SMOOTH and NONSMOOTH NEWTON METHODS for SOLVING SEMI-INFINITE PROGRAMMING

AYSUN TEZEL

DECEMBER 2004

SMOOTH and NONSMOOTH NEWTON METHODS for SOLVING SEMI-INFINITE PROGRAMMING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

AYSUN TEZEL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE
IN
MATHEMATICS

DECEMBER 2004

Approval of the Graduate of Nature and Applied Sciences

Prof. Dr. Canan ÖZGEN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Şafak ALPAY
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Prof. Dr. Bülent Karasözen
Supervisor

Examining Committee Members

Prof. Dr. Bülent Karasözen (METU, MATH and IAM) ____________

Prof. Dr. Gerhard Wilhelm WEBER (METU, IAM) ____________

Assoc. Prof. Tanıl ERGENÇ (METU, MATH and IAM) ____________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name, Last name:

Signature :

ABSTRACT

SMOOTH AND NONSMOOTH NEWTON METHODS FOR SOLVING SEMI-INFINITE PROGRAMMING

Tezel, Aysun
M.Sc., Department of Mathematics
Supervisor: Prof. Dr. Bülent Karasözen

December 2004, 59 pages

In the thesis, a new numerical method which is based on non-smooth Newton's method to solve semi-infinite programming is studied. A problem of semi-infinite programming is an optimization problem in which there may be infinitely many inequality constraints. There are many practical applications of semi-infinite programming problems. For the smooth Newton's method, the results from finitely constrained programming is extended to semi-infinite programming. It will be explored that theoretically and numerically whether the non-smooth approach can lead to a practically efficient solution method for semi-infinite programming.

Keywords: semi-infinite programming, generalized semi-infinite programming, Karush-Kuhn-Tucker systems, first-order optimality conditions, NCP functions, Newton's method, generalized Newton's method, semismooth functions.

ÖZ

YARI SONSUZ OPTİMİZASYON PROBLEMLERİNİN ÇÖZÜMÜ İÇİN DÜZGÜN VE DÜZGÜN OLMAYAN NEWTON METODLARI

Tezel, Aysun
Yüksek Lisans, Matematik Bölümü
Tez Yöneticisi: Prof. Dr. Bülent Karasözen

Aralık 2004, 59 sayfa

Yarı sonsuz ve genelleştirilmiş yarı sonsuz optimizasyon problemleri son yıllarda optimizasyon teorisinde oldukça önem kazanmıştır. Yarı sonsuz optimizasyon problemlerinde sonsuz sayıda eşitsizlikler vardır. Bu problemlerin farklı alanlarda pek çok uygulamaları vardır. Bu tezde yarı sonsuz optimizasyon problemleri incelenerek, bu problemleri çözmek için düzgün olmayan Newton metoduna dayalı yeni bir sayısal yöntem üzerinde çalışılmıştır. Yöntemin yarı sonsuz optimizasyon problemleri için, teorik ve sayısal olarak pratik ve etkin bir çözüm yöntemi olup olmadığı araştırılmaktadır.

Anahtar Kelimeler: Yarı sonsuz optimizasyon problemleri, genelleştirilmiş yarı sonsuz optimizasyon problemleri, Karush-Kuhn-Tucker sistemleri, birinci dereceden optimum koşullar, NCP fonksiyonlar, Newton metodu, genelleştirilmiş Newton metodu, yarı düzgün fonksiyonlar.

To my family

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my supervisor, Prof.Dr. Bülent Karasözen and my coadvisor, Prof.Dr. Gerhard Wilhelm Weber, for their precious guidance and encouragement throughout the research as well as Assoc. Prof. Oliver Stein for patiently guiding, motivating, and encouraging me throughout this study.

Deepest regards and love to my family, who have provided me the love, support and educational background that has enabled me to be successful both in my personal and academic life.

Finally, I am indepted to the members of the Department of Mathematics and all my friends.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

This thesis is devoted to the investigation of a numerical method which is based on nonsmooth Newton method for solving problems of semi-infinite programming. For this purpose, results from finite programming about smooth and nonsmooth Newton methods are included. We also introduce and give a first study about generalized semi-infinite programming, which we can semi-infinite programming regard as a refined version of these problems.

A problem of *semi-infinite programming* (SIP) is an optimization problem of the form

$$\mathcal{P}_{\mathcal{SIP}}(f,h,g,u,v)\left\{\begin{array}{r}\text{Minimize } f(x) \text{ on } M_{\mathcal{SIP}}[h,g], \text{ where}\\ M_{\mathcal{SPI}} := \{x \in \Re^n | h_i(x) = 0 \ (i \in I),\\ \\ g(x,y) \geq 0 \ (y \in Y)\},\end{array}\right\}$$

where $Y \subseteq \Re^m$ is a (possibly) infinite index set. In other words, SIP is an optimization problem in finitely many variables $x = (x_1, \ldots, x_n)^T \in \Re^n$ on a feasible set described by infinitely many constraints.

We assume that $Y \subseteq \Re^m$ is compact, $g(x,y), h_i(x)$ and $f(x)$ are at least two times continuously differentiable on $\Re^n \times Y, \Re^n$ and on $\Re^n$, respectively.

If, in addition, the index set $Y = Y(x)$ depends on the variable x, the program is a semi-infinite problem with *variable index set* or, for short, a problem of *generalized semi-infinite programming* (GSIP):

$$\mathcal{P}_{\mathcal{GSIP}}(f,h,g,u,v) \left\{ \begin{array}{r} \text{Minimize } \ f(x) \ \text{ on } M_{\mathcal{GSIP}}[h,g], \ \text{ where} \\ M_{\mathcal{GSPI}} := \{x \in \Re^n | h_i(x) = 0 \ \ (i \in I), \\ \\ g(x,y) \geq 0 \ \ (y \in Y(x))\}. \end{array} \right\}$$

In order to give more structure to this problem for a better analysis and treatment, we assume thet the sets $Y(x) \subseteq \Re^m$ can be represented as follows:

$$Y(x) := \{y \in \Re^q | u_k(x,y) = 0 \, (k \in K), \ v_l(x,y) \geq 0 \, (l \in L)\} \ \ (x \in \Re^n.)$$

where $h_i : \Re^n \to \Re, i \in I := \{1,\ldots,m\}$, $u_k : \Re^n \times \Re^q \to \Re$, $k \in K := \{1,\ldots,r\}$ and $v_l : \Re^n \times \Re^q \to \Re$, $l \in L := \{1,\ldots,s\}$.

For an introduction to semi-infinite programming problems we refer to the extensive survey by Hettich and Kortanek [5]. Only some of a multitude of applications of semi-infinite programming are Chebychev approximation, robot trajectory planning, the robot maneuverability problem, optimal design problems and optimal filter design.

In the first part of Chapter 2, a formulation of the problems are given, in the second part some real world applications of SIP and GSIP from different fields of science are presented. These applications are, namely, Chebychev and reverse Chebychev approximations, the former can be regarded as a foundation of semi-infinite programming, approximation of a thermo-couple characteristic, which is an application from chemical engineering, optimizing the maneuverability of a robot and time-optimal control problem.

Chebychev approximation is related with approximating a continuous function by a family of continuous functions with an additional parameter and may formulated as semi-infinite programming. Reverse Chebychev approximation is mainly related with enlarging the domain of approximation rather than minimizing the error in approximation and may be formulated as a generalized semi-infinite programming problem.

The approximation of thermo-couple characteristics is formulated by generalized semi-infinite programming and it is a special case of reverse Chebychev approximation. The problem of optimizing the maneuverability of a robot is also an example of generalized semi-infinite programming. In time-minimal control problem, we deal with heating or cooling a given ball from an initial temperature to an end temperature, which is finally formulated as a problem from generalized semi-infinite programming.

In the last part of Chapter 2, a short review of numerical methods to solve problems of SIP and GSIP are listed. There are, namely, discretization methods, exchange methods, methods based on local reduction and the Newton's method. For solving problems from SIP, one of the most important methods is the *discretization method*, in which SIP problem is replaced by a finite programming problem. In general, discretization methods are considered as computationally expensive methods. The methods based on *local reduction* uses mainly, *Reduction Ansatz*, a special stability assumption which allows locally to regard the semi-infinite problem as a finitely constrained one. The Reduction Ansatz, or *Reduction Approach* being an important approach in semi-infinite programming, is given in Chapter 3.3. The algorithms for *discretization* and *exchange methods* are given and some properties of these methods are briefly mentioned. The method based on local reduction and the Newton's method can be directly generalized from semi-infinite programming to generalized semi-infinite programming. In our review of numerical methods, convergence properties of these methods are not mentioned. Instead, we refer to the survey by Hettich and Kortanek [5] and the other references given in this chapter for more details about these numerical methods for solving problems of semi-infinite programming.

In the first part of Chapter 3, to solve a nonlinear system of equations, one of the most important methods: the *Newton's method*, is defined. Only the conditions for convergence are given but convergence under these conditions is not proved. Section 3.1 is devoted to Newton's method, based on the Karush-Kuhn-Tucker system, for solving problems of finite programming. It is known that Newton's method is in general applicable, convergent locally, when applied to problems from finite programming.

In Section 3.2, the most important constraint qualifications for problems of semi-infinite programming as well as the first order optimality conditions for semi-infinite programming are given. The two most important constraint qualifications are *extended linear independence constraint qualification* and *extended Mangasarian Fromovitz constraint qualification*, the latter one is an essential assumption for structural stability results in semi-infinite programming. The *Fritz-John* and *Karush-Kuhn-Tucker conditions*, which are the most important first-order necessary optimality conditions in optimization problems are derived for problems of generalized semi-infinite programming. In Section 3.3, smooth Newton's method for problems of semi-infinite programming is presented which is mainly depend on the Reduction Assumption and Section 3.3.1 is devoted to the convergence of the method for problems of semi-infinite programming. The Newton's method is shown to be quadratically convergent for semi-infinite programming under the appropriate assumptions.

In Section 4.1, the concept of the *generalized Jacobian* is given which is needed for the definition of nonsmooth Newton's method and the convergence properties of the method are given. One of the most important basic assumptions for convergence is the *semismoothness* of the function used in the *nonsmooth* Newton's method. Under semismoothness assumption and nonsingularity of generalized Jacobian of the system at the solution, the semismooth or nonsmooth Newton's method is shown to be locally convergent.

Section 4.2 is devoted to the application of nonsmooth Newton's method for solving problems of *finitely constrained programming*. For this purpose, the Karush-Kuhn Tucker system for problems of finitely constrained programming is derived and the nonlinear complementarity problem functions are used to make the system a semismooth system of equations. Under appropriate conditions, called, *Robinson condition*, local quadratic convergence of the method is proved. In Section 4.3, the Karush-Kuhn Tucker systems are derived for problems of semi-infinite programming, which are derived for upper and lower level problems under the local reduction of the problem. The nonsmooth Newton's method is defined for problems of semi-infinite programming. The assumptions for convergence of the method includes strict complementarity for the upper and lower level problems, an assumption, which is not generally satisfied for problems of semi-infinite programming.

In Chapter 5, some numerical algorithms and applications of these algorithms for semi-infinite programming in MATLAB with results are presented. Three algorithms are presented, the first one is Newton's method, using the usual Jacobian, the second and third one are nonsmooth Newton's method, using the generalized Jacobian, the latter one is the hybrid method to obtain global convergence. The test problems used in this chapter are common problems from literature.
Chapter 6 is the conclusion part for the thesis and includes some possible further research topics.

CHAPTER 2

MOTIVATION OF SIP AND GSIP

2.1 Formulation of SIP and GSIP

An optimization problem of *generalized semi-infinite programming* is of the following form:

$$\mathcal{P}_{\mathcal{GSIP}}(f,h,g,u,v) \left\{ \begin{array}{r} \text{Minimize } f(x) \text{ on } M_{\mathcal{GSIP}}[h,g], \text{ where} \\ M_{\mathcal{GSPI}} := \{x \in \Re^n | h_i(x) = 0 \ (i \in I), \\ \\ g(x,y) \geq 0 \ (y \in Y(x))\}, \end{array} \right\} \tag{2.1}$$

where

$$Y(x) := \{y \in \Re^q | u_k(x,y) = 0 \, (k \in K), \ v_l(x,y) \geq 0 \, (l \in L)\} \ \ (x \in \Re^n). \tag{2.2}$$

Here, we have the objective function $f : \Re^n \to \Re$ and we let

$$h := (h_1, h_2, \ldots, h_m)^T, \ \ u := (u_1, u_2, \ldots, u_r)^T, \ \ v := (v_1, v_2, \ldots, v_s)^T$$

compromise the component functions $h_i : \Re^n \to \Re, i \in I := \{1, 2, \ldots, m\}$, $u_k : \Re^n \times \Re^q \to \Re$, $k \in K := \{1, 2, \ldots, r\}$ and $v_l : \Re^n \times \Re^q \to \Re$, $l \in L := \{1, 2, \ldots, s\}$. Hereby, 'T' denotes transpose. Notice that SIP is a special case of GSIP with $Y(x) \equiv Y$, where the infinite index set does not depend on the variable x.
We assume that $f : \Re^n \to \Re$, $g : \Re^n \times \Re^q \to \Re$, $h_i \ (i \in I)$, $u_k \ (k \in K)$ and $v_l \ (l \in L)$ are continuously differentiable (C^1-) functions. For each C^1-function f we denote the row- and column- vector of the first order partial derivatives $\frac{\partial f}{\partial x_k}(x)$ by $Df(x)$ and $D^T f(x)$, respectively. We also use $D_x g(x,y), D_y g(x,y)$ to

comprise the partial derivatives of g with respect to the coordinates x and y.

2.2 Applications

There are many applications of SIP and GSIP in different fields such as approximation, robotics, boundary and eigenvalue problems, engineering design, optimal control, transportation problems, fuzzy sets, cooperative games, robust optimization and statistics (see, e.g., [5, 27, 36]).
As some further and special practical examples which can, under appropriate assumptions, be modelled semi-infinitely or generalized semi-infinitely, we mention the following problems. Namely there are

- Chebychev and reverse Chebychev approximations,
- approximation of a thermo-couple characteristic,
- optimizing the maneuverability of a robot,
- time-optimal control problems.

In the following, these examples from SIP and GSIP are explained in detail. These examples are taken from [36], to where we refer for more explanations.

