [23] Clarke, F.H., Optimization and Nonsmooth Analysis, John Wiley and Sons, New York, 308 pp.(1983).
- [24] Cole, R.E. and Goodrich, R.K.,  $L_p$ -spectral estimation with an  $L_{\infty}$ -upper bound. Journal of Optimization Theory and Applications, Vol.76, pp.321-355 (1993)
- [25] Colgen, R., Necessary conditions for upper semicontinuity in parametric semiinfinite programming. *Journal of Optimization Theory and Applications*, Vol.48, pp.65-79 (1986).
- [26] Coope, I.D. and Watson, G.A., A projected Lagrangian algorithm for semi-infinite programming. *Mathematical Programming*, Vol.32, pp.337-356 (1985).
- [27] Corwin, L.J. and Szczarba, R.H., Multivariable Calculus, Mar Dekker, Inc. New York, 524 pp.(1982).
- [28] Dennis, J.E., Nonlinear least squares and equations. In: Jacobs, D.A.H. eds. The State of the Art in Numerical Analysis, pp.269-312, Academic Press, New York (1975).
- [29] Dennis, J.E. and Mor, J.J., Quasi-Newton methods: Motivation and theory, SIAM Review, Vol.19, pp.46-89 (1977).
- [30] Dennis, J.E. and Schnabel, R.B., Numerical Methods for Unconstrained Optimization and Nonlinear Equation, Prentice Hall, Englewood Cliffs, New Jersey, 378 pp.(1983).
- [31] Derman, E., Kani, I. and Chriss, N., Implied trinomial tree of the volatility smile. Journal of Derivatives, Vol.3, pp.7-22 (1996).
- [32] Dontchev, A.L. and Kalchev, B.D., Duality and well-posedness in convex interpolation. Numerical Functional Analysis and Optimization, Vol.10, pp.673-689 (1989).

- [33] Dontchev, A. L., Qi, H.D. and Qi, L., Convergence of Newton's method for convex best interpolation. *Numerische Mathematik*, Vol.87, pp.435-456 (2001).
- [34] Dontchev, A. L., Qi, H.D. and Qi, L., Quadratic convergence of Newton's method for convex interpolation and smoothing. *Constructive Approximation*, Vol.19, pp.123-143 (2003).
- [35] Dontchev, A. L., Qi, H.D., Qi, L. and Yin, H., A Newton method for shapepreserving spline interpolation. SIAM Journal on Optimization, Vol.13, pp.588-602 (2003).
- [36] Duffie, D., Dynamic Asset Pricing Theory, Princeton University Press, Princeton, 299 pp.(1996).
- [37] Dumas, B., Flemming, J. and Whaley, R.E., Implied volatility functions: Empirical tests. *Journal of Finance*, Vol.52, pp.2056-2106 (1997).
- [38] Facchinei, F., Jiang, H. and Qi, L., A smoothing method for mathematical programs with equilibrium constraints. *Mathematical Programming*, Vol.85, pp.107-134 (1999).
- [39] Facchinei, F. and Kanzow, C., A nonsmooth inexact Newton method for the solution of large-scale nonlinear complementarity problems. *Mathematical Program*ming, Vol.76, pp.493-512 (1997).
- [40] Facchinei, F. and Pang, J.S., Finite-Dimensional Variational Inequalities and Complementarity Problems, I-II, Springer-verlag, New York, 1234 pp.(2003).
- [41] Fang, S.C. and Wu, S.Y., An entropic path-following approach for linear semiinfinite programming problems. In: *Mathematics Today*, XII-A, pp.1-16 (1994).
- [42] Fang, S.C. and Wu, S.Y., An inexact approach to solving linear semi-infinite programming problems. *Optimization*, Vol.28, pp.291-299 (1994).
- [43] Fiacco, A.V. and Ishizuka, Y., Suggested research topics in sensitivity and stability analysis for semi-infinite programming. *Annals of Operation Research*. Vol.27, pp.65-76 (1990).

