Chapter 1

Introduction

Optimization, in simplest terms, is the act of making the best of anything. Broadly speaking, optimization seeks to change an existing process to increase the occurrence of favorable outcomes and decrease the occurrence of undesirable results. Concerning mathematics, optimization adopts the most significant component of a particular criterion from a set of viable alternatives. Mathematical optimization aims to find a combination of input variables that maximizes or minimizes the output return of a multivariable function. These days mathematical optimization has been transformed into an innovatory tool for powerful modeling and decision-making occurrences in all quantitative disciplines, from computer science and engineering to operations research and economics.

Mathematically, optimization models comprise three significant components: decision variables, objective function, and constraints.

- *Decision variables* designate a value that may vary within the scope of a given optimization problem.

- In a mathematical optimization problem, the *objective function* expresses the problem's main criteria, whose value is either minimized or maximized over the set of feasible alternatives.

- *Constraints* are the logical conditions or allowable values or scopes for the variables in an optimization problem that the solution of a given problem must satisfy.

In an optimization problem, the types of mathematical relationships among the decision variables, the objective function, and the constraints determine how intricate it is to operate and the algorithms that can be used for optimization. There are numerous applications of optimization theory and methods in the fields of applied math, computation math, and operations research, including science, engineering, business management, military, and space technology. It involves

- the construction of model problems,
- the study of optimality conditions of the problems,
- the determination of the algorithmic method of the solution,
- the establishment of convergence theory of the algorithms, and
- numerical experiments with typical and real-life problems.

One of the most common and primary problems in scientific research and engineering practice are optimization problems. According to the number of the optimized objective function, optimization problems can be classified into two categories: single objective and multiobjective optimization problems. Although a substantial amount of research in optimization is conducted concerning single-objective problems, optimization problems with more than one objective are inevitable in various fields, such as engineering design [1], optimal control systems [2], chemical engineering [3], machine learning [4], and many more. For multiobjective optimization problems (MOPs), two or more objective functions need to be optimized simultaneously. MOPs usually involve conflicting objective functions, i.e., an optimal solution for one function will probably not be optimal for another function or other functions. Therefore, it is challenging for an MOP to devise a solution that optimizes all the objective functions. In this situation, a set of trade-off solutions are studied for MOPs. These solutions generally known as Pareto optimal solutions.

In the next section, the fundamentals of MOPs are discussed.

1.1 Mathematical formulation of multiobjective optimization problem

In general, an MOP contains n number of decision variables, $\mathcal{P}$ number of objective functions, $\mathcal{J}$ number of inequality constraints, and $\bar{\mathcal{J}}$ number of equality constraints. Without loss of generality, a minimization MOP is defined as:

$$\left.\begin{aligned} &\text{minimize} \quad F(x) = \left(f_1(x), f_2(x), \ldots, f_{\mathcal{P}}(x)\right)^\top,\ \mathcal{P} \geq 2 \\ &\text{subject to} \quad h_j(x) \geq 0,\ j = 1, 2, \ldots, \mathcal{J}, \\ &\qquad\qquad\quad g_\ell(x) = 0,\ \ell = 1, 2, \ldots, \bar{\mathcal{J}}, \end{aligned}\right\} \tag{1.1}$$

where $f_i : \mathbb{R}^n \to \mathbb{R}$, $h_j : \mathbb{R}^n \to \mathbb{R}$ and $g_\ell : \mathbb{R}^n \to \mathbb{R}$ for all $i \in \{1, 2, \ldots, \mathcal{P}\}$, $j \in \{1, 2, \ldots, \mathcal{J}\}$ and $\ell \in \{1, 2, \ldots, \bar{\mathcal{J}}\}$. If $\mathcal{J} = \bar{\mathcal{J}} = 0$, then MOP (1.1) is reduced to an unconstrained MOP. Throughout the thesis, we denote $\mathcal{I} = \{1, 2, \ldots, \mathcal{P}\}$.

