The Hong Kong Polytechnic University
Department of Applied Mathematics

Exact Penalty Function Methods and Their Applications in Search Engine Advertising Problems

Cheng MA

A thesis submitted in partial fulfilment of
the requirements for the degree of Doctor of Philosophy

April 2012

CERTIFICATE OF ORIGINALITY

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

CHENG MA

Abstract

The penalty function method is one of the most fundamental and useful tools in the modern optimization and has developed into a major research field since 1950s. The study of penalty functions has proliferated in many interesting areas within mathematical optimization society. Nowadays, researchers in optimization fields still pursue unremittingly new breakthroughs in theoretical and algorithmic aspects of penalty function methods. However, it should be mentioned that the currently existing exact penalty functions have a disadvantage that the evaluation of the merit function either needs Jacobian (e.g., augumented Lagrangian penalty functions) or is no longer smooth (l_1 or l_∞ penalty functions).

"It would be a major theoretic breakthrough in nonlinear programming if a simple continuously differentiable function could be exhibited with the property that any unconstrained minimum is a solution of the constrained problem."—Evans, Gould and Tolle [21].

So far, to some extent, the breakthrough of the above quotation has been achieved. Recently, Huyer and Neumair in [38] proposed a new exact and smooth penalty function through adding an auxiliary variable ε to deal with equality constrained minimization problem. The proposed new penalty function enjoys several good properties: (1) good smoothness and exactness properties; (2) bounded below under reasonable conditions; (3) combination of regularization with penalty techniques, which are not possessed by the classical simple and exact penalty functions. Moreover, the new penalty function only involves the information of objective function and constraints function rather than the one of gradient and Hessian matrix. Nevertheless, the new exact penalty function is different from the definition of traditional penalty function, namely, the values of penalty terms are zero on the feasible set and positive outside the feasible set. In spite

of significant differences between the new penalty function and the classical simple and exact penalty functions, naturally, a question arises: What's on earth the relationship between them?

In this thesis, motivated by Huyer and Neumair's work [38], we extend the norm function term of the exact penalty function in [38] to a class of convex functions with a unified framework for some barrier-types and exterior-type penalty functions. We characterize necessary and sufficient conditions for the exact penalty property. Interestingly, we also explore the equivalence between this class of penalty functions and the traditional simple exact penalty functions in the sense of exactness property. These results clarify that this class of penalty functions not only have exactness property as the classical simple penalty function, but also possess the smoothness property, which is not shared by the latter. Furthermore, since the class of penalty functions are bounded below, a revised penalty function method is established. In addition, we verify that, under certain conditions, the proposed algorithm terminates at the optimal solution of the primal problem after finitely many iterations; while in the absence of these conditions, a perturbation theorem for this algorithm can be derived. As a corollary, the global convergence property is presented−namely, every accumulation point of the sequence generated by the algorithm is an optimal solution of the primal problem. The numerical outputs verify the correctness of our developed theory as desired.

We propose a new exact and smooth penalty function for semi-infinite programming problems. The main feature of our penalty function is that we only need to add one variable ε to handle infinitely many constraints. The merit function is considered as a function of x and ε simultaneously which has good smoothness and exactness properties, without involving gradient and Jacobian matrices. We derive another useful property that the minimizer $(x^\star, \varepsilon^\star)$ of the penalty problem satisfies $\varepsilon^\star = 0$ if and only if $x^\star$ solves semi-infinite programming problem. This property demonstrates that the introduced new variable ε can be viewed as an indicator variable of a local (global) minimizer of semi-infinite programming problem. Alternatively, under some mild conditions, the local exactness proof is shown. The numerical results demonstrate that it is an effective and promising approach for solving constrained semi-infinite programming problems. Similarly, we also apply a new exact penalty function to tackle the min-max programming problem and establish necessary and sufficient conditions for the exactness property. In addition, we characterize the second-order sufficient conditions for the local exactness property.

We model and explore the search-based advertising auction as a large scale integer programming problem with more realistic situations, e.g., multiple slots, advertisers with choice behavior and the popular generalized second price mechanism etc.. And then, we apply the new penalty function to this proposed integer programming. In addition, we give numerical simulations to address managerial insights on both operational and theoretical aspects and compare the numerical performances with currently existing algorithms for search engine advertising problems.

Acknowledgments

I would like to express my gratitude to all the people who contributed in making my three years stay at Hong Kong Polytechnic University a fun time, full of invaluable experiences

First of all, I wish to express deep gratitude to my supervisors Chair Professor Liqun Qi and Assistant Professor Xun Li. Thanks to them for providing me a so great study environments and helping me to find the right research direction to follow, for teaching me how to correctly analyze kinds of stuffs with patience, for caring me in every possible way and for careful reading of my manuscripts and useful comments and suggestions on them over the last three years. I am greatly indebted to them, not only professionally, but also personally, for providing me with the opportunity to study at Hong Kong Polytechnic University.

In the last three years, a number of people have given me great generous help and advice. I would like to thank Professor Xiaojun Chen (The Hong Kong Polytechnic University) for paying attention to my research work and for always encouraging me. I would love to appreciate Professor Chi Kin Chan (The Hong Kong Polytechnic University) for his patient guidance, careful reading of my manuscripts and constructive suggestions and pleasant collaboration in the supply chain coordination field. I would love to thank Professor Ka-Fai Cedric Yiu (The Hong Kong Polytechnic University) for careful reading of my manuscripts and constructive suggestions during our joint work. I would love to express my deep appreciation to Professor Tsan Ming Choi (The Hong Kong Polytechnic University) for his careful guidance about how to model search engine advertising auction problem in the joint work. I would love to thank Professor T. L. Yip (The Hong Kong Polytechnic University) for his generous and careful guidance and pleasant collaboration in the supply chain and games fields. I would love to express my deep appreciation to Professor Liansheng Zhang (Shanghai University) for introducing topics about new exact and smooth penalty function and for careful reading my manuscripts and precious comments. I would love to thank Professor Yongjian Yang (Shanghai University) for careful reading my manuscripts and precious comments. I wish to thank Professor Changyu Wang (Qufu Normal University) for introducing me how to generalize the new exact penalty function to a general formulation and compare the differences between it and classical simple and exact penalty function. I also would like to express my deep appreciation Professor Yiju Wang (Qufu Normal University) for long term supports for ont only my study and my life. I wish to thank Professor John Birge (Chicago University) for careful guidance for my researches and generous help.

I also wish to thank many good friends and colleagues: Dr. Congpei An, Hongli An, Chuanxin Bian, Shirong Deng, Zhangyou Chen, Dr. Eugene Lee, Mingjie Gao, Yaohua Hu, Jingzhen Liu, Wenhao Jiang, Haiyang Jin, Boshi Tian, Yan Wei, Cong Xie, Yi Xu, Limeng Yan, Dr. Xinzhen Zhang, Yanfang Zhang, Hongjun Zhou, Jinchuan Zhou, Yang Zhou and Cai Zhu for their friendship and help. They made my time at Hong Kong Polytechnic University more enjoyable.

Last, but by no means least, I would like to thank my parents who gave me the whole life and have always encouraged me to keep on studying. I would like to say that what I am doing means so much more because of you.

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

- **C. Ma**, X. Li, K. F. C. Yiu, Y. J. Yang and L. S. Zhang, On an Exact Penalty Function Method for Semi-Infinite Programming Problems, *Journal of Industrial Management and Optimization*, 8(3): 705-726, 2012.

- **C. Ma**, X. Li, K. F. C. Yiu and L. S. Zhang, A New Exact Penalty Function for Solving Constrained Finite Min-Max Problems, *Applied Mathematics and Mechanics*, 33(2): 253-270,2012.

- C. Y. Wang, **C. Ma**, J. C. Zhou, A New Class of Exact Penalty Functions and Penalty Function Algorithms, Forthcoming in *Journal of Global Optimization*, 2012.

- T. M. Choi, X. Li and **C. Ma**, Search-based Advertising Auctions with Choice-based Budget Constraint, submitted to *Automatica*, (under review).

The author of the thesis participated in another project during the last three years, which results in papers as follows:

- **C. Ma** and C.Y. Wang, A Nonsmooth Levenberg-Marquardt Method for Solving Semi-infinite Programming Problems, *Journal of Computational and Applied Mathematics*, 230(2): 633-642, 2009.

- C. Y. Wang, Q. Liu, **C. Ma**, Smoothing SQP Algorithm for Semismooth Equations with Box Constraints, Forthcoming in *Computational Optimization and Application*, 2012.

- **C. Ma**, Y. C. Lee, C. K. Chan, Y. Wei, Auction and Contracting Mechanisms for Supply Chain Coordination with Agents' Different Risk Preferences, submitted to *European Journal Operation Research.* (under review)

Contents

List of Tables

List of Figures

Notation

$\mathbb{R}^n$	n dimensional Euclidean space
σ	penalty parameter
$\Delta(x, \varepsilon)$	the sum of constraints violation
$f_\sigma(x, \varepsilon)$	penalty function
$\text{dist}(\cdot)$	the distance function
$\alpha, \beta, \gamma, \delta$	given constants
$\nabla f(x)$	the gradient of the function $f(x)$
$g(x)^+$	the maximum between 0 and the function $g(x)$
$\|x\|$	the Euclidean norm of $x \in \mathbb{R}^n$
$N(x; r)$	the r-neighborhood of a point $x \in \mathbb{R}^n$
$[a, b]$	the interval between a and b
$m(V)$	the measure of the set V
$L(x, \lambda, \mu)$	Lagrangian function with Lagrangian multipliers λ and μ
$\text{rank}(A)$	the rank of matrix A
$\text{argmin}(P)$	the optimal minimizer of the problem (P)
$\text{conv}(A)$	the convex hull of the set A
$L(P)$	the set of local (global) minimizer(s) of (P)
$L(P_\sigma)$	the set of local (global) minimizer(s) of the penalty problem of (P)
NLP	nonlinear programming problem
FJ condition	Fritz-John condition
KKT condition	Karush-Kuhn Tucker condition
SIP	semi-infinite programming problem
CTR	click-through-rate
MNL	multinomial logit model

LICQ	linear independence constraint qualification
MFCQ	Mangasarian-Fromovitz constraint qualification
EMFCQ	extended Mangasarian-Fromovitz constraint qualification
GFP	generalized first price
GSP	generalized second price

Chapter 1

Introduction

In this chapter, we mainly present literature review and backgrounds of nonlinear programming problems, semi-infinite programming problems, min-max programming problems and search engine advertisements problems, respectively. Finally, motivations and outlines of this thesis are presented.

1.1 Nonlinear Programming Problems and Penalty Function Methods

Consider the nonlinear programming problem

$$\begin{aligned} \text{(NLP)} \quad \min \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \le 0, \quad \forall\, i = 1, \cdots, m, \\ & h_j(x) = 0, \quad \forall\, j = m+1, \cdots, m+q, \end{aligned} \tag{1.1.1}$$

where the functions $f : \mathbb{R}^n \to \mathbb{R}, g_i(x) : \mathbb{R}^n \to \mathbb{R}, i = 1, 2, \cdots, m$ and $h_j(x) : \mathbb{R}^n \to \mathbb{R}, j = m+1, \cdots, m+q$ are continuously differentiable.

Nonlinear programming problem (NLP) with equality and inequality constraints is an important type of constrained optimization problems. It received much attention in past decades as an important branch of operations research and has wide applications in the fields of finance, economics, medicine science, engineering optimization and management science [2]. There exist a great number of methods in the literature for the NLP such as various feasible direction methods including the method of Zoutendijk [80], the gradient projection method of Rosen [63] and some penalty and barrier function methods such as Lagrangian and augmented Lagrangian penalty methods proposed by Hestenes [33] and Powell [59]. In addition, another important efficient method for the NLP is the sequential quadratic programming (SQP) algorithm (see Ref. [25]).

Among the numerous algorithms for the NLP, the penalty function method is typical and effective. The basic principle of the penalty function method is to augment the objective function utilizing penalty terms for constraint violations in order to transform constrained optimization problems into unconstrained optimization ones. Therefore, by means of penalty functions, we could directly apply kinds of algorithms for tackling the unconstrained optimization problems to solve the primal problem. As the penalty parameter appropriately updated, minimizers of the penalty functions converge to the optimal solutions set of the primal problem. On the other hand, the penalty functions play important roles in exploring novel algorithms. For example, in the sequential quadratic programming method, the penalty functions are usually considered to be the merit functions to make a decision on whether or not accept a test point. In addition, barrier penalty functions form part of the foundation for interior point algorithms for linear programming and semi-definite programming problems.

The penalty function method is one of the most fundamental and useful tools in the modern optimization and has developed into a major research field since 1950s. The original use of the penalty function to solve constrained nonlinear programming problems is generally attributed to Camp [8], whereas significant progress in solving practical problems by the use of penalty functions follows the classical work of Zangwill [76]. The barrier penalty function approach was first proposed by Carroll [10] and used to solve nonlinear inequality constrained problems by Box et al. [5]. Nevertheless, the ill-conditioness usually occurs as the penalty parameter tends to infinity. In order to overcome the difficulties associated with ill-conditioness as the penalty parameter tends to infinity or barrier term approaches zero, Zangwill [76] proposed a nonsmooth absolute value penalty function in which a single unconstrained optimization problem, with a finite threshold penalty parameter, can yield an optimum solution to the primal problem. As a matter of fact, it is the l_1 exact penalty function that we often said. It should be noticed that, though the l_1 penalty function is exact, the minimization of the l_1 penalty function is made difficult by its nonsmoothness property. Another kind of smooth and exact penalty function, which incorporates both a Lagrangian multiplier term and a penalty function term in the auxiliary function is said to be the augumented Lagrangian penalty function or the multiplier penalty function proposed independently by Hestenes [33] and Powell [59]. Furthermore, Rockefellar [61] generalized the augumented Lagrangian penalty function to the inequality constrained optimization problems.

Although the penalty function method is one of the popular methods for the NLP, it also has some disadvantages. For example, if the penalty function is exact and smooth, then it is not necessarily simple; on the other hand, if the penalty function is simple and smooth, then it is not necessarily exact. Here, the referred word "simple" means that the penalty function includes only the objective and constraints functions of the NLP rather than involves any gradients or Hessian matrices information. To address this issue, take the aforementioned three classes of penalty functions for example. It is well known that the l_1 penalty function is a nonsmooth simple and exact penalty function; the quadratic penalty function is a smooth and simple penalty function, but it is not exact; though the augumented Lagrangian penalty function is smooth and exact, it is not simple, because as a matter of fact, the multiplier term contains the gradients of the objective and constraint functions.

"It would be a major theoretic breakthrough in nonlinear programming if a simple continuously differentiable function could be exhibited with the property that any unconstrained minimum is a solution of the constrained problem."—Evans, Gould and Tolle [21].

In the sprit of the quotation, a considerable amount of researchers devoted to designing improved l_1 exact penalty functions or kinds of new penalty functions which tackle nonlinear programming problems both from the theoretical and computational perspectives. So far, to some extent, the breakthrough of the above quotation has been achieved. For example, Huyer and Neumair in the literature [38], via adding a new variable, proposed a new continuously differentiable and exact penalty function to deal with equality constrained minimization problem. We observe that, the new established penalty function only includes the information of the objective and constraint functions of the primal problem besides its exactness and smoothness properties. In addition, it is worth noting that, in this newly established penalty function, the norm function term $\frac{\Delta(x,\varepsilon)}{1-q\Delta(x,\varepsilon)}$ plays a role as a barrier term which prevents feasible iterates from moving too close to the boundary of the feasible region.

1.2 Semi-Infinite Programming Problems

A semi-infinite programming problem (SIP) is an optimization problem with finitely many variables $x \in \mathbb{R}^n$ and infinitely many constraints. In general, the semi-infinite programming problem can be formulated as

$$\begin{aligned} \text{(SIP)} \quad \min \quad & f(x) \\ \text{s.t.} \quad & g(x,v) \leq 0, \quad \forall\, v \in V, \end{aligned} \tag{1.2.2}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^{n+m} \to \mathbb{R}$ are continuously differentiable functions. V is an infinite compact index set.

The origins of the semi-infinite programming problem are related to Chebyshev approximation. The term "semi-infinite programming" stems from [11], where Charnes, Cooper and Kortanek introduced the dual of the Haar problem. The semi-infinite programming problem has wide applications in the real life such as the design of the flutter of aircraft wings [65], the design of multi-input multi-output control system [57] and economic equilibria [39].

The most prominent feature of the semi-infinite programming problem is that it has finitely many variables but infinitely many constraints, which brings great difficulties in designing effective numerical methods. Due to its wide applications, nowadays, the study on the theory and numerical methods has been a very active research area in applied mathematics.

On the theoretical aspect, Krabs [42] obtained Karush-Kuhn-Tucker optimality condition for the semi-infinite programming problem under Slater's constraint qualification. Based on reduction assumptions, second order necessary and sufficient optimality conditions for (SIP) were first derived by Hettich and Jongen [34]. Bonnans and Shapiro [4] derived a zero duality gap of the convex semi-infinite programming problems and its Lagrangian dual problem under Slater's constraint qualification.

On the algorithmic aspect, in recent years, lots of effective methods have been proposed for solving the semi-infinite programming problem, which can be separated into four categories, namely, exchange methods (Hettich and Kortanek [35]), discretization methods (Reemtsen [60], Still [67]), local reduction methods (Goh and Teo [28]) and homotopy methods (Liu [45]). For a thorough study of this subject, refer interested readers to [35].

1.3 Min-Max Programming Problems

Many applications in engineering design, for example, computer-aid design, circuit design, optimal control, risk management in economy etc., give rise to the min-max programming problem in the form:

$$\min \max_{1 \le i \le q} f_i(x),$$

where $f_i : \mathbb{R}^n \to \mathbb{R}, i = 1, 2, \cdots, q$ are continuously differentiable functions.

Since the objective function contains the "max" operator, it is continuous but nondifferentiable even when $f_i(x), i = 1, 2, \cdots, q$ are all differentiable. Nowadays, there are several different types of algorithms that have been commonly taken to solve the min-max programming problem.

- Nonsmooth optimization techniques

  Take the min-max programming problem as a constrained nonsmooth optimization problem and directly utilize the general nonsmooth optimization technologies, e.g., subgradient methods, bundle methods and cutting plane methods (see Refs. [56, 26]).

- Smooth optimization techniques

  Take the specific structure of nondifferentiability into account so as to make use of certain smooth optimization methods, e.g., the most widely used regularization techniques (see Refs. [74, 58]). The main advantage of smooth techniques lies in the fact that the min-max programming problems can be converted into smooth unconstrained optimization ones solved by a standard unconstrained optimization solver. For example, Polak et al. [58] proposed the following approximation problem

$$\min_{x\in\mathbb{R}^n} F_p(x) = \frac{1}{p}\log\sum_{i=1}^{q} e^{pf_i(x)},$$

where $p > 0$ is a smooth precision parameter. This approximation can be derived through Jayne's maximum entropy principle and thus, could also be referred as an exponential penalty function as follows

$$F_p(x) = f(x) + \frac{1}{p}\log\sum_{i=1}^{q} e^{p(f_i(x)-f(x))}.$$

- Equivalently transformed problem

  Another approach is to transform the min-max programming problem into a standard nonlinear programming through an auxiliary variable denoted as follows.

$$\min_{(x,\alpha)\in\mathbb{R}^{n+1}} \{\alpha | f_j(x) - \alpha \leq 0, j = 1, 2, \cdots, q\}. \tag{1.3.3}$$

  This method has been explored in Refs. [78, 54]. Zhou et al. [78] proposed that successive quadratic programming algorithms for solving the min-max programming problems based on the above programming problem (1.3.3). Di Pillo et al. [54] established a smooth penalty function for the transcribed problem (1.3.3).

1.4 An Introduction on Search Engine Advertising Problems

Internet search engines provide a service where sponsored links will be displayed in the front page in addition to search results after a user has searched for a specific term. Keyword advertising, also known as "sponsored links", is a form of targeted online advertising in which the placement of advertisements is triggered by keywords that internet users search or by keywords embedded in online content. Sponsored links offer advertisers a more targeted method of advertising than traditional forms of advertising such as TV commercials, because they are customized. Just for its conveniences, since its inception on search engines in the late 1990s, sponsored links has quickly grown into a leading form of search-based advertising (SA) which has become a principal source of revenue for search engines such as Yahoo and Google. Industrial reports have estimated that the total revenues from SA will reach US $17.6 billion in the U.S. by 2012 (see Ref. [41]).

The major search engines use auctions to sell positions for sponsored links. For this reason, it is also named as the position auction by Varian [71]. Advertisers' bids determine which advertisers' sponsored links are listed and in which order. When an internet user clicks on the advertisement link associated with the keyword, the advertiser is charged by the search engines. The number of advertisements that the search engine can show to a user is limited, and different positions on the search results page have different desirabilities for advertisers. For instance, an advertisement shown at the top of a page is more likely to be clicked than an advertisement shown in the bottom. In another word, different advertising positions have different Click-Through-Rate (CTR for short), the ratio of the number of clicks on the advertisements to the number of appearances of the advertising web links. Therefore, search engines need a system for how to allocate the positions to advertisers and what price to be charged to each advertiser. Auction mechanism is a natural choice, and it is widely-used in the electricity markets .

For the search-based advertising auctions, some of the features that have been considered include: equilibrium properties [19, 48]; algorithm design [64, 15]; mechanism design [18, 20]; parametric estimate [17], incorporating budgets or not, and pay per click or pay per impression schemes. Subsequently, we review some most representative ones as follows.

In the theoretical aspect, Edelman et al. [19] found that generalized second price auction generally does not have equilibrium in dominant strategies, and truth-telling is not an equilibrium strategy. They defined the locally envy free equilibrium, which shows that there exists some position for each advertiser where the advertiser cannot be better off by swapping bids with the advertiser ranked one position above him. Through the definition of locally envy free equilibrium, Feng et al. [22] further presented a pricing model and derive the optimal reserve price for sponsored search advertising from the standpoint of search engine.

In the algorithmic aspect, Rusmevichientong and Williamson [64] developed an adaptive algorithm to show how to determine the bid price to select profitable keywords from the advertisers' point of view. Moreover, Devanur and Hayes [15] demonstrated how an online learning algorithm with the budgeted advertisers can achieve a competitive ratio of $1-\varepsilon$ under random permutations without the assumption of bidders'arrival distribution. Nevertheless, they all consider this simplified problem without the requirements of multiple slots and the second price payment. When the actual situations are considered, the model and the resulting optimization problem are much more complex yet challenging.

1.5 Motivations and Outlines

To a great extent, this thesis is mainly motivated by Huyer and Neumaier' s work [38]. The proposed new exact barrier penalty function in [38] has significant differences with traditional definitions of penalty functions, which motivates us to explore whether in some sense, the equivalence between the new exact penalty function and traditional penalty functions can be established. On the other hand, the new exact barrier penalty function enjoys exactness, smoothness and lower-bounded properties which are not all shared by the commonly used penalty functions. Based on these good properties, the new exact penalty function motivates us to make an extension to a more general representation with unified framework for some exterior-type penalty functions and barrier-type penalty functions. In addition, it also enlightens us to investigate the scope of the applicability to semi-infinite programming problems, min-max programming problems and some other practical problems encountered in our life.

In Chapter 2, some preliminaries about basic definitions, optimality conditions of nonlinear programming problems and categories of general penalty functions are presented.

In Chapter 3, noticing the fact that the barrier term in the proposed exact penalty function in [38] makes computing interior points of a feasible region essential, we first generalize it to a class of convex functions so that the established penalty functions present a unified framework for some barrier type and exterior type penalty functions.

Subsequently, necessary and sufficient conditions for exact penalty property are developed. We prove that, the optimal solution takes the form (x, ε) with $\varepsilon = 0$ without requirement of conditions that the penalty parameter tends to infinity. In particular, in spite of great differences, we characterize the equivalence between the new class of exact penalty function and the traditional simple exact penalty functions in the sense of exactness. We end up Chapter 3 with a feasible and revised penalty function algorithm. The finite termination property and convergence analysis are presented associated with numerical experiments.

As important extensions, inspired by [38], we propose a new exact and smooth penalty function for the semi-infinite programming problem in Chapter 4. The main feature is that, we only need to add one variable ε to handle infinitely many constraints. We show that, under some constraint qualification, a local optimal solution has the expression of $(x^\star, 0)$. In addition, we derive a useful property that the minimum $(x^\star, \varepsilon^\star)$ of the penalty problem satisfies $\varepsilon^\star = 0$ if and only if $x^\star$ solves the original the semi-infinite programming problem. This property tells us ε can be viewed as an indicator variable of the local (global) minimizer of the semi-infinite programming problem. For the specific structure of the semi-infinite programming, we derive the error bound assumptions as sufficient conditions that ensure local exactness property of the proposed exact penalty function. In the end, we make numerical experiments so as to verify the developed theory correct and compare numerical performances with existing algorithms for the semi-infinite programming problem.

Chapter 5 introduces a new exact and smooth penalty function to tackle the equality constrained min-max programming problem. Like what previously mentioned, the new penalty function method proposed in this chapter can be categorized into equivalently transformed techniques for solving the min-max programming problems. Since our proposed new exact penalty function has good smoothness, it could be dealt by any smooth unconstrained optimization methodologies. Similar to Chapter 4, we show that, under some constraint qualification, a local optimal solution has the expression of $(x^\star, \theta^\star, 0)$. In addition, we derive a useful property that the minimum $(x^\star, \theta^\star, \varepsilon^\star)$ of the penalty problem satisfies $\varepsilon^\star = 0$ if and only if $x^\star$ solves the original min-max programming problem and $\theta^\star$ is the optimal objective function value. Furthermore, we provide that, under some constraint qualification, the penalty function possesses the exactness property, and especially, the local exactness proof is shown, where the objective and constraint functions are not necessarily smooth. In this case, ε may control both the weight of the penalty terms and the regularization of the nonsmooth terms. In the end of Chapter 5, we characterize the second-order sufficient conditions for the local exactness property.

As applications, we utilize the newly proposed exact and smooth penalty functions to an increasingly popular search engine advertising problem in Chapter 6. We model and formulate the search-based advertising problem into a large-scale integer programming problem based on more realistic situations,which are not yet explored in the current literature. For example, (1) multiple slots; (2) generalized second price mechanism; (3) advertisers with their own choice behaviors; (4) quality score factor; (5) reserve price and (6) more than one keyword can match the query. In view of the established integer programming model, in the beginning of Chapter 6, we apply the proposed exact and smooth penalty functions to nonlinear mixed discrete programming. As a special case, we apply the new penalty function to solve search engine advertising problems. Furthermore, we provide numerical simulations to address managerial problems on both operational and theoretical aspects and compare numerical performances with currently existing algorithms for solving search engine advertising problems.

This thesis ends with some conclusions and thoughts on future research directions.

Chapter 2

Preliminaries

This chapter focuses on some basic definitions and existing theoretical results on nonlinear programming problems and penalty functions.

2.1 Nonlinear Programming Problems

Associated with the NLP defined in (1.1.1), the Lagrange function $L : \mathbb{R}^n \times \mathbb{R}^{m+q} \to \mathbb{R}$ is defined as

$$L(x, \lambda) := f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) + \sum_{j=m+1}^{m+q} \mu_j h_j(x).$$

For simplicity in exposition, denote the feasible set

$$\Omega = \left\{ x \in \mathbb{R}^n \,\middle|\, \begin{array}{l} g_i(x) \leq 0, i = 1, \cdots, m; \\ h_j(x) = 0, j = m+1, \cdots, m+q \end{array} \right\} \tag{2.1.1}$$

and assume it to be compact. For $x^\star \in \Omega$, if $f(x^\star) \leq f(x)$ for all feasible point $x \neq x^\star$, $x^\star$ is said to be a global minimum of the NLP. Accordingly, $f(x^\star)$ is said to be the optimal objective function value. Furthermore, if $f(x^\star) < f(x)$ holds for all $x \in \Omega, x \neq x^\star$. $x^\star$ is said to be a strict global minimum. Assume that $x^\star \in \Omega$, if there exists a δ-neighborhood of $x^\star, N(x^\star, \delta)$, where $\delta > 0$, such that for all $x \in N(x^\star, \delta) \cap \Omega, f(x) \leq f(x^\star)$, then $x^\star$ is said to be a local minimum. If $f(x^\star) < f(x)$ for $x \in N(x^\star, \delta) \cap \Omega, x \neq x^\star$, $x^\star$ is called a strict local minimum.

In what follows, we introduce constraint qualifications and optimality conditions of the NLP.

A Karush-Kuhn-Tucker (KKT for short) triplet for the NLP is a triplet $(\bar{x}, \bar{\lambda}, \bar{\mu}) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^q$

$$\begin{aligned} \nabla f(\bar{x}) + \sum_{i=1}^{m} \bar{\lambda}_i \nabla g_i(\bar{x}) + \sum_{j=m+1}^{m+q} \bar{\mu}_j \nabla h_j(\bar{x}) = 0, \\ \bar{\lambda}_i g_i(\bar{x}) = 0, \quad i = 1, 2, \cdots, m, \\ \bar{\lambda}_i \geq 0, \quad i = 1, 2, \cdots, m, \end{aligned}$$

where $\bar{x}$ is said to be a KKT point for the NLP and $(\bar{\lambda}, \bar{\mu})$ is said to be the Lagrange multipliers vector.

As stated in [2], the Karush-Kuhn-Tucker conditions may not hold at a local minimum of the NLP if some regularity conditions are not satisfied. By regularity conditions, we refer to various conditions imposed on the problem data, some of which may only depend on the constraints, while some of which may depend on the objective function in addition to the constraint functions. When regularity conditions are independent of the objective functions, they are well known as the constraint qualifications in the literature. For various constraint qualifications appeared in the literature, we refer readers to the text book [2]. In what follows, we list some frequently appeared in most literature.

For any $x \in \mathbb{R}^n$, we define the index sets:

$$\begin{aligned} I_0(x) &:= \{i = 1, 2, \cdots, m | g_i(x) = 0\}, \\ I_+(x) &:= \{i = 1, 2, \cdots, m | g_i(x) \geq 0\}, \\ I_-(x) &:= \{i = 1, 2, \cdots, m | g_i(x) < 0\}. \end{aligned}$$

The linear independence constraint qualification ([23]) (LICQ) holds at $x \in \mathbb{R}^n$ if the gradients $\nabla g_i(x), i \in I_0(x), \nabla h_j(x), j = m+1, \cdots, m+q$ are linearly independent.

The Mangasarian-Fromovitz constraint qualification ([47]) (MFCQ) holds at $x \in \mathbb{R}^n$ if the set of equality constraint gradients $\nabla h_j(x), j = m+1, \cdots, m+q$ are linearly independent and there exists a vector $\mathbf{w} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ such that

$$\begin{aligned} &\nabla g_i(x)^\top \mathbf{w} < 0, i \in I_0(x); \\ &\nabla h_j(x)^\top \mathbf{w} = 0, j = m+1, \cdots, m+q. \end{aligned}$$

It is well known [27] that MFCQ is equivalent to boundedness of the set of Lagrange multiplier vectors for which the KKT conditions hold. As stated in [23], LICQ implies that the set of Lagrange multiplier vectors consists of a unique vector, and so is trivially bounded. In addition, it should be noticed from [52] that the LICQ implies the MFCQ.

Constraint qualifications play important roles in establishing optimality conditions. Subsequently, first-order necessary and sufficient optimality conditions of the NLP are presented.

With the help of a cone of feasible directions, the first-order sufficient condition for the NLP was shown in [2]. Let us recall definitions of the cone of feasible directions and the cone of descent directions.

Definition 2.1.1 *([2]) Let S be a nonempty set in $\mathbb{R}^n$, and let $\bar{x} \in clS$. The cone of feasible directions of S at $\bar{x}$, denoted by D, is given by*

$$D = \{\mathbf{d} \in \mathbb{R}^n | \mathbf{d} \neq \mathbf{0}, \textit{and } \bar{x} + \lambda \mathbf{d} \in S \textit{ for all } \lambda \in (0, \delta) \textit{ for some } \ \delta > 0\}.$$

Each nonzero vector $\mathbf{d} \in D$ is called a *feasible direction*. Likewise, define $F_0 = \{\mathbf{d} \in \mathbb{R}^n | \nabla f(\bar{x})^\top \mathbf{d} < 0\}$ as the cone of a *descent direction*. At $\bar{x} \in clS, F_0 \cap D = \emptyset$ implies that there is no improved feasible direction, and thus $\bar{x}$ is a local minimum of f. The condition $F_0 \cap D = \emptyset$ can be taken as the *first-order sufficient condition*.

It was shown in [50, Theorem 12.1] that if the conditions that $x^\star$ is a local minimum of the NLP and the LICQ holds at $x^\star$, then the KKT condition holds. The conditions defined in this theorem are called *first-order necessary conditions*.

So far, we have recalled first-order optimality conditions which demonstrate that how the first-order derivatives of the objective f and the active constraints are related to each each other at solution $x^\star$. In addition, it was investigated in [50, Theorem 12.6] that what roles the second-order derivatives of the objective function and constraint functions play in the optimality conditions.

Lemma 2.1.1 *([50]) Suppose that for some feasible point $x^\star \in \mathbb{R}^n$ there exists a Lagrange multiplier vector $\lambda^\star$ such that the KKT conditions are satisfied. In addition,* *suppose that*

$$\mathbf{w}^\top \nabla^2_{xx} L(x^\star, \lambda^\star, \mu^\star)\mathbf{w} > 0, \quad \forall\ \mathbf{w} \in D(x^\star, \lambda^\star, \mu^\star)\backslash\{\mathbf{0}\}.$$

Then $x^\star$ is a strict local minimum of the NLP. Here, $D(x^\star, \lambda^\star, \mu^\star)$ is a set of directions, defined as follows:

$$D(x^\star, \lambda^\star, \mu^\star) = \left\{ \mathbf{d} \in \mathbb{R}^n \;\middle|\; \begin{array}{l} \mathbf{d}^\top \nabla h_j(x) = 0, j = m+1, \cdots, m+q, \\ \mathbf{d}^\top \nabla g_i(x) = 0, i \in I_0(x) \text{ with } \lambda_i^\star > 0, \\ \mathbf{d}^\top \nabla g_i(x) \le 0, i \in I_0(x) \text{ with } \lambda_i^\star = 0. \end{array} \right\} \quad (2.1.2)$$

The above lemma tells us that second-order sufficient optimality conditions ensure that $x^\star$ is a local minimum of the NLP without the requirement of the constraint qualifications.

Furthermore, a necessary condition involving the second-order derivative was also shown in [50, Theorem 12.5]. That is to say, suppose that $x^\star$ is a local solution of the NLP, LICQ condition is satisfied and in addition, let $(\lambda^\star, \mu^\star)$ be the Lagrange multiplier vector for which the KKT conditions are satisfied, then $\mathbf{w}^\top \nabla^2_{xx} L(x^\star, \lambda^\star, \mu^\star)\mathbf{w} \ge 0, \;$ for all $\;\mathbf{w} \in D(x^\star, \lambda^\star, \mu^\star)\backslash\{\mathbf{0}\}$.

Originated from the practical implementation and numerical considerations of iterative methods for the NLP, the study of error bounds has grown and proliferated in many interesting areas within mathematical programming, for instance, the researches on the exact penalization and optimality conditions. In what follows, we introduce the definition of the error bound condition and an important lemma which is later used.

Definition 2.1.2 *([51]) Denote the constraint system*

$$\begin{cases} g_i(x) \le 0, & i = 1, \cdots, m; \\ h_j(x) = 0, & j = m+1, \cdots, m+q \end{cases}$$

as the set Ω. This system is said to satisfy a local error bound at $x^\star$, if there exist positive constants $k > 0$ and $\delta > 0$ such that

$$\operatorname{dist}(x \,|\, \Omega) \le k\left(\sum_{i=1}^{m} \|g_i(x)^+\| + \sum_{j=m+1}^{m+q} \|h_j(x)\|\right)$$

holds, for all $x \in x^\star + \delta\mathbb{B}$, where $\mathbb{B}$ is unit closed ball in $\mathbb{R}^n$.

[51, Theorem 3] illustrates the relationship between the error bound condition and the exactness property.

Lemma 2.1.2 *[51] For the NLP, if $f(x)$ is Lipschitz continuous and the error bound condition holds for the constraint system, then there exist a neighborhood N_0 of $x^\star$ and a constant $\tau > 0$ such that*

$$f(x) \geq f(x^\star) - \tau(\sum_{i=1}^{m} \|g_i(x)^+\| + \sum_{j=m+1}^{m+q} \|h_j(x)\|)$$

holds for all $x \in N_0$.

2.2 Several Classes of Penalty Functions

In this subsection, we present the categories of penalty functions in detail and recall some existing theoretical results.

We first introduce a general class of nonsmooth penalty functions.

The conclusion that nonsmooth penalty functions enjoy exactness property has been derived by Zangwill in [76]. In this subsection, we consider the class of nonsmooth penalty functions defined as

$$J_q(x;\sigma) := f(x) + \sigma\|[\max(0, g(x)), h(x)]\|_q$$

where $1 \leq q \leq \infty$. In particular, we have

$$J_q(x;\sigma) = f(x) + \sigma[\sum_{i=1}^{m}(\max(0, g_i(x)))^q + \sum_{j=m+1}^{m+q} |h_j(x)|^q]^{\frac{1}{q}},$$

for $1 \leq q < \infty$, and

$$J_\infty(x;\sigma) = f(x) + \sigma[\max(0, g_1(x)), \cdots, \max(0, g_m(x)), |h_{m+1}(x)|, \cdots, |h_{m+q}(x)|].$$

$J_1(x;\sigma)$ and $J_\infty(x;\sigma)$ are the most frequently considered l_1 and l_∞ exact penalty functions in the literature. Here, the penalty function $J_q(x;\sigma)$ is said to be *exact* at a local minimum $\bar{x}$ of the NLP as shown in [52], if $\bar{x}$ is an unconstrained local minimum of $J_q(x;\sigma)$ for all sufficiently large but finite penalty parameter σ. For short, this property is referred to be *exact penalization*.

Han and Mangasarian presented that, under suitable assumptions on the NLP, the function $J_1(x;\sigma)$ has exactness property in [31, Theorem 4.4].

Lemma 2.2.1 *([31]) Suppose that $x^\star$ is a local solution of the NLP, at which the first-order necessary conditions are satisfied with Lagrange multiplier vector $(\lambda^\star, \mu^\star)$. Then, there exists a threshold penalty parameter $\sigma^\star$ such that $\sigma \in [\sigma^\star, \infty)$, $x^\star$ is a local minimizer of $J_1(x;\sigma)$, where*

$$\sigma^\star = \|[\lambda^\star, \mu^\star]\|_\infty = \max_{1\le i\le m, m+1\le j\le m+q} \{|\lambda_i^\star|, |\mu_j^\star|\}.$$

If in addition, the second-order sufficient conditions of Lemma 2.1.1 hold for all $\sigma \in [\sigma^\star, \infty)$, then $x^\star$ is a strict local minimizer of $J_1(x;\sigma)$.

Exact penalty functions not only play important roles in practical algorithm because of finite termination property, but also perform key roles in the theory of mathematical programming. The fact that calmness implies the existence of an exact penalty parameter was established in Clarke [12], while the converse implication was first established by Burke [6, 7].

As discussed previously, $J_1(x;\sigma)$ and $J_\infty(x;\sigma)$ are not differentiable at some x because of the presence of the the absolute value and max operators functions. Subsequently, let us consider replacing a constrained optimization problem by a continuously differentiable penalty function. The simplest penalty function of this type is the quadratic penalty function as shown in [50], in which the penalty terms are the square of the constraint violations.

The quadratic penalty function for the NLP can be expressed as follows

$$J(x;\sigma) = f(x) + \sigma \sum_{i=1}^{m} (\max(0, g_i(x)))^2 + \sigma \sum_{j=m+1}^{m+q} (h_j(x))^2, \tag{2.2.3}$$

where $\sigma > 0$ the penalty parameter. By driving $\sigma \to \infty$, the constraint violations are penalized with increasing severity. It is natural to consider a sequence of values $\{\sigma_k\}$ with $\sigma_k \to \infty$ as $k \to \infty$, and to seek the approximate minimum x_k of (2.2.3) for each k. Since the penalty terms in (2.2.3) are smooth, unconstrained optimization techniques can be used to search for x_k. [50, Theorem 17.1] illustrates that with the penalty parameter $\sigma_k \to \infty$, every limit point $x^\star$ of the sequence $\{x_k\}$ is a locally optimal solution of the NLP, and moreover, the sequence $\{\sigma_k \phi_\sigma(x_k)\}$ converges to 0, where the function $\phi_\sigma(x)$ is the penalty term defined by

$$\phi_\sigma(x) = \sum_{i=1}^{m} (\max(0, g_i(x)))^2 + \sum_{j=m+1}^{m+q} (h_j(x))^2.$$

Although the quadratic penalty function has good smoothness property, it requires the penalty parameter σ tends to infinity to achieve the optimal solution of the NLP. As well known, the penalty parameter tending to infinite may be encountered with ill-conditioness, which brings great difficulties in the practical calculation. Therefore, it is natural to raise a question whether we could establish a penalty function that not only obtain an exact optimum for finite penalty parameter but also enjoys the smoothness. Based on the combination of Lagrange function and the quadratic penalty function, Hestenes [33] and Powell [59] independently proposed *the augumented Lagrangian penalty functions* for equality constrained problem, also known as multiplier penalty functions as follows:

$$J(x, \mu, \sigma) = f(x) + \sum_{j=m+1}^{m+q} \mu_j h_j(x) + \sigma \sum_{j=m+1}^{m+q} (h_j(x))^2, \tag{2.2.4}$$

which is one such exact penalty function. The reason why it is called the augumented Lagrangian penalty function is that (2.2.4) is the ordinary Lagrangian function augumented by the quadratic penalty term. Alternatively, (2.2.4) is considered as the inclusion of a multipliers term in the quadratic penalty objective function, thus it is also called a multiplier penalty function. In view of the second-order sufficient conditions of unconstrained problems, that is, the gradient value of the objective function at the optimum is zero and the Hessian matrix positive definite. For the constrained problem, according to second-order sufficient conditions, the gradient value of Lagrange function at the minimum is zero, but the positive definiteness of the associated Hessian matrix is defined on the critical cone at the minimums. Naturally, in order to make the Hessian matrix of Lagrange function positive definite at the minimums, the critical cone should be taken into consideration. The augumented Lagrangian functions that enjoy exactness properties have been investigated in [2, Theorem 9.3.3] for equality constraint problems.