- [44] Fiacco, A.V. and Kortanek, K. O. eds., Semi-Infinite Programming and Applications, Lecture Notes in Economics and Mathematical Systems, Vol.215, Springer-Verlag, Berlin (1983).
- [45] Fischer, A., A special Newton-type optimization method. Optimization, Vol.24, pp.269-284 (1992).
- [46] Fischer, A., Solution of monotone complementarity problems with locally Lipschitzian functions. *Mathematical Programming*, Vol.76, pp.513-532 (1997).
- [47] Fukushima, M. and Qi, L., A globally and superlinearly convergent algorithm for nonsmooth convex minimization. SIAM Journal on Optimization, Vol.6, pp.1106-1120 (1996).
- [48] Gabriel, S.A. and Moré, J.J., Smoothing of mixed complementarity problems. In: Ferris, M.C. and Pang, J.S. eds. *Complementarity and Variational Problems: State* of the Art, SIAM, Philadelphia, pp.105-116 (1997).
- [49] Girsanov, I., Lectures on the mathematical theory of extremum problems. Lecture Notes in Economics and Mathematical Systems, Vol.67, Springer, New York (1972).
- [50] Glashoff, K. and Roleff, K., A new method for Chebyshev approximation of complex-valued functions. *Mathematics of Computation*, Vol.36, pp.233-239 (1981)
- [51] Goberna, M.A. and López, M.A., *Linear Semi-Infinite Optimization*, John Wiley & Sons, New York, 343 pp.(1998).
- [52] Goberna, M.A. and López, M.A., Semi-Infinite Programming: Recent Advances, Kluwer Academic Publishers, 386 pp.(2001).
- [53] Goh, C.J. and Teo, K.L., Alternative algorithms for solving nonlinear function and functional inequalities. *Applied Mathematics and Computation*, Vol.41, pp.159-177 (1991).
- [54] Goodrich, R. K. and Steinhardt, A., L<sub>2</sub> spectral estimation. SIAM Journal on Applied Mathematics, Vol.46, pp.417-426 (1986).
- [55] Gramlich, G., Hettich, R. and Sachs, E.W., Local convergence of SQP methods in semi-infinite programming. SIAM Journal on Optimization, Vol.5, pp.641-658 (1995).

- [56] Graettinger, T.J. and Krogh, B.H., The acceleration radius: A global performance measure for robotic manipulators, *IEE Journal Robotics Automation*, Vol.4 pp.60-69 (1988).
- [57] Grenander, U. and Szero, G., *Toeplitz Form and Their Applications*, University of California Press, Berkeley, 245 pp.(1958).
- [58] Gribik, P.R., Selected applications of semi-infinite programming. In: Constructive Approaches to Mathematical Models, Academic Press, New York, pp.171-187 (1979).
- [59] Gürkan, G., özge, A.Y. and Robinson, S.M., Sample-path solution of stochastic variational inequalities, *Mathematical Programming*, Vol.84, pp.313-333 (1999).
- [60] Gustafson, S.A., On numerical analysis in semi-infinite programming. In: Hettich, R. eds. Semi-Infinite Programming, Necture Notes in Control and Information Sciences, Vol.15, Springer, New York, pp.51-65 (1979).
- [61] Gustafson, S.A., A three-phase algorithm for semi-infinite programming. In: Fiacco, A.V. and Kortanek, K.O. eds. *Semi-Infinite Programming and Applications*, Springer-Verlag, Berlin, pp.138-157 (1983).
- [62] Gustafson, S.A. and Kortanek, K.O., Mathematical programming: The state of the art. In: Fiacco, A.V. and Kortanek, K.O. eds. *Semi-Infinite Programming and Applications*, Springer-Verlag, Berlin, pp.132-137 (1983).
- [63] Harker, P.T. and Xiao, B., Newton method for the nonlinear complementarity problem: A B-differentiable equation approach. *Mathematical Programming*, Vol.48, pp.339-357 (1990).
- [64] Hassouni, A. and Oettli, W., On regularity and optimality in nonlinear semi-infinite programming. In: Goberna, M.Á. and López, M.A. eds. Semi-Infinite Programming: Recent Advances, Kluwer Academic Publishers, pp.59-74 (2001).
- [65] Hettich, R. and Jongen, H.TH., Semi-infinite programming: Conditions of optimality and applications. In: Optimization Techniques, Part 2, Lecture Notes in Control and Information Science, Vol.7, Springer, Berlin, pp.1-11 (1978).

