ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/267399343

Transformation Techniques for Signomial Functions in Global Optimization

Article

CITATIONS
6

READS
52

1 author:

Andreas Lundell
Åbo Akademi University

21 PUBLICATIONS 137 CITATIONS

SEE PROFILE

All content following this page was uploaded by Andreas Lundell on 22 November 2014.

The user has requested enhancement of the downloaded file. All in-text references underlined in blue are added to the original document and are linked to publications on ResearchGate, letting you access and read them immediately.

Transformation Techniques for Signomial Functions in Global Optimization

Andreas Lundell

PhD Thesis in Applied Mathematics
Department of Mathematics
Åbo Akademi University

Åbo, Finland 2009

Preface

The work on this thesis was performed during the period 2006–2009 at the Process Design and Systems Engineering Laboratory at Åbo Akademi University under the supervision of professor Tapio Westerlund. I deeply acknowledge professor Westerlund for all his guidance through the years and for being the one who introduced me to this field of research to begin with.

In addition, I want to thank all my friends and colleagues at the Process Design and Systems Engineering Laboratory, as well as at the Department of Mathematics, in particular professor Göran Högnäs and Eva-Lena Nyby-Iljin for help with the practicalities.

I would never have been able to finish this thesis without financial backing. Therefore, the economic support from the Research Institute of the Foundation of Åbo Akademi University, as well as from the Academy of Finland is gratefully acknowledged.

Finally, I want to thank all my friends and family for providing suitable, and sometimes well-needed, distractions for me during the intensive last years. I would especially like to thank Kim L. for all the discussions — both the serious work-related ones and those simply meant for killing time. Finally, a special thanks goes out to Sofia, Kim A. and Mathilda for all the TV and movie nights.

Åbo, September 2009

Andreas Lundell

Svenskt sammandrag

Global optimering är ett område inom den tillämpade matematiken som fått allt större betydelse i takt med att förutsättningarna för numeriska beräkningar blivit bättre. Eftersom optimeringsproblem som baserar sig på exempelvis modeller av processer i industrin kan vara oerhört komplexa, krävs bra metoder för att kunna lösa dem effektivt. Speciellt optimeringsproblem som innehåller ickelinjäriteter och heltalsvariabler, så kallade MINLP- (Mixed integer nonlinear programming) problem, orsakar ofta svårigheter. Om problemet därtill inte är konvext kan det visa sig vara oerhört svårt att hitta den bästa lösningen. Sådana problem är vanligt förekommande i tillämpningar och därför är utvecklandet av lösningsmetoder för ickekonvexa MINLP-problem av stor betydelse.

I denna avhandling behandlas lösningsmetoder för en speciell klass av MINLP-problem, nämligen problem som innehåller så kallade signomialfunktioner. Eftersom alla polynom, och dessutom bi- och trilinjära termer kan anses vara specialfall av denna typ av funktion, är signomialfunktioner allmänt förekommande i optimeringsproblem. Signomialfunktioner är allmänt sätt olinjära och oftast inte konvexa, men det är dock möjligt att genom olika transformationer överföra det ickekonvexa problemet till en konvex relaxerad form, vars lösningsområde approximerar och överskattar det ursprungliga problemets.

Vilka transformationer som används har direkt inverkan på approximationens kvalitet, och därför är lösningseffektiviteten starkt beroende av vilken typ av transformationer som används. Härmed kan även stora prestandavinster erhållas genom att välja vissa typer av transformationer. I avhandlingen presenteras därför ett antal teoretiska resultat om de olika transformationernas approximeringsegenskaper, bland annat bevisas att vissa typer av transformationer alltid är bättre än andra.

Förutom dessa teoretiska bevis, presenteras en algoritm för att hitta den globala lösningen för MINLP problem som innehåller signomialfunktioner. Algoritmen är en vidareutveckling av en annan algoritm; det som är unikt för den nya algoritmens är att den innehåller en metod för att automatiskt bestämma en optimerad mängd av transformationer som överför problemet på en konvex överskattad form. Detta görs genom att lösa ett så kallat MILP- (Mixed integer linear programming) problem, alltså ett linjärt diskret optimeringsproblem. Slutligen presenteras i sista delen av avhandlingen SIGOPT, en numerisk lösare som använder sig av denna algoritm.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

The universe we live in is to a large extent nonlinear. Therefore, it is only natural that mathematical models trying to mimic the behavior of complex systems or processes also need to be nonlinear to provide an accurate interpretation. In global optimization, *i.e.*, when trying to find the global optimal solution to an optimization problem subject to certain constraints, nonlinear functions often lead to nonconvex problems. These types of problems have the property that they may have local solutions which are not globally optimal, and hence, it is usually not possible to say whether a solution found actually is the best one. Thus, special techniques are required for solving nonconvex problems.

Different methods for handling nonconvex problems exist; one such method is the main topic of this thesis: By replacing the nonconvex functions in the constraints with convex underestimators, *i.e.*, convex functions underestimating the original ones, a problem overestimating the original nonconvex one is obtained. This new problem is convex if all nonconvex functions have been replaced with convex underestimators, and thus, the problem can be solved with standard algorithms for convex problems. If the solution also fulfills the constraints in the original problem, it is also the global solution of the nonconvex problem, since the original problem is contained in the overestimated one.

In this thesis, a special class of optimization problems are considered, namely mixed integer nonlinear programming (MINLP) problems containing signomial functions. As the name states, MINLP problems can contain both integer and real variables. Integer variables are important for modeling discrete choices, *e.g.*, whether a certain machine is used or not in a plant. Problems involving integer variables are generally much harder to solve than problems with only real variables, since in theory, every combination of the values of the variables must be examined. In practice, however, methods exists that often cut down the number of combinations drastically.

The considered class of problems contains, as previously stated, signomial functions. This class of functions is rather common, because it includes, for example, polynomials as well as bilinear and trilinear terms. Since signomial functions often are highly nonlinear and nonconvex, optimization problems involving these are difficult to solve to global optimality, although several methods for solving such problems do exist. The transformation technique for signomial functions presented in this thesis involves applying certain types of single-variable transformations $x = T(X)$ to the individual variables in the signomial terms. The relations between the original variables x and transformation variables X are then approximated by linear functions. Since only the part of the problem involving the nonconvex signomial functions is altered, the result is a flexible technique for transforming the problem to a convex overestimated form.

As will be shown, there are many degrees of freedom when selecting the single-variable transformations. Since different types of transformations lead to different convex underestimators, it is important to know what impact the transformation types have on the errors appearing when the underestimators are introduced. Therefore, some theoretical results are presented regarding the underestimation errors of the different transformation types discussed. Also, the transformations are compared to other convex underestimators through some numerical examples.

Since the transformation technique is meant to be used as an integrated part of a global optimization algorithm, a method for automatically determining the transformations required for convexifying an MINLP problem involving signomial functions is also needed. The method described in the thesis is based on solving a mixed integer linear programming (MILP) problem and it provides an efficient way of obtaining an optimized set of transformations. Several properties of the transformation sets can be emphasized, *e.g.*, sets with as few transformations as possible or with certain types of transformations can be favored.

In the last part of the thesis the signomial global optimization (SGO) algorithm is presented. This algorithm for solving nonconvex MINLP problems containing signomial functions to global optimality utilizes the MILP method as a preprocessing step. The nonconvex problems are solved as sequences of overestimated convex subproblems which are the results of applying the single-variable transformations to the signomial terms. By improving the linear approximation of the nonlinear relation between the original and transformation variables in each iteration, the feasible region and solution of the overestimated problems will converge to the corresponding one of the nonconvex problem.

1.1 List of publications

This thesis is written as a monograph. It is, however, closely based on the following publications:

Paper I A. Lundell and T. Westerlund. Optimization of power transformations in global optimization. *Chemical Engineering Transactions*, 11:95–100, 2007.

Paper II A. Lundell, J. Westerlund and T. Westerlund. Some transformation techniques with applications in global optimization. *Journal of Global Optimization*, 43(2): 391–405, 2009. doi: 10.1007/s10898-007-9223-4.

Paper III A. Lundell and T. Westerlund. Exponential and power transformations for convexifying signomial terms in MINLP problems. In L. Bruzzone, editor, *Proceedings of the 27th IASTED International Conference on Modelling, Identification and Control*, pages 154–159. ACTA Press, 2008. ISBN 978-0-88986-711-6.

Paper IV A. Lundell and T. Westerlund. Convex underestimation strategies for signomial functions. *Optimization Methods and Software*, 24:505–522, 2009. doi: 10.1080/10556780802702278.

Paper V A. Lundell and T. Westerlund. On the relationship between power and exponential transformations for positive signomial functions. *Chemical Engineering Transactions*, 17:1287–1292, 2009. doi: 10.3303/CET0917215.

Paper VI A. Lundell and T. Westerlund. Implementation of a convexification technique for signomial functions. In J. Jezowski and J. Thullie, editors, *19th European Symposium on Computer Aided Process Engineering*, volume 26 of *Computer Aided Chemical Engineering*, pages 579–583. Elsevier, 2009. doi: 10.1016/S1570-7946(09)70097-5.

Paper VII A. Lundell and T. Westerlund. Optimization of transformations for convex relaxations of MINLP problems containing signomial functions. In *Proceedings 10th International Symposium on Process Systems Engineering*. Elsevier, 2009. ISBN 978-0-444-53472-9.

CHAPTER 2

Global optimization preliminaries

In this chapter, some important definitions and results from convex analysis and global optimization are summarized. In the first section, the notions of convex sets and functions are described and after this, the concept of convex underestimators of nonconvex functions is briefly discussed. Furthermore, the signomial functions — the main focus of this thesis — and a class of optimization problems containing this type of functions are defined. Finally, some techniques for formulating piecewise linear functions are given.

The definitions and theorems presented in this chapter are, for the most part, elementary results in convex optimization, and can be found in most of the standard introductory literature, *e.g.*, Boyd and Vandenberghe [2004].

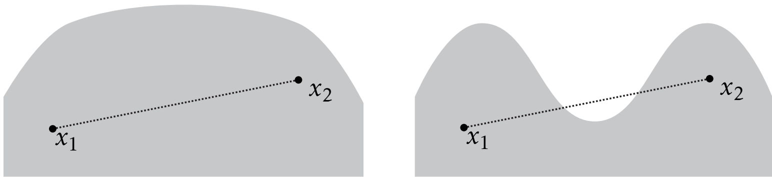
2.1 Definitions of convexity

In this section, convex sets and functions are defined and exemplified. At the end of the section some generalizations of convex functions — pseudo- and quasiconvex functions — are presented.

2.1.1 Convex sets

An important concept used throughout this thesis is that of the *convex set*. A set C in $\mathbb{R}^n$ is said to be convex if it contains all the line segments joining any two points belonging to the set. Since the line between the points x_1 and x_2 can be written as $\lambda x_1 + (1-\lambda)x_2$ for the parameter $\lambda \in [0,1]$, this can be expressed mathematically as

$$\forall x_1, x_2 \in C,\ \lambda \in [0,1]: \quad \lambda x_1 + (1-\lambda)x_2 \in C. \tag{2.1.1}$$



(**a**) A convex set (**b**) A nonconvex set

Figure 2.1: Convex and nonconvex sets

All sets which are not convex are called *nonconvex*. Some simple convex and nonconvex sets in $\mathbb{R}^2$ are illustrated in fig. 2.1. It can be noted that the multidimensional case, *i.e.*, $\mathbb{R}^n$ where $n > 2$, is analogous to the two-dimensional one.

Another important definition in both convex and nonconvex optimization is that of the convex hull. The *convex hull*, $\operatorname{conv} C$, of a set C is the minimal convex set which contains all points in C. If C is a convex set, then $\operatorname{conv} C \equiv C$, and if C is a nonconvex set, then $C \subset \operatorname{conv} C$.

If another convex set B contains the set C then it is always true that $\operatorname{conv} C \subseteq B$. Note that a discrete set is never convex (except when it consists of one point only) since it does not fulfill eq. (2.1.1). Instead, when speaking of a convex discrete set, the meaning is often that the integer-relaxed convex hull of the set is convex. For example, the set $C = \{1,2,3,4\}$ is not convex but the integer-relaxed set $\tilde{C} = [1,4]$ is.

2.1.2 Convex functions

Equally important as the definition of a convex set is the definition of a convex function. In global optimization, convex functions are especially useful, since it is possible to guarantee that a local minimum of a convex function is always global.

A *convex function* is defined as follows: A function $f : C \mapsto \mathbb{R}$, where C is a convex set, is convex if

$$\forall x_1, x_2 \in C,\ \lambda \in [0,1]: \quad f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2). \tag{2.1.2}$$

If the inequality in eq. (2.1.2) is strict, *i.e.*,

$$\forall x_1, x_2 \in C,\ \lambda \in [0,1]: \quad f(\lambda x_1 + (1-\lambda)x_2) < \lambda f(x_1) + (1-\lambda) f(x_2), \tag{2.1.3}$$

the function f is said to be *strictly convex*.

Conversely, if the inequality signs in eqs. (2.1.2) and (2.1.3) are reversed, the function is *concave* and *strictly concave* respectively. Also, if the function f is convex, then $-f$ is concave and if f is concave, then $-f$ is convex.

A function which is not convex is called nonconvex. Thus, a concave function is nonconvex, but not all nonconvex functions are concave. Illustrations of some convex, concave and nonconvex functions are given in fig. 2.2.

Using the definition of convex functions for determining whether a function is convex or not, is impractical. Therefore, more elaborate methods are needed, *e.g.*, the first- and second-order convexity conditions.

Theorem 2.1 (First-order convexity condition). Let f be a differentiable function on the convex set $C \subset \mathbb{R}^n$. Then f is convex if and only if

$$\forall x_1, x_2 \in C: \quad f(x_2) \geq f(x_1) + \nabla f(x_1)^T (x_2 - x_1). \tag{2.1.4}$$

The previous theorem simply states, that the tangent hyperplane of a convex function always underestimates the function. In fact, by using only local information, *i.e.*, the function value and the gradient in the point x_1, a global underestimating function is obtained. This is one of the most important properties of a convex function, and is often utilized in convex optimization methods, *e.g.*, when approximating a nonlinear constraint in an optimization problem with linear constraints.

Theorem 2.2 (Second-order convexity condition). Assume the function f is twice differentiable on the open convex set $C \subset \mathbb{R}^n$. Then $f(\mathbf{x})$ is convex if and only if its *Hessian matrix*

$$H(\mathbf{x}) = \nabla^2 f(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 f}{\partial {x_1}^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial {x_2}^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial {x_n}^2} \end{bmatrix} \tag{2.1.5}$$

is positive semidefinite for all $\mathbf{x} \in C$, *i.e.*, the matrix $H(\mathbf{x})$ has only nonnegative eigenvalues. Conversely, the function f is concave if $H(\mathbf{x})$ is negative semidefinite, *i.e.*, $H(\mathbf{x})$ has only nonpositive eigenvalues. If the matrix $H(\mathbf{x})$ is positive definite, *i.e.*, it has only positive eigenvalues, it is strictly convex.

A condition for the Hessian matrix to be positive semidefinite is that all principal minors of $H(\mathbf{x})$ are nonnegative. The k-th principle minor of a matrix is the determinant of the $k \times k$ submatrix consisting of the first k rows and columns of the original matrix.

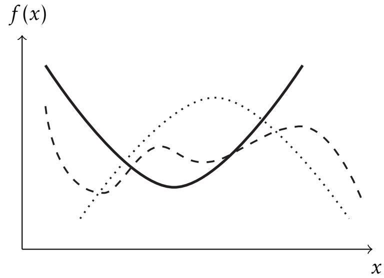
For a function $f(x)$ defined on an interval $[\underline{x}, \overline{x}] \subset \mathbb{R}$, thm. 2.2 reduces to the conditions $f''(x) \geq 0$ for convexity and $f''(x) \leq 0$ for concavity for all $x \in [\underline{x}, \overline{x}]$. The condition that $f''(x) \geq 0$ or more generally that $\nabla^2 f(x)$ is positive semidefinite, can be geometrically interpreted as that the graph of the function f has positive curvature in the point x. This can be seen in fig. 2.2, where some convex and nonconvex functions are illustrated.

The following theorem forms the basis of the convexification techniques in Chapter 3.

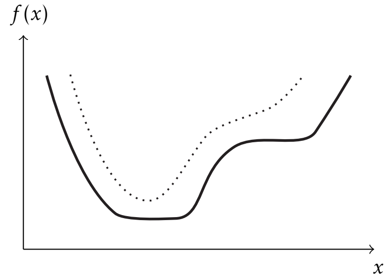
Theorem 2.3. If f_1, $f_2, \ldots,$ f_n are convex functions and w_i, $i = 1, \ldots, n$ are nonnegative real coefficients, then the weighted sum

$$f = w_1 f_1 + w_2 f_2 + \ldots + w_n f_n, \tag{2.1.6}$$

is a convex function.



(**a**) A convex (thick), concave (dotted) and nonconvex (dashed) function



(**b**) A quasiconvex (thick) and pseudoconvex (dotted) function

Figure 2.2: Illustrations of convex and nonconvex functions

However, it should be noted that the converse of thm. 2.3 is not generally true, *i.e.*, a sum of nonconvex or convex and nonconvex functions can still be convex. For example, the function $f(x,y) = x^2 + 2xy + y^2$ is convex on $\mathbb{R}^2$, although the term $2xy$ is nonconvex. This can be interpreted as that the terms x^2 and y^2 are "convex enough" to overpower the nonconvexity of the term $2xy$.

The previous theorem states that the set of convex functions is closed under addition and positive scaling, *i.e.*, the sum of convex functions is convex. This is generally not true for quasi- and pseudoconvex functions, two generalizations of convex functions presented in the next section.

2.1.3 Quasi- and pseudoconvex functions

Here two of the most important extensions of convex functions — quasi- and pseudoconvex functions — are defined.

A differentiable function $f : C \mapsto \mathbb{R}$, where $C \subseteq \mathbb{R}^n$ is a nonempty convex set, is *quasiconvex* if and only if the following condition holds

$$\forall x_1, x_2 \in C: \quad \nabla f(x_1)^T(x_2 - x_1) > 0 \Rightarrow f(x_2) > f(x_1). \tag{2.1.7}$$

An equivalent definition, is that f is quasiconvex if and only if the following is true

$$\forall x_1, x_2 \in C: \quad f(\lambda x_1 + (1-\lambda)x_2) \leq \max\{f(x_1), f(x_2)\}, \tag{2.1.8}$$

for all values on the parameter $\lambda \in [0,1]$.

Equation (2.1.8) states that the value of a quasiconvex function in an interval of its domain does not exceed the maximum values at the interval endpoints. From the definition, it can be deduced that quasiconvex functions can have local minima which are not global; for pseudoconvex functions, however, all the minima are global.

A differentiable function $f : C \mapsto \mathbb{R}$, where $C \subseteq \mathbb{R}^n$ is a nonempty convex set, is *pseudoconvex* if

$$\forall x_1, x_2 \in C: \quad \nabla f(x_1)^T(x_2 - x_1) \geq 0 \Rightarrow f(x_2) \geq f(x_1). \tag{2.1.9}$$

A pseudo- and a quasiconvex function is illustrated in fig. 2.2.

2.2 Convex underestimators

An important tool in nonconvex global optimization is the convex underestimator. Because an underestimating convex approximation or relaxation of the original problem can be obtained, *e.g.*, by replacing the nonconvex functions in the problem with convex underestimators of the functions, it is possible to use all methods available in convex optimization to solve the approximated version of the nonconvex problem.

The function $g(x)$ is a *convex underestimator* of a nonconvex function $f(x)$ for $x \in C$, where C is a convex set, if

(i) $g(x)$ is convex, and

(ii) $g(x) \leq f(x)$ for all $x \in C$.

For a given function, there are infinitely many convex underestimators, providing different levels of approximation errors. The tightest of these, *i.e.*, the underestimator with the smallest error, is called the convex envelope of the function.

The function $g(x)$ is the *convex envelope* of a nonconvex function $f(x)$ for $x \in C$, where C is a convex set, if

(i) $g(x)$ is a convex underestimator of $f(x)$, and

(ii) $g(x) \geq h(x)$ for all x and for all convex underestimators $h(x)$ of $f(x)$.

A problem in nonconvex optimization is that convex envelopes are known only for certain classes of functions, *i.e.*, there is no method available at the moment for obtaining the convex envelope of a general nonconvex function. In the following example, the convex envelopes for positive and negative bilinear terms, often called the McCormick underestimators (McCormick [1976]) are given. Other multilinear extensions can be found in, *e.g.*, Maranas and Floudas [1995].

Example 2.4. The convex envelopes of the positive bilinear function $f(x_1, x_2) = x_1 x_2$ and the negative bilinear function $g(x_1, x_2) = -x_1 x_2$ on the rectangular region $[\underline{x}_1, \overline{x}_1] \times [\underline{x}_2, \overline{x}_2]$ are given by

$$\hat{f}(x_1, x_2) = \max\{\underline{x}_1 x_2 + \underline{x}_2 x_1 - \underline{x}_1 \underline{x}_2,\ \overline{x}_1 z_2 + \overline{x}_2 z_1 - \overline{x}_1 \overline{x}_2\} \tag{2.2.1}$$

and

$$\hat{g}(x_1, x_2) = \max\{-\overline{x}_1 x_2 - \underline{x}_2 x_1 + \overline{x}_1 \underline{x}_2,\ -\underline{x}_1 x_2 - \overline{x}_2 x_1 + \underline{x}_1 \overline{x}_2\} \tag{2.2.2}$$

respectively.

The concepts of convex underestimators and the convex envelope are illustrated in fig. 2.3.

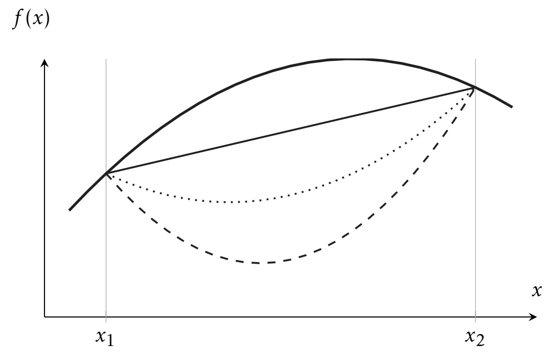


Figure 2.3: A nonconvex function (thick) and two convex underestimators (dashed and dotted) as well as the convex envelope (thin) on the interval $[x_1, x_2]$.

2.3 Signomial functions

In this section, a common type of functions in optimization problems — the signomial functions — is described. The optimization of problems containing this class of functions is the main topic of this thesis. The reason the signomials are so common, is that they constitute a rather large group of functions; for example, sums of polynomials and bi- and trilinear terms are all signomial functions. Signomials can be regarded as multidimensional extensions of polynomials where the powers are allowed to also assume noninteger values.

A *signomial function* is defined as the sum of signomial terms, which in turn consists of products of power functions. Thus, a signomial function of N variables and J signomial terms can be expressed mathematically as

$$\sigma(\mathbf{x}) = \sum_{j=1}^{J} c_j \prod_{i=1}^{N} x_i^{p_{ji}}, \tag{2.3.1}$$

where the coefficients c_j and the powers p_{ji} are reals. The variables x_i are here assumed to be positive reals or integers. Note that, if a certain variable x_i does not exist in the j-th term, then $p_{ji} = 0$.

