A class of Increasing Positively Homogeneous functions for which global optimization problem is NP-hard

by

Nargiz Sultanova, BSc

Thesis

Submitted by Nargiz Sultanova

in total fulfillment of the Requirements for the Degree of

Master of Mathematical Sciences (Research)

School of Information Technology and Mathematical Sciences

University of Ballarat

PO Box 663

University Drive, Mount Helen

Ballarat, Victoria 3353

Australia

September, 2009

A class of Increasing Positively Homogeneous functions for which global optimization problem is NP-hard

Nargiz Sultanova, BSc
University of Ballarat, 2009

Supervisor: Dr. Adil Bagirov
Associate Supervisor: Dr. Eldar Hajilarov

Abstract

It is well known that global optimization problems are, generally speaking, computationally infeasible, that is solving them would require an unreasonably large amount of time and/or space. In certain cases, for example, when objective functions and constraints are convex, it is possible to construct a feasible algorithm for solving global optimization problem successfully. Convexity, however, is not a phenomenon to be often expected in the applications. Nonconvex problems frequently arise in many industrial and scientific areas. Therefore, it is only natural to try to replace convexity with some other structure at least for some classes of nonconvex optimization problems to render the global optimization problem feasible. A theory of abstract convexity has been developed as a result of the above considerations. Monotonic analysis, a branch of abstract convex analysis, is analogous in many ways to convex analysis, and sometimes is even simpler. It turned out that many problems of nonconvex optimization encountered in applications can be described in terms of monotonic functions. The analogies with convex analysis were considered to aid in solving some classes of nonconvex optimization problems. In this thesis we will focus on one of the elements of monotonic analysis - Increasing Positively Homogeneous functions of degree one or in short IPH functions. The aim of present research is to show that finding the solution and ϵ-approximation to the solution of the global optimization problem for IPH functions restricted to a unit simplex is an NP-hard problem. These results can be further extended to positively homogeneous functions of degree η, $\eta > 0$.

A class of Increasing Positively Homogeneous functions for which global optimization problem is NP-hard

Statement of Authorship

Except where explicit reference is made in the text of the thesis, this thesis contains no material published elsewhere or extracted in whole or in part from a thesis by which I have qualified for or been awarded another degree or diploma. No other persons work has been relied upon or used without due acknowledgement in the main text and bibliography of the thesis.

Nargiz Sultanova
September 10, 2009

Acknowledgments

I would like to express my deep and sincere gratitude to my supervisors Eldar Hajilarov and Adil Bagirov for sharing with me their expertise, for their patience, encouragement and many efforts to explain things clearly and simply. Their guidance helped me immensely throughout the past two years of research and in writing the thesis.

I am greatly indebted to the University of Ballarat for giving me the opportunity to study in Australia and gaining the skills and knowledge necessary for my further development. The atmosphere and environment in the university both academic and administrative have been extremely supportive and enjoyable.

I owe my loving thanks to my family for their unfailing support, constant encouragement and care they have given me during the entire period of my study.

Nargiz Sultanova

University of Ballarat

September 2009

Contents

List of Figures

Chapter 1

Introduction

1.1 Global optimization problem

Global optimization deals with the finding of global maximum or minimum of a given function (referred to as an objective function) in a certain feasible region. Global optimization (in short GO) problems are very diverse and come from areas such as chemistry and biology, computer science, engineering, operations research, economics, etc.

Generally speaking, a global optimization problem over an unbounded domain is algorithmically unsolvable. This fact can be shown using the result obtained by Matiyasevich [18] for Hilbert's 10th problem, which asks if it is possible to construct an algorithm that checks if an arbitrary diophantine (multi variable polynomial) equation has integer solutions. As the problem was answered in the negative, it is possible [39] to construct an example of an algorithmically unsolvable unconstrained global optimization problem expressed by the diophantine equation.

Therefore, we will be considering the global optimization problem over a bounded domain. Our main concern is the feasibility of the algorithm that can be constructed to solve the problem. In terms of the complexity of the algorithm we will look at two classes of problems: P and NP, where class P includes problems that are computable in polynomial time and class NP encompasses all the problems in P and many others for which only exponential time algorithms have been discovered so far and all attempts to find polynomial time algorithms have fallen flat. On the other hand, no one has succeeded in proving that these problems cannot be solved in polynomial time. Despite that, problems in NP are considered to be intractable. The question of whether P=NP remains the most important unsolved problem in the field of computer science. In 2000 Clay Institute of Mathematics listed it among its seven Millennium Prize Problems, offering $1 million dollar reward for the solution of each [1].

Classical optimization tools in general cannot solve many problems of global optimization. The reason for this is that there can be solutions that are locally but not globally optimal. As a result, the classical techniques that are used mainly in the study of local optimization can get easily trapped in local minima or maxima. There, however, exist areas of simultaneously local and global optimization. The most well-known one is convex programming.

1.1.1 Convex optimization

The story of convex optimization began in the middle of the XX's century. Convex optimization deals with the minimization of a convex objective function over a convex set. The earliest nontrivial problems with constraints in the form of inequalities appeared in a paper by L.Kantorovich. These problems came to be known as linear programming problems. A great contribution to the theory of convex optimization was made by G. Dantzig who developed an algorithm for the solution of linear programming problem that is known as Simplex Method. Although it was shown by Klee and Minty [15] that the worst-case time bound for Simplex method is exponential, it proved to be fast on the average and was successfully implemented in many applications. The first polynomial time algorithm for linear programming problem was discovered by Nemirovsky, Yudin and Shor and is known as the method of circumscribed ellipsoids. However, it showed to be too slow to be of practical interest. Further major advance was the development of interior point methods by Karmarkar [13]. These are polynomial-time algorithms that can be efficiently implemented to solve real-world problems in linear programming. Interior point methods were subsequently generalized to various convex optimization problems by Nesterov and Nemirovsky [19] and have gained a lot of popularity and sophistication since then.

1.1.2 Nonconvex optimization

Convex functions enjoy an important property: a local minimum (or maximum) of a convex function is also a global one. Therefore, any local minimization algorithm yields the solution to a global minimization problem as well. As mentioned above, in case of convex functions there exist powerful methods that can be used to find the global minimum efficiently. Unfortunately, the same cannot be said with regard to nonconvex problems. For some nonconvex objective functions there exist efficient algorithms. However, generally speaking, nonconvex problems seem to be infeasible. An example of a nonconvex problem with feasible algorithm is a Fractional Linear Programming problem (FLP). These problems arise in various fields, including game theory, network flows, etc. FLP looks as follows:

$$\min \ (p^T x + \gamma)/(q^T x + \delta)$$

$$\textit{subject to } Ax \geq b$$

where p, q are n-vectors, γ, δ are scalars, A is an $m \times n$ matrix, b is an m-vector and superscript T denotes the transpose. Here an assumption is made that $(q^T x + \delta) > 0$ over the domain. A possible method for solving the above problem is Dinkelbach's algorithm. It makes use of Jagannathan's theorem [11] which states that $y \in X$ is an optimal solution for a problem

$$\min\{\theta(x) = \frac{F(x)}{G(x)} : x \in X \subset \mathbb{R}^n\},$$

where $F(x)$ and $G(x)$ are continuous realvalued functions, θ is a continuous function on X and $G(x) > 0$, if and only if y is an optimal solution for $min\{F(x) - \theta(y)G(x) : x \in X\}$. Thus, if the function is defined as

$$f(\theta) = \min\{(p^T x + \gamma) - \theta(q^T x + \delta)\} \tag{1.1}$$

$$subject\ to\ Ax \geq b$$

then the optimal solution of the FLP corresponds exactly to a root of 1.1. Correspondingly, one may find the root of 1.1 using any method of convex programming.

Another example of a tractable nonconvex problem is a quadratic programming problem with ellipsoidal constraints. Although general nonconvex quadratic programming problems are computationally intractable as proved by Sahni [27], quadratic programming problem with an ellipsoidal (sphere in this case) constraint stated as

$$\min\ \frac{1}{2}x^T Q x + c^T x$$

$$subject\ to\ \|x\|_2 \leq r$$

where $x \in \mathbb{R}^n$, Q is symmetric $n \times n$ matrix, c is an n-vector and $\|x\|_2 = (\sum_{i=1}^{n} x_i^2)^{\frac{1}{2}}$ is polynomially solvable. Ye [38] proved that the above problem can be solved in $O(log(log(1/\epsilon)))$ iterations, where ϵ is the error tolerance.

The above two examples of tractable nonconvex problems, are, as mentioned earlier, exception rather than the rule in the field of global optimization.

1.2 Abstract convexity

An important property of a convex function is that it can be represented as a pointwise supremum of a collection of affine minorizing functions. In the 1970s it was realized that many results of the convexity theory do not depend on the linearity of the minorants. These ideas stimulated the development of the theory of Abstract Convexity.

The first book on abstract convexity was published by S.Kutateladze and A.Rubinov in 1976 [17].

A significant role in the study of convex optimization problems belongs to a subdifferential of a function. It allows to construct a so called global affine support to a given function f, which is a function that coincides with f at a point x and is less than f at all other points. Therefore, the global minimum of the underestimate will provide the lower bound to the global minimum of the objective function.

Generalization of the notion of subdifferential has been used in the development of theory of abstract convexity while the main motivation for further advances in the theory was due to its applications in optimization. It was interesting to examine main notions of convexity in a non-convex setting to see whether it might aid in solving nonconvex global optimization problems.

As it turned out, many important results and concepts of convex analysis could be extended to arbitrary sets and functions.

In convex analysis functions can be represented as upper envelopes of a collection of affine functions. Generalizing this idea, abstract convex functions can be represented as upper envelopes of so called elementary functions, which are not necessarily linear.

1.2.1 Monotonic analysis

Monotonic analysis is a branch of abstract convex analysis that uses a special choice of elementary functions. The term monotonic analysis for this theory was introduced in [26].

The simplest and most elegant theory of monotonic analysis can be obtained for functions defined on the cone $\mathbb{R}^n_{++}$ and the subsets of this cone.

Monotonicity is exhibited by many functions arising in various areas of mathematics. Thus, many problems of nonconvex optimization encountered in applications can be described in terms of monotonic functions. These problems include multiplicative programming [34], Lipschitz optimization [22], fractional linear programming [32], etc.

There exist a number of numerical algorithms for finding an approximate solution of some specific classes of non-convex global optimization problems that are based on monotonicity ideas. For instance, not so long ago the cutting angle method has been proposed and studied in [2, 4, 3]. It is an extension of a cutting plane method in convex optimization. Other methods of monotonic optimization are studied in [25, 35, 23].