- [66] Hettich, R. and Honstede, W.V., On quadratically convergent methods for semiinfinite programming. In: Hettich, R. eds. Semi-Infinite Programming, Necture Notes in Control and Information Sciences, Vol.15, Springer, New York, pp.97-111 (1979).
- [67] Hettich, R. and Zencke, P., Numerische Methoden der Approximation und Semi-Infinite Optimierung, B.G. Teubner, Stuttgart, 232 pp.(1982).
- [68] Hettich, R., A review of numerical methods for semi-infinite optimization. In: Fiacco, A.V. and Kortanek, K.O. eds. *Semi-Infinite Programming and Applications*, Springer-Verlag, Berlin, pp.158-178 (1983).
- [69] Hettich, R., An implementation of a discretization method for semi-infinite programming. *Mathematical Programming*, Vol.34, pp.354-361 (1986).
- [70] Hettich, R. and Kortanek, K.O., Semi-infinite programming: Theory, methods, and applications. SIAM Review, Vol.35, pp.380-429 (1993).
- [71] Hettich, R. and Still, G., Semi-infinite programming models in robotics. In: Guddat, J., Jongen, H.T., Kummer, B. and Nozicka, F. eds. *Parametric Optimization* and Related Topics II, Akademie Verjag, Berlin, pp.112-118 (1991).
- [72] Hettich, R. and Still, G., Second-order optimality conditions for generalized semiinfinite programming problems. *Optimization*, Vol.34, pp.195-211 (1995).
- [73] Huang, Z.H., Han, J. and Chen, Z., Predictor-corrector smoothing Newton method, based on a new smoothing function, for solving the nonlinear complementarity problem with P<sub>0</sub> function. Journal of Optimization Theory and Applications, Vol.117, pp.39-68 (2003).
- [74] Huber, P. J., Robust regression: Asympotics, conjectures, and Monte Carlo, Annals of Statistics, Vol.1, pp.799-821 (1973).
- [75] Hull, J. and White, A., The pricing of options with stochastic volatility. *Journal of Finance*, Vol.42, pp.281-300 (1987).
- [76] Hunt, P., Kennedy, J. and Pelasser, A., Markov-functional interest rate models. *Finance and Stochastics*, Vol.4, pp.391-408 (2000).

- [77] Irvine, L.D., Marin, S.P. and Smith, P.W., Constrained interpolation and smoothing. *Constructive Approximation*, Vol.2, pp.129-151 (1986).
- [78] Ito, S., Liu, Y. and Teo, K.L., A dual parameterization method for convex semiinfinite programming. Annals of Operations Research, Vol.98, pp.189-213 (2000).
- [79] Jackwerth, J.C. and Rubinstein, M., Recovering probability distributions from contemporary security prices. *Journal of Finance*, Vol.51, pp.1611-1631 (1996).
- [80] Janin, R., Direction derivative of the marginal function in nonlinear programming. Mathematical Programming Study, Vol.21, pp.127-138 (1984).
- [81] Jennings, L.S. and Teo, K.L., A computational algorithm for functional inequality constrained optimization problems. *Automatica*, Vol.26, pp.371-375 (1990).
- [82] Jiang, H. and Qi, L., Local uniqueness and Newton-type methods for nonsmooth variational inequalities. *Journal of Mathematical Analysis and Applica tions*, Vol.196, pp.314-333 (1995).
- [83] Jiang, H. and Ralph, D., Global and local superlinear convergence analysis of Newton-type methods for semismooth equations with smooth least squares. In: Fukushima, M. and Qi, L. eds. *Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods*, Kluwer Academic Publishers, Dordrecht, pp.181-209 (1998).
- [84] Jiang, H. and Ralph, D., Smooth SQP methods for mathematical programs with equilibrium constraints. *http://www.ms.unimelb.edu.au/~danny/smooth-mpec.ps.*
- [85] Jiang, H. and Ralph, D., Smooth SQP methods for mathematical programs with nonlinear complementarity constraints, SIAM Journal on Optimization, Vol.10, pp.779-808 (2000).
- [86] Jongen, H.TH. Rückmann, J.J. and Stein, O., Generalized semi-infinite optimization: A first order optimality condition and examples. *Mathematical Programming*, Vol.83, pp.145-158 (1998).
- [87] Kanzow, C., Some noninterior continuation methods for linear complementarity problems. SIAM Journal on Matrix Analysis and Applications, Vol.17, pp.851-868 (1996).