Chapter 3

A New Class of Exact Penalty Functions and Penalty Algorithms

3.1 Introduction

A typical approach to solve nonlinear programming problems is to augment objective function or the corresponding Lagrangian function using penalty or barrier terms to take account of the constraints. The resulting merit functions are optimized by utilizing either standard unconstrained (or bound constrained) optimization software or sequential quadratic programming (SQP) techniques, etc. Independently of the technique used, the merit function always depends on a small parameter ε (or a large parameter $\rho = \varepsilon^{-1}$); for example, minimizers of the merit function converge to the set of minimizers of the primal problem, provided that ε approaches to 0. In particular, in some SQP approaches, one utilizes exact penalty functions to produce exact optimizers when ε is sufficiently small. Nevertheless, it should be mentioned that these exact penalty functions have a disadvantage that the evaluation of the merit function either needs Jacobian (see [24, 32, 44, 46, 55]) or is no longer smooth (for instance, see l_1 or l_∞ penalty functions [1, 6, 13, 37, 53, 69, 77]). In addition, for all kinds of penalty functions, there may not exist optimal solutions even if the constrained programming problem has optimal solutions, which may make it difficult or impossible to locate a minimizer.

In this section, we are mainly concerned with the following nonlinear programming problem

$$(P) \qquad \min\ f(x) \\ \text{s.t. } F(x) = 0,\ x \in [u,v],$$

where $[u,v] = \{x \in \mathbb{R}^n | u \le x \le v\}$, $u \in (\{-\infty\} \cup \mathbb{R})^n$, $v \in (\{+\infty\} \cup \mathbb{R})^n$ and $\mathrm{int}[u,v] \neq \emptyset$. The functions $f : D \to \mathbb{R}$ and $F : D \to \mathbb{R}^m$ are continuously differentiable on an open set D satisfying $[u,v] \subseteq D$. We always assume that the feasible region is nonempty and that the function f is bounded below on D; because otherwise f can be replaced by $e^{f(\cdot)}$.

Let $\omega \in \mathbb{R}^m$ be fixed. The problem (P) can be written equivalently as

$$\begin{aligned} &\min\ f(x) \\ &\text{s.t.}\quad F(x) = \varepsilon\omega, \\ &\qquad\ x \in [u,v], \quad \varepsilon = 0. \end{aligned}$$

For this problem, a new exact penalty function was proposed in [38]

$$f_\sigma(x,\varepsilon) = \begin{cases} f(x), & \text{if } \varepsilon = \triangle(x,\varepsilon) = 0; \\ f(x) + \frac{1}{2\varepsilon}\frac{\triangle(x,\varepsilon)}{1-q\triangle(x,\varepsilon)} + \sigma\beta(\varepsilon), & \text{if } \varepsilon > 0, \triangle(x,\varepsilon) < q^{-1}; \\ +\infty, & \text{otherwise,} \end{cases} \tag{3.1.1}$$

where $q > 0$ is a given positive constant, $\sigma > 0$ is a penalty parameter, $\beta : [0,\overline{\varepsilon}] \to [0,\infty)$ is continuous on $[0,\overline{\varepsilon}]$ and continuously differentiable on $(0,\overline{\varepsilon}]$ with $\beta(0) = 0$ (here, a positive constant $\overline{\varepsilon} > 0$ is given in advance). The term $\Delta(x,\varepsilon)$ measures the violation of the constraints, i.e.,

$$\Delta(x,\varepsilon) = \|F(x) - \varepsilon\omega\|^2 = \sum_{j=1}^{m}(F_j(x) - \varepsilon\omega_j)^2.$$

The corresponding penalty problem is

$$(P_\sigma) \qquad \min\ f_\sigma(x,\varepsilon) \\ \text{s.t. } (x,\varepsilon) \in [u,v] \times [0,\overline{\varepsilon}].$$

Let $D_q = \{(x,\varepsilon) \in D \times (0,\overline{\varepsilon}] | \Delta(x,\varepsilon) < q^{-1}\}$. Then f_σ is continuously differentiable on D_q. The main contributions in [38] are to explore the exact penalty property of (P_σ) for the smooth and nonsmooth problem (P), respectively. For the smooth problem (P), sufficient conditions are developed for exact penalty property of (P_σ) or equivalently,

$F(\cdot)$ is level-bounded and every point in the level set satisfies Mangasarian-Fromovitz constraint qualification as in [38, Theorem 2.1]. For the nonsmooth case of (P), through the standard smoothing approximation techniques, sufficient conditions are derived for the local exactness property of the penalty problem (P_σ)(see [38, Theorem 5.3]).

The reason why the new penalty function has significant differences with the classical simple exact penalty functions is that f_σ has good smoothness property which is not shared by classical simple and exact penalty functions (see [2, 31]). Furthermore, by the definition of classical penalty functions ([2]), the values of the penalty term are zero on the feasible set of (P) and positive outside the feasible set, which is not the case for the penalty term of f_σ.

In spite of significant differences between the new penalty function and the classical simple and exact penalty functions, interestingly, the equivalence is shown in the sense of the exactness property in this section.

In this work, we restrict our focus primarily on the case of (P) being smooth. It can be further extended to the nonsmooth case by using the standard smoothing approximation techniques. Our main results are as follows.

We notice that the norm function term $\frac{\Delta(x,\varepsilon)}{1-q\Delta(x,\varepsilon)}$ in the penalty function (3.1.1) proposed in [38] plays a role as a barrier term, which means computing the interior points of D_q is an essential component in the practical applications. This motivates us to extend the norm function term of (3.1.1) to a class of convex functions so that the corresponding penalty functions constructed by this class of convex functions present a unified framework for some barrier-type and exterior-type penalty functions, in which the latter are smooth on $[u, v] \times (0, \bar{\varepsilon}]$ and have larger smooth area than the former. The extended penalty function class provides several alternatives for designing penalty function algorithms. In the remainder of the present work, we make further researches on this extended penalty function class.

Based on the extended penalty function class, we establish a class of corresponding penalty problems for (P). Sufficient conditions for the exact penalty property are primarily obtained for the penalty problem, without requirements that $F(\cdot)$ is level-bounded or every point in the level set satisfies the classical Mangasarian-Fromovitz constraint qualification. We show that the local optimal solution must take the form of

(x, ε) with $\varepsilon = 0$ for all sufficiently large penalty parameters, where x is a local optimal solution of the primal problem (P) (see Theorem 3.3.1). Here, it is merely required that a local optimal solution sequence with finite optimal values satisfies the extended Mangasarian-Fromovitz constraint qualification and the gradient $\nabla f(\cdot)$ is bounded on the level set. As a corollary, [38, Theorem 2.1] can be derived.

As two new results of this section, necessary conditions for exact penalty property are obtained (see Theorem 3.3.2). Alternatively, we derive that these necessary conditions are also necessary and sufficient conditions for the local exactness property (see Theorem 3.4.1), followed by a series of meaningful corollaries can be derived from Theorem 3.4.1, including [38, Theorem 5.3].

In particular, using Theorem 3.4.1, it is recognized explicitly that this new class of simple exact penalty functions is equivalent to the classical simple exact penalty functions in the sense of exactness property. Therefore, these conclusions demonstrate that this class of penalty functions not only have good exactness property as the classical simple penalty functions, but also possess the smoothness property, which is not shared by the classical simple exact penalty functions.

At the beginning of this section, we have mentioned that some exact penalty functions may have no optimal solutions even if the optimal solutions of the primal problem exist. For this case, it is impossible to obtain optimal solutions through the penalty function algorithm. There is also no exception for general smooth penalty functions with lower bound. Here, we take the penalty function proposed in [30] for example, which is smooth and lower-bounded. Although the proposed penalty function algorithm in [30] has good convergence properties, there exists an example to show that, the penalty function has no optimal solutions for any appropriate penalty parameter even if optimal solutions of the primal problem exist. In this environment, the penalty function algorithm proposed in [30] and its revised version included are naturally infeasible. In addition, the sequences generated by several penalty function algorithms with their own specialities only converge to FJ points or KKT points.

Motivated by this observation, in order to avoid the above mentioned cases, we present a class of revised penalty function algorithms based on this class of extended penalty functions proposed in this section. The main feature of the algorithm is that, if the optimal solution of the penalty problem does not exist at every iteration, then we turn to solve $\delta-$optimal solution of the penalty problem. Since this class of penalty functions have lower bound, the $\delta-$optimal solution always exists. Thus, the revised penalty function algorithms are always feasible. Moreover, utilizing the exact penalty property and structural features of this class of penalty functions, we clarify that under certain conditions, the proposed algorithm terminates at the optimal solution of the primal problem after finitely many iterations; while without these conditions, a perturbation theorem for this algorithm can be derived. As a corollary, the global convergence property is presented−namely, every accumulation point of the sequence generated by the algorithm is an optimal solution of the primal problem. In addition, some significant conclusions are also developed based on the perturbation theorem.

The organization of this chapter is as follows. In Section 3.2, we extend the penalty function (3.1.1) to a class of penalty functions, establish corresponding penalty problems and introduce a notion of extended Mangasarian-Fromovitz constraint qualification. The necessary and sufficient conditions for exact penalty property are developed in Section 3.3. In Section 3.4, necessary and sufficient conditions for the local exactness property are established. We also characterize the equivalence between this new class of penalty functions and the classical simple exact penalty function in the sense of exactness property. In Section 3.5, a feasible and revised penalty function algorithm is presented. The finite termination property and global convergence property of the proposed algorithm are analyzed. Finally, in Section 3.6, promising numerical results are reported.

3.2 A Class of Exact Penalty Functions

In this section, we extend the term $\frac{\Delta(x,\varepsilon)}{1-q\Delta(x,\varepsilon)}$ in (3.1.1) to a class of convex functions. The corresponding penalty functions class present a unified framework for some barrier-type and exterior-type penalty functions.

Given $a \in (0, +\infty]$, let a function $\phi : [0, a) \to [0, +\infty)$ satisfy
(i_1) ϕ is convex and continuously differentiable on $[0, a)$ with $\phi(0) = 0$.
(i_2) $\phi'(t) > 0$ for all $t \in [0, a)$.

Many functions satisfy the properties $(i_1), (i_2)$, for example

$$\begin{array}{ll}
\phi_1(t) = \frac{t}{(1-qt)^\alpha} & (a = q^{-1}, \alpha \geq 1); \\
\phi_2(t) = \tan(t) & (a = \frac{\pi}{2}); \\
\phi_3(t) = -\log(1 - t^\alpha) & (a = 1, \alpha \geq 1); \\
\phi_4(t) = t & (a = +\infty); \\
\phi_5(t) = e^t - 1 & (a = +\infty); \\
\phi_6(t) = \frac{1}{2}(\sqrt{t^2+4} + t) - 1 & (a = +\infty).
\end{array}$$

Utilizing $\phi(\cdot)$, a penalty function, defined on $D \times [0, \bar{\varepsilon}]$, is given by

$$\widetilde{f}_\sigma(x, \varepsilon) = \begin{cases} f(x), & \text{if } \varepsilon = \triangle(x, \varepsilon) = 0, \\ f(x) + \frac{1}{2\varepsilon}\phi(\Delta(x, \varepsilon)) + \sigma\beta(\varepsilon), & \text{if } \varepsilon > 0, \triangle(x, \varepsilon) < a, \\ +\infty, & \text{otherwise.} \end{cases} \tag{3.2.2}$$

It is easy to see that $\widetilde{f}_\sigma$ is continuously differentiable on $D_a = \{(x, \varepsilon) \in D \times (0, \bar{\varepsilon}] | \Delta(x, \varepsilon) < a\}$. The barrier-type penalty functions correspond to the case of $a < +\infty$, and the exterior-type one correspond to the case of $a = +\infty$.

In the remainder of this section, we mainly consider the following penalty problem

$$(\tilde{P}_\sigma) \qquad \min \ \widetilde{f}_\sigma(x, \varepsilon) \quad \text{s.t. } (x, \varepsilon) \in [u, v] \times [0, \bar{\varepsilon}].$$

Consider a constraint system

$$\begin{cases} F(z) = 0, \\ g(z) \leq 0, \end{cases} \tag{3.2.3}$$

where $g : D \to \mathbb{R}^l$ is continuously differentiable, and F is defined as before. Denote by $\nabla F(\cdot)$ the Jacobian of F.

Definition 3.2.1 *For constraint system (3.2.3), we say that the Mangasarian-Fromovitz constraint qualification holds at $z^\star$, if*

(1) *rank$(\nabla F(z^\star)) = m$;*

(2) *there exists some $p \in \mathbb{R}^n$ such that $\nabla F(z^\star)^\top p = 0$ and $\nabla g_j(z^\star)^\top p < 0$ for all $j \in J(z^\star)$, where $J(z^\star) = \{j \in J | g_j(z^\star) = 0\}$ and $J = \{1, 2, \cdots, l\}$.*

Now we introduce an extension of the standard Mangasarian-Fromovitz constraint qualification from a single point to a sequence. More precisely, given $K \subseteq \{1, 2, \cdots\}$ and $\{z_k\}_{k \in K}$, denote

$$J^+(K) = \{j \in J | \limsup_{k \in K, k \to \infty} g_j(z_k) \geq 0\},$$
$$J^-(K) = \{j \in J | \limsup_{k \in K, k \to \infty} g_j(z_k) < 0\}.$$

Definition 3.2.2 *For constraint system (3.2.3), we say that the extended Mangasarian-Fromovitz constraint qualification holds for $\{z_k\}_{k \in K}$, if there exist a matrix $\nabla F^\star$ and an infinite subset $K_0 \subseteq K$ such that*

(1) $\lim_{k \in K_0, k \to \infty} \nabla F(z_k) = \nabla F^\star$ *and rank*$(\nabla F^\star) = m$*;*

(2) *there exists some $p \in \mathbb{R}^n$ such that $(\nabla F^\star)^\top p = 0$ and $\limsup_{k \in K_0, k \to \infty} \nabla g_j(z_k)^\top p < 0$ for all $j \in J^+(K_0)$.*

The reason why the extended Mangasarian-Fromovitz constraint qualification is introduced is that the sequence generated by the unconstrained minimization algorithm may not converge to a problem solution, or may be unbounded. Note that in Definition 3.2.2 the sequence z_k is not required to satisfy $g(z_k) \leq 0$. Thus, this extended version of Mangasarian-Fromovitz constraint qualification is applicable to sequences generated by many infeasible algorithms (including penalty function methods), even if the sequence is unbounded (see Example 3.2.1 below). By definition, we have

Proposition 3.2.1 *Let $\{z_k\}_{k \in K} \subset \mathbb{R}^n$. If the standard Mangasarian-Fromovitz constraint qualification holds for some accumulation point, then the extended Mangasarian-Fromovitz constraint qualification holds for $\{z_k\}_{k \in K}$.*

The following example shows that the unbounded and infeasible sequence satisfies the extended Mangasarian-Fromovitz constraint qualification.

Example 3.2.1 *Consider the constraint system*

$$\begin{cases} F(z) &= z_1 + z_2 + z_3^2 = 0, \\ g_1(z) &= z_1 - z_2 \le 0, \\ g_2(z) &= z_1 - z_2 - z_3^2 \le 0. \end{cases}$$

It is easy to see that the extended Mangasarian-Fromovitz constraint qualification holds for the unbounded sequence $\{z_k\} = \{(k + \frac{1}{k}, k, 1 + \frac{1}{k})^\top\}$.

The final part of this section is devoted to the discussion of the extended Mangasarian-Fromovitz constraint qualification in more detail. By virtue of Definition 3.2.1, Definition 3.2.2 and Example 3.2.1, we generalize the Mangasarian-Fromovitz constraint qualification for a single point $z^\star$ satisfying $g(z^\star) \le 0$ to the extended Mangasarian-Fromovitz constraint qualification for an infinite sequence $\{z_k\}_{k \in K}$, which may be unbounded and not necessarily satisfy $g(z_k) \le 0$, (namely, z_k may be infeasible). Therefore, the extended Mangasarian-Fromovitz constraint qualification can be used to weaken some assumptions for the exactness property in some existing literature. To a great extent, it mainly refers to remove the level-bounded assumption. For instance, as stated in [38, Theorem 2.1], the exact penalty property of the penalty function $f_\sigma(z, \varepsilon)$ is developed under conditions that the level set D' is bounded and every point $z \in D'$ satisfies Mangasarian-Fromovitz constraint qualification. The above mentioned assumptions in [38] imply that as $\{\sigma_k\}_{k \in K} \to +\infty$, the corresponding sequence $\{z_k\}_{k \in K}$ is bounded and every accumulation point satisfies Mangasarian-Fromovitz constraint qualification, where (z_k, ε_k) is a local optimal solution of the penalty problem $(\tilde{P}_\sigma)$ and the value of $\tilde{f}_\sigma(z_k, \varepsilon_k)$ is finite. Additionally, based on Proposition 3.2.1, the extended Mangasarian-Fromovitz constraint qualification thus holds for the infinite sequence $\{z_k\}_{k \in K}$. Therefore, the penalty function $f_\sigma(z, \varepsilon)$ is also proven to be exact (see Theorem 3.3.1) merely with requirements that the sequence $\{z_k\}_{k \in K}$ satisfies the extended Mangasarian-Fromovitz constraint qualification and the gradient $\nabla f(\cdot)$ is bounded on the level set instead of the assumptions stated earlier in [38] (see assumptions (A_1) and (A_2)). It is demonstrated that [38, Theorem 2.1] is just a special case in our work.

3.3 Exact Penalty Property

This section deals mainly with the "exact" property of the penalty functions defined in (3.2.2). In particular, [38, Theorem 2.1] is generalized by using the extended Mangasarian-Fromovitz constraint qualification, instead of the standard Mangasarian-Fromovitz constraint qualification. Define the level set

$$D_F = \{x \in [u, v] | \|F(x)\| \leq \sqrt{a} + \bar{\varepsilon}\|\omega\|\},$$

where $a \in (0, +\infty]$. Clearly, $D_F = [u, v]$ in the case of $a = +\infty$. Before proceeding, we need the following assumption.

Assumption (A):

(A_1) $\nabla f(\cdot)$ is bounded on D_F;

(A_2) For constraint system of (P), when $\sigma_k \to +\infty (k \to \infty)$, the extended Mangasarian-Fromovitz constraint qualification holds for the sequence $\{x_k\}$, where (x_k, ε_k) is the optimal solution of $(\tilde{P}_{\sigma_k})$ with finite value.

Lemma 3.3.1 *Suppose that there exists* $\beta_1 > 0$ *such that* $\beta'(\varepsilon) \geq \beta_1$ *for all* $\varepsilon \in (0, \bar{\varepsilon}]$. *If* (x, ε) *is a KKT point of* $(\tilde{P}_\sigma)$ *with* $\varepsilon > 0$, *then*

$$2\beta_1 \phi'(0) \sigma \varepsilon^2 \frac{1}{\phi'(\Delta)^2} \leq \|F(x)\|^2,$$

where we use Δ *to denote* $\Delta(x, \varepsilon)$ *for simplification.*

Proof. If (x, ε) is a KKT point of $(\tilde{P}_\sigma)$ with $\varepsilon > 0$, then by the construction of $\widetilde{f}_\sigma$, there exist $\lambda,\ \eta \in \mathbb{R}^n_+,\ \lambda_{n+1} \geq 0$, and $\eta_{n+1} \geq 0$ such that

$$\nabla f(x) + \frac{1}{\varepsilon}\phi'(\Delta)\nabla F(x)^T(F(x) - \varepsilon\omega) = \lambda - \eta,$$

$$\inf(\lambda_i, x_i - u_i) = \inf(\eta_i, v_i - x_i) = 0, \qquad i = 1, 2, \cdots, n,$$

$$-\frac{1}{2\varepsilon^2}\phi(\Delta) - \frac{1}{\varepsilon}\phi'(\Delta)(F(x) - \varepsilon\omega)^T\omega + \sigma\beta'(\varepsilon) = \lambda_{n+1} - \eta_{n+1}, \tag{3.3.4}$$

$$\lambda_{n+1} = \inf(\eta_{n+1}, \bar{\varepsilon} - \varepsilon) = 0. \tag{3.3.5}$$

It follows from (3.3.4) and (3.3.5) that

$$-\frac{1}{2\varepsilon^2}\phi(\Delta) - \frac{1}{\varepsilon}\phi'(\Delta)(F(x) - \varepsilon\omega)^T\omega + \sigma\beta'(\varepsilon) \leq 0,$$

from which and the fact that $\phi'(\Delta) > 0$ we have

$$-\frac{1}{\phi\,'(\Delta)}\phi(\Delta) - 2\varepsilon(F(x) - \varepsilon\omega)^T\omega + 2\varepsilon^2\sigma\beta'(\varepsilon)\frac{1}{\phi\,'(\Delta)} \leq 0. \tag{3.3.6}$$

Rearranging (3.3.6) yields

$$-\frac{1}{\phi'(\Delta)}\phi(\Delta) + \Delta + \varepsilon^2\|\omega\|^2 + 2\varepsilon^2\sigma\beta'(\varepsilon)\frac{1}{\phi'(\Delta)} \leq \|F(x)\|^2. \tag{3.3.7}$$

The convexity of ϕ and the fact $\phi(0) = 0$ ensure

$$-\Delta \leq (\phi(0) - \phi(\Delta))\frac{1}{\phi'(\Delta)} = -\phi(\Delta)\frac{1}{\phi'(\Delta)}. \tag{3.3.8}$$

Combining (3.3.7) and (3.3.8) yields

$$2\varepsilon^2\sigma\beta'(\varepsilon)\frac{1}{\phi'(\Delta)} \leq \|F(x)\|^2.$$

Noticing that $\beta'(\varepsilon) \geq \beta_1$ and the monotonicity of ϕ', the above inequality implies that

$$2\beta_1\phi'(0)\sigma\varepsilon^2\frac{1}{\phi'(\Delta)^2} \leq \|F(x)\|^2.$$

■

Lemma 3.3.2 *Suppose that there exists $\beta_1 > 0$ such that $\beta'(\varepsilon) \geq \beta_1$ for all $\varepsilon \in (0, \overline{\varepsilon}]$. If $(x^\star, \varepsilon^\star)$ is a local optimal solution of $(\tilde{P}_\sigma)$ with finite optimal value, then $x^\star$ is a local optimal solution of (P) if and only if $\varepsilon^\star = 0$.*

Proof. Since constraint functions of $(\tilde{P}_\sigma)$ are all linear, then $(x^\star, \varepsilon^\star)$ is a KKT point of $(\tilde{P}_\sigma)$. If $x^\star$ is a local optimal solution of (P), then $F(x^\star) = 0$. According to Lemma 3.3.1, we must have $\varepsilon^\star = 0$. Conversely, if $\varepsilon^\star = 0$, taking account of the finiteness of $\widetilde{f}_\sigma(x^\star, 0)$ and the construction of $\widetilde{f}_\sigma$, we have $F(x^\star) = 0$, and hence $x^\star$ is a local optimal solution of (P). ■

Lemma 3.3.3 *Suppose that there exists $\beta_1 > 0$ such that $\beta'(\varepsilon) \geq \beta_1$ for all $\varepsilon \in (0, \overline{\varepsilon}]$. If (x_k, ε_k) is a KKT point of $(\tilde{P}_{\sigma_k})$ with $\varepsilon_k > 0$, then for $\sigma_k \to +\infty (k \to \infty)$, we have*

$$\lim_{k\to\infty} \frac{1}{\varepsilon_k}\phi'(\Delta_k)\sqrt{\Delta_k} = +\infty.$$

where we use Δ to denote $\Delta(x, \varepsilon)$ for simplification.

Proof. Lemma 3.3.1 implies that

$$\lim_{k\to\infty} \frac{1}{\varepsilon_k}\phi'(\Delta_k)\|F(x_k)\| = +\infty.$$

Note that

$$\frac{1}{\varepsilon_k}\phi'(\Delta_k)\|F(x_k)\| \le \phi'(\Delta_k)(\frac{1}{\varepsilon_k}\sqrt{\Delta_k} + \|\omega\|).$$

Therefore,

$$\lim_{k\to\infty} \phi'(\Delta_k)(\frac{1}{\varepsilon_k}\sqrt{\Delta_k} + \|\omega\|) = +\infty,$$

which, together with the monotonicity of ϕ', yields

$$\lim_{k\to\infty} \frac{1}{\varepsilon_k}\phi'(\Delta_k)\sqrt{\Delta_k} = +\infty.$$

■

Theorem 3.3.1 *Suppose that Assumptions (A_1) and (A_2) hold, and that there exists $\beta_1 > 0$ such that $\beta'(\varepsilon) \ge \beta_1$ for all $\varepsilon \in (0, \bar{\varepsilon}]$. When $\sigma > 0$ is sufficiently large, if $(x^\star, \varepsilon^\star)$ is a local optimal solution of $(\tilde{P}_\sigma)$ with finite optimal value, then $\varepsilon^\star = 0$. Furthermore, $x^\star$ is a local optimal solution of (P).*

Proof. We first show that $\varepsilon^\star = 0$. Suppose on the contrary that there exist $\sigma_k \to +\infty$ as $k \to \infty$ and a sequence of local optimal solutions (x_k, ε_k) of $(\tilde{P}_{\sigma_k})$ with finite $\widetilde{f}_{\sigma_k}(x_k, \varepsilon_k)$ and $\varepsilon_k > 0$. Since all constraint functions are linear, then (x_k, ε_k) is a KKT point of $(\tilde{P}_{\sigma_k})$, that is, there exist $\lambda^k, \eta^k \in \mathbb{R}^n_+$ such that

$$\nabla f(x_k) + \frac{1}{\varepsilon_k}\phi'(\Delta_k)\nabla F(x_k)^T(F(x_k) - \varepsilon_k\omega) = \lambda^k - \eta^k, \tag{3.3.9}$$

$$\inf(\lambda_i^k, x_i^k - u_i) = \inf(\eta_i^k, v_i - x_i^k) = 0, \quad i = 1, 2, \cdots, n. \tag{3.3.10}$$

Since the index set $\{1, \cdots, n\}$ is finite, then there exists an infinite subset $K \subseteq \{1, 2, \cdots\}$ such that for $k \in K$,

$$x_i^k = u_i, \qquad i \in I_1, \tag{3.3.11}$$

$$x_i^k = v_i, \qquad i \in I_2, \tag{3.3.12}$$

$$u_i < x_i^k < v_i, \qquad i \in I_3, \tag{3.3.13}$$

where

$$I_1 \cup I_2 \cup I_3 = \{1, 2, \cdots, n\}, \quad I_i \cap I_j = \emptyset, \quad if \quad i \ne j.$$

Invoking Assumption (A_2), there exist an infinite subset $K_0 \subseteq K$ and $p \in \mathbb{R}^n$ such that

$$\lim_{k \in K_0, k \to \infty} \nabla F(x_k) = \nabla F^\star, \quad \text{rank}(\nabla F^\star) = m, \quad \nabla F^\star p = 0, \tag{3.3.14}$$

$$p_i \begin{cases} > 0, & i \in I_1, \\ < 0, & i \in I_2. \end{cases} \tag{3.3.15}$$

Putting (3.3.9)-(3.3.13) together yields

$$\frac{\partial f(x_k)}{\partial x_i} + \frac{1}{\varepsilon_k} \phi'(\triangle_k) \big(\nabla F(x_k)^T (F(x_k) - \varepsilon_k \omega)\big)_i \begin{cases} \geq 0, & i \in I_1, \\ \leq 0, & i \in I_2, \\ = 0, & i \in I_3, \end{cases} \tag{3.3.16}$$

where $\frac{\partial f(x)}{\partial x_i}$ denotes the partial derivative of f with respect to x_i. Let

$$h_k = \frac{1}{\varepsilon_k} \phi'(\Delta_k)(F(x_k) - \varepsilon_k \omega), \quad q_k = \frac{h_k}{\|h_k\|}.$$

Lemma 3.3.3 implies

$$\lim_{k \in K_0, k \to \infty} \|h_k\| = +\infty. \tag{3.3.17}$$

Since $\|q_k\| = 1$ is bounded, we can assume, without loss of generality, that

$$\lim_{k \in K_0, k \to \infty} q_k = \tilde{q} \neq 0. \tag{3.3.18}$$

It then follows from (3.3.16) that

$$\frac{\partial f(x_k)}{\partial x_i} \frac{1}{\|h_k\|} + (\nabla F(x_k)^T q_k)_i \begin{cases} \geq 0, & i \in I_1, \\ \leq 0, & i \in I_2, \\ = 0, & i \in I_3. \end{cases}$$

From Assumption (A_1), (3.3.14), (3.3.17), and (3.3.18), taking limits in the above formula yields

$$\big((\nabla F^\star)^T \tilde{q}\big)_i \begin{cases} \geq & 0, & i \in I_1, \\ \leq & 0, & i \in I_2, \\ = & 0, & i \in I_3. \end{cases} \tag{3.3.19}$$

Combining this with (3.3.14) implies

$$0 = p^T (\nabla F^\star)^T \tilde{q} = \sum_{i \in I_1} p_i \big((\nabla F^\star)^T \tilde{q}\big)_i + \sum_{i \in I_2} p_i \big((\nabla F^\star)^T \tilde{q}\big)_i,$$

which, together with (3.3.15) and (3.3.19) yields $(\nabla F^\star)^T \tilde{q} = 0$, and hence $\tilde{q} = 0$ since $\nabla F^\star$ has full row rank by (3.3.14). This leads to a contradiction to $\tilde{q} \neq 0$ in (3.3.18). Therefore, the desired result follows from Lemma 3.3.2. ■

Corollary 3.3.1 *Suppose that*

(1) *the set D_F is bounded,*

(2) *the standard Mangasarian-Fromovitz constraint qualification holds at each point of D_F,*

(3) *there exists $\beta_1 > 0$ such that $\beta'(\varepsilon) \geq \beta_1$ for all $\varepsilon \in (0, \bar{\varepsilon}]$.*

Then, whenever $\sigma > 0$ is large enough, every local optimal solution $(x^\star, \varepsilon^\star)$ of $(\tilde{P}_\sigma)$ with finite value must satisfy $\varepsilon^\star = 0$. Furthermore, $x^\star$ is a local optimal solution of (P).

Proof. The validity of Assumption (A_1) comes from the condition (1), while Assumption (A_2) is due to the conditions (1), (2), and Proposition 3.2.1. ■

In particular, Theorem 2.1 in [38] is obtained by taking $\phi(t) = \frac{t}{1-qt}$ in Corollary 3.3.1. Inspired by [38], a necessary condition for exact penalty property is given below. Toward this end, consider a class of functions $\Phi : [0, a) \to [0, +\infty)$, where $a > 0$, satisfying

(j_1) Φ is continuous and increasing on $[0, a)$ with $\Phi(0) = 0$,

(j_2) there exists $a' \in (0, a)$ such that $\Phi(t) \geq t$ for all $t \in [0, a']$.

Such a class of functions includes several important functions as special cases, for example

$$\begin{aligned} &\Phi_1(t) = t^\alpha, && t \in [0, +\infty),\ \alpha \in (0, 1]; \\ &\Phi_2(t) = -\log(1 - t^\alpha), && t \in [0, 1),\ \alpha \in (0, 1]; \\ &\Phi_3(t) = e^t - 1, && t \in [0, +\infty). \end{aligned}$$

Assumption (B) Let $x^\star$ be a feasible point of (P). There exist $\gamma > 0$ and a neighborhood $N(x^\star)$ such that

$$f(x) - f(x^\star) + \gamma\Phi(\|F(x)\|) \geq 0, \quad \forall x \in N(x^\star) \cap [u, v], \tag{3.3.20}$$

where Φ satisfies the conditions (j_1) and (j_2).

Theorem 3.3.2 *Suppose that $\beta(\varepsilon) = \Phi(\varepsilon)$ for $\varepsilon > 0$ sufficiently small. If $(x^\star, 0)$ is a local optimal solution of $(\tilde{P}_{\sigma_0})$ with finite value for some $\sigma_0 > 0$ (i.e., $x^\star$ is a local optimal solution of (P)), then Assumption (B) holds at $x^\star$ for Φ with $\gamma \geq \sigma_0 + \phi'(0)(1 + \|\omega\|)^2$.*

Proof. Let $\gamma = \sigma_0 + \phi'(0)(1 + \|\omega\|)^2$. Suppose on the contrary that there exists a sequence $x_k \in [u, v]$ converging to $x^\star$ such that

$$f(x_k) - f(x^\star) + \gamma\Phi(\|F(x_k)\|) < 0. \tag{3.3.21}$$

Since $(x^\star, 0)$ is a local optimal solution of $(\tilde{P}_{\sigma_0})$ and the corresponding optimal value is finite, it follows from the construction of $\widetilde{f}_\sigma$ that $x^\star$ is feasible (cf.(3.2.2)). Hence $x^\star$ is a local optimal solution of (P). This, together with the continuity of F, implies that

$$\|F(x_k)\| > 0, \quad \lim_{k\to\infty} \|F(x_k)\| = 0.$$

Let $\varepsilon_k = \|F(x_k)\|$, i.e., $\varepsilon_k > 0$ and $\varepsilon_k \to 0$ as $k \to \infty$. Notice that

$$\Delta_k = \|F(x_k) - \varepsilon_k\omega\|^2 \leq \varepsilon_k^2(1 + \|\omega\|)^2. \tag{3.3.22}$$

Thus,

$$\lim_{k\to\infty} \Delta_k \leq \lim_{k\to\infty} \varepsilon_k^2(1 + \|\omega\|)^2 = 0.$$

So,

$$\phi\,'(\Delta_k) \leq 2\phi'(0), \tag{3.3.23}$$

whenever k is sufficiently large, since ϕ' is continuous and $\phi'(0) > 0$. Note that $(x^\star, 0)$ is a local optimal solution. Therefore, for k large enough we have

$$\begin{array}{rcll}
0 & \leq & f(x_k) - f(x^\star) + \dfrac{1}{2\varepsilon_k}\phi(\Delta_k) + \sigma_0\beta(\varepsilon_k) & \\
 & = & f(x_k) - f(x^\star) + \dfrac{1}{2\varepsilon_k}\phi(\Delta_k) + \sigma_0\Phi(\varepsilon_k) & \textit{by Assumption } \beta(\varepsilon) = \Phi(\varepsilon) \\
 & < & \dfrac{1}{2\varepsilon_k}\phi(\Delta_k) + \sigma_0\Phi(\varepsilon_k) - \gamma\Phi(\|F(x_k)\|) & \textit{by } (3.3.21) \\
 & \leq & \dfrac{1}{2\varepsilon_k}\phi'(\Delta_k)\Delta_k + \sigma_0\Phi(\varepsilon_k) - \gamma\Phi(\varepsilon_k), & \textit{by the convexity of } \phi \textit{ and } \phi(0) = 0 \\
 & \leq & \varepsilon_k\phi'(0)(1 + \|\omega\|)^2 + (\sigma_0 - \gamma)\Phi(\varepsilon_k) & \textit{by } (3.3.22) \textit{ and } (3.3.23) \\
 & \leq & \Phi(\varepsilon_k)[\phi'(0)(1 + \|\omega\|)^2 + \sigma_0 - \gamma], & \textit{by condition } (j_2) \\
 & \leq & 0, & \textit{since } \gamma \geq \sigma_0 + \phi\,'(0)(1 + \|\omega\|)^2
\end{array}$$

which leads to a contradiction. ■

3.4 Local Exactness Property

In this section, we shall show that, if $x^\star$ is a local optimal solution of (P), then Assumption (B) introduced in Section 3.3 is a necessary and sufficient condition for $(x^\star, 0)$ to be a local optimal solution of $(\tilde{P}_\sigma)$. Based on Theorem 3.3.2, we further characterize the equivalence between this new class of exact penalty functions and the classical simple and exact penalty functions in the sense of "exact penalty property". Nevertheless, it is well known that the classical simple exact penalty function lacks smoothness. Therefore, our results clarify that this class of penalty functions not only have exactness property as the classical simple penalty function, but also possess the smoothness property.

Theorem 3.4.1 *Let $x^\star$ be a local optimal solution of (P). The following statements hold:*

(1) *If Assumption (B) holds at $x^\star$ for Φ, and $\beta(\varepsilon) \geq \Phi(\sqrt{\varepsilon})$ for $\varepsilon > 0$ sufficiently small, then $(x^\star, 0)$ is a local optimal solution of $(\tilde{P}_\sigma)$ for all $\sigma \geq \gamma$;*

(2) *Let $\beta(\varepsilon) = \Phi(\varepsilon)$ for $\varepsilon > 0$ sufficiently small. If there exists $\sigma_0 > 0$ such that $(x^\star, 0)$ is a local optimal solution of $(\tilde{P}_\sigma)$ for all $\sigma \geq \sigma_0$, then Assumption (B) holds at $x^\star$ for Φ with $\gamma \geq \sigma_0 + \phi'(0)(1 + \|\omega\|)^2$.*

Proof. We resort to show the validity of part (1), since part (2) follows from Theorem 3.3.2. Suppose on the contrary that for some $\sigma \geq \gamma$, there exist $(x_k, \varepsilon_k) \in [u, v] \times (0, \overline{\varepsilon}]$, $x_k \to x^\star$, and $\varepsilon_k \to 0$ as $k \to \infty$ such that

$$f(x^\star) = \widetilde{f}_\sigma(x^\star, 0) > \widetilde{f}_\sigma(x_k, \varepsilon_k),$$

that is,

$$\begin{aligned} 0 &> f(x_k) - f(x^\star) + \frac{1}{2\varepsilon_k}\phi(\Delta_k) + \sigma\beta(\varepsilon_k) \\ &\geq f(x_k) - f(x^\star) + \frac{1}{2\varepsilon_k}\phi'(0)\Delta_k + \sigma\beta(\varepsilon_k), \end{aligned} \tag{3.4.24}$$

where we have used the gradient inequality of convex functions, i.e., $\phi(\Delta_k) \geq \phi(0) + \phi'(0)\Delta_k = \phi'(0)\Delta_k$. Taking limits on both sides of (3.4.24) yields $\lim_{k\to\infty} \frac{1}{\varepsilon_k}\Delta_k = 0$, which means that

$$\sqrt{\Delta_k} \leq \frac{1}{2}\sqrt{\varepsilon_k}, \ \ \sqrt{\varepsilon_k} \leq \frac{1}{2\|\omega\|+1}$$

whenever k is sufficiently large. So,

$$\begin{aligned} \|F(x_k)\| &\leq \sqrt{\Delta_k} + \varepsilon_k\|\omega\| \\ &\leq (\frac{1}{2} + \sqrt{\varepsilon_k}\|\omega\|)\sqrt{\varepsilon_k} \\ &\leq \sqrt{\varepsilon_k}. \end{aligned} \tag{3.4.25}$$

Since $\varepsilon_k \to 0$, then by hypothesis, for k large enough,

$$\beta(\varepsilon_k) \geq \Phi(\sqrt{\varepsilon_k}). \tag{3.4.26}$$

Therefore, we conclude that for k sufficiently large,

$$\begin{aligned} 0 &> f(x_k) - f(x^\star) + \frac{1}{2\varepsilon_k}\phi'(0)\Delta_k + \sigma\beta(\varepsilon_k) \\ &\geq f(x_k) - f(x^\star) + \sigma\beta(\varepsilon_k) && \textit{by the nonnegativity of } \phi' \\ &\geq \sigma\beta(\varepsilon_k) - \gamma\Phi(\|F(x_k)\|) && \textit{by Assumption } (B) \\ &\geq \sigma\Phi(\sqrt{\varepsilon_k}) - \gamma\Phi(\|F(x_k)\|) && \textit{by } (3.4.26) \\ &\geq \Phi(\sqrt{\varepsilon_k})(\sigma - \gamma) && \textit{by the monotonicity of } \Phi \textit{ and } (3.4.25) \\ &\geq 0, && \textit{since } \sigma \geq \gamma \end{aligned}$$

which leads to a contradiction. ■

For the primal problem (P), the mapping H takes the following special form:

Define

$$H(x) = \begin{pmatrix} F(x) \\ x_{I^\star} - x^\star_{I^\star} \end{pmatrix}, \ p = m + |I^\star| \leq n,$$

where $|I^\star|$ signifies the number of elements in $I^\star = I_1^\star \cup I_2^\star$, $I_1^\star$ and $I_2^\star$ are, respectively, the active index sets of $u \leq x^\star$ and $x^\star \leq v$, i.e.,

$$I_1^\star = \{i | u_i = x_i^\star\} \ \text{ and } \ I_2^\star = \{i | v_i = x_i^\star\}.$$

The definition of regular zero as described in [38, Definition 4.1], together with Theorem 3.4.1, allows us to derive the following corollary.