A special type of signomial function, where all coefficients $c_j > 0$, $i = 1, \ldots, J$, *i.e.*, all terms are positive, is called a *posynomial function*. Hence, a signomial function can also be defined as the difference of two posynomial functions by grouping the positive and negative terms according to

$$\sigma(\mathbf{x}) = \sum_{j:\ c_j>0} c_j \prod_{i=1}^{N} x_i^{p_{ji}} - \sum_{j:\ c_j<0} |c_j| \prod_{i=1}^{N} x_i^{p_{ji}}, \tag{2.3.2}$$

Signomial functions are closed under addition, subtraction, multiplication and scaling with real constants; posynomials are closed under addition, multiplication and scaling with positive real constants.

2.3.1 Convexity of signomial functions

A signomial function is, in general, not convex. However, the necessary conditions for when a signomial function is convex can be derived. These conditions rely on the fact that a sum of convex terms is convex as stated in thm. 2.3; by determining whether the individual terms in the signomials are convex, the convexity of the whole function can be attained.

As will be shown in the following two theorems, the conditions for when a signomial term is convex depend on the sign of the term. The convexity requirements presented here have previously been given in Maranas and Floudas [1995]. However, the proof here is different. The first theorem gives the convexity requirements for positive signomial terms.

Theorem 2.5 (Convexity of a positive signomial term).
The signomial term $s(\mathbf{x}) = c \cdot x_1^{p_i} \cdots x_N^{p_N}$, where $c > 0$, is convex if one of the following two conditions is fulfilled:

(i) all powers p_i are negative, or

(ii) one power p_k is positive, the rest of the powers $p_i,\ i \neq k$ are negative and

$$\sum_{i=1}^{N} p_i \geq 1, \tag{2.3.3}$$

i.e., the sum of the powers is greater than or equal to one.

Proof Thm. 2.2 states that the function $s(\mathbf{x})$ is convex if the Hessian matrix $H(\mathbf{x})$ of the function is positive semidefinite. The second order partial derivatives of $s(\mathbf{x})$ are

$$\frac{\partial s(\mathbf{x})}{\partial x_i \partial x_j} = \begin{cases} c \cdot \frac{p_i p_j}{x_i x_j} \cdot s(\mathbf{x}) & \text{if } i \neq j, \\ c \cdot \frac{p_i(p_i - 1)}{x_i^2} \cdot s(\mathbf{x}) & \text{if } i = j, \end{cases} \tag{2.3.4}$$

so $H(\mathbf{x})$ will consist of the element $\frac{\partial s(\mathbf{x})}{\partial x_i \partial x_j}$ at position (i,j) in the matrix. Furthermore, it can be shown that

$$\det H(\mathbf{x}) = (-c)^N \left(\prod_{i=1}^{N} p_i x_i^{N p_i - 2} \right) \left(1 - \sum_{i=1}^{N} p_i \right). \tag{2.3.5}$$

For the matrix H to be positive semidefinite, all principal minors, $\det H_l, l = 1, \ldots, N$, of H must be positive, *i.e.*, $\forall l = 1, \ldots, N : \ \det H_l > 0$. The determinant of the l-th principal minor is

$$\det H_l(\mathbf{x}) = (-c)^l \left(\prod_{i=1}^{l} p_i x_i^{l p_i - 2} \right) \left(1 - \sum_{i=1}^{l} p_i \right). \tag{2.3.6}$$

(i) If $c > 0$, $x_i > 0$ and $p_i \leq 0$ are fulfilled, then $\det H_l(\mathbf{x}) > 0$, $\forall l = 1,\ldots,N$. Since all principal minors of H are positive, the function $s(\mathbf{x})$ is convex.

(ii) If $c < 0$, $x_i > 0$, $p_i \leq 0$, $i \neq k$, $p_k \geq 0$ and $\sum_{i=1}^{N} p_i \geq 1$ are fulfilled, then $\forall l = 1,\ldots,N$: $\det H_l(\mathbf{x}) > 0$. Since all principal minors of H are positive, the function $s(\mathbf{x})$ is convex. ∎

The corresponding convexity requirements for negative signomial terms are given by the following theorem.

Theorem 2.6 (Convexity of a negative signomial term).
The signomial term $s(\mathbf{x}) = c \cdot x_1^{p_i} \cdots x_N^{p_N}$, where $c < 0$, is convex if all powers p_i are positive and

$$0 \leq \sum_{i=1}^{N} p_i \leq 1, \tag{2.3.7}$$

i.e., the sum of the powers is between zero and one.

Proof From the proof of the previous theorem, when $c < 0$, $x_i \geq 0$ and $\sum_{i=1}^{N} p_i \leq 1$, all principal minors H_l of H are positive, and thus the function $s(\mathbf{x})$ is convex. ∎

2.4 Different classes of optimization problems

Many different classes of optimization problems exist. The most basic is the *linear programming* (LP) problem. In a LP problem, all variables assume real values and all constraints, as well as the objective function, are linear. If all variables in a LP problem are discrete, the problem is called an *integer programming* (IP) problem. Both the LP and IP problem classes are subclasses of the *mixed integer linear programming* (MILP) problem class. Here variables can be both real- and integer-valued.

If any of the constraints are nonlinear the problem is a *nonlinear programming* (NLP), *integer nonlinear programming* (INLP) or *mixed integer nonlinear programming* (MINLP) problem, respectively, depending on the types of variables in the problem. Note that all of the other classes are subclasses of the MINLP problem class. Thus, it is, in principle, possible to solve all of the other types of problems with a MINLP solver.

A problem is called convex if it contains a convex objective function and convex constraints. Of the above mentioned problem types, only LP problems are always convex. Note that, whenever a IP, MILP or MINLP problem is called convex, this is in the sense that the integer-relaxed problems are convex (compare to the discussion on convex discrete sets in Section 2.1.1).

Definition 2.7 (Mixed integer nonlinear programming (MINLP) problem). A MINLP problem can be formulated as

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}), \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{a}, \quad \mathbf{B}\mathbf{x} \leq \mathbf{b}, \\ & \mathbf{g}(\mathbf{x}) \leq 0, \end{aligned} \tag{2.4.1}$$

where the objective function $f(\mathbf{x})$ is convex, and $\mathbf{A}\mathbf{x} = \mathbf{a}$ and $\mathbf{B}\mathbf{x} \leq \mathbf{b}$ are linear equality and inequality constraints respectively. The constraints $\mathbf{g}(\mathbf{x}) \leq 0$ are nonlinear inequality constraints, which can be convex or nonconvex. The vector of variables $\mathbf{x} = [x_1, x_2, \ldots, x_N]$ can include either integer- or real-valued variables, or both.

The formulation in eq. (2.4.1) is very general, since it also allows nonlinear objective functions as well as nonlinear equality constraints. This is the case, since it is possible to write a nonlinear equality constraint $g(\mathbf{x}) = 0$ as two different constraints $g(\mathbf{x}) \leq 0$ and $-g(\mathbf{x}) \leq 0$; however, at least one of these constraints will always be nonconvex.

It is also possible to have a nonlinear objective function f, by introducing a variable μ as the new objective function and including the inequality $f(\mathbf{x}) - \mu \leq 0$ as an additional constraint. The minimization of μ will drive the objective function value to that of the original objective function $f(\mathbf{x})$. Note however, that the reformulated objective function constraint $g(x) = f(x) - \mu$ has different properties than those of the objective function $f(x)$ itself. If $f(x)$ is convex then $g(x)$ is also convex. However, if $f(x)$ is pseudoconvex, then $g(x)$ is generally not pseudoconvex, as the sum of two pseudoconvex functions need not be pseudoconvex.

Another subclass of the MINLP problem class is the *mixed integer signomial programming* (MISP) problem class — this class of problems is the main focus of this thesis. By using the algorithm described in Chapter 5, it can be solved iteratively to global optimality.

Definition 2.8 (Mixed integer signomial programming (MISP) problem). A MISP problem can be formulated as

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}), \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{a}, \quad \mathbf{B}\mathbf{x} \leq \mathbf{b}, \\ & \mathbf{g}(\mathbf{x}) \leq 0, \\ & \mathbf{q}(\mathbf{x}) + \sigma(\mathbf{x}) \leq 0. \end{aligned} \tag{2.4.2}$$

Here all variables and constraints are the same as in eq. (2.4.1), only the functions $\mathbf{q}(\mathbf{x})$ and $\sigma(\mathbf{x})$, consisting of convex and signomial functions respectively, are added. Also all variables x_i in the signomial functions $\sigma(\mathbf{x})$ are assumed to be positive*. An individual constraint $q_m(\mathbf{x}) + \sigma_m(\mathbf{x})$ is called a generalized signomial constraint. Analogous to the MINLP problem type, the problem is called a signomial programming (SP) problem if all variables are continuous.

The restriction that all variables in the signomial functions must be positive can, at first glance, seem like a severe restriction. However, since imaginary solutions to the problems are not allowed, negative values on the variables when the power is not an integer must be excluded. Also, zero has to be excluded in case a power is negative. There are, however, methods to overcome some of these shortcomings, for example, a lower bound of $\epsilon > 0$ can be set for variables with a lower bound of zero. Also, a translation $x_i' = x_i + \tau_i$, where x_i is a variable with (partly) negative domain or a lower bound of zero and τ_i is a positive parameter which fulfills $\tau_i > |\min x_i|$, can be used. This technique will, however, introduce additional variables in the problem if not all occurrences of the variable x_i in the problem are replaced by x_i'. Also, depending on the powers of x_i in the signomial terms, more signomial terms can appear, which is illustrated in the following example.

Example 2.9. The signomial constraint

$$x_1^2 x_2 - x_1 - x_2 \leq 0, \quad -3 \leq x_1 \leq 1, \quad 1 \leq x_2 \leq 5, \tag{2.4.3}$$

can be written in a form suitable for a MISP of the type in eq. (2.4.2) using the translation

$$\tilde{x}_1 = x_1 + 4, \quad 1 \leq \tilde{x}_1 \leq 5, \tag{2.4.4}$$

in which case the transformed constraint will be

$$(\tilde{x}_1 - 4)^2 x_2 - (\tilde{x}_1 - 4) - x_2 \;=\; \ldots \;=\; \tilde{x}_1^2 x_2 - 8\tilde{x}_1 x_2 - \tilde{x}_1 + 15x_2 + 4 \leq 0. \tag{2.4.5}$$

In this case, one nonconvex signomial term has been replaced with two nonconvex signomial terms.

Signomial programming is often called *generalized geometric programming* (GGP), because it can be regarded as an extension to geometric programming (GP), which has been studied since the early 1960's. A GP problem is not convex in its original form, but can through a simple transformation be written as a convex optimization problem. Geometric programming is presented, *e.g.*, in Boyd and Vandenberghe [2004] and Boyd et al. [2007].

2.5 Piecewise linear functions

Piecewise linear functions (PLFs) are, in the methods presented in this thesis, an integrated part of the transformation techniques for nonconvex MISP problems. PLFs can be used to approximate one-dimensional nonlinear functions in a given interval with linear functions. The simplest type of piecewise linear approximation is a straight line

*In general, strict inequalities are not allowed in numerical solvers, so this requirement is in reality that all variables must have a fixed positive lower bound of $\epsilon > 0$.

through the endpoints of the nonlinear function. Thus, a linear approximation $\hat{f}(x)$ of the function $f(x)$ defined on the interval $x \in [\underline{x}, \overline{x}]$ can be written as

$$\hat{f}(x) = f(\underline{x}) + \frac{f(\overline{x}) - f(\underline{x})}{\overline{x} - \underline{x}}(x - \underline{x}). \tag{2.5.1}$$

By including more breakpoints in the linearization, the approximation can be made finer. However, if additional breakpoints are added in addition to the interval endpoints, more elaborate formulations for the PLFs must be used; here two techniques, one using binary variables and one using special ordered sets, are described.

Note that a PLF approximating a nonlinear convex function will always overestimate the original function, and conversely, a PLF of a nonlinear concave function will always give an underestimation.

2.5.1 Piecewise linear functions using binary variables

For the function $f(x)$ assuming the values $X_k = f(x_k)$ at K consecutive points $x_k \in [\underline{x}, \overline{x}]$, $x_1 < x_2 < \ldots < x_K$, a PLF-approximation $\hat{f}(x)$ in the interval $[\underline{x}, \overline{x}]$ can be expressed as

$$\hat{f}(x) = \sum_{k=1}^{K-1} X_k b_k + (X_{k+1} - X_k)s_k, \tag{2.5.2}$$

where the relation between the original variable x and the binary variables b_k and real variables s_k are given by

$$x = \sum_{k=1}^{K-1} x_k b_k + (x_{k+1} - x_k)s_k \qquad \text{and} \qquad 0 \le s_k \le b_k. \tag{2.5.3}$$

Furthermore, only one of the binary variables b_k is allowed to be nonzero at the same time, *i.e.*,

$$\sum_{k=1}^{K-1} b_k = 1. \tag{2.5.4}$$

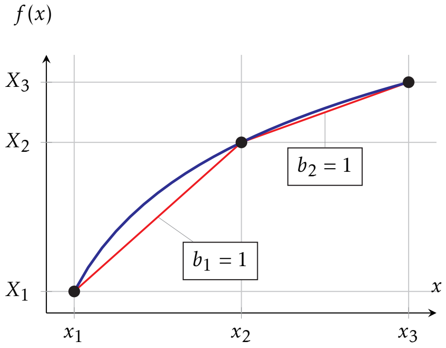
In this formulation two variables, one binary and one real, are added for each additional breakpoint included in the PLF.

There are also other formulations available for PLFs using binary variables, see *e.g.*, Floudas and Pardalos [2001].

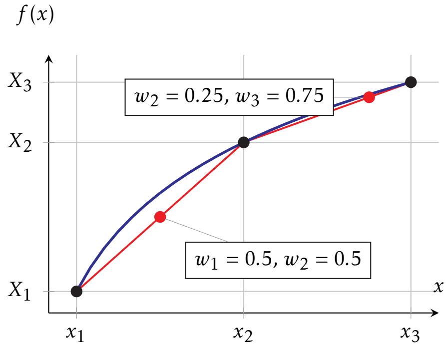
2.5.2 Piecewise linear functions using special ordered sets

It is also possible to formulate PLFs using so-called special ordered sets (SOS). Such formulations can be computationally more efficient in optimization problems than formulations using binary variables, as long as the solver supports SOS-type variables.

A *special ordered set* (SOS) is defined as a set of integers, continuous or mixed integer and continuous variables. There are mainly two different types of special ordered sets:



(a) Binary variable formulation



(b) SOS type 2 formulation

Figure 2.4: Approximation of a function using PLFs

SOS type 1 At most one variable in the set is nonzero.

SOS type 2 At most two variables in the set are nonzero, and if there are two nonzero variables, they must be adjacent in the set.

For example, the set $\{0, 0, \ldots, 0, a, 0, \ldots\}$, $a \in R$ is a SOS type 1 set and the sets $\{1, 0, \ldots, 0\}$ and $\{0, a, b, 0 \ldots, 0\}$, $a, b \in R$ are SOS type 2 sets.

In the PLF-formulation by Beale and Forrest [1976] given here, one SOS of type 2 $\{w_k\}_{k=1}^K$ is used, with the additional conditions that all variables w_k are positive real values between zero and one, and that the sum of the variables in the set is equal to one, *i.e.*,

$$\forall k = 1, \ldots, K: \ 0 \le w_k \le 1 \qquad \text{and} \qquad \sum_{k=1}^{K} w_k = 1. \tag{2.5.5}$$

The variable $x \in [\underline{x}, \overline{x}]$ can then be expressed as

$$x = \sum_{k=1}^{K} x_k w_k, \tag{2.5.6}$$

and the PLF $\hat{f}(x)$ approximating the function f in the interval $[\underline{x}, \overline{x}]$ becomes

$$\hat{f}(x) = \sum_{k=1}^{K} X_k w_k, \tag{2.5.7}$$

where the function f assumes the values $X_k = f(x_k)$ at the K consecutive points $x_k \in [\underline{x}, \overline{x}]$, $x_1 < x_2 < \ldots < x_K$. In this formulation, only one additional variable w is required for each breakpoint added to the PLF.

The binary variable and SOS type 2 formulations for PLFs are illustrated in fig. 2.4.

2.6 A brief review of the advances in signomial programming

The precursor to generalized geometric programming (GGP) or signomial programming (SP) was geometric programming (GP), which was first studied in the 1960's. The name geometric programming originates from the geometric-arithmetic mean inequality and was first introduced in Duffin et al. [1967]. The difference between geometric and signomial programming is that in the former, only positive signomial terms, *i.e.*, posynomials, are allowed in the objective function and constraints. A good overview of the early years of GP can be found in, *e.g.*, Peterson [2001] and an up-to-date tutorial on geometric programming and the different methods available for solving GP problems, is given in Boyd et al. [2007].

Today, even large GP problems can be solved quite efficiently through the means of different reformulation techniques. This is not the case, however, for SP problems; most methods available today solves the problem as converging linear on nonlinear subproblems. For SP problems containing discrete variables, *i.e.*, MISP problems, the outlook is even more dire, as even small problems can cause difficulties.

For continuous SP problems, early summaries of solution methods can be found in Dembo [1978] and Rijckaert and Martens [1978]. A newer global optimization (GO) algorithm is provided in Maranas and Floudas [1997]: By using exponential transformations, the algorithm transforms the signomial functions in the nonconvex constraints to differences of two convex functions, which are then underestimated. After this, a branch-and-bound type algorithm is used to find the global optimal solution by refining the convex underestimators. Another branch-and-bound type GO algorithm employing linear relaxations for continuous SP problems is discussed in Sherali [1998]. One of the main features of this algorithm is that it allows the variables in the signomial terms to take on the value zero. Signomial GO algorithms for SP problems containing only continuous variables are also discussed in Qu et al. [2007, 2008], Shen and Jiao [2006], Shen and Zhang [2004], Shen et al. [2008a,b] and Wang and Liang [2005].

Some conditions for when a signomial term is convex, as well as convex envelopes and tight convex underestimators for certain types of signomial functions, were provided in Maranas and Floudas [1995]. These convexity conditions for signomial terms can also be obtained using the concept of power convex functions from Lindberg [1981], which was applied to signomials in Björk et al. [2003]. Using these conditions, it is possible to deduce a transformation scheme, involving single-variable power transformations applied to the individual variables in the signomial terms in combination with piecewise linear functions, for obtaining convex underestimators of the nonconvex terms. These transformations along with the exponential transformation have been studied in, *e.g.*, Björk et al. [2003] and Pörn et al. [1999, 2008]. The work in this thesis is based on these transformations. Power transformations for use in a branch-and-bound type method were also studied in Li et al. [2007] and Lu et al. [2009].

In the works by Maranas and Floudas [1997] and Pörn et al. [1999], translations were used to handle nonpositive variables occurring in the signomial functions. This approach has the drawback that it introduces additional signomial terms in the signomial functions. Another method is the one in, *e.g.*, Tsai and Lin [2006, 2008], Tsai et al. [2007]

and Tsai [2009], where the problems containing so-called free variables are converted into other forms containing only nonnegative variables. In Tsai and Lin [2006] a GO algorithm for solving such problems involving discrete variables only is presented, and it is extended in Tsai et al. [2007] to also allow for MISP problems with free variables.

In Westerlund and Westerlund [2003] and Westerlund [2005], the generalized geometric programming extended cutting plane (GGPECP) algorithm was introduced. It solves nonconvex MISP to global optimality as a sequence of overestimated convex subproblems. This algorithm is also the foundation for the SGO algorithm described in Chapter 5. The αBB algorithm, described in, *e.g.*, Adjiman et al. [1998] and Floudas [1999], is a general solver for MINLP problems containing nonconvex twice-differentiable functions and can, thus, also solve MISP problems. Finally, an approximative optimization algorithm for MISP problems is presented in Chang [2005].

CHAPTER 3

Convex underestimation of signomial functions

In this chapter, some methods for obtaining convex underestimators for signomial functions are presented. The power and exponential transformations for underestimating signomial functions termwise are described in detail, and the relations in underestimation errors between these are examined. Finally, they are compared numerically to other types of convex underestimators, *e.g.*, the αBB underestimator from Adjiman et al. [1998]. The theoretical results in this chapter are mostly from Papers IV and V, but the transformation techniques have also been described in Papers I, II and III.

3.1 The transformation procedure

The procedure for obtaining convex underestimators of signomial functions described in this chapter is a two-step process. In the first step, the nonconvex signomial terms are convexified using certain types of transformations applied to the individual variables. In the second step, the transformations, or more exactly, the inverse transformations are approximated by PLFs. Since the PLF-formulation requires additional variables, the transformed problem will be convex in the extended space of relaxed variables, consisting of the original as well as the transformation variables. The transformed problem will, thus, be more complex, but since the relaxed transformed problem is convex, it can be solved with a convex MINLP solver*.

The problem is assumed to be of the form in def. 2.8, *i.e.*, a MISP problem. From here on, the index j will correspond to the j-th signomial term in the whole problem, since it

*A convex MINLP solver is a solver which solves MINLP problems, which are convex when the discrete variables are relaxed, to global optimality. Often this type of solver can also solve nonconvex problems, but without guarantee of finding the optimal solution (or any solution at all).

does not matter in the transformation procedure in which constraint the term is located. The total number of signomial terms in the problem is assumed to be J_T.

The transformation procedure for an individual generalized signomial constraint can be illustrated as follows

$$q_m(\mathbf{x}) + \sigma_m(\mathbf{x}) \leq 0 \xrightarrow{(i)} q_m(\mathbf{x}) + \sigma_m^C(\mathbf{x}, \mathbf{X}) \leq 0 \xrightarrow{(ii)} q_m(\mathbf{x}) + \sigma_m^C(\mathbf{x}, \hat{\mathbf{X}}) \leq 0. \tag{3.1.1}$$

In step *(i)*, the nonconvex signomial terms in $\sigma_m(\mathbf{x})$ are convexified by single-variable transformations $x_i = T_{ji}(X_{ji})$, where X_{ji} is the transformation variable. The problem is a reformulated, but still nonconvex, version of the original one, since nonlinear equality constraints describing the relations between the original and transformation variables, *i.e.*, the inverse transformation $X_{ji} = T_{ji}{}^{-1}(x_i)$, are included in the new problem.

However, in step *(ii)* the nonlinear expressions for the inverse transformations are approximated with PLFs and the feasible region of the original and transformed problems will no longer be identical. In fact, the feasible region of the original problem will be overestimated by a relaxed convex region, and therefore the solution to the transformed problem will be a lower bound of the original problem. By improving the approximation of the inverse transformation, *i.e.*, iteratively including more breakpoints in the PLFs, the lower bound will improve and, under certain conditions, converge to the global optimal solution of the original problem. An illustration of how the general MISP problem is transformed using the mentioned procedure is given in fig. 3.1.

This procedure forms the basis of the global optimization algorithm for signomial functions presented in Chapter 5. Note that different sets of transformations applied to the nonconvex signomial terms result in different overestimated feasible regions in the transformed problem. These can have substantial differences in complexities, and therefore, it is of great importance to find transformations providing as tight relaxations as possible.