Example 2.1.1
Chebsyhev Approximation and Reverse Chebsyhev Approximation
Let $h(y) \in C(\Re^m, \Re)$ and a family of approximating functions $H(x, y)$, where $H \in C(\Re^n \times \Re^m, \Re)$ and $x \in \Re^n$ as the parameter vector.
We want to approximate h by functions $H(x, \cdot)$ in the max-norm (Chebsyhev norm). Let $\|h\|_\infty = \max_{y \in Y} |h(y)|$ with respect to a compact set $Y \subseteq \Re^m$.
To minimize the approximation error $\|h(\cdot) - H(x, \cdot)\|_\infty$ leads to the problem:

$$\mathcal{P}^1_{\mathcal{SIP}}: \quad \min_{x,\epsilon} \ \epsilon \quad \text{s.t.} \quad g^{\pm}(x, y) := \pm(h(y) - H(x, y)) \leq \epsilon \quad \text{for all} \quad y \in Y.$$

This problem of **Chebychev approximation**is a semi-infinite optimization problem. The so-called ***reverse Chebyshev problem*****, however, consists of fixing the approximation error** ϵ **and making the region** Y **as large as**

possible. Suppose that the set $Y = Y(d)$ is parametrized by $d \in \Re^k$ and $v(d)$ denotes the volume of $Y(d)$.

The reverse Chebyshev problem then leads to the GSIP of the following form (ϵ fixed):

$$\mathcal{P}^1_{\mathcal{GSIP}}: \quad \max_{x,d} \ v(d) \quad \text{s.t.} \quad g^{\pm}(x,y) := \pm(h(y) - H(x,y)) \le \epsilon \quad \text{for all} \quad y \in Y(d),$$

where the index set $Y(d)$ depends on the variable d.

Example 2.1.2

Approximation of a Thermo-couple Characteristic

A *thermo-couple* f is some spline of polynomials with different degrees between 3 and 13. It is defined on an interval $[a,b] (a < b)$. From the engineer's point of view, the practical use of a thermo-couple characteristic is very sophisticated. There are several reasons, which call for an *approximation* of the characteristic by a simpler function. For instance, the characteristic is not representable in a closed form, the polynomials' degree is too large, and often only a small region of temperatures is of practical interest. Hence, the engineer may look for an approximation by means of only one polynomial $p(y) = \sum_{k=0}^{n^o} x_{k+1} y^k$ $(y \in \Re)$ of some order n^o such that the domain of approximation is as large as possible, where certain *interpolation* properties ought to be fulfilled and lower and upper *error bounds* are not violated. This optimization problem naturally called a *reverse Chebychev approximation problem.* Therein we put $n = n^o + 2$, $x_o = (x_1, \ldots, x_{n^o+1})^T$, $x^T = (x_o^T, x_n) \in \Re^n$ and

$$\Psi(x_o, y) := \sum_{k=0}^{n^o} x_{k+1} y^k, \ \ \delta(x_o, y) := \Psi(x_o, y) - f(y) \ \ (x_o \in \Re^{n-1}, y \in \Re),$$

referring to some $\alpha \in [a, b)$. Then, we may model our problem in $\Re^n$ in the following way, which can easily be seen to be of generalized semi-infinite character:

$$
\mathcal{P}^2_{\mathcal{GSIP}} \left\{ \begin{array}{rcl} \text{Minimize } -x_n \text{ on } M_{\mathcal{GSI}}, \text{ where} & & \\ M_{\mathcal{GSI}} := \{x \in \Re : \Psi(x_o, y^i) - f^i = & 0 & (i \in \mathtt{I}(x_n)) \\ \delta(x_o, y) - \delta^{\mathtt{I}}(y) & \geq & 0\,(y \in [\alpha, x_n]), \\ -\delta(x_o, y) - \delta^{\mathtt{I}}(y) & \geq & 0\,(y \in [\alpha, x_n]), \\ x_n & - & \alpha \geq 0, \\ -x_n & + & b \geq 0 \\ (x^T = (x_o^T, x_n),\ x_o \in \Re^{n-1} & , & x_n \in \Re)\}. \end{array} \right\} \tag{2.3}
$$

Namely, taking two different numbers $c, d \in \Re \setminus [a, b]$ we would define $Y^\upsilon(x) = [\alpha, x_n]$ $(\upsilon \in \{1, 2\}), Y^3(x) \equiv \{c, d\},\ x \in \Re^n$ $(q^\upsilon = 1, \upsilon \in \{1, 2, 3\})$. However, with the appearance of the x-dependent set $\mathtt{I}(x_n) \subseteq I = \{1, 2, \ldots, m\}$, there is an additional generalization of discrete character in $\mathcal{P}^2_{\mathcal{GSIP}}$. Here, the most important practical situation is given by $\mathtt{I}(x_n) = \{i \in I : y^i \in [\alpha, x_n]\}$. Originally, the points (y^i, f^i) can be interpreted as interpolation points.

Example 2.1.3

Optimizing the Maneuverability of a Robot

A *robot* may be regarded as a structure of connected links, where some geometrical parameters $y^1, y^2, \ldots, y^{q^o} \in \Re$ can be controlled by drive motors. For instance, these parameters are given by the lenght of the links or angles in the joints. The equations of motion for a robot have the form

$$
K = A(y_1)y_3 + H(y_1, y_2),
$$

where $K \in \Re^{q^o}$ stands for the vector of forces (torques) $K^\upsilon(\upsilon \in \{1, 2, \ldots, q^o\}), A(y_1)$ is an inertia matrix, and $H(y_1, y_2)$ is the vector of friction, gravity, centrifugal and Coriolis forces. Given vectors of lower and upper bounds of the components of K and operating region $\Omega \subseteq \Re^{q^o} \times \Re^{q^o}$, we may look for the shape of the set

$$\mathsf{A} = \{y_3 \in \Re^{q^o} : K^{-\upsilon} \leq (A(y_1).y_3 + H(y_1, y_2))^{\upsilon} \leq K^{+\upsilon} (\upsilon \in \{1, 2, \ldots, q^o\}, (y_1, y_2) \in \Omega)\}$$

of accelerations, which can be realized in every point $(y_1, y_2) \in \Omega$. Here, Ω may be an implicitly defined set in the sense of *finite* optimization. Information on A may be used to evaluate the usefulness of a given robot for certain tasks. Hence, the engineer may be interested in *approximation* of A, by means of elementary geometrical body $\mathcal{B}_x \subset \Re^{q^o}$ being as large as possible and contained in A. Hereby, the defining constraints are supposed to depend in a C^1-way on $x \in \Re^n$, being some parameter vector. The set $\mathcal{B}_x$ may for example be a *box* B_x with x_j ($j \in \{1, 2, \ldots, n\}$) being the side lengths, whereby $n = q^o$. In the special case of a square (cube, regular box) S_x, these numbers x_j coincide.

Let $V(x)$ stand for some measure of the size of B_x, for example the *volume*. We are interested in the *maximization* of V, and put $q = 3q^o$, and $y = (y_1^T, y_2^T, y_3^T)^T$. As we might introduce $Y^{\upsilon}(x) = \Omega \times \mathsf{B}_x$ ($x \in \Re^{q^o}$), $\upsilon \in \mathcal{Y} = \{1, 2, \ldots, \upsilon^o\}$, $\upsilon^o = 2q^o$, now, we realize that the following *minimization* problem formulation is of generalized semi-infinite form:

$$\mathcal{P}^3_{\mathcal{GSIP}} \left\{ \begin{array}{c} \text{Minimize } \; -V(x) \; \text{ on } M_{\mathcal{GSI}}, \; \text{ where} \\ M_{\mathcal{GSI}} := \{x \in \Re^n : (A(y_1).y_3 + H(y_1, y_2) - K^-)^{\upsilon} \geq 0 \\ \\ (\upsilon \in \{1, \ldots, 2q^o\}, y \in \Omega \times \mathsf{B}_x), \\ (-A(y_1).y_3 - H(y_1, y_2) + K^-)^{\upsilon} \geq 0 \\ \\ (\upsilon \in \{1, \ldots, 2q^o\}, y \in \Omega \times \mathsf{B}_x)\}. \end{array} \right\} \tag{2.4}$$

Of course, for the evaluation of this problem we may additionally have to require to certain inequality constraints for the parameters x_j ($j \in \{1, 2, \ldots, n\}$), for instance, nonnegativity conditions.

Example 2.1.4

Time-Minimal Control in Heating and Cooling Processes

Let us think that a given ball B consists of a homogeneous material. We study the following problem of heating or cooling B from an initial to a terminal temperature:

$$\mathcal{P}^4_{\mathcal{GSIP}} \left\{ \begin{array}{c} \text{Min } \mathtt{I}(T,u) := T \text{ such that} \\ \text{there is a bounded function} \theta : [0,R] \to \Re, \text{where} \\ \theta|(0,R] \times (0,\infty) \text{is partially differentiable,} \\ u = \theta(R,\cdot)|[0,T] \text{is continuous, and} \\ \theta_t(r,t) = a\Delta\theta(r,t) = \frac{a}{r^2}\frac{\partial}{\partial r}(r^2\theta_r(r,t)) \ \ ((r,t) \in (0,R] \times (0,\infty)), \\ \theta(r,0) = \theta_0 \ \ (r \in [0,R]), \\ \theta(R,T) = \theta_E, \\ T \geq 0, \\ |\sigma_u(R,t)| \ \leq \sigma^* \ \ (t \in [0,T]). \end{array} \right\} \tag{2.5}$$

Here, $\Delta\theta$ represents the Laplacian of θ and R denotes the radius of B. The temperature $\theta(r,t)$ is a function of the radial variable r, where r measures the distance from the center point 0 of B, and of the time t. Moreover, we start with an **initial** temperature θ_0 and finish with an ended **target** (end) temperature $\theta_E > \theta_0$ (or $\theta_E < \theta_0$, respectively). The temperature is essentially governed by the implied heat equation, where $a > 0$ stands for the heat conductivity. Furthermore, $\sigma_u(r,t)$ denotes thermal stress tangential to the boundary $\partial\mathsf{B}$ of B $(r = R)$; σ^* is a given upper bound of the stress.

2.3 Review of Numerical Methods

In this part,we want to give a brief review of numerical methods from literature used to solve problems of semi-infinite as well as generalized semi-infinite programming:

- *discretization methods* [5, 31, 32, 30, 33],
- *exchange methods* [5, 32, 33],
- *methods based on local reduction* [5, 31, 32],
- *Newton's method* [5, 31, 33].

Recall that a problem of *semi-infinite programming* (SIP) is an optimization problem of the form

$$\text{SIP:}\ \min\ \ f(x)\ \text{such that}\ g(x,y) \geq 0\ \text{for all}\ y \in Y \tag{2.6}$$

The discretization, exchange methods and methods based on local reduction basically replace SIP by (a sequence of) finitely constrained programming problems, i.e., problems with only a finite number of constraints. These are solved by applying appropriate linear or nonlinear programming algorithms, for which we refer to an extensive literature. The rough classification is made according to the way how the finite problems are generated. Furthermore, these procedures are examples of superlinearly convergent methods to compute numerically a solution of SIP, under the additional smoothness of the constraint, $g(x, y)$, with respect to x.

The method based on local reduction and Newton's method can directly be generalized from SIP to GSIP. But there arise serious difficulties when trying to generalize the so-called exchange or discretization methods from SIP to GSIP. These difficulties are discussed and convergence results are derived under fairly general assumptions on GSIP in [33].

One of the most important methods for solving an SIP problem is the *discretization method*. In a discretization method, the infinite index set Y is approximated by a sequence of finite subsets $\{Y_k\}_{k\in}$ such that Y_k becomes denser and denser in Y as k goes to infinity. Then, the SIP problem is approximated by a sequence of nonlinear programming problems

$$\text{SIP}(Y_k): \ \min \ f(x) \text{ such that } g(x,y) \geq 0 \text{ for all } y \in Y_k \tag{2.7}$$

such that solution x^k of (2.7), as hoped, converges to a solution of (2.6).

Algorithm ***(Conceptual discretization method)***

Step k: Given a discretization $Y_k \subset Y$

i. Compute a solution x_k ***of*** $SIP(Y_k)$***.***

ii. Stop, if x_k ***is feasible within a fixed accuracy*** $\alpha > 0$***, i.e.*** $g(x_k, y) \geq -\alpha, y \in Y$***. Otherwise, select a finer discretization*** $Y_{k+1} \subseteq Y$.

At each iteration of a discretization method, a nonlinear programming problem is solved. Therefore, discretization methods especially suit for solving problems with a solution at which $g(x^*, \cdot)$ **is (almost) constant on** Y**.**

However, in a general discretization method, the subset $Y_k \subset Y$ **must be sufficiently dense in** Y **when** k **is sufficiently large. This makes the algorithm computationally very expensive. The time needed to verify feasibility with respect to (2.7) and to solve this problem increases dramatically as the cardinality of** Y_k **grows. To reduce the computational cost of discretization methods, a so-called *reduction* technique was introduced which results in reduction based methods. Let the set** Y **be specified by**

$$Y = \{y \in \Re^m : v_j(y) \leq 0, \ j = 1, 2, \ldots, q\} \tag{2.8}$$

where $v_j : \Re^m \to \Re \, (j = 1, 2, \ldots, q)$ **are twice continuously differentiable. The process of a typical reduction based method for solving SIP problem with** Y **specified by (2.8) is as follows. At iteration** k**, compute all**

local minimizers of the problem

$$\min -g(x_k, y) \text{ such that } y \in Y \tag{2.9}$$

Denote by S_k the set of all minimizers of (2.9). Solve the problem

$$\min f(x) \text{ such that } g(x,y) \leq 0, \text{ for all } y \in S_k \tag{2.10}$$

to get the next iterate x_{k+1}. Under some regular conditions, it has been proved that the set S_k is finite and, hence, (2.10) reduces to a nonlinear finitely constrained programming problem.

Under a *compactness assumption* on the feasible sets, a general convergence result for discretization method was obtained. Reduction based and discretization methods have been studied, under certain conditions, these methods possess a global convergence property. However, finding all local minimizers of (2.7) is very difficult and very expensive in computation. Discussion on difficulties and numerical labor in finding all local minimizers of (2.7) can be found in [51].

The *exchange method* is often more efficient than a pure discretization method. This method can be seen a compromise between the discretization method and the continuous reduction approach.