These methods allow to find only approximate global minimizers or maximizers of a given function, however the dimensionality of the problems solved can be fairly high due to the fact that many large-scale nonconvex problems can be reduced to monotonic optimization problems of smaller dimension.

Nevertheless, presently no methods of global optimization are able to solve general nonconvex problems efficiently - only specific limited classes of problems can be tackled.

1.3 Structure of thesis

In this thesis we will be concentrating on one element of monotonic analysis - Increasing Positively Homogeneous functions of degree one or, in short, IPH functions. The aim of present research is to show that finding the solution and ϵ-approximation to the solution of global optimization problem for IPH functions restricted to a unit simplex is an NP-hard problem. These results can be further extended to positively homogeneous functions of degree η, $\eta > 0$.

Chapter 2 of the thesis describes the notions of computational complexity and gives insight into P, NP, NP-complete and NP-hard complexity classes. It shows examples of problems that belong to some of the above classes and gives an overview of the relationship between the latter. Chapter 3 is devoted to the theory of abstract convexity and to monotonic analysis. It contains the main definitions and also provides some basic results that have been previously obtained in the field. Chapter 4 contains the results establishing the infeasibility of the global optimization problem for IPH functions on a unit simplex. Chapter 5 is the conclusion.

Chapter 2

Complexity, Class NP and NP-completeness

For many problems we are studying it is important not only to find the solution in principle but to construct an efficient computational method for reaching that solution. Some problems are easy to solve while others are not. For example, some difficult problems require an exponential amount of time in terms of the size of the problem to solve.
Computational complexity theory studies the amount of computational effort that is needed in order to perform certain kinds of computation.
Problems can be classified by *complexity class* according to the amount of time it takes for an algorithm to solve them as a function of the amount of input in the problem. We will be concentrating on two of the most important complexity classes - classes of NP-complete and NP-hard problems.

The theory of NP-completeness restricts attention to decision problems only. A *decision problem* is the one that asks only for a "yes" or "no" answer. An *optimization problem* is the problem of finding the *best* of all feasible solutions. However, decision problems can be derived from optimization problems by imposing a bound on the value to be optimized and vice versa. Therefore, we can extend the implication of the theory of NP-completeness to optimization problems. If we find a fast way to solve the decision problem, then there is a fast way to solve the corresponding optimization problem as well.

Decision problems have a formal counterpart that is called a *language*, which is more suitable in the study of theory of computation. The language framework allows to express the relationship between decision problems and algorithms that solve them. The language is defined as follows [9]:
For any finite set of symbols $\sum$ denote by $\sum^*$ the set of all finite strings of symbols from $\sum$. If L is a subset of $\sum^*$, we say that L is a *language* over the alphabet $\sum$.

2.1 Turing Machine

A Turing machine is an idealized computing device described by Alan Turing in 1936. It is powerful enough to be able to perform any calculation that any other computer can do and at the same time is simple enough to be useful in proving theorems in complexity theory. There are number of types of Turing machines, however it turns out that the computational capability of all these machines is equal.

A deterministic one-tape Turing machine (a basic Turing machine) consists of [37]
1. a doubly infinite *tape* that is marked off into squares. Each square can contain a single character from the character *set* $\sum$ that the machine recognizes. For simplicity, we can assume that the character set contains just three symbols: '0', '1', and '$\sqcup$'(blank)
2. a tape *head* that is capable of either reading a single character from a square on the tape or writing a single character on a square, or moving its position relative to the tape by an increment of one square in either direction.
3. a finite list of *states* Q such that at every instant the machine is in exactly one of those states. The possible states of the machine are, first of all, the regular states $q_1, q_2, ..., q_s$ and, second, three special states

q_0 : the initial state

q_Y : the final state in a problem to which the answer is "Yes"

q_N : the final state in a problem to which the answer is "No"

4. a *program* that directs the machine through the steps of a particular task.

Suppose that at a certain instant the machine is in a state q (other than q_Y or q_N) and that the symbol that has just been read from the tape is "*symbol*". Then from the pair $(q, symbol)$ the program will decide

a) to what state q' the machine shall go next and

b) what single character the machine will now write on the tape in the square over which the head is now positioned and

c) whether the tape head will next move one square to the right or one square to the left.

One step of the program, therefore, goes from

$$(state, symbol) \ to \ (newstate, newsymbol, increment)$$

The above program is presented in the form of a *transition function* $\delta : Q \times \sum \rightarrow Q \times \sum \times \{L, R, N\}$, where L indicates left shift, R right shift, N no shift.

If and when the state reaches q_Y or q_N the computation is over and the machine halts.

We say that a Turing machine M with input alphabet $\sum$ *accepts* string $x \in \sum^*$ if and only if it halts in state q_Y when applied to x.

The collection of strings that the Turing machine accepts is the language of the Turing machine and is denoted by L_M.

We say that a language is *recognized* by the Turing machine if and only if it accepts all the words of the language.

A Turing machine that halts (either accepts or rejects) on all inputs is called a *decider*. A decider that recognized some language is said to *decide* that language.

The following correspondence between the decision problems and languages can be derived: since every decision problem can have only two answers ”yes” and ”no”, we can think of a decision problem as asking if a given word (the input string) does or does not belong to a certain language. Therefore, a Turing machine which decides that language would solve the corresponding decision problem.

2.2 The Class P

Class P contains decision problems which can be solved by deterministic Turing machine in polynomial time. This is a class of problems that are considered to be efficiently solvable. Some problems that are included in this class are:

Reachability Given a directed graph $G = (V, E)$ and two nodes $v_i, v_j \in V$, is there a path from v_i to v_j?

Euler cycle Given a graph G, is there a closed path in G that visits each edge exactly once?

Primes Given a number n, is it possible to test whether it is prime or composite?

Unary partition Given a set of n natural numbers $\{a_1, ..., a_n\}$ represented in unary, is there a subset $P \subseteq \{1, ..., n\}$ such that $\sum_{i \in P} a_i = \sum_{i \notin P} a_i$?

2.3 The Class NP

Another type of a Turing machine is a nondeterministic Turing machine. A nondeterministic Turing machine differs from a deterministic one in that at any point during the computation the machine may proceed according to several possibilities. This means that the given state and symbol under the tape head no longer uniquely specify the 3-tuple $(newstate, newsymbol, increment)$, that is the transition function is multi-valued. The nondeterministic Turing machine is said to accept its input if any sequence of choices of moves leads to an accepting state.

Let N be a nondeterministic Turing machine that is a decider. The *running time* of N is the function $f : N \to N$, where $f(n)$ is the maximum number of steps that N uses on any branch of its computation on any input of length n.[30]

The class NP is defined as a class of decision problems which can be solved by nondeterministic Turing machine in polynomial time. Hence, the term NP comes from *nondeterministic polynomial time*.

An alternative definition [37] of class NP is given below:

A decision problem D is said to belong to class NP if there is an algorithm A such that

1. Associated with each word of the language Q (i.e., with each instance of decision problem I for which the answer is "yes") there is a certificate (an information needed to verify a positive answer) $C(I)$ such that when the pair (I, C(I)) are input to algorithm A it recognizes that I belongs to the language Q.

2. If I is some word that does not belong to the language Q then there is no choice of certificate $C(I)$ that will cause A to accept I

3. Algorithm A operates in polynomial time

Therefore, NP is the class of decision problems for which checking the correctness of a "yes" answer is easy if the answer is "yes" indeed. In other words, the answers should be verifiable in polynomial time by a deterministic Turing machine.

2.4 The Relationship between P and NP

As deterministic Turing machine can be regarded as a simplified version of nondeterministic Turing machine, it is clear that P$\subseteq$ NP.

The question of whether P is a proper subset of NP or not has been one of the most perplexing research problems in computer science. Despite the enormous efforts of many scientists over the decades, no polynomial time algorithms for numerous problems in NP have been found, nor has it been proved that no such algorithms exist.

Considering the given state of knowledge, it seems reasonable to operate with the assumption that P≠NP, that is to assume that there exists problems that are hard to solve but whose solutions can be quickly verified.

2.5 NP-completeness

A decision problem is said to be NP-complete if it belongs to NP and any problem in NP is reducible to it in polynomial time.
The implication of the definition above is that if a polynomial algorithm is found for any NP-complete problem then a polynomial algorithm can be constructed for any problem in NP. Vice versa, a provably slow algorithm for a problem in NP means provably slow for any NP-complete problem.

The first NP-complete problem was discovered by S.Cook in 1971. It is a decision problem from Boolean logic and is called the satisfiability problem or SAT. SAT was proved to be NP-complete with the aid of the theory of Turing machines.

Let $X = (x_1, ..., x_n)$ be a set of Boolean variables. If x is a Boolean variable, then *literal* is either the variable x itself, or its negation x. A *clause* is a set of literals, for example $\{\overline{x_1}, x_3, x_4\}$. A value "true" or "false" is assigned to each variable from X. The literal inherits the truth value of a variable, that is if the variable x is true, then so is the literal x, on the other hand literal $\overline{x}$ is true if and only if the variable x is false. A clause is said to be satisfied if and only if at least one of its literals is "true". We say that a collection of clauses $C_1, ..., C_m$ on the variables $x_1, ..., x_n$ is satisfiable if and only if there exists an assignment of truth values to the variables such that all the clauses are "true". The SAT problem is specified as follows:

Given a set of variables X and a collection of clauses $C_1, ..., C_m$ does there exist a set of truth values for the variables such that every clause is satisfied.

To demonstrate NP-completeness of a problem it is sufficient to show that some other NP-complete problem reduces to it in polynomial time. Here follow several examples of classic NP-complete problems that have been widely used in proving other problems NP-complete.

Hamiltonian cycle Given a graph is there a cycle that visits each vertex exactly once?

Traveling salesman Given a set of n cities and a distance D is there a route that visits each city, returns to the starting city and has total length not more than D?

Subset sum Given a collection of integers $s_1, ..., s_n$ is there a subset of the integers that sums up to exactly s?

Clique Given a graph $G = (V, E)$ and a positive integer K is there a set of K vertices in G such that each of the vertices is joined by an edge in E to all the others?

Vertex cover Given a graph $G = (V, E)$ and a positive integer K, is there a vertex cover S (a subset of vertices V such that each edge in E has at least one endpoint in S) of size at most K?

Knapsack problem Given a set N of n items j with profit p_j, weight w_j and the capacity W, select the subset of N such that the total profit of selected items is maximized without exceeding the capacity W.