- [88] Kanzow, C. and Pieper, H., Jacobian smoothing methods for nonlinear complementarity problems, SIAM Journal on Optimization, Vol.9, pp.342-373 (1999).
- [89] Kaplan, A. and Tichatschke, R., Proximal interior point methods for convex semi-infinite programming. Optimization Methods and Software, Vol.15, pp.87-119 (2001).
- [90] Kay, S.M. and Marple, S. L., Spectrum analysis-a modern perspective, Proc. IEEE, Vol.69, pp.1380-1419 (1981).
- [91] Kortanek, K.O. and Moulin, P., Semi-infinite programming in orthogonal wavelet filter design. In: Reemtsen, R. and Rückmann, J. eds. *Semi-Infinite Programming*, Kluwer academic Publishers, Boston, pp.323-360 (1998).
- [92] Krabs, W., Optimization and Approximation, John Wiley and Sons, Chichester, England, 220 pp.(1979).
- [93] Krabs, W., On Time-minimal heating or cooling of a ball. In: Internet. Ser. Numer. Math. 81, Birkhäuser, Basel, pp.121-131 (1987).
- [94] Kummer, B., Newton's method based on generalized derivatives for nonsmooth functions: convergence analysis. In: Oettli, W. and Pallaschke, D. eds. Advances in Optimization, Springer-Verlag, Berlin, pp.171-194 (1992).
- [95] Lai, H.C. and Wu, S.Y., On linear semi-infinite programming problems: An algorithm. Numerical Functional Analysis and Optimization, Vol.13, pp.287-304 (1992).
- [96] Landau, H. J., Maximum entropy and maximum lilelihood in spectral estimation. *IEEE Transactions on Information Theory*, Vol.44, pp. 1332-1336 (1998).
- [97] Lang, S.W. and McClellan, J.H., Spectral estimation for sensor arrays, *IEEE Transactions on Acoustics Speech and Singnal Processing*, Vol.31, pp.349-358 (1983).
- [98] Li, D., Qi, L., Tam, J. and Wu, S.Y., Smoothing Newton methods for semi-infinite programming, *Journal of Global Optimization*, Vol.30, pp.169-194 (2004).
- [99] Lin, C.J., Fang, S.C. and Wu, S.Y., A dual affine scaling based algorithm for solving linear semi-infinite programming problems. In: Du, D.Z. and Sun, J. eds. Advances in Optimization and Approximation, Kluwer, London, pp.217-233 (1994).

- [100] Lin, D.M. and Wong, E.K., A survey on the maximum entropy method and parameter spectral estimation. *Physics Reports (Review Section of Physics Letters)*, Vol.193, pp.41-135 (1990).
- [101] De Luca, T., Facchinei, F. and Kanzow, C., A semismooth equation approach to the solution of nonlinnear complementarity problems. *Mathematical Programming*, Vol.75, pp.407-439 (1996).
- [102] Mangasarian, O., Nonlinear Programming, McGraw Hill, New York, 220 pp.(1969).
- [103] Marin, S.P., Optimal parameterization of curves for robot trajectory design. IEEE Transactions on Automatic Control, Vol. AC-33, pp.209-214 (1988).
- [104] Martinez, J.M. and Qi, L., Inexact Newton methods for solving nonsmooth equations. Journal of Computational and Applied Mathematics, Vol.60, pp.127-145 (1996).
- [105] Mifflin, R., Semismooth and semiconvex functions in constrained optimization. SIAM Journal on Control and Optimization, Vol.15, pp.957-972 (1977).
- [106] Moré, J.J., Global methods for nonlinear complementarity problems. Mathematics of Operations Research, Vol.21, pp.589-614 (1996).
- [107] Nuernberger, G., Global unicity in semi-infinite programming. Numerical Functional Analysis and Optimization, Vol.8, pp.173-191 (1985).
- [108] Ortega, J.M. and Rheinboldt, W.C., Iterative Solution of Nonlinear Equations in Several Variables, Academic Press, New York, 572 pp.(1970).
- [109] Outrata, J.V. and Zowe, J., A Newton method for a class of quasi-variational inequalities. *Computational Optimization and Applications*, Vol.4, pp.5-21 (1995).
- [110] Pang, J.S., Newton's method for B-differentiable equations, Mathematics of Operations Research, Vol.15, pp.311-341 (1990).
- [111] Pang, J.S. A B-differentiable equation based, globally, and locally quadratically convergent algorithm for nonlinear programs, complementarity and variational inequality problems. *Mathematical Programming*, Vol.51, pp.101-131 (1991).