Let x_k be a feasible point for SIP. We assume $f, g \in C^2$ and that the infinite index set Y is defined as the solution set of inequalities with functions $v_l \in C^2(\Re^n, \Re)$:

$$Y = \{y \in \Re | v_l(x) \geq 0, \ (l \in L)\} \ L := \{1, 2, \ldots, q\}.$$

Define the following problem, so-called *lower level* problem

$$Q(x_k) : \min_y \ g(x_k, y) \text{ such that } v_l \geq 0 \ (l \in L).$$

Algorithm *(Conceptual exchange method)*

Step k*: *Given a discretization $Y_k \subset Y$ ***and a fixed, small value*** $\alpha > 0$

i. Compute a solution x_k ***of*** $SIP(Y_k)$***.***
ii. Compute local solutions y_k^i, $i = 1, 2, \ldots, i_k$ ***of*** $Q(x_k)$ ***such that one of them, say*** y_k^1 ***is a global solution, i.e.,*** $g(x, y_k^1) = \max_{y \in Y} g(x_k, y)$.
iii. Stop, if $g(x, y_k^1) \geq -\alpha$***, with a solution*** $x \approx x_k$***. Otherwise, update***

$$Y_{k+1} = Y_k \cup \{y_k^i | i = 1, 2, \ldots, i_k\}.$$

Under appropriate assumptions, we have convergence results for the exchange method.

Newton's methods, called sometimes *KKT-approach*, are directly based on the system of Karush-Kuhn-Tucker necessary optimality conditions. In these methods, briefly, the first order necessary optimality conditions, which is a nonlinear equation, are derived and then solved by Newton's method. In the following chapter, we will explain the method, convergence properties and some generic properties of this method in finite and semi-infinite programming.

Also, in [29], the authors introduced a bi-level solution method for GSIP, which may also applied to SIP.

Finally, we want to emphasize that the numerical solution of GSIP might be much more difficult than the solution of SIP. Because of this reason in [32], the assumptions under which a GSIP can be transformed into a SIP are considered.

CHAPTER 3

SMOOTH NEWTON's METHOD

Assume we want to solve the nonlinear equation $F(x) = 0$, where $F : \Re^n \to \Re^n$ is a continuously differentiable function.
***Newton's method*, i.e.,**

$$x^{k+1} = x^k - (F'(x^k))^{-1}F(x^k) \tag{3.1}$$

where x_0 given, is a classical method for solving the nonlinear equation

$$F(x) = 0, \tag{3.2}$$

where $F : \Re^n \to \Re^n$ is a continuously differentiable function.
The interpretation of (3.1) is that we model F at the current iterate x^k with a linear function

$$M_k(x) := F(x^k) + F'(x^k)(x - x^k)$$

and let the root of M_k be the next iteration. Here, M_k is called the *local linear model.* If $F'(x^k)$ is nonsingular, then $M_k(x^{k+1}) = 0$ is equivalent to (3.1).
It is well-known that if

- **the equation (3.2) has a solution x^*,**
- **$F' : \Omega \to \Re^{n\times n}$ is Lipschitz continuous, and**
- **$F'(x^*)$ is nonsingular,**

then, if x_0 is sufficiently close to x^*, the iteration (3.1) locally converges q-quadratically to x^*.

For general nonlinear problems, the classical ordinary Newton's method reads

$$F'(x_k)\Delta x^k = -F(x^k), \quad x^{k+1} = x^k + \Delta x^k \ (k = 0, 1, \ldots).$$

For $F : D \subseteq \Re^n \to \Re^n$ a Jacobian (n, n)-matrix is required. Sufficiently accurate Jacobian approximations can be computed by symbolic differentiation or by numerical differencing. The above form of the linear system deliberately reflects the actual sequence of computations: first, compute the Newton corrections Δx^k, then improve the iterates x^k to obtain x^{k+1}. For more information about Newton's method we refer to [53].

3.1 Newton's method for Finitely Constrained Programming

Consider a finite program

$$\text{FP:} \ \min_x \ f(x) \quad \text{such that} \quad g_j(x) \geq 0, \ \text{ for all } j \in J = \{1, 2, \ldots, m\}.$$

Under *linear independence constraint qualification* at a local minimizer $\bar{x}$, necessarily the KKT-equations

$$\begin{aligned} D_x\mathcal{L}(\bar{x}, \bar{\mu}) = Df(\bar{x}) - \textstyle\sum_{j \in J_0(\bar{x})} \bar{\mu}_j Dg_j(\bar{x}) &= 0 \\ g_j(\bar{x}) &= 0, j \in J_0(\bar{x}). \end{aligned} \tag{3.3}$$

must hold with (unique) multipliers $\bar{\mu}_j \geq 0, j \in J_0(\bar{x})$, where $\mathcal{L}(x, \mu) := f(x) - \sum_{j \in J_0(\bar{x})} \mu_j g_j(x)$ denotes the *Lagrange function* at $\bar{x}$. By putting $G := (g_j)_{j \in J_0(\bar{x})}$, the Jacobian of the system (3.3) at a solution $(\bar{x}, \bar{\mu})$ is:

$$J(\bar{x}, \bar{\mu}) = \begin{pmatrix} D_x^2\mathcal{L}(\bar{x}, \bar{\mu}) & -D^T G(\bar{x}) \\ DG(\bar{x}) & 0 \end{pmatrix}.$$

The equation (3.3) represents a system of $n + |J_0(\bar{x})|$ equations in equally many unknowns x_i, μ_j. So, we may apply Newton's method to solve the system

$$\begin{pmatrix} D_x\mathcal{L}(\bar{x},\bar{\mu}) \\ g_j(\bar{x}) \end{pmatrix} = 0. \tag{3.4}$$

It is well-known that the Newton's method (locally) has quadratic convergence if the Jacobian $J(\bar{x},\bar{\mu})$ is regular at a solution $(\bar{x},\bar{\mu})$ (cf., e.g., [12],[13]).
We will see that we can expect that in the "general" (generic) case this regularity condition is satisfied. By the following theorem, it is shown that Newton's method is genericly applicable.
Let $\mathcal{P} := C^\infty(\Re^n, \Re)^{1+m}$ denote the set of problem functions $P = (f, g_1, \ldots, g_m)$.

THEOREM 3.1. (Jongen, Jonker, Twilt) [4] *There is an open and dense subset $\mathcal{P}_0 \subseteq \mathcal{P}$ such that for all finite programs $P \in \mathcal{P}_0$, LICQ holds at each feasible point x and at each solution $(\bar{x},\bar{\mu})$ of (3.4) the Jacobian $J(\bar{x},\bar{\mu})$ is regular.*

3.2 First Order Optimality Conditions in SIP

3.2.1 Introduction

In this section, first-order necessary optimality conditions are derived for SIP. The definitions can be found in many references, in particular, we refer to [31]. We assume that the functions are of class C^1.

DEFINITION 3.1. *A feasible point $\bar{x} \in \mathcal{F}$ is called a* ***local minimizer*** **of SIP if there is some $\epsilon > 0$ such that**

$$f(x) - f(\bar{x}) \geq 0 \textbf{ for all } \; x \in \mathcal{F} \textbf{ with } \; \|x - \bar{x}\| < \epsilon.$$

The minimizer $\bar{x}$ is said to be **global** if this relation holds for any $\epsilon > 0$.

LEMMA 3.1. *Let $\bar{x} \in \mathcal{F}$ be a local minimizer of SIP. Then, there cannot exist a strictly feasible descent direction d., i.e., a vector $d \in \Re^n$ such that*

$$Df(\bar{x})d < 0, \; D_x g(\bar{x}, y)d > 0 \; \textit{for all } y \in Y_0(\bar{x})$$

Here, the set $Y_0(\bar{x})$ is called the set of active indices and given by

$$Y_0(\bar{x}) := \{y \in Y : g(\bar{x}, y) = 0\}.$$

In the following we will derive the famous **Fritz John** (FJ) and the **Karush-Kuhn-Tucker** (KKT) optimality conditions.

3.2.2 Constraint Qualifications in SIP

For $\bar{x} \in \mathcal{F}$ we recall *active index* set $Y_0(\bar{x}) = \{y \in Y : g(\bar{x}, y) = 0\}$.
(i) The *Extended Linear Independence Constraint Qualification* (ELICQ) is said to hold at $\bar{x} \in \mathcal{F}$ if the vectors

$D_x g(\bar{x}, y)$, $(y \in Y_0(\bar{x}))$, are linearly independent as a family.

(ii) The *Extended Mangasarian Fromovitz Constraint Qualification* (EMFCQ) is said to hold at $\bar{x}$ if there exists a vector $d \in \mathbb{R}^n$ such that

$D_x g(\bar{x}, y) d > 0$ for all $y \in Y_0(\bar{x})$.

It is well-known that ELICQ implies EMFCQ, i.e., ELICQ is a stronger constraint qualification. The proof of the following theorem is mainly from [31].

THEOREM 3.2. *Let $\bar{x}$ be a local minimizer of SIP. Then, the following holds:*
(a) There exists multipliers $\mu_0, \mu_1, \ldots, \mu_k \geq 0$ and active indices $y_1, y_2, \ldots, y_k \in Y_0(\bar{x}), 0 \leq k \leq n$, such that $\sum_{j=0}^{k} \mu_j = 1$ and

$$\mu_0 Df(\bar{x}) - \sum_{j=1}^{k} \mu_j D_x g(\bar{x}, y_j) = 0 \quad \text{(FJ-condition)}. \tag{3.5}$$

(b) If EMFCQ holds at $\bar{x}$, then there exist multipliers $\mu_1, \ldots, \mu_k \geq 0$ and active indices $y_1, \ldots, y_k \in Y_0(\bar{x}), 0 \leq k \leq n$, such that

$$Df(\bar{x}) - \sum_{j=1}^{k} \mu_j D_x g(\bar{x}, y_j) = 0 \quad \text{(KKT-condition)}. \tag{3.6}$$

Proof. Consider us introduce the set $S := \{Df(\bar{x})\} \bigcup \{-D_x g(\bar{x}, y) | y \in Y_0(\bar{x})\} \subseteq \Re^n$. Since $\bar{x}$ is local minimizer of SIP, there is no strictly feasible descent direction d at $\bar{x}$. This means that, there is no $d \in \Re^n$ with $d^T s < 0$ for all $s \in S$. By assumption, $Y_0(\bar{x})$ is compact, therefore, by continuity of $D_x g(\bar{x}, y)$, S is also compact. We will use the following lemma.

Generalized Gordan Lemma: Let $A \subset \Re^n$ be compact set. Then, exactly one of the following two condition holds

- $0 \in \text{conv}A$.
- There exists some $d \in \Re^n$ such that $d^T s < 0$ for all $s \in A$.

By Generalized Gordan Lemma, since S is compact and there is no $d \in \Re^n$ with $d^T s < 0$ for all $s \in S$, we have $0 \in \text{conv}S$. We need the next lemma.

Caratheodory Lemma: For $A \subset \Re^n$, each $a \in \text{conv}A$ can be represented as a convex combination of (at most) $n+1$ vectors: $a = \sum_{j=1}^{n+1} \lambda_j a_j$.

Then, by Caratheodory Lemma, 0 is a convex combination of at most $n+1$ elements of S, i.e.,

$$\sum_{j=0}^{k} \mu_j s_j = 0 \;\; s_j \in S, \; \mu_j \geq 0, \; \sum_{j=0}^{k} \mu_j = 1, \;\; \text{some} \;\; k \leq n.$$

This implies part (a).

Let us assume $d \in \Re^n$ is a strictly feasible direction, i.e., MFCQ holds. For statement (b) it suffices to show that $\mu_0 \neq 0$ in the representation (a), as division by $\mu_0 > 0$ yields a representation of desired type. Suppose to the contrary that $\mu_0 = 0$ is true, and multiply

$$\mu_0 Df(\bar{x}) - \sum_{j=1}^{k} \mu_j D_x g(\bar{x}, y_j) = 0,$$

by d

$$\mu_0 Df(\bar{x}) d - \sum_{j=1}^{k} \mu_j D_x g(\bar{x}, y_j) d = 0.$$

Then, we obtain the contradiction

$$0 > -\sum_{j=1}^{k} \mu_j D_x g(\bar{x}, y_j) d = 0^T d = 0. \qquad \square$$

3.3 Newton's method for SIP

3.3.1 Introduction

A common method for solving SIP is to apply Newton's method to the necessary optimality conditions. We need some theoretical considerations in order to derive optimality conditions for SIP and GSIP.
The reduction approach is a common way to obtain optimality conditions and Newton-type methods for SIP (see, e.g., [5]). The idea here is to locally transform SIP and GSIP into finite (parametric) optimization problems.

3.3.2 Reduction Ansatz

Reduction Assumption: For any $\bar{y} \in Y_0(\bar{x})$ the following holds:

1. LICQ : $D_y v_l(\bar{x}, \bar{y})(l \in L_0(\bar{x}, \bar{y}))$, are a linearly independent family.
2. SCS : (strict complementary slackness) There exists a multiplier $\gamma \in \Re^{|L_0(\bar{x},\bar{y})|}$ such that $D_y \mathcal{L}(\bar{x}, \bar{y}, \bar{\gamma}) = 0$ and $\bar{\gamma}_l > 0 \quad (l \in L_0(\bar{x}, \bar{y}))$.
3. SOC: (Second order condition) with $\bar{\gamma}$ in 2

   $$\eta^T D_y^2 \mathcal{L}(\bar{x}, \bar{y}, \bar{\gamma}) \eta > 0 \text{ for all } \eta \in T(\bar{x}, \bar{y}) \setminus \{0\},$$

   with the tangent space $T(\bar{x}, \bar{y}) = \{\eta \in \Re^m : D_y v_l(\bar{x}, \bar{y}) \eta = 0 \ (l \in L_0(\bar{x}, \bar{y}))\}$.