3.2 The single-variable transformations

The goal is to find such single-variable transformations $x = T(X)$ so that when applied to a nonconvex signomial term, the term will be convex. An additional condition is that the convexified term must be underestimated when the relation between the transformation and original variables $X = T^{-1}(x)$ are approximated with PLFs. Here two different classes of transformations are explained, the exponential transformation and power transformations. The transformations have been studied previously by several authors. For example, convex underestimators for signomial terms based on power transformations for use in a branch-and-bound (BB) type framework have been discussed in Li et al. [2007] and Maranas and Floudas [1995, 1997]. The power and exponential transformations for convexifying and underestimating signomial terms using the PLF-technique, has been presented in, *e.g.*, Pörn [2000], Björk [2002] and Westerlund [2005]. Some general results about single-variable transformations in geometric and signomial programming are given in Gounaris and Floudas [2008].

$$
\begin{aligned}
\text{minimize} \quad & f(\mathbf{x}), \\
\text{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{a}, \quad \mathbf{B}\mathbf{x} \le \mathbf{b}, \\
& \mathbf{g}(\mathbf{x}) \le 0, \\
& \mathbf{q}(\mathbf{x}) + \sigma(\mathbf{x}) \le 0.
\end{aligned}
$$

Step *(i)*: The nonconvex signomial terms are convexified using single-variable transformations $x_i = T_{ji}(X_{ji})$

$$
\begin{aligned}
& \mathbf{q}(\mathbf{x}) + \sigma_{\mathbf{C}}(\mathbf{x}, \mathbf{X}) \le 0, \\
& X_{ji} = T_{ji}{}^{-1}(x_i), \\
& \qquad \forall i = 1, \dots, I, \; j = 1, \dots, J_T.
\end{aligned}
$$

Step *(ii)*: The inverse transformations $X_{ji} = T_{ji}{}^{-1}(x_i)$ are approximated by the PLFs $\hat{X}_{ji}$

$$
\begin{aligned}
\text{minimize} \quad & f(\mathbf{x}), \\
\text{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{a}, \quad \mathbf{B}\mathbf{x} \le \mathbf{b}, \\
& \mathbf{g}(\mathbf{x}) \le 0, \\
& \mathbf{q}(\mathbf{x}) + \sigma_{\mathbf{C}}(\mathbf{x}, \hat{\mathbf{X}}) \le 0, \\
& \hat{X}_{ji} \text{ is the PLF of } X_{ji} = T_{ji}{}^{-1}(x_i), \\
& \qquad \qquad \forall i = 1, \dots, I, \; j = 1, \dots, J_T.
\end{aligned}
$$

Figure 3.1: The two-step transformation procedure

Since the convexity requirements, according to thms. 2.5 and 2.6, are different for positive and negative signomial terms, it is only natural that different types of transformations are required in these two cases.

3.2.1 Transformations for positive terms

According to thm. 2.5, a positive signomial term is convex if either all the powers are negative, or exactly one is positive and the sum of the powers is greater than or equal to one. However, in addition to the convexity requirements, the signomial term must also be underestimated when the PLFs are used to approximate the inverse transformations. A detailed discussion on this topic can be found in Westerlund [2005].

The result is that two different compositions of power transformations (PTs) — the negative power transformation (NPT) and positive power transformation (PPT) — can be used to obtain a convex underestimator for a positive signomial term. The first one, the NPT, corresponds to the case when all powers of the signomial term are negative.

Definition 3.1 (Negative power transformation, NPT). The NPT convex underestimator for a positive signomial term is obtained by applying the transformation

$$x_i = X_i^{Q_i}, \quad Q_i < 0, \tag{3.2.1}$$

to all variables x_i with positive powers ($p_i > 0$) as long as the inverse transformation

$$X_i = x_i^{1/Q_i} \tag{3.2.2}$$

is approximated by a PLF $\hat{X}_i$.

In the NPT, the requirement for convexification is that $Q_i < 0$ for all i such that p_i is positive. For the PLF to underestimate the inverse transformation, the function $X_i = x_i^{1/Q_i}$ must be concave, which is also the case whenever $Q_i < 0$. Therefore, no extra underestimation condition on Q_i is needed.

The other type of transformation, the PPT, corresponds to the case of convexity when exactly one power in the signomial term is positive.

Definition 3.2 (Positive power transformation, PPT). The PPT convex underestimator for a positive signomial term is obtained by applying the transformation

$$x_i = X_i^{Q_i}, \tag{3.2.3}$$

to all variables with positive powers, where the transformation powers $Q_i < 0$ for all indices i except for one ($i = k$), where $Q_k \geq 1$, as long as the condition

$$\sum_{i:p_i>0} p_i Q_i + \sum_{i:p_i<0} p_i \geq 1 \tag{3.2.4}$$

is fulfilled, and the inverse transformation

$$X_i = x_i^{1/Q_i} \tag{3.2.5}$$

is approximated by a piecewise linear function $\hat{X}_i$.

In the PPT for the transformations with negative powers, Q_i, $i \neq k$, the same underestimation requirements on the powers Q_i as discussed above for the NPT are valid, *i.e.*, $Q_i < 0$ is required for the PLFs to correctly underestimate the inverse transformation. For the transformation with positive power, $x_k = X^{Q_k}$, the requirement for convexity is that the power Q_k is positive. For the PLF to underestimate the inverse transformation $X_k = x_k^{1/Q_k}$, the function $X_k(x_k)$ must be concave, which is the case if $Q_k \geq 1$. Combining the convexification and underestimation requirements then gives $Q_k \geq 1$. Note that if $Q_k = 1$, the k-th variable is not transformed since $X_k = x_k$.

Thus, by using the NPT or PPT, a positive nonconvex signomial term $s(\mathbf{x})$ is convexified and underestimated as follows:

$$s(\mathbf{x}) = c \prod_i x_i^{p_i} = c \prod_{i:p_i<0} x_i^{p_i} \cdot \prod_{i:p_i>0} X_i^{p_i Q_i} = s_C(\mathbf{x}, \mathbf{X}) \geq s_C(\mathbf{x}, \hat{\mathbf{X}}). \tag{3.2.6}$$

An illustration of the transformation procedure for convexifying the positive signomial term $x_1 x_2$, *i.e.*, a bilinear term, using the NPT and PPT is illustrated in fig. 3.2.

Another transformation which can be used to transform a nonconvex positive signomial term to a convex underestimated form is the exponential transformation. It is based on the following theorem.

Theorem 3.3. The function

$$f(\mathbf{x}) = c \cdot e^{p_1 x_1 + p_2 x_2 + \ldots + p_i x_i} \cdot x_{i+1}^{p_{i+1}} x_{i+2}^{p_{i+2}} \cdots x_I^{p_I}, \tag{3.2.7}$$

where $c > 0$, $p_1, \ldots, p_i > 0$ and $p_{i+1}, \ldots, p_I < 0$, is convex on $\mathbb{R}^n_+$.

Proof The function $f(\mathbf{x})$ can be rewritten according to

$$\begin{aligned} f(\mathbf{x}) &= c \cdot \exp(p_1 x_1 + \ldots + p_i x_i) \cdot x_{i+1}^{p_{i+1}} \cdots x_I^{p_I} \\ &= c \cdot \exp(p_1 x_1 + \ldots + p_i x_i + p_{i+1} \ln x_{i+1} + \ldots + p_I \ln x_I). \end{aligned} \tag{3.2.8}$$

Since $\ln x_k$, $k = i+1, \ldots, I$ is concave, $p_k \ln x_k$ is convex because all powers p_k, $k = i+1, \ldots, I$ are negative. Thus the function in eq. (3.2.8) is of the type $f(x) = c \cdot e^{g(x)}$, where g is a convex function, which is convex according to standard convex analysis, see for example Boyd and Vandenberghe [2004]. ∎

By transforming a positive signomial term to the form in eq. (3.2.7) using single-variable transformations, which can be approximated and underestimated using PLFs, the transformed term could be used as a convex underestimator for the original nonconvex term. This type of transformation is called the exponential transformation (ET).

Definition 3.4 (Exponential transformation, ET). A convex underestimator for a positive signomial term is obtained by applying the transformation

$$x_i = e^{X_i} \tag{3.2.9}$$

to the individual variables with positive powers as long as the inverse transformation

$$X_i = \ln x_i \tag{3.2.10}$$

is approximated by a PLF $\hat{X}_i$.

Since the inverse transformation $X_i = \ln x_i$ is concave for $x_i > 0$, it will always be underestimated by a PLF, so the ET can be used to convexify and underestimate a positive signomial term.

The ET applied to a positive nonconvex term convexifies it according to:

$$s(\mathbf{x}) = c \prod_{i:p_i<0} x_i^{p_i} \cdot \prod_{i:p_i>0} x_i^{p_i} = c \prod_{i:p_i<0} x_i^{p_i} \cdot \prod_{i:p_i>0} e^{p_i X_i} = s_C(\mathbf{x}, \mathbf{X}) \geq s_C(\mathbf{x}, \hat{\mathbf{X}}). \tag{3.2.11}$$

3.2.2 Transformations for negative terms

According to thm. 2.6, a negative signomial term is convex if all powers are positive, and the sum is greater than zero but less than or equal to one. This requirement can be adapted to obtain the transformation composition given here. To differentiate it from the NPT and PPT for positive signomial terms, it is here simply called the power transformation (PT) for negative terms.

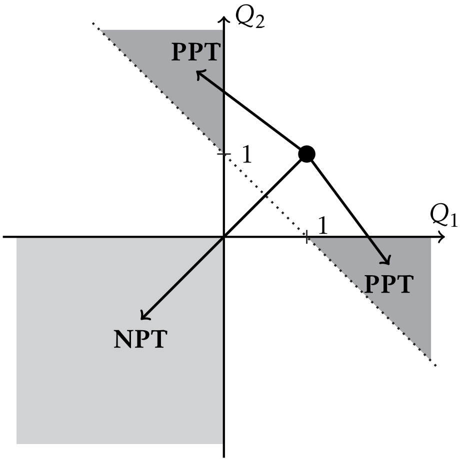
Definition 3.5 (Power transformation for negative terms). A convex underestimator for a negative signomial term is obtained by applying the transformation

$$x_i = X_i^{Q_i}, \tag{3.2.12}$$

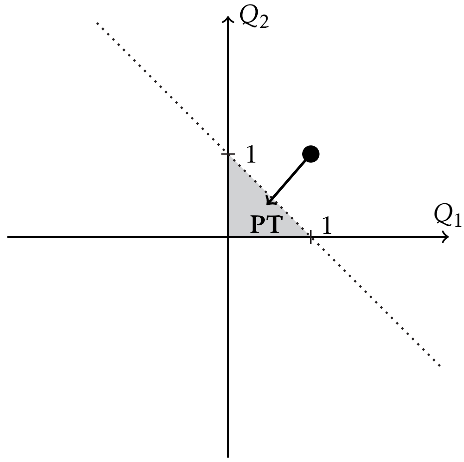
where $0 < Q_i \leq 1$ for all variables with positive powers and $Q_i < 0$ for all variables with negative power, to the individual variables in the term. Furthermore, the condition

$$0 < \sum_i p_i Q_i \leq 1, \tag{3.2.13}$$

must be fulfilled and the inverse transformation $X_i = x_i^{1/Q_i}$ approximated by a PLF $\hat{X}_i$.



(a) PPT and NPT



(b) PT for negative terms

Figure 3.2: Schematic overviews of how the PPT and NPT transforms the bilinear term $x_1x_2 \to X_1^{Q_1}X_2^{Q_2}$, and the PT for negative terms the bilinear term $-x_1x_2 \to -X_1^{Q_1}X_2^{Q_2}$. The colored regions indicates for which combination of powers the terms are convex.

Similar to the case with the positive signomial term, conditions guaranteeing that the convexified term really underestimates the original term must be introduced. Since the term now is negative, the PLFs must overestimate the inverse transformations $X_i = x_i^{1/Q_i}$. This is true if the inverse transformations are convex. Thus, for the variables with positive powers, where $Q_i > 0$ is the convexity condition, the transformation power must fulfill $Q_i \leq 1$, so the combined requirements become $0 < Q_i \leq 1$. For variables with negative powers, the convexity as well as underestimation requirements are $Q_i < 0$. So, no additional underestimation requirement is needed in addition to these for variables with negative powers.

Thus, by using the PT for negative signomial terms, the nonconvex term $s(\mathbf{x})$ is convexified and underestimated as follows

$$s(\mathbf{x}) = c\prod_i x_i^{p_i} = c\prod_{i:p_i<0} X_i^{p_iQ_i} \cdot \prod_{i:p_i>0} X_i^{p_iQ_i} = s_C(\mathbf{x},\mathbf{X}) \geq s_C(\mathbf{x},\hat{\mathbf{X}}). \tag{3.2.14}$$

The transformation procedure for convexifying the negative bilinear term $-x_1x_2$ is illustrated in fig. 3.2.

3.3 An illustrative example

In this section, a simple example of the transformations applied to a univariate signomial function is provided to illustrate how the convexification and underestimation procedure works.

Example 3.6. The nonconvex function

$$f(x) = 0.05x^3 - 8x + 25x^{0.5}, \quad 0 < \underline{x} \le x \le \overline{x}, \tag{3.3.1}$$

consist of three signomial terms. According to thms. 2.5 and 2.6 the first two terms are convex and only the last is nonconvex. Since the nonconvex term $25x^{0.5}$ is positive, either of the PPT, NPT or ET can be used to transform it. Applying any of the PTs with the power Q will give the convex underestimator

$$f_P(x, \hat{X}_P) = 0.05x^3 - 8x + 25\hat{X}_P^{0.5Q}, \tag{3.3.2}$$

where the piecewise linear approximation in one step of the inverse transformation $X_P = x^{1/Q}$ is given as

$$\hat{X}_P(x) = \underline{x}^{1/Q} + \frac{\overline{x}^{1/Q} - \underline{x}^{1/Q}}{\overline{x} - \underline{x}}(x - \underline{x}). \tag{3.3.3}$$

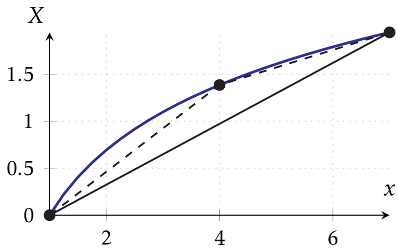
In the PPT, the transformation power Q must be larger than or equal to two to fulfill $0.5Q \ge 1$, and in the NPT, Q must be negative. The corresponding underestimator for the ET is

$$f_E(x, \hat{X}_E) = 0.05x^3 - 8x + 25e^{0.5\hat{X}_E}, \tag{3.3.4}$$

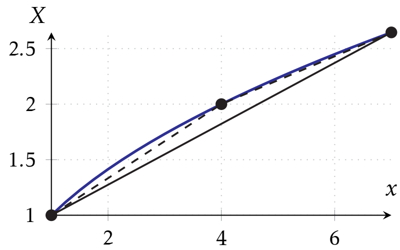
where the linear approximation of the inverse transformation $X_E(x) = \ln x$ in one interval is given as

$$\hat{X}_E(x) = \ln \underline{x} + \frac{\ln \overline{x} - \ln \underline{x}}{\overline{x} - \underline{x}}(x - \underline{x}). \tag{3.3.5}$$

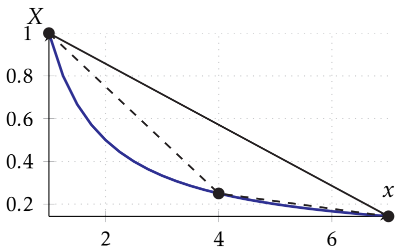
Graphs illustrating how the inverse transformations are approximated by PLFs are given in fig. 3.3 for $[\underline{x}, \overline{x}] = [1, 7]$. To illustrate how the approximations improve as the number of gridpoints used in the PLFs are increased, an additional gridpoint at $x = 4$ is also added. The convex underestimators of the entire function $f(x)$ are illustrated in fig. 3.4.



(a) ET: $X = \ln x$

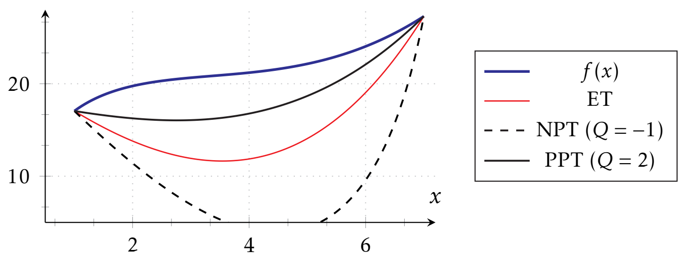


(b) PPT with $Q = 2$: $X = x^{0.5}$

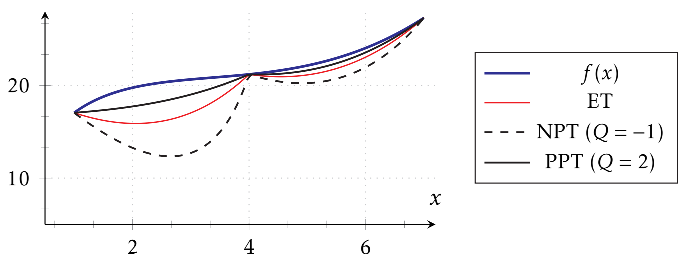


(c) NPT with $Q = -1$: $X = x^{-1}$

Figure 3.3: The approximation of the inverse transformations in ex. 3.6 with the interval endpoints as breakpoints (black, solid) as well as with an additional breakpoint at $x = 4$ (black, dashed).



(a) With only the interval endpoints as gridpoints



(b) With an additional gridpoint at $x = 4$

Figure 3.4: The function $f(x) = 0.05x^3 - 8x + 25x^{0.5}$ and the convex underestimators resulting from the ET, PPT and NPT in ex. 3.6.

3.4 Relationships between the transformations

In this section, some theoretical results regarding the relationship between the ET, NPT and PPT for positive signomial terms are given. It will be shown, that for a general positive signomial term, the ET always gives a tighter convex underestimator than the NPT, and that the PPT gives a tighter convex underestimator than the NPT under certain conditions. Furthermore, it will be proved that neither the ET nor the PPT is better in the whole domain for positive signomial terms of more than one variable. Most of the results have been published earlier in Papers IV and VI.

The first theorem gives results regarding the underestimation properties of the PPT, NPT and ET when applied to a single-variable power function x^p.

Theorem 3.7. Assume that the ET, PPT and NPT, *i.e.*, the transformations

$$x = e^{X_E}, \qquad x = X_P^{Q_P},\ Q_P \geq 1, \qquad \text{and} \qquad x = X_N^{Q_N},\ Q_N < 0,$$

are applied to the power function x^p, $p > 0$, where $x \in \mathbb{R}_+$ or $x \in \mathbb{Z}$ and $x \in [\underline{x}, \overline{x}]$. Then the following is true

$$\left(\hat{X}_P^{Q_P}\right)^p \geq \left(e^{\hat{X}_E}\right)^p \geq \left(\hat{X}_N^{Q_N}\right)^p, \tag{3.4.1}$$

when the inverse transformations

$$X_E = \ln x, \qquad X_P = x^{1/Q_P} \qquad \text{and} \qquad X_N = x^{1/Q_N},$$

have been replaced by the PLFs $\hat{X}_E$, $\hat{X}_P$ and $\hat{X}_N$ respectively. That is, for x^p the PPT always gives a tighter convex underestimator than the ET and the ET a tighter convex underestimator than the NPT.

Proof The PLF of the inverse transformation for the ET is in the interval $[\underline{x}, \overline{x}]$

$$\hat{X}_E = \ln \underline{x} + \frac{\ln \overline{x} - \ln \underline{x}}{\overline{x} - \underline{x}}(x - \underline{x}), \tag{3.4.2}$$

and for the PPT and NPT

$$\hat{X}_P = \underline{x}^{1/Q_P} + \frac{\overline{x}^{1/Q_P} - \underline{x}^{1/Q_P}}{\overline{x} - \underline{x}}(x - \underline{x}) \qquad \text{and} \tag{3.4.3}$$

$$\hat{X}_N = \underline{x}^{1/Q_N} + \frac{\overline{x}^{1/Q_N} - \underline{x}^{1/Q_N}}{\overline{x} - \underline{x}}(x - \underline{x}). \tag{3.4.4}$$

First the case of the PPT versus the ET is proved, *i.e.*, the claim is

$$\left(\hat{X}_P^{Q_P}\right)^p \geq \left(e^{\hat{X}_E}\right)^p. \tag{3.4.5}$$

Since $p > 0$ the following is true

$$\left(\hat{X}_P^{Q_P}\right)^p \geq \left(e^{\hat{X}_E}\right)^p \Longleftrightarrow \hat{X}_P^{Q_P} \geq e^{\hat{X}_E}. \tag{3.4.6}$$

The expressions for the PLFs are now inserted into eq. (3.4.6) resulting in

$$\left(\underline{x}^{1/Q_P} + \frac{\overline{x}^{1/Q_P} - \underline{x}^{1/Q_P}}{\overline{x} - \underline{x}}(x - \underline{x})\right)^{Q_P} \geq \exp\left(\ln \underline{x} + \frac{\ln \overline{x} - \ln \underline{x}}{\overline{x} - \underline{x}}(x - \underline{x})\right). \tag{3.4.7}$$

Since both the left- and right-hand sides of the this inequality are positive, this can be rewritten as

$$Q_P \cdot \ln\left[\underline{x}^{1/Q_P} + \frac{\overline{x}^{1/Q_P} - \underline{x}^{1/Q_P}}{\overline{x} - \underline{x}}(x - \underline{x})\right] \geq \ln \underline{x} + \frac{\ln \overline{x} - \ln \underline{x}}{\overline{x} - \underline{x}}(x - \underline{x}). \tag{3.4.8}$$

By manipulating this expression, the following equivalences are obtained

$$\begin{aligned}
& Q_P \cdot \ln\left[\left(\underline{x}^{1/Q_P} + \frac{\overline{x}^{1/Q_P} - \underline{x}^{1/Q_P}}{\overline{x} - \underline{x}}(x - \underline{x})\right)\Big/ \underline{x}^{1/Q_P}\right] \geq \ln\left(\frac{\overline{x}}{\underline{x}}\right)\frac{x - \underline{x}}{\overline{x} - \underline{x}} \\
\iff\quad & \ln\left[\left(1 + \left(\left(\frac{\overline{x}}{\underline{x}}\right)^{1/Q_P} - 1\right)\frac{x - \underline{x}}{\overline{x} - \underline{x}}\right)^{Q_P}\right] \geq \ln\left[\left(\frac{\overline{x}}{\underline{x}}\right)^{\frac{x - \underline{x}}{\overline{x} - \underline{x}}}\right] \\
\iff\quad & \left[\left(\frac{\overline{x}}{\underline{x}}\right)^{1/Q_P}\frac{x - \underline{x}}{\overline{x} - \underline{x}} + \frac{\overline{x} - x}{\overline{x} - \underline{x}}\right]^{Q_P} \geq \left(\frac{\overline{x}}{\underline{x}}\right)^{\frac{x - \underline{x}}{\overline{x} - \underline{x}}}.
\end{aligned} \tag{3.4.9}$$

Finally, by setting

$$k = \frac{\overline{x}}{\underline{x}}, \qquad \lambda = \frac{x - \underline{x}}{\overline{x} - \underline{x}} \qquad \text{and} \qquad 1 - \lambda = \frac{\overline{x} - x}{\overline{x} - \underline{x}},$$

in the last inequality of eq. (3.4.9) the following equivalent forms can be obtained

$$\begin{aligned}
\left(\lambda k^{1/Q_P} + (1 - \lambda)\right)^{Q_P} \geq k^{\lambda} \quad & \overset{Q_P > 0}{\iff} \quad \lambda k^{1/Q_P} + (1 - \lambda) \geq \left(k^{1/Q_P}\right)^{\lambda} \\
& \iff \quad \lambda z \geq z^{\lambda} + \lambda - 1.
\end{aligned} \tag{3.4.10}$$

Here $z = k^{1/Q_P} \geq 0$, since $k \geq 1$ and $Q > 0$. The last inequality is equivalent to

$$f(z) = \lambda z - z^{\lambda} - \lambda + 1 \geq 0, \tag{3.4.11}$$

which is true since $f'(z) = \lambda - \lambda z^{\lambda - 1}$ only has a root at $z = 1$ and $f''(1) \geq 0$ for $\lambda \in [0,1]$. Thus the proof for the case when

$$\left(\hat{X}_P^{Q_P}\right)^p \geq \left(e^{\hat{X}_E}\right)^p. \tag{3.4.12}$$

is complete.