- [112] Pang, J.S. and Qi, L., Nonsmooth equations: Motivation and algorithms, SIAM Journal on Optimization, Vol.3, pp.443-465 (1993).
- [113] Pang, J.S. and Qi, L., A globally convergent Newton method for convex SC<sup>1</sup> minimization problems. Journal of Optimization Theory and Applications, Vol.85, pp.633-648 (1995).
- [114] Pang, J.S. and Ralph, D., Piecewise smoothness, local invertiability, and parametric analysis of normal maps, *Mathematics of Operations Research*, Vol.21, pp.401-426 (1996).
- [115] Polak, E., On the mathematical foundations of nondifferentiable optimization in engineering design. SIAM Review, Vol.29, pp.21-89 (1987).
- [116] Polak, E., On the use of consistent approximations in the solution of semi-infinite optimization and optimal control problems, *Mathematical Programming*, Vol.62, pp.385-414 (1993).
- [117] Polak, E., Optimization: Algorithms and Consistent Approximation, Springer-Verlag, New York, 779 pp.(1997).
- [118] Palak, E. and Tits, A.L., A recursive quadratic programming algorithm for semiinfinite programming problems. *Applied Mathematics and Optimization*, Vol.8, pp.325-349 (1982).
- [119] Potter, L.C., Constrained Signal Reconstruction, PhD. Thesis, University of Illinois at Urbana-Champaign (1990).
- [120] Pshenichnyi, B.N., Necessary Conditions for an Extremum, Marcel Dekker, NY, 230 pp.(1971).
- [121] Qi, H. D. and Liao, L.Z., A smoothing Newton method for general nonlinear complementarity problems. *Computational Optimization and Applications*, Vol.17, pp.231-253 (2000).
- [122] Qi, L., Convergence analysis of some algorithms for solving nonsmooth equations. Mathematics of Operations and Research, Vol.18, pp.227-244 (1993).
- [123] Qi, L., Trust region algorithms for solving nonsmooth equations. SIAM Journal on Optimization, Vol.5, pp.219-230 (1995).

- [124] Qi, L., Semismoothness properties and applications of an integral function. In: Tamura, A. and Ito, H. eds. *Proceedings of The Fourteenth RAMP Symposium*, RAMP, Kyoto, pp.103-114 (2002).
- [125] Qi, L. and Chen, X., A globally convergent successive approximation method for severely nonsmooth equations. SIAM Journal on Control and Optimization, Vol.33, pp.402-418 (1995).
- [126] Qi, L. and Jiang, H., Semismooth Karush-Kuhn-Tucker equations and convergence analysis of Newton and quasi-Newton methods for solving these equations. *Mathematics of Operations Research*, Vol.22, pp.301-325 (1997).
- [127] Qi, L., Shapiro, A. and Ling, C., Differentiability and semismoothness properties of integral functions and their applications. *Mathematical Programming*, Ser.A, Vol. 102, pp. 223-248 (2005).
- [128] Qi, L. and Sun, D., A survey oof some nonsmooth equations and smoothing Newton methods. In: Eberhard, A., Hill, R., Ralph, D. and Glover, B.M. eds. *Progress* in Optimization, Kluwer Academic Publishers, Dordrecht, pp.121-146 (1999).
- [129] Qi, L. and Sun, D., Smoothing functions and smoothing Newton method for complementarity and variational inequality problems. *Journal of Optimization Theory* and Applications, Vol.113, pp.121-147 (2002).
- [130] Qi, L., Sun, D. and Zhou, G., A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities. *Mathematical Programming*, Vol.87, pp.1-35 (2000).
- [131] Qi, L. and Sun, J., A nonsmooth version of Newton's method. Mathematical Programming, Vol.58, pp.353-367 (1993).
- [132] Qi, L. and Tseng, P., Almost smooth functions. In: Takahashi, W. and Tanaka, T. eds. Nonlinear Analysis and Convex Analysis, Yokohama Publishing Co., Tokyo, pp. 405-413 (2003).
- [133] Qi, L., and Wei, Z., On the constant positive linear dependence condition and its applications to SQP methods. SIAM Journal on Optimization, Vol.10, pp.963-981 (2000).