THEOREM 3.3. *Let at the feasible point $\bar{x}$ for GSIP and the reduction assumption is satisfied. Then, the subsequent statements can be made:*

- *The active index set is finite,* $Y_0(\bar{x}) = \{\bar{y}_1, \bar{y}_2, \dots, \bar{y}_r\}$*, there exist neighborhoods* $U_{\bar{x}}$ *of* $\bar{x}$ *and* $V_{\bar{y}_j}$ *of* $\bar{y}_j$ *and continuous mappings*

  $y_j : U_{\bar{x}} \to V_{\bar{y}_j}$ *, where* $y_j(\bar{x}) = \bar{y}_j$*,*

  $\gamma_j : U_{\bar{x}} \to \Re^{|L_0(\bar{x},\bar{y})|}$*, where* $\gamma_j(\bar{x}) = \bar{\gamma}_j$*,*

  such that for every $x \in U_{\bar{x}}$ *the value* $y_j(x)$ *is the unique local minimizer of* $Q(x)$ *in* $V_{\bar{y}_j}$ *with corresponding Lagrange multiplier vector* $\gamma_j(x)$*.*

- *With the functions in (a) the following finite reduction holds:* $\bar{x}$ *is a solution of GSIP, locally in a relative neighbourhood* $U_{\bar{x}} \cap \mathcal{F}$*, if and only if* $\bar{x}$ *is a local solution of the so-called reduced problem*

$$GSIP_{red}: \min_{x \in U_{\bar{x}}} f(x) \quad s.t. \quad g_j := g(x, y_j(x)) \geq 0, \quad for\ all\ j = 1, ..., n.$$

For the proof of Theorem (3.3) we refer to [5] and [31].
By this theorem, locally near $\bar{x}$, the problem GSIP is equivalent to $GSIP_{red}(\bar{x})$, which is a common finitely constrained optimization problem. Thus, standard optimality conditions of finitely constrained optimization can be applied to obtain optimality conditions for GSIP.
Let $\bar{x}$ be a local minimum of a GSIP. Under LICQ and reduction assumption the minimum $\bar{x}$ necessarily satisfies the following complete system of optimality conditions with active index set $Y_0(\bar{x}) = \{\bar{y}_1, \bar{y}_2, \dots, \bar{y}_r\}$:

$$Df(x) - \sum_{j=1}^{r} \mu_j D_x \left(g(x, y_j) - \sum_{l \in L_0(\bar{x}, \bar{y}_j)} \gamma_j^l v_l(x, y_j) \right) = 0, g(x, y_j) = 0 \ (j = 1, 2, \dots, r). \tag{3.7}$$

and for $j = 1, 2, \dots, r$

$$D_y g(x, y_j) - \sum_{l \in L_0(\bar{x}, \bar{y}_j)} \gamma_j^l v_l(x, y_j) = 0, \ \ v_l(x, y_j) = 0 \ \ (l \in L_0(x, y_j)). \tag{3.8}$$

The system has as many equations as unknowns x, μ_j, y_j and γ_j. So, again Newton-type methods may be applied to solve SIP and GSIP numerically.

For SIP problems, since v_l does not depend on x, in the first equation, the sum over $\gamma_j^l D_x v_l(x, y_j)$ vanishes.
For common SIP we have the following genericity result which gives the theoretical basis for Kuhn-Tucker methods on solving SIP.

THEOREM 3.4. (Jongen/Zwier) *The set $\mathcal{P}$ of all SIP problems of class C^∞ defined by functions f, g, v_l, $(l \in L)$ contains an open and dense subset $\mathcal{P}_0 \subseteq \mathcal{P}$ such that for all programs $P \in \mathcal{P}_0$ the regularity condition RC is satisfied.*

This result cannot be generalized to GSIP. For a counterexample we refer to [31].

3.3.3 Convergence of Newton's method for SIP

For the following SIP, we will show that Newton's method is convergent

$$SIP: \qquad \min f(x) \text{ s.t. } x \in \mathcal{F},$$

where $\mathcal{F} = \{x \in \Re^n | g(x, y) \geq 0,\ y \in Y\}$ with $Y = \{y \in \Re^r | v_l(y) \geq 0,\ l \in L\}$, where $L = \{1, 2, \ldots, q\}$.
For $\bar{x} \in \mathcal{F}$ we define the set of *active* points

$$Y_0(\bar{x}) = \{\bar{y} \in Y : g(\bar{x}, \bar{y}) = 0\}.$$

Obviously, for feasible $\bar{x} \in \mathcal{F}$, any point $\bar{y} \in Y_0(\bar{x})$ is a (global) minimum of the following parametric optimization problem, so-called *lower level* problem

$$Q(\bar{x}): \min_y \ g(\bar{x}, y) \text{ s.t. } y \in Y$$

Given $\bar{x} \in \mathcal{F}, \bar{y} \in Y$, we define active index set $L_0(\bar{y})$ with respect to $Q(\bar{x})$, $L_0(\bar{y}) = \{l \in L : v_l(\bar{y}) = 0\}$.
We say that at $\bar{y} \in Y$ the *extended linear independence constraint qualification* (ELICQ) is satisfied for $Q(\bar{x})$ if the vectors

$$D_y v_l(\bar{y}),\ \ (l \in L_0(\bar{y}))\text{are linearly dependent as a family.} \tag{3.9}$$

The weaker *extended Mangasarian Fromovitz constraint qualification* (EMFCQ) is said to hold at $\bar{y} \in Y$ if

$$\text{there exists a vector } \mu \text{ such that } \; D_y v_l(\bar{y})\mu > 0, \; (l \in L_0(\bar{y})).$$

Let a local minimum be given $\bar{x} \in \mathcal{F}, \bar{y} \in Y_0(\bar{x})$. If at $\bar{y}$ the EMFCQ is satisfied then, necessarily the following Kuhn-Tucker condition is fulfilled: There exists a multiplier vector $\bar{\gamma} \in \Re^{|L_0(\bar{y})|}$ such that

$$D_y L(\bar{x}, \bar{y}, \bar{\gamma}) = 0, \; \bar{\gamma} \geq 0 \text{ with } L(x, y, \gamma) := g(x, y) - \sum_{l \in L_0(\bar{y})} \gamma_l v_l(y) \tag{3.10}$$

the *Lagrange function*. The following Fritz John type optimality condition holds for SIP.

THEOREM 3.5. *Let be given $\bar{x} \in \mathcal{F}$. Suppose, at any point $\bar{y} \in Y_0(\bar{x})$ the EMFCQ is satisfied for $Q(\bar{x})$. Then, there exist $\bar{y}^j \in Y_0(\bar{x}), \bar{\gamma}^j \in \Re^{|L_0(\bar{y})|}, \bar{\gamma}^j \geq 0, j = 1, \ldots, p$, and multipliers $\bar{\mu}_0, \bar{\mu}_1, \ldots, \bar{\mu}_p \geq 0$, not all zero, such that*

$$\bar{\mu}_0 Df(\bar{x}) - \sum_{j=1}^{p} \bar{\mu}_j D_x g(\bar{x}, \bar{y}^j) = 0. \tag{3.11}$$

If $Y_0(\bar{x}) = \{\bar{y}^1, \bar{y}^2, \ldots, \bar{y}^p\}$ and ELICQ is satisfied at $\bar{x}$ for SIP,

$$D_x g(\bar{x}, \bar{y}^j)(j = 1, 2, \ldots, p) \text{ are linearly independent} \tag{3.12}$$

then, we can assume $\bar{\mu}_0 = 1$ (Kuhn-Tucker condition) and the multipliers $\bar{\mu}_1, \mu_2, \ldots, \bar{\mu}_p$ are uniquely determined.

For later purposes, we summarize second order optimality conditions for SIP. Under reduction assumption, i.e., at any active point $\bar{y}^j \in Y_0(\bar{x})$ condition (3.9) and (3.10) with $\bar{\gamma}^j > 0$ (strict complementary slackness) hold as well as the second order condition,

$$\mu^T D_y^2 L(\bar{x}, \bar{y}^j, \bar{\gamma}^j)\mu > 0 \text{ for all } \mu \in T(\bar{x}, \bar{y}^j) \setminus \{0\}, \tag{3.13}$$

where $T(\bar{x}, \bar{y}^j) = \{\mu \in \Re^r | D_y v_l(\bar{y}^j)\mu = 0 (l \in L_0(\bar{y}^j))\}$. In the following we put $v^j := [v_l, l \in L_0(\bar{y}^j)]^T$ (a matrix with rows v_l). The implicit function theorem applied to the system

$$D_y L(x, y^j, \gamma^j) = 0, \quad v^j(y^j) = 0 \tag{3.14}$$

implies existence of C^1-functions $y^j(x), \gamma^j(x)$ defined on a neighbourhood $U(\bar{x})$ of $\bar{x}$ such that on $U(\bar{x})$ the value $y^j(x)$ is a local solution of $Q(x)$ with corresponding multiplier vector $\gamma^j(x)$ satisfying $y^j(\bar{x}) = \bar{y}^j$, $\gamma^j(\bar{x}) = \bar{\gamma}^j$. By implicitly differentiating (3.14) with respect to x we find the following formula for $Dy^j, D\gamma^j$,

$$-D_{xy} g(\bar{x}, \bar{y}^j) = D_y^2 L(\bar{x}, \bar{y}^j, \bar{\gamma}^j) Dy^j(\bar{x}) - D_y^T v^j(\bar{y}^j) D\gamma^j(\bar{x}) \tag{3.15}$$

$$0 = D_y v^j(\bar{y}^j) Dy^j(\bar{x}). \tag{3.16}$$

The assumptions (3.9) and (3.13) imply that the following matrices being the Jacobian of (3.14) with respect to y, γ

$$\bar{M}^j := \begin{pmatrix} D_y^2 L(\bar{x}, \bar{y}^j \bar{\gamma}^j) & -D_y^T v^j(\bar{y}^j) \\ D_y v^j(\bar{y}^j) & 0 \end{pmatrix} \tag{3.17}$$

are regular. Moreover, these conditions imply that the set $Y_0(\bar{x})$ is finite, $Y_0(\bar{x}) = \{\bar{y}^1, \bar{y}^2, \ldots, \bar{y}^p\}$. Under these strong assumptions, the problem SIP can locally, in a neighbourhood $U(\bar{x})$ of $\bar{x}$, be transformed into the following equivalent finite optimization problem (reduced problem):

$$SIP_{loc}(\bar{x}): \qquad \min\ f(x)\ \text{ such that }\ g^j := g(x, y^j(x)) \geq 0\ \ (j = 1, 2, \ldots, p).$$

Here, the functions $y^j(x)$ are the local solutions of $Q(x)$ constructed above. By applying optimality conditions of finite optimization to the problem $SIP_{loc}(\bar{x})$ we obtain the following sufficient optimality conditions for SIP: Let at all points in $Y_0(\bar{x}) = \{\bar{y}^1, \bar{y}^2, \ldots, \bar{y}^p\}$ the above standard assumptions be satisfied. Assume that at $\bar{x} \in \mathcal{F}$ the ELICQ (3.12) is fulfilled, as well as the Kuhn-Tucker condition (3.11) holds with $\bar{\mu}_0 = 1$ and the second order condition,

$$\xi^T \bar{M}_0 \xi > 0 \text{ for all } \xi \in T \setminus \{0\}, \tag{3.18}$$

with the tangent space $T := \{\xi \in \Re^n | D_x g(\bar{x}, \bar{y}^j)\xi = 0, j = 1, 2, \ldots, p\}$ and

$$\bar{M}_0 := \mu_0 D^2 f(\bar{x}) - \sum_{j=1}^{p} \bar{\mu}_j D_x^2 g(\bar{x}, \bar{y}^j) + \sum_{j=1}^{p} \bar{\mu}_j D^T y^j(\bar{x}) D_y^2 L(\bar{x}, \bar{y}^j, \bar{\gamma}^j) D y^j(\bar{x}). \tag{3.19}$$

Then, $\bar{x}$ is a local minimizer of SIP. Consider $\bar{x} \in \mathcal{F}$ such that at any point $\bar{y}^j \in Y_0(\bar{x}), j = 1, 2, \ldots, p$, the conditions (3.9), (3.13) are satisfied. Let moreover (3.12) and (3.18) be fulfilled. Then, necessarily $\bar{x}, \bar{\mu}, \bar{y}^j, \bar{\gamma}^j, (j = 1, 2, \ldots, p)$, will solve the following system of Karush-Kuhn-Tucker equations of SIP and the corresponding lower level problem $Q(\bar{x})$:

$$\begin{aligned} Df(x) - \textstyle\sum_{j=1}^{p} \mu_j D_x g(x, y^j) = 0, \\ g(x, y^j) - \textstyle\sum_{L_0(\bar{y}^j)} \gamma_l^j v_l(y^j) = 0 \ \ (j = 1, 2, \ldots, p). \end{aligned} \tag{3.20}$$

and for $j = 1, 2, \ldots, p$

$$D_y g(x, y^j) - \sum_{l \in L_0(\bar{y}^j)} \gamma_l^j D_y v_l(y^j) = 0, \ \ v_l(y^j) = 0 \ \ (l \in L_0(y^j)). \tag{3.21}$$

This system consists of $K := n + p + \sum_{j=1}^{p} (r + |L_0(\bar{y}^j)|)$ equations for the K unknowns $x \in \Re^n$, $\mu_j \in \Re$, $y^j \in \Re^r$, $\gamma^j \in \Re^{|L_0(\bar{y}^j)|}$, $j = 1, 2, \ldots, p$. The following Lemma shows that under our assumptions the Jacobian of the system (3.20) and (3.21) is regular at the solution. This in particular implies that the Newton's method applied to (3.20) and (3.21) will locally converge quadratically.
The proof of the following lemma is the modification of the proof of Lemma 1 in [33], which will be stated in the next section, to the SIP.