We refer to $x = (x_1, x_2, \ldots, x_n)^\top$ as the vector of decision variables and the set $\mathcal{X} = \{x \in \mathbb{R}^n : h_j(x) \geq 0,\ j = 1, 2, \ldots, \mathcal{J},\ g_\ell(x) = 0,\ \ell = 1, 2, \ldots, \bar{\mathcal{J}}\}$ as the feasible set in decision space. We represent the image of $\mathcal{X}$ under the vector-valued objective function F by $\mathcal{Y}$, i.e., $\mathcal{Y} = F(\mathcal{X}) = \left\{\left(f_1(x), f_2(x), \ldots, f_{\mathcal{P}}(x)\right)^\top : x \in \mathcal{X}\right\}$. The set $\mathcal{Y}$ is referred as the feasible set in objective space. Note that $\mathcal{X} \subseteq \mathbb{R}^n$ and $\mathcal{Y} \subseteq \mathbb{R}^{\mathcal{P}}$. Geometrical view of the sets $\mathcal{X}$ and $\mathcal{Y}$ for bi-objective optimization problem with three decision variables are shown in Figure 1.1.

Linear and nonlinear MOP: If all the objective functions and constraints are linear of MOP (1.1), then problem is named as linear multiobjective optimization problem. In

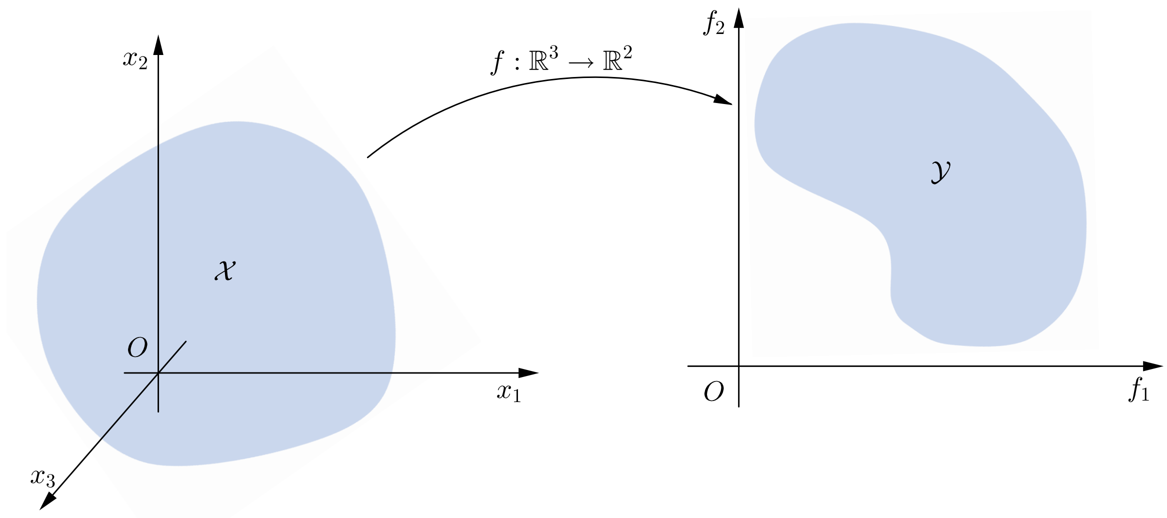

Figure 1.1: Feasible decision region (left) and feasible objective region (right)

contrast, if one or more objective functions and/or constrains are nonlinear then (1.1) is nonlinear MOP.

Convex and nonconvex MOP: MOP (1.1) is called convex if its objective functions and feasible region are convex. Otherwise, MOP (1.1) is called nonconvex.

When dealing with MOP (1.1), it is essential to explain what 'solving' an MOP means. In the next section, few definitions are presented to describe the concept of solution for (1.1).

1.2 Pareto optimality

In MOPs, there are rare problems in which only a single point simultaneously can optimize all the objective functions. Therefore, we normally look for "trade-offs", rather than single solutions when dealing with MOPs. The notion of "optimality" is therefore, different from single objective optimization problems in MOPs. The most commonly adopted notion of optimality is originally proposed by Francis Ysidro Edgeworth and later generalized by Vilfredo Pareto. The definition of Pareto optimality is based on a dominance relation in $\mathbb{R}^{\mathcal{P}}$. For the required dominance relation for Pareto optimality, we use the following notations ($\mathcal{P} > 1$):