Although no proof of it has been found, it is commonly believed that NP-complete problems are intractable. Therefore, once the problem has been shown to be NP-complete, it is more practical to construct an approximation algorithm rather than solving the problem exactly, or to use a heuristic.

2.6 NP-hardness

A problem H is said to be NP-hard if and only if there is an NP-complete problem that is polynomial time reducible to H. Informally, NP-hard is a class of problems that are at least as hard as the NP-complete problems. NP-hard problems can be both of decision and optimization types. An optimization version of an NP-complete decision problem is NP-hard. From the definition of NP-hardness follows that if P≠NP then NP-hard problems will have no polynomial time solution, however from P=NP does not follow polynomial solvability of the latter.

The relationship between P, NP, NP-complete and NP-hard classes of problems can be given with the aid of the following diagrams. The first one describes the case when P=NP, and the second one the case for P≠NP.

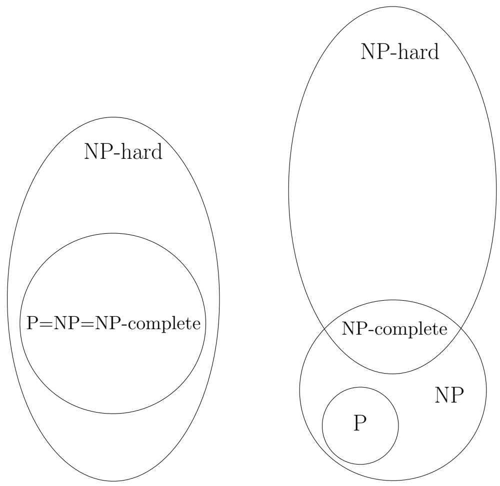

Figure 2.1: On the left: P=NP; On the right: P≠ NP

Chapter 3

Abstract convexity and monotonic analysis

3.1 Abstract convex functions and sets

Many important concepts and results of convex analysis can be naturally extended to the abstract convexity framework.

In this section we introduce the main notions of the theory [29, 20, 24, 21].

Before we proceed, we need to introduce some notation that will be used in the sequel.

$\mathbb{R} = (-\infty, +\infty)$

$\bar{\mathbb{R}} = [-\infty, +\infty]$

$\mathbb{R}^n = \{(x_1, ..., x_n) : x_i \in \mathbb{R}, i = 1, ..., n\}$

$\mathbb{R}^n_+ = \{(x_1, ..., x_n) \in \mathbb{R}^n : x_i \geq 0, i = 1, ..., n\}$

$\mathbb{R}^n_{++} = \{(x_1, ..., x_n) \in \mathbb{R}^n : x_i > 0, i = 1, ..., n\}$

Definition 3.1.1. Let H be a nonempty set of functions $h : X \longrightarrow \bar{\mathbb{R}}$. A *function* $f : X \rightarrow \bar{\mathbb{R}}$ is called *abstract convex* with respect to H (or H-convex) if there exists a set $U \subset H$ such that f is the upper envelope of this set:

$$f(x) = sup\{h(x) : h \in U\} \ for \ all \ x \in X$$

The set H will be referred to as a set of *elementary functions*.

Definition 3.1.2. Let $f : X \to \bar{\mathbb{R}}$. The set

$$supp(f, H) = \{h \in H, h \leq f\}$$

of all H-minorants of f is called the *support set* of the function f with respect to the set of elementary functions H.

Definition 3.1.3. Let $f : X \to \bar{\mathbb{R}}$. The function $co_H f$ defined by

$$co_H f(x) = sup\{h(x) : h \in supp(f, H)\}, \ (x \in X)$$

is called the *H-convex hull* of a function f.

Definition 3.1.4. A *set* $U \subset H$ is called *abstract convex* with respect to X (or (H, X)-convex) if there exists a function $f : X \to \bar{\mathbb{R}}$ such that $U = supp(f, H)$.

Definition 3.1.5. The intersection of all (H, X)-convex sets containing a set $U \subset H$ is called the *abstract convex hull* or (H, X)-convex hull of the set U.

The following lemma is a reformulation of the definition of *abstract convex sets* in terms of the separation property.

Lemma 3.1.6 (Separation property)**.** *Let H be a set of elementary functions defined on a set X. Let U be a proper subset of H, that is $U \neq \emptyset$, $U \neq H$. Then the set U is (H, X)-convex if and only if for each h that does not belong to U there exists $x \in X$ such that*

$$h(x) > sup_{h' \in U} h'(x)$$

3.1.1 Subgradient

We have mentioned earlier that subdifferential plays an important role in convex optimization. Here follows the definition of a subdifferential for abstract convex functions.

Definition 3.1.7. Let L be a set of elementary functions $l : X \to \mathbb{R}$. A function $l \in L$ is called an *abstract subgradient* (or *L-subgradient*) of a function $f : X \to \mathbb{R}_{+\infty}$ at a point y if $f(x) - f(y) \geq l(x) - l(y)$ for all $x \in X$. The set $\partial_L f(y) = \{l \in L : f(x) - f(y) \geq l(x) - l(y) \ for \ all \ x \in X\}$ of all abstract subgradients of f at y is referred to as the *abstract subdifferential* (or *L-subdifferential*) of the function f at a point y.

We will regard set L as a set of *abstract linear* functions. Let H_L be the closure of L under vertical shifts, that is $H_L = \{h' : h'(x) = h(x) - c, h \in H, c \in \mathbb{R}\}$. The set H_L will be referred to as the set of *abstract affine* functions with respect to L or *L-affine functions.*

3.2 IPH functions and their properties

Monotonic analysis is abstract convex analysis with respect to a special choice of elementary functions.

Many of the concepts and results presented in this section can be found in [20, 21, 6]. We, however, include them here for the purpose of a clearer view and a better understanding of the subsequent results.

Let us first introduce some basic notation. For any two vectors $x = (x_1, ..., x_n) \in \mathbb{R}^n$, $y = (y_1, ..., y_n) \in \mathbb{R}^n$ $x \geq y$ means that for any $i = 1, ..., n$ $x_i \geq y_i$. Two vectors x, y for which $x_i \geq y_i$, $i = 1, ..., n$ does not hold shall be referred to as incomparable.

A function $f : \mathbb{R}^n_+ \to \mathbb{R}$ is said to be increasing if $x \geq y$, where $x, y \in \mathbb{R}^n_+$ implies $f(x) \geq f(y)$.

Many functions encountered in mathematical economics are increasing in that sense. One example of it is a well known Cobb-Douglas production function $f(x) = \prod_i x_i^{\alpha^i}$, $\alpha_i \geq 0$

A subset C of a vector space is called a *conic set* or a *cone* if $\lambda x \in C$ for each $x \in C$ and $\lambda > 0$

Here we will study monotonic, that is increasing and decreasing functions defined on a conic set with a certain property along rays.

Let C be a conic set. We say that a property (P) of a function $f : C \to \mathbb{R}$ holds along rays if the restriction of f to the ray $R_x = \{\alpha x : \alpha \geq 0\}$ starting from zero and passing through x enjoys property (P) for each $x \in C$. That is, (P) holds along rays if the function f_x defined by $f_x(\alpha) = f(\alpha x),\ \alpha \geq 0$ possesses the property (P).

Several examples of functions with properties along rays are:

1) Continuous along rays functions f: the function f_x is continuous for all $x \in C$.

2) Convex along rays functions f: f_x is convex for all $x \in C$.

3) Positively homogeneous functions of degree γ: a function f is said to be positively homogeneous of degree γ if $f(\lambda x) = \lambda^\gamma f(x)$ for any $\lambda \geq 0$.

We shall consider increasing positively homogeneous functions of degree one, the common abbreviation for which is IPH. Hence, the formal definition of IPH functions is stated as follows:

Definition 3.2.1. (IPH functions) Let Q be either $\mathbb{R}^n_+$ or $\mathbb{R}^n_{++}$. A function $f : Q \to \mathbb{R}_{+\infty}$ is an IPH function if the following conditions are satisfied:

1. $x \geq y$ implies $f(x) \geq f(y)$ (monotonicity condition);
2. $f(\lambda x) = \lambda f(x)$ for $\lambda > 0$ *and* $x \in Q$ (positive homogeniety condition).

Examples of IPH functions:

1. $f(x) = \sum_{i \in I} a_i x_i,\ a_i \geq 0$;
2. $f(x) = \| x \|_p,\ p > 0$;
3. $f(x) = \sqrt{[Ax, x]},$ where A is a matrix with nonnegative terms;
4. $f(x) = \prod_{j \in J} x_j^{t_j},\ J \subset I = \{1, ..., n\},\ t_j > 0, \sum_{j \in J} t_j = 1$.

It is easy to see that if f_1 and f_2 are IPH functions, then for any $\lambda_1 \geq 0, \lambda_2 \geq 0$ the function $\lambda_1 f_1 + \lambda_2 f_2$ is IPH as well.

We proceed with some properties of IPH functions defined on $\mathbb{R}^n_{++}$ [20]. Let f be an IPH function, then

1. $f(x) \geq 0$ for all $x \in \mathbb{R}^n_{++}$. Indeed, consider a vector $x \in \mathbb{R}^n_{++}$. Since $(1/2)x \leq x$, it follows that $(1/2)f(x) = f((1/2)x) \leq f(x)$. Hence $f(x) \geq 0$.

2. If f is IPH and there exists a point $\bar{x} \in \mathbb{R}^n_{++}$ such that $f(\bar{x}) = +\infty$, then $f(x) = +\infty$ for all $x \in \mathbb{R}^n_{++}$. Indeed, if $x \in \mathbb{R}^n_{++}$, then there exists $\lambda > 0$ such that $x \geq \lambda\bar{x}$. Therefore $f(x) \geq f(\lambda\bar{x}) = \lambda f(\bar{x}) = +\infty$.

3. If there exists a point $\bar{x} \in \mathbb{R}^n_{++}$ such that $f(\bar{x}) = 0$, then $f(x) = 0$ for all $x \in \mathbb{R}^n_{++}$. In fact, for each $x \in \mathbb{R}^n_{++}$ there exist $\lambda > 0$ such that $x \leq \lambda\bar{x}$. Hence $0 \leq f(x) \leq \lambda f(\bar{x}) = 0$.

4. Each IPH function f is continuous on $\mathbb{R}^n_{++}$. Assume that $f : \mathbb{R}^n_{++} \rightarrow \mathbb{R}_{++}$. Let $x \in \mathbb{R}^n_{++}$ and $x_n \rightarrow x$. For $\varepsilon > 0$ and large n we have $(1-\varepsilon)x \leq x_n \leq (1+\varepsilon)x$. Then $(1-\varepsilon)f(x) \leq f(x_n) \leq (1+\varepsilon)f(x)$, that is $f(x_n) \rightarrow f(x)$.