Corollary 3.4.1 *Let $x^\star$ be a local optimal solution of (P) and $\beta(\varepsilon) \geq \sqrt{\varepsilon}$ for all ε sufficiently small. If $x^\star$ is a regular zero of H, then there exists $\gamma > 0$ such that $(x^\star, 0)$ is a local optimal solution of $(\tilde{P}_\sigma)$ for all $\sigma \geq \gamma$.*

Proof. Since $x^\star$ is a regular zero of H, it then follows from [38, Lemmas 5.1 and 5.2] that Assumption (B) holds at $x^\star$ for $\Phi(t) = t$. Combining this and Theorem 3.4.1 yields the desired result. ■

In particular, [38, Theorem 5.3] is obtained by taking $\phi(t) = \frac{t}{1-qt}$ in $\widetilde{f}_\sigma$.

Corollary 3.4.2 *Let $x^\star$ be a strict local optimal solution of (P) and $\beta(\varepsilon) \geq \sqrt{\varepsilon}$ for $\varepsilon > 0$ sufficiently small. If the Mangasarian-Fromovitz constraint qualification holds at $x^\star$, then there exists $\gamma > 0$ such that $(x^\star, 0)$ is a local optimal solution of $(\tilde{P}_\sigma)$ for all $\sigma \geq \gamma$.*

Proof. Since the Mangasarian-Fromovitz constraint qualification holds at $x^\star$, it then follows from [31, Theorem 4.4] that Assumption (B) holds at $x^\star$ for $\Phi(t) = t$. Therefore, the desired result follows from Theorem 3.4.1. ■

Actually, these two sufficient conditions given in Corollaries 3.4.1 and 3.4.2 are different. To illustrate this point, we consider the following simple constraint system.

Example 3.4.1

$$\begin{aligned} &F(x) = x_1 - x_2 = 0, \\ &x = (x_1, x_2) \in [0, +\infty) \times [0, \infty). \end{aligned}$$

It is easy to see that the Mangasarian-Fromovitz constraint qualification holds at $x^\star = (0, 0)$, while $x^\star$ is not a regular zero of H.

We now show that the second-order sufficient conditions guarantee the validity of Assumption (B) as well. Recall that the second-order sufficient conditions as stated in Lemma 2.1.1 are said to hold at $x^\star$ if

(a) $x^\star$ is a KKT point, i.e., there exist $\lambda^\star, \eta^\star \in \mathbb{R}^n_+$, and $\mu^\star \in \mathbb{R}^m$ such that

$$\nabla f(x^\star) - \lambda^\star + \eta^\star + \nabla F(x^\star)^T \mu^\star = 0, \tag{3.4.27}$$

$$\inf(\lambda_i^\star, x_i^\star - u_i) = \inf(\eta_i^\star, v_i - x_i^\star) = 0, \quad i = 1, 2, \cdots, n. \tag{3.4.28}$$

(b) the matrix $\nabla^2_{xx}L(x^\star, \mu^\star)$ is positive definite on the cone $\{d \neq 0 | \nabla F(x^\star)d = 0, d_i = 0,\ as\ \lambda_i^\star > 0,\ or\ \eta_i^\star > 0\}$, where $L(x, \mu) = f(x) + \mu^T F(x)$.

The following example shows that the second-order sufficient conditions are independent of the Mangasarian-Fromovitz constraint qualification and regular zero of $\tilde{H}$.

Example 3.4.2

$$\begin{aligned} \min\ \ & f(x) = x_1^2 + x_2^2 \\ \text{s.t.}\ \ & F(x) = x_1 - x_2 = 0, \\ & x = (x_1, x_2) \in [0, +\infty) \times (-\infty, 0]. \end{aligned}$$

It is easy to check that at $x^\star = (0, 0)$, the second-order sufficient conditions hold, while the Mangasarian-Fromovitz constraint qualification and regular zero of $\tilde{H}$ do not hold true.

Corollary 3.4.3 *If $\beta(\varepsilon) \geq \sqrt{\varepsilon}$ for all $\varepsilon > 0$ sufficiently small and the second-order sufficient conditions hold at $x^\star$, then there exists $\gamma > 0$ such that $(x^\star, 0)$ is a local strict optimal solution of $(\tilde{P}_\sigma)$ for all $\sigma \geq \gamma$.*

Proof. Since the second-order sufficient conditions hold at $x^\star$, it then follows from [31, Theorem 4.6] that Assumption (B) is valid for $\Phi(t) = t$ with the inequality being strict. The desired result follows from Theorem 3.4.1. ∎

In what follows, we propose a new condition, which also guarantees the validity of Assumption (B).

Proposition 3.4.1 *Let $x^\star$ be a local optimal solution of (P). If $x^\star$ is a KKT point and*

$$\lim_{\substack{x \in [u,v] \to x^\star \\ F(x) \neq 0}} \frac{1}{\Phi(\|F(x)\|)} \int_0^1 (\nabla_x L(x^\star + s(x - x^\star), \mu^\star) - \nabla_x L(x^\star, \mu^\star))^T (x - x^\star) ds = 0, \quad (3.4.29)$$

where $\lambda^\star, \eta^\star \in \mathbb{R}^n_+$, $\mu^\star \in \mathbb{R}^m$ are the corresponding Lagrangian multipliers, then Assumption (B) holds at $x^\star$ for Φ with $\gamma = 1 + \|\mu^\star\|$.

Proof. We need to show the existence of a neighborhood of $x^\star$, say $N(x^\star)$, such that

$$f(x) - f(x^\star) + \gamma \Phi(\|F(x)\|) \geq 0, \quad x \in N(x^\star) \cap [u, v]. \quad (3.4.30)$$

Since $x^\star$ is a local optimal solution of (P), then there exists a neighborhood of $x^\star$, say $\widetilde{N}(x^\star)$, such that (3.4.30) holds true for all $x \in \widetilde{N}(x^\star) \cap [u,v]$ with $F(x) = 0$. Now consider the case of $x \in \widetilde{N}(x^\star) \cap [u,v]$ with $F(x) \neq 0$. Putting (3.4.27), (3.4.28), and (3.4.29) together yields

$$\begin{aligned}
f(x) - f(x^\star) &= \nabla f(x^\star)^T(x - x^\star) + \int_0^1 (\nabla f(x^\star + s(x - x^\star)) - \nabla f(x^\star))^T(x - x^\star)ds \\
&= \sum_{i \in I_1^\star} \lambda_i^\star(x_i - x_i^\star) - \sum_{i \in I_2^\star} \eta_i^\star(x_i - x_i^\star) - \mu^{*T}\nabla F(x^\star)(x - x^\star) \\
&\quad + \int_0^1 (\nabla f(x^\star + s(x - x^\star)) - \nabla f(x^\star))^T(x - x^\star)ds \\
&\geq -\mu^{*T}\nabla F(x^\star)(x - x^\star) + \int_0^1 (\nabla f(x^\star + s(x - x^\star)) - \nabla f(x^\star))^T(x - x^\star)ds \\
&= -\mu^{*T}F(x) + \int_0^1 \mu^{*T}(\nabla F(x^\star + s(x - x^\star)) - \nabla F(x^\star))(x - x^\star)ds \\
&\quad + \int_0^1 (\nabla f(x^\star + s(x - x^\star)) - \nabla f(x^\star))^T(x - x^\star)ds \\
&= -\mu^{*T}F(x) + \int_0^1 (\nabla_x L(x^\star + s(x - x^\star), \mu^\star) - \nabla_x L(x^\star, \mu^\star))^T(x - x^\star)ds \\
&= -\mu^{*T}F(x) + o(\Phi(\|F(x)\|)). \qquad (3.4.31)
\end{aligned}$$

This guarantees the existence of a neighborhood $N(x^\star) \subseteq \widetilde{N}(x^\star)$ of $x^\star$ such that

$$\frac{1}{\Phi(\|F(x)\|)}|o(\Phi(\|F(x)\|))| \leq \frac{1}{2}, \qquad (3.4.32)$$

whenever $x \in N(x^\star) \cap [u,v]$ with $F(x) \neq 0$. Therefore,

$$\begin{aligned}
f(x) - f(x^\star) + \gamma\Phi(\|F(x)\|) &\geq \gamma\Phi(\|F(x)\|) - \mu^{*T}F(x) + o(\Phi(\|F(x)\|)) \quad \textit{by } (3.4.31) \\
&\geq \gamma\Phi(\|F(x)\|) - \|\mu^\star\|\|F(x)\| + o(\Phi(\|F(x)\|)) \\
&\geq \Phi(\|F(x)\|)(\gamma - \|\mu^\star\|) + o(\Phi(\|F(x)\|)) \quad \textit{by } \Phi(t) \geq t \\
&= \Phi(\|F(x)\|) + o(\Phi(\|F(x)\|)) \quad \textit{by } \gamma = 1 + \|\mu^\star\| \\
&\geq \frac{1}{2}\Phi(\|F(x)\|) \quad \textit{by } (3.4.32) \\
&> 0.
\end{aligned}$$

This yields the inequality as desired. ∎

It should be emphasized that the condition (3.4.29) is independent with the other conditions used in the previous discussions, which is illustrated by the following example.

Example 3.4.3

$$\begin{aligned} \min\ & f(x) = -|x|^{\frac{3}{2}} \\ \text{s.t.}\ & F(x) = x^2 = 0, \\ & x \in (-\infty, +\infty). \end{aligned}$$

By a simple calculation, we know that, at $x^\star = 0$, the condition (3.4.29) holds when $\Phi(t) = \sqrt{t}$, while the Mangasarian-Fromovitz constraint qualification, the second-order sufficient conditions, and regular zero of F do not hold at $x^\star$. In addition, the condition (3.4.29) is also true for Example 3.4.2 when $\Phi(t) = t$.

Direct applications of Theorem 3.4.1 and Proposition 3.4.1 yields the following corollary.

Corollary 3.4.4 *Let $x^\star$ be a local optimal solution of (P) and $\beta(\varepsilon) \geq \Phi(\sqrt{\varepsilon})$ for all $\varepsilon > 0$ sufficiently small. If the condition (3.4.29) holds at $x^\star$, then $(x^\star, 0)$ is a local optimal solution of $(\widetilde{P}_\sigma)$ for $\sigma \geq 1 + \|\mu^\star\|$.*

We end this section by addressing the relationship between this new class of exact penalty functions and the classical simple penalty functions in the sense of "exactness" property. Define

$$f_\gamma(x) = f(x) + \gamma\Phi(\|F(x)\|),$$

where Φ satisfies (j_1) and (j_2). Clearly, $f_\gamma(\cdot)$ is a classical simple and exact penalty function. The associated penalty problem is

$$(P_\gamma) \qquad \begin{aligned} \min\ & f_\gamma(x) \\ \text{s.t.}\ & x \in [u, v]. \end{aligned}$$

In the present case, the condition (3.3.20) given in Assumption (B) means that $x^\star$ is a local optimal solution of (P_γ). Assume that $x^\star$ is a local optimal solution of (P) and according to Theorem 3.4.1, the relationship between $(\tilde{P}_\sigma)$ and (P_γ) is summarized as follows:

(1) Suppose that for sufficiently small $\varepsilon > 0$, $\beta(\varepsilon) \geq \Phi(\sqrt{\varepsilon})$ holds. If there exists $\gamma > 0$ such that $x^\star$ is a local optimal solution of the penalty problem (P_γ), then $(x^\star, 0)$ is a local optimal solution of the penalty problem $(\tilde{P}_\sigma)$ when $\sigma > 0$ is sufficiently large.

(2) Suppose that for sufficiently small $\varepsilon > 0$, $\beta(\varepsilon) = \Phi(\varepsilon)$ holds. If there exists $\sigma > 0$ such that $(x^\star, 0)$ is a local optimal solution of the penalty problem $(\tilde{P}_\sigma)$, then $x^\star$ is a local optimal solution of the penalty problem (P_γ) when $\gamma > 0$ is sufficiently large.

3.5 Penalty Function Methods

In this section, we present a revised penalty function algorithm based on $\tilde{f}_\sigma(\cdot,\cdot)$. Since $\tilde{f}_\sigma(\cdot,\cdot)$ is lower-bounded, the algorithm is always feasible. It should be noted that the parameter ε in $\tilde{f}_\sigma(\cdot,\cdot)$ plays a key role in the framework of the algorithm. It follows from the Assumption (A) and Theorem 3.3.1 that the proposed algorithm terminates at the optimal solutions of (P) after finitely many iterations (see Theorem 3.5.1). We further analyze the case without requirement of the Assumption (A). Under mild conditions, we present a perturbation theorem about the algorithm (see Theorem 3.5.2). As a corollary of this perturbation theorem, global convergence property of the algorithm can be derived (see Corollary 3.5.1).

Assumption (C) The function β satisfies $\inf\limits_{\varepsilon \geq \varepsilon_0} \beta(\varepsilon) > 0$ for all $\varepsilon_0 > 0$.

It is clear that the function β given in previous discussion satisfies this assumption. In the rest, for simplicity in exposition, we denote by $\operatorname{argmin}(\widetilde{P}_{\sigma_k})$ the optimal solutions set of the penalty problem $(\tilde{P}_{\sigma_k})$.

Algorithm 3.5.1 *Let $\alpha \in (0,1)$ be sufficiently small number, $\delta_0 > 0$, $\sigma_0 \geq 1$, and $k := 0$;*

Step 1. Solve $(\widetilde{P}_{\sigma_k})$ and let (x_k, ε_k) be the optimal solution obtained and then go to Step 2. Otherwise, find a δ-optimal solution (x_k, ε_k) such that

$$\widetilde{f}_{\sigma_k}(x,\varepsilon) \leq \inf\{\widetilde{f}_{\sigma_k}(x,\varepsilon) | (x,\varepsilon) \in [u,v] \times [0,\bar{\varepsilon}]\} + \delta_k,$$

and go to Step 3;

Step 2. If $\varepsilon_k = 0$, stop. Otherwise, go to Step 4;

Step 3. If $\varepsilon_k = 0$ and $\delta_k \leq \alpha$, stop;

Step 4. Let $\delta_{k+1} = \frac{1}{2}\delta_k$ *and*

$$\sigma_{k+1} = \begin{cases} \sigma_k, & \textit{if } \varepsilon_k = 0, \\ 2\sigma_k, & \textit{otherwise.} \end{cases}$$

Step 5. Let $k := k + 1$ *and go back to step 1.*

This algorithm is always feasible, since the existence of δ-optimal solution is ensured by the lower-boundedness of $\widetilde{f}_\sigma(\cdot,\cdot)$ over $D \times [0, \bar{\varepsilon}]$. In addition, due to the smoothness of $\widetilde{f}_{\sigma_k}(\cdot, \varepsilon)$ on D_F, many descent algorithms can be used to find (x_k, ε_k). Given $\eta \geq 0$, define a perturbation function of (P) as follows

$$\theta(\eta) = \inf\{f(x)|x \in [u, v], \|F(x)\| \leq \eta\}.$$

Clearly, $\theta(\eta)$ is a non-increasing function with $\theta(0)$ equal to the optimal value of (P). Since $f(x)$ is lower-bounded on $[u, v]$, then the limit of θ as η approaches to 0^+ exists and

$$\lim_{\eta \to 0^+} \theta(\eta) \leq \theta(0),$$

i.e., θ is upper semi-continuous at zero.

Theorem 3.5.1 *Under the assumptions of Theorem 3.3.1, let* $\{(x_k, \varepsilon_k)\}$ *be a sequence generated by Algorithm 3.5.1 and* $(x_k, \varepsilon_k) \in \mathit{argmin}(\widetilde{P}_{\sigma_k})$*. Then there exists* k_0 *such that when* $\sigma_k \geq \sigma_{k_0}$*,* $\varepsilon_k = 0$*, i.e.,* x_k *is an optimal solution of* (P)*.*

Proof. From Theorem 3.3.1, there exists k_0 such that when $\sigma_k \geq \sigma_{k_0}$, $\varepsilon_k = 0$. It follows from the fact of finite value $\tilde{f}_{\sigma_k}(x_k, 0)$ and the definition of $\tilde{f}_\sigma$ that $F(x_k) = 0$, namely, $\tilde{f}_{\sigma_k}(x_k, 0) = f(x_k)$ holds. Furthermore, $(x_k, 0) \in \text{argmin}(\widetilde{P}_{\sigma_k})$ yields that $x_k \in \text{argmin}(P)$. ■

Lemma 3.5.1 *Let* β *satisfy Assumption* (C)*, and* $\{(x_k, \varepsilon_k)\}$ *be an infinite sequence generated by Algorithm 3.5.1. Then there exists* k_0 *such that* $\varepsilon_k > 0$ *for all* $k \geq k_0$*, and*

(1) $\lim_{k\to\infty} \varepsilon_k = 0$*;*

(2) $\lim_{k\to\infty} F(x_k) = 0$.

Proof. Since $\{(x_k, \varepsilon_k)\}$ is an infinite sequence, at least one of the conditions in Step 3 fails to hold for all k. This, along with the reduction of δ_k in Step 4, implies that there exists k_0 such that $\varepsilon_k > 0$ for all $k \geq k_0$. Then, by the condition for the update of σ_k in Step 4, we have

$$\lim_{k\to\infty} \sigma_k = \infty. \tag{3.5.33}$$

From Step 1, we have

$$\begin{aligned} f(x_k) + \frac{1}{2\varepsilon_k}\phi(\Delta_k) + \sigma_k\beta(\varepsilon_k) \;&\leq\; \inf\{\widetilde{f}_{\sigma_k}(x,\varepsilon)|(x,\varepsilon) \in [u,v]\times[0,\overline{\varepsilon}]\} + \delta_k \\ &\leq\; f(\overline{x}) + \delta_k, \end{aligned} \tag{3.5.34}$$

where $\overline{x}$ is a feasible point of (P). Note that f is lower-bounded on D and ϕ is nonnegative. It readily follows from (3.5.33) and (3.5.34) that $\lim\limits_{k\to\infty} \beta(\varepsilon_k) = 0$, which, together with Assumption (C), further implies that

$$\lim_{k\to\infty} \varepsilon_k = 0. \tag{3.5.35}$$

Similarly, according to the nonnegativity of β, we get from (3.5.34) and (3.5.35) that $\lim\limits_{k\to\infty} \phi(\Delta_k) = 0$, and hence

$$\lim_{k\to\infty} \Delta_k = 0, \tag{3.5.36}$$

where we have used the gradient inequality of convex function ϕ, i.e., $\phi(\Delta_k) \geq \phi'(0)\Delta_k \geq 0$. Since $\|F(x_k)\| \leq \sqrt{\Delta_k} + \varepsilon_k\|\omega\|$, putting (3.5.35) and (3.5.36) together yields $\lim\limits_{k\to\infty} \|F(x_k)\| = 0$ as claimed. ■

The global convergence property of Algorithm 3.5.1 is given below.

Theorem 3.5.2 *Let β satisfy Assumption (C). The following statements hold:*

(1) *Assume Algorithm 3.5.1 stops after k iterations, if k belongs to Step 2, then x_k is an optimal solution of (P), and if k belongs to Step 3, then x_k is α-optimal solution of (P).*

(2) *If an infinite sequence is generated, then*

$$\lim_{k\to\infty} f(x_k) = \lim_{\eta\to 0^+} \theta(\eta),$$

and

$$\lim_{k\to\infty} \inf\{\widetilde{f}_{\sigma_k}(x,\varepsilon) \mid (x,\varepsilon) \in [u,v]\times[0,\overline{\varepsilon}]\} = \lim_{\eta\to 0^+} \theta(\eta).$$

Proof. (1) Suppose the algorithm stops after the k-th iteration, then $\varepsilon_k = 0$. Since $\widetilde{f}_{\sigma_k}(x_k, 0)$ is finite, then $F(x_k) = 0$, and $\widetilde{f}_{\sigma_k}(x_k, 0) = f(x_k)$ must hold according to the definition of $\tilde{f}_\sigma$. Therefore, if k belongs to Step 2, it further follows from Step 1 that x_k is an optimal solution to (P); if k belongs to Step 3, then it follows from Step 3 that x_k is α-optimal solution of (P).

(2) If the algorithm does not stop after the k-th iteration, then we must have $\lim_{k\to\infty} \sigma_k = +\infty$. By Lemma 3.5.1, there exists k_0 such that $\varepsilon_k > 0$ for all $k \geq k_0$, since one of the termination conditions $\delta_k \leq \alpha$ is always true as k sufficiently large. Because $\widetilde{f}_{\sigma_k}(x_k, \varepsilon_k)$ is finite, then

$$\widetilde{f}_{\sigma_k}(x_k, \varepsilon_k) = f(x_k) + \frac{1}{2\varepsilon_k}\phi(\Delta_k) + \sigma_k \beta(\varepsilon_k).$$

The continuity of β and the fact $\beta(0) = 0$ guarantee the existence of $\bar{\varepsilon} \geq \bar{\varepsilon}_k > 0$ such that $\lim_{k\to\infty} \bar{\varepsilon}_k = 0$ and

$$\lim_{k\to\infty} \sigma_k \beta(\bar{\varepsilon}_k) = 0. \tag{3.5.37}$$

Indeed, for each k, we can choose $\bar{\varepsilon}_k$ such that $\beta(\bar{\varepsilon}_k) \leq \frac{1}{\sigma_k^2}$. Choose another sequence $\xi_k > 0$ satisfying $\lim_{k\to\infty} \xi_k = 0$. According to the definition of infimum in θ, there exists $y_k \in [u, v]$ such that $\|F(y_k)\| \leq \bar{\varepsilon}_k$, and

$$f(y_k) \leq \theta(\bar{\varepsilon}_k) + \xi_k. \tag{3.5.38}$$

Let $\overline{\Delta}_k = \|F(y_k) - \bar{\varepsilon}_k \omega\|^2$. Since $\|F(y_k)\| \leq \bar{\varepsilon}_k$, then

$$\overline{\Delta}_k \leq \bar{\varepsilon}_k^2 (1 + \|\omega\|)^2, \tag{3.5.39}$$

from which and the fact $\lim_{k\to\infty} \bar{\varepsilon}_k = 0$, we get $\overline{\Delta}_k < a$ (cf. (3.2.2)) for all k sufficiently large. Thus, $f_{\sigma_k}(y_k, \bar{\varepsilon}_k)$ is finite by (3.2.2). Lemma 3.5.1 implies that, for any η, we have $\|F(x_k)\| \leq \eta$ as k large enough. Hence, $\theta(\eta) \leq f(x_k)$ holds by the definition of θ.

Therefore,

$$
\begin{aligned}
\theta(\eta) &\leq f(x_k) \\
&\leq f(x_k) + \frac{1}{2\varepsilon_k}\phi(\Delta_k) + \sigma_k\beta(\varepsilon_k) \\
&= \widetilde{f}_{\sigma_k}(x_k, \varepsilon_k) \\
&\leq \inf\{\widetilde{f}_{\sigma_k}(x,\varepsilon) | (x,\varepsilon) \in [u,v] \times [0,\overline{\varepsilon}]\} + \delta_k \\
&\leq \widetilde{f}_{\sigma_k}(y_k, \overline{\varepsilon}_k) + \delta_k \\
&= f(y_k) + \frac{1}{2\overline{\varepsilon}_k}\phi(\overline{\Delta}_k) + \sigma_k\beta(\overline{\varepsilon}_k) + \delta_k \\
&\leq \theta(\overline{\varepsilon}_k) + \xi_k + \frac{1}{2\overline{\varepsilon}_k}\phi'(\overline{\Delta}_k)\overline{\Delta}_k + \sigma_k\beta(\overline{\varepsilon}_k) + \delta_k \\
&\leq \theta(\overline{\varepsilon}_k) + \xi_k + \frac{1}{2}\phi'(\overline{\Delta}_k)\overline{\varepsilon}_k(1 + \|\omega\|)^2 + \sigma_k\beta(\overline{\varepsilon}_k) + \delta_k,
\end{aligned}
$$

where the fifth inequality comes from (3.5.38) and the convexity of ϕ as before, and the last inequality is due to (3.5.39). The desired result follows by taking limits on both sides of above inequality and using (3.5.37). ■

Corollary 3.5.1 *Let β satisfy Assumption (C), and $\{(x_k, \varepsilon_k)\}$ be an infinite sequence generated by Algorithm 3.5.1. The following statements hold.*

(1) $\lim\limits_{k\to\infty} f(x_k) = \theta(0)$ *if and only if θ is lower semi-continuous at zero.*

(2) *If $x^\star$ is an accumulation point of $\{x_k\}$, then $x^\star$ is a global optimal solution of (P).*

Proof. (1) This follows from statement 2 of Theorem 3.5.2, since θ is always upper semicontinuous at zero.

(2) If $x^\star$ is an accumulation point, then $F(x^\star) = 0$ by Lemma 3.5.1, i.e., $x^\star$ is feasible. According to statement 2 of Theorem 3.5.2, we have

$$\lim_{\eta\to 0^+} \theta(\eta) = \lim_{k\to\infty} f(x_k) = f(x^\star) \geq \theta(0),$$

which means the lower continuity of θ at zero. This, together with statement (1), yields $\lim\limits_{k\to\infty} f(x_k) = \theta(0)$, i.e., $f(x^\star) = \theta(0)$, which means that $x^\star$ is a global optimal solution of (P). ■

3.6 Numerical Results

To give some insight into the behavior of our proposed algorithm presented in this section, numerical tests are performed on four nonlinear programming problems with equality constraints obtained from [36]. It is implemented in Matlab 7.8.0 and runs are made on Intel Core 2 CPU 2.39 GHz with 1.99 GB memory. Tables 3.1-3.5 show the computational results for the corresponding problems with the following items:

$\phi_i(t)\,(i = 1, 2, \ldots, 6)$ as defined in Section 3.2,
σ_k the penalty parameter,
x_k, ε_k the final iterate,
$\Delta(x_k, \varepsilon_k)$ the violation of the constraint,
$\widetilde{f}_{\sigma_k}(x_k, \varepsilon_k)$ the value of penalty function $\widetilde{f}_\sigma(x, \varepsilon)$ at the final (x_k, ε_k) when the penalty parameter σ_k,

Example 3.6.1 *([36])*

$$\begin{aligned} &\min f(x) = -x_1 \\ &\text{s.t. } F_1(x) = x_2 - x_1^3 - x_3^2 = 0, \\ &\qquad F_2(x) = x_1^2 - x_2 - x_4^2 = 0. \end{aligned}$$

Then, the point $\bar{x} = (1, 1, 0, 0)^\top$ is a minimizer with the optimal objective function value -1.0000 .

Example 3.6.2 *([36])*

$$\begin{aligned} &\min f(x) = (x_1 - x_2)^2 + (x_2 - x_3)^2 + (x_3 - x_4)^4 + (x_4 - x_5)^4 \\ &\text{s.t. } F_1(x) = x_1 + x_2^2 + x_3^3 - 3 = 0, \\ &\qquad F_2(x) = x_2 - x_3^2 + x_4 - 1 = 0. \end{aligned}$$

The point $\bar{x} = (1, 1, 1, 1, 1)^\top$ is a (global) minimizer with the optimal objective function value 0.

Table 3.1: Numerical results of Example 3.6.1

$\phi_i(t)$	σ_k	x_k	ε_k	$\Delta(x_k,\varepsilon_k)$	$\widetilde{f}_{\sigma_k}(x_k,\varepsilon_k)$
$\phi_1(t)$	1	(1.0242, 1.0620, -0.0033, 0.0214)	0.0060	5.3129e-004	-0.9024
$= \frac{t}{1-0.1t}$	2	(0.9955, 0.9884, 0.0445, 0.0494)	5.7578e-005	2.0030e-008	-0.9801
	4	(1.0000, 1.0000, -5.7647e-004,0.0032)	1.7402e-007	5.2608e-010	-0.9968
$\phi_4(t) = t$	1	(0.9994, 0.9984, -0.0252, -0.0157)	0.0014	3.7181e-006	-0.9600
	2	(1.0001, 1.0004, 0.0185, -0.0047)	1.3967e-004	2.7711e-007	-0.9755
	4	(0.9999, 0.9999, 0.0003, -0.0022)	2.6207e-007	9.9595e-010	-0.9960
$\phi_6(t)$	1	(1.0064, 1.0296, 0.0860, -0.0031)	0.0028	2.9153e-004	-0.9274
$= \frac{1}{2}(\sqrt{t^2+4}+t) - 1$	2	(0.9953, 0.9878, 0.0439, -0.0528)	5.4413e-007	4.8769e-009	-0.9919
	4	(0.9998, 0.9993, 9.3599e-004, 0.0152)	8.6056e-009	1.3469e-011	-0.9990

Example 3.6.3 *([36])*

$$\begin{aligned} \min\ & f(x) = (4x_1 - x_2)^2 + (x_2 + x_3 - 2)^2 + (x_4 - 1)^2 + (x_5 - 1)^2 \\ \text{s.t. } & F_1(x) = x_1 + 3x_2 = 0, \\ & F_2(x) = x_3 + x_4 - 2x_5 = 0, \\ & F_3(x) = x_2 - x_5 = 0. \end{aligned}$$

Then, the point $\bar{x} = (-\frac{33}{349}, \frac{11}{349}, \frac{180}{349}, -\frac{158}{349}, \frac{11}{349})^\top$ is a minimizer with the optimal objective function value $\frac{1859}{349}$.

Example 3.6.4 *([36])*

$$\begin{aligned} \min\ & f(x) = \sum_{j=1}^{10} x_j(c_j + \ln \tfrac{x_j}{x_1+\cdots+x_{10}}) \\ \text{s.t. } & F_1(x) = x_1 + 2x_2 + 2x_3 + x_6 + x_{10} - 2 = 0, \\ & F_2(x) = x_4 + 2x_5 + x_6 + x_7 - 1 = 0, \\ & F_3(x) = x_3 + x_7 + x_8 + 2x_9 + x_{10} = 0, \\ & x_i \geq 0, i = 1, 2, \cdots, 10. \end{aligned}$$

where $c_1 = -6.089, c_2 = -17.164, c_3 = -34.054, c_4 = -5.914, c_5 = -24.721, c_6 = -14.986, c_7 = -24.100, c_8 = -10.708, c_9 = -26.662, c_{10} = -22.179$. Then, the point

Table 3.2: Numerical results of Example 3.6.2

$\phi_i(t)$	σ_k	x_k	ε_k	$\Delta(x_k, \varepsilon_k)$	$\widetilde{f}_{\sigma_k}(x_k, \varepsilon_k)$
$\phi_1(t)$	1	(1.0026, 1.0019, 0.9979, 0.9936, 0.9971)	6.4899e-008	2.5015e-007	0.0100
$= \frac{t}{1-0.1t}$	2	(0.9997, 0.9908, 1.0062, 1.0216, 1.0003)	5.6659e-008	2.3655e-011	0.0010
	4	(0.9972, 1.0071, 0.9962, 0.9852, 1.0028)	1.3359e-008	8.6253e-012	0.0010
$\phi_4(t)$	1	(0.9915, 0.9905, 1.0090, 1.0272, 1.0084)	6.4169e-005	2.1140e-007	0.0100
	2	(1.0034, 1.0016, 0.9978, 0.9941, 0.9966)	1.2454e-006	6.8508e-009	0.0050
	4	(0.9979, 1.0006, 1.0003, 0.9999, 1.0021)	1.0000e-006	6.5704e-011	0.0040
$\phi_6(t)$	1	(0.9835, 0.9875, 1.0135, 1.0397, 1.0163)	3.7318e-005	4.7781e-007	0.0100
	2	(1.0221, 1.0006, 0.9922, 0.9840, 0.9784)	1.5117e-006	1.2133e-008	0.0050
	4	(1.0083, 1.0038, 0.9947, 0.9856, 0.9917)	8.8059e-007	6.7950e-010	0.0040

$\bar{x} = (0.1083, 0.9438, 0, 0.0004, 0.4978, 0.0040, 0, 0, 0, 0)^\top$ is a (but not unique) local minimizer with the optimal objective function value -30.581215.

We make numerical tests using different choices of the function $\phi_i, i = 1, 4, 6$ defined in Section 3.2, where $\phi_1(t)$ is barrier-type penalty function and $\phi_4(t), \phi_6(t)$ are exterior-type penalty functions. Here, $\beta(\varepsilon)$ in (3.2.2) is set as $\sqrt{\varepsilon}$. The data in Tables 3.1-3.5 illustrate the practical behavior of the algorithm proposed in this paper. The algorithm starts with σ_0. The penalty problem is (approximately) solved by any unconstrained smooth minimization techniques (for instance, trust-region methods, Newton-type methods and conjugate gradient methods) since the proposed penalty function has good smoothness property. We can thus denote by (x_0, ε_0) the optimal solution when the penalty parameter is σ_0. Once the iterate point $\varepsilon_0 \neq 0$, set $\sigma_1 = 2\sigma_0$ and the algorithm runs over again. As the penalty parameter gradually increases, the indicator variable ε_k, the constraints violation measure value $\Delta(x_k, \varepsilon_k)$ and the penalty function value $\widetilde{f}_{\sigma_k}(x_k, \varepsilon_k)$ decrease as desired. Additionally, it is not difficult to observe that the minimizers can be obtained without requirements of large penalty parameters for the kinds of choices of the functions ϕ_i considered here. Numerical performances verify the correctness of the developed theory as desired. For example, as illustrated in Tables 3.1-3.5, the iterates (x_k, ε_k) are already quite close to the point $(\bar{x}, 0)$, where $\bar{x}$ is a minimizer of the original problem. Just as stated in Theorem 3.3.1, the optimal solution

Table 3.3: Numerical results of Example 3.6.3

$\phi_i(t)$	σ_k	Iters	x_k	ε_k	$\Delta(x_k, \varepsilon_k)$	$\widetilde{f}_{\sigma_k}(x_k, \varepsilon_k)$
$\phi_1(t)$	10	62	(-0.1083, 0.0360, 0.5365, -0.4644,0.0360)	1.3508e-006	7.7016e-006	5.3317
$= \frac{t}{1-0.01t}$	20	52	(-0.0935, 0.0311, 0.5130, -0.4506,0.0311)	1.0000e-006	6.0195e-011	5.3266
	40	57	(-0.0936, 0.0312, 0.5130,-0.4505,0.0312)	1.0000e-006	5.7181e-011	5.3266
$\phi_4(t)$	10	47	(-0.0701, 0.0551, 0.7457, -0.1994,0.2297)	0.0299	0.0475	3.5825
	20	57	(-0.0936, 0.0312, 0.5124, -0.4499,0.0312)	1.0515e-006	5.8772e-011	5.3266
	40	57	(-0.0944, 0.0315, 0.5126, -0.4495,0.0315)	1.1337e-006	9.8943e-011	5.3265
$\phi_6(t)$	10	41	(-0.0365, 0.0864, 1.0685, 0.1549,0.5093)	0.0588	0.2727	1.7232
	20	62	(-0.0997, 0.0332, 0.5246, -0.4583,0.0332)	1.0000e-006	5.7256e-008	5.3280
	40	54	(-0.0942, 0.0314, 0.5149, -0.4521,0.0314)	1.0626e-006	3.6097e-006	5.3265

Table 3.4: Numerical results of Example 3.6.4

$\phi_i(t)$	σ_k	Iters	x_k
$\phi_1(t)$	1	66	(0.1191,0.9396,7.7552e-005, 0.0028, 0.4978,0.0015, 0, 0, 0, 0)
$= \frac{t}{1-0.01t}$	4	60	(0.9749, 0.5096, 0.0002, 0.0054, 0.4945, 0.0053, 0, 0, 0, 0)
	16	70	(0.0998,0.9482,1.2600e-005,0.0014,0.4975,0.0035,0,7.4320e-005,0,0.0002)
$\phi_4(t)$	2	42	(0.1002,0.9487,2.3861e-004,0.0013,0.4985,0.0017,6.7980e-007,7.7881e-007,0,1.2701e-005)
	4	53	(0.1013, 0.9480, 0, 0.0011,0.4981,0.0025,5.370e-007, 0, 0, 0)
	8	44	(0.1083, 0.9438, 1.3200e-006, 2.6780e-004, 0.4978, 0.0040, 0, 0, 0, 0)
$\phi_6(t)$	1	43	(0.1003,0.9486,0.0003,0.0013,0.4985,0.0017,1.1820e-006,2.1168e-006,0,2.5201e-005)
	4	49	(0.1003,0.9486,0.0003,0.0013,0.4985,0.0018,4.3192e-006,1.5402e-006,0,1.4780e-005)
	16	64	(0.1059, 0.9458, 2.9000e-007, 0.0011, 0.4982, 0.0023, 0, 0, 0, 0)

Table 3.5: Numerical results of Example 3.6.4

$\phi_i(t)$	ε_k	$\Delta(x_k, \varepsilon_k)$	$\widetilde{f}_{\sigma_k}(x_k, \varepsilon_k)$
$\phi_1(t)$	1.5441e-006	3.8468e-008	-30.5591
$= \frac{t}{1-0.01t}$	1.1971e-006	2.5382e-007	-29.1162
	9.7000e-006	1.2211e-007	-30.5731
$\phi_4(t)$	1.0000e-005	2.5026e-012	-30.5803
	1.0000e-008	3.8461e-013	-30.5826
	1.0000e-008	8.8444e-008	-30.5815
$\phi_6(t)$	1.0000e-005	2.1877e-007	-30.5873
	1.0000e-005	2.1907e-007	-30.5778
	1.0000e-008	2.7457e-013	-30.5813

of the penalty problem must take the form of (x_k, ε_k) with $\varepsilon_k = 0$ for sufficiently large penalty parameters, which means x_k is a local optimal solution of the primal problem (P). In summary, our numerical experiments on four classical nonlinear programming problems with equality constraints confirm the efficiency of the proposed algorithm. As shown in Tables 3.1-3.5, the numerical outputs for the different choices for the functions $\phi_i, i = 1, 4, 6$ seem to have no significant differences, which demonstrates the class of convex functions presenting an integrated representation for both barrier- and exterior-type functions are effective for solving nonlinear programming problems. Nevertheless, there exists a little difference in the algorithm implementation process for solving exterior-type penalty functions and barrier-type penalty functions. For barrier-type penalty functions, we must consider the additional constraint $\Delta(x, \varepsilon) < a$ compared with exterior-type penalty functions. Here, for example, $a = q^{-1}$ for $\phi_1(t)$, $a = \frac{\pi}{2}$ for $\phi_2(t)$ and $a = 1$ for $\phi_3(t)$.

Chapter 4

On an Exact Penalty Function Method for Semi-Infinite Programming Problems

4.1 Introduction

In the context of this chapter, we consider the following semi-infinite programming problem:

$$
\begin{aligned}
(Q) \qquad & \min \quad f(x) \\
& \text{s.t.} \quad g(x,v) \le 0 \quad \forall\, v \in V,
\end{aligned}
$$

where V is a nonempty compact set and $f : \mathbb{R}^n \to \mathbb{R},\ g : \mathbb{R}^{n+m} \to \mathbb{R}$ are continuously differentiable functions with respects to x and v.

The semi-infinite programming problem has attracted much attention due to its various applications. It is encountered in optimal control problems, the design of the flutter of aircraft wings and economic equilibria, etc. It has become an active field of research in applied mathematics. The most prominent feature of the semi-infinite programming problem is that it has a finite number of variables but infinitely many constraints, which brings great difficulties in designing efficient algorithms. Nevertheless, in recent years, many effective methods have been proposed for solving the semi-infinite programming problem such as exchange methods, discretization methods local reduction methods etc. In particular, some researchers also develop penalty function algorithms for the semi-infinite programming problem. For instance, Conn and Gould developed an exact penalty function based on a generalization of the l_1 exact rather than smooth penalty function in [14] for a class of semi-infinite programming problems, which makes smoothing optimization methods inapplicable. In addition, it involves restrictive assumptions, such as linearly independent constraint qualification in semi-infinite programming and strong (convexity) assumptions. Nevertheless, to our knowledge, the reference [14] is the first article to introduce an exact penalty function for solving semi-infinite programming.

Motivated by [38], in this section, we propose a new exact and smooth penalty function for the semi-infinite programming problems. The main feature of our penalty function is that we only need to add one variable ε to handle infinitely many constraints. The merit function is considered as a function of x and ε simultaneously which has good smoothness and exactness properties, without involving gradient and Jacobian matrices. It remains bounded below whenever $f(x)$ is bounded below on S, which is not shared by the l_1 exact penalty function. It is well-known that the ill-conditioness introduced by a large penalty parameter may be detrimental. Therefore, for the new exact penalty function algorithm, we require to increase the penalty parameter gradually by adding a relatively small constant in order to keep the penalty parameter as small as possible avoiding ill-conditioness to occur, which is illustrated in the comparison with l_1 exact penalty function in Section 4.5. We will also present the result that, if a local optimal solution to the penalty problem satisfies the extended Mangasarian-Fromovitz constraint qualification, then the minimizer has the expression of $(x^\star, 0)$. In addition, we derive another useful property that the minimizer $(x^\star, \varepsilon^\star)$ of the penalty problem satisfies $\varepsilon^\star = 0$ if and only if $x^\star$ solves the original problem (Q). This property demonstrates that the introduced new variable ε can be viewed as an indicator variable of a local (global) minimizer of primal problem (Q). Besides the above properties, we show that the penalty problem possesses exact penalty property or equivalently, for sufficiently large penalty parameter, the minimizer of penalty problem is the minimizer of original problem. Alternatively, under some mild conditions, the local exactness proof is shown.