LEMMA 3.2. *Let $\bar{x} \in \mathcal{F}$ be given such that at any point $\bar{y}^j \in Y_0(\bar{x})$, $j = 1, 2, \ldots, p$ the conditions (3.9), (3.13) are satisfied and let (3.12) and (3.18) be fulfilled. Then, the Jacobian of (3.20) and (3.21) at $\bar{x}, \bar{\mu}, \bar{y}^j, \bar{\gamma}^j$, $j = 1, 2, \ldots, p$, is regular.*

PROOF. The Jacobian of the system (3.20) and (3.21) exhibits all functions evaluated at $\bar{x}, \bar{\mu}, \bar{y}^j, \bar{\gamma}^j$:

$$\begin{pmatrix} D^2 f - \sum_{j=1}^{p} \bar{\mu}_j D_x^2 \bar{g}^j & -B^T & -\bar{\mu}_1 D_{yx} \bar{g}^1 & 0 & \dots & -\bar{\mu}_p D_{yx} \bar{g}^p & 0 \\ B & 0 & 0 & 0 & \dots & 0 & 0 \\ D_{xy} \bar{g}^1 & 0 & D_y^2 L^{\bar{y}^1} & -D_y^T v^1 & \dots & 0 & 0 \\ 0 & 0 & D_y v^1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ D_{xy} \bar{g}^p & 0 & 0 & 0 & \vdots & D_y^2 L^{\bar{y}^p} & -D_y^T v^p \\ 0 & 0 & 0 & 0 & \vdots & D_y v^p & 0 \end{pmatrix} \tag{3.22}$$

where $\bar{g}^j, j = 1, 2, \dots, p$ denotes $g(\bar{x}, \bar{y}^j)$, and $B^T := [D_x^T g(\bar{x}, \bar{y}^1), D_x^T g(\bar{x}, \bar{y}^2), \dots, D_x^T g(\bar{x}, \bar{y}^p)]$ and in the rows $n+1, n+2, \dots, n+p$ we have used the relations $D_y L^{\bar{y}^j}, v^j = 0$. Now, for $j = 1, 2, \dots, p$, we add to the first n columns of (3.22) a combination Dy^j of the columns corresponding to the variable y^j and a combination $D\gamma^j$ of the columns corresponding to the variable γ^j. Then, by using (3.15),(3.16) and (3.17) the matrix (3.22) is transformed into the following matrix without changing the determinant:

$$\begin{pmatrix} M_0 & -B^T & -\bar{\mu}_1 D_{yx} g(\bar{x}, \bar{y}^1) & 0 & \dots & -\bar{\mu}^p D_{yx} L^{\bar{y}^p} & 0 \\ B & 0 & 0 & & \dots & 0 & \\ 0 & 0 & M_1 & & \dots & 0 & \\ \vdots & \vdots & & & \ddots & \vdots & \\ 0 & 0 & 0 & & \dots & M_p & \end{pmatrix} \tag{3.23}$$

Here, the $(n \times n)$ submatrix M_0 has the form

$$M_0 := D^2 f - \sum_{j=1}^{p} \bar{\mu}_j D_x^2 g(\bar{x}, \bar{y}^j) + \sum_{j=1}^{p} -\bar{\mu}_j D_{yx} g(\bar{x}, \bar{y}^j) Dy^j. \tag{3.24}$$

In view of (3.15) and (3.16) it follows that

$$-D_{yx} g(\bar{x}, \bar{y}^j) Dy^j = D^T y^j D_y^2 L^{\bar{y}^j} Dy^j - D^T \gamma^j D_y v^j Dy^j = D^T y^j D_y^2 L^{\bar{y}^j} Dy^j.$$

By substituting this relation into (3.24) we find that M_0 equals the matrix $\bar{M}_0$ in (3.19) with $\bar{\mu}_0 = 1$. In view of our assumptions (3.12) and (3.18), the matrix

$$\begin{pmatrix} M_0 & -B^T \\ B & 0 \end{pmatrix}$$

is regular. Hence, by using (3.17), the matrix (3.23) and, therefore, also the matrix (3.22) are regular. □

In practice, to obtain a globally convergent Newton-type method, one has to apply a globally convergent method for finitely constrained problems to the locally reduced problems $SIP_{loc}(x)$. For SIP such an algorithm is described in [5], Algorithm 7.4. The result for the problems of GSIP can be found in [33].

CHAPTER 4

NON-SMOOTH NEWTON's METHOD

In this chapter Newton's method is generalized to solve non-smooth equations, especially, semismooth equations. Applications, convergence results and properties of generalized Newton's method for finite programming as well as semi-infinite programming are presented.

4.1 A Nonsmooth Version of Newton's Method

In this section, Newton's method for solving a nonlinear equation of several variables is extended to a nonsmooth case by using generalized Jacobian instead of the derivative.
Let us recall the classical Newton's method

$$x^{k+1} = x^k - (F'(x^k))^{-1}F(x^k), \tag{4.1}$$

where x_0 given, is a classical method for solving the nonlinear equation

$$F(x) = 0, \tag{4.2}$$

where $F : \Re^n \to \Re^n$ is a continuously differentiable (i.e., a smooth) function.
Suppose now that F is not a smooth, but a locally Lipschitzian function. Let $\partial F(x^k)$ be the generalized Jacobian of F at x^k defined by Clarke (e.g., [1]). In this case, instead of (4.1), one may use the following procedure to solve (4.2):

$$x^{k+1} = x^k - V_k^{-1}F(x^k) \tag{4.3}$$

with some $V_k \in \partial F(x^k)$.
For vector-valued functions $F : \Re^n \to \Re^m$, written in terms of component functions as $F(x) = (f^1(x), f^2(x), \ldots, f^m(x))^T$. We assume that each f^i (and, hence, F) is Lipshitz near a given point x of interest. *Rademacher's Theorem* asserts that F is differentiable (i.e., each f^i is differentiable) almost everywhere on any neighbourhood of x in which F is Lipschitz. Let us denote the set of points in $\Re^n$ at which F fails to be differentiable by Ω_F. We shall write $JF(y)$ for the usual $(m \times n)$ Jacobian matrix of partial derivatives whenever y is a point at which the necessary partial derivatives exist.

DEFINITION 4.1. *The* ***generalized Jacobian*** *of F at x, denoted by $\partial F(x)$, is the convex hull of all $(m \times n)$ matrices Z obtained as the limit of a sequence of the form $JF(x_i)$, where $x_i \to x$ and $x_i \notin \Omega_F$.*

Symbolically, then, one has

$$\partial F(x) := \text{co}\{\lim_{i \to \infty} JF(x_i) | x_i \to x,\ (i \to \infty), x_i \notin \Omega_F\}. \tag{4.4}$$

In order to show local and global convergence results hold for (4.3) we need to give definition semismoothness of a function F.

Semismooth functions of several variables

DEFINITION 4.2. *A locally Lipschitzian function $F : \Re^n \to \Re^m$ is called* ***semismooth*** *at $x \in \Re^n$ if F is directionally differentiable at x and for all $V \in \partial F(x+d)$ and $d \to 0$*

$$F'(x; d) = Vd + o(\|d\|).$$

In some sense, semismoothness is equivalent to the uniform convergence of directional derivatives in all directions. Semismoothness was introduced by Mifflin [18] for functionals.
Convex functions, piecewise linear functions and smooth functions are examples of semismooth functions. In [24], the definition of semismooth functions was extended to $F : \Re^n \to \Re^m$. It was proved that F is semismooth at x if and only if all its component functions are semismooth.

Definition 4.3. *A function, F is called **strongly semismooth** at x if F is semismooth at x and for all $V \in \partial F(x+d)$ and $d \to 0$*

$$Vd - F'(x;d) = O(\|d\|^2).$$

For other definitions and properties of semismoothness see, e.g., [24].
To prove that the nonsmooth Newton's method locally converges, we need the following lemma.

Lemma 4.1. *If $F : \Re^n \to \Re^m$ a locally Lipschitzian function which is also semismooth at x, then for any $V \in \partial F(x+h), h \to 0$,*

$$Vh - F'(x;h) = o(\|h\|) \tag{4.5}$$

and for any $h \to 0$,

$$F(x+h) - F(x) - F'(x;h) = o(\|h\|). \tag{4.6}$$

For a proof of two properties we refer to [24]. We need the following proposition to prove convergence of the method.

Proposition 4.1. *If all $V \in \partial F(x)$ are nonsingular, then there is a neighbourhood $N(x)$ of x and a constant C such that for any $y \in N(x)$ and any $V \in \partial F(y)$, V is nonsingular and*

$$\left\|V^{-1}\right\| \leq C.$$

Proof. If the conclusion is not true, then there is a sequence $y^k \to x$, $(k \to \infty) V_k \in \partial F(y^k)$ such that either all V_k are nonsingular or $\left\|V_k^{-1}\right\| \to \infty$, $(k \to \infty)$. Since F is locally Lipschitzian, ∂F is bounded in a neighbourhood of x. By passing to a subsequence, we may assume that $V_k \to V$, $(k \to \infty)$. Then V must be singular, a contradiction to the assumption of nonsingularity of $V \in \partial F(x)$. This completes the proof. □

Theorem 4.1. *Suppose that x^* is a solution of (4.2). Let a locally Lipschitzian function $F : \Re^n \to \Re^n$ be given which is semismooth at x^*, and all $V \in \partial F(x^*)$ be nonsingular. Then, the iteration method (4.3) is well-defined and convergent to x^*, locally, in a neighbourhood of x^*.*

Proof. By Propositions 4.1, 4.3 is well-defined in a neighbourhood of x^* for the first step $k = 0$. Now

$$\begin{aligned} \left\|x^{k+1} - x^*\right\| &= \left\|x^k - V_k^{-1}F(x^k) - x^*\right\| \\ &= \left\|x^k - x^* - V_k^{-1}F(x^k)\right\| \\ &\leq \left\|V_k^{-1}[F(x^k) - F(x^*) - F'(x^*; x^k - x^*)]\right\| \\ &+ \left\|V_k^{-1}[V_k(x^k - x^*) - F'(x^*; x^k - x^*)]\right\| \\ &= o(\left\|x^k - x^*\right\|). \end{aligned} \tag{4.7}$$

The last equality is due to Propositions 4.1, 4.5, and 4.6. □

Theorem 4.1 states the local convergence of generalized Newton's method, semismoothness and nonsingularity of generalized Jacobians at the solution are necessary conditions for convergence. Proposition 4.1 and Theorem 4.1 can be found in [24].

4.2 Semismooth Newton's method for Finitely Constrained Programming

In this section, nonsmooth Newton's method will be applied to problem of finitely constrained programming, and also convergence poperties are investigated. This section is mainly taken from [21], with small changes in the definition of problem of finite programming and differentiability assumptions. In the definition of finite programming, the equality constraint is not considered, namely we define a problem of finite programming without equality constraints.

Consider, a problem of the type nonlinear programming with constraints

$$\min f(x) \qquad \text{such that } g(x) \leq 0, \tag{4.8}$$

where f and g are continuously differentiable functions from $\Re^n$ to $\Re$ and $\Re^p$ respectively. Let $N = n + p$. The Karush-Kuhn-Tucker (KKT) system for this problem is:

$$(KKT) \left\{ \begin{array}{c} Df(x) + \sum_{j=1}^{p} \mu_j Dg_j(x) = 0 \\ \\ \mu \geq 0, \;\; g(x) \leq 0, \\ \\ \mu^T g(x) = 0. \end{array} \right\} \tag{4.9}$$

We denote $z^T := (x^T, \mu^T)$. The KKT system plays a central role in the theory and algorithms for problems of nonlinear programming. A local optimal solution of (4.8) is a solution of the KKT system (4.9) if some constraint qualifications such as well-known Mangasarian-Fromovitz condition is satisfied. In this section, we assume f and g are twice continuously differentiable and $D^2 f, D^2 g$ are locally Lipschitzian.

Many iterative methods have been developed to solve KKT systems. We refer to [3] for a comphrehensive treatment of these methods. Since the 1970s, different methods have been developed to formulate the KKT system as a system of nonsmooth equations. But it is a recent approach to construct generalized Newton's method for solving these nonsmooth KKT equations.

We will reformulate our above system of relations by using nonsmooth functions, namely, nonlinear complementarity problem functions, in short, *NCP-functions*:

DEFINITION 4.4. *A function $\phi : \Re^2 \to \mathbb{R}$ is called an NCP-function if*

$$\phi(a, b) = 0 \;\; \textit{if and only if} \;\; a \geq 0, \;\; b \geq 0 \;\; \textit{and} \;\; ab = 0.$$

We will give two important examples of NCP-functions:

Example 1. $\phi_{min}(a, b) := min\{a, b\}$, *minimum function,*

Example 2. $\phi_{FB}(a, b) := \sqrt{a^2 + b^2} - a - b$, *Fischer-Burmeister function.*

They are used in order to formulate nonlinear complementarity problems (NCP's), as an equation. Consider NCP's of the form: Find an $x \in \Re^n$ such that

$$x \geq 0, \quad F(x) \geq 0, \quad x^T F(x) = 0,$$

where $F : \Re^n \to \Re^n$. The NCP has received a lot of attention due to its various applications in operations research, economic equilibrium, and engineering design. A popular way to solve the NCP is via an NCP function to reformulate the NCP as (nonsmooth) equations.

By using NCP-functions, the system becomes

$$H(z) = \begin{pmatrix} Df(x) + \sum_{j=1}^{p} \mu_j Dg_j(x) = 0 \\ \\ \phi(\mu, -g(x)) \end{pmatrix} = 0 \tag{4.10}$$

Definition 4.5. *Suppose $F : \Re^n \to \Re^m$ with $n = m$. We say that F is CD-regular at a point if all $V \in \partial F(x)$ are nonsingular.*

Let us consider, two typical versions of KKT equations, by using the NCP-functions *min function* and *Fisher-Burmeister function*, which are equivalent to (4.9). These two typical versions of KKT equations are semismooth. We denote the Lagrangian of (4.8) by

$$L(x, \mu) := f(x) + \mu^T g(x)$$

and denote its gradient with respect to x by

$$F_\mu(x) := D_x L(x, \mu)$$

By using *min* NCP-function, the KKT system becomes

$$H_1(z) = \begin{pmatrix} Df(x) + \sum_{j=1}^{p} \mu_j Dg_j(x) = 0 \\ \\ \min\{\mu, -g(x)\} \end{pmatrix} = 0. \tag{4.11}$$

By using *Fischer-Burmeister* NCP-function, the KKT system becomes

$$H_2(z) = \begin{pmatrix} Df(x) + \sum_{j=1}^{p} \mu_j Dg_j(x) = 0 \\ \\ \sqrt{\mu_1^2 + (g_1(x))^2} + g_1(x) - \mu_1 \\ \\ \vdots \\ \sqrt{\mu_p^2 + (g_p(x))^2} + g_p(x) - \mu_p \end{pmatrix} = 0. \tag{4.12}$$

The first n rows of $H_1(z)$ and $H_2(z)$ are $F_\mu(x)$. Clearly a smooth function whose derivative is locally Lipschitzian is strongly semismooth everywhere. By definition, a vector-valued function is strongly semismooth if and only if component functions are strongly semismooth. Obviously, the sum and the minimum of two smooth functions with locally Lipschitzian derivatives are strongly semismooth. Hence, H_1 is a strongly semismooth function. We will show that H_2 is also strongly semismooth.