- $\mathbb{R}^{\mathcal{P}}_{\geqq} = \{y \in \mathbb{R}^{\mathcal{P}} : y \geqq 0\}$, the nonnegative orthant of $\mathbb{R}^{\mathcal{P}}$; by $y \geqq 0$ we mean all the components of y are nonnegative.
- $\mathbb{R}^{\mathcal{P}}_{\geq} = \{y \in \mathbb{R}^{\mathcal{P}} : y \geq 0\}$, where $y \geq 0$ denotes $y \geqq 0$ but $y \neq 0$.
- $\mathbb{R}^{\mathcal{P}}_{>} = \{y \in \mathbb{R}^{\mathcal{P}} : y > 0\}$ represents the interior of $\mathbb{R}^{\mathcal{P}}_{\geqq}$, where $y > 0$ means all the components of y are positive.
- The relations $\leqq, \leq$ and $<$ can also be defined in a similar way.
- For two vectors $y^1, y^2 \in \mathbb{R}^{\mathcal{P}}$, we say that the vector y^1 dominates y^2, for a minimization problem, if $y^1 \leq y^2$.

Definition 1.1 (Pareto optimality (*e.g.* [5])). *A feasible solution $\hat{x} \in \mathcal{X}$ is called efficient or Pareto optimal if there is no $x \in \mathcal{X}$ such that $F(x) \leq F(\hat{x})$. If $\hat{x}$ is efficient, $F(\hat{x})$ is called a nondominated point.*

The set of all efficient solutions of the MOP (1.1) is denoted by $\mathcal{X}_E$. The set of all nondominated points is represented by $\mathcal{Y}_N$. Evidently, $\mathcal{Y}_N = F(\mathcal{X}_E)$.

Definition 1.2 (Weak Pareto optimality (*e.g.* [5])). *A feasible solution $\hat{x} \in \mathcal{X}$ is called weakly efficient or weakly Pareto optimal if there is no $x \in \mathcal{X}$ such that $F(x) < F(\hat{x})$. The point $\hat{y} = F(\hat{x})$ is then said to be weakly nondominated.*

The set of all weakly efficient solutions of the MOP (1.1) is denoted by $\mathcal{X}_{wE}$. The collection of all weakly nondominated points is represented by $\mathcal{Y}_{wN}$.

The solution notion for MOP (1.1) can also be defined with respect to an ordering cone $\mathbb{R}^{\mathcal{P}}_{\geqq}$ which is used for ordering the criterion space $\mathbb{R}^{\mathcal{P}}$. Utilizing the ordering cone $\mathbb{R}^{\mathcal{P}}_{\geqq}$, a feasible solution $\hat{x} \in \mathcal{X}$ is called an efficient solution of MOP (1.1) if $\left(F(\hat{x}) - \mathbb{R}^{\mathcal{P}}_{\geqq}\right) \bigcap F(\mathcal{X}) = \{F(\hat{x})\}$. Likewise, we say $\hat{x} \in \mathcal{X}$ is a weakly efficient solution of MOP (1.1) if $\left(F(\hat{x}) - \mathbb{R}^{\mathcal{P}}_{>}\right) \bigcap F(\mathcal{X}) = \emptyset$.

Apparently, every efficient point is also weakly efficient [5]. The two concepts are illustrated in the feasible objective space ($\mathcal{Y}$) for a bi-objective optimization problem ($\mathcal{P} = 2$) in Figure 1.2. It shows an efficient point $\hat{x}^1$ and a weakly efficient point $\hat{x}^2$.

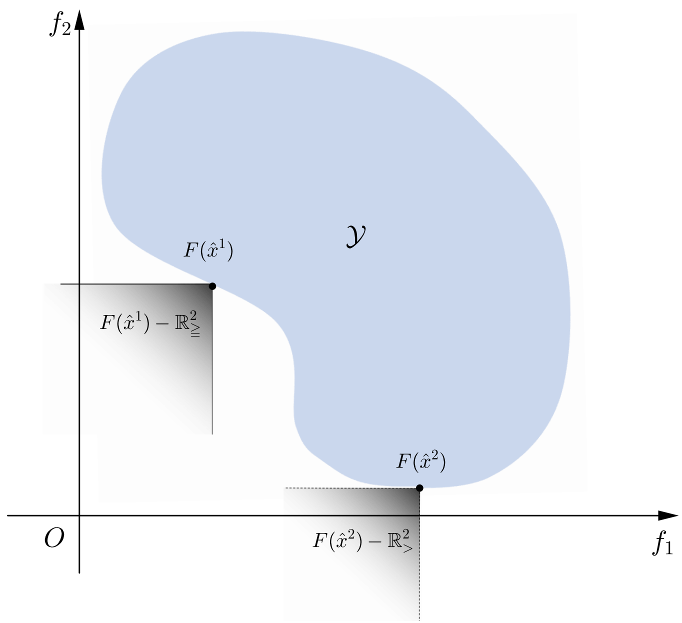

Figure 1.2: Nondominated and weakly nondominated points

The ideal point (also known as utopia point) and the nadir point are usually referenced in multiobjective optimization methods. Formal definitions of these are defined in Definitions 1.3 and 1.4, respectively [5].