Consider the set L of all functions $l(x)$ defined on $\mathbb{R}^n_{++}$ as per the following definition.

Definition 3.2.2. (min-function) A function $l(x) = \left(\frac{x}{l}\right) = \min_{i \in I} \frac{x_i}{l_i}$, where $x, l \in \mathbb{R}^n_{++}$, $I = \{1, ..., n\}$ is called a min-function.

If vectors x and l are collinear, the min-function produces a number by which vector l needs to be multiplied in order to be shrunk or stretched to the length of vector x. Figure 3.1 is a diagram for the case of 2-dimensional space.

If they are not collinear then $l(x)$ is a minimum number which we need to multiply the vector l by so that it reaches the boundary of the box generated by the vector x (Figure 3.2).

It is easy to see that min-functions are IPH.

For $x \leq l$, $\left(\frac{x}{l}\right) \leq 1$. Indeed, $x \leq l$ implies $x_1 \leq l_1, ..., x_n \leq l_n$, that is $\frac{x_1}{l_1} \leq 1, ..., \frac{x_n}{l_n} \leq 1$.
By definition $\left(\frac{x}{l}\right) = \min_{i \in I} \frac{x_i}{l_i} = \min\left(\frac{x_1}{l_1}, ..., \frac{x_n}{l_n}\right)$. Therefore, $\left(\frac{x}{l}\right) \leq 1$.
Similarly, for $x \geq l$, $\left(\frac{x}{l}\right) \geq 1$.

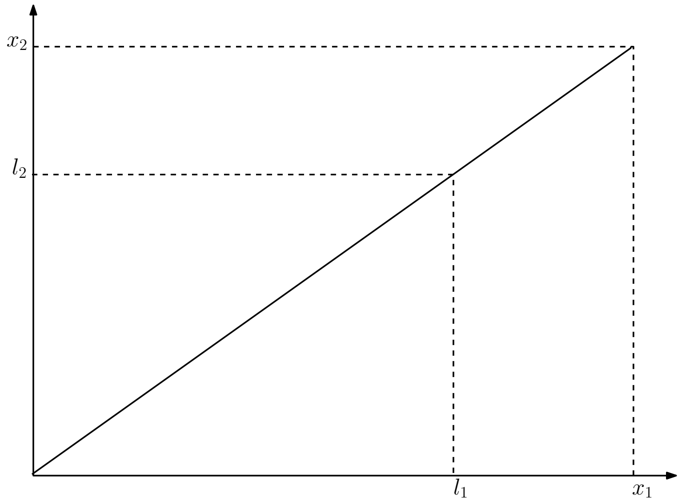

Figure 3.1: Collinear vectors

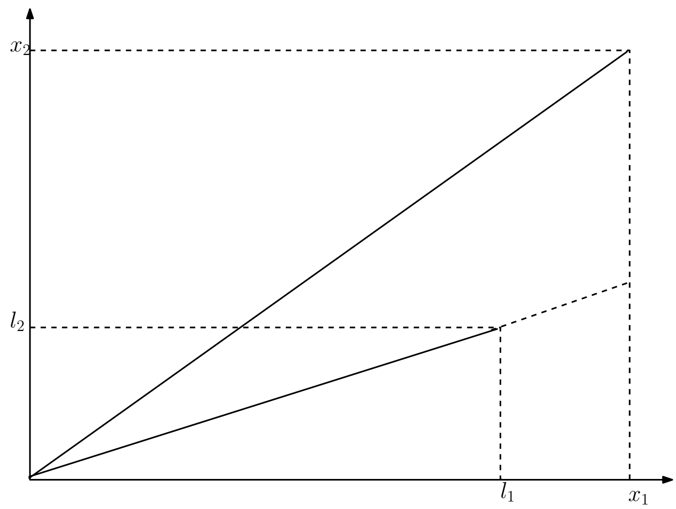

Figure 3.2: Noncollinear vectors

Proposition 3.2.1. *A function $f : \mathbb{R}^n_{++} \to \mathbb{R}_{+\infty}$ is abstract convex with respect to L if and only if it is IPH.*

Proof. Since functions $\left(\frac{x}{l}\right)$ are increasing and positively homogeneous, any L-convex function is IPH. Conversely, suppose that p is IPH and $y \in \mathbb{R}^n_{++}$. Let $l_y = \frac{y}{p(y)}$. We have $l_y(y) = \left(\frac{y}{l_y}\right) = p(y)$. Let $x \in \mathbb{R}^n_{++}$ and $\lambda = \min_i \frac{x_i}{y_i}$. Then $x \geq \lambda y$. Therefore,

$$p(x) \geq p(\lambda y) = \lambda p(y) = \min_i \frac{x_i p(y)}{y_i} = \min \frac{x_i}{l_y} = \left(\frac{x}{l_y}\right)$$

Let $U = \{l_x : x \in \mathbb{R}^n_{++}\}$. Then for each $y \in \mathbb{R}^n_{++}$

$$p(y) = \left(\frac{y}{l_y}\right) = \max_{l \in U} \left(\frac{y}{l}\right)$$

□

Lemma 3.2.3. *For any $x, y \in \mathbb{R}^n_{++}$ the following inequality is true*

$$y \left(\frac{x}{y}\right) \leq x \tag{3.1}$$

Proof. Proof of lemma. Indeed,

$$y \left(\frac{x}{y}\right) = (y_1, \ldots, y_n) \min_i \frac{x_i}{y_i} = \left(x_1 \frac{y_1}{x_1} \min_i \frac{x_i}{y_i}, \ldots, x_n \frac{y_n}{x_n} \min_i \frac{x_i}{y_i}\right) \leq$$

$$\leq \left(x_1 \frac{y_1}{x_1}\frac{x_1}{y_1}, \ldots, x_n \frac{y_n}{x_n}\frac{x_n}{y_n}\right) = (x_1, \ldots, x_n) = x$$

The lemma is proved. □

Let us now introduce the following criterion for a function to be IPH.

Theorem 3.2.4. *A function p is IPH if and only if for any $x, y \in \mathbb{R}^n_{++}$ it satisfies the following inequality*

$$\left(\frac{x}{y}\right) p(y) \leq p(x) \tag{3.2}$$

Proof. Let p be an IPH function. Since p is IPH, applying it to 3.1 as follows

$$p\left(y\left(\frac{x}{y}\right)\right) \leq p(x)$$

we obtain

$$\left(\frac{x}{y}\right)p(y) \leq p(x)$$

Let us now show that the reverse is true as well. First we need to show that p is positively homogeneous. Assume that inequality 3.2 holds. Then the following inequality is true as well

$$\left(\frac{y}{x}\right)p(x) \leq p(y) \tag{3.3}$$

From 3.2 we get

$$p(y) \leq \frac{p(x)}{\left(\frac{x}{y}\right)} \tag{3.4}$$

Hence, from 3.3 and 3.4

$$p(x)\left(\frac{y}{x}\right) \leq p(y) \leq \frac{p(x)}{\left(\frac{x}{y}\right)}$$

Taking $y = \alpha x$, where $\alpha > 0$, we obtain

$$p(x)\left(\frac{\alpha x}{x}\right) \leq p(\alpha x) \leq \frac{p(x)}{\left(\frac{x}{\alpha x}\right)}$$

$$\alpha p(x) \leq p(\alpha x) \leq \alpha p(x)$$

i.e.

$$p(\alpha x) = \alpha p(x)$$

which means that function p is positively homogeneous.

It is now sufficient to show that function p is monotonic. Let $y \leq x$. Then $\left(\frac{x}{y}\right) \geq 1$ and from $\left(\frac{x}{y}\right) p(y) \leq p(x)$ it follows that $p(y) \leq p(x)$. Thus function p is monotonic. Therefore, p is an IPH function. □

Let us now examine two very important notions of abstract convexity - the support set and the subdifferential. One of the significant results of abstract convex analysis is that we can obtain explicit representation of both of the above.

Proposition 3.2.2. *Let p be an IPH function defined on $\mathbb{R}^n_{++}$ and L be the set of min-functions l. Then*

$$supp(p, L) = (l : p(l) \geq 1)$$

Proof. By definition the support set for p is the following

$$supp(p, L) = (l \in L : p(x) \geq l(x)),$$

which in turn is equivalent to

$$supp(p, L) = \left(l \in L : p\left(\frac{x}{l(x)}\right) \geq 1\right) \tag{3.5}$$

For $x = l$ from 3.5 we obtain

$$supp(p, L) = (l : p(l) \geq 1) \tag{3.6}$$

We now need to show that if $p(l) \geq 1$ then 3.5 holds true as well. Let us write out the expression for $p\left(\frac{x}{l(x)}\right)$

$$p\left(\frac{x}{l(x)}\right) = p\left(\frac{x}{\min_i \frac{x_i}{l_i}}\right) = p\left(\frac{x_1}{\min_i \frac{x_i}{l_i}}, ..., \frac{x_n}{\min_i \frac{x_i}{l_i}}\right) = p\left(l_1\frac{\frac{x_1}{l_1}}{\min_i \frac{x_i}{l_i}}, ..., l_n\frac{\frac{x_n}{l_n}}{\min_i \frac{x_i}{l_i}}\right)$$

As for all $i, j = 1, ..., n$ $\dfrac{\frac{x_j}{l_j}}{\min_i \frac{x_i}{l_i}} \geq 1$,

$$p\left(l_1 \frac{\frac{x_1}{l_1}}{\min_i \frac{x_i}{l_i}}, ..., l_n \frac{\frac{x_n}{l_n}}{\min_i \frac{x_i}{l_i}}\right) \geq p(l_1, ..., l_n) = p(l)$$

Thus, for any x

$$p\left(\frac{x}{l(x)}\right) \geq p(l) \geq 1$$

. □

Proposition 3.2.3. *Let $p : \mathbb{R}^n_{++} \to \mathbb{R}$ be an IPH function, L be the set of min-functions l and $x_o, y \in \mathbb{R}^n_{++}$. Then*

$$\partial_L p(x_o) = \{l \in R^n_{++} : p(l) = 1, p(y) \geq \left(\frac{y}{l}\right), p(x_o) = \left(\frac{x_o}{l}\right) \ for\ all\ y \in \mathbb{R}^n_{++}\} \tag{3.7}$$

Proof. We know that the general definition of a subdifferential for an IPH function p is the following:

$$\partial_L p(x_o) = \{l : p(y) - p(x_o) \geq l(y) - l(x_o) \ for\ all\ y \in \mathbb{R}^n_{++}\} \tag{3.8}$$

Here $l(y)$ is a min-function $l(y) = \left(\frac{y}{l}\right) = \min_i \dfrac{y_i}{l_i}$.