The proof for the other case, *i.e.*, the claim that

$$\left(e^{\hat{X}_E}\right)^p \geq \left(\hat{X}_N^{Q_N}\right)^p \tag{3.4.13}$$

is analogous to the previous one except X_P and Q_P are replaced with X_N and Q_N, and the inequality signs in eqs. (3.4.6)–(3.4.10) change direction. ■

These results correspond to the convex underestimators obtained for the nonconvex term in ex. 3.6 as shown in fig. 3.4.

Although the previous theorem states that the ET applied to a power function always gives a better approximation than the NPT, the following theorem gives that the latter, in fact, gets arbitrarily close to the ET when $Q \to -\infty$. The same is true for the PPT in the case when $Q \to \infty$, however, in this case, the convex underestimators of the PPT will always lie above that of the ET while the convex underestimators of the NPT always lie under that of the ET.

Theorem 3.8. For the piecewise linear approximations $\hat{X}_P$, $\hat{X}_N$ and $\hat{X}_E$ of the PPT, NPT and ET respectively, the following statement is true

$$\lim_{Q\to\infty} \hat{X}_P^Q = \lim_{Q\to-\infty} \hat{X}_N^Q = e^{\hat{X}_E}, \tag{3.4.14}$$

i.e., a with positive or negative power tends to the exponential transformation as the transformation powers tend to plus and minus infinity respectively.

Proof Since the proof for the cases $\lim_{Q\to\infty} \hat{X}_P^Q = e^{\hat{X}_E}$ and $\lim_{Q\to-\infty} \hat{X}_N^Q = e^{\hat{X}_E}$ are the same when replacing Q by $-Q$, these can be combined into

$$\begin{aligned}\lim_{Q\to\infty} \hat{X}^Q &= \lim_{Q\to\infty}\left[\underline{x}^{1/Q} + \frac{\overline{x}^{1/Q} - \underline{x}^{1/Q}}{\overline{x}-\underline{x}}(x-\underline{x})\right]^Q \\ &= \underline{x}\lim_{Q\to\infty}\left[1 + \frac{x-\underline{x}}{\overline{x}-\underline{x}}\left(\left(\frac{\overline{x}}{\underline{x}}\right)^{1/Q} - 1\right)\right]^Q.\end{aligned} \tag{3.4.15}$$

By introducing the variable r, defined as

$$r = \left(\frac{\overline{x}}{\underline{x}}\right)^{1/Q} - 1 \quad \Longrightarrow \quad Q = \frac{\ln(\overline{x}/\underline{x})}{\ln(r+1)},$$

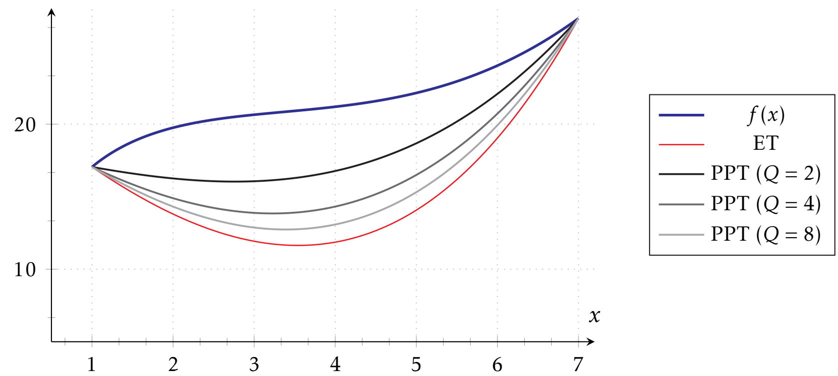
where $r \to 0$ as $Q \to \pm\infty$, eq. (3.4.15) can be rewritten as

$$\underline{x}\cdot\lim_{r\to0}\left[1 + \frac{x-\underline{x}}{\overline{x}-\underline{x}}\,r\right]^{\frac{\ln(\overline{x}/\underline{x})}{\ln(r+1)}} = \underline{x}\cdot\lim_{r\to0}\left(\left(\left[1 + \frac{x-\underline{x}}{\overline{x}-\underline{x}}\,r\right]^{\frac{1}{r}}\right)^{\frac{r}{\ln(r+1)}}\right)^{\ln(\overline{x}/\underline{x})}. \tag{3.4.16}$$

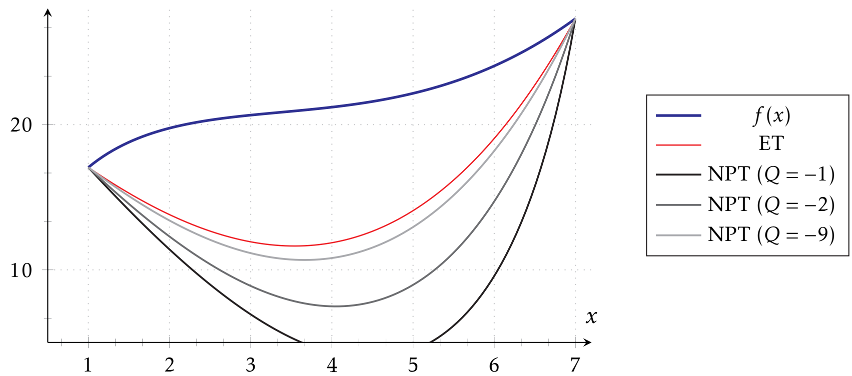
Since $(1+kr)^{1/r} \to e^k$ and $r/\ln(r+1) \to 1$, whenever $r \to 0$ and $|k| < 1$, eq. (3.4.16) is equal to

$$\underline{x}\cdot\exp\left[\frac{x-\underline{x}}{\overline{x}-\underline{x}}\ln\left(\frac{\overline{x}}{\underline{x}}\right)\right] = \exp\left[\ln\underline{x} + \frac{\ln\overline{x}-\ln\underline{x}}{\overline{x}-\underline{x}}(x-\underline{x})\right] = e^{\hat{X}_E}. \tag{3.4.17}$$

So the errors between the single-variable PPT, NPT and ET tend to zero as Q goes to plus or minus infinity. ∎



(a) The ET and the PPT



(b) The ET and the NPT

Figure 3.5: The impact of the transformation power Q for the convex underestimators of the function $f(x) = 0.05x^3 - 8x + 25x^{0.5}$ in ex. 3.6.

Note however, that it is not feasible to choose a too large or too small value on the transformation power Q due to restrictions in computational accuracy when using numerical methods. A method for calculating the smallest possible value for the power Q in PTs with negative powers is given in Lu et al. [2009].

The results in thm. 3.8 can be illustrated by again returning to ex. 3.6. In fig. 3.5 the convex underestimators resulting from the ET, as well as from the PPT and NPT with powers $Q = 2, 4, 8$ and $Q = -1, -4, -9$, respectively. It is clear that the PPT and NPT underestimators tend to the ET underestimator, when the power Q increases and decreases respectively.

Using the results in thm. 3.7, the following theorem, stating that the ET always gives a tighter convex underestimator than the NPT for a general positive signomial term, can be obtained.

Theorem 3.9. The ET applied to the general nonconvex positive signomial term

$$s(\mathbf{x}) = c \prod_{i:p_i<0} x_i^{p_i} \cdot \prod_{i:p_i>0} x_i^{p_i}, \quad c > 0, \quad x_i \in [\underline{x}_i, \overline{x}_i], \tag{3.4.18}$$

always results in a tighter convex underestimator than when applying the NPT.

Proof In both the ET and the NPT, the only variables requiring transformations are those with positive powers p_i. Thus, the convexified and underestimated terms become

$$\hat{s}_N(\mathbf{x}) = \prod_{i:p_i<0} x_i^{p_i} \cdot \prod_{i:p_i>0} \hat{X}_{i,N}^{p_i Q_i} \quad \text{and} \quad \hat{s}_E(\mathbf{x}) = \prod_{i:p_i<0} x_i^{p_i} \cdot \prod_{i:p_i>0} e^{p_i \hat{X}_{i,E}}, \tag{3.4.19}$$

where $\hat{s}_N(\mathbf{x})$ and $\hat{s}_E(\mathbf{x})$ are the transformed terms resulting from the NPT and ET respectively. The claim is then equivalent to the inequality $\hat{s}_E(\mathbf{x}) \geq \hat{s}_N(\mathbf{x})$ holding. Since $x_i > 0$ for all i, the untransformed power functions $x_i^{p_i}$ are cancelled out on both sides of the inequality. Therefore, the ET gives a tighter underestimator than the NPT if

$$\forall i: \ p_i > 0: \quad \left(\hat{X}_{i,N}^{Q_i}\right)^{p_i} \leq \left(e^{\hat{X}_{i,E}}\right)^{p_i}. \tag{3.4.20}$$

The last statement holds according to thm. 3.7, and so the proof is finished. ∎

A corresponding result regarding the relation between the PPT and the NPT can also be deduced under the additional condition that the negative powers used in the power transformations in the PPT must be less or equal to the corresponding ones used in the NPT.

Theorem 3.10. The PPT gives a tighter convex underestimator than the NPT in the whole domain when applied to the general positive nonconvex signomial term

$$s(\mathbf{x}) = c \prod_{i:p_i<0} x_i^{p_i} \cdot \prod_{i:p_i>0} x_i^{p_i}, \quad c > 0, \quad x_i \in [\underline{x}_i, \overline{x}_i], \tag{3.4.21}$$

of more than one variable, as long as the transformation powers $Q_{i,N}$ and $Q_{i,P}$ in the NPT and PPT respectively, fulfill the condition

$$\forall i : p_i > 0, \ i \neq k : \quad Q_{i,P} \leq Q_{i,N}, \tag{3.4.22}$$

where the index k corresponds to the power $p_k Q_k$ remaining positive in the PPT.

Proof The claim is equivalent to

$$\hat{s}_P(\mathbf{x}) \geq \hat{s}_N(\mathbf{x}), \tag{3.4.23}$$

where $\hat{s}_P(\mathbf{x})$ and $\hat{s}_N(\mathbf{x})$ are the convex underestimators of

$$s(\mathbf{x}) = c \prod_{i:p_i<0} x_i^{p_i} \cdot \prod_{i:p_i>0} x_i^{p_i}, \quad c > 0, \quad x_i \in [\underline{x}_i, \overline{x}_i], \tag{3.4.24}$$

obtained when transforming the term using the PPT and NPT respectively. When applying the PPT or the NPT to $s(\mathbf{x})$ only the variables with positive powers need to be transformed. Assuming that the powers $Q_{i,P}$ and $Q_{i,N}$, for i such that $p_i > 0$ are used in the PPT and NPT transformations respectively, the following holds

$$\begin{aligned}
\hat{s}_P(\mathbf{x}) &= \prod_{i:p_i>0} \left(\hat{X}_{i,P}(x_i)\right)^{p_i Q_{i,P}} \cdot \prod_{i:p_i<0} x_i^{p_i} \\
&= \left(\hat{X}_{k,P}(x_k)\right)^{p_k Q_{k,P}} \cdot \prod_{\substack{i:p_i>0 \\ i\neq k}} \left(\hat{X}_{i,P}(x_i)\right)^{p_i Q_{i,P}} \cdot \prod_{i:p_i<0} x_i^{p_i} \\
&\overset{(i)}{\geq} \left(\hat{X}_{k,N}(x_k)\right)^{p_k Q_{k,N}} \cdot \prod_{\substack{i:p_i>0 \\ i\neq k}} \left(\hat{X}_{i,P}(x_i)\right)^{p_i Q_{i,P}} \cdot \prod_{i:p_i<0} x_i^{p_i} \\
&\overset{(ii)}{\geq} \left(\hat{X}_{k,N}(x_k)\right)^{p_k Q_{k,N}} \cdot \prod_{\substack{i:p_i>0 \\ i\neq k}} \left(\hat{X}_{i,N}(x_i)\right)^{p_i Q_{i,N}} \cdot \prod_{i:p_i<0} x_i^{p_i} \\
&= \prod_{i:p_i>0} \left(\hat{X}_{i,N}(x_i)\right)^{p_i Q_{i,N}} \cdot \prod_{i:p_i<0} x_i^{p_i} = \hat{s}_N(\mathbf{x})
\end{aligned} \tag{3.4.25}$$

The inequality in step *(i)* is true since the PPT always gives a tighter underestimator than the NPT according to thm. 3.7. Furthermore, the inequality in step *(ii)* is true since single-variable transformations using the NPT becomes tighter as the power Q increases according to thm. 3.8 and the transformation powers fulfill condition (3.4.22). ∎

When applying the PPT and NPT to the same signomial term, and one or more of the powers $Q_{i,N}$, $i \neq k$ used in the NPT is less than the corresponding $Q_{i,P}$, *i.e.*,

$$\exists i \neq k: \quad Q_{i,N} < Q_{i,P}, \tag{3.4.26}$$

then thm. 3.10 is no longer automatically true. However, the region of the domain where the PPT provides a tighter underestimator than the NPT can be specified as those points x fulfilling the condition

$$\prod_{i:p_i>0} \hat{X}_{i,P}^{p_i Q_{i,P}} \geq \prod_{i:p_i>0} \hat{X}_{i,N}^{p_i Q_{i,N}}. \tag{3.4.27}$$

For such i that $Q_{i,P} = Q_{i,N}$ the corresponding factors in eq. (3.4.27) cancel out each other.

Thus, the ET always gives a tighter underestimator than the NPT and the PPT always gives a tighter underestimator than the NPT if the conditions in thm. 3.10 are fulfilled. One question remains: What is the relation between the ET and the PPT? This is answered in the next theorem, where it will be shown that neither the PPT nor the ET gives a tighter underestimator in the whole domain, instead both transformations have a region where it is tighter than the other.

Theorem 3.11. Neither the PPT nor the ET gives a tighter convex underestimator in the whole domain when applied to the general positive nonconvex signomial term

$$s(\mathbf{x}) = \prod_{i:p_i<0} x_i^{p_i} \cdot \prod_{i:p_i>0} x_i^{p_i}, \quad c > 0, \quad x_i \in [\underline{x}_i, \overline{x}_i], \tag{3.4.28}$$

of more than one variable.

Proof In the case of the ET, single-variable exponential transformations are used on all variables with positive powers and in the case of the PPT, one of the variables (with index k) with positive powers is transformed using a single-variable power transformation with positive power and the rest with single-variable power transformations with negative powers. From thm. 3.7 it is known that

$$\hat{X}_{k,P}^{Q_k p_k} \geq \left(e^{\hat{X}_{k,E}}\right)^{p_k} \quad \text{and} \quad \forall i \neq k: \ \hat{X}_{i,P}^{Q_i p_i} \leq \left(e^{\hat{X}_{i,E}}\right)^{p_i} \tag{3.4.29}$$

when the transformation variables $X_{i,P}$ and $X_{i,E}$ have been replaced by the PLFs $\hat{X}_{i,P}$ and $\hat{X}_{i,E}$. Now take a point $\mathbf{x}^* = (x_1^*, \ldots, x_N^*)$ such that

$$\begin{cases} \underline{x}_i < x_i^* < \overline{x}_i & \text{if } i = k, \\ x_i^* = \underline{x}_i \vee x_i^* = \overline{x}_i & \text{if } i: \ p_i > 0 \wedge i \neq k, \\ \underline{x}_i \leq x_i^* \leq \overline{x}_i & \text{if } i: p_i < 0, \end{cases} \tag{3.4.30}$$

and another point $\mathbf{x}^\# = (x_1^\#, \ldots, x_N^\#)$ such that

$$\begin{cases} x_i^\# = \underline{x}_i \vee x_i^\# = \overline{x}_i & \text{if } i = k, \\ \underline{x}_i < x_i^\# < \overline{x}_i & \text{if } i: \ p_i > 0 \wedge i \neq k, \\ \underline{x}_i \leq x_i^\# \leq \overline{x}_i & \text{if } i: p_i < 0. \end{cases} \tag{3.4.31}$$

Then for the point $\mathbf{x}^*$ the following is true

$$\begin{aligned} \hat{s}_P(\mathbf{x}^*) &= \prod_{i:p_i>0} \left(\hat{X}_{i,P}(\mathbf{x}^*)\right)^{p_i Q_i} \cdot \prod_{i:p_i<0} (\mathbf{x}^*)^{p_i} \\ &\overset{(i)}{=} \left(\hat{X}_{k,P}\left(x_k^*\right)\right)^{p_k Q_k} \cdot \prod_{\substack{i:p_i>0 \\ i\neq k}} \left(e^{\hat{X}_{i,E}(x_i^*)}\right)^{p_i} \cdot \prod_{i:p_i<0} \left(x_i^*\right)^{p_i} \\ &\overset{(ii)}{>} \left(e^{\hat{X}_{k,P}(x_k^*)}\right)^{p_k} \cdot \prod_{\substack{i:p_i>0 \\ i\neq k}} \left(e^{\hat{X}_{i,E}(x_i^*)}\right)^{p_i} \cdot \prod_{i:p_i<0} \left(x_i^*\right)^{p_i} \\ &= \prod_{i:p_i>0} \left(e^{\hat{X}_{i,E}(x_i^*)}\right)^{p_i} \cdot \prod_{i:p_i<0} \left(x_i^*\right)^{p_i} = \hat{s}_E(\mathbf{x}^*). \end{aligned} \tag{3.4.32}$$

Step *(i)* is true since the transformation with negative powers in the PPT and the ET are equal at the breakpoints (in this case the interval endpoints $\underline{x}_i$ or $\overline{x}_i$). Step *(ii)* is true according to thm. 3.7 and the fact that the transformations are equal only at the breakpoints. Thus, in the point $\mathbf{x}^*$ the PPT gives a tighter convex underestimator than the ET. For the point $\mathbf{x}^\#$ the corresponding relation is

$$
\begin{aligned}
\hat{s}_P\left(\mathbf{x}^\#\right) &= \prod_{i:p_i>0} \left(\hat{X}_{i,P}\left(x_i^\#\right)\right)^{p_i Q_i} \cdot \prod_{i:p_i<0} \left(x_i^\#\right)^{p_i} \\
&\overset{(i)}{=} \left(e^{\hat{X}_{k,E}\left(x_k^\#\right)}\right)^{p_k} \cdot \prod_{\substack{i:p_i>0 \\ i\neq k}} \left(\hat{X}_{i,P}\left(x_i^\#\right)\right)^{p_i Q_i} \cdot \prod_{i:p_i<0} \left(x_i^\#\right)^{p_i} \\
&\overset{(ii)}{<} \left(e^{\hat{X}_{k,P}\left(x_k^\#\right)}\right)^{p_k} \cdot \prod_{\substack{i:p_i>0 \\ i\neq k}} \left(e^{\hat{X}_{i,E}\left(x_i^\#\right)}\right)^{p_i} \cdot \prod_{i:p_i<0} \left(x_i^\#\right)^{p_i} \\
&= \prod_{i:p_i>0} \left(e^{\hat{X}_{i,E}\left(x_i^\#\right)}\right)^{p_i} \cdot \prod_{i:p_i<0} \left(x_i^\#\right)^{p_i} = \hat{s}_E\left(\mathbf{x}^\#\right).
\end{aligned} \tag{3.4.33}
$$

Step *(i)* is true since the PPT and ET are equal at the breakpoints (here $\underline{x}_k$ or $\overline{x}_k$). Step *(ii)* is true according to thm. 3.7 and since the transformations are equal only at the breakpoints.

Thus, since $\hat{s}_P(\mathbf{x}^*) > \hat{s}_E(\mathbf{x}^*)$ and $\hat{s}_P(\mathbf{x}^\#) < \hat{s}_E(\mathbf{x}^\#)$ for the different points $\mathbf{x}^*$ and $\mathbf{x}^\#$, neither the PPT nor the ET gives a tighter convex underestimator of the nonconvex function in the whole domain. ∎

Since the convex underestimators resulting from applying the PPT and the ET are continuous, and neither gives a tighter underestimator in the whole domain, there must exist parts of the domain where they are equal. These regions are where the following expression is true

$$
\prod_{i:p_i>0} \hat{X}_{i,P}^{p_i Q_i} = \prod_{i:p_i>0} e^{p_i \hat{X}_{i,E}}, \tag{3.4.34}
$$

i.e., the PPT gives a tighter underestimator than the ET in the region where the following condition is fulfilled

$$
\prod_{i:p_i>0} \hat{X}_{i,P}^{p_i Q_i} > \prod_{i:p_i>0} e^{p_i \hat{X}_{i,E}}. \tag{3.4.35}
$$

Note that thms. 3.7–3.11 also hold when more gridpoints are included in the PLFs as long as the same points are added to PLF-approximations $\hat{X}_{i,P}$ and $\hat{X}_{i,E}$ of the inverse transformations of the corresponding variables x_i.

3.5 Underestimation errors

In this section, the errors caused by the underestimation step of the transformation technique for signomial functions are examined.

The underestimation error $\Delta T(x)$ in the interval $[\underline{x}, \overline{x}]$, when using one of the single-variable exponential or power transformations given in Section 3.2, is simply the difference between the exact inverse transformation $X = T^{-1}(x)$ and the PLF $\hat{X}$ underestimating the approximation, *i.e.*, a straight line connecting the endpoints. Thus, the error is

$$\Delta T(x) = T^{-1}(x) - \left(T^{-1}(\underline{x}) + \frac{T^{-1}(\overline{x}) - T^{-1}(\underline{x})}{\overline{x} - \underline{x}} (x - \underline{x}) \right). \tag{3.5.1}$$

By inserting the transformations $T_E(X) = e^X$ and $T_P(X) = X^Q$ into the expression for the error, explicit functions for the errors can be obtained.