Definition 1.3 (Ideal point (*e.g.* [5])) *The point* $y^I = \left(y_1^I, y_2^I, \ldots, y_{\mathcal{P}}^I\right)$ *given by* $y_i^I = \min_{x \in \mathcal{X}} f_i(x)$ *is ideal point (or utopia point) of the MOP* (1.1)*. Also, a hyperplane that contains all the utopia points is called the utopia hyperplane.*

Definition 1.4 (Nadir point (*e.g.* [5])) *The point* $y^N = \left(y_1^N, y_2^N, \ldots, y_{\mathcal{P}}^N\right)$ *given by* $y_i^N = \max_{x \in \mathcal{X}} f_i(x)$ *is nadir point of the MOP* (1.1)*.*

Ideally, solving MOP (1.1) means to obtain the entire efficient set, i.e., all the efficient points, and its corresponding nondominated set. However, it is difficult to figure out an accurate description of the efficient or nondominated set for most MOPs.

This is because these sets frequently have an infinite number of points. As far as we are aware, only two exact general methods, namely, two interval branch-and-bound methods [6, 7], have been proposed in the literature which obtain an enclosure of those sets up to a pre-specified precision. Particularly, they present a list of boxes (multi-dimensional intervals) whose union contains the complete efficient set (and their images, the corresponding nondominated set) as a solution. However, they are time consuming. Furthermore, they have large memory requirements, hence, only small instances can be solved with them. The common approach in the literature is to approximate the nondominated set by a finite set of points. Multiobjective optimization methods try to get solutions that are as close as possible to the nondominated set and also uniformly distributed. Such methods will have good convergence and diversity. For this purpose, several performance metrics have been defined (see [8]). As it is difficult to evaluate the performance metrics of any algorithm theoretically, researchers generally use test functions to verify the algorithm's performance.

In the next section, we discuss about some popular performance metrics and test functions.

1.3 Performance metrics and test functions

To compare the performance of MOPs algorithms, the quality of Pareto optimal sets is measured in terms of various performance metrics [8]. These metrics create unique values, and in this way, the algorithms can be compared. According to Okabe et al. [9], performance metrics consider the following two main aspects of a solution set:

- the convergence of the solution set; and
- the diversity, distribution as well as spread.

According to Riquelme et al. [8], hypervolume (HV) is the most used metric, followed by the generational distance (GD) and the inverted generational distance (IGD).

1.3.1 Hypervolume (HV)

HV, also known as S metric, hyper-area, or Lebesgue measure, is the volume of an approximation set relative to some reference points [8]. HV takes accuracy, diversity, and cardinality into account during calculations. A set with a larger HV value will likely present a better solution than the sets with a lower HV value. Let $R(\bar{x}) = (f_1(\bar{x}), f_2(\bar{x}), \ldots, f_{\mathcal{P}}(\bar{x}))$ be a reference point such that $R(\bar{x}) \leq F(x)$ for all $x \in \mathcal{X}$. Suppose that the solution set is sorted by increasing order with respect to f_1. Then, HV is calculated by

$$\text{HV} = \sum_{j=0}^{\mathcal{J}-1} \prod_{i=1}^{\mathcal{P}} \left(f_i(\bar{x}) - f_i(x_{j+1}) \right). \tag{1.2}$$

1.3.2 Generational distance (GD)

Among the existing metrics, GD is a pure measure of convergence. The GD metric captures the average distance between each element of the obtained nondominated set and its closest neighbor in a discrete representation of the true nondominated set. In order to calculate GD, we consider the solution set $\{x_1, x_2, \ldots, x_N\}$, where N represents the number of obtained nondominated points. Then, the following formula determines the GD:

$$GD = \frac{1}{N} \sqrt{\sum_{i=1}^{\mathcal{P}} \| F(x_i) - F(x_i^*) \|^2}, \tag{1.3}$$

where $F(x_i^*)$ is the i-th Pareto efficient point that is closest to $F(x_i)$.