Consider inequality

$$p(y) - p(x_o) \geq \left(\frac{y}{l}\right) - \left(\frac{x_o}{l}\right)$$

Dividing both sides by $\left(\frac{y}{l}\right)$ as y tends to infinity we obtain

$$\lim_{y\to\infty}\left[\frac{p(y)}{(\frac{y}{l})} - \frac{p(x_o)}{(\frac{y}{l})}\right] \geq \lim_{y\to\infty}\left[1 - \frac{(\frac{x_o}{l})}{(\frac{y}{l})}\right] \tag{3.9}$$

As p is an IPH function

$$\lim_{y\to\infty}\frac{p(y)}{(\frac{y}{l})} = \frac{p(y)}{(\frac{y}{l})}$$

Thus we get $\frac{p(y)}{\left(\frac{y}{l}\right)} \geq 1$ and

$$p(y) \geq \left(\frac{y}{l}\right) \tag{3.10}$$

By making y tend to 0 in 3.9 we get

$$0 - p(x_o) \geq 0 - \left(\frac{x_o}{l}\right)$$

From here

$$\left(\frac{x_o}{l}\right) \geq p(x_o) \tag{3.11}$$

From 3.10 for $y = x_o$ we get $p(x_o) \geq \left(\frac{x_o}{l}\right)$. Taking into account 3.11 it follows that

$$p(x_o) = \left(\frac{x_o}{l}\right) \tag{3.12}$$

From 3.10 for $y = l$ we have $p(l) \geq 1$. From theorem 3.2.4 $p(l)\left(\frac{x_o}{l}\right) \leq p(x_o)$. Taking into account 3.12 we obtain $p(l) \leq 1$. As a result we have

$$p(l) = 1 \tag{3.13}$$

If we add 3.10 to 3.12 we will get 3.8. Thus, the subdifferential for an IPH function p will look as follows

$$\partial_L p(x_o) = \{l \in \mathbb{R}^n_{++} : p(l) = 1, p(y) \geq \left(\frac{y}{l}\right), p(x_o) = \left(\frac{x_o}{l}\right)\}$$

□

Chapter 4

Global optimization problem for IPH functions

In this chapter we will look into the global optimization problem for the IPH functions. The results in this section of the thesis are new. We first introduce some auxiliary results that will aid in the investigation of the above problem. For the following theorem we will introduce an auxiliary definition of a so called set of incomparable elements.

Definition 4.0.5. A set M is called a set of incomparable elements if any two elements $x, y \in M$ are incomparable, that is neither $x \leq y$ nor $x \geq y$ holds.

For example, a unit simplex $S = \{(x_1, ...x_n) \in \mathbb{R}^n_{++}, \sum_{i=1}^n x_i = 1\}$ is a set of incomparable elements.

Note that for any two points x, y on a set of incomparable elements $\left(\frac{x}{y}\right) \leq 1$.

Definition 4.0.6. Let M be a set of incomparable elements. We say that the extension of a function $p : M \to \mathbb{R}$ onto a cone C generated by M is positively homogeneous if for any $x \in M$ and for any $z \in C$ such that $z = \alpha x$, where $\alpha > 0$

$$p(\alpha x) = \alpha p(x)$$

Theorem 4.0.7. *Let us assume that we are given a function p defined on a set of incomparable elements $M \subset \mathbb{R}^n_{++}$. Then positive homogeneous extension of this function onto a conic set generated by M is an IPH function if and only if for any $x, y \in M$ it satisfies inequality*

$$\left(\frac{x}{y}\right) p(y) \leq p(x) \tag{4.1}$$

Proof. Let $x, y \in M$ and $a, b \in \mathbb{R}^n_{++}$ be arbitrary points such that $a = \alpha x$ and $b = \beta y$, $\alpha, \beta > 0$. Assume that p satisfies inequality 4.1. Then

$$\left(\frac{x}{y}\right) p(y) \leq p(x),$$

$$\frac{\alpha}{\beta}\left(\frac{x}{y}\right) p(y)\beta \leq p(x)\alpha,$$

$$\left(\frac{\alpha x}{\beta y}\right) p(\beta y) \leq p(\alpha x),$$

$$\left(\frac{a}{b}\right) p(b) \leq p(a),$$

Similarly, reversing the above scheme we obtain that if the positively homogeneous extension of function p is IPH, then p satisfies the inequality 4.1. Thus, it follows from Theorem 3.2.4 that the extension of the function is IPH. □

4.1 Lipschitz functions embeddable in IPH functions

Let us now introduce the notation we will be using throughout the rest of the chapter.

1) $\Delta^{n-1} = \{(x_1, ...x_n) \in \mathbb{R}^n_{++}, \sum_{i=1}^n x_i = 1\}$ is a unit simplex.

2) A denotes a compact set which belongs to the interior of Δ^{n-1}

Let $f(x)$ be a Lipschitz function defined on A. Since A is compact and $f(x)$ is continuous on A, it achieves its minimum there.

3) We will denote the minimal value of $f(x)$ on A by m.

The min-function $\left(\frac{x}{y}\right)$ is defined on A^2. Since A^2 is compact and $\left(\frac{x}{y}\right)$ is continuous on it, it achieves its maximum there.

4) By M we will denote the maximal value of $\left(\frac{x}{y}\right)$ on A^2.

5) $N = \sup \max_i y_i$, $y = (y_1, ..., y_n) \in A$.

Theorem 4.1.1. *Let f be a Lipschitz function defined on A and L be the Lipschitz constant of the function f. Let*

$$L \leq \frac{m}{M\sqrt{n}N}$$

Then the positively homogeneous extension of f is an IPH function.

Proof. According to theorem 4.0.7 in order to prove that the function $f(x)$ can be extended homogeneously as an IPH function we should show that for any $x, y \in A \subset \Delta^{n-1}$ it satisfies the inequality $f(y)\left(\frac{x}{y}\right) \leq f(x)$. If $f(y) \leq f(x)$ then as x and y are incomparable and $\left(\frac{x}{y}\right) \leq 1$ the above inequality holds. Let us now look at the case when $f(y) \geq f(x)$. To prove the theorem we will need to evaluate the following expression

$$f(x)\left(\frac{1-\left(\frac{x}{y}\right)}{\left(\frac{x}{y}\right)}\right)$$

We can proceed as follows:

$$f(x)\left(\frac{1-\left(\frac{x}{y}\right)}{\left(\frac{x}{y}\right)}\right) \geq \frac{m}{M}\left(1-\left(\frac{x}{y}\right)\right) = \frac{m}{M}\left(1-\min_i \frac{x_i}{y_i}\right) = \frac{m}{M}\max_i\left(1-\frac{x_i}{y_i}\right) =$$

$$= \frac{m}{M}\max_i\left(\frac{y_i - x_i}{y_i}\right) = \frac{m}{M\sqrt{n}}\sqrt{n\max_i\left(\frac{y_i - x_i}{y_i}\right)^2} \geq$$

$$\geq \frac{m}{M\sqrt{n}}\sqrt{\frac{(y_1-x_1)^2}{y_1^2}+\ldots+\frac{(y_n-x_n)^2}{y_n^2}} \geq \frac{m}{M\sqrt{n}}\sqrt{\frac{(y_1-x_1)^2}{N^2}+\ldots\frac{(y_n-x_n)^2}{N^2}} =$$

$$= \frac{m}{M\sqrt{n}N}\sqrt{(y_1-x_1)^2+\ldots(y_n-x_n)^2} = \frac{m}{M\sqrt{n}N}|y-x| \tag{4.2}$$

As $f(x)$ is a Lipschitz function with a Lipschitz constant $L \leq \frac{m}{M\sqrt{n}N}$, we can write

$$\frac{m}{M\sqrt{n}N}|y-x| \geq L|y-x| \geq |f(y)-f(x)| = f(y)-f(x) \tag{4.3}$$

Hence, combining 4.2 and 4.3 we have obtained the following inequality

$$f(x)\left(\frac{1-\left(\frac{x}{y}\right)}{\left(\frac{x}{y}\right)}\right) \geq f(y)-f(x)$$

simplifying which we get the needed inequality

$$f(y)\left(\frac{x}{y}\right) \leq f(x)$$

Thus, we have shown that a positively homogeneous extension of a Lipschitz function $f(x)$ on a compact set $A \subset \Delta^{n-1}$ with a Lipschitz constant $L \leq \frac{m}{M\sqrt{n}N}$ is an IPH function. □

Definition 4.1.2. Let

$\widetilde{\Delta}_{n-1} = \{(x_1,...,x_{n-1}) \in \mathbb{R}^{n-1} : x_i \geq 0, i=1,...,n-1, x_1+...+x_{n-1} \leq 1\}$,

$\Delta^{n-1} = \{(x_1,...x_n) \in \mathbb{R}^n_{++}, \sum_{i=1}^n x_i = 1\}$

and $\Psi : \widetilde{\Delta}_{n-1} \to \Delta^{n-1}$ be the following mapping

$$\Psi(x_1,...,x_{n-1}) = (x_1,...,x_{n-1}, 1-(x_1+...+x_{n-1}))$$

We will refer to Ψ as a lifting function.

For any function $f(x_1, ..., x_{n-1} : \widetilde{\Delta}_{n-1}) \to \mathbb{R}$ we will denote by $\hat{f} : \Delta^{n-1} \to \mathbb{R}$ the following function

$$\hat{f}(x_1, ..., x_{n-1}, 1 - (x_1 + ... + x_{n-1}) = f(\Psi^{-1}(x_1, ..., x_{n-1}, 1 - (x_1 + ... + x_{n-1})))$$

4.2 Reduction to subset sum problem

The subset sum problem is one of the most well-known NP-complete problems. Traditionally there are three formulations of the problem:

(S1) Given a set of n positive integers $a_1, ..., a_n$ and a bound B, find a subset of the integers whose sum is closest to but does not exceed the bound.

(S2) Given a set of n integers, find a subset that sums to exactly 0.

(S3) Given a set of m integers $S = \{s_1, ..., s_m\}$ and an integer s, find a subset of S that sums exactly to s.

In what follows m in (S3) will be either equal to $n - 1$ or n. We will be using the third formulation of the problem to show that the following problem is NP-complete as well.