The rest of this chapter is organized as follows. In Section 4.2, we introduce an exact and smooth penalty function for the semi-infinite programming problems. In Section 4.3, a penalty function algorithm and convergence analysis are presented. In Section 4.4, we discuss local exactness property of this exact penalty function. Some numerical tests are reported in Section 4.5.

4.2 A New Exact and Smooth Penalty Function for Semi-Infinite Programming Problems

For the semi-infinite programming problem,

$$\min_{x\in S} f(x), \quad S = \{x \in \mathbb{R}^n : g(x,v) \leq 0 \ \ \forall v \in V\} \neq \emptyset. \tag{4.2.1}$$

We introduce a new variable ε into the constraint function such that

$$S_\varepsilon = \{(x,\varepsilon) \in \mathbb{R}^n \times \mathbb{R}_+ : g(x,v) \leq \varepsilon^\gamma w \ \ \forall v \in V\},$$

where γ is a positive number. We make some assumptions:

(1) There exists the global minimizer of the problem (Q), this means that $f(x)$ is bounded below on S;

(2) Let $L(Q)$ denote the set of local minimizers of the problem (Q). If $x^\star \in L(Q)$, then $L_{x^\star} = \{x \in L(Q) : f(x) = f(x^\star)\}$ is a compact set.

In order to construct the penalty function for the semi-infinite programming problem, the integral function technique is employed for the constraint set. Denote

$$\Delta(x,\varepsilon) = \int_V (\max(0, g(x,v) - \varepsilon^\gamma w))^2 d\mu(v) = \int_{V^+(x,\varepsilon)} (g(x,v) - \varepsilon^\gamma w)^2 d\mu(v),$$

where $V^+(x,\varepsilon) = \{v \in V : g(x,v) - \varepsilon^\gamma w \geq 0\}$ and $\mu(v)$ is a given measure on V. The term $\Delta(x,\varepsilon)$ measures the total constraint violation at (x,ε). Clearly, for any $x \in \mathbb{R}^n, \Delta(x,\varepsilon) \geq 0$. Then the penalty function $f_\sigma(x,\varepsilon)$ can be formulated as follows.

$$(Q_\sigma) \qquad \min_{(x,\varepsilon)\in\mathbb{R}^n\times(0,\bar{\varepsilon})} f_\sigma(x,\varepsilon),$$

$$f_\sigma(x,\varepsilon) = \begin{cases} f(x), & \text{if } \varepsilon = 0, x \in S, \\ f(x) + \frac{\varepsilon^{-\alpha}\Delta(x,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)} + \sigma\varepsilon^{\beta}, & \text{if } \varepsilon > 0, 0 < 1 - 2\varepsilon^{-2\delta}\Delta(x,\varepsilon) < 1, \\ +\infty, & \text{otherwise,} \end{cases}$$

where $\alpha, \beta, \delta, \gamma$ are positive numbers, and $\beta > 1$. Denote

$$D = \{(x,\varepsilon) \in \mathbb{R}^n \times (0, \bar{\varepsilon}) \;\mid\; \varepsilon > 0,\ 0 < 1 - 2\varepsilon^{-2\delta}\Delta(x,\varepsilon) < 1\}.$$

For $\varepsilon = 0, x \in S$ and $(x,\varepsilon) \in D$,

$$f_\sigma(x,\varepsilon) = f(x) + \frac{\varepsilon^{-\alpha}\Delta(x,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)} + \sigma\varepsilon^{\beta} \geq f(x) = f_\sigma(x,0), \text{ if } \; g(x,v) \leq 0, \forall v \in V.$$

Also, $f_\sigma(x,\varepsilon)$ is bounded below on $\mathbb{R}^n \times [0,\bar{\varepsilon}]$ whenever $f(x)$ is bounded below on the set

$$D' = \{x \in \mathbb{R}^n | \|g(x,v)\| \leq \Big(\frac{1}{2m(V)}\Big)^{\frac{1}{2}} \bar{\varepsilon}^{\delta} + \bar{\varepsilon}^{\gamma}\|w\|, \forall v \in V\},$$

whenever $f(x)$ is bounded below on the set D', in which $m(V)$ represents the measure of the set V. This is a reasonable condition since when f is bounded below on the feasible set, $\bar{\varepsilon}$ is small enough. To illustrate the boundedness property, in what follows, we consider a simple semi-infinite programming example.

Example 4.2.1

$$\begin{aligned} \min \quad & x_1^3 x_2^3, \\ s.t. \quad & x_1^2 + x_2^2 - v \leq 0 \quad \forall\, v \in [1,2]. \end{aligned}$$

There are two global minimizers $x_1^\star = (\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2})$ and $x_2^\star = (-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$ with $f(x_1^\star) = f(x_2^\star) = -\frac{1}{8}$. If we use the traditional penalty function, we have the following conclusions:

- The l_1 penalty function:

$$l_1(x) = x_1^3 x_2^3 + \frac{1}{\varepsilon^{\alpha}}\Big(\int_1^2 (x_1^2 + x_2^2 - v)^+ du(v)\Big)$$

  is unbounded below for any $\varepsilon > 0$. Because when $x = (-m, m)^\top, m \to +\infty$, $l_1(x) \to -\infty$.

- The quadratic penalty function:

$$l_2(x) = x_1^3x_2^3 + \frac{1}{\varepsilon^\alpha}\Big(\int_1^2 (x_1^2 + x_2^2 - v)^+ du(v)\Big)^2$$

is unbounded below for any $\varepsilon > 0$. Because when $x = (-m, m)^\top, m \to +\infty$, $l_2(x) \to -\infty$.

For this new penalty function, choosing $w = 1$, we have

$$f_\sigma(x,\varepsilon) = \begin{cases} x_1^3x_2^3, & \text{if } \varepsilon = \Delta(x,\varepsilon) = 0, \\ x_1^3x_2^3 + \frac{\varepsilon^{-\alpha}\Delta(x,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)} + \sigma\varepsilon^\beta, & \text{if } \varepsilon > 0, 0 < 1 - 2\varepsilon^{-2\delta}\Delta(x,\varepsilon) < 1, \\ +\infty, & \text{otherwise,} \end{cases}$$

where $\Delta(x,\varepsilon) = \int_1^2 (\max(0, x_1^2 + x_2^2 - v - 0.5\varepsilon^\gamma))^2 du(v)$. Since $f_\sigma(x,\varepsilon) = +\infty$, if $\|x\| \geq \sqrt{2 + 0.5\varepsilon^\gamma + \frac{\sqrt{2}}{2}\varepsilon^\delta}$ for $\varepsilon > 0$. When $\|x\| < \sqrt{2 + 0.5\varepsilon^\gamma + \frac{\sqrt{2}}{2}\varepsilon^\delta}$, the boundedness below of this new penalty function is easily to be verified, which is not shared by l_1 penalty and quadratic penalty functions.

In what follows, we shall show that, $f_\sigma(x,\varepsilon)$ is continuously differentiable with continuous limits on the part of the boundary with finite values.

Proposition 4.2.1 *Let $x \to x^\star \in S, 0 < \varepsilon \to \varepsilon^\star = 0$. Suppose that*

$$\begin{cases} \gamma > \delta > \alpha > 0, \\ -\alpha - 1 + 2\delta > 0, \\ \beta > 1, \end{cases} \tag{4.2.2}$$

then $\lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ x \to x^\star \in S}} f_\sigma(x,\varepsilon) = f_\sigma(x^\star, 0) = f(x^\star)$, $\lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ x \to x^\star \in S}} \nabla_x f_\sigma(x,\varepsilon) = \nabla f(x^\star)$ *and* $\lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ x \to x^\star \in S}} \frac{\partial f_\sigma(x,\varepsilon)}{\partial \varepsilon} = 0.$

Proof. From the fact that $\varepsilon \neq 0, 0 < 1 - 2\varepsilon^{-2\delta}\Delta(x,\varepsilon) < 1$, we have $\Delta(x,\varepsilon) = O(\varepsilon^{2\delta})$ and $\lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ x \to x^\star \in S}} 1 - \varepsilon^{-2\delta}\Delta(x,\varepsilon) = c^\star \in [\frac{1}{2}, 1]$ when $\varepsilon \to \varepsilon^\star = 0$ and $x \to x^\star = 0$. Through the Cauchy inequality, it yields that $\int_V \max(0, g(x,v) - \varepsilon^\gamma w) d\mu(v) = O(\varepsilon^\delta)$ as $\varepsilon \to \varepsilon^\star = 0$.

As specified in (4.2.2), we know $2\delta > \alpha$ and $\beta > 1$. This yields

$$\lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ x \to x^\star \in S}} f_\sigma(x,\varepsilon) = \lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ x \to x^\star \in S}} f(x) + \frac{\varepsilon^{-\alpha}\Delta(x,\varepsilon)}{1 - \varepsilon^{-2\delta}\Delta(x,\varepsilon)} + \sigma\varepsilon^\beta = f(x^\star).$$

Notice that $f_\sigma(x,\varepsilon)$ is continuously differentiable in the set D. The gradient of $f_\sigma(x,\varepsilon)$ at (x,ε) can be easily derived as follows.

$$\nabla_{(x,\varepsilon)} f_\sigma(x,\varepsilon) = \Big(\nabla_x f_\sigma(x,\varepsilon), \frac{\partial f_\sigma(x,\varepsilon)}{\partial\varepsilon}\Big),$$

where the derivation of $f_\sigma(x,\varepsilon)$ with respect to x,

$$\begin{aligned}
\nabla_x f_\sigma(x,\varepsilon) &= \nabla f(x) + \varepsilon^{-\alpha}\big(\tfrac{\partial_x\Delta(x,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)} + \Delta(x,\varepsilon)\tfrac{\varepsilon^{-2\delta}\partial_x\Delta(x,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon))^2}\big)\\
&= \nabla f(x) + \varepsilon^{-\alpha}\tfrac{\partial_x\Delta(x,\varepsilon)(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon))+\Delta(x,\varepsilon)\varepsilon^{-2\delta}\partial_x\Delta(x,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon))^2}\\
&= \nabla f(x) + \varepsilon^{-\alpha}\tfrac{\partial_x\Delta(x,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon))^2}\\
&= \nabla f(x) + \tfrac{2\varepsilon^{-\alpha}\int_V \max(0,g(x,v)-\varepsilon^\gamma w)\frac{\partial g(x,v)}{\partial x}d\mu(v)}{(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon))^2},
\end{aligned} \tag{4.2.3}$$

and the derivation of $f_\sigma(x,\varepsilon)$ with respect to ε,

$$\begin{aligned}
&\frac{\partial f_\sigma(x,\varepsilon)}{\partial\varepsilon}\\
&= -\alpha\varepsilon^{-\alpha-1}\tfrac{\Delta(x,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)} + \varepsilon^{-\alpha}\Big(\tfrac{\partial_\varepsilon\Delta(x,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)} + \tfrac{\Delta(x,\varepsilon)(-2\delta\varepsilon^{-2\delta-1}\Delta(x,\varepsilon)+\varepsilon^{-2\delta}\partial_\varepsilon\Delta(x,\varepsilon))}{(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon))^2}\Big)\\
&\quad +\sigma\beta\varepsilon^{\beta-1}\\
&= -\alpha\varepsilon^{-\alpha-1}\tfrac{\Delta(x,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)} + \tfrac{\varepsilon^{-\alpha}(\partial_\varepsilon\Delta(x,\varepsilon)(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)))-2\delta\varepsilon^{-2\delta-1}\Delta^2(x,\varepsilon)+\varepsilon^{-2\delta}\partial_\varepsilon\Delta(x,\varepsilon)\Delta(x,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon))^2}\\
&\quad +\sigma\beta\varepsilon^{\beta-1}\\
&= -\alpha\varepsilon^{-\alpha-1}\tfrac{\Delta(x,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)} + \varepsilon^{-\alpha}\tfrac{\partial_\varepsilon\Delta(x,\varepsilon)-2\delta\varepsilon^{-2\delta-1}\Delta^2(x,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon))^2} + \sigma\beta\varepsilon^{\beta-1}\\
&= \tfrac{\varepsilon^{-\alpha-1}}{(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon))^2}[-\alpha\Delta(x,\varepsilon)(1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)) - 2\gamma\varepsilon^\gamma w\textstyle\int_V \max(0,g(x,v)-\varepsilon^\gamma w)d\mu(v)\\
&\quad -2\delta\varepsilon^{-2\delta}\Delta^2(x,\varepsilon)] + \sigma\beta\varepsilon^{\beta-1}.
\end{aligned} \tag{4.2.4}$$

Combining with (4.2.2), (4.2.3) and (4.2.4) yields $\lim_{\substack{\varepsilon\to\varepsilon^\star=0\\ x\to x^\star\in S}} \nabla_x f_\sigma(x,\varepsilon) = \nabla f(x^\star)$ and $\lim_{\substack{\varepsilon\to\varepsilon^\star=0\\ x\to x^\star\in S}} \frac{\partial f_\sigma(x,\varepsilon)}{\partial\varepsilon} = 0$. This yields the conclusion as desired. ■

In the next section, we proceed to introduce the penalty function algorithm and present the corresponding convergence analysis.

4.3 Algorithm

Step 1. Choose $\tilde{\varepsilon}, \bar{\varepsilon} > 0$, $\eta > 0$ arbitrarily small, $\sigma_0 > 0$, $\rho > 0$ and $(x_0,\varepsilon_0) \in \mathbb{R}^n \times (0,\bar{\varepsilon})$, set $k := 0$.

Step 2. For the semi-infinite programming, we construct the following penalty function

$$f_\sigma(x,\varepsilon)=\begin{cases} f(x), & \text{if } \varepsilon=0, x\in S,\\ f(x)+\frac{\varepsilon^{-\alpha}\Delta(x,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\varepsilon)}+\sigma\varepsilon^{\beta}, & \text{if } \varepsilon>0, 0<1-2\varepsilon^{-2\delta}\Delta(x,\varepsilon)<1,\\ +\infty, & \text{otherwise,}\end{cases}$$

where $\Delta(x,\varepsilon)=\int_V(\max(0,g(x,v)-\varepsilon^{\gamma}w))^2d\mu(v)=\int_{V^+(x,\varepsilon)}(g(x,v)-\varepsilon^{\gamma}w)^2d\mu(v), w\in(0,1), \alpha,\beta,\gamma,\delta$ satisfy (4.2.2). Compute (x_k,ε_k) such that

$$(x_k,\varepsilon_k)\in \underset{(x,\varepsilon)\in\mathbb{R}^n\times(0,\bar{\varepsilon})}{\operatorname{argmin}} f_{\sigma_k}(x,\varepsilon)$$

and denote the solution (x_k,ε_k) of the penalty problem (Q_{σ_k}).

Step 3. If $|\varepsilon_k|\le\tilde{\varepsilon}, \|\nabla_{(x,\varepsilon)}f_{\sigma_k}(x_k,\varepsilon_k)\|\le\eta$, then stop. The point obtained x_k is an approximation solution of (Q). Otherwise, choose $\sigma_{k+1}=\sigma_k+\rho$.

Step 4. Set $k:=k+1$ and return to Step 2.

Here, it should be mentioned that, the unconstrained optimization problem in Step 2 can be solved by some unconstrained optimization methods, for instance, trust-region method, newton method and conjugate gradient method. Subsequently, the convergence properties analysis are presented.

Lemma 4.3.1 *If $(x_k,\varepsilon_k)\in L(Q_{\sigma_k})$ generated by the algorithm with finite $f_{\sigma_k}(x_k,\varepsilon_k)$ and $\varepsilon_k>0$, then $(x_k,\varepsilon_k)\notin S_{\varepsilon_k}$.*

Proof. By $(x_k,\varepsilon_k)\in L(Q_{\sigma_k})$ with finite $f_{\sigma_k}(x_k,\varepsilon_k)$, $\varepsilon_k>0$, then $\frac{\partial f_\sigma(x_k,\varepsilon_k)}{\partial\varepsilon}=0$, we have

$$\begin{aligned}&\frac{\partial f_\sigma(x_k,\varepsilon_k)}{\partial\varepsilon}\\ =\ &\frac{\varepsilon_k^{-\alpha-1}}{(1-\varepsilon_k^{-2\delta}\Delta(x_k,\varepsilon_k))^2}[-\alpha\Delta(x_k,\varepsilon_k)(1-\varepsilon_k^{-2\delta}\Delta(x_k,\varepsilon_k))\\ &-2\gamma\varepsilon_k^{\gamma}w\int_V\max(0,g(x_k,v)-\varepsilon_k^{\gamma}w)d\mu(v)-2\delta\varepsilon_k^{-2\delta}\Delta^2(x_k,\varepsilon_k)]+\sigma_k\beta\varepsilon_k^{\beta-1}\\ =\ &0.\end{aligned}\tag{4.3.5}$$

If $(x_k,\varepsilon_k)\in S_{\varepsilon_k}$, then the left hand of the above equals to $\sigma_k\beta\varepsilon_k^{\beta-1}>0$. This is a contradiction. Hence $(x_k,\varepsilon_k)\notin S_{\varepsilon_k}$. ■

Remark 4.3.1 *In fact, the following result is true: if $\nabla_{(x,\varepsilon)}f_{\sigma_k}(x_k,\varepsilon_k)=0$ with finite $f_{\sigma_k}(x_k,\varepsilon_k)$ and $\varepsilon_k>0$, then $(x_k,\varepsilon_k)\notin S_{\varepsilon_k}$.*

Now we introduce the definition of the extended Mangasarian-Fromovitz constraint qualifications for the semi-infinite programming problem.

Definition 4.3.1 *([40]) It is said that the extended Mangasarian-Fromovitz constraint qualifications holds at $(x^\star, v)$, for all $v \in V^+(x^\star, \varepsilon^\star)$, if there exists a vector $h \in \mathbb{R}^n$ such that*

$$\nabla_x g(x^\star, v)^\top h < 0, \forall\, v \in V^+(x^\star, \varepsilon^\star).$$

For convenience, we refer to the notation "$(x_k, \varepsilon_k) \xrightarrow{k} (x^\star, \varepsilon^\star)$" as when k sufficiently large, the point sequence (x_k, ε_k) tends to $(x^\star, \varepsilon^\star)$.

Lemma 4.3.2 *If $(x_k, \varepsilon_k) \in L(Q_{\sigma_k})$ generated by the algorithm with finite $f_{\sigma_k}(x_k, \varepsilon_k)$ and $\varepsilon_k > 0$, $(x_k, \varepsilon_k) \xrightarrow{k} (x^\star, \varepsilon^\star)$ and $\nabla_x g(x^\star, v), \forall\, v \in V^+(x^\star, \varepsilon^\star)$ satisfying the extended Mangasarian-Fromovitz constraint qualifications, then $\varepsilon^\star = 0, x^\star \in S$.*

Proof. As the preceding discussion, denote

$$V^+(x_k, \varepsilon_k) = \{v \in V | g(x_k, v) \geq \varepsilon_k^\gamma w\}.$$

We first show that $\varepsilon^\star = 0$. By Lemma 4.3.1, since $\varepsilon_k > 0$, we have $(x_k, \varepsilon_k) \notin S_{\varepsilon_k}$. Therefore, $V^+(x_k, \varepsilon_k) \neq \emptyset$. From $(x_k, \varepsilon_k) \in L(Q_{\sigma_k})$, one has

$$\nabla_x f_{\sigma_k}(x_k, \varepsilon_k) = \nabla f(x_k) + \frac{2\varepsilon_k^{-\alpha} \int_{V^+(x_k,\varepsilon_k)} (g(x_k, v) - \varepsilon_k^\gamma w) \nabla_x g(x_k, v) d\mu(v)}{(1 - \varepsilon_k^{-2\delta} \Delta(x_k, \varepsilon_k))^2} = 0, \quad (4.3.6)$$

and

$$\begin{aligned} &\frac{\partial f_{\sigma_k}(x_k, \varepsilon_k)}{\partial \varepsilon} \\ = \; &\frac{\varepsilon_k^{-\alpha-1}}{(1-\varepsilon_k^{-2\delta}\Delta(x_k,\varepsilon_k))^2} \Big[-\alpha \Delta(x_k, \varepsilon_k)(1 - \varepsilon_k^{-2\delta} \Delta(x_k, \varepsilon_k)) \\ &-2\gamma \varepsilon_k^\gamma w \int_{V^+(x_k,\varepsilon_k)} (g(x_k, v) - \varepsilon_k^\gamma w) d\mu(v) - 2\delta \varepsilon_k^{-2\delta} \Delta^2(x_k, \varepsilon_k) \Big] + \sigma_k \beta \varepsilon_k^{\beta-1} \\ = \; &0. \end{aligned} \quad (4.3.7)$$

Rearranging (4.3.7), we have

$$\begin{aligned} &-\alpha \varepsilon_k^{2\delta} \Delta(x_k, \varepsilon_k)(1 - \varepsilon_k^{-2\delta} \Delta(x_k, \varepsilon_k)) - 2\gamma \varepsilon_k^{\gamma+2\delta} w \int_{V^+(x_k,\varepsilon_k)} (g(x_k, v) - \varepsilon_k^\gamma w) d\mu(v) \\ &-2\delta \Delta^2(x_k, \varepsilon_k) + (1 - \varepsilon_k^{-2\delta} \Delta(x_k, \varepsilon_k))^2 \sigma_k \beta \varepsilon_k^{\alpha+\beta+2\delta} = 0. \end{aligned} \quad (4.3.8)$$

Taking $\sigma_k \to +\infty$, by (4.3.8), the first term to the third term tend to finite. From the construction of the penalty function $f_\sigma(x, \varepsilon)$, one has

$$\lim_{k\to+\infty} 1 - \varepsilon_k^{-2\delta}\Delta(x_k, \varepsilon_k) \neq 0. \tag{4.3.9}$$

It holds that $\lim_{k\to+\infty} \varepsilon_k = \varepsilon^\star = 0$. We proceed to prove $x^\star \in S$. Together with (4.3.6), we can obtain

$$\varepsilon_k^\alpha(1 - \varepsilon_k^{-2\delta}\Delta(x_k, \varepsilon_k))^2\nabla f(x_k) + 2\int_{V^+(x_k,\varepsilon_k)} (g(x_k, v) - \varepsilon_k^\gamma w)\nabla_x g(x_k, v)d\mu(v) = 0. \tag{4.3.10}$$

Let $k \to +\infty$. Clearly,

$$\int_{V^+(x^\star,\varepsilon^\star)} (g(x^\star, v) - (\varepsilon^\star)^\gamma w)\nabla_x g(x^\star, v)d\mu(v) = 0. \tag{4.3.11}$$

From the condition that $\nabla_x g(x^\star, v)$ satisfy the extended Mangasarian-Fromovitz constraint qualifications, for all $v \in V^+(x^\star, \varepsilon^\star)$, and hence $g(x^\star, v) = (\varepsilon^\star)^\gamma w$. It is obvious that

$$\Delta(x^\star, \varepsilon^\star) = \int_V (\max(0, g(x^\star, v)-(\varepsilon^\star)^\gamma w))^2 d\mu(v) = \int_{V^+(x^\star,\varepsilon^\star)} (g(x^\star, v)-(\varepsilon^\star)^\gamma w)^2 d\mu(v) = 0. \tag{4.3.12}$$

Thus, for all $v \in V$, we obtain

$$g(x^\star, v) \leq (\varepsilon^\star)^\gamma w_j = 0,$$

i.e., $x^\star \in S$. The proof is completed. ■

Theorem 4.3.1 *Suppose that $(x_k, \varepsilon_k) \in L(Q_{\sigma_k})$ generated by the algorithm with finite $f_{\sigma_k}(x_k, \varepsilon_k)$. For any accumulation point $(x^\star, \varepsilon^\star)$, $\nabla_x g(x^\star, v)$ satisfy the extended Mangasarian-Fromovitz constraint qualification for all $v \in V^+(x^\star, \varepsilon^\star)$, then $x^\star$ is a local optimal solution of (Q).*

Proof. From the conditions of Theorem 4.3.1, we know there exists subsequence $\{(x_k, \varepsilon_k)\}_\aleph$ such that $(x_k, \varepsilon_k) \xrightarrow{k} (x^\star, \varepsilon^\star)$, It then follows from Lemma 4.3.2 that $\varepsilon^\star = 0$ and $x^\star$ is feasible point of the problem (Q). Therefore, there exists a neighbor $o(x^\star, 0)$ and consider an arbitrary point $(x, 0) \in o(x^\star, 0) \cap (S \times \{0\})$, by the definition of (x_k, ε_k), one has

$$f(x^\star) = f_\sigma(x^\star, 0) \leq f_\sigma(x, 0) = f(x).$$

Therefore, $x^\star$ is a local optimal solution of (Q). The proof is completed. ■

Corollary 4.3.1 *Suppose that every local minimizer* $(x^\star, \varepsilon^\star)$ *of the penalty problem* (Q_σ) *with finite* $f_\sigma(x^\star, \varepsilon^\star)$ *and* $\nabla_x g(x^\star, v)$ *satisfy the extended Mangasarian-Fromovitz constraint qualification for all* $v \in V^+(x^\star, \varepsilon^\star)$*, then* $x^\star$ *is local minimizer of the primal problem* (Q) *if and only if* $\varepsilon^\star = 0$*.*

Proof. If $x^\star$ is local minimizer of the primal problem (Q), then $g(x^\star, v) \le 0$ for all $v \in V$, $\Delta(x^\star, \varepsilon) = 0$ for any $\varepsilon \in \mathbb{R}$. Using proof by contradiction, from Lemma 4.3.2, we have $\varepsilon^\star = 0$. Alternatively, if $\varepsilon^\star = 0$, in view of the construction of $f_\sigma(x, \varepsilon)$, $g(x^\star, v) \le 0$ holds for all $v \in V$. $x^\star$ is a feasible point of (Q). From the hypothesis that $(x^\star, 0)$ is optimal solution of (Q_σ), $x^\star$ is a local minimizer of the primal problem (Q). ■

Remark 4.3.2 *Corollary 4.3.1 demonstrates another advantage of this penalty function is that* ε *can be regarded as an indicator variable of a local (global) minimizer. In other words, under fairly general conditions,* $\varepsilon^\star = 0$ *is equivalent to* $x^\star$ *is an optimal solution of* (Q)*.*

The next theorem explores that the exactness property of the penalty function $f_\sigma(x, \varepsilon)$. Through this conclusion, the optimal solutions of primal problem (Q) can be achieved within finite steps.

Theorem 4.3.2 *If* $(x_k, \varepsilon_k) \in L(Q_{\sigma_k})$ *generated by the algorithm with finite* $f_{\sigma_k}(x_k, \varepsilon_k)$*,* $(x_k, \varepsilon_k) \xrightarrow{k} (x^\star, \varepsilon^\star)$ *and* $\nabla_x g(x^\star, v), \forall\, v \in V^+(x^\star, \varepsilon^\star)$ *satisfy the extended Mangasarian-Fromovitz constraint qualifications.* $\alpha, \beta, \gamma, \delta$ *satisfy*

$$-\alpha - \beta + 2\delta \ge 0 \quad \textit{and} \quad \gamma > \delta, \tag{4.3.13}$$

then there exists $k_0 > 0$*, when* $k \ge k_0$*, we have* $\varepsilon_k = 0,\ x_k \in L(P)$*.*

Proof. We prove this theorem by contradiction. Assume the theorem is not true, then there exists a subsequence $\{(x_{n_k}, \varepsilon_{n_k})\}_{\aleph} \subseteq \{(x_k, \varepsilon_k)\}$ such that for any $k_0 > 0$, when $n_k \ge k_0, (x_{n_k}, \varepsilon_{n_k}) \in L(Q_{n_k})$ with finite $f_{\sigma_{n_k}}(x_{n_k}, \varepsilon_{n_k})$ and $\varepsilon_{n_k} > 0$ and the conditions of Theorem 4.3.2 hold for such subsequence. From the statement of Lemma 4.3.1,

$(x_{n_k}, \varepsilon_{n_k}) \notin S_{\varepsilon_{n_k}}$ holds. From (4.2.4), we know

$$\begin{aligned}
&\frac{\partial f_{\sigma_{n_k}}(x_{n_k}, \varepsilon_{n_k})}{\partial \varepsilon} \\
&= \frac{\varepsilon_{n_k}^{-\alpha-\beta}}{(1-\varepsilon_{n_k}^{-2\delta}\Delta(x_{n_k},\varepsilon_{n_k}))^2}[-\alpha\Delta(x_{n_k}, \varepsilon_{n_k})(1 - \varepsilon_{n_k}^{-2\delta}\Delta(x_{n_k}, \varepsilon_{n_k})) - 2\gamma\varepsilon_{n_k}^{\gamma}w \\
&\quad \int_{V^+(x_{n_k},\varepsilon_{n_k})}(g(x_{n_k}, v) - \varepsilon_{n_k}^{\gamma}w)d\mu(v) - 2\delta\varepsilon_{n_k}^{-2\delta}\Delta^2(x_{n_k}, \varepsilon_{n_k})] + \sigma_{n_k}\beta \\
&= 0.
\end{aligned} \tag{4.3.14}$$

From Lemma 4.3.2, we derive $\varepsilon_{n_k} \to \varepsilon^\star = 0$, $x_{n_k} \to x^\star \in S$. Combing with $\varepsilon_{n_k} > 0, 0 < 1 - 2\varepsilon_{n_k}^{-2\delta}\Delta(x_{n_k}, \varepsilon_{n_k}) < 1$, we have $\lim\limits_{\substack{\varepsilon_{n_k} \to \varepsilon^\star = 0 \\ x_{n_k} \to x^\star \in S}} 1 - \varepsilon_{n_k}^{-2\delta}\Delta(x_{n_k}, \varepsilon_{n_k}) = c^\star \in [\frac{1}{2}, 1]$.

Let $-\alpha - \beta + 2\delta \geq 0, -\alpha - \beta + \delta + \gamma \geq 0$. The first term of (4.3.14) tends to 0 and the second term tends to infinity, which leads to contradiction. It implies that such subsequence cannot exist. Therefore, there exists $k_0 > 0$, when $k \geq k_0, \varepsilon_k = 0$, $(x_k, 0) \in L(Q_{\sigma_k})$. Thus, by $(x_k, 0) \in L(Q_{\sigma_k})$, there exists a neighbor $o(x_k, 0)$ at $(x_k, 0), \sigma_k > 0$, for all $(x, 0) \in o((x_k, 0), \sigma_k) \cap (S \times \{0\})$, it holds

$$f(x_k) = f_{\sigma_k}(x_k, 0) \leq f_{\sigma_k}(x, 0) = f(x).$$

Thus, $x_k \in L(Q)$. The proof is completed. ■

4.4 Local Exactness Property

As previously stated, since the penalty function approach is an attempt to solve a constrained problem by minimization of an unconstrained minimization problem, it is of great interest the study of converse properties, which ensure that local (global) minimum points of the constrained problem are also local (global) solutions of the penalty function. Therefore, in this section, we shall show that, under a regular condition, $(x^\star, 0)$ is a local optimal solution of penalty problem (Q_σ) if $x^\star$ is a local minimizer of the original problem (Q).

Now, we first introduce the conception of the error bound of semi-infinite programming and sufficient conditions for the existence of error bound for semi-infinite programming. The semi-infinite programming problem can be formulated as a cone constrained problem. As before, the feasible set S of the semi-infinite programming problem is

$$S = \{x \in \mathbb{R}^n : g(x, v) \leq 0 \ \ \forall\, v \in V\}.$$

The distance function of $g(x, v)$ to $\mathbb{R}_-$ can be described as

$$\mathrm{dist}(g(x, v) \,|\, \mathbb{R}_-) = \|g(x, v) - g(x, v)^-\|_V = \|g(x, v)^+\|_V = \max_{v \in V} g(x, v)^+.$$

Thus, the error bound for semi-infinite programming can be described as below: there exists a positive number $k > 0$ such that

$$\mathrm{dist}(x \,|\, S) \leq k \max_{v \in V} g(x, v)^+.$$

We can regard the error bound condition as a regularity condition. The definition of regularity for semi-infinite programming can be given as follows.

Definition 4.4.1 *For a system*

$$\{x \in \mathbb{R}^n : g(x, v) \leq 0 \ \ \forall\, v \in V\},$$

it is said that the error bound condition holds at a solution $x^\star$ if there exist positive constants $k > 0$ and $\delta > 0$ such that

$$\mathrm{dist}(x \,|\, S) \leq k \max_{v \in V} g(x, v)^+$$

for all $x \in x^\star + \delta\mathbb{B}$, where $S = \{x \in \mathbb{R}^n : g(x, v) \leq 0, \forall\, v \in V\}$ and $\mathbb{B}$ is unit closed ball in $\mathbb{R}^n$.

In what follows, we present the conditions that guarantee the error bound condition. By [62], we represent

$$h(x) = \max_{v \in V} g(x, v)^+, \quad T(x) = \operatorname*{argmax}_{v \in V} g(x, v)^+$$

and

$$\partial h(x) = conv\{\nabla_x g(x, \bar{v}) | \bar{v} \in T(x)\}$$

From Theorem 2.2 of [73], we know if there exists $0 < \delta, \mu < \infty$, for all $\xi \in \partial h(x)$, $x \in B(x^\star, \delta)$ and each $\|\xi\| \geq \mu^{-1}$, then $S = \{x \in \mathbb{R}^n | g(x, v) \leq 0, \forall v \in V\}$ is nonempty and the error bound condition for semi-infinite programming holds. Furthermore, we make the following assumption.

(H_1) $f(x)$ is Lipschitz continuous with Lipschitz constant L.

Combining with [51, Theorem 3] (Lemma 2.1.2), we obtain the following conclusion.

Lemma 4.4.1 *If (H_1) and error bound conditions for semi-infinite programming hold, there exist a neighborhood N_0 of $x^\star$, and a constant $\tau > 0$ (in fact, $\tau \geq kL$)such that*

$$f(x) \geq f(x^\star) - \tau \max_{v \in V} g(x, v)^+.$$

Now we present an important theoretical result for the local exactness. Before proving this result, one more assumption is given as follows.

(H_2) δ, β, γ are positive numbers and satisfy $\delta \geq \beta$ and $\gamma \geq \beta$.

Based on the above hypothesis, we can prove the main results in this section.

Theorem 4.4.1 *Suppose the above assumptions of Lemma 4.4.1 and (H_2) hold, for sufficiently large σ, there are a neighborhood $N \subseteq N_0$ of $x^\star$ and sufficiently small $0 < \varepsilon^{'} << 1$ such that*

$$f_\sigma(x, \varepsilon) > f_\sigma(x^\star, 0) = f(x^\star) \quad \text{for all} \quad (x, \varepsilon) \in N \times (0, \varepsilon^{'}].$$

In particular, $(x^\star, 0)$ is a local minimizer of $f_\sigma(x, \varepsilon)$.

Proof. Let the neighborhood $N \subseteq N_0$ of $x^\star$ be sufficiently small such that

$$\sup_{x \in N}\{f(x^\star) - f(x)\} \leq 1,$$

and assume that the penalty parameter

$$\sigma \geq \tau(R_0 + 2\|w\|).$$

We divide into two cases for further analysis.

(i) $\Delta(x, \varepsilon) \geq \varepsilon^{2\delta}$;

(ii) $\Delta(x,\varepsilon) < \varepsilon^{2\delta}$,

for $x \in N, \varepsilon \in (0, \varepsilon^{'}]$.

Case (i). By the definition of penalty function, $f_\sigma(x,\varepsilon) = +\infty$. Therefore, $f_\sigma(x,\varepsilon) > f_\sigma(x^\star, 0)$.

We proceed to analyze case (ii). For case (ii), $\Delta(x,\varepsilon) < \varepsilon^{2\delta}$, we denote

$$(\max_{v\in V^+(x,\varepsilon)}(g(x,v) - \varepsilon^\gamma w))^2 = \|g(x,v) - \varepsilon^\gamma w\|^2_{L^\infty(V)}$$

and

$$\int_{V^+(x,\varepsilon)}(g(x,v) - \varepsilon^\gamma w)^2 d\mu(v) = \|g(x,v) - \varepsilon^\gamma w\|^2_{L^2(V)}.$$

By interpolation theorem of Riesz-Thorin [16], this yields

$$\frac{1}{R_0^2}(\max_{v\in V^+(x,\varepsilon)}(g(x,v) - \varepsilon^\gamma w))^2 \le \int_{V^+(x,\varepsilon)}(g(x,v) - \varepsilon^\gamma w)^2 d\mu(v) < \varepsilon^{2\delta},$$

where $I : L^2(V) \to L^\infty(V)$ is an identity functional operator and $\|I\|_{L^2(V)\to L^\infty(V)} = R_0 > 0$. Furthermore,

$$\max_{v\in V^+(x,\varepsilon)}(g(x,v) - \varepsilon^\gamma w) < R_0\varepsilon^\delta.$$

We note that

$$\max_{v\in V^+(x,\varepsilon)} g(x,v) - \varepsilon^\gamma\|w\| \le \max_{v\in V^+(x,\varepsilon)}(g(x,v) - \varepsilon^\gamma w).$$

Thus,

$$\max_{v\in V^+(x,\varepsilon)} g(x,v) < R_0\varepsilon^\delta + \varepsilon^\gamma\|w\|.$$

It is worth noting the fact

$$|\max_{v\in V^+(x,\varepsilon)} g(x,v) - \max_{v\in V^+(x,0)} g(x,v)| \le \varepsilon^\gamma\|w\|.$$

This assertion can be easily verified as follows.

- When there exists at least a $(x,v) \in \mathbb{R}^{n+m}$ such that $g(x,v) \ge \varepsilon^\gamma w$, that is, $V^+(x,\varepsilon)$ is nonempty, then the $\operatorname{argmax}_{v\in V} g(x,v)$ belongs to $V^+(x,\varepsilon)$ from the definitions of $V^+(x,\varepsilon)$ and $V^+(x,0)$ and the fact $V^+(x,\varepsilon) \subseteq V^+(x,0)$. Therefore $\max_{v\in V^+(x,\varepsilon)} g(x,v)$ and $\max_{v\in V^+(x,0)} g(x,v)$ have the same value;

- Suppose for any $(x,v) \in \mathbb{R}^{n+m}$, $g(x,v) < \varepsilon^\gamma w$, i.e., $V^+(x,\varepsilon) = \emptyset$, if $V^+(x,0) = \emptyset$, therefore, x is feasible point. For this case, we reformulate as $\max\limits_{v \in V^+(x,\varepsilon)} g(x,v) = \max\limits_{v \in V^+(x,0)} g(x,v) = 0$. If $V^+(x,0) \neq \emptyset$, the maximum difference between $\max\limits_{v \in V^+(x,\varepsilon)} g(x,v)$ and $\max\limits_{v \in V^+(x,0)} g(x,v)$ at most is $\varepsilon^\gamma \|w\|$.

This yields the assertion as desired. Therefore, it implies

$$\max_{v \in V^+(x,0)} g(x,v) \leq \max_{v \in V^+(x,\varepsilon)} g(x,v) + \varepsilon^\gamma \|w\|.$$

Furthermore, combining with Lemma 4.4.1 and assumption (H_2), one has

$$\begin{aligned} f(x^\star) \leq & \ f(x) + \tau \max_{v \in V^+(x,0)} g(x,v) \\ \leq & \ f(x) + \tau(\max_{v \in V^+(x,\varepsilon)} g(x,v) + \varepsilon^\gamma \|w\|) \\ < & \ f(x) + \tau(R_0 \varepsilon^\delta + \varepsilon^\gamma \|w\|) + \tau \varepsilon^\gamma \|w\| \\ \leq & \ f(x) + \tau(R_0 + 2\|w\|)\varepsilon^\beta \\ \leq & \ f(x) + \sigma \varepsilon^\beta. \end{aligned}$$

Therefore, $f(x^\star) < f(x) + \sigma\varepsilon^\beta \leq f_\sigma(x,\varepsilon)$. This yields the inequality as desired. ■

4.5 Numerical Examples

To give some insight into the behavior of the algorithm presented in this section, we demonstrate the method by a few examples. They are implemented in Matlab 7.8.0 executed on Intel Core 2 CPU 2.39 GHz with 1.99 GB memory. We use $\|\nabla_{(x,\varepsilon)} f_\sigma(x,\varepsilon)\| \leq 10^{-6}$ as stopping criteria. Tables 4.1-4.7 show the computational result for the corresponding problem with the following items:

Iter-numbers of iterations of penalty function algorithm,
σ_k – the penalty parameter,
x_k, ε_k – the final iterate,
$f(x_k)$ – the function value of $f(x)$ at the final x_k,
$\Delta(x_k, \varepsilon_k)$ – the total constraint violation at (x_k, ε_k).

Example 4.5.1

$$\begin{aligned} \min \quad & 1.21\exp(x_1) + \exp(x_2), \\ s.t. \quad & v - \exp(x_1 + x_2) \le 0 \quad \forall\, v \in [0,1], \end{aligned}$$

where, in order to make $\alpha, \beta, \gamma, \delta$ satisfy (4.2.2) and (4.3.13), in this example, the parameters used in this algorithm are set as $\alpha = 0.5, \beta = 2, \gamma = 6, \delta = 3$ and $\rho = 5$. The optimal solution and optimal value are $x^\star = (-0.0953, 0.0953)$ and $f(x^\star) = 2.2000$. We choose initial point $x_0 = (0,0), \varepsilon_0 = 1$. This problem can also be solved by the l_1 exact penalty function in 13 iterations, with a final value of the penalty parameter of 10^4, optimal solution $(-0.0954, 0.0952)$ and optimal value 2.1999. The figure of constraint is depicted in Table 4.1 at the final iteration of penalty function algorithm.