1.3.3 Inverted generational distance (IGD)

Another commonly used performance measure is IGD. IGD is calculated as follows

$$\text{IGD}(G, G^*) = \frac{\sum_{x \in G^*} d(x, G)}{|G^*|}, \tag{1.4}$$

where the set G consists the approximation of the Pareto front and the set G^* contains uniformly distributed known nondominated points.

Zitzler et al. [10] created a test problems suite called ZDT test function set, which consists of six problems ZDT1–ZDT6. These problems are bi-objective optimization problems with different forms of expression and properties. Because their Pareto fronts are known, they are one of the most commonly used test problems. Therein, ZDT1 and ZDT4 are convex functions, while ZDT2 and ZDT6 are concave functions, ZDT3 is a non-continuous function, ZDT4 is a multi-modal function and ZDT5 is a function with the deceptive property.

Deb et al. [11] constructed a DTLZ test function set, which allowed the decision variables and objective functions to extend to any dimension. The DTLZ test function set includes seven unconstrained optimization problems, DTLZ1–DTLZ7, and two constrained optimization problems, DTLZ8–DTLZ9. They are also widely used for testing the performance of optimization algorithms. There are also some other common test problems like Schaffer's study (SCH) [12], Fonseca and Fleming's study (FON) [13], etc.

In the next section, we discuss the existing methods to solve MOPs in the literature.

1.4 Methods to solve MOPs

1.4.1 Classical methods

In the recent past, many researchers have started extending the classical methods of scalar optimization to solve unconstrained MOPs. These extensions include steepest descent [14,15], conditional gradient [16], Newton method [17,18], quasi-Newton method [19], conjugate gradient [20, 21], projected gradient [22, 23] proximal methods [24] and trust-region methods [25, 26]. A very short details of these papers are as follows.

Fliege and Svaiter [15] introduced the steepest descent for both constrained and unconstrained MOPs. In [15], the objective function is considered to be continuously differentiable for unconstrained problems, and Lipschitz-continuously differentiable for a constrained problem.

The conditional gradient method or Frank-Wolfe method was proposed in [16] for constrained MOPs. In [16], the objective functions are considered continuously differentiable, and the constraint set is assumed convex and compact.

In the extension of Newton's method [17] and quasi-Newton's method [19] for unconstrained MOPs, the objective functions are considered twice continuously differentiable with the local strongly convex property.

Lucambio Pérez and Prudente [21] generalize various conjugate gradient methods, such as the Fletcher-Reeves, conjugate descent, Dai-Yuan, Polak-Ribiére-Polyak, and Hestenes-Stiefel methods for continuously differentiable MOPs.

In [22], an extension of the projected gradient method has been introduced for MOPs. In [22], vector-valued functions are used directly instead of scalar-valued objectives.

In the extension of the proximal point method [24] for scalar-valued convex optimization, the subproblems consist of finding weakly efficient points for suitable regularizations. In [24], an exact and an inexact version of subproblems are solved approximately.

The above discussed approaches are line search techniques. Trust-region methods are also developed to solve unconstrained MOPs in [25, 26]. In [26], the trust-region method is established for smooth and nonsmooth MOPs with convergence results. Trust-region methods developed in [25] is able to handle convex and nonconvex MOPs.

1.4.2 Scalarization techniques

Apart from the classical methods, another common approach towards constrained and unconstrained MOPs is to combine the objective functions and form single objective optimization subproblems which can be solved using conventional optimization techniques. This is known as scalarization [5, 27] or decomposition [28]. There are several scalarization techniques in the literature, with the most popular being as follows: weighted sum [17,29,30], adaptive weight sum method [31,32], ε-constraint [15,33], normal constraint [34], normal boundary intersection [20, 35], physical programming [36], directed search domain [29]. A brief discussion of the above scalarization techniques are as follows.

The weighted sum method builds a single objective as a weighted sum of all the objective functions. Thereafter, using different weights, the problem can be solved repeatedly to approximate the nondominated set. This method cannot find all the solutions on the trade-off surface when the Pareto surface is nonconvex [37–39].