Theorem 3.12. The underestimation error in the point $x \in [\underline{x}, \overline{x}]$ when replacing the inverse transformation with a PLF-approximation is for the single-variable ET

$$\Delta T_E(x) = \ln\left(\frac{x}{\underline{x}}\right) + \frac{x - \underline{x}}{\overline{x} - \underline{x}} \cdot \ln\left(\frac{\overline{x}}{\underline{x}}\right) \tag{3.5.2}$$

and for any of the single-variable PTs

$$\Delta T_P(x) = x^{1/Q} - \left(\underline{x}^{1/Q} + \frac{\overline{x}^{1/Q} - \underline{x}^{1/Q}}{\overline{x} - \underline{x}} (x - \underline{x}) \right). \tag{3.5.3}$$

Using the previous theorem, the point where the underestimation deviates the most from the inverse transformation, *i.e.*, the largest error occurs, can be found by differentiating $\Delta T(x)$ with respect to x. The largest error is then found in the point x which fulfills the following equation

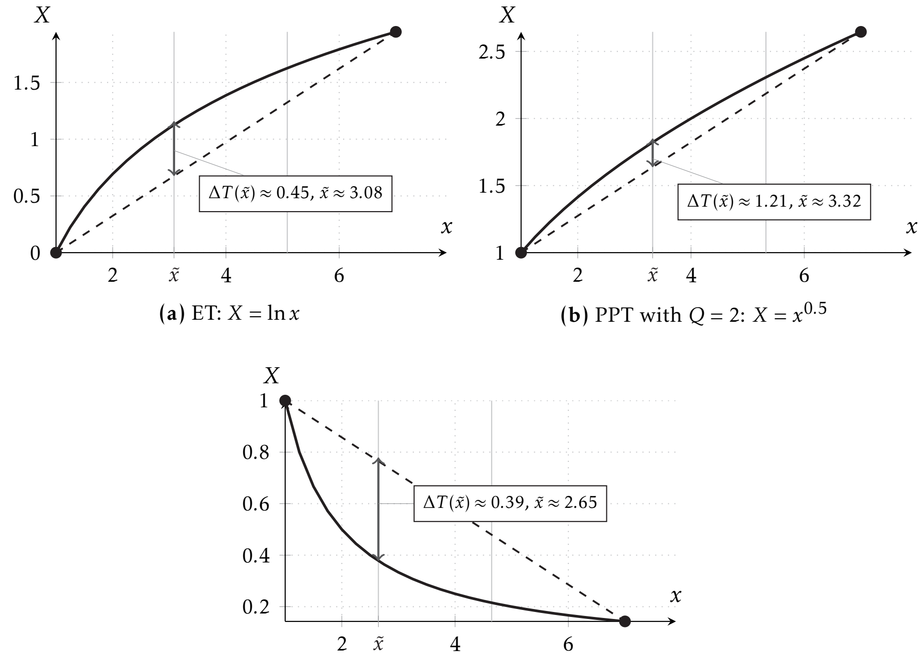
$$\frac{\mathrm{d}}{\mathrm{d}x} T^{-1}(x) = \frac{T^{-1}(\overline{x}) - T^{-1}(\underline{x})}{\overline{x} - \underline{x}}. \tag{3.5.4}$$

By replacing the function T^{-1} with the inverse transformation for the different transformations, thm. 3.13 is obtained.

Theorem 3.13. The largest underestimation errors in the interval $[\underline{x}, \overline{x}]$ when replacing the inverse transformation with the PLF-approximation occurs in the point $\tilde{x}$ given by the expressions

$$\tilde{x} = \frac{\overline{x} - \underline{x}}{\ln(\overline{x}/\underline{x})} \qquad \text{and} \qquad \tilde{x} = \left(Q \cdot \frac{\overline{x}^{1/Q} - \underline{x}^{1/Q}}{\overline{x} - \underline{x}} \right)^{\frac{Q}{1-Q}} \tag{3.5.5}$$

for the single-variable ET and any of the single-variable PTs respectively.



(**a**) ET: $X = \ln x$

(**b**) PPT with $Q = 2$: $X = x^{0.5}$

(**c**) NPT with $Q = -1$: $X = x^{-1}$

Figure 3.6: The maximal errors when approximating the inverse transformations of the ET, PPT and NPT with PLFs.

Note that, when the variable being transformed is discrete, *i.e.*, $x \in \mathbb{Z}_+$, the maximal error is the nearest integer to $\tilde{x}$ calculated in thm. 3.13. The largest errors for an ET, as well as a PPT and a NPT transformation is indicated in fig. 3.6.

According to thm. 3.8, the absolute difference between the underestimation error of the single-variable ET and either of the single-variable PPT or NPT tends to zero as $Q \to \infty$ or $Q \to -\infty$ respectively. Thus the following statement is true regarding the underestimation error of these underestimators

$$\lim_{Q\to\pm\infty} \left(Q \cdot \frac{\overline{x}^{1/Q} - \underline{x}^{1/Q}}{\overline{x} - \underline{x}} \right)^{\frac{Q}{1-Q}} = \frac{\overline{x} - \underline{x}}{\ln(\overline{x}/\underline{x})}. \tag{3.5.6}$$

This is the maximal error for the PPT and the minimal error for the NPT, when applied to the power function x^p, according to thm. 3.7.

3.6 Other convex underestimators

In this section, some other types of convex underestimators which can be used for obtaining convex underestimators for signomial functions are presented. An overview of different types of underestimation methods in global optimization can be found in Liberti and Maculan [2006], where also some automatic reformulation methods for nonconvex problems are presented.

The first underestimator is the αBB underestimator from, *e.g.*, Adjiman et al. [1998] and Floudas [1999]. Although newer versions of the underestimator have been published, see *e.g.*, Akrotirianakis and Floudas [2004], the original version is presented here.

The αBB underestimator can be used as a convex underestimator for any twice-differentiable function on a convex set, and is thus not limited to (or especially designed for) signomial functions. The underestimator is based on the fact that it is possible to find a convex function, which is "convex enough" so that the sum of this function and the original nonconvex function is convex.

Theorem 3.14 (The αBB underestimator). The function

$$L(\mathbf{x}) = f(\mathbf{x}) + \sum_i \alpha(\underline{x}_i - x_i)(\overline{x}_i - x_i) \tag{3.6.1}$$

is a convex underestimator of the function $f(\mathbf{x}) \in C^2$ on the interval $[\underline{x}_i, \overline{x}_i]$, $x_i \in \mathbb{R}$, if and only if, the parameters α fulfill

$$\alpha \geq \max\left\{0,\ -\frac{1}{2}\min_i \lambda_i\right\}, \tag{3.6.2}$$

where the λ_i's are the eigenvalues of $H(\mathbf{x})$, the Hessian matrix of the function $f(\mathbf{x})$ in the region $[\underline{x}_1, \overline{x}_1] \times \cdots \times [\underline{x}_I, \overline{x}_I]$.

It is also possible to use individual values of the parameters α_i for each variable x_i. The Hessian matrix of eq. (3.6.1) is then given by

$$H(\mathbf{x}) + 2\,\mathrm{diag}(\alpha_i), \tag{3.6.3}$$

where $H(\mathbf{x})$ is the Hessian matrix of the function f. If the values of α_i are selected such that the matrix in eq. (3.6.3) is positive semidefinite for all $x_i \in [\underline{x}_i, \overline{x}_i]$, then $L(\mathbf{x})$ (with individual α_i-values) is a valid convex underestimator of the function f in the region $[\underline{x}_1, \overline{x}_1] \times \cdots \times [\underline{x}_I, \overline{x}_I]$.

The problem is then to find a large enough value of the α-parameters. Some different techniques can be found in Adjiman et al. [1998]. In this paper the *scaled Gerschgorin* method, which uses interval analysis, has been used. An implementation of this method was programmed in *Wolfram Mathematica 7.0*, and this implementation was used to calculate the α-values used in Section 3.7.

By dividing the domain in subregions, different α values can be calculated in the different regions, resulting in tighter convex underestimators.

Another convex underestimator for positive signomial terms is the method proposed in Li et al. [2007]. Here power transformations are applied to the variables having positive powers.

Theorem 3.15. The signomial function $f(\mathbf{x}) = c \cdot x_1^{p_1} x_2^{p_2} \ldots x_I^{p_I}$, $c > 0$, $x_i \in [\underline{x}_i, \overline{x}_i]$, $\underline{x}_i > 0$ can be underestimated by the function

$$f(\mathbf{x}, \mathbf{X}) = c \prod_{i:p_i<0} x_i^{p_i} \cdot \prod_{i:p_i>0} X_i^{-p_i}, \tag{3.6.4}$$

where the following conditions are included for all indices i such that $p_i > 0$

$$\frac{x_i}{\overline{x}_i} + \underline{x}_i X_i - \frac{\underline{x}_i}{\overline{x}_i} \leq 1. \tag{3.6.5}$$

This technique actually corresponds to the NPT with the transformation power $Q = -1$, since the conditions in eq. (3.6.5) correspond to a PLF in one step overestimating the inverse transformation $X = x^{-1}$, which can also be written as

$$X(x) \leq \hat{X}(x), \quad \hat{X}(x) = \underline{x}^{1/Q} + \frac{\overline{x}^{1/Q} - \underline{x}^{1/Q}}{\overline{x} - \underline{x}}(x - \underline{x}), \quad \text{where} \quad Q = -1. \tag{3.6.6}$$

3.7 Numerical comparisons of convex underestimators

In this section, some comparisons of the convex underestimators presented previously in this chapter are performed through the means of numerical examples. The comparisons are, to a large extent, the same as in Paper VI.

It is in often difficult to say that a certain type of convex underestimator is better or worse than another, since one underestimator can be better in certain parts of the feasible region and the other in other parts. In the one- and two-dimensional case, it is, of course, possible to do a visual comparison, but this is difficult in more general cases. One possibility is to use the lower bound of the convex underestimator as an error metric.

Definition 3.16. For the function f the lower bound of a convex underestimator $\hat{f}$ is given by the MINLP problem

$$\begin{aligned} &\text{minimize} && \hat{f}(\mathbf{x}), \\ &\text{subject to} && \underline{\mathbf{x}} \leq \mathbf{x} \leq \overline{\mathbf{x}}. \end{aligned} \tag{3.7.1}$$

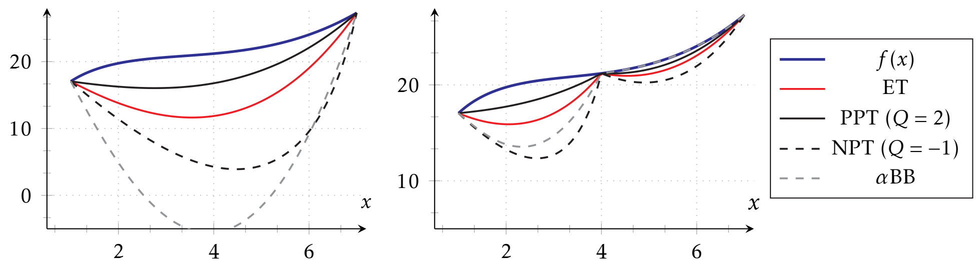


Figure 3.7: The convex underestimators for the function $f(x) = 0.05x^3 - 8x + 25x^{0.5}$ in ex. 3.17.

3.7.1 Univariate function

To compare the NPT, PPT and ET with the αBB underestimator for a function of one variable, ex. 3.6 is revisited.

Example 3.17. The nonconvex function $f(x) = 0.05x^3 - 8x + 25x^{0.5}$, $0 < \underline{x} \leq x \leq \overline{x}$, consists of three signomial terms. The first two terms are convex and only the term $25x^{0.5}$ is nonconvex. The expressions for the convex underestimators resulting from applying the NPT, PPT or ET was given in ex. 3.6. The αBB underestimator is, for example, given on the interval $[1,7]$ as

$$L^{[1,7]}(x) = 0.05x^3 - 8x + 25x^{0.5} + 2.975(1-x)(7-x), \tag{3.7.2}$$

i.e., the value of the parameter α in thm. 3.14 is 2.975. If the domain is partitioned into two intervals $[1,4]$ and $[4,7]$ the αBB underestimators can be given as

$$L^{[1,4]}(x) = 0.05x^3 - 8x + 25x^{0.5} + 2.975(1-x)(4-x), \tag{3.7.3}$$

$$L^{[4,7]}(x) = 0.05x^3 - 8x + 25x^{0.5}. \tag{3.7.4}$$

Since f is convex on the interval $[4,7]$, the convex underestimator is equal to the function itself, *i.e.*, the α-value is equal to zero. The convex underestimators given in this example are illustrated in fig. 3.7

The results from ex. 3.6 show that in the first case, *i.e.*, when no additional breakpoint is added, the PPT gives the tightest convex underestimator. A direct consequence of thm. 3.8 is that the PPT is a tighter underestimator than both the ET and the NPT. In this special case, the PPT gives the convex envelope of the nonconvex term $x^{0.5}$. Note however, that this is not the same as the convex envelope of the whole function. In the second case, when an additional breakpoint at $x = 4$ is added, the PPT is still the tightest in the interval $[1,4]$; in the interval $[4,7]$, however, the αBB underestimator actually coincides with the function f itself, since it is already convex on this interval.

3.7.2 Bivariate functions

In this section convex underestimations of functions of two variables are compared. Positive signomial terms are considered in the first example and a negative signomial term in the second.

Example 3.18. The PPT, ET, as well as the αBB underestimator are now applied to the positive signomial functions *(a)* $f_1(x_1,x_2) = x_1x_2$, *(b)* $f_2(x_1,x_2) = x_1^{0.95}x_2^{0.9}$, and *(c)* $f_3(x_1,x_2) = x_1^{0.6}x_2^{0.5}$, where $x_1,x_2 \in [1,7]$. The NPT is not included in the comparison, since it is known from thms. 3.9 and 3.10 that the ET and PPT gives tighter underestimations.

(a) Piecewise convex underestimators of f_1 are given by the functions

$$\hat{f}_{1,P}(\hat{X}_{1,P},\hat{X}_{2,P}) = \hat{X}_{1,P}^{Q_1}\hat{X}_{2,P}^{Q_2} \quad \text{and} \quad \hat{f}_{1,E}(\hat{X}_{1,E},\hat{X}_{2,E}) = e^{\hat{X}_{1,E}+\hat{X}_{2,E}}, \tag{3.7.5}$$

when using the PPT and ET respectively. Here $\hat{X}_{1,P}$, $\hat{X}_{2,P}$ and $\hat{X}_{1,E}$, $\hat{X}_{2,E}$ are the piecewise linear approximations of the inverse functions $X_{1,P} = x_1^{1/Q_1}$, $X_{2,P} = x_2^{1/Q_2}$ and $X_{1,E} = \ln x_1$, $X_{2,E} = \ln x_2$. The transformation conditions in the PPT are that $Q_1 + Q_2 \geq 1$, as well as that one of the powers Q_1 and Q_2 must be positive, and the other negative. So, for example $Q_1 = 2$ and $Q_2 = -1$ can be chosen. A corresponding αBB underestimator for the function f_1 is

$$L_1(x_1,x_2) = x_1x_2 + 0.5(1-x_1)(7-x_1) + 0.5(1-x_2)(7-x_2). \tag{3.7.6}$$

The convex envelope, *i.e.*, the McCormick underestimator, for the bilinear term is according to ex. 2.4 given by

$$\hat{f}_{1,M}(x_1,x_2) = \max\{x_1 + x_2 - 1, 7x_1 + 7x_2 - 49\}. \tag{3.7.7}$$

(b) Piecewise convex underestimators of f_2 are given by the functions

$$\hat{f}_{2,P}(\hat{X}_{1,P},\hat{X}_{2,P}) = \hat{X}_{1,P}^{0.95Q_1}\hat{X}_{2,P}^{0.9Q_2} \quad \text{and} \quad \hat{f}_{2,E}(\hat{X}_{1,E},\hat{X}_{2,E}) = e^{0.95\hat{X}_{1,E}+0.9\hat{X}_{2,E}}, \tag{3.7.8}$$

when using the PPT and ET respectively. Here $\hat{X}_{1,P}$, $\hat{X}_{2,P}$ and $\hat{X}_{1,E}$, $\hat{X}_{2,E}$ are the PLF-approximations of the inverse functions $X_{1,P} = x_1^{1/Q_1}$, $X_{2,P} = x_2^{1/Q_2}$, $X_{1,E} = \ln x_1$ and $X_{2,E} = \ln x_2$. The powers in the PPT are chosen to be $Q_1 = 19/9 \approx 2.11$ and $Q_2 = -1$. An αBB underestimator for the function f_2 is, for example,

$$L_2(x_1,x_2) = x_1^{0.95}x_2^{0.9} + 0.564(1-x_1)(7-x_1) + 0.713(1-x_2)(7-x_2). \tag{3.7.9}$$

(c) Similarly, the piecewise convex underestimators

$$\hat{f}_{3,P}(\hat{X}_{1,P},\hat{X}_{2,P}) = \hat{X}_{1,P}^{0.6Q_1}\hat{X}_{2,P}^{0.5Q_2} \quad \text{and} \quad \hat{f}_{3,E}(\hat{X}_{1,E},\hat{X}_{2,E}) = e^{0.6\hat{X}_{1,E}+0.5\hat{X}_{2,E}} \tag{3.7.10}$$

of f_3 are obtained when using the PPT and ET respectively. The powers in the PPT are here chosen to be $Q_1 = 2.5$ and $Q_2 = -1$. An αBB underestimator for the function f_3 is

$$L_3(x_1, x_2) = x_1^{0.6} x_2^{0.5} + 0.467(1 - x_1)(7 - x_1) + 0.552(1 - x_2)(7 - x_2). \tag{3.7.11}$$

An illustration in 3D of the convex underestimator is provided in fig. 3.11.

The convex envelope for the functions in *(b)* and *(c)* are not known, so they cannot be included in the comparison. Plots of the errors of the different underestimators in this example are given in fig. 3.8 when using only the interval endpoints as gridpoints. In fig. 3.9 it is illustrated how the underestimators become tighter as more breakpoints are added to the PLFs for the ET and PPT, and partitioning the domain for the αBB and McCormick underestimator. Note that partitioning the domain leads to different αBB and McCormick underestimators in each of the four subregions.

It is difficult to draw any conclusions from the results of ex. 3.18 presented in figs. 3.8 and 3.9. Of course, the convex envelope gives the tightest convex underestimator, but the others were not that much worse (at least not in the sense of the largest underestimation error), and unfortunately, the convex envelopes are known only for certain signomial terms. The PPT did not give very good results when underestimating f_1 and f_2, because the negative PT used was not very good. A larger positive value of Q_1 and a larger negative value of Q_2 would have given better results. However, according to thms. 3.11, the ET will always be better in some region than the PPT regardless of the powers Q chosen. This region is shown in fig. 3.10.

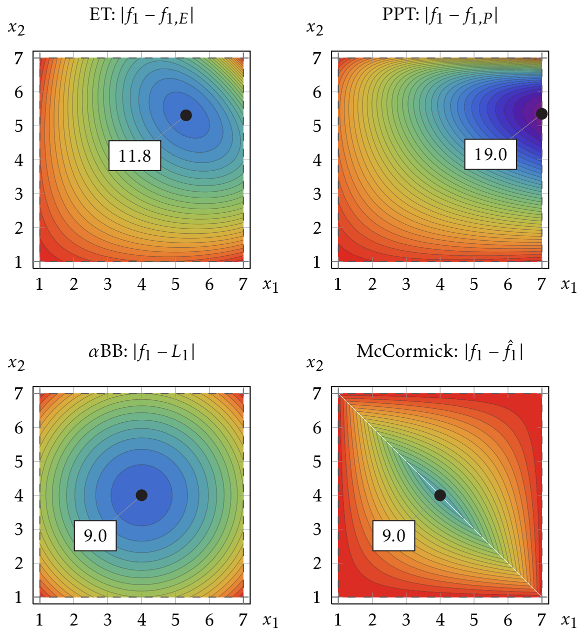
Example 3.19. The PT for negative signomial terms and the αBB underestimator are now applied to the signomial function $f(x_1, x_2) = -x_1^{0.7} x_2$, where $x_1, x_2 \in [1, 9]$. In the PTs, the transformation powers Q can be chosen in many different ways; here two different pairs of Q-values are given, the first is $Q_1 = 0.5/0.7$, $Q_2 = 0.5$ and the second is $Q_1 = 1$, $Q_2 = 0.3$. To illustrate how the approximations improve as the domain is partitioned or additional gridpoints are added to the PLFs, three cases are considered *(a)* with only the interval endpoints as grid points, *(b)* with extra grid points at $x_1 = 5$ and $x_2 = 5$, and *(c)* with extra grid points at $x_2 = 3, 5, 7$ when $Q_1 = 1$ and $Q_2 = 0.3$.