Table 4.1: Numerical results of Example 4.5.1

Iter	σ_k	x_k	ε_k	$f(x_k)$	$\Delta(x_k, \varepsilon_k)$
17	5	(-0.1256, 0.1256)	0.0110	2.2016	1.7838e-012
12	10	(-0.0430, 0.0429)	0.0158	2.2054	1.5537e-011
11	15	(-0.0304, 0.0304)	0.0079	2.2056	2.3998e-013
18	20	(-0.0953, 0.0953)	0.0019	2.2000	4.6960e-017

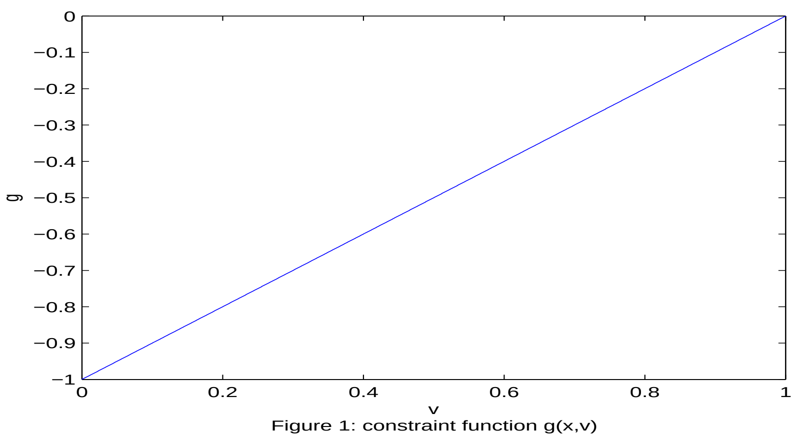

Figure 4.1: Constraint function values of Example 4.5.1

Example 4.5.2

$$\begin{array}{ll} \min & 2.25\exp(x_1) + \exp(x_2), \\ s.t. & v - \exp(x_1 + x_2) \leq 0 \quad \forall\, v \in [0,1], \end{array}$$

where, the parameters used in this algorithm are set as $\alpha = 0.5, \beta = 2, \gamma = 6, \delta = 3$. The optimal solution and optimal value are $x^\star = (-0.4050, 0.4050)$ and $f(x^\star) = 3.0000$. We choose initial point $x_0 = (0,0), \varepsilon_0 = 1$. This problem can also be solved by the l_1 exact penalty function in 14 iterations, with a final value of the penalty parameter of 10^4, optimal solution $(-0.4055, 0.4054)$ and optimal value 2.9999. The figure of constraint is depicted in Table 4.2 at the final iteration of this example.

Table 4.2: Numerical results of Example 4.5.2

Iter	σ_k	x_k	ε_k	$f(x_k)$	$\Delta(x_k, \varepsilon_k)$
16	5	(-0.4000, 0.3998)	0.0202	3.0017	6.9094e-011
30	10	(-0.4004, 0.4004)	0.0117	3.0013	2.5273e-012
19	15	(-0.5450, 0.5430)	0.0413	3.0514	4.9619e-009
16	20	(-0.5614, 0.5591)	0.0434	3.0704	6.7490e-009
19	25	(-0.4053, 0.4053)	0.0008	3.0000	4.7190e-020

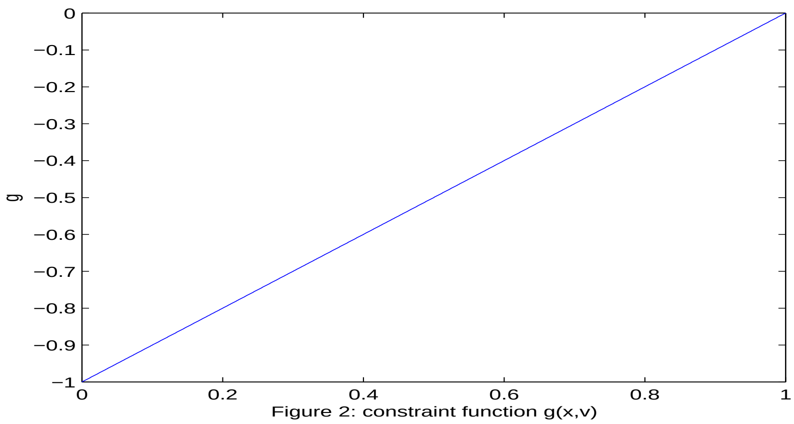

Figure 4.2: Constraint function values of Example 4.5.2

Example 4.5.3

$$\begin{aligned} \min \quad & 2x_1 + x_2, \\ s.t. \quad & -v^2 + v - vx_1 + (v-1)x_2 \leq 0 \quad \forall\, v \in [0,1], \end{aligned}$$

the parameters used in this algorithm are set as $\alpha = 0.5, \beta = 3.2, \gamma = 4, \delta = 3$ and $\rho = 5$. The optimal solution and optimal value are $x^\star = (0.1111, 0.4444)$ and $f(x^\star) = 0.6660$. We choose initial point $x_0 = (0, 0), \varepsilon_0 = 1$. This problem can also be solved by the l_1 exact penalty function in 24 iterations, with a final value of the penalty parameter of 10^3, optimal solution $(0.1111, 0.4445)$ and optimal value 0.6667. The figure of constraint is depicted in Table 4.3 at the final iteration of penalty function algorithm.

Table 4.3: Numerical results of Example 4.5.3

Iter	σ_k	x_k	ε_k	$f(x_k)$	$\Delta(x_k, \varepsilon_k)$
27	5	(0.0115, 0.3448)	0.3736	0.6011	0.0022
50	10	(0.1006, 0.4339)	0.1503	0.6590	1.0982e-005
49	15	(0.1080, 0.4414)	0.0904	0.6644	5.3225e-007
52	20	(0.1098, 0.4432)	0.0631	0.6657	6.2273e-008
52	25	(0.1105, 0.4450)	0.0291	0.6664	6.1025e-010
79	30	(0.1104, 0.4448)	0.0367	0.6664	2.4221e-009

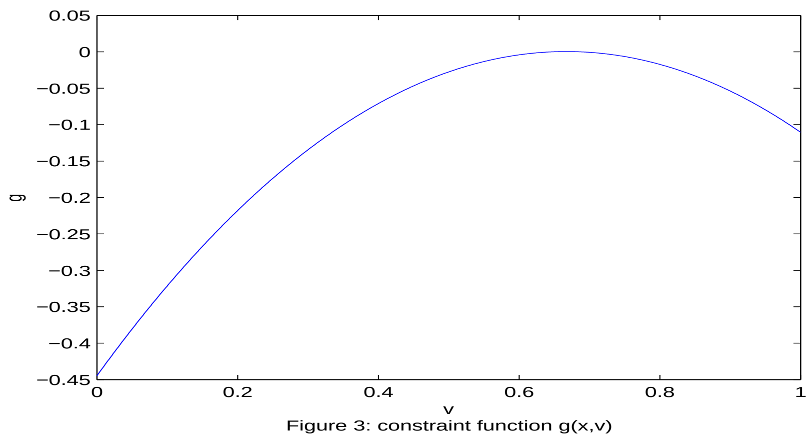

Figure 4.3: Constraint function values of Example 4.5.3

Example 4.5.4

$$\begin{aligned} \min \quad & \frac{1}{2}x^\top x, \\ s.t. \quad & 3 + 4.5\sin\left(\frac{4.7\pi(v-1.23)}{8}\right) - \sum_{i=1}^{n} x_i v^{i-1} \le 0 \quad n = 10 \quad \forall\, v \in [0,1], \end{aligned}$$

the parameters used in this algorithm are set as $\alpha = 0.5, \beta = 2, \gamma = 4, \delta = 3$ and $\rho = 1$. The optimal solution and optimal value are $(0.1147, 0.1147, 0.1147, 0.1147, 0.1147, 0.1147, 0.1147, 0.1147, 0.1147, 0.1147)$ and $f(x^\star) = 0.0657$, respectively. We choose initial point $x_0 = (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), \varepsilon_0 = 0.8$. This problem can also be solved by the l_1 exact penalty function in 12 iterations, with a final value of the penalty parameter of 10^3, the optimal solution and the optimal value are $(0.1147, 0.1147, 0.1147, 0.1146, 0.1146, 0.1146, 0.1146, 0.1146, 0.1146, 0.1146)$ and 0.0657 respectively. The figure of constraint is depicted in Table 4.4 at the final iteration of this example.

Table 4.4: Numerical results of Example 4.5.4

Iter	σ_k	x_k	ε_k	$f(x_k)$	$\Delta(x_k, \varepsilon_k)$
12	1	(0.1138,0.1150,0.1149,0.1148,0.1148,0.1147,0.1147,0.1146,0.1146,0.1146)	0.0016	0.0657	1.9163e-017
22	2	(0.1156,0.1152,0.1150,0.1149,0.1147,0.1145,0.1144,0.1142,0.1141,0.1139)	0.0008	0.0657	2.2739e-019
29	3	(0.1111,0.1150,0.1150,0.1150,0.1150,0.1151,0.1151,0.1151,0.1151,0.1151)	0.0001	0.0657	9.7293e-024
28	4	(0.1145,0.1145,0.1145,0.1146,0.1146,0.1147,0.1147,0.1148,0.1148,0.1149)	0.0000	0.0657	0

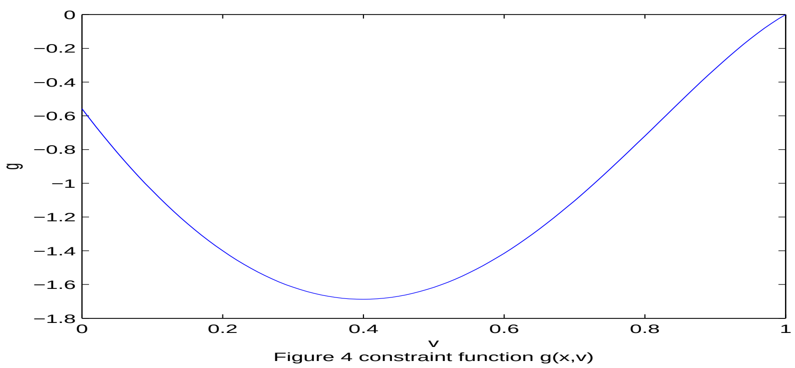

Figure 4.4: Constraint function values of Example 4.5.4

Example 4.5.5

$$\begin{aligned} \min \quad & x_1 + (x_2 - 3)^2, \\ s.t. \quad & x_2 - 2 + x_1 \sin(\frac{v}{x_2} - 0.5) \le 0 \quad \forall v \in [0, 10], \end{aligned}$$

where, the parameters used in this algorithm are set as $\alpha = 0.5, \beta = 2.2, \gamma = 6, \delta = 3$ and $\rho = 5$. The optimal solution and optimal value are $x^\star = (0, 2)$ and $f(x^\star) = 1.0000$. We choose initial point $x_0 = (3, 3), \varepsilon_0 = 1.4$. This problem can also be solved by the l_1 exact penalty function in 14 iterations, with a final value of the penalty parameter of 10, optimal solution $(0.0000, 2.0000)$ and optimal value 1.0000. The figure of constraint is depicted in Table 4.5 at the final iteration of penalty function algorithm.

Table 4.5: Numerical results of Example 4.5.5

Iter	σ_k	x_k	ε_k	$f(x_k)$	$\Delta(x_k, \varepsilon_k)$
37	10	(0.0849, 1.9131)	0.0000	1.2662	0
28	15	(0.0746, 1.9255)	0.0291	1.2355	6.1114e-010
28	20	(0.0304, 1.9696)	0.0107	1.0930	1.5615e-012
39	25	(0.0004, 1.9992)	-0.0000	1.0019	0

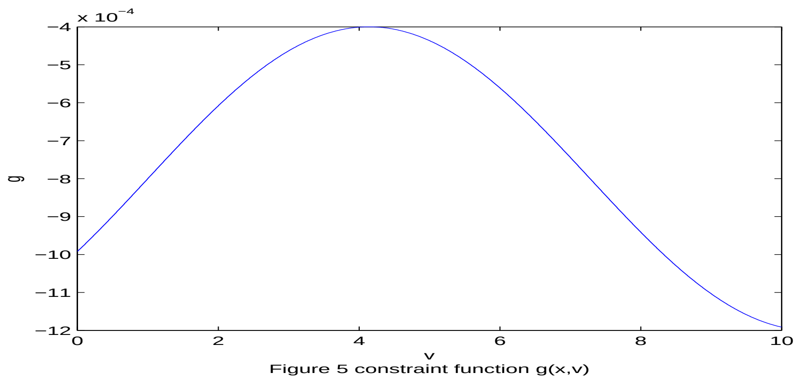

Figure 4.5: Constraint function values of Example 4.5.5

Example 4.5.6

$$\begin{aligned} \min \quad & x_1^2 + x_2^2 + x_3^2, \\ s.t. \quad & x_1(v_1 + v_2^2 + 1) + x_2(v_1 v_2 - v_2^2) + x_3(v_1 v_2 + v_2^2 + v_2) + 1 \leq 0 \qquad \forall v \in [0,1] \times [0,1], \end{aligned}$$

the parameters used in this problem are set as $\alpha = 0.5, \beta = 2, \gamma = 6, \delta = 3$ and $\rho = 2$. The optimal solution and optimal value are $x^\star = (-1, 0, 0)$ and $f(x^\star) = 1.0000$. Choose initial point $x_0 = (2, 3, 2), \varepsilon_0 = 1.6$. This problem can also be solved by the l_1 exact penalty function in 13 iterations, with a final value of the penalty parameter of 10^4, optimal solution $(-0.9998, 0.0023, -0.0053)$ and optimal value 0.9999. The figure of constraint is depicted in Table 4.6 at the final iteration of penalty function algorithm.

Table 4.6: Numerical results of Example 4.5.6

Iter	σ_k	x_k	ε_k	$f(x_k)$	$\Delta(x_k, \varepsilon_k)$
25	2	(-0.7391, 0.0128, -0.1203)	0.3532	0.8266	0.8266
41	4	(-0.9081, 0.0071, -0.0793)	0.1842	0.9687	3.7390e-005
63	6	(-0.9850, 0.0017, -0.0342)	0.0715	1.0022	1.3282e-007
31	8	(-0.9985, 0.1873, -0.0407)	0.0259	1.0392	3.0057e-010
101	10	(-1.0000, -0.0009, 0.0113)	0.0000	1.0001	0

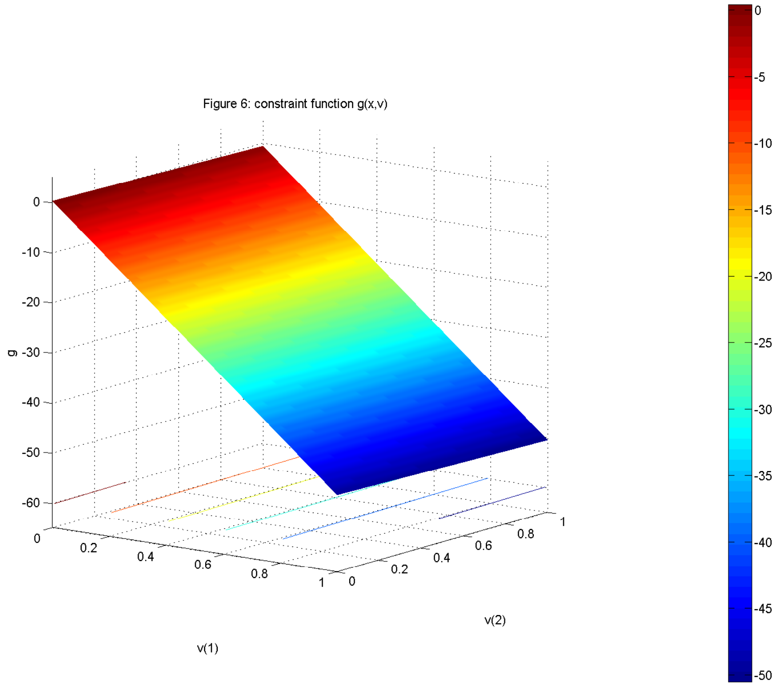

Figure 4.6: Constraint function values of Example 4.5.6

Example 4.5.7

$$\begin{aligned} \min \quad & x_1^2 + x_2^2 + x_3^2, \\ s.t. \quad & x_1 + x_2 \exp(x_3 v_1) + \exp(2v_2) - 2\sin(4v_1) \leq 0 \quad \forall v \in [0,1] \times [0,1], \end{aligned}$$

where, the parameters used in this algorithm are set as $\alpha = 0.5, \beta = 2, \gamma = 6, \delta = 3$ and $\rho = 10$. The optimal solution and optimal value are $(-3.6812, -3.7079, 0.3423)$ and $f(x^\star) = 27.4166$ respectively. Choose initial point $x_0 = (0, 0, 0), \varepsilon_0 = 1.5$. This problem can also be solved by the l_1 exact penalty function in 15 iterations, with a final value of the penalty parameter of 10^4, optimal solution $(-3.5731, -3.6581, 0.3763)$ and optimal value 26.5236. The figure of constraint function is depicted in Table 4.7 at the final iteration of penalty function algorithm.

Numerical outputs verify the correctness of the developed theory as desired. From the above numerical examples, it is observed that we obtain the optimal solution without

Table 4.7: Numerical results of Example 4.5.7

Iter	σ_k	x_k	ε_k	$f(x_k)$	$\Delta(x_k, \varepsilon_k)$
32	10	(-1.5281, -2.0117, 1.1691)	0.7777	14.4864	0.1622
43	20	(-2.1202, -2.4145, 0.9567)	0.6183	19.3066	0.0478
51	30	(-2.5632, -2.7488, 0.7858)	0.4978	22.4085	0.0139
55	40	(-2.8948, -3.0168, 0.6563)	0.3996	24.4160	0.0038
94	50	(-3.1384, -3.2230, 0.5612)	0.3192	25.7039	0.0010
25	60	(-3.6920, -3.6988, 0.3427)	0.0010	27.4293	1.4191e-018

the requirement of a large penalty parameter σ. Tables 4.1-4.7 demonstrate that the minimizers obtained by this new exact penalty function method are all feasible points. The introduced new variable ε is zero or quite close to zero after a finite number of iterations. From the result of Corollary 4.3.1, ε can completely be regarded as an indicator variable for the minimizer. Numerical results show this new smooth and exact penalty function is reasonable and well-behaved.

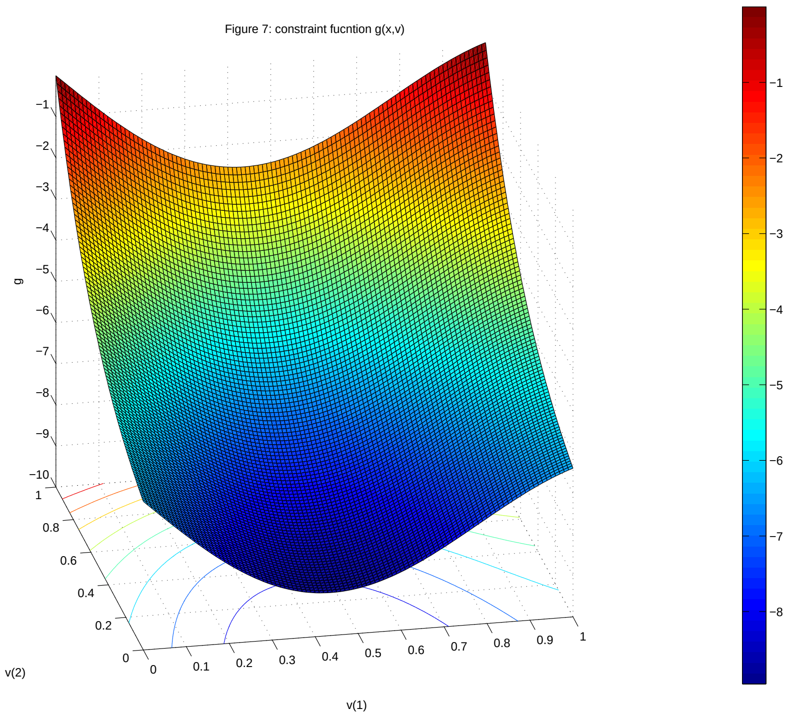

Figure 4.7: Constraint function values of Example 4.5.7

Chapter 5

On a New Exact Penalty Function for Solving Constrained Finite Min-Max Programming Problems

5.1 Introduction

In the context of this chapter, we primarily restrict our attention to the following constrained min-max programming problem with equality constraints

$$(\bar{R}) \qquad \begin{cases} \min & \max\limits_{1\le j\le q} f_j(x) \\ \text{s.t.} & F_j(x)=0 \quad \forall j=q+1,\dots,m, \\ & x\in\mathbb{R}^n, \end{cases}$$

where $f_j:\mathbb{R}^n\to\mathbb{R}, j=1,2,\cdots,q$ and $F_j:\mathbb{R}^n\to\mathbb{R}, j=q+1,\cdots,m$ are continuously differentiable functions.

Since the objective function $f(x)$ contains the max operator, it is continuous but non-differentiable even when $f_j(x), j=1,2,\dots,q$ are all differentiable. Thus, unconstrained optimization algorithms with the use of derivatives cannot be applied directly. There are several algorithms proposed for solving the min-max programming problems. They can be divided into three categories. The problem $(\bar{R})$ can be viewed as a constrained nonsmooth optimization problem. Therefore, we can use the general methods of solving nonsmooth optimization problems, such as subgradient methods, bundle methods and cutting plane methods (see Refs. [26, 56]). Another type of method is to consider the special structure of its nondifferentiability in order to make use of certain smooth optimization methods. These include regularization techniques based on the max function by means of a smooth function (see Refs. [54, 75]). Among them, Ye et al. [75] proposed a smooth trust-region Newton-CG algorithm by an exponential penalty function approximation method. Di Pillo et al. [54] firstly extended the exact penalty approach to the case of constrained min-max programming problems. There are also some other approaches based on the following equivalent nonlinear programming problem through introducing a new variable $\theta \in \mathbb{R}$,

$$(R) \qquad \begin{cases} & \min \theta, \\ \text{s.t.} & f_j(x) \leq \theta \quad \forall j = 1, 2, \cdots, q, \\ & F_j(x) = 0 \quad \forall j = q+1, \cdots, m, \\ & x \in \mathbb{R}^n. \end{cases}$$

For example, Zhou et al. [79] SQP algorithms for solving the min-max programming problems based on the above programming problem (R), advantageously exploiting the special structure. Based on this nonlinear programming formulation (R) and the construction of a continuously differentiable exact barrier penalty function, it was shown that the minimizers are also the optimal solutions to the constrained min-max programming problems for finite values of the penalty parameter.

Motivated by the idea in [38], in this section, we propose a new exact and smooth penalty function for min-max programming problems. The main feature of our penalty function is that we only need to add a single variable ε for whatever constraints. The merit function is considered as a function of x and ε simultaneously which has good smoothness and exactness properties even without involving gradient and Jacobian matrices. It remains bounded below whenever $f(x)$ is bounded below on S, which is not shared by the l_1 exact penalty function. It is well-known that the ill-conditioness introduced by a large penalty parameter may be detrimental. Therefore, for the computational algorithm, we only require the penalty parameter to increase by adding a relatively small constant in order to keep the penalty parameter as small as possible, avoiding ill-conditioness occurring. We present the result that, if a local optimal solution to the penalty problem satisfies the linearly independent constraint qualification, then the minimizer has the expression of $(x^\star, \theta^\star, 0)$. In addition, we derive a quite useful conclusion that the minimizer $(x^\star, \theta^\star, \varepsilon^\star)$ of the penalty problem satisfies $\varepsilon^\star = 0$ if and only if $x^\star$ solves the original problem (R) and $\theta^\star$ is the optimal objective function value. This property demonstrates that the introduced new variable ε can be viewed as an indicator variable of a local (global) minimizer of primal problem (R). Besides the above properties, we provide that the penalty problem possesses exactness property. Furthermore, under the mild conditions, the local exactness proof is shown, where the objective and constraint functions are not necessarily smooth.

The rest of the chapter is organized as follows. In Section 5.2, we introduce a smooth and exact penalty function for constrained min-max programming problems. In Section 5.3, a penalty function algorithm and convergence analysis are presented. In Section 5.4, we discuss local exactness property of this new exact and smooth penalty function. Section 5.5 establishes the second-order sufficient optimality conditions for the local exactness property. Some numerical performances are reported in Section 5.6.

5.2 A New Exact and Smooth Penalty Function

As stated above, we convert the min-max programming problem into the following optimization problem by adding a variable θ:

$$\begin{aligned} \min_{(x,\theta)\in S} \theta, \quad S = \{(x,\theta) \in \mathbb{R}^{n+1} : f_j(x) - \theta \leq 0 \ \ \forall j = 1, 2, \cdots q; \\ F_j(x) = 0 \ \ \forall j = q+1, \cdots m\}. \end{aligned} \tag{5.2.1}$$

This is equivalent to

$$\begin{aligned} \min_{(x,\theta,\varepsilon)\in S_0} \theta, \quad S_0 = \{(x,\theta,\varepsilon) \in \mathbb{R}^{n+2} : f_j(x) - \theta \leq \varepsilon^\gamma w_j \ \ \forall j = 1, 2, \cdots, q; \\ F_j(x) = \varepsilon^\gamma w_j \ \ \forall j = q+1, \cdots, m, \varepsilon = 0\}. \end{aligned}$$

where $w_j \in (0,1), j = 1, 2, \cdots, m$. Likewise, we denote

$$\begin{aligned} S_\varepsilon = \{(x,\theta,\varepsilon) \in \mathbb{R}^{n+2} : f_j(x) - \theta \leq \varepsilon^\gamma w_j \ \ \forall j = 1, 2, \cdots, q; \\ F_j(x) = \varepsilon^\gamma w_j \ \ \forall j = q+1, \cdots, m\}. \end{aligned}$$

It implies that solving the min-max programming problem $(\bar{R})$ is equivalent to solving the problem (R).

We make some assumptions in the following:

(1) There exists the global minimizer of (R), this means that $f(x)$ is bounded below on S;

(2) If $x^\star \in L(R)$, then $L_{x^\star} = \{x \in L(R) : f(x) = f(x^\star)\}$ is a compact set, where $L(R)$ denotes the set of local minimizers of the problem (R).

We construct the penalty function for the min-max programming problem as follows,

$$f_\sigma(x,\theta,\varepsilon) = \begin{cases} \theta, & \text{if } \varepsilon = 0, (x,\theta) \in S, \\ \theta + \frac{\varepsilon^{-\alpha}\Delta(x,\theta,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)} + \sigma\varepsilon^\beta, & \text{if } \varepsilon \neq 0, 0 < 1 - 2\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon) < 1, \\ +\infty, & \text{otherwise,} \end{cases} \tag{5.2.2}$$

where $\Delta(x,\theta,\varepsilon) = \sum_{j=1}^{q}(\max(f_j(x) - \theta - \varepsilon^\gamma w_j, 0))^2 + \sum_{j=q+1}^{m}(F_j(x) - \varepsilon^\gamma w_j)^2$ and $\alpha, \beta, \gamma, \delta$ are positive even numbers. The corresponding penalty problem (R_σ) is

$$(R_\sigma) \qquad \min_{(x,\theta,\varepsilon)\in\mathbb{R}^{n+1}\times(-\bar{\varepsilon},\bar{\varepsilon})} f_\sigma(x,\theta,\varepsilon).$$

For $\varepsilon \neq 0, 0 < 1 - 2\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon) < 1$, we have

$$f_\sigma(x,\theta,\varepsilon) = \theta + \frac{\varepsilon^{-\alpha}\Delta(x,\theta,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)} + \sigma\varepsilon^\beta \geq \theta, \text{if } F_j(x) = 0 \quad \forall j = q+1,\cdots,m.$$

Therefore, $f_\sigma(x,\theta,\varepsilon)$ is bounded below on $\mathbb{R}^{n+1}\times[-\bar{\varepsilon},\bar{\varepsilon}]$ whenever $f_j(x), \forall j = 1,2,\cdots,q$ are bounded below on the set

$$D' = \{x \in \mathbb{R}^n | \|F(x)\| \leq \frac{\sqrt{2}}{2}\bar{\varepsilon}^\delta + \bar{\varepsilon}^\gamma\|w\|\}.$$

This is a reasonable condition since when $f_j(x), j = 1,2,\cdots,q$ are bounded below on the feasible set, $\bar{\varepsilon}$ is small enough. In what follows, we shall show that, under some mild conditions, $f_\sigma(x,\theta,\varepsilon)$ is continuously differentiable with continuous limits on the part of the boundary with finite values.

Proposition 5.2.1 *Let $(x,\theta) \to (x^\star,\theta^\star) \in S, 0 \neq \varepsilon \to \varepsilon^\star = 0$. Suppose that*

$$\begin{cases} \gamma > \delta > \alpha > 0, \\ -\alpha - 1 + 2\delta > 0, \\ \beta > 1, \end{cases} \tag{5.2.3}$$

then

$$\lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ (x,\theta) \to (x^\star,\theta^\star) \in S}} f_\sigma(x,\theta,\varepsilon) = f_\sigma(x^\star,\theta^\star,0) = \theta^\star, \qquad \lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ (x,\theta) \to (x^\star,\theta^\star) \in S}} \nabla_x f_\sigma(x,\theta,\varepsilon) = 0,$$

$$\lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ (x,\theta) \to (x^\star,\theta^\star) \in S}} \frac{\partial f_\sigma(x,\theta,\varepsilon)}{\partial \varepsilon} = 0, \qquad \lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ (x,\theta) \to (x^\star,\theta^\star) \in S}} \frac{\partial f_\sigma(x,\theta,\varepsilon)}{\partial \theta} = 1$$

Proof. From the fact that $\varepsilon \neq 0, 0 < 1 - 2\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon) < 1$, we have $\Delta(x,\theta,\varepsilon) = O(\varepsilon^{2\delta})$, $|f_j(x) - \theta - \varepsilon^\gamma w_j| = O(\varepsilon^\delta) \quad \forall j = 1,2,\cdots,q; |F_j(x) - \varepsilon^\gamma w_j| = O(\varepsilon^\delta) \quad \forall j = q+1,\cdots,m$ and $\lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ (x,\theta) \to (x^\star,\theta^\star) \in S}} 1 - \varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon) = c^\star \in [\frac{1}{2},1]$.

As specified in (5.2.3), we know $2\delta > \alpha$ and $\beta > 1$. This yields

$$\lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ (x,\theta) \to (x^\star,\theta^\star) \in S}} f_\sigma(x,\theta,\varepsilon) = \lim_{\substack{\varepsilon \to \varepsilon^\star = 0 \\ (x,\theta) \to (x^\star,\theta^\star) \in S}} \theta + \frac{\varepsilon^{-\alpha}\Delta(x,\theta,\varepsilon)}{1 - \varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)} + \sigma\varepsilon^\beta = \theta^\star.$$

Note that $f_\sigma(x,\theta,\varepsilon)$ is continuously differentiable when (x,θ,ε) satisfies $\varepsilon \neq 0, 0 < 1 - 2\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon) < 1$, the gradient of $f_\sigma(x,\theta,\varepsilon)$ at (x,θ,ε) is

$$\nabla_{(x,\theta,\varepsilon)} f_\sigma(x,\theta,\varepsilon) = \Big(\nabla_x f_\sigma(x,\theta,\varepsilon), \frac{\partial f_\sigma(x,\theta,\varepsilon)}{\partial \theta}, \frac{\partial f_\sigma(x,\theta,\varepsilon)}{\partial \varepsilon}\Big),$$

where

$$\begin{aligned} \nabla_x f_\sigma(x,\theta,\varepsilon) &= \varepsilon^{-\alpha}\Big(\frac{\partial_x\Delta(x,\theta,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)} + \Delta(x,\theta,\varepsilon)\frac{\varepsilon^{-2\delta}\partial_x\Delta(x,\theta,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2}\Big) \\ &= \varepsilon^{-\alpha}\frac{\partial_x\Delta(x,\theta,\varepsilon)(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))+\Delta(x,\theta,\varepsilon)\varepsilon^{-2\delta}\partial_x\Delta(x,\theta,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2} \\ &= \varepsilon^{-\alpha}\frac{\partial_x\Delta(x,\theta,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2} \\ &= \frac{2\varepsilon^{-\alpha}[\sum_{j=1}^{q}(\max(f_j(x)-\theta-\varepsilon^\gamma w_j,0)\nabla f_j(x) + \sum_{j=q+1}^{m}(F_j(x)-\varepsilon^\gamma w_j)\nabla F_j(x)]}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2}, \end{aligned} \tag{5.2.4}$$

$$\begin{aligned} \frac{\partial f_\sigma(x,\theta,\varepsilon)}{\partial\theta} &= 1 + \varepsilon^{-\alpha}\Big(\frac{\partial_\theta\Delta(x,\theta,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)} + \Delta(x,\theta,\varepsilon)\frac{\varepsilon^{-2\delta}\partial_\theta\Delta(x,\theta,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2}\Big) \\ &= 1 + \varepsilon^{-\alpha}\frac{\partial_\theta\Delta(x,\theta,\varepsilon)(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))+\Delta(x,\theta,\varepsilon)\varepsilon^{-2\delta}\partial_\theta\Delta(x,\theta,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2} \\ &= 1 + \varepsilon^{-\alpha}\frac{\partial_\theta\Delta(x,\theta,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2} \\ &= 1 - \frac{2\varepsilon^{-\alpha}\sum_{j=1}^{q}\max(f_j(x)-\theta-\varepsilon^\gamma w_j,0)}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2} \end{aligned} \tag{5.2.5}$$

and

$$
\begin{aligned}
&\frac{\partial f_\sigma(x,\theta,\varepsilon)}{\partial\varepsilon}\\
= \quad & -\alpha\varepsilon^{-\alpha-1}\frac{\Delta(x,\theta,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)} + \varepsilon^{-\alpha}\left(\frac{\partial_\varepsilon\Delta(x,\theta,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)} + \frac{\Delta(x,\theta,\varepsilon)(-2\delta\varepsilon^{-2\delta-1}\Delta(x,\theta,\varepsilon)+\varepsilon^{-2\delta}\partial_\varepsilon\Delta(x,\theta,\varepsilon))}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2}\right)\\
&+\beta\varepsilon^{\beta-1}\sigma\\
= \quad & -\alpha\varepsilon^{-\alpha-1}\frac{\Delta(x,\theta,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)} + \frac{\varepsilon^{-\alpha}(\partial_\varepsilon\Delta(x,\theta,\varepsilon)(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)))-2\delta\varepsilon^{-2\delta-1}\Delta^2(x,\theta,\varepsilon)+\varepsilon^{-2\delta}\partial_\varepsilon\Delta(x,\theta,\varepsilon)\Delta(x,\theta,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2}\\
&+\beta\varepsilon^{\beta-1}\sigma\\
= \quad & -\alpha\varepsilon^{-\alpha-1}\frac{\Delta(x,\theta,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)} + \varepsilon^{-\alpha}\frac{\partial_\varepsilon\Delta(x,\theta,\varepsilon)-2\delta\varepsilon^{-2\delta-1}\Delta^2(x,\theta,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2} + \beta\varepsilon^{\beta-1}\sigma\\
= \quad & \frac{-\alpha\varepsilon^{-\alpha-1}\Delta(x,\theta,\varepsilon)(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))-2\gamma\varepsilon^{-\alpha+\gamma-1}[\sum\limits_{j=1}^{q}\max(f_j(x)-\theta-\varepsilon^\gamma w_j,0)w_j+\sum\limits_{j=q+1}^{m}(F_j(x)-\varepsilon^\gamma w_j)w_j]}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2}\\
&-\frac{2\delta\varepsilon^{-\alpha-2\delta-1}\Delta^2(x,\theta,\varepsilon)}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2} + \beta\varepsilon^{\beta-1}\sigma\\
= \quad & \frac{\varepsilon^{-\alpha-1}}{(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon))^2}[-\alpha\Delta(x,\theta,\varepsilon)(1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)) - 2\gamma\varepsilon^\gamma(\sum_{j=1}^{q}\max(f_j(x)-\theta-\varepsilon^\gamma w_j,0)w_j\\
&+\sum_{j=q+1}^{m}(F_j(x)-\varepsilon^\gamma w_j)w_j) - 2\delta\varepsilon^{-2\delta}\Delta^2(x,\theta,\varepsilon)] + \beta\varepsilon^{\beta-1}\sigma. \qquad (5.2.6)
\end{aligned}
$$

By (5.2.3), (5.2.4), (5.2.5) and (5.2.6), we have $\lim\limits_{\substack{\varepsilon\to\varepsilon^\star=0,\\(x,\theta)\to(x^\star,\theta^\star)\in S}}\nabla_x f_\sigma(x,\theta,\varepsilon)=0$, $\lim\limits_{\substack{\varepsilon\to\varepsilon^\star=0,\\(x,\theta)\to(x^\star,\theta^\star)\in S}}\frac{\partial f_\sigma(x,\theta,\varepsilon)}{\partial\varepsilon}=0$, $\lim\limits_{\substack{\varepsilon\to\varepsilon^\star=0,\\(x,\theta)\to(x^\star,\theta^\star)\in S}}\frac{\partial f_\sigma(x,\theta,\varepsilon)}{\partial\theta}=1$ for any $\sigma>0$. ■

5.3 Algorithm

Step 1. Choose $\bar\varepsilon,\tilde\varepsilon>0$, $\eta>0$ arbitrarily small, $\sigma>0,\rho>0$ and $(x_0,\theta_0,\varepsilon_0)\in\mathbb{R}^{n+1}\times(-\bar\varepsilon,\bar\varepsilon),\varepsilon_0\neq0$, set $k:=0$.

Step 2. For the problem (R), we construct the simple penalty function

$$
f_\sigma(x,\theta,\varepsilon)=\begin{cases}\theta, & \text{if } \varepsilon=0,(x,\theta)\in S,\\ \theta+\frac{\varepsilon^{-\alpha}\Delta(x,\theta,\varepsilon)}{1-\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)}+\sigma\varepsilon^\beta, & \text{if } \varepsilon\neq0,\quad 0<1-2\varepsilon^{-2\delta}\Delta(x,\theta,\varepsilon)<1,\\ +\infty, & \text{otherwise,}\end{cases}
$$

where

$$
\Delta(x,\theta,\varepsilon)=\sum_{j=1}^{q}(\max(f_j(x)-\theta-\varepsilon^\gamma w_j),0)^2+\sum_{j=q+1}^{m}(F_j(x)-\varepsilon^\gamma w_j)^2, w_j\in(0,1), j=1,2,\cdots,m.
$$

Using any unconstrained optimization algorithms to solve

$$
\min_{(x,\theta,\varepsilon)\in\mathbb{R}^{n+1}\times(-\bar\varepsilon,\bar\varepsilon)} f_\sigma(x,\theta,\varepsilon)
$$

and denote $(x_\sigma, \theta_\sigma, \varepsilon_\sigma)$ the solution to (R_σ).

Step 3. If $|\varepsilon_\sigma| \le \tilde{\varepsilon}$, $\|\nabla_{(x,\theta,\varepsilon)} f_\sigma(x, \theta, \varepsilon)\| \le \eta > 0$, then stop. $(x_\sigma, \theta_\sigma)$ is an approximate solution to (R). Otherwise, $\sigma := \sigma + \rho$, go back to Step 2.