Like the weighted sum method, adaptive weighted sum (AWS) method [31] employs a similar mathematical formulation. It also performs like weighted sum method with fewer weights in its first step (or iteration). The main feature of this method is that it focuses on the nonconvex region of the nondominated set by specifying additional constraints for the part where further refinement is needed. Based on a distance criterion, the subproblem is repeated until a desired solution distribution density is achieved. There is a drawback to this method as it relies on solutions derived from the initial weighted sum step. If the weighted sum method cannot find an effective Pareto solution in the first iteration, the AWS method may fail to determine the entire nondominated set.

Another traditional method from the field of multiobjective optimization to generate the nondominated points is the ε-constraint method [15, 33]. The ε-constraint method works by choosing one objective function as the only objective and the remaining objective functions as constraints. Different elements of the nondominated set can be obtained by a systematic variation of the constraint bounds. The idea of the ε-constraint method is to iteratively increase the constraint bound by a predefined constant ε. The necessity to choose such a value represents the main drawback of this approach.

The normal boundary intersection (NBI) method was proposed by Das and Dennis [40]. It is based on the well-known fact that a nondominated set is related to the boundary of a feasible objective region towards the minimization of objective functions [41]. Geometrically, this method constructs the nondominated set by the intersection of lines, which are normal to the utopia hyperplane, and the boundary of the feasible objective region. Mathematically, additional equality constraints are imposed in the feasible objective region so that the solution lie on a line that is normal to the utopia hyperplane. The distance away from the utopia hyperplane is then maximized (towards the utopia point). One drawback of the this method is that dominated solutions can be obtained as optimal solutions because the algorithm finds a solution regardless of whether the point is dominated or not. Another drawback of this method is that it introduces equality constraints, which are usually more difficult to handle numerically.

The normal constraint (NC) method was proposed by Messac et al. [34] in 2003 and Messac and Mattson [42] in 2004 as an improvement over the normal boundary intersection method. The single objective optimization problem, used in the NC, is based only on inequality constraints. An inequality constraint is applied perpendicular to the utopia hyperplane in the normalized space. One of the two objective functions is then optimized, depending on the direction of the inequality constraint; this limits the feasible domain. As the problem progresses, the feasible domain is gradually reduced. With the exception of the weighted sum method, the NC method was found to be one of the simplest and fastest to implement and understand. However, the method has the potential to find dominated solutions.

Physical programming method was introduced by Messac [36]. This method generates convex and nonconvex nondominated points [43]. Additionally, it does not use any weight coefficients and allows one to take into account the decision maker experience immediately. The algorithm given by Messac and Mattson [43] is able to generate an evenly distributed Pareto set. However, it contains a few free parameters whose optimal choice requires prior information about the Pareto set.

Directed search domain method was first presented and applied for the modification of the physical programming method (see [44]). It introduces a search domain based on the local linear transformation of objective functions and searches for the solution within each domain. To guarantee a well distributed Pareto set, it evenly spreads local search domains. In the case of a nonconvex boundary, it is possible not to obtain any feasible solution in the search domain. In this case, the search domain should be flipped to the opposite side of the utopia hyperplane to capture the points on the nondominated set. This method may bring some difficulties due to introducing new coordinate systems.

The motivation and purpose of this thesis are detailed in the next section.

1.5 Motivation and objective of the thesis

Due to the complex nature of the single objective optimization subproblems that are obtained by scalarization technique, stochastic search methods [45] are commonly employed to obtain the solutions. While performing these algorithms from one run to another the obtained solutions of single objective optimization subproblems may be differ widely. Also, these methods do not guarantee that the final solutions will be Pareto optimal solutions.

Using deterministic numerical optimization approaches with effective scalarization techniques can yield great benefits in solving MOPs. Motivated by this fact, we propose a novel decomposition aided IPMs-based approaches for solving MOPs with twice continuously differentiable F, h, and g.

It is reported in [46] that the scalarization techniques those are mentioned in Section 1.4.2 either cannot find all nondominated solutions or need some prior information about the location of the nondominated set. Recently, Ghosh and Chakraborty [47] introduced a nondominated set generating method, known as the cone method [47], for MOPs. The cone method is theoretically (i.e., in the limit) able to retrieve all the nondominated points (which are infinite, in general). However, in practice, as we need to employ a direction-based discretization to solve an MOP by the cone method, only a discrete representative of the whole nondominated set is obtained. The formulation of the cone method is similar to the Pascoletti-Serafini technique [48] for vector optimization. The cone method has the ability to generate both convex and nonconvex parts of the nondominated set. Although the formulation of the cone method is detailed in [47], how to effectively solve the set of direction-based parametric problems of the cone method is not given therein.