(a) A piecewise convex underestimator of the function f is given by the function

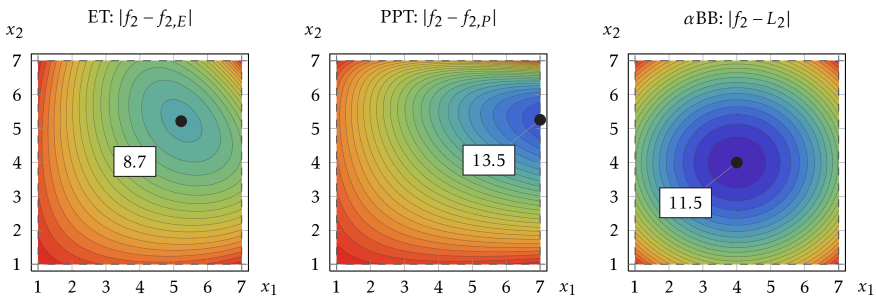
$$\hat{f}_{1,P}(\hat{X}_{1,1}, \hat{X}_{2,1}) = -\hat{X}_{1,1}^{0.7 \cdot 0.5/0.7} \hat{X}_{2,1}^{1 \cdot 0.5} = -\hat{X}_{1,1}^{0.5} \hat{X}_{2,1}^{0.5}, \tag{3.7.12}$$

when using the PT with the powers $Q_1 = 0.5/0.7$, $Q_2 = 0.5$, and the function

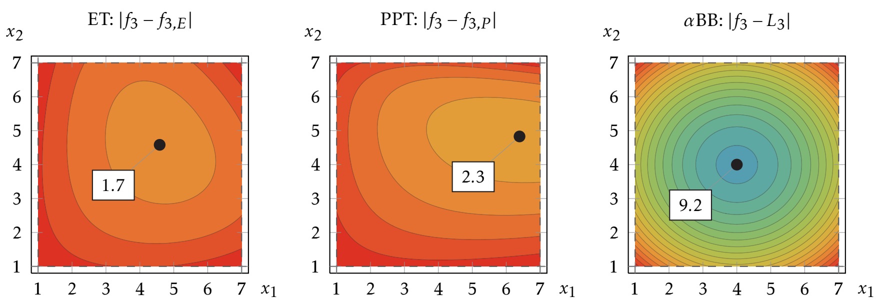
$$\hat{f}_{2,P}(x, \hat{X}_{2,2}) = -x^{0.7} \hat{X}_{2,2}^{1 \cdot 0.3} = -x^{0.7} \hat{X}_{2,2}^{0.3}, \tag{3.7.13}$$



(a) $f_1(x_1, x_2) = x_1 x_2$

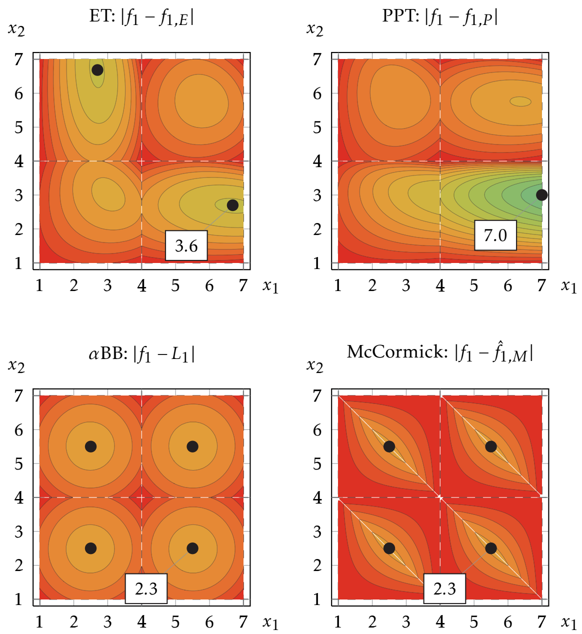


(b) $f_2(x_1, x_2) = x_1^{0.95} x_2^{0.9}$

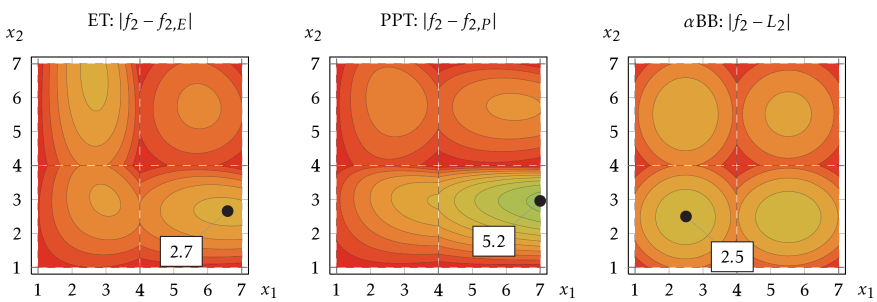


(c) $f_3(x_1, x_2) = x_1^{0.6} x_2^{0.5}$

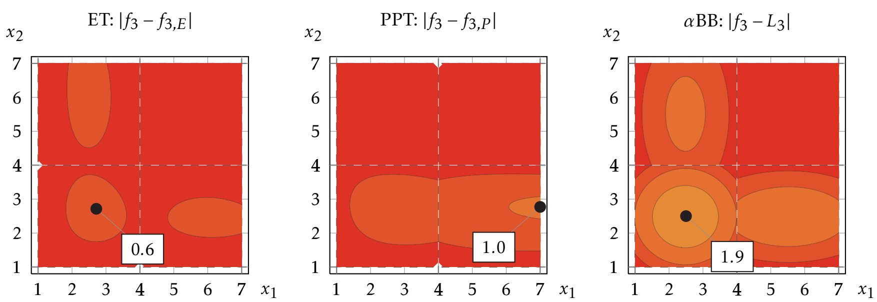
Figure 3.8: The underestimation errors of the convex underestimators in ex. 3.18. The contour distance is 0.5. The maximal error is indicated in each figure.



(a) $f_1(x_1, x_2) = x_1 x_2$

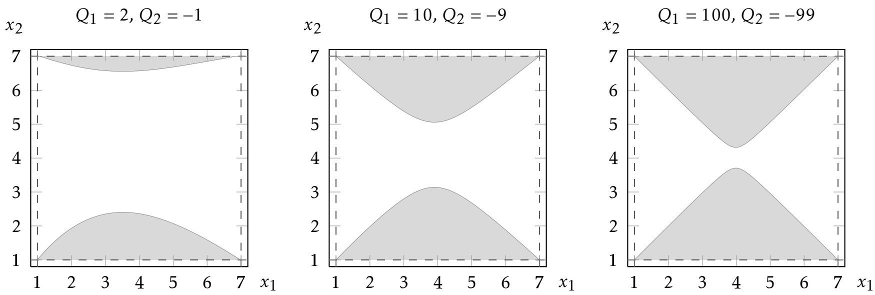


(b) $f_2(x_1, x_2) = x_1^{0.95} x_2^{0.9}$

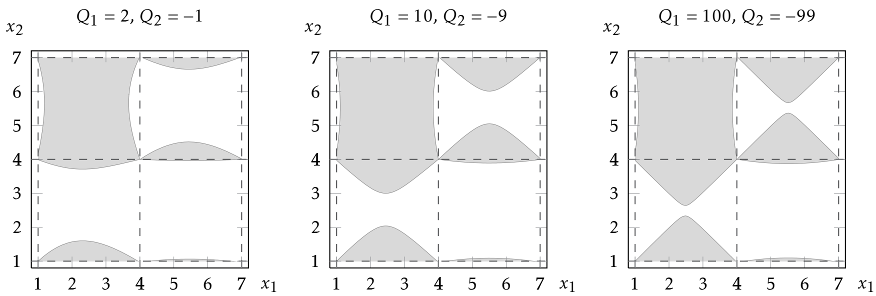


(c) $f_3(x_1, x_2) = x_1^{0.6} x_2^{0.5}$

Figure 3.9: The underestimation errors of the underestimators in ex. 3.18 after adding the grid-points $x_1 = x_2 = 4$. The contour distance is 0.5. The maximal errors are indicated in the figures.

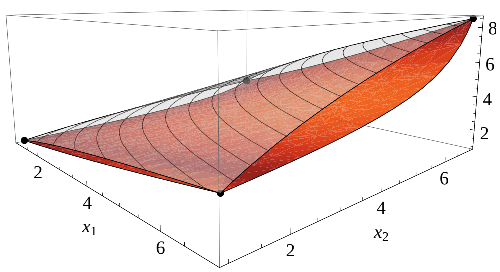


(a) Only the interval endpoints as gridpoints

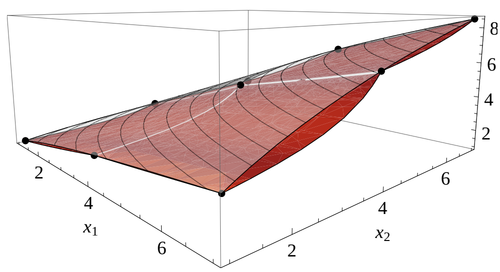


(b) With additional gridpoints at $x_1 = x_2 = 4$

Figure 3.10: Comparison of the regions of the domain where the PPT is tighter than the ET, indicated by the shaded area, when underestimating the function $f(x_1, x_2) = x_1 x_2$ in ex. 3.18.



(a) Only the interval endpoints as gridpoints



(b) With additional gridpoints at $x_1 = x_2 = 4$

Figure 3.11: The nonconvex function $f_3(x_1, x_2) = x_1^{0.6} x_2^{0.5}$ in ex. 3.18 underestimated by the PPT. The black points are the gridpoints for the PLFs, *i.e.*, where the underestimation is exact.

when using the PT with the powers $Q_1 = 1$ and $Q_2 = 0.3$. Since the Q_1-value is equal to one, the variable x is not transformed in this case. A corresponding αBB underestimator for the function f is

$$L_1(x_1, x_2) = -x_1^{0.7} x_2 + 0.344(1-x_1)(9-x_1) + 0.350(1-x_2)(9-x_2). \tag{3.7.14}$$

(b) As in *(a)*, piecewise convex underestimators for the function f are given by

$$\hat{f}_{1,P}(\hat{X}_{1,1}, \hat{X}_{2,1}) = -\hat{X}_{1,1}^{0.7\cdot 0.5/0.7} \hat{X}_{2,1}^{1\cdot 0.5} = -\hat{X}_{1,1}^{0.5} \hat{X}_{2,1}^{0.5} \quad \text{and} \tag{3.7.15}$$

$$\hat{f}_{2,P}(x_1, \hat{X}_{2,2}) = -x_1^{0.7} \hat{X}_{2,2}^{1\cdot 0.3} = -x_1^{0.7} \hat{X}_{2,2}^{0.3}, \tag{3.7.16}$$

for the value-pairs $Q_1 = 0.5/0.7$, $Q_2 = 0.5$, and $Q_1 = 1$, $Q_2 = 0.3$, respectively. However, additional breakpoints are added to the piecewise linear approximations $\hat{X}_{1,1}$, $\hat{X}_{2,1}$ and $\hat{X}_{2,2}$. The αBB underestimators in the four partitions of the domain for the function f are

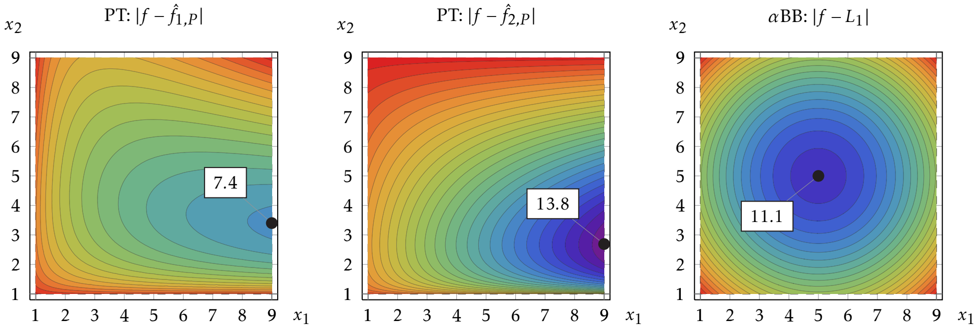
$$\begin{aligned}
L_2^{[1,5]\times[1,5]}(x_1, x_2) &= -x_1^{0.7} x_2 + 0.34(1-x_1)(5-x_1) + 0.35(1-x_2)(5-x_2), \\
L_2^{[1,5]\times[5,9]}(x_1, x_2) &= -x_1^{0.7} x_2 + 0.29(1-x_1)(5-x_1) + 0.35(5-x_2)(9-x_2), \\
L_2^{[5,9]\times[1,5]}(x_1, x_2) &= -x_1^{0.7} x_2 + 0.21(5-x_1)(9-x_1) + 0.22(1-x_2)(5-x_2), \\
L_2^{[5,9]\times[5,9]}(x_1, x_2) &= -x_1^{0.7} x_2 + 0.19(5-x_1)(9-x_1) + 0.22(5-x_2)(9-x_2).
\end{aligned} \tag{3.7.17}$$

(c) In the case when $Q_1 = 1$ and $Q_2 = 0.3$, only the variable x_2 is transformed and a piecewise convex underestimator for the function f is given by

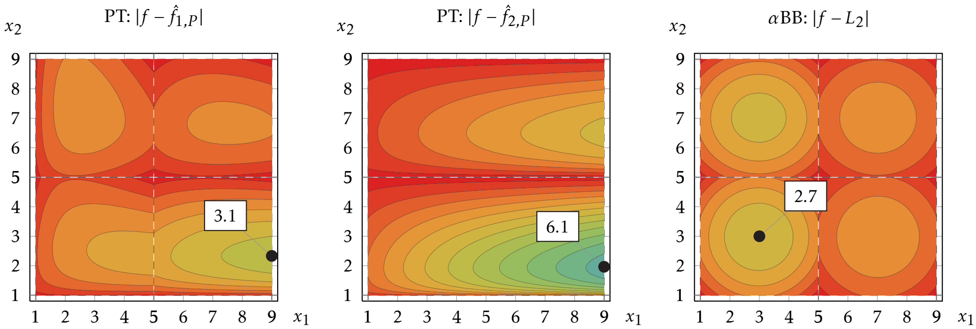
$$f_{2,P}(x_1, \hat{X}_{2,2}) = -x_1^{0.7} \hat{X}_{2,2}^{1\cdot 0.3} = -x_1^{0.7} \hat{X}_{2,2}^{0.3}. \tag{3.7.18}$$

Since gridpoints can only be added to the PLF $\hat{X}_{2,2}$, the domain is split into only two partitions when adding the point $x_2 = 5$ as in case *(b)*. For a comparable combinatorial complexity as when transforming both variables, two additional gridpoints can be added, in this case $x_2 = 3$ and $x_2 = 5$ are chosen. The result is a tighter convex underestimator than in any of the previous cases.

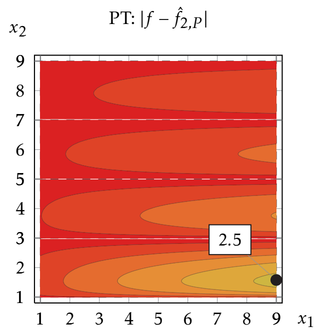
This example shows the advantages of transforming as few variables as possible, as it is possible to include more breakpoints in the transformed variables and still have the same combinatorial complexity. This fact is utilized in the method for determining an optimized set of transformations given in Chapter 4.



(a) Only the interval endpoints as gridpoints



(b) Additional gridpoints at $x_1 = 5$ and $x_2 = 5$



(c) Additional gridpoints at $x_2 = \{3, 5, 7\}$

Figure 3.12: The underestimation errors of the convex underestimators in ex. 3.19. The contour distance is 0.5. The maximal error is indicated in each figure.

3.7.3 Multivariate functions

In this section, convex underestimators for some more difficult signomial functions from Li et al. [2007] are considered. The lower bound in def. 3.16 is used to compare the underestimators. Since the domains of the variables are relatively large in these examples, the αBB underestimator did not give a reasonable underestimator without partitioning the domains of the variables, and it is therefore not included in the comparisons. Instead, the solver Branch and reduce optimization navigator (BARON) is included. BARON is a global MINLP solver which utilizes several different techniques for solving certain classes of nonconvex problems to global optimality, see *e.g.*, Tawarmalani and Sahinidis [2004] and Sahinidis and Tawarmalani [2005].

Example 3.20. In the signomial function

$$f(\mathbf{x}) = x_1x_2x_3x_4x_5 - x_2^{0.5}x_4^{0.5} - 3x_1 - x_5, \qquad 1 \leq x_1, x_2, x_3, x_4, x_5 \leq 100, \tag{3.7.19}$$

only the first term is nonconvex and must be transformed. Transforming it using the ET gives the following convex underestimator

$$\hat{f}_E(\mathbf{x}, \hat{\mathbf{X}}_\mathbf{E}) = e^{\hat{X}_{1,E}+\hat{X}_{2,E}+\hat{X}_{3,E}+\hat{X}_{4,E}+\hat{X}_{5,E}} - x_2^{0.5}x_4^{0.5} - 3x_1 - x_5, \tag{3.7.20}$$

where $\hat{X}_{1,E}, \ldots, \hat{X}_{5,E}$ are the piecewise linear approximations of the inverse transformations $X_{i,E} = \ln x_i$, *i.e.*,

$$\hat{X}_{i,E} = \ln \underline{x}_i + \frac{\ln \overline{x}_i - \ln \underline{x}_i}{\overline{x}_i - \underline{x}_i}(x - \underline{x}_i). \tag{3.7.21}$$

Transforming $f(\mathbf{x})$ using the PPT or NPT gives the convex underestimator

$$\hat{f}_P(\mathbf{x}, \hat{\mathbf{X}}_\mathbf{P}) = \hat{X}_{1,P}^{Q_1}\hat{X}_{2,P}^{Q_2}\hat{X}_{3,P}^{Q_3}\hat{X}_{4,P}^{Q_4}\hat{X}_{5,P}^{Q_5} - x_2^{0.5}x_4^{0.5} - 3x_1 - x_5. \tag{3.7.22}$$

For the PPT the transformation powers are chosen so that $Q_1 = 5$ and $Q_2 = \ldots = Q_5 = -1$, and for the NPT so that $Q_1 = \ldots = Q_5 = -10$. The function $\hat{f}_P(\mathbf{x}, \hat{\mathbf{X}}_\mathbf{P})$ is then a convex underestimator of $f(\mathbf{x})$ if $\hat{X}_{1,P}, \ldots, \hat{X}_{5,P}$ are the piecewise linear approximations of the inverse transformations $X_{i,P} = x_i^{1/Q}$, *i.e.*,

$$\hat{X}_{i,P} = \underline{x}_i^{1/Q_i} + \frac{\overline{x}_i^{1/Q_i} - \underline{x}_i^{1/Q_i}}{\overline{x}_i - \underline{x}_i}(x - \underline{x}_i). \tag{3.7.23}$$

The lower bounds of the different convex underestimators are provided in table 3.1, and the impact of the power Q in the NPT is shown in fig 3.13a.

Table 3.1: Comparison of the LB of the convex underestimation techniques in ex. 3.20

Technique	ET	NPT[a]	PPT	BARON[b]	Li et al[c]	Opt. sol.[d]
Lower bound	-209.22	-215.73	-202.03	-224.44	-317.08	-202.00

[a] With the transformation powers $Q_i = -10$
[b] Obtained at the root node
[c] From Li et al. [2007]
[d] The lower bound of the nonconvex function

Example 3.21. In the function

$$f(\mathbf{x}) = x_1^{-2} x_2^{-1.5} x_3^{1.2} x_4^{3} - 3x_3^{0.5} + x_2 - 4x_4, \qquad 1 \leq x_1, x_2, x_3, x_4 \leq 10, \tag{3.7.24}$$

only the first term is nonconvex. Transforming this term using the ET gives the following expression for the convex underestimator

$$\hat{f}_E(\mathbf{x}, \hat{\mathbf{X}}_\mathbf{E}) = x_1^{-2} x_2^{-1.5} e^{1.2\hat{X}_{3,E} + 3\hat{X}_{4,E}} - 3x_3^{0.5} + x_2 - 4x_4, \tag{3.7.25}$$

where $\hat{X}_{1,E}, \ldots, \hat{X}_{2,E}$ are the PLF-approximations of the inverse transformations $X_{i,E} = \ln x_i$. Transforming $f(\mathbf{x})$ using the PPT or NPT gives the convex function

$$\hat{f}_P(\mathbf{x}, \hat{\mathbf{X}}_\mathbf{P}) = x_1^{-2} x_2^{-1.5} \hat{X}_{3,P}^{1.2Q_3} \hat{X}_{4,P}^{3Q_4} - 3x_3^{0.5} + x_2 - 4x_4, \tag{3.7.26}$$

which underestimates $f(\mathbf{x})$, if $\hat{X}_{3,P}$ and $\hat{X}_{4,P}$ are the PLF-approximations of the inverse transformations $X_{3,P} = x_3^{1/Q_3}$ and $X_{4,P} = x_4^{1/Q_4}$, and the powers Q are chosen so that the transformed term is convex according to defs. 3.1 and 3.2. Here, the values on the powers Q_3 and Q_4 in the PPT were chosen as $Q_3 = -1$ and $Q_4 = 1.9$ as well as $Q_3 = 10.5$ and $Q = -2$; for the NPT the powers were chosen as $Q_3 = Q_4 = -10$.

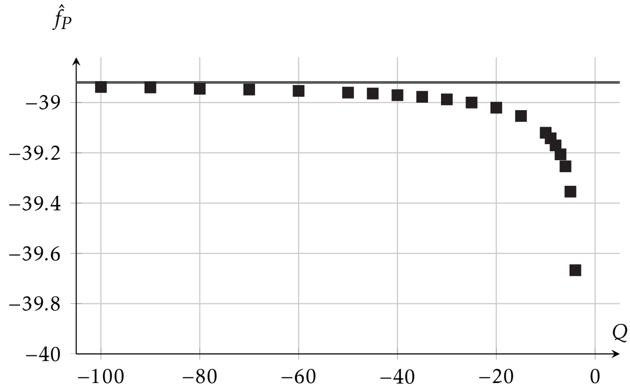
The lower bounds of the different convex underestimators are provided in table 3.2, and the impact of the power Q in the NPT is presented in fig 3.13b.

The results from exs. 3.20 and 3.21 show that the ET, PPT and NPT (with $Q_i = -10$) all result in quite good underestimators. In ex. 3.20 the PPT is better than the ET (and is almost equal to the global optimal solution). In ex. 3.21, however, the ET is better than the PPT in the case when $Q_3 = -1$, $Q_4 = 1.9$, but the PPT better than the ET when $Q_3 = 10.5$ and $Q_4 = -2$. According to thm. 3.11, neither of the ET nor the PPT is tighter in the whole domain, so which one is better depends only on where in the domain the optimal point lies. It also shows that the method from Li et al. [2007] (which corresponds to the NPT with $Q_i = -1$) is not very good in this case.

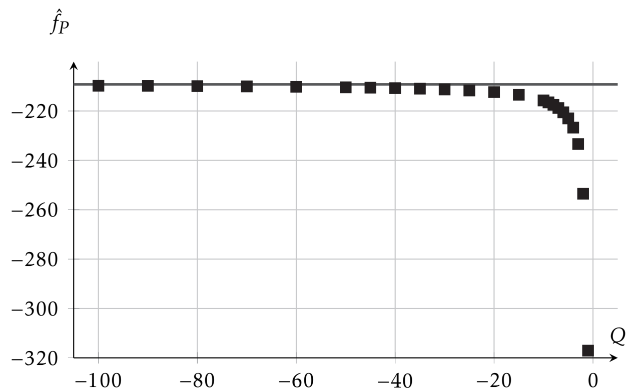
Table 3.2: Comparison of the LB of the convex underestimation techniques in ex. 3.21

Technique	ET	NPT[a]	PPT	BARON[b]	Li et al[c]	Opt. sol.[d]
Lower bound	-38.92	-39.12	-40.49[e] -38.73[f]	-42.31	-41.24	-38.08

[a] With the transformation powers $Q_i = -10$
[b] Obtained at the root node
[c] From Li et al. [2007]
[d] The lower bound of the nonconvex function
[e] With the powers $Q_3 = -1$ and $Q_4 = 1.9$
[f] With the powers $Q_3 = 10.5$ and $Q_4 = -2$



(**a**) Ex. 3.20



(**b**) Ex. 3.21

Figure 3.13: The impact of the transformation power Q used in the NPT convex underestimators in exs. 3.20 and 3.21. The gray line is the lower bound of the ET underestimator and the black squares the lower bound of the NPT underestimator.

CHAPTER 4

Optimizing the single-variable transformations

As mentioned in Chapter 3, there are often many ways of convexifying nonconvex signomial terms using the ET or any of the PTs. When transforming signomial functions of more than one term, or problems containing several signomial functions, there are even more combinations of transformations applicable, since different transformations can be used in different terms. It can be difficult to determine just by looking at the signomials, what set of transformations will result in, *e.g.*, the minimum amount of transformations or the minimum amount of variables transformed. Therefore, in this chapter, a method for obtaining an optimized set of transformations for the signomial terms in a nonconvex signomial problem, is presented. This method is based on formulating and solving a MILP problem.

Several versions of the MILP problem formulation have been presented in the articles which this thesis is based on. In Papers I and II the method for optimizing PTs was introduced. It was extended to also include the ET in Paper III. Finally in Paper VII, some additional enhancements were presented, and it is this form which the method described here mostly resembles.

4.1 The MILP method

The goal is to formulate a MILP problem, the solution of which is an optimized set of transformations for convexifying a given MISP consisting of one or more signomial terms (numbered $j = 1,\ldots,J_T$) in one or more variables x_i, $i = 1,\ldots,I$. Depending on the strategy parameters in the MILP problem, different sets of transformations are obtained, and it will be shown later on, that these can have a large impact on the combinatorial complexity of the resulting transformed MISP. The complete MILP problem formulation is also compiled in Appendix A.