- [134] Qi, L., Wu, S.Y. and Zhou, G., Semismooth Newton methods for solving semiinfinite programming problems. *Journal of Global Optimization*, Vol.47, pp.215-232 (2003).
- [135] Qi, L. and Yang, Y.F., NCP functions applied to Lagrangian globalization for the nonlinear complementarity problem. *Journal of Global Optimization*, Vol.24, pp.261-283 (2002).
- [136] Qi, L., and Yin, H., A strongly semismooth integral function and its application. Computational Optimization and Applications, Vol.25, pp.223-246 (2003).
- [137] Qi, L. and Zhou, G., A smoothing Newton method for minimizing a sum of Euclidean norms. SIAM Journal on Optimization, Vol.11, pp.389-410 (2000).
- [138] Ralph, D., Global convergence of damped Newton's method for nonsmooth equations, via the path search. *Mathematics of Operations Research*, Vol.19, pp.352-389 (1994).
- [139] Ralph, D., Private Communication, (2002).
- [140] Reemtsen, R., Discretization methods for the solutions of semi-infinite programming problems. Journal of Optimization Theory and Applications, Vol.71, pp. 85-104 (1991).
- [141] Reemtsen, R., Some outer approximation methods for semi-infinite optimization problems. Journal of Computational and Applied Mathematics, Vol.53, pp.87-108 (1994).
- [142] Reemtsen, R. and Göner, S., Numerical methods for semi-infinite programming: A survey. In: Reemtsen, R. and Rükmann, J. eds. *Semi-Infinite Programming*, Kluwer Academic Publishers, Boston, pp.195-275 (1998).
- [143] Reemtsen, R. and Rückmann, J., Semi-Infinite Programming, Kluwer Academic Publishers, Boston, 411 pp.(1998).
- [144] Robinson, S.M., Stability theorems for systems of inequalities, Part II: Differentiable nonlinear systems. SIAM Journal on Numerical Analysis, Vol.13, pp.497-513 (1976).

- [145] Robinson, S.M., Normal maps induced by linear transformation. Mathematics of Operations Research, Vol.17, pp.691-714 (1992).
- [146] Roleff, K., A stable multiple exchange algorithm for linear SIP. In: Hettich, R. eds. Semi-Infinite Programming, Necture Notes in Control and Information Sciences, Vol.15, Springer, New York, pp.83-96 (1979).
- [147] Rubinstein, M., Implied binomial tree. Journal of Finance, Vol.49, pp.771-818 (1994).
- [148] Rückmann, J.J. and Shapiro, A., First-order condition in generalized semi-infinite programming. *Journal of Optimization Theory and Applications*, Vol.10, pp.677-691 (1999).
- [149] Ruszczyński, A. and Shapiro, A. eds. Stochastic Programming. In: Handbooks in Operations Research and Management Science, Vol. 10, North-Holland Publishing Company, Amsterdam (2003).
- [150] Sachs, E.W., Semi-infinite programming in control. In: Reemtsen, R. and Rückmann, J. eds. *Semi-Infinite Programming*, Kluwer Academic Publishers, Boston, pp.389-411 (1998).
- [151] Schäfer, E., Ein konstruktionsverfahren bei allgemeiner linearer approximation. Numerische Mathematik, Vol.18, pp.113-126 (1971).
- [152] Shapiro, A., Second-orderderivatives of extremal-value functions and optimality conditions for semi-infinite programs, *Mathematics of Operations Research*, Vol.10, pp.207-219 (1985).
- [153] Shapiro, A., On concepts of directional differentiability. Journal of Optimization Theory and Applications, Vol.66, pp.477-487 (1990).
- [154] Shapiro, A., On Lipschitzian stability of optimal solutions of parametrized semiinfinite programs. *Mathematics of Operations Research*, Vol.19, pp.743-752 (1994).
- [155] Shapiro, A., Directional differentiability of the optimal value function in convex semi-infinite programming, *Mathematical Programming*, Vol.70, pp.149-157 (1995).