Lemma 5.3.1 *If $(x_k, \theta_k, \varepsilon_k) \in L(R_{\sigma_k})$ with finite $f_{\sigma_k}(x_k, \theta_k, \varepsilon_k)$ and $\varepsilon_k \ne 0$, then $(x_k, \theta_k, \varepsilon_k) \notin S_{\varepsilon_k}$.*

Proof. By $(x_k, \theta_k, \varepsilon_k) \in L(R_{\sigma_k})$ with finite $f_{\sigma_k}(x_k, \theta_k, \varepsilon_k)$, $\varepsilon_k \ne 0$, we have

$$\begin{aligned}
\tfrac{\partial f_\sigma(x_k,\theta_k,\varepsilon_k)}{\partial \varepsilon} \quad = \quad & \tfrac{\varepsilon_k^{-\alpha-1}}{(1-\varepsilon_k^{-2\delta}\Delta(x_k,\theta_k,\varepsilon_k))^2}\Big[- \alpha\Delta(x_k, \theta_k, \varepsilon_k)(1 - \varepsilon_k^{-2\delta}\Delta(x_k, \theta_k, \varepsilon_k)) \\
& -2\gamma\varepsilon_k^{\gamma}\Big(\sum_{j=1}^{q} \max(f_j(x_k) - \theta_k - \varepsilon_k^{\gamma} w_j, 0)w_j + \sum_{j=q+1}^{m}(F_j(x_k) - \varepsilon_k^{\gamma} w_j)w_j\Big) \\
& -2\delta\varepsilon_k^{-2\delta}\Delta^2(x_k, \theta_k, \varepsilon_k)\Big] + \beta\varepsilon_k^{\beta-1}\sigma_k = 0.
\end{aligned}$$

If $(x_k, \theta_k, \varepsilon_k) \in S_{\varepsilon_k}$, then the left-hand side of the above equals to $\sigma_k\beta\varepsilon_k^{\beta-1} \ne 0$. This is a contradiction. Hence $(x_k, \theta_k, \varepsilon_k) \notin S_{\varepsilon_k}$. ■

Remark 5.3.1 *In fact, this result is true: if $\nabla_{(x,\theta,\varepsilon)} f_{\sigma_k}(x_k, \theta_k, \varepsilon_k) = 0$ with finite $f_{\sigma_k}(x_k, \theta_k, \varepsilon_k)$ and $\varepsilon_k \ne 0$, then $(x_k, \theta_k, \varepsilon_k) \notin S_{\varepsilon_k}$.*

Lemma 5.3.2 *If $(x_k, \theta_k, \varepsilon_k) \in L(R_{\sigma_k})$ with finite $f_{\sigma_k}(x_k, \theta_k, \varepsilon_k), \varepsilon_k \ne 0$, $(x_k, \theta_k, \varepsilon_k) \xrightarrow{k} (x^\star, \theta^\star, \varepsilon^\star)$ and $\nabla F_j(x^\star), j = q{+}1, \cdots, m$ are linearly independent, then, $\varepsilon^\star = 0$, $(x^\star, \theta^\star) \in S$.*

Proof. We first show that $\varepsilon^\star = 0$. From $\varepsilon_k \ne 0$, and Lemma 5.3.1, we have $(x_k, \theta_k, \varepsilon_k) \notin S_\varepsilon$. According to the definition of $(x_k, \theta_k, \varepsilon_k) \in L(R_{\sigma_k})$,

$$\nabla_x f_{\sigma_k}(x_k, \theta_k, \varepsilon_k) = \frac{2\varepsilon^{-\alpha}\left[\sum\limits_{j=1}^{q}(\max(f_j(x_k) - \theta_k - \varepsilon_k^{\gamma} w_j, 0)\nabla f_j(x_k) + \sum\limits_{j=q+1}^{m}(F_j(x_k) - \varepsilon_k^{\gamma} w_j)\nabla F_j(x_k)\right]}{(1-\varepsilon_k^{-2\delta}\Delta(x_k,\theta_k,\varepsilon_k))^2} = 0, \tag{5.3.7}$$

$$\frac{\partial f_{\sigma_k}(x_k,\theta_k,\varepsilon_k)}{\partial \theta} = 1 - \frac{2\varepsilon_k^{-\alpha}\sum\limits_{j=1}^{q}\max(f_j(x_k) - \theta_k - \varepsilon_k^{\gamma} w_j, 0)}{(1-\varepsilon_k^{-2\delta}\Delta(x_k,\theta_k,\varepsilon_k))^2} = 0 \tag{5.3.8}$$

and

$$\begin{aligned}
\frac{\partial f_{\sigma_k}(x_k,\theta_k,\varepsilon_k)}{\partial \varepsilon} &= \frac{\varepsilon_k^{-\alpha-1}}{(1-\varepsilon_k^{-2\delta}\Delta(x_k,\theta_k,\varepsilon_k))^2}\Big(-\alpha\Delta(x_k,\theta_k,\varepsilon_k)(1-\varepsilon_k^{-2\delta}\Delta(x_k,\theta_k,\varepsilon_k)) \\
&- 2\gamma\varepsilon_k^{\gamma}(\sum_{j=1}^{q}\max(f_j(x_k)-\theta_k-\varepsilon_k^{\gamma}w_j,0)w_j \\
&+ \sum_{j=q+1}^{m}(F_j(x_k)-\varepsilon_k^{\gamma}w_j)w_j)2\delta\varepsilon_k^{-2\delta}\Delta^2(x_k,\theta_k,\varepsilon_k)\Big) + \beta\varepsilon_k^{\beta-1}\sigma_k \\
&= 0.
\end{aligned} \tag{5.3.9}$$

From (5.3.9), we obtain

$$-\alpha\Delta(x_k,\theta_k,\varepsilon_k)(\varepsilon_k^{2\delta}-\Delta(x_k,\theta_k,\varepsilon_k)) - 2\gamma\varepsilon_k^{2\delta+\gamma}(\sum_{j=1}^{q}\max(f_j(x_k)-\theta_k-\varepsilon_k^{\gamma}w_j,0)w_j$$
$$+\sum_{j=q+1}^{m}(F_j(x_k)-\varepsilon_k^{\gamma}w_j)w_j) - 2\delta\Delta^2(x_k,\theta_k,\varepsilon_k) + \beta(1-\varepsilon_k^{-2\delta}\Delta(x_k,\theta_k,\varepsilon_k))^2\varepsilon_k^{\alpha+\beta+2\delta}\sigma_k = 0. \tag{5.3.10}$$

Let $k \to +\infty$, by (5.3.10), the first term to the third term tend to finite limits. From the construction of the penalty function $f_\sigma(x,\theta,\varepsilon)$, one has

$$\lim_{k\to+\infty} 1-\varepsilon_k^{-2\delta}\Delta(x_k,\theta_k,\varepsilon_k) \neq 0.$$

It holds that

$$\lim_{k\to+\infty}\varepsilon_k = \varepsilon^\star = 0. \tag{5.3.11}$$

We proceed to prove $(x^\star,\theta^\star)\in S$. By (5.3.8), we can obtain

$$\varepsilon_k^{\alpha}(1-\varepsilon_k^{-2\delta}\Delta(x_k,\theta_k,\varepsilon_k))^2 - 2\sum_{j=1}^{q}\max(f_j(x_k)-\theta_k-\varepsilon_k^{\gamma}w_j,0) = 0. \tag{5.3.12}$$

Let $k \to +\infty$, we obtain

$$\sum_{j=1}^{q}\max(f_j(x^\star)-\theta^\star-(\varepsilon^\star)^{\gamma}w_j,0) = 0.$$

Therefore,

$$f_j(x^\star)-\theta^\star \leq (\varepsilon^\star)^{\gamma}w_j, \forall j=1,2,\cdots q. \tag{5.3.13}$$

Furthermore, from (5.3.7), we know

$$\sum_{j=1}^{q}(\max(f_j(x_k)-\theta_k-\varepsilon_k^{\gamma}w_j,0)\nabla f_j(x_k) + \sum_{j=q+1}^{m}(F_j(x_k)-\varepsilon_k^{\gamma}w_j)\nabla F_j(x_k) = 0.$$

Let $k \to +\infty$, by (5.3.13), it is thus immediate that

$$\sum_{j=q+1}^{m}(F_j(x^\star)-(\varepsilon^\star)^{\gamma}w_j)\nabla F_j(x^\star) = 0.$$

From the condition of theorem, we know $\nabla F_j(x^\star), \forall j = q+1, \cdots m$ are linearly independent. We obtain

$$F_j(x^\star) - (\varepsilon^\star)^\gamma w_j = 0, \quad \forall j = q+1, \cdots m. \tag{5.3.14}$$

It is immediate that

$$\Delta(x^\star, \theta^\star, \varepsilon^\star) = \sum_{j=1}^{q} (\max(f_j(x^\star) - \theta^\star - (\varepsilon^\star)^\gamma w_j, 0))^2 + \sum_{j=q+1}^{m} (F_j(x^\star) - (\varepsilon^\star)^\gamma w_j)^2 = 0.$$

Thus, one has

$$\begin{aligned} f_j(x^\star) - \theta^\star \leq 0 \qquad\qquad\qquad & \forall j = 1, 2, \cdots, q, \\ F_j(x^\star) - (\varepsilon^\star)^\gamma w_j = F_j(x^\star) = 0 \quad & \forall j = q+1, \cdots, m, \end{aligned}$$

namely, $(x^\star, \theta^\star) \in S$. ■

Theorem 5.3.1 *Suppose that $(x_k, \theta_k, \varepsilon_k) \in L(R_{\sigma_k})$ with finite $f_{\sigma_k}(x_k, \theta_k, \varepsilon_k), \varepsilon_k \neq 0$. For any accumulation point $(x^\star, \theta^\star, \varepsilon^\star)$, $\nabla F_j(x^\star), j = q+1, \cdots m$ satisfying linearly independent constraint qualification, then $(x^\star, \theta^\star)$ is a local optimal solution to (R) and $\theta^\star$ is the associated optimal objective function value.*

Proof. From the conditions of Theorem 5.3.1, it is thus immediate that there exists subsequence $\{(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k})\}_{\aleph} \subseteq \{(x_k, \theta_k, \varepsilon_k)\}$ such that $(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k}) \xrightarrow{k} (x^\star, \theta^\star, \varepsilon^\star)$. It then follows from Lemma 5.3.2 that $\varepsilon^\star = 0$ and $(x^\star, \theta^\star)$ is feasible point of the problem (R) satisfying $f_j(x^\star) \leq \theta^\star, \forall j = 1, 2, \cdots, q$. Therefore, there exists a neighbor $o(x^\star, \theta^\star, 0)$ and consider an arbitrary point $(x, \theta, 0) \in o(x^\star, \theta^\star, 0) \cap (S \times \{0\})$, in particular, $\theta = \max_{j=1,2,\cdots,q} f_j(x)$. Then, by the definition of $(x_k, \theta_k, \varepsilon_k)$, one has, for any $j = 1, 2, \cdots q$,

$$f_j(x^\star) \leq \theta^\star = f_\sigma(x^\star, \theta^\star, 0) \leq f_\sigma(x, \theta, 0) = \theta.$$

Therefore, $(x^\star, \theta^\star)$ is a local optimal solution to (R) and $\theta^\star$ is the optimal value. In fact, $\theta^\star = \max_{1 \leq j \leq q} f_j(x^\star)$. The proof is completed. ■

Corollary 5.3.1 *Suppose that every local minimizer $(x^\star, \theta^\star, \varepsilon^\star)$ of the penalty problem (R_σ) with finite $f_\sigma(x^\star, \theta^\star, \varepsilon^\star)$ and $\nabla F_j(x^\star), j = q+1, \cdots, m$ satisfy linearly independent constraint qualification, then $(x^\star, \theta^\star)$ is local minimizer of the primal problem (R) if and only if $\varepsilon^\star = 0$.*

Proof. If $(x^\star, \theta^\star)$ is local minimizer of the primal problem (R), then $f_j(x^\star) \le \theta^\star, j = 1, 2, \cdots, q$ and $F_j(x^\star) = 0, j = q+1, \cdots, m$. Using proof by contradiction, from Lemma 5.3.2, we have $\varepsilon^\star = 0$. Alternatively, if $\varepsilon^\star = 0$, in view of the construction of $f_\sigma(x, \theta, \varepsilon)$, then $(x^\star, \theta^\star) \in S$, i.e., $f_j(x^\star) \le \theta^\star$ and $F_j(x^\star) = 0, j = q+1, \cdots, m$. $(x^\star, \theta^\star)$ is a feasible point of (R). From the hypothesis that $(x^\star, \theta^\star, 0)$ is optimal solution to (R_σ), then $(x^\star, \theta^\star)$ is a local minimizer of the primal problem (R). ■

The next theorem explores that the finite termination property of the penalty function $f_\sigma(x, \theta, \varepsilon)$. Through this conclusion, the optimal solutions of primal problem (R) can be achieved within finite steps.

Theorem 5.3.2 *If the conditions of Theorem 5.3.1 hold and $\alpha, \beta, \gamma, \delta$ satisfy*

$$-\alpha - \beta + 2\delta \ge 0, \gamma > \delta, \tag{5.3.15}$$

then there exists $k_0 > 0$, when $k \ge k_0$, we have $\varepsilon_k = 0, x_k \in L(R)$.

Proof. We prove this theorem by contradiction. Assume the theorem is not true, then there exists a subsequence $\{(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k})\}_{\aleph} \subseteq \{(x_k, \theta_k, \varepsilon_k)\}$ such that for any $k_0 > 0$, when $n_k \ge k_0, (x_{n_k}, \theta_{n_k}, \varepsilon_{n_k}) \in L(R_{\sigma_{n_k}})$ with finite $f_{\sigma_{n_k}}(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k})$ and $\varepsilon_{n_k} \ne 0$ and the conditions of Theorem 5.3.2 hold for such subsequence. From Lemma 5.3.1, $(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k}) \notin S_{\varepsilon_{n_k}}$ holds. Therefore,

$$\frac{\partial f_{\sigma_{n_k}}(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k})}{\partial \varepsilon} = 0,$$

or equivalently,

$$\begin{aligned}
&\tfrac{\varepsilon_{n_k}^{-\alpha-\beta}}{(1-\varepsilon_{n_k}^{-2\delta}\Delta(x_{n_k},\theta_{n_k},\varepsilon_{n_k}))^2}[-\alpha\Delta(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k})(1 - \varepsilon_{n_k}^{-2\delta}\Delta(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k})\\
&-2\gamma\varepsilon_{n_k}^{\gamma}(\sum_{j=1}^{q} \max(f_j(x_{n_k}) - \theta_{n_k} - \varepsilon_{n_k}^{\gamma} w_j, 0) w_j + \sum_{j=q+1}^{m} (F_j(x_{n_k}) - \varepsilon_{n_k}^{\gamma} w_j) w_j)\\
&-2\delta\varepsilon_{n_k}^{-2\delta}\Delta^2(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k})] + \beta\sigma_{n_k} = 0.
\end{aligned} \tag{5.3.16}$$

From Lemma 5.3.2, we derive $\varepsilon_{n_k} \to \varepsilon^\star = 0, (x_{n_k}, \theta_{n_k}) \xrightarrow{k} (x^\star, \theta^\star) \in S$. Combining with $\varepsilon_{n_k} \ne 0, 0 < 1 - 2\varepsilon_{n_k}^{-2\delta}\Delta(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k}) < 1$, we have $\lim\limits_{\substack{\varepsilon_{n_k} \to \varepsilon^\star = 0 \\ x_{n_k} \to x^\star \in S}} 1 - \varepsilon_{n_k}^{-2\delta}\Delta(x_{n_k}, \theta_{n_k}, \varepsilon_{n_k}) = c^\star \in [\frac{1}{2}, 1]$.

Let $-\alpha - \beta + 2\delta \ge 0, -\alpha - \beta + \delta + \gamma \ge 0$, the first term of (5.3.16) tends to finite, and the second term tends to infinite which leads to contradiction. Therefore, there does not exist such subsequence. There exist $k_0 > 0$, when $k \geq k_0$, we have $\varepsilon_k = 0, (x_k, \theta_k, 0) \in L(R_{\sigma_k})$, where x_k, θ_k satisfy

$$f_j(x_k) \leq \theta_k, j = 1, 2, \cdots, q;$$
$$F_j(x_k) = 0, j = q+1, \cdots, m.$$

Thus, by $(x_k, \theta_k, 0) \in L(R_{\sigma_k})$, there exists a neighbor $o(x_k, \theta_k, 0)$ at $(x_k, \theta_k, 0), \sigma_k > 0$, for all $(x, \theta, 0) \in o((x_k, \theta_k, 0)) \cap (S \times \{0\})$, in particular, $\theta = \max_{1 \leq j \leq q} f_j(x)$, we have

$$\theta_k = f_{\sigma_k}(x_k, \theta_k, 0) \leq f_{\sigma_k}(x, \theta, 0) = \theta.$$

Thus, $(x_k, \theta_k) \in L(R)$. ■

5.4 Local Exactness Property

In this section, we shall show that, under fairly general conditions and some additional hypothesis, $(x^\star, \theta^\star, 0)$ is a local optimal solution to penalty problem (R_σ) if $(x^\star, \theta^\star)$ is a local minimizer of the original problem (R) for sufficiently large penalty parameter σ.

We now consider the nonsmooth case. Assume $f_j(x), j = 1, 2, \cdots, q$ and $F_j(x), j = q+1, \cdots, m$ are nonsmooth functions. In order to regularize $f_j, j = 1, 2, \cdots, q$ and $F_j, j = q+1, \cdots, m$, we embed $f_j(x), j = 1, 2, \cdots, q$ and $F_j(x), j = q+1, \cdots, m$ into the smoothing function $f_j(x, \varepsilon), F_j(x, \varepsilon)$ by introducing the above variable ε. Therefore, the introduced additional variable ε plays a critical role in solving the problem (R). The variable ε has active actions not only in perturbation for constraint system no matter how many constrained functions, but also in regularization of the nonsmooth case. After regularization, the regularized functions $f_j(x, \varepsilon)$ $j = 1, 2, \cdots, q$ and $F_j(x, \varepsilon)$ $j = q+1, \cdots, m$ are continuously differentiable in (x, ε), when $\varepsilon \neq 0$ and satisfy

$$f_j(x) = f_j(x, 0) = \lim_{\varepsilon \to 0} f_j(x, \varepsilon), \quad j = 1, 2, \cdots, q,$$
$$F_j(x) = F_j(x, 0) = \lim_{\varepsilon \to 0} F_j(x, \varepsilon), \quad j = q+1, \cdots, m.$$

We consider the following system

$$\begin{aligned} &\min_{\theta \in \mathbb{R}} \theta \\ &s.t. \quad f_j(x, \varepsilon) \leq \theta \quad \forall j = 1, 2, \cdots, q, \\ &\qquad\; F_j(x, \varepsilon) = 0 \quad \forall j = q+1, \cdots, m. \end{aligned} \tag{5.4.17}$$

Let $h_j(x,\theta,\varepsilon) = f_j(x,\varepsilon) - \theta, \forall j = 1,2,\cdots,q$, then (5.4.17) can be also formulated as

$$
\begin{aligned}
(R_\varepsilon) \quad & \min_{\theta\in\mathbb{R}} \theta \\
& s.t. \quad h_j(x,\theta,\varepsilon) \le 0 \quad j = 1,2,\cdots,q, \\
& \qquad\quad F_j(x,\varepsilon) = 0 \quad\quad j = q+1,\cdots,m.
\end{aligned}
$$

In what follows, the conditions that the error bound for the programming problem (R_ε) exists are considered. We make some assumptions:

(H_3) The Managasarian-Fromovitz constraint qualification holds at $(x^\star, \theta^\star, 0)$ for (R_ε), i.e., $\nabla F_j(x^\star, 0), j = q+1,\cdots,m$ are linearly independent and there exists a vector $\mathbf{s} \in \mathbb{R}^n$ such that

$$
(\mathbf{s},1)^\top \begin{pmatrix} \nabla_x h_j(x^\star,\theta^\star,0) \\ \frac{\partial h_j(x^\star,\theta^\star,0)}{\partial\theta} \end{pmatrix} > 0, \forall j \in I(x^\star,\theta^\star); \mathbf{s}^\top \nabla F_j(x^\star,0) = 0, \forall j = q+1,\cdots,m,
$$

where $I(x^\star,\theta^\star) = \{j = 1,2,\cdots,q \,|\, h_j(x^\star,\theta^\star,0) = 0\}$. According to the results reported in Ref. [7], we know the assumption (H_3) guarantees the error bound condition holds. Furthermore, combining with [51, Theorem 3] (Lemma 2.1.2), we obtain the following conclusion.

Lemma 5.4.1 *If the error bound condition holds, then there exist a neighborhood N_0 of $(x^\star,\theta^\star)$ and a constant $\tau > 0$ such that*

$$
\theta \ge \theta^\star - \tau\Big(\sum_{j=1}^{q} \|h_j(x,\theta,0)^+\| + \sum_{j=q+1}^{m} \|F_j(x,0)\|\Big)
$$

holds for $(x,\theta) \in N_0$.

Subsequently, we present an important theoretical result of the local exactness property. Before proving this result, some more assumptions are first given as follows.

(H_4) δ, β, γ are positive even numbers and satisfy $\delta \ge \beta$ and $\gamma \ge \beta$;

(H_5) For sufficiently small $0 < \varepsilon' << 1$,

$$
\|h_j(x,\theta,\varepsilon) - h_j(x,\theta,0)\| \le K\varepsilon^\beta, \forall j = 1,2,\cdots,q, \varepsilon \in [-\varepsilon',0) \cup (0,\varepsilon'],
$$

$$
\|F_j(x,\varepsilon) - F_j(x,0)\| \le K\varepsilon^\beta, \forall j = q+1,\cdots,m, \varepsilon \in [-\varepsilon',0) \cup (0,\varepsilon'].
$$

Theorem 5.4.1 *Suppose the above assumptions that* $(H_3), (H_4)$ *and* (H_5) *hold, for sufficiently large* σ*, there are a neighborhood* $N \subseteq N_0$ *of* $(x^\star, \theta^\star)$ *and sufficiently small* $0 < \varepsilon' << 1$ *such that*

$$f_\sigma(x, \theta, \varepsilon) > f_\sigma(x^\star, \theta^\star, 0) = \theta^\star \quad \textit{for all} \quad (x, \theta, \varepsilon) \in N \times [-\varepsilon', 0) \cup (0, \varepsilon'].$$

In particular, $(x^\star, \theta^\star, 0)$ *is a local minimizer of* $f_\sigma(x, \theta, \varepsilon)$*.*

Proof. Assume that the penalty parameter

$$\sigma \geq m\tau(K + 2).$$

We divide into two cases for further analysis: (i) $\Delta(x, \theta, \varepsilon) \geq \varepsilon^{2\delta}$ and (ii) $\Delta(x, \theta, \varepsilon) < \varepsilon^{2\delta}$ for $(x, \theta) \in N, \varepsilon \in [-\varepsilon', 0) \cup (0, \varepsilon']$.

Case (i). By the construction of penalty function, $f_\sigma(x, \theta, \varepsilon) = +\infty$. Therefore, $f_\sigma(x, \theta, \varepsilon) > f_\sigma(x^\star, \theta^\star, 0)$.
Case (ii). $\Delta(x, \theta, \varepsilon) < \varepsilon^{2\delta}$, i.e.,

$$\sum_{j=1}^{q}(\max(h_j(x, \theta, \varepsilon) - \varepsilon^\gamma w_j, 0))^2 + \sum_{q+1}^{m}(F_j(x, \varepsilon) - \varepsilon^\gamma w_j)^2 < \varepsilon^{2\delta},$$

this yields that

$$\begin{aligned} \|F_j(x, \varepsilon)\| &\leq \varepsilon^\gamma |w_j| + \|F_j(x, \varepsilon) - \varepsilon^\gamma w_j\| < \varepsilon^\gamma |w_j| + \varepsilon^\delta, \forall j = q + 1, \cdots, m; \\ \|h_j(x, \theta, \varepsilon)\| &\leq \varepsilon^\gamma |w_j| + \|h_j(x, \theta, \varepsilon) - \varepsilon^\gamma w_j\| < \varepsilon^\gamma |w_j| + \varepsilon^\delta, \forall j \in J^+(x, \theta, \varepsilon). \end{aligned}$$

where $J^+(x, \theta, \varepsilon) = \{j = 1, 2 \cdots, q | h_j(x, \theta, \varepsilon) \geq \varepsilon^\gamma w_j\}$. Furthermore, together with Lemma 5.4.1 and assumptions (H_4) and (H_5),

$$\begin{aligned} \theta^\star &\leq \theta + \tau(\sum_{j=1}^{q} \|h_j(x, \theta, 0)^+\| + \sum_{j=q+1}^{m} \|F_j(x, 0)\|) \\ &= \theta + \tau(\sum_{j \in J^+(x,\theta,0)} \|h_j(x, \theta, 0)\| + \sum_{j=q+1}^{m} \|F_j(x, 0)\|) \\ &\leq \theta + \tau(\sum_{j \in J^+(x,\theta,0)} \|h_j(x, \theta, \varepsilon)\| + \sum_{j \in J^+(x,\theta,0)} K\varepsilon^\beta + \sum_{j=q+1}^{m} \|F_j(x, \varepsilon)\| + \sum_{j=q+1}^{m} K\varepsilon^\beta) \\ &< \theta + \tau(\sum_{j=1}^{q} \varepsilon^\gamma |w_j| + q\varepsilon^\delta + Kq\varepsilon^\beta + \sum_{j=q+1}^{m} \varepsilon^\gamma |w_j| + (m - q)\varepsilon^\delta + K(m - q)\varepsilon^\beta) \\ &\leq \theta + m\tau(K + 2)\varepsilon^\beta \\ &\leq \theta + \sigma\varepsilon^\beta. \end{aligned}$$

The second inequality and the fourth inequality follow immediately from the assumptions (H_5) and (H_4), respectively. Therefore,

$$f_\sigma(x^\star, \theta^\star, 0) = \theta^\star < \theta + \sigma\varepsilon^\beta \leq f_\sigma(x, \theta, \varepsilon), \quad \text{for all} \quad (x, \theta, \varepsilon) \in N \times [-\varepsilon', 0) \cup (0, \varepsilon'].$$

$(x^\star, \theta^\star, 0)$ is a local minimizer of $f_\sigma(x, \theta, \varepsilon)$. This yields the inequality as desired. ■

5.5 Second-Order Sufficient Conditions for Local Exactness Property

In this section, let $f_j : \mathbb{R}^n \to \mathbb{R}, j = 1, 2, \ldots, q$ and $F_j : \mathbb{R}^n \to \mathbb{R}, j = q+1, \ldots, m$ be twice continuously differentiable functions. Motivated by Theorem 4.4 of [31] (Lemma 2.2.1), similar to the l_1 exact penalty function, we also establish the second-order sufficient optimality conditions of the exactness property for the proposed exact penalty function in this section. We present that, under the second-order sufficient conditions, $(x^\star, \theta^\star, 0)$ is a local minimizer of $f_\sigma(x, \theta, \varepsilon)$ when $(x^\star, \theta^\star)$ is a local minimizer of the programming problem (R). Compared with the first-order sufficient condition mentioned above, here, the second-order sufficient condition need not require any constraint qualifications. For the problem (R), define $h_j(x, \theta) = f_j(x) - \theta, j = 1, 2, \ldots, q$. Therefore, from the hypothesis, $h_j(x, \theta), j = 1, 2, \ldots, q$ are also twice continuously differentiable. Thus, the programming problem (R) can also be expressed as follows

$$\begin{cases} \min & \theta \\ \text{s.t.} & h_j(x, \theta) \leq 0 \quad \forall j = 1, 2, \ldots, q, \\ & F_j(x) = 0 \qquad \forall j = q+1, \ldots, m, \\ & x \in \mathbb{R}^n. \end{cases}$$

For simplicity in exposition, let $z = (x, \theta)^\top$, thus $z^\star = (x^\star, \theta^\star)^\top$. Define $I(z) = \{j = 1, 2, \ldots, q | h_j(z) = 0\}$. For $\lambda_j \in \mathbb{R}_+, j = 1, 2, \cdots, q, \mu_j \in \mathbb{R}, j = q+1, \cdots, m$, as shown in preliminaries section, we set the Lagrangian function as follows

$$L(z, \lambda, \mu) = \theta + \sum_{j=1}^{q} \lambda_j h_j(z) + \sum_{j=q+1}^{m} \mu_j F_j(x)$$

and divide $I(z)$ into sets $J(z) = \{j \in I(z) | \lambda_j > 0\}$ and $M(z) = \{j \in I(z) | \lambda_j = 0\}$.

Based on [50, Theorem 12.6] (Lemma 2.1.1), the second-order conditions of the problem of (R) are defined as follows.

Second-Order Sufficient Conditions

$z^\star$ is an optimal solution to the problem (R). $\lambda_j^\star, j = 1, 2, \cdots, q$ and $\mu_j^\star, j = q+1, \cdots, m$ are KKT multipliers at $z^\star$ and satisfy

$$\mathbf{d}^\top \nabla_{zz}^2 L(z^\star, \lambda^\star, \mu^\star)\mathbf{d} > 0,$$

where for all $\mathbf{d} \in D(z) \setminus \{\mathbf{0}\}$. $D(z)$ is set of directions is defined as follows:

$$D(z) = \{\mathbf{d} \in \mathbb{R}^{n+1} | \nabla h_j(z)^\top \mathbf{d} \le 0, \forall j \in M(z), \nabla h_j(z)^\top \mathbf{d} = 0, \forall j \in J(z),$$
$$\begin{pmatrix} \nabla F_j(x) \\ 0 \end{pmatrix}^\top \mathbf{d} = 0, j = q+1, \cdots, m\}.$$

Theorem 5.5.1 *Suppose that the assumption (H_4) and the second-order sufficient condition hold at $z^\star$. When $\sigma \ge 2\|\mu^\star\| + 2$,$(z^\star, 0)$ is a local minimizer of (R_σ).*

Proof. We prove this conclusion by the contradiction. Assume that $(z^\star, 0)$ is not a local minimizer of the penalty problem (R_σ). Therefore, there exist a neighborhood N of $(z^\star, 0)$, $0 < \bar{\varepsilon}_k << 1$ and a sequence $\{(z_k, \varepsilon_k)\} \subset \mathbb{R}^{n+1} \times [-\bar{\varepsilon}_k, 0) \cup (0, \bar{\varepsilon}_k]$ such that

$$\theta^\star = f_\sigma(z^\star, 0) \ge f_\sigma(z_k, \varepsilon_k), \qquad \text{for all } \ (z_k, \varepsilon_k) \in N \times [-\bar{\varepsilon}_k, 0) \cup (0, \bar{\varepsilon}_k].$$

From the above expression and the definition of penalty function, we have

$$\begin{aligned} 0 &\ge f_\sigma(z_k, \varepsilon_k) - f_\sigma(z^\star, 0) \\ &= \theta_k - \theta^\star + \tfrac{\varepsilon_k^{-\alpha}\Delta(z_k,\varepsilon_k)}{1-\varepsilon_k^{-2\delta}\Delta(z_k,\varepsilon_k)} + \sigma\varepsilon_k^\beta \\ &\ge \theta_k - \theta^\star + \tfrac{\Delta(z_k,\varepsilon_k)}{\varepsilon_k^\alpha} + \sigma\varepsilon_k^\beta. \end{aligned} \tag{5.5.18}$$

Therefore, from (5.5.18), one has $\lim\limits_{k\to\infty} \frac{\Delta(z_k,\varepsilon_k)}{\varepsilon_k^\alpha} = 0$. When k is sufficiently large, we have $\Delta(z_k, \varepsilon_k) < \varepsilon_k^\alpha$. From the parameters settings, we know $0 < \alpha < \delta$, and thus $\varepsilon_k^{2\delta} < \varepsilon_k^\delta < \varepsilon_k^\alpha$. By the construction of the penalty function, we know $\Delta(z_k, \varepsilon_k) < \varepsilon_k^{2\delta}$, otherwise, $f_{\sigma_k}(z_k, \varepsilon_k) = +\infty$. Therefore, this yields that

$$\|h_j(z_k)\| \le \varepsilon_k^\gamma |w_j| + \|h_j(z_k) - \varepsilon_k^\gamma w_j\| < \varepsilon_k^\gamma |w_j| + \varepsilon_k^\delta < 2\varepsilon_k^\beta, \forall j = 1, 2, \cdots, q.$$

The second inequality comes from Assumption (H_4). Likewise,

$$\|F_j(x_k)\| \le \varepsilon_k^\gamma |w_j| + \|F_j(x_k) - \varepsilon_k^\gamma w_j\| < \varepsilon_k^\gamma |w_j| + \varepsilon_k^\delta < 2\varepsilon_k^\beta, \forall j = q+1, \cdots, m.$$

Therefore, (5.5.18) yields that

$$
\begin{aligned}
0 &\geq \theta_k - \theta^\star + \tfrac{\Delta(z_k,\varepsilon_k)}{\varepsilon_k^\alpha} + \sigma\varepsilon_k^\beta. \\
&\geq \theta_k - \theta^\star + \sigma\varepsilon_k^\beta \\
&= L(z_k,\lambda^\star,\mu^\star) - L(z^\star,\lambda^\star,\mu^\star) - \sum_{j\in I(z^\star)} \lambda_j^\star(h_j(z_k) - h_j(z^\star)) - \sum_{q+1}^{m} \mu_j^\star F_j(x_k) \\
&\quad +\sigma\varepsilon_k^\beta \\
&= \tfrac{1}{2}(z_k - z^\star)^\top \nabla^2_{zz} L(z^\star,\lambda^\star,\mu^\star)(z_k - z^\star) + o(\|z_k - z^\star\|^2) - \sum_{j\in I(z^\star)} \lambda_j^\star \nabla h_j(\zeta_k)^\top (z_k - z^\star) \\
&\quad -{\mu^\star}^\top F(x_k) + \sigma\varepsilon_k^\beta \\
&\geq \tfrac{1}{2}(z_k - z^\star)^\top \nabla^2_{zz} L(z^\star,\lambda^\star,\mu^\star)(z_k - z^\star) + o(\|z_k - z^\star\|^2) - \sum_{j\in I(z^\star)} \lambda_j^\star \nabla h_j(\zeta_k)^\top (z_k - z^\star) \\
&\quad -\|\mu^\star\|\|F(x_k)\| + \sigma\varepsilon_k^\beta \\
&> \tfrac{1}{2}(z_k - z^\star)^\top \nabla^2_{zz} L(z^\star,\lambda^\star,\mu^\star)(z_k - z^\star) + o(\|z_k - z^\star\|^2) - \sum_{j\in I(z^\star)} \lambda_j^\star \nabla h_j(\zeta_k)^\top (z_k - z^\star) \\
&\quad -\|\mu^\star\|\|F(x_k)\| + \tfrac{\sigma}{2}\|F(x_k)\| \\
&\geq \tfrac{1}{2}(z_k - z^\star)^\top \nabla^2_{zz} L(z^\star,\lambda^\star,\mu^\star)(z_k - z^\star) + o(\|z_k - z^\star\|^2) \\
&\quad - \sum_{j\in I(z^\star)} \lambda_j^\star \nabla h_j(\zeta_k)^\top (z_k - z^\star) + \|F(x_k)\| \\
&\geq \tfrac{1}{2}(z_k - z^\star)^\top \nabla^2_{zz} L(z^\star,\lambda^\star,\mu^\star)(z_k - z^\star) + o(\|z_k - z^\star\|^2) - \sum_{j\in I(z^\star)} \lambda_j^\star \nabla h_j(\zeta_k)^\top (z_k - z^\star) \\
&\quad +\|\nabla F(\xi_k)^\top (z_k - z^\star)\|, \qquad (5.5.19)
\end{aligned}
$$

where the second equality comes from the mean value theorem where $\xi_k, \zeta_k \in (z_k, z^\star)$, the third inequality stems from Cauthy-Schwarz inequality and the fifth inequality comes from the assumption $\sigma \geq 2\|\mu^\star\| + 2$. Divided by $\|z_k - z^\star\|^2$ both sides of (5.5.19), we have

$$
0 > \frac{1}{2}{\mathbf{d}^k}^\top \nabla^2_{zz} L(z^\star,\lambda^\star,\mu^\star)\mathbf{d}^k + o(1) - \sum_{j\in I(z^\star)} \lambda_j^\star \frac{\nabla h_j(\zeta_k)^\top \mathbf{d}^k}{\|z_k - z^\star\|} + \|\nabla F(\xi_k)^\top \frac{\mathbf{d}^k}{\|z_k - z^\star\|}\|, \tag{5.5.20}
$$

where $\mathbf{d}^k = \frac{z_k - z^\star}{\|z_k - z^\star\|}$. Since for all k, $\|\mathbf{d}^k\| = 1$, thus without loss of generality, $\mathbf{d}^k \to \mathbf{d}^\star$ when $k \to \infty$. Obviously, $\mathbf{d}^\star \neq 0$. It follows from (5.5.20) and continuously differentiable property of $F(\cdot)$ that

$$
\lim_{k\to\infty} \nabla F(\xi_k)^\top \mathbf{d}^k = \nabla F(z^\star)^\top \mathbf{d}^\star = 0.
$$

Therefore,

$$\begin{pmatrix} \nabla F_j(x^\star) \\ 0 \end{pmatrix}^\top \mathbf{d}^\star = 0, \qquad\qquad j = q+1, \cdots, m.$$

Furthermore, as $k \to \infty$, then $z_k \to z^\star, \varepsilon_k \to 0$ and $\lim_{k\to\infty} h_j(z_k) \le 0$. Therefore, when $j \in M(z^\star)$, $z_k \to z^\star$ as $k \to \infty$, thus,

$$\lim_{k\to\infty} \nabla h_j(\zeta_k)^\top \mathbf{d}^k = \nabla h_j(z^\star)^\top \mathbf{d}^\star \le 0.$$

Similarly, when $j \in J(z^\star)$,

$$\lim_{k\to\infty} \nabla h_j(\zeta_k)^\top \mathbf{d}^k = \nabla h_j(z^\star)^\top \mathbf{d}^\star = 0.$$

Therefore, according to the second-order optimality condition, we have

$$0 \ge \frac{1}{2} \mathbf{d}^{\star\top} \nabla^2_{zz} L(z^\star, \lambda^\star, \mu^\star) \mathbf{d}^\star > 0,$$

which leads to a contradiction. This yields the contradiction as desired. ■

5.6 Numerical Examples

To give some insight into the behavior of the method presented in this section, the algorithm is implemented in Matlab 7.8.0 and executed on Intel Core 2CPU 2.39GHz with 1.99GB memory. We use $\|\nabla_{(x,\theta,\varepsilon)} f_\sigma(x,\theta,\varepsilon)\| \le 10^{-6}$ as stopping criteria. Tables 5.1-5.5 show the penalty parameter σ_k, $x_k, \theta_k, \varepsilon_k$ of the final iterate and $f(x_k)$ the function value of f at the final x_k for the corresponding problem.

Example 5.6.1

Let

$$\begin{aligned} f_1(x) &= 10(x_2 - x_1^2), \\ f_2(x) &= -10(x_2 - x_1^2), \\ f_3(x) &= 1 - x_1, \\ f_4(x) &= 1 + x_1, \\ F_1(x) &= 100x_1^2 + x_2^2 - 101, \\ F_2(x) &= 80x_1^2 - x_2^2 - 79. \end{aligned}$$

In order to make $\alpha, \beta, \gamma, \delta$ satisfy (5.2.3) and (5.3.15), in this example, the parameters used in this algorithm are set as $\alpha = 6, \beta = 8, \gamma = 10, \delta = 8$. The optimal solution and optimal value are $x^\star = (1, 1)$ and $f(x^\star) = 0$. We choose the initial point $x_0 = (2, -1), \theta_0 = -5, \varepsilon_0 = 2$.

Table 5.1: Numerical results of Example 5.6.1

σ_k	x_k	θ_k	ε_k	$f_{\sigma_k}(x_k, \theta_k, \varepsilon_k)$
2	(1.0000, 1.0000)	0.0000	0.2266	1.7559e-005
4	(1.0000, 1.0000)	0.0000	0.2991	2.8779e-004
6	(1.0000, 1.0000)	0.0000	0.2655	1.5005e-004
8	(1.0000, 1.0000)	0.0000	0.3041	6.0927e-004
10	(1.0000, 1.0000)	0.0004	0.0515	5.7905e-004

Example 5.6.2

Let

$$\begin{aligned}
f_1(x) &= \tfrac{1}{2}(x_1 + \tfrac{10x_1}{x_1+0.1} + 2x_2^2),\\
f_2(x) &= \tfrac{1}{2}(-x_1 + \tfrac{10x_1}{x_1+0.1} + 2x_2^2),\\
f_3(x) &= \tfrac{1}{2}(x_1 - \tfrac{10x_1}{x_1+0.1} - 2x_2^2),\\
F_1(x) &= x_1^2 + x_2^2 + x_1x_2,\\
F_2(x) &= -x_1 + x_2^2.
\end{aligned}$$

Here, the parameters used in this algorithm are set as $\alpha = 6, \beta = 8, \gamma = 10, \delta = 8$. The optimal solution and optimal value are $x^\star = (0, 0)$ and $f(x^\star) = 0$. We choose the initial point $x_0 = (2, 1), \theta_0 = -1, \varepsilon_0 = 2$.

Table 5.2: Numerical results of Example 5.6.2

σ_k	x_k	θ_k	ε_k	$f_{\sigma_k}(x_k, \theta_k, \varepsilon_k)$
2	(-0.0000, 0.0003)	0.0000	0.1938	5.2922e-006
4	(-0.0000, 0.0000)	0.0001	0.2210	9.8229e-005
6	(-0.0002, -0.0149)	0.0082	0.4125	0.0158
8	(0.0000, 0.0022)	0.0002	0.2259	2.7298e-004
10	(0.0000, 0.0020)	0.0002	0.2377	3.7813e-004

Example 5.6.3

Let

$$
\begin{aligned}
f_1(x) &= x_1^2 + x_2^2, \\
f_2(x) &= (2 - x_1)^2 + (2 - x_2)^2, \\
f_3(x) &= 2\exp(x_2 - x_1), \\
F_1(x) &= x_1 + x_2 - 2, \\
F_2(x) &= -x_1^2 - x_2^2 + 2.25.
\end{aligned}
$$

The parameters used in this algorithm are set as $\alpha = 6, \beta = 8, \gamma = 10, \delta = 8$. The optimal solution and optimal value are $x^\star = (1.35355, 0.64645)$ and $f(x^\star) = 2.25$. We choose the initial point $x_0 = (0, 0), \theta_0 = 1, \varepsilon_0 = 2$.

Table 5.3: Numerical results of Example 5.6.3

σ_k	x_k	θ_k	ε_k	$f_{\sigma_k}(x_k, \theta_k, \varepsilon_k)$
2	(1.0071, 1.0771)	2.1347	0.7886	2.5227
4	(1.3536, 0.6464)	2.2500	0.1803	2.2500
6	(1.3536, 0.6464)	2.2500	0.2064	2.2500
8	(1.3536, 0.6464)	2.2500	0.1603	2.2500
10	(1.3535, 0.6465)	2.2500	0.2401	2.2501

Example 5.6.4

Let

$$
\begin{aligned}
f_1(x) &= \exp\left(\frac{x_1^2}{1000} + (x_2 - 1)^2\right), \\
f_2(x) &= \exp\left(\frac{x_1^2}{1000} + (x_2 + 1)^2\right), \\
F_1(x) &= \frac{x_1^2}{1000} + x_2^2 + x_1 x_2, \\
F_2(x) &= -x_1 + x_2^2.
\end{aligned}
$$

The parameters used in this algorithm are set as $\alpha = 6, \beta = 8, \gamma = 10, \delta = 8$. The optimal solution and optimal value are $x^\star = (0, 0)$ and $f(x^\star) = 2.71828$. We choose the initial point $x_0 = (0, 0), \theta_0 = 1, \varepsilon_0 = 2$.