Before we proceed, we introduce the following class of functions.

$$f(x_1, ..., x_n) = 1 + \sum_{i=1}^{n} \left(\frac{x_i - \alpha}{\beta - \alpha}\right)^2 \left(\frac{x_i - \beta}{\beta - \alpha}\right)^2 + \left(\sum_{i=1}^{n} s_i \left(\frac{x_i - \alpha}{\beta - \alpha}\right) - s\right)^2 \tag{4.4}$$

Here $\alpha, \beta \in \mathbb{R}$, $x_i \in [\alpha, \beta]$, $i = 1, ..., n$ and $s_1, ..., s_n, s$ are integers. We will denote this class of functions by $\mathfrak{F}(\alpha, \beta)$.

Remark 4.2.1. Note that the value of the functions in $\mathfrak{F}(\alpha, \beta)$ is greater than or equal to 1 since the first term of the functions is 1 and the second and third terms are always nonnegative.

Let us formulate the following problem:

$(P(\alpha,\beta))$ Given a function $f \in \mathfrak{F}(\alpha,\beta)$, is the global minimum of this function equal to 1 or not?

Theorem 4.2.2. *The problem $(P(\alpha,\beta))$ is equivalent to the subset sum problem.*

Proof. Let $f(x) \in \mathfrak{F}(\alpha,\beta)$, that is

$$f(x) = 1 + \sum_{i=1}^{n} \left(\frac{x_i-\alpha}{\beta-\alpha}\right)^2 \left(\frac{x_i-\beta}{\beta-\alpha}\right)^2 + \left(\sum_{i=1}^{n} s_i \left(\frac{x_i-\alpha}{\beta-\alpha}\right) - s\right)^2$$

where $x_i \in [\alpha,\beta]$, $\alpha,\beta \in \mathbb{R}$ and $s_1,...,s_n,s$ are arbitrary integers.

Let us prove that the global minimum of function $f(x)$ is equal to 1 if and only if there exists a subset $\{s_{i_1},...,s_{i_k}\}$ of $\{s_1,...,s_n\}$ such that $\sum s_{i_j} = s$

Assume that the value of the function $f(x)$ is equal to 1. Then, as it is a sum of non-negative terms, both its second and third terms should be equal to 0. The second term of the above function is equal to 0 if and only if for arbitrary $i = 1,...,n$ either $x_i = \alpha$ or $x_i = \beta$. Therefore, for any $i = 1,...,n$, $\dfrac{x_i-\alpha}{\beta-\alpha}$ would be either equal to 1 or to 0. The third term can be, thus, equal to 0 if and only if $\sum s_i \left(\dfrac{x_i-\alpha}{\beta-\alpha}\right) = s$, that is if and only if there exist indices $\{i_1,...,i_k\} \subset \{1,...,n\}$ such that $x_{i_j} = \alpha$ and $\sum s_{i_j} = s$.

On the other hand, if there exists a set of indices $\{i_1,...,i_k\} \subset \{1,...,n\}$ such that $\sum s_{i_j} = s$ then we can choose $x_{i_1} = \beta,...,x_{i_k} = \beta$ and the rest of x_i equal to α. In this case, second and third terms of the function will be equal to 0 and, hence, the value of the function $f(x)$ will be equal to 1. That is the problem (P(α,β)) and the subset sum problem are reducible to each other. This completes the proof. □

Corollary 4.2.3. *$(P(\alpha,\beta))$ is NP-complete.*

As we mentioned above, subset sum (S3) is an NP-complete problem. Since its solution yields the solution to $(P(\alpha,\beta))$ and vice versa, $(P(\alpha,\beta))$ is NP-complete as well

Corollary 4.2.4. *The problem of finding the global minimum of a given function $f(x) \in \mathfrak{F}(\alpha,\beta)$ is NP-hard.*

Let us assume that there exists a polynomial time algorithm for finding the global minimum of $f(x)$. Then comparing this minimum with 1 would require additional polynomial time computation (polynomial of degree zero). This means that there would be a polynomial time algorithm for solving problem $(P(\alpha, \beta))$. We arrive at a contradiction. Hence, corollary 4.2.4 is proved.

Remark 4.2.5. Note that global optimization problems for functions $f(x) \in \mathfrak{F}(\alpha, \beta)$ and $a + bf(x)$, where a is an arbitrary real number and b is a positive real number, are equivalent in the sense that finding global optimum for one of them yields the solution for the other one.

4.3 NP-hardness of GO problem for IPH functions

Theorem 4.3.1. *Let $f(x)$ be a Lipschitz function defined on an $(n-1)$-dimensional box $B = \left[\frac{1}{2n}, \frac{2n-1}{2n(n-1)}\right]^{n-1}$, $B \subset \widetilde{\Delta}_{n-1}$ with the Lipschitz constant $L \leq \frac{m}{2n\sqrt{n}}$, where m is the minimal value of $f(x)$ on B and Ψ is the lifting function, $\hat{B} \subset \Delta^{n-1}$, $\hat{B} = \Psi(B)$. Let $\hat{f}(x) : \hat{B} \to \mathbb{R}$ be the function defined as*

$$\hat{f}(x_1, ..., x_{n-1}, 1-(x_1+...+x_{n-1})) = f(\Psi^{-1}(x_1, ..., x_{n-1}, 1-(x_1+...+x_{n-1})) = f(x_1, ..., x_{n-1})$$

Then $\hat{f}(x)$ is Lipschitz continuous on $\hat{B}$ and the positively homogeneous extension of $\hat{f}(x)$ to $\mathbb{R}^n_{++}$ is an IPH function.

Proof. First we show that $\frac{m}{2n\sqrt{n}}$ can be used as a Lipschitz constant for $\hat{f}(x)$ as well. Indeed, as $\frac{m}{2n\sqrt{n}}$ is a Lipschitz constant for $f(x)$,

$$\frac{|f(x) - f(y)|}{|x - y|} \leq \frac{m}{2n\sqrt{n}}$$

On the other hand, from the definition of function $\hat{f}(x)$ it follows that

$$\frac{|\hat{f}(\hat{x}) - \hat{f}(\hat{y})|}{|\hat{x} - \hat{y}|} \leq \frac{|f(x) - f(y)|}{|x - y|} \leq \frac{m}{2n\sqrt{n}}$$

Here $\hat{x} = (x_1, ..., x_{n-1}, 1 - (x_1 + ... + x_{n-1}))$

According to Theorem 4.1.1, it is sufficient to show that $\frac{m}{2n\sqrt{n}} \leq \frac{m}{M\sqrt{n}N}$ (here we use the notation of Theorem 4.1.1).

It is easy to check that $N \leq 1$ and $M \leq 2n$. Indeed, as $N = \sup \max_i y_i$, $y = (y_1, ..., y_n) \in B$, N will be less than or equal to the upper bound of the box B, that is $N \leq \frac{2n-1}{2n(n-1)} \leq 1$. M is the maximal value of min-function $\left(\frac{x}{y}\right)$, $x, y \in B$. Obviously, the maximum of the function $\left(\frac{x}{y}\right)$ can be attained at the boundary points of the box when $x_i = \frac{2n-1}{2n(n-1)}$ and $y_i = \frac{1}{2n}$, $i = 1, ..., n$. Hence, $M \leq \frac{\frac{2n-1}{2n(n-1)}}{\frac{1}{2n}} = \frac{2n-1}{n-1} \leq 2n$ Therefore, the inequality $\frac{m}{2n\sqrt{n}} \leq \frac{m}{M\sqrt{n}N}$ holds. □

Theorem 4.3.2. *Let $f(x) : B \to \mathbb{R}$ be defined as*

$$f(x_1, ..., x_{n-1}) =$$

$$= 1 + \frac{1}{[4(n-1)^2 + 2(n-1)(Q+s)Q] \cdot n\sqrt{n}} \left[\sum_{i=1}^{n-1} \left(\frac{x_i - \alpha}{\beta - \alpha}\right)^2 \left(\frac{x_i - \beta}{\beta - \alpha}\right)^2 + \left(\sum_{i=1}^{n-1} s_i \left(\frac{x_i - \alpha}{\beta - \alpha}\right) - s \right)^2 \right]$$

where $\alpha = \frac{1}{2n}$, $\beta = \frac{2n-1}{2n(n-1)}$, $s_1, ..., s_{n-1}, s$ are arbitrary integers and $Q = \sum_{i=1}^{n-1} s_i$, and let $\hat{f}(x)$ be the lifting of $f(x)$.

Then the positively homogeneous extension of $\hat{f}(x)$ to $\mathbb{R}^n_{++}$ is an IPH function.

Proof. Since the minimal value of $f(x)$ is 1, it is sufficient to check that $\frac{1}{2n\sqrt{n}}$ is a Lipschitz constant for $f(x)$. Indeed, according to the mean value theorem, $\frac{|f(x) - f(y)|}{\|x - y\|} = \left|\frac{\partial f(z)}{\partial \xi}\right|$, where z is some point between x and y, $x, y \in B$, and $\frac{\partial f}{\partial \xi}$ is the directional derivative of f along unit vector $\xi = (\xi_1, ..., \xi_{n-1})$, $\|\xi\| = 1$. Let us show that the value of the derivative of $f(x)$ is always less than or equal to $\frac{1}{2n\sqrt{n}}$. Indeed,

$$\left|\frac{\partial f(z)}{\partial \xi}\right| = \left|\sum_{i=1}^{n-1} \xi_i \frac{\partial f(z)}{\partial x_i}\right| \leq \sum_{i=1}^{n-1} |\xi_i| \left|\frac{\partial f(z)}{\partial x_i}\right| \leq \sum_{i=1}^{n-1} \left|\frac{\partial f(z)}{\partial x_i}\right|$$