Interior-point methods (IPMs) are the most efficient for solving linear, convex, and nonconvex optimization problems. IPMs are also well suited to large-scale optimization since they feature a consistently small number of iterations needed to reach the optimal solution of the problem. Furthermore, the number of iterations do not increase significantly, even for the problems with millions of variables.

In this thesis, we develop IPMs to solve the set of direction-based parametric problems of the cone method.

In the next section, the review of IPMs is detailed.

1.6 Literature review of interior-point methods

An area of mathematical programming known as linear programming (LP) deals with minimizing or maximising a linear function under linear constraints. Equalities and inequalities are examples of these constraints. The well-known simplex method of solving LP problems was proposed by Dantzig in 1947. Using an LP model, Dantzig proposed a solution for finding the best assignment of 70 people to 70 jobs (see [49]).

The optimal solution of an LP always lies at a vertex of the feasible region. The simplex method proceeds from one vertex to a neighboring vertex until it reaches the optimal one. In practice, because the simplex method routinely and efficiently solved very large linear programs, it retained unquestioned preeminence as the solution method of choice. However, the simplex method was viewed with nagging discontent by those interested in computational complexity (a field whose importance increased during the 1960s and 1970s). An underlying tenet of theoretical computer science is that any "fast" algorithm must be polynomial-time, meaning that the number of arithmetic operations required to solve the problem should be bounded above by a polynomial in the problem size.

Although the simplex method almost always solves the real-world problems in a number of iterations, i.e., a small multiple of the problem dimension, it is shown by Klee and George Minty that the simplex method can visit every vertex of the feasible region (see [50]). In the worst case, Klee and Minty proved that the simplex method takes an exponential number of iterations to reach the optimal solution. In their problem (1.5) with n variables, n restrictions and $2n$ vertex, they showed that the algorithm must visit every vertex before reaching the optimal solution (see [51]).

$$\left.\begin{aligned} &\text{Maximize} \quad 100\sum_{j=1}^{n} 10^{n-1}x_j \\ &\text{subject to} \quad 2\sum_{j=1}^{i-1} 10^{i-j}x_j + x_i \leq 100^{i-1}, \quad i = 1, 2, \ldots, n, \\ &\qquad\qquad\quad x_j \geq 0. \end{aligned}\right\} \tag{1.5}$$

In 1979, Khachiyan presented a polynomial algorithm, namely the ellipsoid method, to solve LP (see [52]). Khachiyan showed that the ellipsoid method had a polynomial complexity of order $(nm^3 + m^4)L$, where $m, \ n, \ L$ denotes the number of rows in a LP formulation, the number of columns and the length of the data, respectively [53]. Khachiyan method was studied intensively by practitioners and theoreticians. The best implementation of the ellipsoid method was not competitive with the simplex method as expected to be faster with LP polynomials than with simplex (see [54] for survey).

In late 1984, A polynomial time algorithm proposed by Karmarkar [55] for LP showed promising results in practice. The idea of this algorithm was totally different compared to the simplex method. In contrast to the simplex method, iterates are calculated within the feasible region instead of at the boundary. The complexity of the Karmakar's method is of order $(nm^2 + m^3)L$ (see [53]). The interior-point method that Karmakar developed belongs to a class of methods known as the interior-point method. These methods have been proved to be competitive with the simplex method and usually superior on very large problems.

Hundreds of polynomial time IPM variants were developed for LP. Interior point software implementations have challenged simplex method implementations and frequently surpassed their performance (Bixby [56]). All aspects of linear optimization had to be revisited. Not only the research on algorithms exploded, but research on duality theory (Roos et al. [57], Terlaky [58]) and implementation strategies (Gondzio and Terlaky [59], Andersen et al. [60]) got new impulse. The significance and applications of a strictly complementary solution were explored (Greenberg [61]), and novel concepts of sensitivity analysis were introduced (Jansen et al. [62], Koltai and Terlaky [63], Ghaffari Hadigheh et al. [64], Roos et al. [57]).