Example 5.6.5

Table 5.4: Numerical results of Example 5.6.4

σ_k	x_k	θ_k	ε_k	$f_{\sigma_k}(x_k, \theta_k, \varepsilon_k)$
2	(-0.0000, -0.0000)	2.7183	0.1773	2.7183
4	(0.0000, 0.0000)	2.7183	0.2325	2.7183
6	(0.0000, 0.0000)	2.7183	0.2260	2.7184

Let

$$\begin{aligned}
f_1(x) &= x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4,\\
f_2(x) &= f_1(x) + 10F_1(x),\\
f_3(x) &= f_1(x) + 10F_2(x),\\
f_4(x) &= f_1(x) + 10F_3(x),\\
F_1(x) &= x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 - 8,\\
F_2(x) &= x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 - 9,\\
F_3(x) &= x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 - 5.
\end{aligned}$$

The parameters used in this algorithm are set as $\alpha = 6, \beta = 8, \gamma = 10, \delta = 8$. The optimal solution and optimal value are $x^\star = (0, 0, 2, -1)$ and $f(x^\star) = -44$. We choose the initial point $x_0 = (-1, 1.5, 1.8, -1), \theta_0 = -50, \varepsilon_0 = 2$.

Table 5.5: Numerical results of Example 5.6.5

σ_k	x_k	θ_k	ε_k	$f_{\sigma_k}(x_k, \theta_k, \varepsilon_k)$
2	(-0.0068, 1.0008, 2.0046, -0.9942)	-43.9998	0.3791	-43.9988
10	(0.0434, 0.9945, 1.9699, -1.0360)	-43.9854	0.3967	-43.9788
18	(0.0309, 0.9961, 1.9787, -1.0258)	-43.9930	0.0073	-43.9786

Chapter 6

On an Exact Penalty Function Method for Nonlinear Mixed Discrete Programming Problems and Its Applications in Search Engine Advertising Problems

6.1 Introduction

In the context of this section, we firstly apply a novel exact and smoothing penalty function to a general class of nonlinear mixed discrete programming problems and then solve the new popular search engine advertising problem by utilizing the new proposed penalty function.

We consider a mixed discrete nonlinear programming problem given below

$$(\bar{U}) \qquad \begin{cases} \min & f(x,y) \\ \text{s.t.} & h_l(x,y) = 0 \quad \forall l = 1,2,\cdots,L_1, \\ & g_\ell(x,y) \le 0 \quad \forall \ell = 1,2,\cdots,L_2. \\ & x = [x_1, x_2, \cdots, x_n]^\top \in \mathbb{R}^n, \\ & y = [y_1, y_2, \cdots, y_m]^\top \in D_1 \times D_2 \cdots \times D_m, \end{cases}$$

where L_1 and L_2 are non-negative integers. The functions $h_l, l = 1,2,\cdots,L_1$ and $g_\ell, \ell = 1,2,\cdots,L_2$ are continuously differentiable with respect to all their arguments. For $i = 1,2,\cdots,m$, $D_i = \{a_{i,1}, a_{i,2}, \cdots, a_{i,J_i}\}$ is a discrete points set with J_i elements, where $a_{i,j}, j = 1,2,\cdots,J_i$ are given discrete values.

Many practical optimum design problems in the engineering and finance management fields can be formulated as nonlinear mixed discrete programming problems. For example, in the stocks trade marketing, a stock holder is merely requested to buy or sell integer number stocks with others. Thus, in order to maximize stock holders' total revenues, a mixed discrete programming problem should be considered.

Nowadays, available approaches to nonlinear mixed discrete programming problems are the cutting plane method [29] and the Branch-and-Bound method [43]. In both of the methods, a mixed discrete programming problem is first solved ignoring any discrete restrictions. These two methods have their drawbacks themselves. In particular, the former normally requires a large number of cuts and for the latter, as stated in Ref. [68], if the number of discrete variables is large, the number of codes created in the branching process becomes quite large, and consequently, the computational cost, as well, computers storage consuming will be quite a bit high. Another difficult associated with these methods is that they often fail to yield the global optimum.

In this section, nonlinear mixed programming problems is considered to be equivalently transformed into a continuous nonlinear programming problem. Each of the discrete variables is first represented by a set of new variables $v_{i,j}, i = 1,2,\cdots,m, j = 1,2,\cdots,J_i$ with a linear constraint, each of which takes value 0 or 1. The new discrete variables are then relaxed by introducing an auxiliary function, which constitutes a quadratic constraint, so that the new employed discrete variables become continuous on the interval $[0,1]$. Consequently, the mixed discrete programming problem is transformed into a continuous nonlinear programming problem. It is shown that, from

Theorem 3.1 in [72], under the linear and quadratic constraints, the transformed continuous optimization problem is equivalent to the mixed discrete programming problem. Unlike some of the existing literature, the new constraints introduced in this formulation are at most quadratic, and hence will not increase the number of local optima of a given problem. And then, we establish a novel exact and smooth penalty function to tackle this transcribed nonlinear continuous optimization problem.

Subsequently, we apply the proposed penalty function to search engine advertising problems. Internet search engines such as Google and Yahoo! provide a service where after a user has searched a specific term, sponsored links may be displayed in the front page in addition to search results. Sponsored links offer advertisers a more targeted method of advertising than traditional forms of advertising such as TV commercials, because they are customized. Search-based advertising has become a principal source of revenue for search engines. In the later section, the goal is to maximize the search engine revenue by choosing the optimal advertisers' bidding position. Motivated by the industrial practice and the reviewed literatures, we model the search engine advertisement auction problem as a large scale 0-1 integer programme. The constructed model is based on more realistic situations, e.g., (1) multiple slots, (2) generalized second price mechanism, (3) advertisers with their own choice behaviors, (4) quality score factor, (5) more than one keyword can match the query, which are not yet explored in the existing literature. Undoubtedly, the large number of variables would lead to computational challenges. The optimal strategies are quite computationally expensive. Here, we utilize the proposed exact and smooth penalty function for the formulated 0-1 integer programming, which is a special case of nonlinear mixed discrete programming problems. Finally, numerical experiments are conducted to verify that our method is effective and practical.

6.2 Equivalent Continuous Optimization Problems

To transform Problem $(\bar{U})$ into a constrained optimization problem with continuous variable, for each $i = 1, 2, \cdots, m$, introduce a transformation

$$y_i = \sum_{j=1}^{J_i} v_{i,j} a_{i,j}, \quad \forall i = 1, 2, \cdots, m,$$

where $a_{i,j} \in D_i$ for $j = 1, 2, \cdots, J_i$ and $v_i := (v_{i,1}, v_{i,2}, \cdots, v_{i,J_i})$ is a set of new variables and satisfy the following formulations

$$v_{i,j} = 0 \text{ or } 1, j = 1, 2, \cdots, J_i$$

$$\sum_{j=1}^{J_i} v_{i,j} = 1.$$

Therefore, with this transformation, Problem $(\bar{U})$ is transformed into another equivalent mixed discrete optimization problem:

$$\begin{cases} \min & f(x, v) \\ \text{s.t.} & h_l(x, v) = 0 \quad \forall l = 1, 2, \cdots, L_1, \\ & g_\ell(x, v) \leq 0 \quad \forall \ell = 1, 2, \cdots, L_2, \\ & \sum_{j=1}^{J_i} v_{i,j} - 1 = 0, i = 1, 2, \cdots, m, \\ & v_{i,j} = 0 \text{ or } 1, i = 1, 2, \cdots, m, \\ & x = [x_1, x_2, \cdots, x_n]^\top \in \mathbb{R}^n. \end{cases} \tag{6.2.1}$$

Furthermore, in order to use the existing algorithm for continuous constrained optimization problem, we define an auxiliary function as follows.

$$Q(v_i) = \sum_{j=1}^{J_i} j^2 v_{i,j} - (\sum_{j=1}^{J_i} j v_{i,j})^2, \forall i = 1, 2, \cdots, m.$$

The following theorem derived by Wang et al. [72] shows that $Q(v_i)$ is non-negative and the solutions to $Q(v_i) = 0$ under the constraint $v_{i,j} = 1$ for only one j and $v_{i,j} = 0$ for all other j.

Theorem 6.2.1 *([72]) Let $Q(v)$ be the quadratic polynomial defined by*

$$Q(v_i) = \sum_{j=1}^{J_i} j^2 v_{i,j} - (\sum_{j=1}^{J_i} j v_{i,j})^2, \forall i = 1, 2, \cdots, m.$$

If $v_{i,j} \geq 0$ for $j = 1, 2, \cdots, J_i, i = 1, 2, \cdots, m$ and

$$\sum_{j=1}^{J_i} v_{i,j} = 1, \forall i = 1, 2, \cdots, m,$$

then,

(1) for all $i = 1, 2, \cdots, m$, $Q(v_i) \geq 0$ and

(2) $Q(v_i) = 0$ if and only if $v_{i,j} = 1$ for one j and $v_{i,j} = 0$ for all other j.

Therefore, based on Theorem 6.2.1, we next consider the following continuous constrained programming problem, which is referred to as follows.

$$(U) \qquad \begin{cases} \min & f(x,v) \\ \text{s.t.} & H_l(x,v) = 0 \quad \forall l = 1,2,\cdots,M, \\ & G_\ell(x,v) \leq 0 \quad \forall \ell = 1,2,\cdots,N, \end{cases}$$

where

$$\begin{aligned} & H_l(x,v) = h_l(x,v), l = 1,2,\cdots,L_1, \\ & H_{l+L_1}(x,v) = \sum_{j=1}^{J_l} v_{i,j} - 1, l = 1,2,\cdots,m, \\ & H_{l+L_1+m}(x,v) = Q(v_l), l = 1,2,\cdots,m, \\ & G_\ell(x,v) = g_\ell(x,v), \ell = 1,2,\cdots,L_2, \\ & G_{\ell+j+L_2}(x,v) = v_{\ell,j} - 1, \ell = 1,2,\cdots,m, j = 1,2,\cdots,J_\ell, \\ & G_{\ell+j+\sum_{\ell=1}^{m} J_\ell + L_2}(x,v) = -v_{\ell,j}, \ell = 1,2,\cdots,m, j = 1,2,\cdots,J_\ell, \end{aligned}$$

where $M = L_1 + 2m$ and $N = L_2 + 2\sum_{\ell=1}^{m} J_\ell$.

Theorem 6.2.1 indicates that for each $i = 1,2,\cdots,m$, y_i can only take a discrete value from the set D_i. It implies that the mixed discrete Problem $(\bar{U})$ is transformed into an equivalent nonlinear programming problem (U) with continuous variables.

6.3 An Exact and Smooth Penalty Function for Equality and Inequality Constrained Minimization Problem

We reformulate the feasible region as a set S as follows:

$$S = \{(x,v) \in \mathbb{R}^r : H_l(x,v) = 0, l = 1,\cdots,M, G_\ell(x,v) \leq 0, \ell = 1,\cdots,N\} \neq \emptyset,$$

where $r = n + \sum_{i=1}^{m} J_i$. We introduce a new variable ε into the constraint function such that

$$S_\varepsilon = \{(x,v,\varepsilon) \in \mathbb{R}^{r+1} : H_l(x,v) = \varepsilon^\gamma w_l, \forall l = 1,\cdots,M, G_\ell(x,v) \leq \varepsilon^\gamma w_\ell, \forall \ell = 1,\cdots,N\},$$

where $w_l \in (0,1), l = 1,2,\cdots,M$. $w_\ell \in (0,1), \ell = 1,2,\cdots,N$. In particular, when $\varepsilon = 0, S_\varepsilon = S$. We make some assumptions for (U):

(1) There exists the global minimizer of (U), this means that $f(x,v)$ is bounded below on S;

(2) Let $L(U)$ be the set of local minimizers of problem (U). If $(x^\star, v^\star) \in L(U)$, then $L_{(x^\star,v^\star)} = \{(x,v) \in L(U) : f(x,v) = f(x^\star, v^\star)\}$ is a compact set.

The penalty function $f_\sigma(x,v,\varepsilon)$ and the associated penalty problem (U_σ) can be formulated as follows

$$(U_\sigma) \qquad \min_{(x,v,\varepsilon)\in\mathbb{R}^r\times(-\bar{\varepsilon},\bar{\varepsilon})} f_\sigma(x,v,\varepsilon).$$

$$f_\sigma(x,v,\varepsilon) = \begin{cases} f(x,v), & \text{if } \varepsilon = 0, (x,v) \in S, \\ f(x,v) - \varepsilon^\alpha \ln\Big(1 - \varepsilon^{-2\delta}\Delta(x,v,\varepsilon)\Big) + \sigma\varepsilon^\beta, & \\ & \text{if } \varepsilon \neq 0, 0 < 1 - 2\varepsilon^{-2\delta}\Delta(x,v,\varepsilon) < 1, \\ +\infty, & \text{otherwise,} \end{cases} \tag{6.3.1}$$

where $\alpha, \beta, \delta, \gamma$ are positive even numbers and $\beta > 1$, in particular, $\gamma > \delta$ throughout this section. $\sigma > 0$ is a penalty parameter. Denote the summation of constraint violation as follows

$$\begin{aligned} \Delta(x,v,\varepsilon) &= \sum_{l=1}^{M}(H_l(x,v) - \varepsilon^\gamma w_l)^2 + \sum_{\ell=1}^{N}(\max(0, G_\ell(x,v) - \varepsilon^\gamma w_\ell))^2 \\ &= \sum_{l=1}^{M}(H_l(x,v) - \varepsilon^\gamma w_l)^2 + \sum_{\ell\in I^+(x,v,\varepsilon)}(G_\ell(x,v) - \varepsilon^\gamma w_\ell)^2, \end{aligned}$$

where $I^+(x,v,\varepsilon) = \{\ell = 1, 2, \cdots, N | G_\ell(x,v) \geq \varepsilon^\gamma w_\ell\}$.

For convenience, let $z = [x,v]^\top \in \mathbb{R}^r$. Hereafter, we replace the variables vector (x,v) with the vector z.

Similar to previous statements in Chapter 5, the finite termination property can be obtained by a certain constraint qualification and appropriate parameters settings. So we omit the proof. Nevertheless, compared with aforementioned results, the proof of local exactness property has some differences. Therefore, for this proposed penalty function in this section, we arrive at the local exactness proof as follows.

6.4 Local Exactness Property

In this section, we shall show that, under fairly general conditions and some additional hypothesis, $(z^\star, 0)$ is a local optimal solution to penalty problem (U_σ) if $z^\star$ is a local minimizer of the original problem (U) for sufficiently large penalty parameter σ.

We now consider the nonsmooth case. Assume $f(z)$ and $H_l(z), l = 1, 2, \cdots, M$, and $G_\ell(z), \ell = 1, 2, \cdots, N$ are nonsmooth functions. In order to regularize f, H and G, we embed $f(z), H_l(z), l = 1, 2, \cdots, M$ and $G_\ell(z), \ell = 1, 2, \cdots, N$ into the smoothing function $f(z, \varepsilon), H_l(z, \varepsilon), \forall l = 1, 2, \cdots, M$ and $G_\ell(z, \varepsilon), \forall \ell = 1, 2, \cdots, N$ by introducing the above variable ε. Therefore, the introduced additional variable ε play critical roles in solving the problem (U). The variable ε has active actions not only in perturbation for constraint system no matter how many constrained functions, but also in regularization of the nonsmooth case. After regularization, the regularized functions $f(z, \varepsilon), H_l(z, \varepsilon)$ and $G_\ell(z, \varepsilon)$ are continuously differentiable in (z, ε), when $\varepsilon \neq 0$ and satisfy

$$
\begin{aligned}
f(z) &= f(z, 0) = \lim_{\varepsilon \to 0} f(z, \varepsilon), \\
H_l(z) &= H_l(z, 0) = \lim_{\varepsilon \to 0} H_l(z, \varepsilon), \forall l = 1, 2, \cdots, M, \\
G_\ell(z) &= G_\ell(z, 0) = \lim_{\varepsilon \to 0} G_\ell(z, \varepsilon), \forall \ell = 1, 2, \cdots, N.
\end{aligned}
$$

We consider the following system

$$
\begin{aligned}
(U_\varepsilon) \qquad & \min_{(z,\varepsilon) \in \mathbb{R}^{r+1}} f(z, \varepsilon) \\
& \quad s.t. \quad H_l(z, \varepsilon) = 0, \quad \forall l = 1, 2, \cdots, M, \\
& \qquad\qquad G_\ell(z, \varepsilon) \leq 0, \quad \forall \ell = 1, 2, \cdots, N.
\end{aligned}
$$

In the following part, the conditions that the error bound for (U_ε) exists are considered. We make some assumptions:

(H_6) $f(\cdot, 0)$ is Lipschitz continuous with Lipschitz constant L;

(H_7) The Managasarian-Fromovitz constraint qualification holds at $(z^\star, 0)$, i.e., $\nabla H_l(z^\star, 0)$, $l = 1, \cdots, M$ are linearly independent and there exists nonzero vector $\mathbf{s} \in \mathbb{R}^n$ such that

$$
\mathbf{s}^\top \nabla H_l(z^\star, 0) = 0, l = 1, 2, \cdots, M; \quad \mathbf{s}^\top \nabla G_\ell(z^\star, 0) > 0, \ell \in I(z^\star, 0),
$$

where $I(z^\star,0)=\{\ell=1,2,\cdots,N|G_\ell(z^\star,0)=0\}$. According to the results reported in [7], we know the assumption (H_7) guarantees the error bound condition holds. Furthermore, from [51, Theorem 3] (Lemma 2.1.2), we obtain the following conclusion.

Lemma 6.4.1 *Suppose that assumptions (H_6) and (H_7) hold, there exist a neighborhood N_0 of $z^\star$, and a constant $\tau>0$ such that*

$$f(z,0)\geq f(z^\star,0)-\tau(\sum_{l=1}^{M}\|H_l(z,0)\|+\sum_{\ell=1}^{N}\|G_\ell(z,0)^+\|)$$

holds for $z\in N_0$.

Now we present an important theoretical result of the local exactness proof. Before proving this result, some more assumptions are first given as follows.

(H_8) For sufficiently small $0<\varepsilon^{'}<<1$,

$$\|G_\ell(z,\varepsilon)-G_\ell(z,0)\|\leq K\varepsilon^\beta,\forall\ell=1,2,\cdots,N,\varepsilon\in[-\varepsilon^{'},0)\cup(0,\varepsilon^{'}],$$

$$\|H_l(z,\varepsilon)-H_l(z,0)\|\leq K\varepsilon^\beta,\forall l=1,2,\cdots,M,\varepsilon\in[-\varepsilon^{'},0)\cup(0,\varepsilon^{'}];$$

(H_9) $|f(z,\varepsilon)-f(z,0)|\leq K\varepsilon^\beta$, the domain of ε as (H_8).

Theorem 6.4.1 *Suppose the above assumptions that $(H_4),(H_6)-(H_9)$ hold, for sufficiently large σ, there are a neighborhood $N\subseteq N_0$ of $z^\star$ and sufficiently small $0<\varepsilon^{'}<<1$ such that*

$$f_\sigma(z,\varepsilon)>f_\sigma(z^\star,0)=f(z^\star)\quad\textit{for all}\quad(z,\varepsilon)\in N\times[-\varepsilon^{'},0)\cup(0,\varepsilon^{'}].$$

In particular, $(z^\star,0)$ is a local minimizer of $f_\sigma(z,\varepsilon)$.

Proof. Assume that the penalty parameter

$$\sigma\geq K+\tau(K+2)(M+N).$$

We divide into two cases for further analysis, namely, (i) $\Delta(z,\varepsilon)\geq\varepsilon^{2\delta}$ and (ii) $\Delta(z,\varepsilon)<\varepsilon^{2\delta}$ for $z\in N,\varepsilon\in[-\varepsilon^{'},0)\cup(0,\varepsilon^{'}]$.

Case (i). By the construction of penalty function, $f_\sigma(z,\varepsilon) = +\infty$. Therefore, $f_\sigma(z,\varepsilon) > f_\sigma(z^\star,0)$.

For case (ii), $\Delta(z,\varepsilon) < \varepsilon^{2\delta}$, i.e.,

$$\sum_{l=1}^{M}(H_l(z)-\varepsilon^\gamma w_l)^2 + \sum_{\ell\in I^+(z,\varepsilon)}(G_\ell(z)-\varepsilon^\gamma w_\ell)^2 < \varepsilon^{2\delta},$$

which yields

$$\begin{aligned}
&\|H_l(z,\varepsilon)\| \le \varepsilon^\gamma|w_l| + \|H_l(z,\varepsilon)-\varepsilon^\gamma w_l\| < \varepsilon^\gamma|w_l| + \varepsilon^\delta, \forall l = 1,2,\cdots,M;\\
&\|G_\ell(z,\varepsilon)\| \le \varepsilon^\gamma|w_\ell| + \|G_\ell(z,\varepsilon)-\varepsilon^\gamma w_\ell\| < \varepsilon^\gamma|w_\ell| + \varepsilon^\delta, \forall \ell \in I^+(z,\varepsilon).
\end{aligned}$$

Furthermore, together with Lemma 6.4.1 and assumptions $(H_4), (H_8), (H_9)$, one has

$$\begin{aligned}
f(z^\star,0) &\le f(z,0) + \tau\Big(\sum_{l=1}^{M}\|H_l(z,0)\| + \sum_{\ell\in I^+(z,0)}\|G_\ell(z,0)\|\Big)\\
&\le f(z,\varepsilon) + K\varepsilon^\beta + \tau\Big(\sum_{l=1}^{M}\|H_l(z,\varepsilon)\| + \sum_{l=1}^{M}K\varepsilon^\beta + \sum_{\ell\in I^+(z,0)}\|G_\ell(z,\varepsilon)\| + \sum_{\ell\in I^+(z,0)}K\varepsilon^\beta\Big)\\
&< f(z,\varepsilon) + K\varepsilon^\beta + \tau\Big(\sum_{l=1}^{M}\varepsilon^\gamma|w_l| + \varepsilon^\delta M + K\varepsilon^\beta M + \sum_{\ell=1}^{N}\varepsilon^\gamma|w_\ell| + \varepsilon^\delta N + K\varepsilon^\beta N\Big)\\
&\le f(z,\varepsilon) + K\varepsilon^\beta + \tau(K+2)(M+N)\varepsilon^\beta\\
&= f(z,\varepsilon) + [\tau(K+2)(M+N)+K]\varepsilon^\beta\\
&\le f(z,\varepsilon) + \sigma\varepsilon^\beta.
\end{aligned}$$

where M, N denote the dimension of equality constraint and inequality constraint of the programming problem (U_ε), respectively. The second inequality follows from (H_8) and (H_9). The fourth inequality follows immediately from the assumption (H_4). Therefore,

$$f(z^\star,0) < f(z,\varepsilon) + \sigma\varepsilon^\beta \le f_\sigma(z,\varepsilon).$$

This yields the inequality as desired. ∎

6.5 Numerical Examples

To give some insight into the behavior of the algorithm presented in this section.It is implemented in Matlab 7.8.0 and runs are made on Intel Core 2CPU 2.39GHz with 1.99GB memory. We use $\|\nabla_{(x,\varepsilon)}f_\sigma(x,\varepsilon)\| \le 10^{-6}$ as stopping criteria. Tables 6.1-6.2

show the computational results for the corresponding problem with the following items: the penalty parameter σ_k, x_k, ε_k of the final iterate and $f(x_k)$ the function value of f at the final x_k, and the constraint violation measure $\Delta(x_k, v_k, \varepsilon_k)$.

Example 6.5.1

$$\begin{aligned}
\min \quad & 5x_1x_2x_3 - \tfrac{1}{2}x_1^2 + 10(x_1-1)^2 - 2x_2x_3 - x_3 - \tfrac{3}{2}x_2^2 - x_3^2 \\
s.t. \quad & -x_1^2 - x_3^2 - x_1 - 2x_2 - x_3 + 2 = 0, \\
& x_1 + \tfrac{3}{4} \geq 0, \\
& (x_1 - x_3)^2 + x_2^3 - 0.1x_1 + 0.05x_1^2 + 1.05 \geq 0, \\
& x_i \in D_i = \{-3, -2, -1, 0, 1, 2, 3\}, \ i = 1, 2, 3.
\end{aligned}$$

The parameters used in this algorithm are set as $\alpha = 6, \beta = 8, \gamma = 10, \delta = 8$ and $\rho = 5$. We choose $x_0 = (0, 0, 0), \varepsilon_0 = 20$ as initial point. The optimal solution and optimal value are $x^\star = (1, -1, 1)$ and $f(x^\star) = -7.0000$ of the above example. This problem can also be solved by the l_1 exact penalty function in 74 iterations, with a final value of the penalty parameter of 10^3, optimal solution $(0.9996, -1.0006, 1.0011)$ and optimal value -7.0038.

Table 6.1: Numerical results of Example 6.5.1

Iter	σ_k	x_k	ε_k	$f(x_k)$	$\Delta(x_k, v_k, \varepsilon_k)$
29	10	(1.0650, -1.1281, 0.4050)	0.0714	-4.4556	0
31	15	(1.0002, -0.8388, 0.8880)	-0.0000	-5.4674	0
30	20	(1.1504, -1.0034, 0.8353)	0.0370	-6.5850	2.5619e-009
30	25	(0.9910, -0.9937, 1.0048)	0.0112	-6.9315	1.8309e-012
66	30	(1.0000, -1.0002, 0.9964)	0.0000	-6.9913	0

Example 6.5.2

$$\begin{aligned}
\min \quad & x_1^2 + x_1x_2 + 2x_2^2 - 6x_1 - 14x_2 - 12x_3 \\
s.t. \quad & \\
& x_1 + x_2 + x_3 = 20; \\
& -x_1 + 2x_2 \leq 30; \\
& x_i \in D_i = \{0.5(j-1), j = 1, 2, \cdots, 30\}, i = 1, 2, 3.
\end{aligned}$$

The parameters are set as $\alpha = 6, \beta = 8, \gamma = 10, \delta = 8$ and $\rho = 5$. $x_0 = (7, 7, 7), \varepsilon_0 = 2$ as initial point. The optimal solution and optimal value are $x^\star = (0, 0.5, 19.5)$ and $f(x^\star) = -240.5$. This problem can also be solved by l_1 exact penalty function in 38 iterations, with a final value of the penalty parameter of 10^4, optimal solution $(-0.0003, 0.5012, 19.4997)$ and optimal value -240.5047.

Table 6.2: Numerical results of Example 6.5.2

Iter	σ_k	x_k	ε_k	$f(x_k)$	$\Delta(x_k, v_k, \varepsilon_k)$
38	10	(-0.0326, -0.0207, 20.0408)	0.2763	-239.2208	0.0016
53	15	(-0.0093, 0.4979, 19.5078)	0.1591	-240.1362	9.8269e-005
95	20	(-0.0001, 0.5000, 19.5000)	0.1310	-240.1991	3.7044e-005

6.6 Search Engine Advertising Problems

In the search engine advertising problem, most search engines use auctions to sell positions for sponsored links. For this reason, it is also named as the position auction by Varian [71]. A separate auction is run for each search term. Advertisers' bids determine which advertisers' sponsored links are listed and in which order. When an internet user clicks on the advertisement link associated with the keyword, the advertiser is charged by the search engines. The number of advertisements that the search engine can show to a user is limited, and different positions on the search results page have different desirabilities for advertisers. For instance, an advertisement shown at the top of a page is more likely to be clicked than an advertisement shown in the bottom. In another word, different advertising positions have different Click-Through-Rate (CTR), defined as the ratio of the number of clicks on the advertisements to the number of appearances of the advertising links. Therefore, search engines employ an auction system to allocate the positions to advertisers and determine the charging price.

The most common sealed-bid auctions are the first price and second price auctions. In 1997, the search engines introduced the generalized first price (GFP) auction (see Edelman et al. [19] for more detail). As we know, the key roles of auction for the search engine advertising problems are to determine how to allocate positions for advertisers and what price the search engine should charge. The principle of GFP is when an internet user clicks on the advertisement link associated with a keyword, advertisers are allocated positions from high to low by their bidding price and pay the search engine with what they bid. However, Edelman and Ostrovsky [18] showed that there is no pure strategy equilibrium for the GFP. Furthermore, the GFP mechanism encourages insufficient investment and results in volatile bidding prices. To remedy the inefficiency, Edelman et al. [19] introduced a new system called the generalized second-price (GSP) auction mechanism for multiple slots. Before stating GSP, we first introduce the concept of standard second price auction. The standard second price auction is a type of sealed-bid auction, where a bidder submits written bidding price without knowing the bidding prices of the other bidders in the auction, and in which the highest bidder wins, but the actual price paid is the second-highest bid. Likewise, when the advertisers are sorted for different positions, the GSP mechanism stipulates that an advertiser in position i pays a price per click that equals the bidding price of the advertiser in position $i + 1$. In particular, if only one advertising slot is provided by search engine in the "per result page", GSP is equivalent to the standard second price auction.

Regarding to the rank of advertisers, the two main search engines, Google and Yahoo!, adopt different ranking mechanism. Yahoo! ranks advertisers in the descending order of their bid prices directly. Google's ranking mechanism is more complicated than Yahoo!'s. Google uses the "rank number" as ranking criterion, resulted by multiplying each advertiser's bid price with his "quality score". In the late 2008, Google revealed that quality score was used to determine which advertisement it would show in the sponsored link. There are many factors in determining quality score and the most important factor is CTR that we discussed earlier. Thus, Google ranks advertisers in the descending order of each advertiser's "rank number".

For the search-based advertising auctions, some of the features that have been considered in the literature include: equilibrium properties [22]; algorithm design [15, 64]; mechanism design [19, 20]; parametric estimate [17], incorporating budgets or not, and pay per click or pay per impression schemes. Subsequently, we review some most representative ones in the following. Edelman et al. [19] found that GSP auction generally does not have equilibrium in dominant strategies, and truth-telling is not an equilibrium strategy. They define the locally envy free equilibrium, which shows that there exists some position for each advertiser where the advertiser cannot be better off by swapping bids with the advertiser ranked one position above him. Through the definition and formulation of locally envy free equilibrium, Feng et al. [22] further presented a pricing model and derive the optimal reserve price for sponsored search advertising from the standpoint of search engine. In the algorithm aspect, most algorithms are designed for the budgeted advertisers. The optimization objective of the budget optimization problem is on the problem of a search engine trying to assign a sequence of search keywords matched with the users' queries to a set of competing bidders, each with a spending budget. Rusmevichientong and Williamson [64] developed an adaptive algorithm to show how to determine the bid price for selecting profitable keywords from the advertisers' point of view. Devanur and Hayes [15] demonstrated how an online learning algorithm with the budgeted advertisers can achieve a competitive ratio of $1-\varepsilon$ under random permutations without the assumption of bidders' arrival distribution. However, they all consider the simplified problem without the requirements of multiple slots and the second price payment. When the actual situations are considered, the model and the resulting optimization problem are much more complex. As we all know, there are thousands of queries received by a search engine every day. For every query, there exists an uncertain number of advertisers with budget constraints to bid for the advertisement position. However, it is obvious that there are only a finite number of slots to show these advertisements. As a result, for the search engine, it is critical to answer the question on how to select the profitable advertisers to maximize the revenue of search engine. Meanwhile, the advertisers' budget constraints are satisfied. On the other hand, every advertiser may be interested in more than one keyword and has different preference weights for different keywords.

In this section, our model's optimization objective is to maximize the search engine revenue by choosing the optimal advertisers' bidding position. Motivated by the industrial practice and the reviewed literatures, we model the search engine advertisement auction problem as a $0-1$ integer programme. The constructed model is based on more realistic situations, e.g., (1) multiple slots, (2) generalized second price mechanism, (3) advertisers with their own choice behaviors, (4) quality score factor, (5) more than one keyword can match the query, which are not yet explored in the current literature [15]. We apply new exact and smooth penalty function to tackle general scale search engine advertising problems. Undoubtedly, the large number of variables would lead to computational challenges. The optimal strategies are quite computationally expensive.

Therefore, on the other hand, we present an effective approach based on Lagrangian relaxation coupled with subgradient algorithm and column generation methodology to solve large scale search engine advertising problems. Furthermore, we provide numerical simulations to address managerial insights on both operational and theoretical aspects and compare numerical performances with currently existing algorithms for solving search engine advertising problems.

6.6.1 Problem Formulation and Model Description

In this section, we model the SA auction. The list of notation used in the section is given as follows:

t : query t;

j : bidder (advertiser) j;

b_j: bidder j's budget;

λ_k: the probability of an arriving advertiser belongs to market segment k;

N : total keywords set for all queries;

S^t : the keywords set for query t for auction;

$b_j^{\ell t}$: bidding price of bidder j for keyword $\ell \in S^t$, where S^t is a specified keywords set for the query t;

L_ℓ : $\{j_p : p = 1, 2, \cdots, P_\ell\}$, where the index j_p is sorted by the bidding price, P_ℓ is the number of bidders for the keyword of $\ell \in S^t$;

$|I_{\ell t}|$: the slot capacity of a keyword ℓ corresponding to a query t;

$Q_{j_p}^{\ell t}$: the quality scores of bidder j in position p for a keyword ℓ corresponding to a query t;

$n_{j_p}^{\ell t}$: the expected number of clicks by search engine users for the advertiser j in position p for a keyword ℓ corresponding to a query t.

For every keyword, we take a GSP auction among those advertisers who are interested in these keywords, denoted as $\underset{l \in S^t}{\cup} L_l$. It is possible that an advertiser has interests in more than one keywords for the same query. So an internet user may see the same advertiser appears more than one time but for different keywords associated the same query.

Given an offer set $S \subset N$ of available keywords is the search engine's offer set, an arriving advertiser j chooses a keyword $\ell \in S$ with a probability $P_j^\ell(S)$, where $P_j^\ell(S) = 0$ if $\ell \notin S$. We denote the no-purchase probability as $P_j^0(S)$, and we have $\sum_{\ell \in S} P_j^\ell(S) + P_j^0(S) = 1$. Advertisers belong to different market segments $k = 1, 2, \cdots, L$ and each segment is characterized by one consideration set $C_k \subset N$. As is generally the case in the choice behavior related revenue management literature, these probabilities will be based on the multinomial logit (MNL) model (e.g., see [3] for a detailed description of the MNL model). Under the MNL model, the choice probability that an advertiser j belongs to a segment k is defined by a preference vector $\mathbf{v_k} \geq 0$, that indicates the advertiser preference weight for each keyword contained in C_k. This vector, together with the no-purchase preference v_{k0}, determines an advertiser's choice probabilities as follows: if we let $P_{\ell j}^k(S)$ denote the probability that an advertiser j from C_k chooses advertisement keyword $\ell \in C_k \cap S$, when S is offered, then $P_{\ell j}^k(S) = \frac{v_{k\ell}}{\sum_{h \in C_k \cap S} v_{kh} + v_{k0}}$. If $\ell \notin C_k \cap S$ or $\ell \notin C_k$, then $v_{k\ell} = 0$ and hence $P_{\ell j}^k(S) = 0$. Noting that from search engine's perspective, the keywords set of an advertiser is not distinguishable, and the probability that an arriving advertiser j chooses a keyword $\ell \in S$ is given by $P_j^\ell(S) = \sum_{k=1}^{L} \lambda_k P_{\ell j}^k(S)$, where $|L| \leq 2^{|S|} - 1$ and $\sum_{k=1}^{L} \lambda_k = 1$.

To illustrate the choice behavior of advertisers, let's consider a specific example. Suppose that a customer enters "household electrical appliances " into the search engine "Google". Google may assign, for example, "washing machine", "TV set", "refrigerator" (see Table 6.3), as keywords for bidders to bid. There are five household electrical appliances producers who want to bid for these keywords. However, among these household electrical appliances producers, there are one all-round type, three producers may have expertise in some two kinds of products, and one producer chooses only one product keyword as follows.

As illustrated in Table 6.4, advertisers are divided into five different segments, which

Table 6.3: Keywords Products

Product	Keyword(s)
1	washing machine
2	TV set
3	refrigerator

are characterized by consideration sets, i.e., a subset of the advertisement keywords provided by the search engine. In this section, we consider general version of this model, in which advertisers may belong to overlapping segments. The second column of Table 6.4 demonstrates the probability of each arriving advertiser belonging to a particular segment. The third column describes the advertisers' corresponding consideration set. As stated above, the consideration sets from different segments may be overlapping (such as segments 1 and 2). The fourth column shows the preference values for each product in the consideration set, and the preference values for the no purchase weight (last coordinate in the vector), as shown in Table 6.4. Therefore, the probability of an advertiser choosing the first keyword "washing machine" is

$$p = 0.2 \cdot \frac{6}{6+1} + 0.2 \cdot \frac{5}{5+8+1} + 0.25 \cdot \frac{6}{6+5+2} + 0.25 \cdot \frac{9}{9+5+2} + 0.1 \cdot \frac{9}{9+3+6+2} = 0.4032.$$

Next, we show the detail formulation. Define "rank number $=$ bidding price $\times$ quality score", where quality score is mainly based on CTR, relevancy of the keyword pertaining to the advertiser's business and other factors. Thus, by the definition of rank number, $b_{j_p}^{\ell t} Q_{j_p}^{\ell t}$ denotes the rank number of the pth advertiser assigned to keyword ℓ for query t. So, for the search engine, before assigning which advertisers to match the keywords, the calculations of quality scores of the participated advertising bidders for associated keywords should be done. Therefore, through the second price mechanism, the actual cost per click of the advertiser j in position p is $\frac{b_{j_{p+1}}^{\ell t} Q_{j_{p+1}}^{\ell t}}{Q_{j_p}^{\ell t}}$.

Table 6.4: The Segment Definition

Segment	λ_k	Consideration Set	Preference Vector	Producers Description
1	0.2	{1}	(6, 1)	Only expertise in washing machine
2	0.2	{1, 2}	(5, 8, 1)	Non all-round type, more expertise in TV set compared with washing machine
3	0.25	{1, 3}	(6, 5, 2)	Non all-round type, expertise both in washing machine and refrigerator
4	0.25	{2, 3}	(9, 5, 2)	Non all-round type, more expertise in TV set compared with refrigerator
5	0.1	{1, 2, 3}	(9, 3, 6, 2)	All-round type, especially for washing machine

$$\begin{cases} \max & \sum_j \sum_t \sum_{\ell \in S^t} b_{j_{p+1}}^{\ell t} \frac{Q_{j_{p+1}}^{\ell t}}{Q_{j_p}^{\ell t}} n_{j_p}^{\ell t} P_{j_p}^{\ell}(S^t) x_{j_p}^{\ell t} \\ \text{s.t.} & \sum_t \sum_{\ell \in S^t} b_{j_{p+1}}^{\ell t} \frac{Q_{j_{p+1}}^{\ell t}}{Q_{j_p}^{\ell t}} n_{j_p}^{\ell t} P_{j_p}^{\ell}(S^t) x_{j_p}^{\ell t} \leq b_j \quad \forall j, \\ & \sum_j x_{j_p}^{\ell t} \leq |I_{\ell t}| \quad \forall \ell, t, \\ & x_{j_p}^{\ell t} \in \{0, 1\}, \end{cases} \tag{6.6.1}$$

where $j_p^{\ell t} \in L_\ell^t$ and

$$x_{j_p}^{\ell t} = \begin{cases} 1, & \text{bidder } j \text{ in the position } p \text{ choosing keyword } \ell \in S^t, \text{where } S^t \\ & \text{is offer set given by the search engine for query } t \text{ is selected out;} \\ 0, & \text{otherwise.} \end{cases}$$

The objective function is to maximize the total revenue of the search engine under the second price mechanism. In particular, the MNL model is also introduced. For this purpose, $x_{j_p}^{\ell t}$, as the controlled variable, is introduced to determine which advertisers, with their own advertisements preference, will be chosen for displaying. The first constraint means that the advertising cost cannot exceed the advertiser's limited budget. The second constraint shows that one advertiser cannot display his advertisement more than one time for the same keyword and the amount of displayed advertisers cannot exceed the slot capacity of every keyword $\ell \in S^t$ for some query t. Note that under the second price mechanism, the actual cost per click is related to the next bidder's rank number. Specifically, we set the advertiser who is in the last position to pay for the reserve price predetermined by the search engine when the number of bidder is less than the permissible capacity slots. In view of the advertisers' choice behavior, we introduce the MNL criterion $P_{\ell j}^k(S^t) = \frac{v_{k\ell}}{\sum_{h\in C_k\cap S^t} v_{kh}+v_{k0}}$ into the optimization problem, which yields

$$\begin{cases} \max & \sum_j \sum_t \sum_{\ell\in S^t} b_{j_{p+1}}^{\ell t} \frac{Q_{j_{p+1}}^{\ell t}}{Q_{j_p}^{\ell t}} n_{j_p}^{\ell t} \sum_{k=1}^{L} \lambda_k \frac{v_{k\ell}}{\sum_{h\in C_k\cap S^t} v_{kh}+v_{k0}} x_{j_p}^{\ell t} \\ \text{s.t.} & \sum_t \sum_{\ell\in S^t} b_{j_{p+1}}^{\ell t} \frac{Q_{j_{p+1}}^{\ell t}}{Q_{j_p}^{\ell t}} n_{j_p}^{\ell t} \sum_{k=1}^{L} \lambda_k \frac{v_{k\ell}}{\sum_{h\in C_k\cap S^t} v_{kh}+v_{k0}} x_{j_p}^{\ell t} \le b_j \quad \forall j, \\ & \sum_j x_{j_p}^{\ell t} \le |I_{\ell t}| \qquad \forall \ell, t, \\ & x_{j_p}^{\ell t} \in \{0,1\}. \end{cases}$$

By Theorem 6.2.1, the integer programming can be transformed into a general nonlinear programming.

$$\begin{cases} \max & \sum_j \sum_t \sum_{\ell\in S^t} b_{j_{p+1}}^{\ell t} \frac{Q_{j_{p+1}}^{\ell t}}{Q_{j_p}^{\ell t}} n_{j_p}^{\ell t} \sum_{k=1}^{L} \lambda_k \frac{v_{k\ell}}{\sum_{h\in C_k\cap S^t} v_{kh}+v_{k0}} x_{j_p}^{\ell t} \\ \text{s.t.} & \sum_t \sum_{\ell\in S^t} b_{j_{p+1}}^{\ell t} \frac{Q_{j_{p+1}}^{\ell t}}{Q_{j_p}^{\ell t}} n_{j_p}^{\ell t} \sum_{k=1}^{L} \lambda_k \frac{v_{k\ell}}{\sum_{h\in C_k\cap S^t} v_{kh}+v_{k0}} x_{j_p}^{\ell t} \le b_j \quad \forall j, \\ & \sum_j x_{j_p}^{\ell t} \le |I_{\ell t}| \qquad \forall \ell, t, \\ & x_{j_p}^{\ell t}(x_{j_p}^{\ell t} - 1) = 0 \qquad \forall j, \ell, t. \end{cases}$$

Based on the previous developed new exact penalty function method, we could deal with this nonlinear programming problem.