Since the convexity requirements for positive and negative signomial terms are different according to thms. 2.5 and 2.6, the conditions enforcing correct transformations must also be different in these two cases. Also, there are more degrees of freedom when convexifying positive terms, as there are two types of applicable single-variable transformations, namely the exponential and power transformations. Furthermore, the latter allows for the choice between the PPT and NPT.

4.1.1 The variables in the MILP problem formulation

In this section the variables used in the MILP problem are briefly described. They are also summarized in tables 4.1 and 4.2.

In the MILP problem formulation, three types of real variables are used. The first one is the transformation power Q_{ji} in the single-variable PT, *i.e.*, $x_i = {X_{ji}}^{Q_{ji}}$. If the variable x_i in the j-th term is transformed using a single-variable ET, the power Q_{ji} is set to be equal to one. This is also the case if the variable is not transformed at all. The other types of real variables are Δ_{ji} and Δ'_{ji}, which are used to favor numerically stable transformations. These are further explained in Section 4.1.5.

To be able to determine whether a certain variable is transformed in a term, the binary variable b_{ji} is used. Its value is one if the variable x_i is transformed in the j-th signomial term and zero otherwise. Related to this one is the binary variable B_i which takes the value one if the variable x_i is transformed in any of the signomial terms in the whole problem. This logical condition can be written as

$$\forall i: \quad \sum_{j=1}^{J_T} b_{ji} \leq J_T B_i, \tag{4.1.1}$$

where J_T is the total number of signomial terms in the problem.

For positive signomial terms, additional binary variables are needed to provide the logic for enabling the choice between the ET and the PTs. In the first case, *i.e.*, when the variable x_i in the j-th term is transformed using a single-variable ET, the variable b_{ji}^{ET} is one and b_{ji}^{PT} is zero, and vice versa if any of the PTs are used. If the variable is not transformed at all $b_{ji}^{ET} = b_{ji}^{PT} = 0$. Furthermore, the variable α_{ji} is needed to differentiate between the PPT and NPT in such a way that α_{ji} is equal to one if Q_{ji} is positive (or the ET is used) and zero otherwise. The variable β_{ji} is equal to one if the power $p_{ji}Q_{ji}$ for the transformed variable X_{ji} is positive and zero if it is negative.

A member of the last group of binary variables needed is $\gamma_{j_1 j_2 i}$, which assumes the value one if different transformations are used on the variable x_i in the two signomial terms with indices j_1 and j_2.

4.1.2 The objective function and strategy parameters

Different sets of optimized transformations are obtained by changing the values on the strategy parameters δ_R, δ_Z, δ_{NT}, δ_{NS}, δ_{ET}, δ_{PT}, δ_P and δ_I in the objective of the MILP

Table 4.1: The binary decision variables in the MILP problem formulation

Variable	Value	Description
B_i	1	x_i is transformed in some signomial term.
	0	x_i is not transformed in any signomial term.
b_{ji}	1	x_i is transformed in the j-th term.
	0	x_i is not transformed in the j-th term.
b_{ji}^{ET*}	1	x_i is transformed using the ET in the j-th term.
	0	x_i is not transformed using the ET in the j-th term.
b_{ji}^{PT*}	1	x_i is transformed using any of the PTs in the j-th term.
	0	x_i is not transformed using any of the PTs in the j-th term.
$\alpha_{ji}{}^*$	1	The PPT or ET is used on x_i in the j-th term ($1 \leq Q_{ji} \leq Q_{\max}$).
	0	The PPT or ET is not used on x_i in the j-th term ($-Q_{\min} \leq Q_{ji} \leq -\epsilon$.
$\beta_{ji}{}^*$	1	The power $p_{ji}Q_{ji}$ is positive ($0 < p_{ji}Q_{ji} \leq p_{ji}Q_{\max}$).
	0	The power $p_{ji}Q_{ji}$ is negative ($-p_{ji}Q_{\min} \leq p_{ji}Q_{ji} < 0$).
$\gamma_{j_1 j_2 i}$	1	Different transformations are used on x_i in the j_1-st and j_2-nd terms.
	0	Identical transformations are used on x_i in the j_1-st and j_2-nd terms.

* These variables are only defined for $j: \; c_j > 0$, *i.e.*, for positive signomial terms

problem, which is to minimize the function

$$\begin{aligned} &\underbrace{\delta_R \sum_{\substack{i=1 \\ x_i \in \mathbb{R}}}^{I} r_i B_i + \delta_Z \sum_{\substack{i=1 \\ x_i \in \mathbb{Z}}}^{I} r_i B_i}_{(I)} \quad + \quad \underbrace{\sum_{j=1}^{J_T} \sum_{\substack{i=1 \\ p_{ji} \neq 0}}^{I} (\delta_{NT} b_{ji} + \delta_{NS} \Delta_{ji})}_{(II)} \quad + \\ &+ \quad \underbrace{\sum_{\substack{j=1 \\ c_j > 0}}^{J_T} \sum_{\substack{i=1 \\ p_{ji} > 0}}^{I} (\delta_{ET} b_{ji}^{ET} + \delta_{PT} b_{ji}^{PT} + \delta_P \beta_{ji})}_{(III)} \quad + \quad \underbrace{\delta_I \sum_{j_1=1}^{J_T} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{J_T} \sum_{\substack{i=1 \\ p_{ji} \neq 0}}^{I} \gamma_{j_1 j_2 i}}_{(IV)} \, . \end{aligned} \tag{4.1.2}$$

The strategy parameters are summarized in table 4.3.

In part (I) of eq. (4.1.2) the goal is to minimize the number of original variables x_i transformed at all in any of the terms. The reason for this is that parts of the expressions and variables used in the PLFs can be reused when transforming the same original variable in different signomial terms, even if the transformations used are not the same. For example, even though the variable x_i is transformed using a single-variable PT $x_i = X_{j_1 i}^{Q_{j_1 i}}$ in the j_1-st term and a single-variable ET $x_i = \exp X_{j_2 i}$ in the j_2-nd term, the same binary or SOS2 variables can be used in the PLFs for both transformations. Thus, by minimizing the number of transformed original variables, the combinatorial complexity

Table 4.2: The real variables in the MILP problem formulation

Variable	Description	Domain
Q_{ji}	The transformation power used on x_i in the j-th term.	$\mathbb{R}$
Δ_{ji}	Negative terms: The deviation from one for the sum of the powers in the j-th term (plus a penalty from Δ'_{ji}). Positive terms: The deviation from P_{pos} for the sum of the powers in the j-th term if a PT with positive power is used on x_{ji}, otherwise the deviation from P_{neg}.	$\mathbb{R}$
Δ'_{ji}	Negative terms: The deviation for the powers $p_{ji}Q_{ji}$ from the mean of the powers for the whole term.	$\mathbb{R}_+$

Table 4.3: The strategy parameters in the MILP problem formulation

Parameter	Description	Domain
δ_R	Penalizes the number of transformed real variables.	$\mathbb{R}_+ \cup \{0\}$
δ_Z	Penalizes the number of transformed discrete variables.	$\mathbb{R}_+ \cup \{0\}$
δ_{NT}	Penalizes the number of transformations.	$\mathbb{R}_+ \cup \{0\}$
δ_{NS}	Penalizes numerical unstable transformations.	$\mathbb{R}_+ \cup \{0\}$
δ_{ET}	Penalizes the ET in positive terms.	$\mathbb{R}_+ \cup \{0\}$
δ_{PT}	Penalizes the PTs in positive terms.	$\mathbb{R}_+ \cup \{0\}$
δ_P	Penalizes the PPT if positive, favors the PPT if negative.	$\mathbb{R}$
δ_I	Penalizes different transformations for the same variables in different terms.	$\mathbb{R}_+ \cup \{0\}$
Q_{min}	$-Q_{\text{min}}$ is the smallest transformation power Q allowed.	$\mathbb{R}_+$
Q_{max}	Q_{max} is the largest transformation power Q allowed.	$\mathbb{R}_+$
P_{neg}	The desired power (pQ) in the PTs with negative powers.	$\mathbb{R}_-$
P_{pos}	The desired power (pQ) in the PTs with positive powers.	$\mathbb{R}_+$

of the transformed problem can be reduced.

The reason the sum in (I) is split into parts corresponding to the discrete and real variables is that the PLFs for discrete variables only require a finite amount of breakpoints, namely the points corresponding to the domain of the variables, to provide an exact representation of the original variable, whereas the PLFs for real variables theoretically require an infinite amount of breakpoints. Therefore, it may prove beneficial to favor transformations in discrete variables to keep the number of variables used in the PLFs to a minimum in the long run. The optimization strategy is altered by changing the value of the parameters δ_R and δ_Z, corresponding to the real and discrete variables respectively.

Following the same path of reasoning, an additional penalty factor r_i can be included to additionally favor transformations in variables with small domains. This factor can, *e.g.*, be defined according to

$$r_i = \begin{cases} 1 + \epsilon_d(\overline{x}_i - \underline{x}_i)^d & \text{if } x_i \in \mathbb{R}, \\ 1 + \epsilon_d(\overline{x}_i - \underline{x}_i - 1)^d & \text{if } x_i \in \mathbb{Z}, \end{cases} \tag{4.1.3}$$

where ϵ_d is a small positive parameter, d is a positive integer and $x \in [\underline{x}, \overline{x}]$. Using this definition of r_i a binary variable receives no additional penalty, but real- or integer variables with larger domains do. Note that, since r_i is a parameter which is calculated before solving the MILP problem, including this parameter does not affect the linearity of the objective function.

Part (II) of eq. (4.1.2) consists itself of two parts. The parameters δ_{NT} and δ_{NS} penalizes the total number of transformations in the whole problem and numerically unstable transformations respectively.

In part (III) of the objective function, certain types of transformations in the positive signomial terms can be favored by altering the values of the parameters δ_{ET}, δ_{PT} and δ_P. Depending on the value of the first two parameters, the ET or any the PTs is favored in positive terms. Additionally, the parameter δ_P favors the PPT if negative and penalizes the PPT if positive.

In part (IV) of eq. (4.1.2) the goal is to minimize the total number of different transformations used for each original variable. The reason for this is, similarly to the case above, that the same expressions and variables for the PLFs can be used for different transformations of the same variable in different terms. However, when the same transformation $x_i = T_{ji}(X_{ji})$ is used on the same variable x_i, the transformation variable X_{ji} can also be reused. Therefore, it does not matter, complexity-wise, whether we transform a variable in one or a hundred terms, as long as, the same transformation is used. The emphasis on minimizing the number of different transformations in the whole problem is controlled by the strategy parameter δ_I.

In addition to the objective function, constraints must be included in the MILP problem guaranteeing transformations which result in convexified signomial terms in line with defs. 3.1, 3.2 and 3.4 for positive signomial terms and def. 3.5 for negative terms. These constraints are described in the following two sections.

4.1.3 Conditions for positive terms

Defs. 3.1, 3.2 and 3.4 state that a positive signomial term can be convexified using either the ET or one of the NPT or PPT. However, only variables x_i with positive powers p_{ji} in a term need to be transformed, and, at most, one of the transformations ET, NPT and PPT should be used in any term.

If the PPT is used, the sum of the powers $p_{ji}Q_{ji}$ in the transformed term should be larger than or equal to one, which can be written as the constraint

$$\forall j: \sum_{\substack{i=1 \\ p_{ji} \neq 0}}^{I} p_{ji}Q_{ji} - M_1 \sum_{\substack{i=1 \\ p_{ji} > 0}}^{I} \alpha_{ji} + M_1 \sum_{\substack{i=1 \\ p_{ji} > 0}}^{I} b_{ji}^{ET} \geq 1 - M_1. \tag{4.1.4}$$

This constraint is relaxed whenever the term is not transformed or either one of the ET or NPT is used, as long as the positive parameter M_1 is chosen large enough. For example, it can be chosen according to

$$M_1 > \max_j \left(1 + Q_{\min} \sum_{i=1}^{I} p_{ji} \right), \tag{4.1.5}$$

where $Q_{\min}$ is a parameter specifying the minimum allowed transformation power in a PT with negative power.

Furthermore, in the PPT only one of the variables in the term can have a positive power after the convexification step, which can be enforced with the constraint

$$\forall j: \sum_{\substack{i=1\\ p_{ji}>0}}^{I} \alpha_{ji} - \sum_{\substack{i=1\\ p_{ji}>0}}^{I} b_{ji}^{ET} \leq 1. \tag{4.1.6}$$

This constraint is relaxed if the ET is used on the term, because then all the α_{ji}'s are equal to one. To set the correct limits on the value of the transformation power, depending on the value of α_{ji}, the following condition is used

$$\forall i,j: \ p_{ji} > 0: \quad -Q_{\min} + (Q_{\min}+1)\alpha_{ji} \leq Q_{ji} \leq Q_{\max}\alpha_{ji} - \epsilon(1-\alpha_{ji}). \tag{4.1.7}$$

here $-Q_{\min} < 0$ and $Q_{\max} > 1$ are the smallest and largest powers allowed in the PTs respectively. The parameter ϵ is a small positive value, *e.g.*, $\epsilon = 1/\max\{Q_{\min}, Q_{\max}\}$. Also, if the ET is used to transform x_i in the j-th term, then Q_{ji} should be equal to one, which can be guaranteed by the following condition

$$\forall i,j: \ p_{ji} > 0: \quad -Q_{\min} + (Q_{\min}+1)b_{ji}^{ET} \leq Q_{ji} \leq (Q_{\max}-1)(1-b_{ji}^{ET}) + 1. \tag{4.1.8}$$

The binary variable b_{ji}^{PT}, indicating whether x_i is transformed by a single-variable PT in the j-th term, also depends on the value of α_{ji} and Q_{ji}, according to

$$\forall i: \ p_{ji} > 0: \quad b_{ji}^{PT} \geq 1 - \alpha_{ji} \tag{4.1.9}$$

and

$$\forall i: \ p_{ji} > 0: \quad \epsilon(Q_{ji}-1) \leq b_{ji}^{PT} \leq (1-\epsilon)Q_{ji} + M_1(1-\alpha_{ji}). \tag{4.1.10}$$

The following condition, which determines the value for the binary variable β_{ji}, indicating whether the corresponding power in the transformed term, *i.e.*, $p_{ji}Q_{ji}$, is positive or negative, must also be added

$$\forall i: \ p_{ji} > 0: \quad -Q_{\min}(1-\beta_{ji}) \leq Q_{ji} \leq Q_{\max}\beta_{ji}. \tag{4.1.11}$$

For variables with positive powers in positive signomial terms, either the ET or one of the two PTs can be applied to the term. However, when one variable is transformed using the single-variable ET, all the other variables with positive powers must undergo the same transformation. This can be enforced using the following condition

$$\forall i: \; p_{ji} > 0: \quad b_{ji}^{ET} \geq \frac{1}{I_j} \sum_{\substack{i=1 \\ p_{ji}>0}}^{I} b_{ji}^{ET}, \tag{4.1.12}$$

where I_j is the number of variables in the current term, *i.e.*,

$$\forall j: \; I_j = \mathrm{card}\{x_i \mid p_{ji} \neq 0\}. \tag{4.1.13}$$

Furthermore, at most, one type of transformation can be applied to an individual variable, which can be formulated as

$$\forall i: \; p_{ji} > 0: \quad b_{ji}^{ET} + b_{ji}^{PT} \leq 1. \tag{4.1.14}$$

The value of the variable b_{ji}, indicating whether x_i is transformed in the j-th term, is equal to one if the variable is transformed using a single-variable ET or PT, so

$$\forall i: \; p_{ji} > 0: \quad b_{ji} = \max\{b_{ji}^{ET}, b_{ji}^{PT}\} \quad \Longleftrightarrow \quad \begin{cases} b_{ji} \geq b_{ji}^{ET}, \\ b_{ji} \geq b_{ji}^{PT}. \end{cases} \tag{4.1.15}$$

Since variables with negative powers in a positive term do not need to be transformed, the transformation power Q_{ji} and the binary variable b_{ji}, indicating a transformation, are fixed to one and zero respectively, *i.e.*,

$$\forall i: \; p_{ji} < 0: \quad Q_{ji} = 1, \quad b_{ji} = 0. \tag{4.1.16}$$

4.1.4 Conditions for negative terms

The conditions for obtaining correct transformations for negative signomial terms are simpler than those for positive signomial terms, since the PT for negative signomial terms from def. 3.5 is the only applicable transformation.

The only condition for convexity for a transformed term is that the sum of the powers $p_{ji}Q_{ji}$ should be larger than zero and less than or equal to one, *i.e.*,

$$0 < \sum_{i=1}^{I} p_{ji} Q_{ji} \leq 1. \tag{4.1.17}$$

The binary b_{ji}, indicating whether a transformation is performed on the variable x_i in the j-th term, should be zero if the transformation power Q_{ji} is equal to one, *i.e.*, no transformation occurs, and one otherwise. For a variable with a positive power, the relation between Q_{ji} and b_{ji} can be expressed

$$1 - b_{ji} \leq Q_{ji} \leq 1 - \epsilon b_{ji} \qquad \text{and} \qquad Q_{ji} \geq \epsilon. \tag{4.1.18}$$

The parameter ϵ is a small positive value, *e.g.*, $\epsilon = 1/\max\{Q_{\min}, Q_{\max}\}$. A transformation is always necessary for a variable with a negative power, so the corresponding constraints are

$$-Q_{\min} \leq Q_{ji} \leq -\epsilon \qquad \text{and} \qquad b_{ji} = 1. \tag{4.1.19}$$

Also, here a lower bound on the transformation power Q_{ji} of $-Q_{\min}$ is set.

4.1.5 Conditions for favoring numerical stable transformations

To get numerically stable power transformations, the conditions given in this section can be used. For positive signomial terms, the deviation Δ_{ji} from either of the two parameters P_{pos} and P_{neg}, depending on whether the power $p_{ji}Q_{ji}$ in the transformed term is positive or negative, is used to penalize transformations with too large or too small powers. The conditions on the real variables Δ_{ji} are

$$\forall i: \ p_{ji} > 0: \quad |p_{ji}Q_{ji} - P_{\text{pos}}| \leq \Delta_{ji} + M_2(1 - \beta_{ji} + b_{ji}^{ET}) \quad \text{and} \tag{4.1.20}$$

$$\forall i: \ p_{ji} > 0: \quad |p_{ji}Q_{ji} - P_{\text{neg}}| \leq \Delta_{ji} + M_2(\beta_{ji} + b_{ji}^{ET}). \tag{4.1.21}$$

These constraints can be rewritten without absolute values using two inequalities each as

$$\forall i: \ p_{ji} > 0: \ -\Delta_{ji} - M_2(1 - \beta_{ji} + b_{ji}^{ET}) \leq p_{ji}Q_{ji} - P_{\text{pos}} \leq \Delta_{ji} + M_2(1 - \beta_{ji} + b_{ji}^{ET}) \tag{4.1.22}$$

and

$$\forall i: \ p_{ji} > 0: \ -\Delta_{ji} - M_2(\beta_{ji} + b_{ji}^{ET}) \leq p_{ji}Q_{ji} - P_{\text{neg}} \leq \Delta_{ji} + M_2(\beta_{ji} + b_{ji}^{ET}). \tag{4.1.23}$$

Here M_2 can be chosen, *e.g.*, according to $M_2 > \max\{|P_{\text{neg}}|,\ P_{\text{pos}}\}$. For variables with negative powers in a positive term, the values of the deviation are set to zero, *i.e.*,

$$\forall i: \ p_{ji} < 0: \quad \Delta_{ji} = 0. \tag{4.1.24}$$

For negative signomial terms, transformation powers which result in as similar powers $p_{ji}Q_{ji}$ in the convexified terms as possible are favored here. This is done by introducing an extra positive real variable Δ'_{ji}, which corresponds to the deviation of the power $p_{ji}Q_{ji}$ from the mean of the powers in the transformed term. This can be mathematically written as the constraint

$$\forall i: \ p_{ji} \neq 0: \quad \Delta'_{ji} \geq \left| p_{ji}Q_{ji} - \frac{1}{I_j} \sum_{i=1}^{I} p_{ji}Q_{ji} \right|. \tag{4.1.25}$$

The value of the variable Δ_{ji} is then determined by

$$\forall i: \ p_{ji} \neq 0: \quad \Delta_{ji} \geq 1 - \sum_{i=1}^{I} p_{ji}Q_{ji} + \epsilon \Delta'_{ji}, \tag{4.1.26}$$

i.e., Δ_{ji} is the deviation between one and the sum of the powers $p_{ji}Q_{ji}$ plus a small contribution from Δ'_{ji}. Using these conditions, the powers $p_{ji}Q_{ji}$ are kept as close as possible to each other and the sum of the powers in the transformed term as close to one as possible.

4.1.6 Conditions for favoring identical transformations

To keep the number of additional binary or SOS2 variables needed in the PLFs as small as possible when adding more breakpoints, identical transformations for the same variable in different terms can be favored. Different transformations are defined as either transformations of different types (single-variable exponential and power transformations) or two single-variable power transformations with different transformation powers ($Q_{j_1 i} \neq Q_{j_2 i}$). The values for the binaries γ are determined using the following conditions

$$\forall i \in \{1,\ldots,I\},\ j_1,\ j_2 \in \{1,\ldots,J\},\ j_1 \neq j_2,\ p_{j_1 i},\ p_{j_2 i} \neq 0,\ \operatorname{sgn} c_{j_1} = \operatorname{sgn} c_{j_2}:$$
$$\begin{cases} Q_{j_1 i} - Q_{j_2 i} + b^{ET}_{j_1 i} - b^{ET}_{j_2 i} - M_1(2 - b_{j_1 i} - b_{j_2 i}) \leq M_1 \gamma_{j_1 j_2 i}, \\ \gamma_{j_1 j_2 i} = \gamma_{j_2 j_1 i}, \end{cases} \tag{4.1.27}$$

and where the parameter M_1 can be chosen, *e.g.*, as $M_1 = \max\{Q_{\min}, Q_{\max}\}$.

4.2 Impact of the strategy parameters

As mentioned earlier, the set of transformations obtained through the solution of the MILP problem described above, depends on the value of the parameters in table 4.3. By providing different values of the strategy parameters in the objective function, different types of transformations or certain properties can be emphasized.

There is some redundancy in the parameters, *e.g.*, the values of the parameters δ_R, δ_Z and δ_I all favor as few transformations as possible. However, all of these options are included to allow the user of the method to specify which properties to favor. The magnitude of a strategy parameter relative to another determines the emphasis. Also, since it is possible to set the value of one or more of the strategy parameters in the objective function to zero, it is possible to favor only certain aspects of the transformations.

The following GP problem originally from Rijckaert and Martens [1978] is used to illustrate how the value of the strategy parameters impact on the set of transformations obtained from the MILP method.