A decade of turbulent research developed a good understanding of the fundamentals of IPMs. From 1994 on, Many books have been published that summarize and explore different aspects of IPMs. IPMs with primal and dual path-following are examined extensively by Den Hertog [65] for linear and structured convex optimization problems. Nesterov and Nemirovski [66] developed the most general framework for polynomial time IPMs for smooth convex optimization problems. Jansen [67] discusses primal-dual target-following algorithms for linear optimization and complementarity problems. Saigal's [68] book deals with affine scaling algorithms. The volume edited by Terlaky [69] contains 13 survey papers that cover most of the key aspects of IPMs, their extensions, and some applications. Wright [70] concentrates on primal-dual infeasible IPMs, numerical issues, and local and asymptotic convergence properties. The book of Ye [71] is a rich source of polynomial IPMs not only for LP, but for convex optimization problems as well. It extends the IPM theory to derive bounds and approximations for classes of nonconvex optimization problems as well. Finally, Roos et al. [57] present a thorough treatment of the IPM-based theory—duality, initialization, complexity, sensitivity analysis, and implementation strategies, and large classes of IPMs for LP.

IPMs not only provide polynomial time algorithms and a powerful methodology for LP, but they are generalized to solve large classes of optimization problems. IPMs were generalized to solve conic linear, in particular, semidefinite and second-order conic optimization problems (Alizadeh [72], Alizadeh and Goldfarb [73], Nesterov and Nemirovski [66]). IPM methodology allowed to solve large classes of novel engineering and control problems (Ben-Tal and Nemirovski [74], Boyd and Vandenberghe [75], Pólik and Terlaky [76]), as well as solve, or approximately solve, combinatorial optimization problems (Alizadeh [72, 77], Goemans and Williamson [78]). A rich theory of IPMs for smooth nonlinear optimization problems was developed from the early days of IPMs (Nesterov and Nemirovski [66], Renegar [79]).

Now, all major commercial optimization software systems provide implementations of IPMs in order to solve large-scale, sparse, structured linear problems. The software implementations of IPMs for nonlinear optimization problems can be found in Andersen and Ye [80], Wächter [81], Wächter and Biegler [82], Byrd et al. [83], and Vanderbei [84].

1.7 Organization of the thesis

The thesis is composed of seven chapters, including an introductory chapter and a chapter containing a conclusion and future scopes. In the introductory chapter, a concise but adequate literature review has been provided on these topics. The outline of the thesis is as follows.

In Chapter 2, an infeasible interior-point technique is proposed to generate the nondominated sets of nonlinear multiobjective optimization problems with the help of the direction-based cone method. This method deals with both convex and nonconvex MOPs. The convergence analysis of the method and an estimate of the number of iterations to reach an ϵ-precise solution are also provided. Performance comparison between the proposed method and popular existing solvers is provided with respect to two performance measures and the corresponding relative efficiency measures.

In Chapter 3, a Newton-type globally convergent interior-point technique for detecting the nondominated points of multiobjective optimization problems using the direction-based cone method is introduced. Under some mild conditions, the proposed algorithm is shown to be globally convergent. Numerical results of unconstrained and constrained multiobjective optimization test problems are presented.

Chapter 4 deals with a primal-dual interior-point technique to solve multiobjective optimization problems with an application to optimal control problem. To demonstrate the efficiency of the proposed method, we applied it to some constrained test problems. As an application, we use proposed algorithm to an optimal control problem of carbon dioxide emission from energy sector.

Chapter 5 demonstrates a trust-region interior-point technique to solve multiobjective optimization problems and an application to a tuberculosis optimal control problem. To show the efficiency of the proposed method, we show its performance on some standard test problems. As an application, we apply the proposed algorithm to solve an optimal control problem for a tuberculosis model.

Chapter 6 presents a new trust-region method to obtain the Pareto critical points of unconstrained nonsmooth multiobjective optimization problems. The gradients of the each objective function are approximated with the help of ϵ-subgradients. Also, the second-order informations of each objective function are approximated by using quasi-Newton methods. Under some mild assumptions on the objective functions and using the BFGS method to update the second order information of the objective functions, we prove the convergence of that the proposed algorithm. At last, the proposed algorithm is tested on some nonsmooth examples through the MATLAB.

Finally, Chapter 7 summarizes the main conclusions and forecasts potential directions for future research.

****