6.6.2 Simulation Results

In this simulation section, our purpose is to verify the efficiency of penalty function algorithm for solving the search engine advertising auction problem. Here, simulation results between our integer programming model and the greedy algorithm are investigated. The main idea of greedy algorithm is to allocate the slots to advertisers based on the bidding price from high to low for every keyword, regardless of another possible bidding price on the another keyword if this advertiser has more than one bidding keywords.

Table 6.5: Revenues Comparisons Between Penalty Function Method, Greedy Algorithm and Linear Programming Relaxation

Type	Slots	Penalty function method	Greedy algorithm	Linear programming relaxation	ε_k	$\Delta(x_k, \varepsilon_k)$
(i)	2	987	592	1092	0.0500	3.4943e-005
	5	1071	765	1146	0.0544	6.8094e-015
(ii)	2	1231	886	1331	0.0500	4.6082e-005
	5	1469	1060	1624	0.0434	4.0000e-012
(iii)	2	2162	1208	2339	0.2000	0.0800
	5	3018	1506	3081	0.1630	0.0806
(iv)	2	1484	1361	1760	0.0300	0.0442
	5	2344	1221	2720	0.9580	0.8425

We generate the bidding price of every advertiser for several keywords, which follows the uniform distribution in the interval $[0, 1]$. We assume that the advertisers' quality scores are independently and identically drawn from a uniform distribution in the interval $[0, 10]$. Thus, the advertiser's rank number is the product of his bidding price and corresponding quality score. For convenience of numerical results, we assume the number of advertisement position provided by the search engine for every keyword is 2 and 5, respectively. The reserve price is setted to be 0.1. The click through rates of advertisement slots follow the uniform distribution in the interval $[0, 100]$ in the decreasing sequence. Moreover, budget of every advertiser follows the uniform distribution in the interval $[1000, 1200]$. In order to simulate the MNL model, we consider the case when the keyword number that every advertiser in a certain segment can bid is less than 5 and there exist different preference weights for different keywords. Every advertiser's preference weights for his choosing keywords follow a unform distribution in the interval of $[0, 10]$.

For different numbers of slots, we compare the numerical results for the following four cases, (i) 10 advertisers for bidding 4 keywords, (ii) 10 advertisers for bidding 5 keywords, (iii) 10 advertisers for bidding 8 keywords, (iv) 10 keywords for bidding 10 keywords, respectively.

It is not difficult to find that the constraint violations of the penalty function method is quite small. This demonstrate that the obtained solutions by the penalty function method are feasible solutions. On the other hand, it is easily observed from Table 6.5 that the numerical behaviors of integer programming model show significant improvements in the revenue of search engine compared to the greedy algorithm for every case. However, it is should be noticed that in the simulation test, the cases shown in the Table 6.5 are not large scale. For the general scale search engine advertising problem, penalty function algorithm is effective and practical. For the large scale search engine advertising problem, in consideration of practical calculation confinement such as accumulative errors factor, there still exists some difficulties need to get over. Therefore, this is our future research issue that we are engaged in.

6.6.3 A Lagrangian-based method for search engine advertising problem

As well known, the numbers of users' queries and the corresponding advertisers every day are quite large. Therefore, it results in a large scale 0-1 integer programming. Considering that numerical behavior of penalty function algorithm is low efficiency in the large scale search engine advertising problem, we design a Lagrangian-based method to tackle with large scale problems. We present an extension to the sub-gradient algorithm based on Lagrangian relaxation coupled with column generation method in order to improve the dual Lagrangian multipliers and accelerate its convergence.

Dantzig-Wolfe decomposition as applied to an integer program is a specific form of problem reformulation that aims at providing a tighter linear programming relaxation bound, which is a well established methodology in large-scale integer programming. Now we show that the Dantzig-Wolfe decomposition principle of linear programming has its equivalence in 0-1 integer programming.

From the well known theorem of Minkowski and Weyl [66], we obtain the result that $X \subseteq \{0,1\}^n, \forall x \in X$ can be expressed as the convex combination of a finite set of extreme points $x = \sum\limits_{i \in Q} \omega_i \hat{x}_i$, where $\hat{x}_i, i = 1, 2, \cdots, |Q|$ are extreme points of conv(X), the index set Q is finite and $\sum\limits_{i \in Q} \omega_i = 1, \omega_i \in \{0, 1\}$.

Motivated by this fact, in our model, we let

$$X = \Big\{x^{\ell t}_{j_p} \in \{0,1\} \Big| \sum_t \sum_{\ell \in S^t} b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t) x^{\ell t}_{j_p} \le b_j \Big\}.$$

If X is finite, we replace X by conv(X) and there exist finite extreme points $\{\hat{x}_i\}_{i \in Q} \in$ conv(X) such that

$$\begin{cases} \max & \sum_j \sum_t \sum_{\ell \in S^t} b^{\ell}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t)(\sum_i \omega^{\ell t}_{ij_p} \hat{x}_i) \\ \text{s.t.} & \sum_j \sum_i \omega^{\ell t}_{ij_p} \hat{x}_i \le |I_{\ell t}| \qquad \forall \ell, t, \\ & \sum_i \omega^{\ell t}_{ij_p} = 1 \qquad \forall j, \ell, t, \\ & 0 \le \omega^{\ell t}_{ij_p} \le 1. \end{cases}$$

Here, we relax $\omega^{\ell t}_{ij_p} \in \{0,1\}$ as $0 \le \omega^{\ell t}_{ij_p} \le 1$. Therefore, the obtained optimal objective function value of the above linear programming provides the upper bound value from Dantzig-Wolfe decomposition. We now investigate the relationship between Lagrangian relaxation and Dantzig-Wolfe decomposition. We introduce the Lagrangian relaxation function of (6.6.1) as follows.

$$L(u) = \min_{x^{\ell t}_{j_p} \in X} -\sum_j \sum_t \sum_{\ell \in S^t} b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t) x^{\ell t}_{j_p} + \Big[\sum_t \sum_{\ell \in S^t} u^t_\ell (\sum_j x^{\ell t}_{j_p} - |I_{\ell t}|)\Big].$$

The Lagrangian function $L(u)$ is concave for all $u \ge 0$. It is piecewise linear and only sub-differential in its breakpoints. The Lagrangian dual problem is denoted as $\pounds := \max_{u \ge 0} L(u)$. The optimality of $u^\star$ implies that $u^{t\star}_\ell (\sum_j x^{\ell t}_{j_p} - |I_{\ell t}|) = 0$ which is based on the complementary slackness property. $\sum_j x^{\ell t}_{j_p} \le |I_{\ell t}|$ (feasibility) must be verified to prove optimality. Once this condition is violated, the primal-dual pair $(x, u^\star)$ is not optimal. Let

$$v = \min_{x^{\ell t}_{j_p} \in X} \sum_j \sum_t \sum_{\ell \in S^t} \Big(u^t_\ell - b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t)\Big) x^{\ell t}_{j_p} - \sum_t \sum_{\ell \in S^t} u^t_\ell |I_{\ell t}|$$

and replace X by conv(X). We know $x^{\ell t}_{j_p} \in X$ is an extreme point $\hat{x}_i, i \in Q$ of the set conv(X).

$$\begin{cases} \max & v \\ \text{s.t.} & \sum_t \sum_{\ell \in S^t} (|I_{\ell t}| - \sum_j x^{\ell t}_{j_p}) u^t_\ell + v \le -\sum_j \sum_t \sum_{\ell \in S^t} b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t) \hat{x}_i, \\ & u^t_\ell \ge 0 \qquad \forall \ell,\ t, \end{cases}$$

where $\hat{x}_i, i \in Q$ are extreme points of the set conv(X). The dual problem of the above model is

$$\begin{cases} \max & \sum_j \sum_t \sum_{\ell \in S^t} \sum_i b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} (\omega^{\ell t}_{j_p})_i P^{\ell}_{j_p}(S^t) \hat{x}_i \\ \text{s.t.} & \sum_j \sum_i (\omega^{\ell t}_{j_p})_i \hat{x}_i \leq \sum_i (\omega^{\ell t}_{j_p})_i |I_{\ell t}| \quad \forall \ell, t, \\ & \sum_i (\omega^{\ell t}_{j_p})_i = 1, \\ & (\omega^{\ell t}_{j_p})_i \geq 0. \end{cases}$$

This result shows the upper bound obtained from Dantzig-Wolfe decomposition is the same as the Lagrangian bound.

6.6.4 Algorithm

Motivated by [9], we propose an approach based on Lagrangian relaxation with subgradient optimization for solving SA auction problem. Our scheme is based on dual information associated with the widely-used Lagrangian relaxation. Define the Lagrangian relaxation problem as follows:

$$\begin{cases} \min & -\sum_j \sum_t \sum_{\ell \in S^t} b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t) x^{\ell t}_{j_p} + \Big[\sum_t \sum_{\ell \in S^t} z^t_\ell (\sum_j x^{\ell t}_{j_p} - |I_{\ell t}|) \Big] \\ & + \sum_j y_{j_p} (\sum_t \sum_{\ell \in S^t} b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} \alpha^t_{\ell, j_p} P^{\ell}_{j_p}(S^t) x^{\ell t}_{j_p} - b_j) \\ \text{s.t.} & x^{\ell t}_{j_p} \in \{0, 1\}, \end{cases}$$

where $j^{\ell t}_p \in L^t_\ell$, z^t_ℓ and y_{j_p} are Lagrangian multipliers associated to the constraints $\sum_j x^{\ell t}_{j_p} - |I_{\ell t}| \leq 0$ and $\sum_t \sum_{\ell \in S^t} b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t) x^{\ell t}_{j_p} \leq b_j \quad \forall j$, respectively. The above Lagrangian relaxation problem has the integrality property, any optimal Lagrangian multipliers to the dual of the linear programming relaxation of this integer programming problem are also an optimal solution to the Lagrangian problem.

$$z^t_\ell + (y_{j_p} - 1) b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} \alpha^t_{\ell, j_p} P^{\ell}_{j_p}(S^t) < 0$$

is the Lagrangian cost associated with the advertiser j for bidding the keyword $\ell \in S^t$. In our computational experience, we can produce "good" multipliers in a few iterations. Compared to computing an optimal multiplier vector, solving a linear programming is typically computationally expensive for large scale problems. A commonly used approach for finding a near-optimal multiplier vector and giving a reasonable upper bound with a relatively small effort, is to use the sub-gradient algorithm. This procedure is attractive because of its low computational cost. For this reason, we decide to use the sub-gradient algorithm as a"filter" to improve the multipliers before they are sent to the subproblems. An outline of the algorithm is given as follows.

Algorithm:

Step 1: Initialization

Construct the initial points for the primal problem given the initial Lagrangian dual multipliers vectors.

Step 2: Computation of Lagrangian dual multipliers

Solve a Lagrangian dual problem with sub-gradient method and compute the Lagrangian dual function value. If (i) the gap between this value and the upper bound of the optimal value is sufficiently small or (ii) a predetermined fixed number of iterations is reached, go to Step 3.

Step 3: Generation of profitable bidders

Compute the Lagrangian cost in the set of the currently not yet assigned bidders set. Select the most profitable bidders from the negative Lagrangian cost and add them to the selected bidders set until a maximum number of bidders have been added.

In the remainder of this section, we discuss Steps 1-3 in details.

Initialization

Similar to Vanderbeck [70], the generalized simplex algorithm used to tackle the column generation formulation must be started with a feasible primal solution. To find the feasible primal solution, we employ the versatile method based on [70], and it can be constructed heuristically. Alternatively, we can introduce artificial columns. The artificial columns are defined by their cost, constraint coefficient, and upper bound (c, a, μ). In this paper, we set the upper bound as 1, because it can reduce the possibility of degeneracy in the primal and multiple optimal solutions in the dual problem and hence stabilize the column generation procedure. Dual methods need not start with a primal feasible solution but they can benefit anyway from a warm start. Observe that using simplex re-optimization after adding a column can bring some form of stabilization towards the primal solution. Therefore, for the online advertising auction problem, we propose the following method to generate the feasible primal solution. For the query

$\ell \in S^t$, given the Lagrangian dual multipliers vectors $z^0, y^0 \geq 0$ (see [70, 49] for the details of estimate), we wish to find $\operatorname{argmin}_{j \in L^t_{\ell k}} \Big\{ z^0 + (y^0 - 1) b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t) \Big\}$ and record the best primal solution found if it has a negative Lagrangian cost where $L^t_{\ell k} \subseteq L^t_\ell$. This is because an explicit expression of L^t_ℓ may be computationally impossible when $|L^t_\ell|$ is huge. In practice, one works with a reasonably small subset $L^t_{\ell k} \subseteq L^t_\ell$ of columns with a restricted subproblem. The subset $L^t_{\ell k}$ maintains the same order as L^t_ℓ. The corresponding columns are introduced into the subproblem.

Sub-gradient Phase

In order to find the near-optimal multiplier vectors within a short computing time, we use the sub-gradient vectors. Clearly, an optimal solution to the above Lagrangian relaxation problem takes the following form:
$x^{\ell t}_{j_p} = 1$, if $z^t_\ell + (y_{j_p} - 1) b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t) < 0$,
$x^{\ell t}_{j_p} = 0$, if $z^t_\ell + (y_{j_p} - 1) b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t) \geq 0$.

Set $u^k_{j\ell t} = (z^{tk}_\ell, y^k_{j_p})^\top$. For $u^k_{j\ell t}$, we can get the optimal $(x_k, L(u^k_{j\ell t}))$ from the above subproblem, where $L(u^k_{j\ell t})$ represents the optimal value at x_k. For any j, compute

$$v^k_{j\ell t} = \left(\begin{array}{c} |I_{\ell t}| - \sum\limits_j (x^{\ell t}_{j_p})_k \\ b_j - \sum\limits_t \sum\limits_{\ell \in S^t} b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^{\ell}_{j_p}(S^t) (x^{\ell t}_{j_p})_k \end{array} \right).$$

This approach generates a sequence $v^k_{j\ell t}$. From the definition of multipliers, a possible choice consists of using the following updating formula $u^{k+1}_{j\ell t} = \max \Big\{ u^k_{j\ell t} + \lambda \frac{UB - L(u^k_{j\ell t})}{\|v\|^2} v^k_{j\ell t}, 0 \Big\}$, where UB is an upper bound on (6.6.1) and $0 \leq \lambda \leq 2$. One can iterate this procedure until the gap between UB and $L(u^k_{j\ell t})$ is small enough or for a predetermined number of iterations.

Remark 6.6.1 *Let m be the expected query number and $|S^t|$ be the expected number of keywords for any query. Similarly, $|L^t_\ell|$ is the expected number of bidders for the keyword $\ell \in S^t$. Each iteration of the sub-gradient phase requires (a) computing the Lagrangian costs associated with the current multiplier vector, which is done in $O(|L^t_\ell|)$ time; (b) computing the sub-gradient in $O(|L^t_\ell| \log |L^t_\ell| + |L^t_\ell|)$ time and (c) updating the multiplier* *vector in $O(m|S^t|)$ time. Since the maximum number of sub-gradient iteration allowed is $O(m|S^t|)$, the overall time complexity of this phase is $O(m|S^t|(|L^t_\ell|\log|L^t_\ell|+|L^t_\ell|))$.*

Column Generation Phase

For the (near) optimal Lagrangian multipliers $u^\star$, we compute the score function ranking the columns according to their likelihood to be selected in an optimal solution through the column generation algorithm:

1. We call $T \subseteq L^t_\ell$ to be the set of the currently not yet assigned bidders set for keyword ℓ of the corresponding query t, and M be the selected bidders set.
2. A column corresponding to one bidder j_p of L^t_ℓ can be profitably introduced into the model if
$$z^\star_{\ell t} + (y^\star_{j_p} - 1)b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^\ell_{j_p}(S^t) < 0 \quad \forall j^{\ell t}_p \in L^t_\ell, \ell \in S^t. \tag{6.6.2}$$
For each keyword $\ell \in S^t$, we seek to find
$$j^\star = \underset{j}{\operatorname{argmin}} \Big\{ z^\star_{\ell t} + (y^\star_{j_p} - 1)b^{\ell t}_{j_{p+1}} \frac{Q^{\ell t}_{j_{p+1}}}{Q^{\ell t}_{j_p}} n^{\ell t}_{j_p} P^\ell_{j_p}(S^t) \Big\}$$
overall $L^t_\ell \backslash M$. This term is also called the column generator. If a bidder is found such that (6.6.2) is satisfied, its variable and corresponding column are introduced.
3. $T := T\backslash\{j^\star\}, M := M \cup \{j^\star\}$ return to Step 2.
4. Until M is not modified.

For the bidders from M, we define $(x^{\ell t}_{j_p})^\star = 1$. Our aim is to obtain the better respective columns of bidders, who may be the most profitable one in the optimal solution, i.e., the scheme assigns a score to the chosen columns fixes to 1 the variables associated with the best-scored columns and re-optimizes the subproblem.

6.6.5 Numerical Results

In this section, our purpose is to verify the efficiency of the Lagrangian relaxation algorithm for solving the search engine advertising auction problem. We compare the numerical results between the Lagrangian relaxation and the linear programming relaxation. Furthermore, the numerical comparisons between the Lagrangian relaxation method and greedy algorithm for the search engine advertising auction problem are also investigated. The main idea of greedy algorithm is to allocate the slots to advertisers based on the bidding price from high to low for every keyword, regardless of another possible bidding price on the another keyword if this advertiser has more than one bidding keywords.

Table 6.6: Iterations Comparisons Between Linear Programming Relaxation and Lagrangian Relaxation Algorithms

Type	Reserve price	linear programming relaxation (Iters)	Lagrangian relaxation algorithm (Iters)
1	$r = 0.2$	18	29
	$r = 0.5$	27	12
	$r = 0.8$	31	32
2	$r = 0.2$	29	35
	$r = 0.5$	43	35
	$r = 0.8$	38	32
3	$r = 0.2$	29	39
	$r = 0.5$	30	34
	$r = 0.8$	61	31
4	$r = 0.2$	43	19
	$r = 0.5$	33	37
	$r = 0.8$	28	32
5	$r = 0.2$	48	38
	$r = 0.5$	50	37
	$r = 0.8$	62	37
6	$r = 0.2$	55	38
	$r = 0.5$	62	36
	$r = 0.8$	27	12

We generate the bidding price of every advertiser for several keywords, which follows the uniform distribution in the interval $[0, 1]$. We assume that the advertisers' quality scores are independently and identically drawn from a uniform distribution in the interval $[0, 10]$. Thus, the advertiser's rank number is the product of his bidding price and corresponding quality score. We assume the number of advertisement position provided by the search engine for every keyword is 10. The click through rates of advertisement slots follow the uniform distribution in the interval $[0, 100]$ in the decreasing sequence from slot 1 to slot 10. Moreover, budget of every advertiser follows the uniform distribution in the interval $[10000, 12000]$. In order to simulate the MNL model, we consider the case when the keyword number that every advertiser in a certain segment can bid is less than 10 and there exist different preference weights for different keywords. Every advertiser's preference weights for his choosing keywords follow a unform distribution in the interval of $[0, 10]$. During the sub-gradient phase, we set the step-size λ to be 1.5. As illustrated in Figure 6.1, we compare the numerical results for the following six cases, (i) 40 advertisers for bidding 25 keywords, (ii) 100 advertisers for bidding 50 keywords, (iii) 200 advertisers for bidding 50 keywords, (iv) 200 keywords for bidding 100 keywords, (v) 500 advertisers for bidding 100 keywords, (vi) 1000 advertisers for bidding 100 keywords, respectively. It is not difficult to find that the Lagrangian-based method shows significant improvements in the revenue of search engine compared to the greedy algorithm for every case. Figure 6.1 demonstrates that the revenue of search engine gained from the greedy algorithm is monotonically decreasing as the reserve price increases. Moreover, from Figure 6.1, we also observe the fact that the optimal reserve price is neither too high nor too low. The optimal reserve price also varies with the number of advertisers and the bidding prices of advertisers.

Table 6.6 demonstrates that taking the reserve prices 0.2, 0.5 and 0.8 as examples, the Lagrangian relaxation algorithm needs much fewer iterations than linear programming relaxation. As illustrated in Table 6.7, taking "1000 advertisers for bidding 100 keywords" for example, with optimal reserve price 0.84, the search engine can increase its revenue by 76.15% relative to the case with a zero reserve price. The search engine can improve its revenue by 118.18% when using the Lagrangian relaxation algorithm instead of greedy algorithm.

Remark 6.6.2 *In the simulation process, for simplicity in exposition, we assume that the parameters follow uniform distribution. In fact, the numerical results are basically*

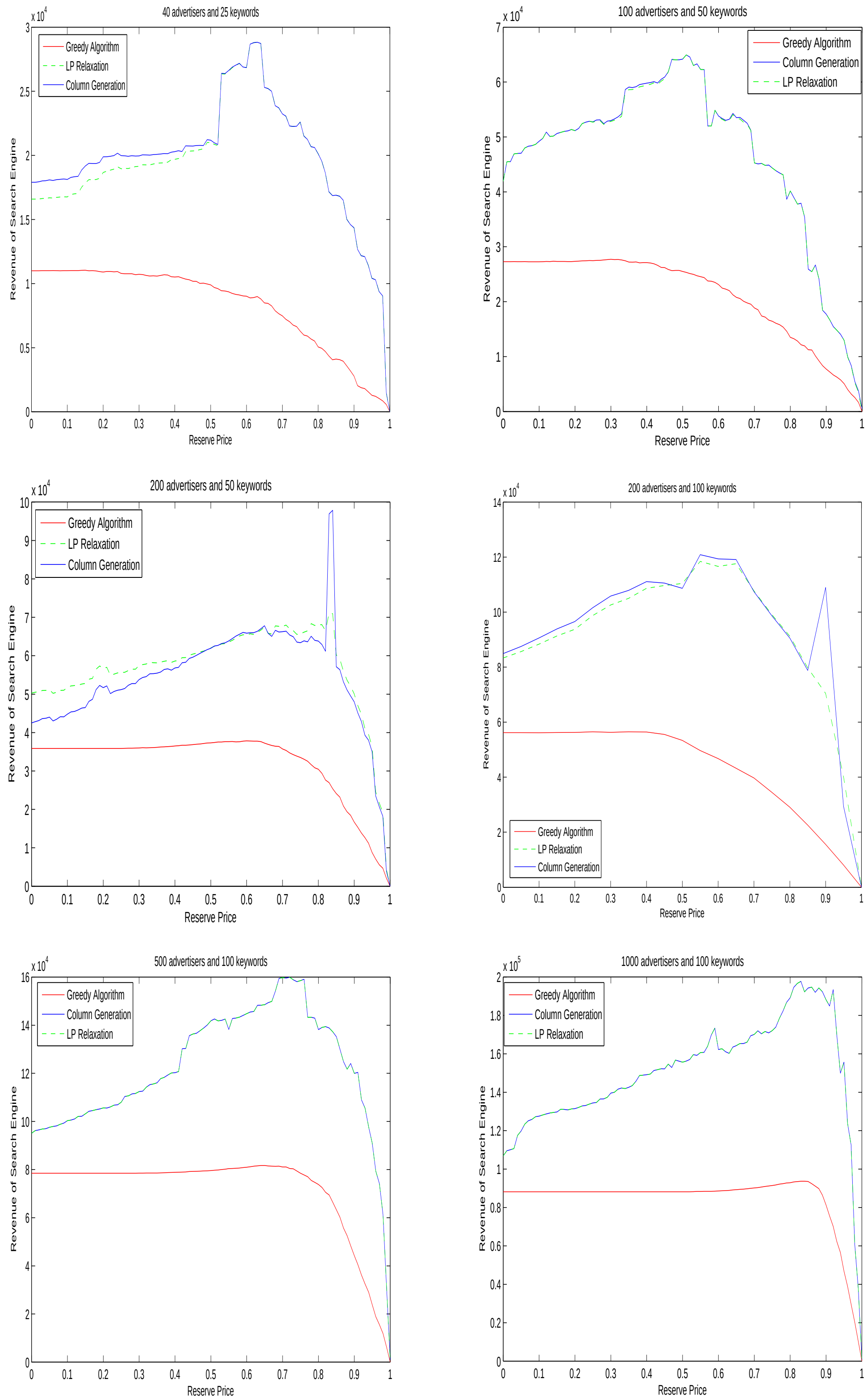

Figure 6.1: Comparisons between the Lagrangian relaxation method and the greedy algorithm

Table 6.7: Optimal Reserve Price, Percent Increase in Search Engine (SE) Revenue when Search Engines Set Optimal Prices

Type	Optimal Reserve Price	% Incr in SE compared with zero reserve price	% Incr in SE from Lagrangian relaxation method compared with greedy algorithm
1	0.65	57.14%	150%
2	0.52	52.94%	132.14%
3	0.85	40%	100%
4	0.56	41.46%	110.91%
5	0.75	68.42%	102.53%
6	0.84	76.15%	118.18%

similar even if we try other sets of parameters such as Gaussian distribution, exponential distribution.

Chapter 7

Conclusions and Future Work

In this thesis, primarily inspired by the literature [38], for nonlinear programming problems, we propose a new class of smooth exact penalty functions, which includes a unified framework both barrier-type and exterior-type penalty functions as special cases. We develop necessary and sufficient conditions for exact penalty property and local exactness properties, respectively. Furthermore, utilizing these conditions, we characterize the equivalence between the class of penalty functions and the classical simple exact penalty functions in the sense of exactness property. Based on the class of penalty functions, a class of feasible penalty function algorithms are presented. Under certain conditions, we present that the proposed algorithm terminates at the optimal solution to the primal problem after finite iterations and while under mild assumptions, the algorithm possesses globally convergent property. In addition, we design and apply new smooth and exact penalty functions for tackling the semi-infinite programming problems and the min-max programming problems. Here, the merit function is considered as a function of x and ε simultaneously which has good smoothness and exactness properties, without involving gradient and Jacobian matrices. We derive another useful property that the minimizer $(x^\star, \varepsilon^\star)$ of the penalty problem satisfies $\varepsilon^\star = 0$ if and only if $x^\star$ solves the original problem. This property demonstrates that the introduced new variable ε can be viewed as an indicator variable of a local (global) minimizer of primal problem. For the semi-infinite programming and min-max programming problems, the local exactness proofs are also shown. Furthermore, the second-order sufficient conditions for the local exactness properties are characterized for the proposed exact and smooth penalty function. As important applications, we solve an increasingly popular search engine advertising problem which stems from the online advertising auction via the new proposed penalty function.

Beyond these positive results and contributions, there are many other issues that are needed to deal with in the future work.

Applying our proposed exact penalty functions for solving large scale programming problems will be our main future research issue. In addition, we have characterized the equivalence between the new established class of penalty functions and the classical simple exact penalty functions in the sense of exactness property. Therefore, further investigation on the potential and hidden properties of the new exact and smooth barrier function could be also an interesting research topic in the future.

Bibliography

[1] T. Antczak. Exact penalty functions method for mathematical programming problems involving invex functions. *European Journal Operational Research*, 198: 29-36, 2009.

[2] M. S. Bazaraa, H. D. Sherali and C. M. Shetty. *Nonlinear Programming.* John Wiley and Sons, 2006.

[3] M. Ben-Akiva and S. Lerman. *Discrete Choice Analysis: Theory and Applications to Travel Demand*. The MIT Press, Cambrige, MA, 1994.

[4] J. K. Bonnans and A. Shapiro. *Perturbation analysis of optimization problems*. Springer-Verlag, Berlin, 2000.

[5] M. J. Box, D. Davies and W. H. Swann. *Nonlinear optimization techniques*. I. C. I. Monograph, Oliver and Boyd, Edinburgh, 1969.

[6] J. V. Burke. Calmness and exact penalization. *SIAM Journal on Control and Optimization*, 29(2): 493-497, 1991.

[7] J. V. Burke. An exact penalization viewpoint of constrained optimization. *SIAM Journal on Control and Optimization*, 29(4): 968-998, 1991.

[8] G. D. Camp. Inequality-constrained stationary-value problems. *Operation Research*, 3: 548-550, 1955.

[9] A. Caprara, M. Fischetti, P. Toth. A heuristic method for the set covering problem. *Operations Research*, 47(5): 730-743, 1999.

[10] C. W. Carroll. The created response surface technique for optimizing nonlinear restrained systems. *Operation Research*, 9: 169-184, 1961.

[11] A. Charnes, W. W. Cooper, K. O. Kortanek. Duality, Haar programs and finite sequence spaces. *Proceedings of the National Academy of Science*, 48: 783-786, 1962.

[12] F. H. Clarke. A new approach to Lagrange multipliers. *Mathematics of Operations Research*, 1(2): 165-174, 1976.

[13] R. Cominetti and J. P. Dussault. Stable exponential-penalty algorithm with superlinear convergence. *Journal of Optimization Theory and Applications*, 83: 285-390, 1994.

[14] A. R. Conn and N. I. M. Gpild. An exact penalty function for semi-infinite programming. *Mathematical Programming*, 37: 19-40, 1987.

[15] N. R. Devanur and T. P. Hayes. The adwords problem: Online keyword matching with budgeted bidders under random permutations. In EC'09: Proceedings of the 10th ACM conference in Electronic commerence, 71-78, 2009.

[16] R. DeVore and G. Lorentz. *Constructive Approximation*. Springer Verlag, New York, 1993.

[17] J. P. Dube, G. J. Hitsch and P. Manchanda. An empirical model of advertising dynamic. *Quantitative Marketing and Economics*, 3: 107-144, 2005.

[18] B. Edelman and M. Ostrovsky. Strategic bidder behavior in sponsored search auctions. *Decision Support Systems*, 43(1): 192-198, 2007.

[19] B. Edelman, M. Ostrovsky and M. Schwarz. Internet advertising and the generalized second-price auction: selling billions of dollars worth of keywords. *American Economic Review*, 97(1): 242-259, 2007.

[20] B. Edelman and M. Schwarz. Optimal auction design and equilibrium selection in sponsored search auctions. *American Economic Review*, 100(2): 597-602, 2010.

[21] J. P. Evans, F. J. Gould and J. W. Tolle. Exact penalty functions in nonlinear programming. *Mathematical Programming*, 4(1): 72-97, 1973.

[22] Y. Feng, B. Xiao, and W. Yang. Optimal reserve price in sponsored search advertising. The Chinese University of Hong Kong, Working paper, 2009.

[23] A. V. Fiacco and G. P. McCormick. *Nonlinear programming: sequential unconstrained minimization techniques*. Wiley, New York., 1968.

[24] R. Fletcher. An exact penalty function for nonlinear programming with inequalities. *Mathematical Programming*, 5(1): 129-150, 1973.

[25] R. Fletcher. *Practical Methods of Optimization* (second edition). John Wiley and Sons, New York, 1987.

[26] M. Gaudioso and M. F. Monaco. A bundle type approach to the unconstrained minimization of convex nonsmooth functions. *Mathematical Programming*, 23: 216-226, 1982.

[27] J. Gauvin. A necessary and sufficient regularity condition to have bounded multipliers in nonconvex programming. *Mathematical Programming*, 12: 136-138. 1977.

[28] C. J. Goh and K. L. Teo. Alternative algorithms for solving nonlinear function and functional inequalities. *Applied Mathematics and Computation*, 41: 159-177, 1991.

[29] R. E. Gomory. *An algorithm for integer solutions to linear programs*, in Recent Advances in Mathematical Programming,eds. Graves and Wolfe. McGraw Hill, 1963.

[30] C. C. Gonzaga and R. A. Castillo. A nonlinear programming algorithm based on non-coercive penalty functions. *Mathematical Programming*, 96: 87-101, 2003.

[31] S. P. Han and O. L. Mangasarian. Exact penalty functions in nonlinear programming. *Mathematical Programming*, 17(1): 251-269, 1979.

[32] S. P. Han and O. L. Mangasarian. A dual differentiable exact penalty function. *Mathematical Programming*, 25(3): 293-306, 1983.

[33] M. R. Hestenes. Multiplier and gradient method. *Journal of Optimization Theory and Applications*, 4(5): 303-320, 1969.

[34] R. Hettich and H. T. Jongen. *Semi-infinite programming: conditions of optimality and applications*, in optimization techniques (Proc. 8th IFIP conference Wrzburg, 1977), Part 2, Lecture Notes in Control and Information Science, Springer, Berlin, 7: 1-11, 1978.

[35] R. Hettich and K. O. Kortanek. Semi-infinite programming: Theory, methods and applications. *SIAM Review*, 35: 380-429, 1993.

[36] W. Hock, K. Schittkowski. *Test examples for nonlinear programming codes.* Springer-Verlag, New York, 1981.

[37] T. Hoheisel, C. Kanzowa, J. Outrata. Exact penalty results for mathematical programs with vanishing constraints. *Nonlinear Anal.*, 72: 2514-2526, 2010.

[38] W. Huyer and A. Neumair. A New Exact Penalty Function. *SIAM Journal on Optimization*, 13: 1141-1159, 2003.

[39] B. Jerez. General equilibrium with asymmetric information: A dual approach. *Journal of Economic Theory*, in press.

[40] H. T. Jongen, F. Twilt and G. W. Weber. Semi-infinite optimization: structure and stability of the feasible set. *Journal of Optimization Theory and Application*, 72: 529-552, 1992.

[41] Z. Katona and M. Sarvary. The race for sponsored links: Bidding patterns for search advertising. INSEAD, Working paper, 2008.

[42] W. Krabs. *Optimization and Approximation.* John Wiley and Sons, Chichester, New York, Brisbane, Toronto, 1979.

[43] A. H. Lan and A. Doig. An automatic method of solving discrete programming problems. *Econometrica*, 28: 497-520, 1960.

[44] W. Li, J. Peng. Exact penalty functions for constrained minimization problems via regularized gap function for variational inequality. *Journal of Global Optimization*, 37: 85-94, 2007.

[45] G. X. Liu. A homotopy interior point method for semi-infinite programming problems. *Journal of Global Optimization*, 37: 631-646, 2007.

[46] S. Lucidi, New results on a continuously differentiable exact penalty function. *SIAM Journal on Optimization*, 2: 558-574, 1992.

[47] O. L. Mangasarian and S. Fromovitz. The Fritz John necessary optimality conditions in the presence of equality and inequality constraints. *Journal of Mathematical Analysis and Applications*, 17(1): 37-47, 1967.

[48] G. Martin-Herrana, O. Rubelb and G. Zaccourc. Competing for consumer's attention. *Automatica*, 44: 361-370, 2008.

[49] M. Mourgaya and F. Vanderbeck. Column generation based heuristic for tactical planning in multi-period vehicle routing. *European Journal of Operational Research*, 183(3): 1028-1041, 2007.

[50] J. Nocedal and S. Wright. *Numerical Optimization*. Springer, New York, 1999.

[51] J. S. Pang. Error bound in Mathematical Programming. *Mathematical Programming*, 79: 299-332, 1997.

[52] G. Di Pillo. *Exact penalty methods*, in: E. Spedicato Ed.Algorithms for Continuous Optimization: the State of the Art. Boston: Kluwer Academic Press, 209-253, 1994.

[53] G. D. Pillo, L. Groppo. An exact penalty function method with global convergence properties for nonlinear programming problems. *Mathematical Programming*, 36: 1-18, 1986.

[54] G. Di Pillo, L. Grippo and S. Lucidi. A smooth method for the finite minimax problem. *Mathematical Programming*, 60: 187-214, 1993.

[55] G. D. Pillo, S. Lucidi. An augmented Lagrangian function with improved exactness properties. *SIAM Journal on Optimization*, 12: 376-406, 2001.

[56] E. Polak, D. H. Mayne and J. E. Higgins, Superlinearly convergent algorithm for min-max problems. *Journal of Optimization Theory and Applications*, 69: 407-439, 1991.

[57] E. Polak, D. Q. Mayne and D. M. Stimler. Control system design via semi-infinite optimization: a review. *Proceedings of the IEEE*, 72(12): 1777-1794, 1984.

[58] E. Polak, J. O. Royset and R. S. Womersley. Algorithms with adaptive smoothing for finite minimax problems. *Journal of Optimization Theory and Applications*, 119(3): 459-484, 2003.

[59] M. J. D. Powell. *A method for nonlinear constraints in minimization problem*, in Optimization ed. by R. Fletcher, Academic Press, New York, 283-298, 1969.

[60] R. Reemtsen. Some outer approximation methods for semi-infinite optimization problems. *Journal of Computational and Applied Mathematics*, 53: 87-108, 1994.

[61] R. T. Rockafellar. A dual approach to solving nonlinear programming problems by unconstrained optimization. *Mathematical Programming*, 5(1): 354-373, 1973.

[62] R. T. Rockafellar and R. J-B Wets. *Variational Analysis*. Grundlehren der Math. Wissenschaften 317, Springer Verlag, 1997.

[63] J. B. Rosen. The gradient projection methods for nonlinear programming, Part I-linear constraints. *SIAM Journal of Applied Mathematics*, 8: 181-217, 1960.

[64] P. Rusmevichientong and D. P. Williamson. An adaptive algorithm for selecting profitable keywords for search-based advertising services. In EC'06: Proceedings of the 7th ACM conference in Electronic commerence, 260-269, 2006.

[65] E. W. Sachs. *Semi-infinite programming in control*, in R. Reemtsen and J.-J. R*u*ckmann (Eds.), Semi-Infinite Programming, Kluwer Academic Publishers, Boston, MA, 389-411, 1998.

[66] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley and Sons, Chichester, 1986.

[67] G. Still. Discretization in semi-infinite programming: The rate of convergence. *Mathematical Programming*, 91: 53-69, 2001.

[68] H. Taha. *Integer programming*. Academic Press, Orlando, 1975.

[69] P. Tseng, D. P. Bertsekas. On the convergence of the exponential multiplier method for convex programming. *Mathematical Programming*, 60: 1-19, 1993.

[70] F. Vanderbeck. *Implementing mixed integer column generation*, in G. Desaulniers, J. Desrosiers, and M. M. Solomon, editors. *Column Generation*, Kluwer Academic Publishers, Boston, MA, 2005.

[71] H. R. Varian. Position auctions. *International Journal of Industrial Organization*, 25(6): 1163-1178, 2007.

[72] S. Wang, K. L. Teo and H. W. J. Lee. A new approach to nonlinear mixed discrete programming problems. *Enginerring Optimization*, 30: 249-262, 1998.

[73] Z. L. Wu and J. J. Ye. First-order and second-order conditions for error bounds. *SIAM Journal on Optimization*, 14: 621-645, 2003.

[74] S. Xu. Smoothing method for minimax problems. *Computational Optimization and Applications*, 20: 267-279, 2001.

[75] F. Ye, H. Liu, S. Zhou and S. Liu. A smoothing trust-region Newton-CG method for minimax problem. *Applied Mathematics and Computation*, 199(2): 581-589, 2008.

[76] W. Zangwill. Non-linear programming via penalty functions. *Management Science*, 13: 344-358, 1967.

[77] A. J. Zaslavski. Existence of exact penalty for constrained optimization problems in Hilbert spaces. *Nonlinear Analysis*, 67: 238-248, 2007.

[78] J. L. Zhou and A. L. Tits. Nonmonotone line search for minimax problems. *Journal of Optimization Theory and Applications*, 76: 455-476, 1993.

[79] J. L. Zhou and A. L. Tits. An SQP algorithm for finely discretized continuous minimax problems and other minimax problems with many objective functions. *SIAM J. Optimization*, 6(2): 461-487, 1996.

[80] G. Zoutendijk. *Methods of feasible directions*. Elsevier, Amsterdam, 1960.