- [156] Shapiro, A., First and second order optimality conditions and perturbation analysis of semi-infinite programming problems. In: Reemtsen, R. and Rückmann, J. eds. *Semi-Infinite Programming*, Kluwer Academic Publishers, Boston, pp.103-133 (1998).
- [157] Shaw, F.H. and Geyer, C.J., Estimation and testing in constrained covariance component models. *Biometrika*, Vol.84, pp.95-102 (1997).
- [158] Sheu, R.L., Wu, S.Y. and Fang, S.C., A primal-dual infeasible-interior-point algorithm for linear semi-infinite programming. *Computers and Mathematics with Applications*, Vol.29, pp.7-18 (1995).
- [159] Stein, O., First-order optimality conditions for degenerate index sets in generalized semi-infinite programming problems. *Mathematics of Operations and Research*, Vol.26, pp.565-582 (2001).
- [160] Stein, O. and Still, G., On optimality conditions for generalized semi-infinite programming problems. *Journal of Optimization Theory and Applications*, Vol.104, pp.443-458 (2000).
- [161] Stein, O. and Still, G., Solving semi-infinite optimization problems with interior point techniques. SIAM Journal on Control and Optimization, Vol.42, pp.769-788 (2003).
- [162] Still, G., Generalized semi-infinite programming: Theory and methods. European Journal of Operational Research, Vol.119, pp.301-313 (1999).
- [163] Still, G., Generalized Semi-Infinite Programming: Numerical Aspects. Optimization, Vol.49, pp.223-242 (2001).
- [164] Still, G., Discretization in semi-infinite programming: The rate of convergence. Mathematical Programming, Vol.91, pp.53-69 (2001).
- [165] Sun, D. A further result on an implicit function theorem for locally Lipschitz functions. Operation Research Letter, Vol.28, pp.193-198 (2001).
- [166] Sun, D. and Qi, L., On NCP-functions. Computational Optimization and Applications, Vol.13, pp.201-220 (1999).

- [167] Sun, D. and Qi, L., Solving variational inequality problems via smoothingnonsmooth reformulations. *Journal of Computational and Applied Mathematics*, Vol.129, pp.37-62 (2001).
- [168] Sun, D., Womersley, R.S. and Qi, H.D., A feasible semismooth asymptotically Newton method for mixed complementarity problems. *Mathematical Programming*, Vol.94, pp.167-187 (2002).
- [169] Sun, J., Sun, D. and Qi, L., A squared smoothing Newton method for nonsmooth matrix equations and its applications in semidefinite optimization problems. SIAM Journal on Optimization, Vol.14, pp.783-806 (2004).
- [170] Tanaka, Y., Fukushima, M. and Ibaraki, T., A globally convergent SQP method for semi-infinite nonlinear optimization. *Journal of Computational and Applied Mathematics*, Vol.23, pp.141-153 (1988).
- [171] Teo, K.L., Goh, C.J. and Wong, K.H., A unified computational approach to optimal control problems. *Pitman Monographs and Surveys in Pure and Applied Mathematics*, Vol. 55, Longman Scientific & Technical, New York (1991).
- [172] Teo, K. L. and Jennings, L.S., Nonlinear optimal control problems with continuous state inequality constraints. *Journal of Optimization Theory and Applications*, Vol.63, pp.1-22 (1989).
- [173] Teo, K. L., Rehbock, V. and Jennings, L.S., A new computational algorithm for functional inequality constrained optimization problems. *Automatica*, Vol.29, pp.789-792 (1993).
- [174] Teo, K.L., Yang, X.Q. and Jennings, L.S., Computational discretization algorithms for functional inequality constrained optimization. Annals of Operation Research, Vol.98, pp.215-234 (2000).
- [175] Tichatschke, R., Stetigkeitseigenschaften und konvergenz von folgen diskretisierter semi-infiniter konvexer optimierungsaufgaben. Wiss. Z. TH Karl-Marx-Stadt, Vol.21, pp.577-586 (1979).
- [176] Todd, M.J., Interior-point algorithms for semi-infinite programming. Mathematical Programming, Vol.65, pp.217-245 (1994).