For simplicity we will denote $\frac{1}{[4(n-1)^2 + 2(n-1)(Q+s)Q] \cdot n\sqrt{n}}$ by K. Then

$$\sum_{i=1}^{n-1} \left|\frac{\partial f(z)}{\partial x_i}\right| = K \sum_{i=1}^{n-1} \left| \frac{2}{\beta-\alpha}\left(\frac{x_i-\alpha}{\beta-\alpha}\right)\left(\frac{x_i-\beta}{\beta-\alpha}\right)^2 + \frac{2}{\beta-\alpha}\left(\frac{x_i-\alpha}{\beta-\alpha}\right)^2\left(\frac{x_i-\beta}{\beta-\alpha}\right) + \right.$$

$$\left. +2\left(\sum_{i=1}^{n-1} s_i\left(\frac{x_i-\alpha}{\beta-\alpha}\right) - s\right)\left(\frac{s_i}{\beta-\alpha}\right)\right| \leq$$

$$\leq K \sum_{i=1}^{n-1}\left[\frac{2}{\beta-\alpha}\left|\frac{x_i-\alpha}{\beta-\alpha}\right|\left|\left(\frac{x_i-\beta}{\beta-\alpha}\right)^2\right| + \frac{2}{\beta-\alpha}\left|\left(\frac{x_i-\alpha}{\beta-\alpha}\right)^2\right|\left|\frac{x_i-\beta}{\beta-\alpha}\right| + \right.$$

$$\left. + \left|2\left(\sum_{i=1}^{n-1} s_i\left(\frac{x_i-\alpha}{\beta-\alpha}\right) - s\right)\right|\left|\frac{s_i}{\beta-\alpha}\right|\right]$$

As $x_i \in [\alpha, \beta]$, for any $i = 1, ..., n$ $\left|\frac{x_i-\alpha}{\beta-\alpha}\right| \leq 1$ and $\left|\frac{x_i-\beta}{\beta-\alpha}\right| \leq 1$. Therefore, we can proceed as follows

$$\sum_{i=1}^{n-1}\left|\frac{\partial f(z)}{\partial x_i}\right| \leq K\sum_{i=1}^{n-1}\left[\frac{2}{\beta-\alpha} + \frac{2}{\beta-\alpha} + 2(Q+s)\left(\frac{s_i}{\beta-\alpha}\right)\right] =$$

$$= K\left[\sum_{i=1}^{n-1}\frac{4}{(\beta-\alpha)} + \left(\frac{2(Q+s)}{\beta-\alpha}\right)\sum_{i=1}^{n-1} s_i\right] =$$

$$= \frac{K}{\beta-\alpha}\left[\sum_{i=1}^{n-1} 4 + 2(Q+s))\sum_{i=1}^{n-1} s_i\right] = \frac{K}{\beta-\alpha}[4(n-1) + 2(Q+s)Q]$$

Since $\beta - \alpha = \frac{2n-1}{2n(n-1)} - \frac{1}{2n} = \frac{1}{2(n-1)}$, we can further write

$$\sum_{i=1}^{n-1}\left|\frac{\partial f(z)}{\partial x_i}\right| \leq K[8(n-1)^2+4(n-1)(Q+s)Q] = \frac{1}{2n\sqrt{n}}$$

Hence it follows that

$$\frac{|f(x)-f(y)|}{\|x-y\|} = \left|\frac{\partial f(z)}{\partial \xi}\right| \leq \sum_{i=1}^{n-1}\left|\frac{\partial f(z)}{\partial x_i}\right| \leq \frac{1}{2n\sqrt{n}}$$

that is

$$|f(x)-f(y)| \leq \frac{1}{2n\sqrt{n}}\|x-y\|$$

Therefore $\dfrac{1}{2n\sqrt{n}}$ can be considered as a Lipschitz constant of $f(x)$. The rest follows from the theorem 4.3.1. □

Theorem 4.3.3. *Global optimization problem for IPH functions restricted to a unit simplex is NP-hard.*

Proof. According to corollary 4.2.4 and remark 4.2.5 the global optimization problem for any function of form

$$f(x) = a + b\sum_{i=1}^{n}\left(\frac{x_i-\alpha}{\beta-\alpha}\right)^2\left(\frac{x_i-\beta}{\beta-\alpha}\right)^2 + \left(\sum_{i=1}^{n} s_i\left(\frac{x_i-\alpha}{\beta-\alpha}\right) - s\right)^2$$

where a and b are real numbers, is NP-hard. From theorem 4.3.2 it follows that the lifting of functions of this form onto a unit simplex can be extended as an IPH function. Therefore, the global optimization problem for IPH functions on a unit simplex is NP-hard. □

4.4 NP-hardness of ε-approximation

Since we can be almost certain that NP hard problems cannot be solved efficiently, we have to limit ourselves to find approximate solutions. Approximation algorithms can have different measure of efficiency, or so called performance guarantee. It may be absolute, as described in the following definition

Definition 4.4.1. An algorithm is said to be an ε-absolute approximation algorithm for a problem P, if for for some constant ε and for any instance I of this problem it outputs an estimate m' such that

$$|m' - m| \leq \varepsilon$$

where m is the exact solution of P.

More commonly though, performance guarantee is relative, that is the approximate value found by the algorithm is within some fixed percentage of the optimal value.

Definition 4.4.2. An algorithm is said to be an ε-approximation algorithm for a problem P, if for some constant ε and for any instance I of this problem it outputs an estimate m' such that

$$\frac{m}{1+\varepsilon} \leq m' \leq (1+\varepsilon)m$$

where m is the exact solution of P. Here $1+\varepsilon$ is called the approximation ratio.

However, even the problem of finding approximate solutions cannot always be tackled successfully. Some problems have extremely good approximation algorithms, while for others finding approximate solution is no easier than solving them exactly. For example, knapsack problem can be approximated within any approximation ratio [10]. The traveling salesman problem, on the other hand, cannot be solved approximately within any constant factor [28].

In the previous section we have shown that the global optimization problem for IPH functions restricted to a unit simplex is NP-hard. A natural question arises: is there a way to find efficient approximation algorithm for the above problem? In this section we will show that the problem cannot be approximated at all.

Consider the set of functions $f : \mathbb{R}^n \to \mathbb{R}$ of the following form

$$f(x) = \sum_{i=1}^{n} x_i^2(1-x_i)^2 + \left(\sum_{i=1}^{n} s_i x_i - s\right)^2 \tag{4.5}$$

here $s_1, ..., s_n, s > 0$ are arbitrary integers, $x \in \mathbb{R}^n$, and $0 \le x_i \le 1$ for any $i = 1, ..., n$. The following theorem holds

Theorem 4.4.3. *If the global minimum of function $f(x)$ on $I^n = [0,1]^n$ is not equal to 0 then it is greater than $\left(\frac{2Q-1}{4Q^2}\right)^2$, where $Q = \sum_{i=1}^{n} s_i$.*

Remark 4.4.4. Note that if there exists a subset $\{s_{i_1}, ..., s_{i_k}\}$ of $s_1, ..., s_n$ such that $\sum s_{i_j} = s$ then the global minimum of $f(x)$ is equal to 0, and vice versa. Therefore, the fact that the global minimum is not equal to 0 means that $|\sum s_{i_j} - s| > 1$.

Proof. Let us assume that for some $x \in I^n$ $f(x) < \left(\frac{2Q-1}{4Q^2}\right)^2$. Since all terms in the expression for $f(x)$ are nonnegative, for any $i = 1, ...n$ we have

$$x_i^2(1-x_i)^2 < \left(\frac{2Q-1}{4Q^2}\right)^2$$

or

$$x_i^2 - x_i + \frac{2Q-1}{4Q^2} > 0$$

Taking into account that $0 \le x_i \le 1$ and solving the quadratic inequality explicitly it is easy to see that $0 \le x_i < \frac{1}{2Q} \vee 1 - \frac{1}{2Q} < x_i \le 1$, $\frac{1}{2Q} < \frac{1}{2}$. Let us now evaluate the following expression

$$\left|\sum_{i=1}^{n} s_i x_i - s\right|$$

We will split the sum $\sum_{i=1}^{n} s_i x_i$ into two terms $\sum_{I} s_i x_i$ and $\sum_{J} s_j x_j$ where I is the set of indices for which $0 \le x_i < \frac{1}{2Q}$ and J is the set of indices for which $1 - \frac{1}{2Q} < x_j \le 1$. Therefore, we can write

$$\left|\sum_{i=1}^{n} s_i x_i - s\right| = \left|\sum_I s_i x_i + \sum_J s_j x_j - s\right| = \left|\sum_I s_i x_i + \sum_J s_j(1-\tilde{x}_j) - s\right|$$

Here $1-\tilde{x}_j = x_j$ and therefore $0 \leq \tilde{x}_j < \dfrac{1}{2Q}$

$$\left|\sum_I s_i x_i + \sum_J s_j(1-\tilde{x}_j) - s\right| = \left|\sum_I s_i x_i - \sum_J s_j \tilde{x}_j + \sum_J s_j - s\right| =$$

$$= \left|\left(\sum_J s_j - s\right) - \left(\sum_J s_j\tilde{x}_j - \sum_I s_i x_i\right)\right| \geq \left|\sum_J s_j - s\right| - \left|\sum_J s_j\tilde{x}_j - \sum_I s_i x_i\right| \geq$$

$$\geq \left|\sum_J s_j - s\right| - \left(\sum_J s_j\tilde{x}_j + \sum_I s_i x_i\right) \geq 1 - \frac{1}{2Q}\left(\sum_J s_j + \sum_I s_i\right) = 1 - \frac{1}{2Q}Q = \frac{1}{2}$$

Therefore, $\left(\sum_{i=1}^{n} s_i x_i - s\right)^2 \geq \dfrac{1}{4}$

Hence, it follows that $f(x) = \sum_{i=1}^{n} x_i^2(1-x_i)^2 + \left(\sum_{i=1}^{n} s_i x_i - s\right)^2 > \left(\sum_{i=1}^{n} s_i x_i - s\right)^2 \geq \dfrac{1}{4}$.

On the other hand, we assumed that $f(x) < \left(\dfrac{2Q-1}{4Q^2}\right)^2 < \left(\dfrac{2Q}{4Q^2}\right)^2 = \dfrac{1}{4Q^2} < \dfrac{1}{4}$.

We arrive at a contradiction. Thus the global minimum of $f(x)$ is greater than $\left(\dfrac{2Q-1}{4Q^2}\right)^2$.