LEMMA 4.2. *Let* $z = (x^T, \alpha)^T, x \in \Re^n, \alpha \in \Re$. *Suppose that* $\psi : \Re^n \to \Re$ *is a twice continuosly differentiable function for which* $D^2\psi$ *is locally Lipschitzian. Let* $\phi : \Re^{n+1} \to \Re$ *be defined by*

$$\phi(z) := \sqrt{(\psi(x))^2 + \alpha^2} + \psi(x) - \alpha.$$

Then, ϕ *is strongly semismooth.*

For the proof of Lemma 4.2, we refer to [21].

THEOREM 4.2. *Both* H_1 *and* H_2 *defined by (4.11) and (4.12) are strongly semismooth.*

PROOF. We only need to prove the results for H_2. By Lemma (4.2), $\phi_j(z) = \sqrt{\mu_j^2 + (g_j(x))^2} + g_j(x) - \mu_j$ is a strongly semismooth function. Since the sum of two LC^1 functions is strongly semismooth and the product of an LC^1 function and a strongly semismooth function is strongly semismooth, where by an LC^1

function, we mean a smooth function with locally Lipschitzian derivative, H_2 is a strongly semismooth function. This completes the proof. $\square$

We will investigate convergence of the following nonsmooth version of Newton's method to solve the equations $H_1(z) = 0$ and $H_2(z) = 0$:

$$z^{k+1} = z^k - W_k^{-1} H(z^k), \quad W_k \in \partial H(z^k). \tag{4.13}$$

By Theorem 4.2 above we see that H_2 is a strongly semismooth function.
Hence, the key thing is to identify the conditions such that both all elements in the generalized Jacobians at the solution point z^* of H_1 and H_2 are nonsingular. This is equivalent to say that both H_1 and H_2 are CD-regular at a KKT-point z^*.

The Robinson condition.

DEFINITION 4.6. *Suppose that $z^T = (x^T, \mu^T) \in \Re^N$. Let $P = \{1, 2, \ldots, p\}$,*

$$I(z) := \{j | j \in P, g_j(x) = 0\},$$

$$I_0(z) := \{j \in I(z) | \mu_j = 0\},$$

$$I_1(z) := \{j \in I(z) | \mu_j > 0\},$$

and

$$G(z) := \{d \in \Re^n | g_j^{'}(x; d) = 0 \quad (j \in I_1(z))\}.$$

A point $z \in \Re^N$ is said to satisfy the strong second-order sufficiency condition for (4.8) if it satisfies the first-order KKT condition (4.9) and if $d^T V d \geq 0$ for all $d \in G(z) \setminus \{0\}$ and all $V \in \partial F_\mu(x)$. We say that a KKT point z of (4.8) satisfies the linear independence condition if $Dg_j(x)\,(j \in I(z))$ are linearly independent.
A point $z \in \Re^N$ is said to satisfy the Robinson condition if it satisfies both the linear indepence condition and the strong second-order sufficiency condition.

THEOREM 4.3. *Suppose that $z^* \in \Re^N$ is a KKT point of (4.8) and satisfies the Robinson condition. Then, both H_1 and H_2 are CD-regular at z^*.*

PROOF. Let $W \in \partial H_1(z^*)$. Then,

$$W = \begin{pmatrix} V & Dg_{I_1}(x^*)^T & Dg_{I_0}(x^*)^T & Dg_J(x^*)^T \\ -Dg_{I_1}(x^*) & 0 & 0 & 0 \\ -\Lambda_0 Dg_{I_0}(x^*) & 0 & \Gamma_0 & 0 \\ 0 & 0 & 0 & E_J \end{pmatrix}, \qquad (4.14)$$

where $V \in \partial F_{\mu^*}(x^*), I_1(z^*) \subseteq I_1 \subseteq I(z^*), I_0 \subseteq (I_0(z^*) \setminus I_1), J = P \setminus (I_1 \cup I_0), E_J$ is the identity matrix of dimension $|J|, \Lambda_0$ and Γ_0 are positive definite diagonal matrices of dimension $|I_0|$ and diagonal elements $\lambda_j \in [0,1]$ and $\gamma_j = 1 - \lambda_j$, respectively, for $j \in I_0$, the order of $j \in P$ is reordered to separate I_1 and I_0 and J. Suppose that

$$W \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix} = 0, \qquad (4.15)$$

where $d_1 \in \Re^n, d_2 \in \Re^{|I_1|}, d_3 \in \Re^{|I_0|}, d_4 \in \Re^{|J|}$. Note that the jth element of d_3 is equal to 0 by (4.15) if $\lambda_j = 0$. Without loss of generality, we may assume that $\lambda_j \in (0,1]$. Then (4.15) implies $d_4 = 0$,

$$V d_1 + Dg_{I_1}(x^*)^T d_2 + Dg_{I_0}(x^*)^T d_3 = 0, \qquad (4.16)$$

$$Dg_{I_1}(x^*) d_1 = 0, \qquad (4.17)$$

and

$$Dg_{I_0}(x^*) d_1 = \Lambda_0^{-1} \Gamma_0 d_3. \qquad (4.18)$$

Suppose that d_1 satisfies (4.17). Then $d_1 \in G(z^*)$. Multiplying (4.16) with d_1^T, by (4.17) and (4.18), we have

$$d_1^T V d_1 + d_3^T \Gamma_0 \Lambda_0^{-1} d_3 = 0.$$

By the strong second-order sufficiency condition and positive definiteness of $\Gamma_0 \Lambda_0^{-1}, d_1 =$

0 and $d_3 = 0$. Now, (4.16) yields

$$Dg_{I_1}(x^*)^T d_2 = 0.$$

By the linear independence condition, $d_2 = 0$. Hence, $d = 0$. This shows that W is nonsingular. Therefore, H_1 is CD-regular at z^*.
For any $z \in \mathbb{R}^N$, H_2 is differentiable at z if and only if $DF_\mu(x)$ exists and $\mu^2 + (g_j(x))^2 > 0$ for all $p \in P$. For these points z,

$$DH_2(z) = \begin{pmatrix} DF_\mu(x) & Dg(x)^T \\ \Lambda Dg(x) & \Gamma \end{pmatrix}, \tag{4.19}$$

where $\Lambda := \text{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_p\}, \Gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_p\}$,

$$\lambda_j = \frac{g_j(x)}{\sqrt{\mu_j^2 + (g_j(x))^2}} + 1 \tag{4.20}$$

and

$$\gamma_j = \frac{g_j(x)}{\sqrt{\mu_j^2 + (g_j(x))^2}} - 1 \tag{4.21}$$

for $j \in P$. By (4.20) and (4.21), we have

$$(\lambda_j - 1)^2 + (\gamma_j + 1)^2 = 1. \tag{4.22}$$

By (4.19),(4.22) and the definition of the generalized Jacobian of H_2, if $W \in \partial H_2(z^*)$, we have,

$$W = \begin{pmatrix} V & Dg(x^*)^T \\ \Lambda Dg(x^*) & \Gamma \end{pmatrix}. \tag{4.23}$$

where $V \in \partial F_{\mu^*}(x^*)$ and again $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_p\}, \Gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_p\}$, and

$$(\lambda_j - 1)^2 + (\gamma_j + 1)^2 \leq 1. \tag{4.24}$$

Suppose that

$$W\begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = 0, \tag{4.25}$$

where $d_1 \in \Re^n, d_2 \in \Re^p$. Use d_{2_j} to denote the components of d_2. Then, (4.25) implies

$$Vd_1 + Dg(x^*)^T d_2 = 0, \tag{4.26}$$

and, for $j \in P$

$$\lambda_j Dg_j(x^*)d_1 + \gamma_j d_{2_j} = 0, \tag{4.27}$$

Let $P_1 := \{j \in P, \lambda_j > 0, \gamma_j < 0\}, P_2 := \{j \in P, \lambda_j = 0\}$ and $P_3 := \{j \in P, \gamma_j = 0\}$. Then $I_1(z^*) \subseteq P_3$. By (4.27) and (4.24), $d_{2_j} = 0$ if $j \in P_2$,

$$Dg_j(x^*)d_1 = 0 \tag{4.28}$$

if $j \in P_3$, and

$$Dg_j(x^*)d_1 = \upsilon_j d_{2_j}, \tag{4.29}$$

where $\upsilon_j = -\gamma_j/\lambda_j > 0$ if $j \in P_1$. Multiplying (4.26) with d_1^T, by (4.28) and (4.29),

$$d_1^T V d_1 + \sum_{j \in P_1} \upsilon_j d_{2_j}^2 = 0.$$

Since $I_1(z^*) \subseteq P_3$, by (4.28), $d_1 \in G(z^*)$. Since $\upsilon_j > 0$ for $j \in P_1$, by the strong second-order sufficiency condition, $d_1 = 0$ and $d_{2_j} = 0$ for $j \in P_1$. Now, (4.26) yields

$$\sum_{j \in P_3} Dg_j(x^*)^T d_{2_j} = 0.$$

Notice $P_3 \subseteq I(z^*)$. By linear independence condition, $d_{2_j} = 0$ for $j \in P_3$. Hence, $d = 0$. This shows that W is nonsingular. Therefore, H_2 is CD-regular at z^*. This completes the proof. □

By Theorems 4.1, 4.2 and 4.3, we have the following theorem

THEOREM 4.4. *Let $H = H_i$ for $(i = 1, 2)$ in Newton's method (4.13). Suppose* *that z^* is a solution of (4.9) and satisfies Robinson condition. Then the iterative method (4.13) is well-defined and $\{z^k\}$, the sequence generated by (4.13), converges to z^* q-quadratic in a neighbourhood of z^*.*

Theorem 4.4 can be found in [21].

4.3 Semismooth Newton's method for SIP

Consider a problem of semi-infinite programming of the form

$$\text{SIP:} \qquad \min_x \ f(x) \ \text{ such that } \quad g(x,y) \leq 0 \quad \text{for all } y \in Y, \tag{4.30}$$

where $Y = \{y \in \Re^m : c(y) \leq 0\}$, $f : \Re^n \to \Re, g : \Re^n \times \Re^m \to \Re$ and $c : \Re^m \to \Re^q$ are twice continuously differentiable functions. Let,

$$Y_0(\bar{x}) = \{y \in Y : g(\bar{x}, y) = 0\}.$$

If $\bar{x}$ is a local minimum of the SIP problem (4.30) and extended Mangasarian-Fromovitz constraint qualification (EMFCQ) holds at $\bar{x}$, i.e., there is a vector $h \in \Re^n$ such that

$$D_x g(\bar{x}, y) h < 0$$

for all $y \in Y_0(\bar{x})$, then there are p positive numbers $\bar{u}_i$ and p vectors $\bar{y}^i \in Y_0(\bar{x})$ such that

$$D_x L(\bar{x}, \bar{u}, \bar{y}) = 0 \tag{4.31}$$

with $p \leq n$. Consider, lower level problem,

$$\min \ -g(\bar{x}, y) \ \text{such that } \ c(y) \leq 0 \tag{4.32}$$

If a constraint qualification for (4.32) holds, then there are q Lagrange multipliers $\bar{w}_j \in \Re^p$ for $j = 1, 2, \ldots, q$ such that for $i = 1, 2, \ldots, p$,

$$D_y \mathcal{L}(\bar{x}, \bar{u}_i, \bar{y}^i, \bar{w}^i) = 0,$$

$$\bar{w}_j^i \geq 0, \quad c_j(\bar{y}^i) \leq 0, \tag{4.33}$$

$$\bar{w}_j^i c_j(\bar{y}^i) = 0 \ \ (i = 1, 2, \ldots, p; j = 1, 2, \ldots, q);$$

where upper level Lagrangian is defined as

$$L(x, u, y) := f(x) + \sum_{i=1}^{p} u_i g(x, y^i),$$

and lower level Lagrangian (modified) look as follows:

$$\mathcal{L}(\bar{x}, u, y, w) := ug(\bar{x}, y) + \sum_{j=1}^{q} w_j c_j(y).$$

The constraint qualification may be one of the following: linear indepedence constraint qualification, Mangasarian-Fromovitz constraint qualification or Slater constraint qualification.
By, (4.31) and (4.33), we have the following equalities and inequalities,

$$Df(\bar{x}) + \sum_{i=1}^{p} \bar{u}_i D_x g(\bar{x}, \bar{y}^i) = 0,$$

$$\bar{u}_i \geq 0, \ \ g(\bar{x}, \bar{y}^i) = 0, \text{ for } i = 1, 2, \ldots, p,$$

$$\bar{u}_i D_y g(\bar{x}, \bar{y}^i) + \sum_{j=1}^{q} \bar{w}_j^i D c_j(\bar{y}^i) = 0, \text{ for } \ i = 1, 2, \ldots, p, \tag{4.34}$$

$$\bar{w}_j^i \geq 0, c_j(\bar{y}^i) \leq 0, \ \text{ for } i = 1, 2, \ldots, p, \ j = 1, 2, \ldots, q,$$

$$\bar{w}_j^i c_j(\bar{y}^i) = 0, \ \text{ for } i = 1, 2, \ldots, p \text{and } j = 1, 2, \ldots, q.$$

Let ϕ be a semismooth NCP function. Then we may reformulate (4.34) as a system of semismooth equations, by using an NCP function ϕ:

$$Df(\bar{x}) + \sum_{i=1}^{p} \bar{u}_i D_x g(\bar{x}, \bar{y}^i) = 0,$$

$$\phi(\bar{u}_i, -g(\bar{x}, \bar{y}^i)) = 0, \ \text{ for } i = 1, 2, \ldots, p,$$

(4.35)

$$\bar{u}_i D_y g(\bar{x}, \bar{y}^i) + \sum_{j=1}^{q} \bar{w}_j^i Dc_j(\bar{y}^i) = 0, \quad (i = 1, 2, \ldots, p),$$

$$\phi(\bar{w}_j^i, -c_j(\bar{y}^i)) = 0, \quad (i = 1, 2, \ldots, p,\ j = 1, 2, \ldots, q).$$

For the ease of presentation, in the following, we used x, u_i, y^i, w_j^i instead of $\bar{x}$, $\bar{u}_i$, $\bar{y}^i$ and $\bar{w}_j^i$, respectively.
Let us define H by

$$H(z) := \begin{pmatrix} Df(x) + \sum_{i=1}^{p} u_i D_x g(x, y^i) \\ \phi(u_1, -g(x, y^1)) \\ \vdots \\ \phi(u_p, -g(x, y^p)) \\ u_1 D_y g(x, y^1) + \sum_{j=1}^{q} w_j^1 Dc_j(y^1) \\ \vdots \\ u_p D_y g(x, y^p) + \sum_{j=1}^{q} w_j^p Dc_j(y^p) \\ \phi(w_1^1, -c_1(y^1)) \\ \vdots \\ \phi(w_q^1, -c_q(y^1)) \\ \vdots \\ \phi(w_1^p, -c_1(y^p)) \\ \vdots \\ \phi(w_q^p, -c_q(y^p)) \end{pmatrix} = 0, \qquad (4.36)$$

where $z = (x, u, y, w) \in \Re^{n+(m+q+1)p}, u \in \Re^p, y = (y^1, y^2, \ldots, y^p) \in \Re^{mp}$ and $w = (w^1, w^2, \ldots, w^p) \in \Re^{qp}$. Let $H = H_2$ if $\phi = \phi_{FB}$ where ϕ_{FB} denotes Fischer-Burmeister NCP function given by

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

In appearance, the system of equations (4.35) is not "totally" equivalent to (4.34). Our system (4.35) allows the case that $\mu_i = 0$, $g(x, y^i) < 0$. But, if there is a vector satisfying (4.35), dropping the part indexed by i where $\mu_i = 0$. Thus, we get a solution of (4.34). On the other hand, a solution of (4.34) obviously satisfies (4.35). In this sense, (4.34) is equivalent (4.35).