- [177] Tuncel, L. and Todd, J.M., Asymptotic behavior of interior point methods: A view from semi-infinite programming. *Mathematics of Operations Research*, Vol.21, pp.354-381 (1996).
- [178] Ulbrich, M., Semismooth Newton methods for operator equations in function spaces. SIAM Journal on Optimization, Vol.13, pp. 805-842 (2003).
- [179] Wang, Y., Yin, H. and Qi, L., No-arbitrage interpolation of the option price function and its reformulation. *Journal of Optimization Theory and Applications*, Vol.120, pp.629-649 (2004).
- [180] Watson, G.A., A multiple exchange algorithm for multivariate Chebyshev approximation. SIAM Journal on Numerical Analysis, Vol.12, pp.46-52 (1975).
- [181] Watson, G.A., Numerical experiments with globally convergent methods for semiinfinite programming problems. In: Fiacco, A.V. and Kortanek, K.O. eds. Semi-Infinite Programming and Applications, Springer-Verlag, Berlin, pp.193-205 (1983).
- [182] Wets, R.J-B., Stochastic programming: Solution techniques and approximation schemes. In: Bachem, A., Grötschel, M. and Korte, B. eds. *Mathematical Programming: State-of-the-Art 1982*, Springer-Verlag, Berlin, pp.560-603 (1983).
- [183] Wetterling, W., Definitheitsbedingungen fur relative extrema bei optimierungs und approximationsaufgaben. Numerische Mathematik, Vol.15, pp.122-136 (1970).
- [184] Wu, S.Y., Fang, S.C. and Lin, C.J., Relaxed cutting plane method for solving linear semi-infinite programming problems. *Journal of Optimization Theory and Applications*, Vol.99, pp.759–779 (1998).
- [185] Wu, S.Y., Fang, S.C. and Lin, C.J., Analytic center based cutting plane method for linear semi-infinite programming. In: Goberna, M.Á. and López, M.A. eds. Semi-Infinite Programming: Recent Advances, Kluwer Academic Publishers, pp.221-233 (2001).
- [186] Wu, S.Y., Fang, S.C., and Lin, C.J., Solving quadratic semi-infinite programming problems by using relaxed cutting plane scheme. *Journal Computational and Applied Mathematics*, Vol.129, pp.89-104 (2001).

- [187] Wu, S.Y., Li, D.H., Qi, L. and Zhou, G., An iterative method for solving KKT system of the semi-infinite programming. To appear in: *Optimization Methods and Software.*
- [188] Xiao, B. and Harker, P.T., A nonsmooth Newton method for variational inequalities, I: Theory. *Mathematical Programming*, Vol.65, pp.151-194 (1994).
- [189] Xiao, B. and Harker, P.T., A nonsmooth Newton method for variational inequalities, II: Numerical results. *Mathematical Programming*, Vol.65, pp.195-216 (1994).
- [190] Yang, Y. and Qi, L., Smoothing trust region methods for nonlinear complementarity problems with P<sub>0</sub> functions, Annals of Operations Research, Vol.133, pp.99-117 (2005).
- [191] Zhou, G., Sun, D. and Qi, L., Numerical experiments for a class of squared smoothing Newton methods for box constrained variational inequality problems. In: Fukushima, M. and Qi, L. eds. *Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods.* Kluwer Academic Publishers, Dordrecht, pp.421-441 (1999).