□

Remark 4.4.5. Note that for functions of form

$$f(y) = 1 + K\left[\sum_{i=1}^{n}\left(\frac{y_i-\alpha}{\beta-\alpha}\right)^2\left(\frac{y_i-\beta}{\beta-\alpha}\right)^2 + \left(\sum_{i=1}^{n} s_i\left(\frac{y_i-\beta}{\beta-\alpha}\right) - s\right)^2\right] \tag{4.6}$$

where K, α, β are positive real numbers, $\alpha \leq y_i \leq \beta$ and $s_1, ..., s_n, s$ are arbitrary integers the global minimum is either equal to 1 or is greater than $1 + K\left(\dfrac{2Q-1}{4Q^2}\right)^2$

Theorem 4.4.6. *The problem of finding ε-absolute approximation to the global minimum of 4.6 is NP-hard.*

Proof. Suppose that we have an algorithm for solving this ε-absolute approximation problem in polynomial time. Then there would exist a polynomial algorithm for checking whether the exact minimum of function is equal to 1 or is greater than 1. Indeed, let $\varepsilon(Q,K) = K\left(\frac{2Q-1}{4Q^2}\right)^2$. Calculating $\varepsilon(Q,K)$ would require a polynomial amount of time. Next we find $\frac{\varepsilon(Q,K)}{2}$-absolute approximation m' to the global minimum m such that $|m'-m| < \frac{\varepsilon(Q,K)}{2}$. According to our assumption, this can be done in polynomial time. From Theorem 4.4.3 and Remark 4.4.5 follows that the global minimum of function $f(y)$ can either be 1 or be greater than $1+\varepsilon(Q,K)$. Then by comparing m' with $1+\frac{\varepsilon(Q,K)}{2}$ we can tell if $m=1$ or $m>1$. Indeed, if $m' > 1+\frac{\varepsilon(Q,K)}{2}$ then $m > m' - \frac{\varepsilon(Q,K)}{2} > 1$. On the other hand, if $m' < 1+\frac{\varepsilon(Q,K)}{2}$ then

$$m - m' < |m'-m| < \frac{\varepsilon(Q,K)}{2}$$

$$m < m' + \frac{\varepsilon(Q,K)}{2} < 1+\varepsilon(Q,K),$$

that is $m=1$.

We have just shown a polynomial way for checking whether global minimum of $f(y)$ is equal to 1 or not. However, we proved earlier (Theorem 4.2.2) that this problem is NP-hard. Thus, we have arrived at contradiction. Therefore, finding ε-absolute approximation to the global minimum of $f(y)$ is NP-hard. □

Theorem 4.4.7. *The problem of finding ε-approximation to the global minimum of 4.6 is NP-hard.*

Proof. Assume that we have a polynomial algorithm to find an ε-approximation E to the global minimum m of 4.6. Let us assume that $m=1$. By definition of ε-approximation, $\frac{E}{1+\varepsilon} \le m$. Since $m=1$, $E \le 1+\varepsilon$, that is $E-1 \le \varepsilon$. Then by definition of ε-absolute approximation, we can see that E is an ε-absolute approximation to the global minimum m as well. However, we have previously shown that the problem of finding ε-absolute approximation to the global minimum 4.6 is NP-hard. We have arrived at contradiction. Theorem is proved. □

Theorem 4.4.8. *Finding ε-absolute approximation and ε-approximation to the solution of global optimization problem for IPH functions restricted to a unit simplex is NP-hard*

Proof. Indeed, from theorems 4.4.6 and 4.4.7 follows that for functions of form

$$f(y) = 1 + K\left[\sum_{i=1}^{n}\left(\frac{y_i-\alpha}{\beta-\alpha}\right)^2\left(\frac{y_i-\beta}{\beta-\alpha}\right)^2 + \left(\sum_{i=1}^{n} s_i\left(\frac{y_i-\beta}{\beta-\alpha}\right) - s\right)^2\right]$$

ε-absolute approximation and ε-approximation problems are NP-hard. On the other hand, theorem 4.3.2 proposes that the liftings of functions of form $f(y)$ with $K = \dfrac{1}{[4(n-1)^2 + 2(n-1)(Q+s)Q]\cdot n\sqrt{n}}$ onto a unit simplex can be extended as IPH functions. Therefore, ε-absolute approximation and ε-approximation problem for these functions is NP-hard. □

Chapter 5

Conclusion

In this thesis we have looked into one problem of global optimization. We have given a short overview of convex and nonconvex optimization problems. It is known that nonconvex problems are, generally speaking, hard to solve. We, therefore, have narrowed our focus to one particular class of nonconvex optimization problems - the ones that can be described by means of monotonic, namely IPH functions.

We have talked about the classification of the complexity of problems, particularly about the classes of NP-complete and NP-hard problems, which can be thought of as intractable. Basic properties of IPH functions are covered in this thesis, along with the basic notions and propositions of abstract convex analysis which these functions are one element of. We have shown that any Lipschitz function can be described in terms of IPH functions on a unit simplex.

The results we have obtained demonstrate the NP-hardness of the global optimization problem for the IPH functions restricted to a unit simplex .

These results suggest that monotonicity of a function is not sufficient for solving the respective global optimization problem, thus additional restrictions should be applied to this class of functions to narrow it down further.

Bibliography

[1] http://www.claymath.org/millennium/

[2] Andramonov, M., Rubinov, A. and Glover, B., *Cutting angle method for minimizing increasing convex-along-rays functions*, Research Report, 1997, vol. 97(7)

[3] Bagirov, A. and Rubinov, A., *Modified versions of the cutting angle method*, Advances in Convex Analysis and Global Optimization: Honoring the Memory of C. Caratheodory (1873-1950), 2001, p. 245

[4] Bagirov, A. M. and Rubinov, A. M., *Cutting Angle Method and a Local Search*, Journal of Global Optimization, 2003, vol. 27(2):pp. 193–213

[5] Cormen, T., Leiserson, C., Rivest, R. and Stein, C., *Introduction to Algorithms*, MIT Press, 2001

[6] Dutta, J., Martínez-Legaz, J. and Rubinov, A., *Monotonic Analysis over Cones: I*, Optimization, 2004, vol. 53(2):pp. 129–146

[7] Dutta, J., Martínez-Legaz, J. and Rubinov, A., *Monotonic analysis over cones: II*, Optimization, 2004, vol. 53(5):pp. 529–547

[8] Dutta, J., Martínez-Legaz, J. and Rubinov, A., *Monotonic analysis over cones: III*, Journal of Convex Analysis, 2008, vol. 15(5):pp. 561–579

[9] Garey, M. and Johnson, D., *Computers and Intractability: A Guide to the Theory of NP-completeness*, WH Freeman San Francisco, 1979

[10] Ibarra, O. and Kim, C., *Fast Approximation Algorithms for the Knapsack and Sum of Subset Problems*, Journal of the ACM (JACM), 1975, vol. 22(4):pp. 463–468

[11] Jagannathan, R., *On some properties of programming problems in parametric form pertaining to fractional programming*, Management Science, 1966, pp. 609–615

[12] Kann, V., *On the approximability of NP-complete optimization problems*, Ph.D. thesis, Department of Numerical Analysis and Computing Science, Royal Institute of Technology, Sweden, 1992

[13] Karmarkar, N., *A new polynomial-time algorithm for linear programming*, Combinatorica, 1984, vol. 4(4):pp. 373–395

[14] Kearfott, R. and Kreinovich, V., *Beyond Convex? Global Optimization Is Feasible Only for Convex Objective Functions: A Theorem*, Journal of Global Optimization, 2005, vol. 33(4):pp. 617–624

[15] Klee, V. and Minty, G., *How good is the Simplex algorithm*, 1970

[16] Klein, P. N. and Young, N. E., *Approximation algorithms for np-hard optimization problems*, in *In Algorithms and Theory of Computation Handbook*, CRC Press, 1999

[17] Kutateladze, S. and Rubinov, A. M., *Minkowski Duality and Its Applications [in Russian],*, Nauka, Novosibirsk, 1976

[18] Matiyasevich, Y., *Enumerable sets are diophantine*, in *Doklady Akad. Nauk SSSR*, vol. 191, 1970 pp. 279–282

[19] Nesterov, Y. and Nemirovsky, A., *Self-Concordant Functions and Polynomial Time Methods in Convex Programming*, 1989

[20] Rubinov, A., *Abstract Convexity and Global Optimization*, Kluwer Academic Publishers, 2000

[21] Rubinov, A., *Monotonic Analysis*, Victoria International Conference, Victoria University, Wellington, New Zealand, 2004

[22] Rubinov, A. and Andramonov, M., *Lipschitz programming via increasing convex-along-rays functions*, Optimization Methods and Software, 1999, vol. 10(6):pp. 763–781

[23] Rubinov, A. and Andramonov, M., *Minimizing Increasing Star-shaped Functions Based on Abstract Convexity*, Journal of Global Optimization, 1999, vol. 15(1):pp. 19–39

[24] Rubinov, A. and Dutta, J., *Abstract convexity*, in Handbook of generalized convexity and generalized monotonicity, 2005, pp. 293–333

[25] Rubinov, A., Tuy, H. and Mays, H., *An alogrithm for monotonic global optimization problems*, Optimization, 2001, vol. 49(3):pp. 205–221

[26] Rubinov, A. M., *Towards monotonic analysis*, Nonlinear Analysis and Related Topics, Proceedings of Institute of Mathematics, Minsk, 1999, vol. 2:pp. 147–154

[27] Sahni, S., *Computationally Related Problems*, SIAM Journal on Computing, 1974, vol. 3:p. 262

[28] Sahni, S. and Gonzalez, T., *P-Complete Approximation Problems*, Journal of the ACM (JACM), 1976, vol. 23(3):pp. 555–565

[29] Singer, I., *Abstract convex analysis*, Wiley-Interscience, New York, 1997

[30] Sipser, M., *Introduction to the Theory of Computation*, PWS Publishing Company, Boston, MA, 1996

[31] Sudan, M., *Efficient Checking of Polynomials and Proofs and the Hardness of Approximation Problems*, Springer Berlin/Heidelberg, 1995

[32] Thi Hoai Phuong, N. and Tuy, H., *A Unified Monotonic Approach to Generalized Linear Fractional Programming*, Journal of Global Optimization, 2003, vol. 26(3):pp. 229–259

[33] Tikhomirov, V., *The Evolution of Methods of Convex Optimization*, The American Mathematical Monthly, 1996, vol. 103(1):pp. 65–71

[34] Tuy, H., *Monotonic Optimization: Problems and Solution Approaches*, SIAM Journal on Optimization, 2000, vol. 11(2):pp. 464–494

[35] Tuy, H., Al-Khayyal, F. and Thach, P., *Monotonic optimization: branch and cut methods*, Essays And Surveys In Global Optimization, 2005

[36] Vavasis, S., *Complexity issues in global optimization: a survey*, in *Handbook of Global Optimization*, Kluwer, 1995 pp. 27–41

[37] Wilf, H., *Algorithms and complexity*, Prentice-Hall Englewood Cliffs, NJ, 1986

[38] Ye, Y., *A new complexity result on minimization of a quadratic function with a sphere constraint*, Princeton Series In Computer Science, 1992, pp. 19–31

[39] Zhu, W., *Unsolvability of some optimization problems*, Applied Mathematics and Computation, 2006, vol. 174(2):pp. 921–926