A semismooth Newton's method for solving $H(z) = 0$ looks as follows: Having the vector z^k, compute z^{k+1} by:

$$z^{k+1} = z^k - W_k^{-1} H(z^k), \quad W_k \in \partial H(z^k). \tag{4.37}$$

For various examples of semismooth KKT equations, using different NCP functions, we refer to [21].

In [25], the authors proved that under some regularity assumptions including strict complementarity slackness, all $W \in \partial H(z)$ are nonsingular at the solution z of the system, for ϕ_{min} and ϕ_{FB}. Then, by Theorem 4.1, Newton's method converges to the solution. In fact, in this case it converges q-superlinearly (cf., e.g., [25]). If, in addition, f, g and v are three times continuously differentiable, then the convergence is even q-quadratic.

Let $P = \{1, 2, \ldots, p\}$ and $Q = \{1, 2, \ldots, q\}$. In fact, in [25] the assumptions are:

(A1) For all $i \in P$, $\mu_i > 0$,

(A2) The vectors $D_x g(x, y^i)$, $i \in P$ linearly independent,

We define $F(x, \mu, y) := Df(x) + \sum_{i=1}^{p} \mu_i D_x g(x, y^i)$

For $i \in P$, we define

$I(y^i) = \{j : \ j \in Q, \ v_j(y^i) = 0\}$

$J(y^i) = Q \setminus I(y^i)$

and

$L(x, \mu_i, y^i, \gamma^i) := g(x, y^i) - \sum_{j=1}^{q} \gamma_j^i v_j(y^i)$

Let $G(x, y)$ be the set of all $(d, \xi_1, \ldots\ldots, \xi_p)$ satisfying

$d^T D_x g(x, y^i) + \xi_i^T D_y g(x, y^i) = 0$ for $i \in P$ and

$\xi_i^T D v_j(y^i) = 0$ for $i \in P$, $j \in I(y^i)$.

Further, they suppose that the following assumptions hold:

(A3) For each $i \in P$, the vectors $Dv_j(y^i), j \in I(y^i)$ are linearly independent.

(A4) $\gamma_j^i - v_j(y^i) \neq 0$ ($i \in P$ and $j \in Q$)

(A5) For all $(d, \xi_1, \ldots., \xi_p) \in G(x, y) \setminus \{0\}$

$$d^T D_x F(x,\mu,y)d + 2\sum_{i=1}^{p} \mu_i d^T D^2_{xv} g(x,y^i)\xi_i + \sum_{i=1}^{p} \xi_i^T D^2_{yy} L(x,\mu_i,y^i,\gamma^i)\xi_i > 0$$

Let $H = H_1$ for ϕ_{min} and $H = H_2$ for ϕ_{FB}.
Then, we have the following theorem in [25].

THEOREM 4.5. *Let $H := H_i (i = 1,2)$. Suppose that $z^* = (x^*, \mu^*, y^*, \gamma^*)$ is a solution of $H(z) = 0$ and satisfies (A1)-(A5). Then, the iteration method (4.37) is well-defined and the sequence z^k generated by (4.37) converges to z^* q-superlinearly in a neighbourhood of z^*. If, in addition, f, g and v are three times continuously differentiable, then convergence is q-quadratic.*

For the proof of theorem, we refer to [25]. Finally, we want to mention that other versions of Newton's method may also be used to solve nonsmooth equations. For example, in [35] the authors used a quasi-Newton's method for a class of nonsmooth equations. Furthermore, in [25] the authors proposed damped Newton and Gauss-Newton's methods to solve SIP numerically and the algorithm of generalized damped Newton's method was implemented with a small number of examples in MATLAB, numerical results were also given. In the next section, one of the algorithms from [25] and the test problems in [25] are used to find numerical results.

CHAPTER 5

NUMERICAL RESULTS AND ALGORITHMS

5.1 Introduction

In this section, three algorithms for solving

$$H_2(z) = 0, \tag{5.1}$$

is given, where $H : \Re^N \to \Re^N$ is defined in Chapter 4.3, and $N := n+(m+q+1)p$.

ALGORITHM 1.

Step 1. Let $z^0 \in \Re^N$ and let $k = 0$.

Step 2. If $H_2(z^k) = 0$, stop. Otherwise, let d^k be a solution of

$$H_2(z^k) + J^k d = 0,$$

where $J^k := JH_2(z^k)$ and $JH_2(z^k)$ is the usual Jacobian of $H_2(z)$ (if it exists) at z^k.

Step 3. Set $z^{k+1} := z^k + d^k$ and $k := k + 1$. Go to Step 2.

This algorithm is just the usual Newton's method. In order to test Newton's method applied on those problems of this chapter where strict complementary slackness holds for both upper and lower level problems, we use this algorithm. Let

$$\theta(z) = \frac{1}{2} H_2(z)^T H_2(z),$$

Then θ is continuously differentiable with the gradient given by

$$\nabla\theta(z) = W^T H_2(z),$$

where, $W \in \partial H_2(z)$ and where $\partial H_2(z)$ denotes the generalized Jacobian of H_2 at z.

ALGORITHM 2.

Step 1. Let $z^0 \in \Re^N,\ \sigma, \rho \in (0,1),\ \eta > 0,\ a > 2$ and $k = 0$.

Step 2. If $H_2(z^k) = 0$, stop. Otherwise, let d^k be a solution of

$$H_2(z^k) + W^k d = 0, \tag{5.2}$$

where $W^k \in \partial H(z^k)$.

If (5.2) is not solvable, or if

$$\nabla\theta(z^k)^T d^k > -\eta \left\|d^k\right\|^a,$$

set $d^k = -\nabla\theta(z^k)$

Step 3. Let $z^{k+1} := z^k + d^k$ and $k := k + 1$. Go to Step 2.

This algorithm is taken from [25] with step size constant over all iterates. We have only local convergence guaranteed for it. In order to obtain a global convergence algorithm, a globalization strategy from unconstrained optimization will be used. Solving (5.1) is equivalent to finding stationary points of the unconstrained optimization problem

$$\min_{z\in\Re^N} \ \theta(z), \tag{5.3}$$

whenever (5.1) is solvable. We note that θ is continuously differentiable on the whole space $\Re^N$. It is possible to invoke any global convergence algorithm for solving (5.3). We know that the steepest descent method can guarantee the global convergence of (5.3). But one of the disadvantages of this method is that it usually performs poorly, in particularly when the solution point is close. We propose a hybrid method which combines the steepest descent method for solving (5.3) and the generalized Newton's method for solving (5.1). We first check if the

Newton step gives a sufficient decrease to θ. If it does, another point is obtained. Otherwise, the steepest descent method is applied to the minimization problem (5.3) providing a new iteration point. Moreover, when the iterated point is close enough to the solution point of (5.1), the algorithm never requires any steepest descent step.

ALGORITHM 3.

Step 1. Let $z^0 \in \Re^N$, $\sigma \in (0, 1/2)$, $\rho, \beta \in (0, 1)$ and $k = 0$.

Step 2. If $H_2(z^k) = 0$, stop. Otherwise, let d^k be a solution of

$$H_2(z^k) + W^k d = 0, \tag{5.4}$$

where $W^k \in \partial H(z^k)$.

Step 3. If (5.4) is solvable and $\theta(z^k + d^k) \leq \beta\theta(z^k)$, go to Step 5. Otherwise, go to Step 4.

Step 4. Let $\bar{d}^k = -\nabla\theta(z^k)$. Find a minimum nonnegative integer, say, m, such that

$$\theta(z^k + \rho^m \bar{d}^k) \leq \theta(z^k) + \sigma\rho^m \nabla\theta(z^k)^T \bar{d}^k,$$

Let $d^k = \rho^m \bar{d}^k$, and go to Step 5.

Step 5. Let $z^{k+1} := z^k + d^k$ and $k := k + 1$. Go to Step 2.

This algorithm is taken from [7]. There, global convergence of Algorithm 3 is proved under nonsingularity of $W \in \partial H_2(z)$ at the solution z^*.

5.2 Numerical Results

In this part, we present some numerical experiments for three algorithms., i.e., usual Newton's method (*Algorithm 1*), the generalized Newton's method having local convergence (*Algorithm 2*), and the generalized Newton's method having global convergence (*Algorithm 3*). All computational results were undertaken by

MATLAB.

Throughout the computational experiments, the parameters used in Algorithm 2 were $\rho = 0.5$, $a = 2.1$, $\eta = 10^{-8}$ and $\sigma = 10^{-4}$ and for Algorithm 3, the parameters $\sigma = 10^{-4}$, $\rho = 0.5$ and $\beta = 0.95$ were used. In all of the experiments, the termination criteria was taken as $\|H_2(z^k)\| < 10^{-6}$.

The numerical results are summarized in Tables 5.1 through Table 5.7, where with starting points given in the first columns and the numbers in the other columns are the number of iterations corresponding to given algorithms in the first row of each column.

Problems 1-4 are from [25], Problem 5 is from [10], Problem 6 is from [31] with a small change on the constraint function, and Problem 7 is from [5]. In the following, by a *limit point* z, we mean a point that $\|H_2(z)\| \leq 10^{-6}$ and obtained from an algorithm given above, by the *exact solution*, we mean a point that for which $H_2(z) = 0$.

Problem 1.

$f(x) = 1.21\mathrm{e}^{x_1} + \mathrm{e}^{x_2}$, $g(x,y) = y - \mathrm{e}^{x_1+x_2}$,

$Y = [0,1]$, $p = 1$.

This example has one nondegenerate limit point, (-0.09531,0.09531,1.1,1,1.1) and infinitely many degenerate limit points, $(\lambda, \lambda$,0,0,0) with $\lambda < -15$ and exact solution occurs when $\lambda \leq -750$.

Starting point	Algorithm 1	Algorithm 2	Algorithm 3
(0.07,0.07,0.9,0.8,0.9)	5	5	23
(-0.06,0.06,0.8,0.7,0.8)	6	4	30
(-0.05,0.05,0.7,0.6,0.7)	7	7	30
(1,1,1,1,1)	27*	-	31
(1,1,2,2,2)	20*	-	30*

Table 5.1: Number of iterations for Problem 1.

In Table 5.1, the asterisk (*) indicate that the limit point generated by the algorithms is the degenerate limit point, otherwise it is the nondegenerate limit point and the bar (-) denotes that the method did not stop (or may be not convergent).

Since Algorithm 3 is very slowly converging when the initial point is close to the limit point or exact solution, we will not run this algorithm for points sufficiently close to the solution and give only results of Algorithm 3 for points far away from the solution.

Problem 2.

$f(x) = x_1^2 + x_2^2 + x_3^2,\ \ g(x, y) = x_1 + x_2 e^{x_3 y} + e^{2y} - 2\sin(4y),$

$Y = [0, 1],\ p = 1.$

This example has three nondegenerate limit points, (-0.213,-1.36,1.85,0.43,1,1.7),(-0.5,-0.5,0,1,0,6) and (0,0,0,0,0.33,0), with the latter two ones being are exact solutions of the problem. In fact, we observed that for any γ for which $0.23 \leq \gamma \leq 0.33$, the points (0,0,0,0,$\gamma$,0) become exact solutions.

Starting point	Algorithm 1	Algorithm 2
(-0.2,-1.3,1.8,0.4,1,1.7)	4	4
(-0.15,1.2,1.7,0.35,1.1,1.6)	4	4
(-0.1,-1.1,1.6,0.3,1.2,1.5)	5	5
(-0.1,-1,1.5,0.2,1.1,1.4)	5	5
(1,1,1,1,1,1)	8*	9*

Table 5.2: Number of iterations for Problem 2.

In Table 5.2, the asterisk (*) indicate that the limit point generated by the algorithms is the degenerate limit point, otherwise it is the nondegenerate limit point.

Problem 3.

$f(x) = (x_1 - 2x_2 + 5x_2^2 - x_2^3 - 13)^2 + (x_1 - 14x_2 + x_2^2 + x_2^3 - 29)^2,$

$g(x, y) = x_1^2 + 2x_2 y^2 + e^{x_1 + x_2} - e^y,$

$Y = [0, 1],\ p = 1.$

This example has one nondegenerate limit point, namely, (0.72,-1.45,4.92,0,4.92).

Starting point	Algorithm 1	Algorithm 2
(0.7,-1.4,4.8,0.1,4.8)	3	4
(0.6,-1.3,4.7,0.2,4.7)	4	4
(0.5,-1.2,4.6,0.1,4.6)	4	4
(0.4,-1.1,4.5,0,4.5)	6	7

Table 5.3: Number of iterations for Problem 3.

The problem may have other limit points and also exact solution(s) different from given the limit point.

Problem 4.

$f(x) = \frac{1}{3}x_1^2 + \frac{1}{2}x_1 + x_2^2, \ g(x,y) = (1 - x_1^2 y^2)^2 - x_1 y^2 - x_2^2 + x_2,$

$Y = [0,1], \ p = 2.$

This example has one degenerate limit point (-0.75,-0.62,0,0.55,1,0,0,0). The problem may have other limit points and also exact solution(s) different from given limit point.

Starting point	Algorithm 1	Algorithm 2
(-0.7,-0.6,0.1,0.5,0.9,0.1,0.1,0.1)	13*	14*
(-0.6,-0.5,0.2,0.4,1,0.1,0.1,0.1)	9	9
(-0.5,-0.4,0.3,0.5,1,0.2,0.2,0.2)	10	9

Table 5.4: Number of iterations for Problem 4.

Problem 5.

$f(x) = x_1^2 + 3x_2^2 + x_3$

$g(x,y) = \frac{1}{2}(y_1 - x_1 - x_2)^2 + (y_2 - x_2)^2 + x_3,$

$Y = \{y \in \Re^2 : \ c(y) \leq 0\}$, with $c(y) = (c_1(y), c_2(y), c_3(y))$, where

$c_1(y) = -y_1 - y_2,$

$c_2(y) = -y_2$

$c_3(y) = y_1^2 + y_2^2 - 1, \ p = 1.$

This example has one degenerate exact solution, namely, (0,0,0,1,0,0,0,0,0).

In Table 5.5, *Nan* denotes all outputs of the functions (i.e, the final x and y coordinates, norm of H_2 at the final point z, and the value of objective function f

Starting point	Algorithm 1	Algorithm 2
(0.1,0.1,0.1,0.9,0,0,0,0,0)	*Nan*	5
(0.1,0.1,0.1,0.9,0.1,0.1,0.1,0.1,0.1)	3	3
(0.1,0.1,0.1,0.9,0.1,0,0,0,0)	*Nan*	3
(0.5,0.5,0.5,0.4,0.5,0,0,0,0)	*Nan*	5
(0.5,0.5,0.5,0.4,0.5,0.5,0.5,0.5,0.5)	5	6

Table 5.5: Number of iterations for Problem 5.

at the final point x.) are *Nan* at the 2nd iteration due to strict complementarity slackness violated at the initial point, implying that we cannot use the usual Newton's method, where strict complementary slackness violated, which is a fact known in advanced, since the usual Jacobian of reformulated $H_2(z)$ by NCP function, does not exist for that type of points.

Problem 6.

$f(x) = x_1,\;\; g(x,y) = -x_1(\cos y + 2) - x_2 \sin y + 1,$

$Y = [\pi, \frac{3\pi}{2}],\; p = 1.$

This example has one degenerate limit point given by (1,0,1,π,0).

Starting point	Algorithm 1	Algorithm 2
(1.1,0.1,1.1,3,0.1)	3	3
(1.2,0.2,1.2,2.9,0.2)	3	4
(1.3,0.3,1.3,2.8,0.3)	4	4
(1.5,0.5,1.5,2.5,0.5)	5	5
(1.7,0.8,1.7,2.3,0.8)	7	7

Table 5.6: Number of iterations for Problem 6.

Problem 7.

$f(x) = -x_1^2 - (x_2+5)^2,\;\; g(x,y) = 2yx_1 + x_2 - y^2,$

$Y = [-1,1],\; p = 1.$

This example has one degenerate exact solution given by (0,0,10,0,0).

Starting point	Algorithm 1	Algorithm 2
(0.2,0.2,9.8,0.2,0.2)	3	3
(0.3,0.3,9.7,0.3,0.3)	3	4
(0.4,0.4,9.6,0.4,0.4)	4	4
(0.8,0.8,9.2,0.8,0.8)	6	7

Table 5.7: Number of iterations for Problem 7.

CHAPTER 6

CONCLUSION

In this thesis, we deal with a numerical method to solve problems of semi-infinite programming. The numerical applications of smooth and non-smooth Newton's methods for problems of semi-infinite programming are included. The results of these methods for problems of finitely constrained programming are given in detail, since we may think that this kind of optimization forms a basis for problems of semi-infinite programming.

Real-world applications of SIP and GSIP from different fields of sciences and the convergence properties of smooth Newton's method for SIP are given. The Reduction Ansatz which is essential for deriving first-order optimality conditions for problems of SIP as well as problems of GSIP is presented.

Compared with the other methods for solving SIP, the advantage of Newton's methods proposed in this thesis is that only a system of linear equations needs to be solved at each iteration. We give convergence of nonsmooth Newton's method for finitely constrained programming in which the so-called Robinson condition plays a central role.

Further studies should be carried out for the convergence of non-smooth Newton's method for problems of semi-infinite programming. In Chapter 4, some results related with nonsmooth Newton's method from SIP are presented. The assumptions for convergence of the nonsmooth Newton's method includes strict complementarity slackness for both upper level and lower level problems. This result should be generalized to the case where strict complementarity slackness does not hold in upper and/or lower level problems.

The numerical results in Chapter 5 lead to the conclusion that Newton's methods are efficient for the test problems presented here. The number of iterations for convergence given in the tables for each test problem can be considered as quite low. But, the problems considered are preliminary and the number of unknowns are not very large. Further tests are needed with more complex real-world problems.

In the future, the number of active indices at the solution should also be investigated. Since, this number is not known in advance, by using another numerical technique, we should determine or approximate this number. Also, the other types of Newton's methods or combinations with other numerical methods from finitely constrained programming should be used to solve the nonlinear system of equations, which may give more accurate results and a better algorithm.

The next step is to generalize the results from semi-infinite programming to the generalized semi-infinite programming and to investigate whether both smooth and nonsmooth Newton's methods are numerically efficient and practical algorithms for problems of GSIP.

REFERENCES

[1] Clarke, F.H.*Optimization and Nonsmooth Analysis*, Wiley, Newyork, (1983).

[2] Faigle, U.; Kern, W. and Still, G. *Algorithmic Principles of Mathematical Programming*, Kluwer , Dordrecht, (2002).

[3] Fletcher, R. *Practical Methods of Optimization*, John Wiley and Sons, England, (1987).

[4] Jongen, H.Th; Jonker, P. and Twilt, F. *Nonlinear Optimization in Finite Dimensions*, Kluwer, Netherlands, (2000).

[5] Hettich, R. and Kortanek, K. *Semi-infinite Programming:Theory,Methods and Applications*, SIAM Review 3, 380-429, vol. 35, (1993).

[6] Hettich, R. and Still, G. *Second Order Optimality Conditions for Generalized Semi-infinite Programming Problems*, Optimization 34, 195-211, (1995).

[7] Jiang, H. and Qi, L. *A new nonsmooth equations approach to nonlinear complementarity problems*, SIAM J. Cont. Optim. 35, 178-193, (1997).

[8] Jongen, H.Th.; Twilt, F. and Weber, G.-W. *Semi-infinite Optimization:Structure and Stability of the Feasible Set*, J. Optim. Theory Applic. 72, 529-552, (1992).

[9] Jongen, H.Th. and Weber, G.-W. *Nonlinear Optimization:Characterization of Structural Stability*, J.Global Optim. 1, 47-64, (1991).

[10] Jongen, H.Th.; Wetterling, W. and Zwier, G. *On Sufficient Conditions for Local Optimality in Semi-infinite Programming*, Optimization 18, 165-178, (1987).

[11] Jongen, H.Th. and Zwier, G. *On the Local Structure of the Feasible Set in Semi-infinite Optimization*, Optimization 72, 185-202, (1984).

[12] Kelley, C.T. *Iterative Methods for Linear and Nonlinear Equations*, SIAM, Philadelphia, (1995).

[13] Kress, R. *Numerical Analysis*, Springer Verlag, Newyork, (1998).

[14] Klatte, D. and Kummer, B. *Nonsmooth Equations in Optimization:Regularity,Calculus,Methods and Applications*, Kluwer Academic Publishers, Dordrecht, (2002).

[15] Kummer, B. *Generalized Newton and NCP-methods:Convergence,Regularity,Actions*, Control and Optimization 20, 209-244, (2000).

[16] Kummer, B. *Newton's Method Based on Generalized Derivatives for Nonsmooth Functions:Convergence Analysis*, Advances in Optimization, 171-194, (1992).

[17] Kummer, B. *Newton's Method for Nondifferentiable Functions*, Advances in Math.Optim., 114-125, (1998).

[18] Mifflin, R. *Semismooth and Semiconvex Functions in Constrained Optimization*, SIAM J.Control Optim., 957-972, (1977).

[19] Mifflin, R. *An Algorithm for Constrained Optimization with Semismooth Functions*, Math.of Op.Res., 191-207, (1977).

[20] Ouellette, D.V. *Schur Complements and Statistics*, Lin.Alg.Appl. 36, 187-295, (1981).

[21] Qi, L. and Jiang, H. *Semismooth KKT Equations and Convergence Analysis of Newton and Quasi-Newton Methods for Solving These Equations*, Math.of Op.Research 22, 301-325, (1997).

[22] Qi, L. and Martinez, J.M. *Inexact Newton Methods for Solving Nonsmooth Equations*, J.Comput. Appl. Math. 60, 127-145, (1995).

[23] Qi, L. *Convergence analysis of some algorithms for solving nonsmooth equations*, Math. Oper. Res. 18, 227-244, (1993).

[24] Qi, L. and Sun, J. *A Nonsmooth Version of Newton's Method*, Math. Programming 58, 353-367, (1993).

[25] Qi, L.; Wu, S.-Y. and Zhou, G. *Semismooth Newton Methods for Solving Semi-infinite Problems*, Journal of Global Optimization 27, 215-232, (2003).

[26] Robinson, S.M. *Newton's Method for a Class of Nonsmooth Functions*, Industrial Engineering Working Paper, University of Wisconsin, (1998).

[27] Stein, O. *Bi-Level Strategies in Semi-infinite Programming*, Kluwer,(2003).

[28] Stein, O. *On Parametric Semi-infinite Optimization*, Phd Thesis, University of Trier,Department of Math., (1997).

[29] Stein, O. and Still, G. *Solving Semi-infinite Optimization Problems with Interior Point Techniques*, SIAM J.Control Optim. 3, 769-788, vol. 42, (2003).

[30] Still, G. *Discretization in Semi-infinite Programming:The Rate of Convergence*, Math. Program. 91, 53-69, (2001).

[31] Still, G. *Semi-infinite Programming:An Introduction*, preprint, University of Twente, (2004).

[32] Still, G. *Generalized Semi-infinite Programming:Theory and Methods*, European J. of Op. Res. 119, 301-313, (1999).

[33] Still, G. *Generalized Semi-infinite Programming:Numerical Aspects*, Optimization 49, 223-242, (2001).

[34] Sun, D. and Qi, L. *On NCP-Functions*, Comp. Optim. and Appl. 13, 201-220, (1999).

[35] Sun, D. and Han, J. *Newton and Quasi-Newton Methods for a Class of Nonsmooth Equations and Related Topics*, SIAM J. Optim. 7, 463-480, (1997).

[36] Weber, G.-W. *Generalized Semi-infinite Optimization and Related Topics*, Heldermann Verlag, (2003).

[37] Weber, G.-W. *Generalized Semi-infinite Optimization:on Some Foundations*, J. of Comp. Tech. 4, 41-61, (1999).

[38] Dong-Hui, L.; Qi, L., Tam, J. and Wu, S.-Y. *A Smoothing Newton Method for Semi-infinite Programming*, J. of Global Optim. 30, 169-194, (2004).

[39] Nocedal, J. and Wright, S.J.*Numerical Optimization*, Springer, (1999).

[40] Dong-Hui, L.; Qi, L., Tam, J. and Wu, S.-Y. *A Smoothing Newton Method for Semi-infinite Programming*, J. of Global Optim. 30, 169-194, (2004).

[41] Nasher, G. and Sofer, A.*Linear and Nonlinear Programming*, McGraw-Hill, Newyork, (1998).

[42] Kocvara, M.; Outrata, J. and Zowe, J. *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints:Theory,Applications and Numerical Results*, Kluwer, Dordrecht, (1998).

[43] Dong-Hui, L.; Yamashita, N. and Fukushima, M. *Nonsmooth Equation Based BFGS Method for Solving KKT Systems in Mathematical Programming*, J. of Opt. Theory and Appl. 1, 123-167, vol. 109, (2001).

[44] Han, S.P.; Pang, J. and Rangaraj, N. *Globally Convergent Newton Methods for Nonsmooth Equations*, Math. of Op.Res. 3, 586-607, vol. 17, (1992).

[45] Jongen, H.Th. and Weber, G.-W.*On Parametric Nonlinear Programming*, Annals of Op.Res. 27, 253-284, (1990).

[46] Deuflhard, P.*Newton Methods for Nonlinear Problems Affine Invariance and Adaptive Algorithms*, Springer, (2004).

[47] Stein, O. and Still, G.*On Optimality Conditions for Generalized Semi-infinite Programming Problems*, J.of Optim. Theory and Appl. 2, 443-458, vol. 104, (2000).

[48] Rückmann, J.-J. and Stein, O.*On Linear and Linearized Generalized Semi-infinite Optimization Problems*, Annals of Op. Res. 11, 191-208, (2001).

[49] Yamashita, N.*Properties of Restricted NCP Functions for Nonlinear Complementarity Problems*, J.of Optim.Theory and Appl. 3, 701-717, vol. 98, (1998).

[50] Jittorntrum, K.*Solution Point Differentiability without Strict Complemenrtarity in Nonlinear Programming*, Math.Programming 21, 127-138, (1984).

[51] Reemsten, R. and Görner, S.*Numerical methods for semi-infinite programming: a survey.*, Semi-infinite Programming, 195-275, (1998).

[52] Reemsten, R. and Rückmann, J.-J.*Semi-Infinite Programming*, Kluwer Academic Publishers, Boston, (1998).

[53] Kelley, C.T. *Solving Nonlinear Equations with Newton's Method*, SIAM, Philedelphia, (2003).

[54] Alber, Ya.I. and Iusem, A.N. *Extension of Subgradient Techniques for Nonsmooth Optimization in Banach Spaces*, Set Valued Analysis 9, 315-335, (2001).

[55] Markov M. Makela *Survey of Bundle Methods for Nonsmooth Optimization*, Optimization Methods and Software 17, 1-29, (2001).

[56] Luksan, L. and Vlcek, J.*Nonsmooth Equation Method for Nonlinear Nonconvex Optimization*, preprint, Institute of Computer Science, Academy of Sciences of the Czech Republic, (2001).