Example 4.1. The original problem is reformulated to the following SP problem:

$$\begin{aligned}
\text{minimize} \quad & \mu, \\
\text{subject to} \quad & 2.0\, x_1^{0.9} x_2^{-1.5} x_3^{-3} + \underline{5.0\, x_4^{-0.3} x_5^{2.6}} + 4.7\, x_6^{-1.8} x_7^{-0.5} x_8 - \mu \leq 0, \\
& 7.2\, x_1^{-3.8} x_2^{2.2} x_3^{4.3} + \underline{0.5\, x_4^{-0.7} x_5^{-1.6}} + 0.2\, x_6^{4.3} x_7^{-1.9} x_8^{8.5} \leq 1, \\
& 10.0\, x_1^{2.3} x_2^{1.7} x_3^{4.5} \leq 1, \qquad 0.6\, x_4^{-2.1} x_5^{0.4} \leq 1, \\
& \underline{6.2\, x_6^{4.5} x_7^{-2.7} x_8^{-0.6}} \leq 1, \qquad 3.1\, x_1^{1.6} x_2^{0.4} x_3^{-3.8} \leq 1, \\
& \underline{3.7\, x_4^{5.4} x_5^{1.3}} \leq 1, \qquad 0.3\, x_6^{-1.1} x_7^{7.3} x_8^{-5.6} \leq 1, \\
& x_1, \ldots, x_8 \geq 0.01.
\end{aligned}$$

Table 4.4: The values of the parameters in the MILP formulation in ex. 4.1.

Strategy	δ_R	δ_Z	δ_{NT}	δ_{NS}	δ_{ET}	δ_{PT}	δ_P	δ_I
I	-	-	0.1	0.01	-	10	-	-
II	10	-	0.1	0.01	-	-	-	-
III	-	-	10	0.01	-	-	-	-
IV	1	-	0.1	0.01	-	-	-	10

Table 4.5: The number of transformations required for convexification of the problem in ex. 4.1.

Strategy	# Transformations	# Transf. orig. var.	# Different transf.
I	15	8	8
II	12	6	12
III	12	8	12
IV	12	6	6

The problem has eight continuous real positive variables and 12 signomial terms, of which three are convex (underlined). The transformations required for convexifying the problem subject to four different sets of strategy parameters are obtained by solving the MILP problem. The values of the strategy parameters are indicated in table 4.4; the values of the other parameters are $Q_{\text{min}} = Q_{\text{max}} = M_1 = M_2 = 20$, $\epsilon = 0.05$, $P_{\text{neg}} = -1$, $P_{\text{pos}} = 1$, and $r_i = 1$. The different cases can be verbally explained according to:

Strategy I The ET is favored over the PTs.

Strategy II The number of original (real) variables transformed in the whole problem is minimized.

Strategy III The total number of transformations is minimized.

Strategy IV The number of different transformations is minimized.

The number of transformations obtained from solving the MILP problem using the different strategies are given in table 4.5. The results indicate that there can be a significant difference in the combinatorial complexity of the transformed problem.

CHAPTER 5

SGO – A GO algorithm for MISP problems

Solving MINLP problems containing signomial functions in the objective function, constraints or both, is often a challenging task. Even simple signomial functions are usually nonconvex, implying that most convex MINLP solvers will find only local solutions.

In this chapter, an iterative global optimization (GO) algorithm for MINLP problems containing signomial functions is described. The first version of the algorithm was presented in Westerlund and Westerlund [2003] as the Generalized geometric programming extended cutting plane (GGPECP) algorithm. It is also described in detail in Westerlund [2005]. The GGPECP algorithm used fixed transformation techniques based on the power and exponential transformations in Chapter 3. In the algorithm presented in Paper II, the optimization method for the transformations from Chapter 4 was included as a preprocessing step to provide the transformations used for convexifying the nonconvex signomial terms. Previous versions of the algorithm used αECP (Westerlund and Pörn [2002]) as the MINLP solver, however in Paper VI, a more general algorithm was described, which could use any convex MINLP solver. This algorithm is the topic of this chapter.

The Signomial global optimization (SGO) algorithm, solves MISP problems of the form in def. 2.8 as a sequence of convexified and overestimated MINLP problems. In the problem, only the signomial terms are allowed to be nonconvex, since they are the only ones being transformed. A flowchart of the SGO algorithm is given in fig. 5.1.

5.1 The preprocessing step

To obtain transformations transforming the original MISP to an overestimated convex relaxed form, the convexification and underestimation techniques in Chapter 3 are im-

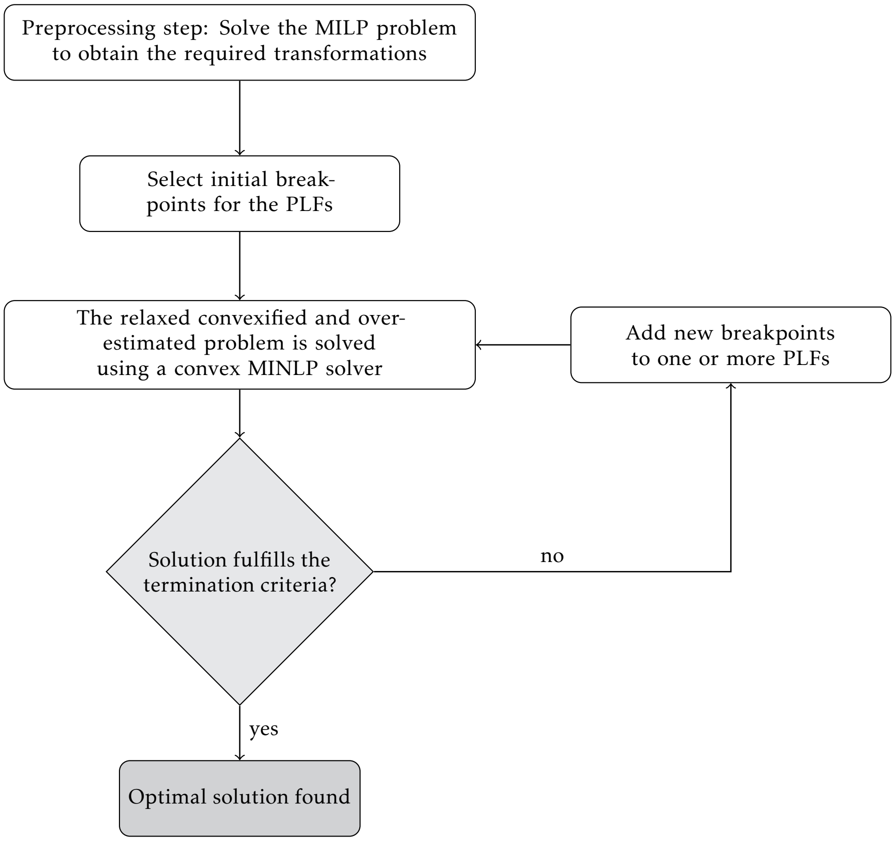


Figure 5.1: Flowchart of the SGO algorithm

plemented using the MILP method in Chapter 4 as a preprocessing step. The parameters in the MILP problem will then determine which set of solutions is used to convexify the signomial terms in the problem.

After the initial transformation step, the PLFs used for approximating the inverse transformations are updated in subsequent iterations. As additional breakpoints are added to the PLFs, the optimal solution of the transformed problem will, under certain conditions, converge to that of the original nonconvex problem. Thus, the selection of what breakpoints to add is of crucial importance for the convergence of the method.

5.2 Discretization strategies

When determining how to improve the PLFs in subsequent iterations, there are two things to take into consideration. First, which PLFs should be modified, *i.e.*, which approximation of the transformation variables should be improved. The second thing to consider is what breakpoint or breakpoints to add to the selected PLFs. The strategy for selecting the variables and breakpoints can have a significant impact on the solution-time, and for some problems, whether the global solution is actually found or not. The discretization strategies are more thoroughly described in Westerlund [2005]; here only a quick summary is presented.

5.2.1 Selection of the variables

The simplest strategy is, of course, to add breakpoints to all the transformation variables in the whole problem. Since adding a breakpoint corresponds to adding at least one binary or SOS2 variable per transformation variable in this case, the combinatorial complexity will, in this case, grow out of control for large problems with transformations of many different variables. Therefore, the goal should be to only add breakpoints to as few approximations of transformation variables as possible. For example, there is no need to add more breakpoints to variables only appearing in generalized signomial constraints already fulfilled.

Additional restrictions can also be introduced, for example, only the PLFs of the transformation variables appearing in the original nonconvex generalized signomial constraints which are the most violated can be selected. Furthermore, only the variables where adding the breakpoint will have the largest impact on the error in this constraint, can be improved. In this last case, the breakpoint to be added must, of course, be determined before selecting the variables.

5.2.2 Selection of the breakpoints

There are also several different strategies available for determining what new breakpoint(s) to include in the PLFs of the transformation variables. For example, the following strategies can be used:

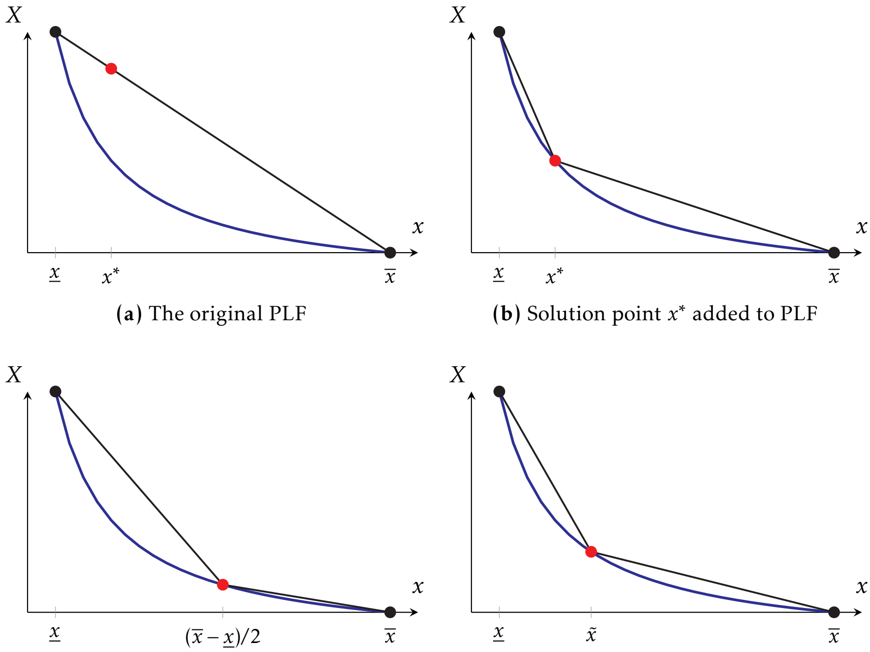
1. Add the solution point of the previous iteration.

2. Add the midpoint of the interval of existing breakpoints in which the previous solution is.

3. Add the most deviating point, *i.e.*, the point with the largest approximation error according to thm. 3.13, of the transformation in the interval in which the previous solution is.

How the PLF changes when these strategies are used to add a new breakpoint is illustrated in fig. 5.2.

When the original variable in question is discrete, it is possible to exactly describe the inverse transformation with PLFs with an finite amount of breakpoints (corresponding to the number of discrete values the variable can assume). In this case, it makes no sense to add breakpoints which do not belong to the original set, *i.e.*, only integer breakpoints should be added. However, in the second strategy, noninteger breakpoints can be selected, but these should be replaced, for example, with the closest integer value currently not in the set of breakpoints.



(a) The original PLF

(b) Solution point x^* added to PLF

(c) Midpoint of interval added to PLF

(d) Most deviating point $\tilde{x}$ added to PLF

Figure 5.2: Illustration of the impact on the PLF-approximation when using the different strategies for adding new breakpoints.

The problem with the first strategy is that this can lead to the global optimal solution not being found, as the breakpoints can be added "too close" to each other. For the second strategy, it was proved in Westerlund [2005] that the global optimal solution will always be found.

The third strategy, *i.e.*, adding the point where the approximation error is the largest in the PLFs, has one major drawback. Since the breakpoint in this case depends on what single-variable transformation is used, the same set of binary or SOS2 variables can no longer be utilized in the PLF-formulations for different transformations of the same original variable.

Although, convergence to the global optimal solution can not be guaranteed if including only the solution points of the previous iteration as new breakpoints in the next iteration, the solution time and/or number of iterations required can be less in this case. Therefore depending on the problem, hybrid methods, where adding more than one breakpoint in each iteration to the transformation variables selected, could also prove to be efficient in some cases.

5.3 Termination criteria

The SGO algorithm is based on iteratively solving overestimated subproblems, so there must be some kind of criteria for when the algorithm terminates. Since the feasible region of the subproblems overestimate the feasible region of the original nonconvex problem, the global optimal solution is found whenever all the generalized signomial constraints in the original problem are satisfied in the overestimated problem. However, since this is a numerical method, the constraints can only be required to be satisfied to an epsilon-accuracy. This can be stated, for the $m = 1,\ldots,M$ generalized signomial constraints, as the condition

$$\max_m \left(q_m(\mathbf{x}^*) + \sigma_m(\mathbf{x}^*)\right) \leq \epsilon_T, \tag{5.3.1}$$

where $\mathbf{x}^*$ is the optimal value of the current subproblem and $\epsilon_T \geq 0$.

Since the piecewise linear approximations are exact at the breakpoints, another criterion for termination is when the maximum distances from the solution values of the variables involved in the transformations to the nearest breakpoint values is less than a small positive value ϵ_D. That is, if the set of breakpoints for the variable x_i is $\{\check{x}_{i,k}\}_{k=1}^{K_i}$, then

$$\max_i \left(\min_k |x_i^* - \check{x}_{i,k}| \right) \leq \epsilon_D, \tag{5.3.2}$$

where x_i^* is the optimal value for the variable x_i in the current iteration.

Of course, in practical use of the algorithm, the solution process can also be terminated when a maximum iteration count or a time-limit has been reached.

5.4 A numerical example

To illustrate how the SGO algorithm works, the following example from Paper II is included. This nonconvex MISP problem contains a linear objective function and two linear constraints as well as a generalized signomial constraint.

$$\begin{aligned}
\text{minimize} \quad & y - 3x, \\
\text{subject to} \quad & y + 5x \leq 36, \\
& -y + 0.25x \leq -1, \\
& \underbrace{2y^2 - 2y^{0.5} + 11y + 8x - 39}_{q(x,y)} \underbrace{-2x^{0.5}y^2 + 0.1x^{1.5}y^{1.5}}_{\sigma(x,y)} \leq 0, \\
& 1 \leq x \leq 7, \quad 1 \leq y \leq 7, \\
& x \in \mathbb{R}_+, \quad y \in \mathbb{Z}_+.
\end{aligned} \tag{5.4.1}$$

The only nonconvex part is the generalized signomial constraint $q(x,y) + \sigma(x,y)$. The variable x is a positive real variable and y a positive discrete variable. The integer-relaxed feasible region of this problem consists of two disjoint regions as shown in fig. 5.3.

The first step in the SGO algorithm is the preprocessing step, *i.e.*, to obtain the necessary transformations using the MILP method in Chapter 4. Depending on the

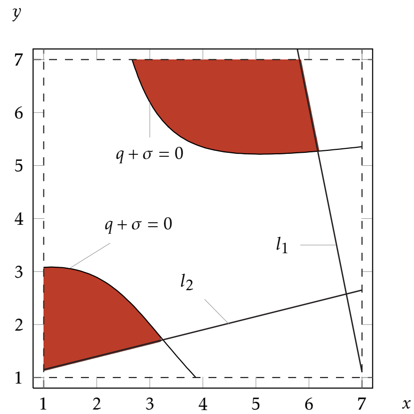


Figure 5.3: The integer-relaxed feasible region of the problem in Section 5.4. The contour at zero of the generalized signomial constraint $q(x,y)+\sigma(x,y)$, as well as the linear constraints $l_1:\ y+5x-36=0$ and $l_2:\ -y+0.25x+1=0$ are also shown in the figure.

values on the parameters in table 4.3, different transformations are obtained. The solution of the MILP problem (with $\delta_R=\delta_Z=1$, $\delta_{NS}=0.1$, $Q_{\max}=Q_{\min}=10$, $P_{\text{neg}}=-1$ and $P_{\text{pos}}=1$) gives the following transformations:

$$y=Y_1^{0.25} \text{ and } y=Y_2^{-1/3} \quad\Longrightarrow\quad Y_1=y^4 \text{ and } Y_2=y^{-3}. \tag{5.4.2}$$

That is, the variable x need not be transformed at all. This will reduce the amount of extra variables required in the PLFs, since the same SOS can be used in the approximations of both Y_1 and Y_2. However, since the transformation powers are different, the same transformation variable can not, unfortunately, be used in both terms.

The variable y in the two terms in $\sigma(x,y)$ are now replaced by the transformation variables Y_1 and Y_2 in the first and second term respectively, *i.e.*, the convexified generalized signomial constraint will be of the form

$$2y^2-2y^{0.5}+11y+8x-39-2x^{0.5}Y_1^{0.5}+0.1x^{1.5}Y_2^{-0.5}\le 0. \tag{5.4.3}$$

Thus far, the MINLP problem has not been changed, at least not in the sense that the problem is still nonconvex, because the relations between the transformation variables Y_1 and Y_2 are included in the problem; the nonconvexities have simply been moved from the generalized signomial constraint to the nonlinear equality constraints $Y_1=y^4$ and $Y_2=y^{-3}$. However, by approximating the inverse transformations with PLFs, the integer-relaxed problem will be convexified and overestimated. Initially, the interval endpoints $y=1$ and $y=7$ are used as breakpoints, *i.e.*, the PLFs will be

$$\begin{aligned}
\hat{Y}_1 &= 1\cdot w_1+2401\cdot w_2,\\
\hat{Y}_2 &= 1\cdot w_1+0.0029\cdot w_2,\\
y &= 1\cdot w_1+7\cdot w_2,\\
w_1+w_2 &= 1,
\end{aligned} \tag{5.4.4}$$

where w_1 and w_2 belong to a SOS of type 2.

Now a convex MINLP solver is used to solve the following convex MINLP problem

$$\begin{aligned}
\text{minimize} \quad & y - 3x, \\
\text{subject to} \quad & y + 5x \leq 36, \\
& -y + 0.25x \leq -1, \\
& 2y^2 - 2y^{0.5} + 11y + 8x - 39 - 2x^{0.5}\hat{Y}_1^{0.5} + 0.1x^{1.5}\hat{Y}_2^{-0.5} \leq 0, \\
& \hat{Y}_1 = 1 \cdot w_1 + 2401 \cdot w_2, \\
& \hat{Y}_2 = 1 \cdot w_1 + 0.0029 \cdot w_2, \\
& y = 1 \cdot w_1 + 7 \cdot w_2, \\
& w_1 + w_2 = 1, \\
& 1 \leq x \leq 7, \quad 1 \leq y \leq 7, \\
& x \in \mathbb{R}_+, \ y \in \mathbb{Z}_+, \ w_1, w_2 \in \text{SOS2}.
\end{aligned} \tag{5.4.5}$$

The feasible region at iteration 1 is shown in fig 5.5a. The optimal solution for this problem is $(x, y) = (6.6, 3)$ and the objective function value is -16.8. The LHS-value of the original generalized signomial constraint is 23.9. This value should be less than or equal to zero for the solution to be contained in the feasible region of the original problem. Therefore, more breakpoints are needed in the PLFs to improve the approximations.

In the next iteration, the solution point $y = 3$ is added as an additional gridpoint to the PLFs, *i.e.*, their expressions are updated according to:

$$\begin{aligned}
\hat{Y}_1 &= 1 \cdot w_1 + 81 \cdot w_2 + 2401 \cdot w_3, \\
\hat{Y}_2 &= 1 \cdot w_1 + 0.0370 \cdot w_2 + 0.0029 \cdot w_3, \\
y &= 1 \cdot w_1 + 3 \cdot w_2 + 7 \cdot w_3, \\
w_1 &+ w_2 + w_3 = 1.
\end{aligned} \tag{5.4.6}$$

The feasible region of the problem at iteration 2 is shown in fig 5.5b. Note that the previous solution point is excluded from the new feasible region. An illustration of the convex feasible region of the overestimated and convexified problem in three dimensions is given in fig. 5.4. Although this region is, in fact, five-dimensional with the variables w_1, w_2, w_3, x and y, it is possible to illustrate it in the three dimensions w_1, w_3 and x using the facts that

$$w_2 = 1 - w_1 - w_3 \quad \text{and} \quad y = w_1 + 3w_2 + 7w_3 = 3 - 2w_1 + 4w_3. \tag{5.4.7}$$

However, after adding more breakpoints, it is no longer possible to illustrate it in three dimensions, instead four or more would be required.

The updated problem is now solved and the obtained optimal solution is $(x, y) = (6.4, 4)$ with the objective function value -15.2. The original generalized signomial constraint still has the LHS-value 16.2, *i.e.*, the constraint is not yet satisfied, so further iterations are needed.

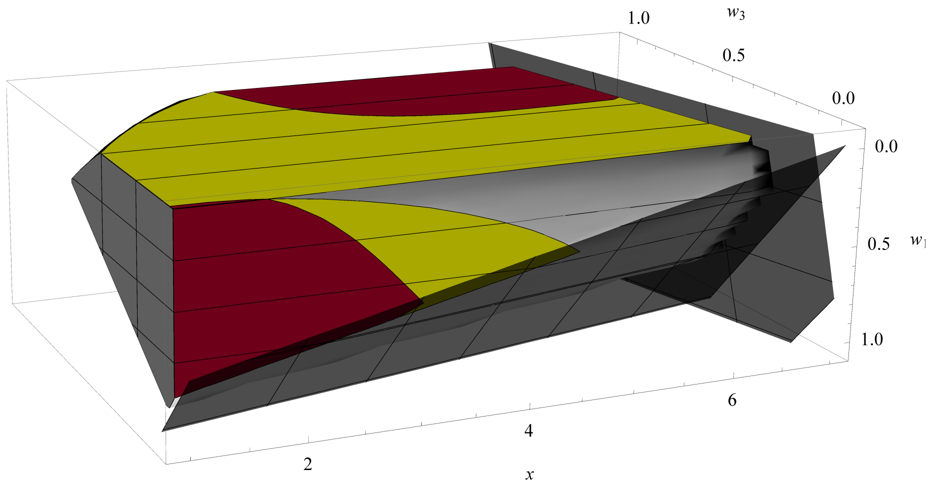


Figure 5.4: An illustration of the convexified feasible region of the problem in Section 5.4 when a breakpoint at $y = 3$ has been added. Also included in the figure are two planes representing the linear constraints. Note that the feasible region is convex.

Again, the solution value $y = 4$ is added as a new gridpoint and the problem with the PLFs

$$\begin{aligned}
\hat{Y}_1 &= 1 \cdot w_1 + 81 \cdot w_2 + 256 \cdot w_3 + 2401 \cdot w_4, \\
\hat{Y}_2 &= 1 \cdot w_1 + 0.0370 \cdot w_2 + 0.0156 \cdot w_3 + 0.0029 \cdot w_4, \\
y &= 1 \cdot w_1 + 3 \cdot w_2 + 4 \cdot w_3 + 7 \cdot w_4, \\
w_1 &+ w_2 + w_3 + w_4 = 1,
\end{aligned} \tag{5.4.8}$$

is solved. The feasible region is shown in fig 5.5c. The optimal solution in this iteration is $(x, y) = (6.2, 5)$ with the objective function value -13.6. The original generalized signomial constraint has the LHS-value 3.9.

Finally, the previous solution $y = 5$ is added as a gridpoint to the PLFs, according to

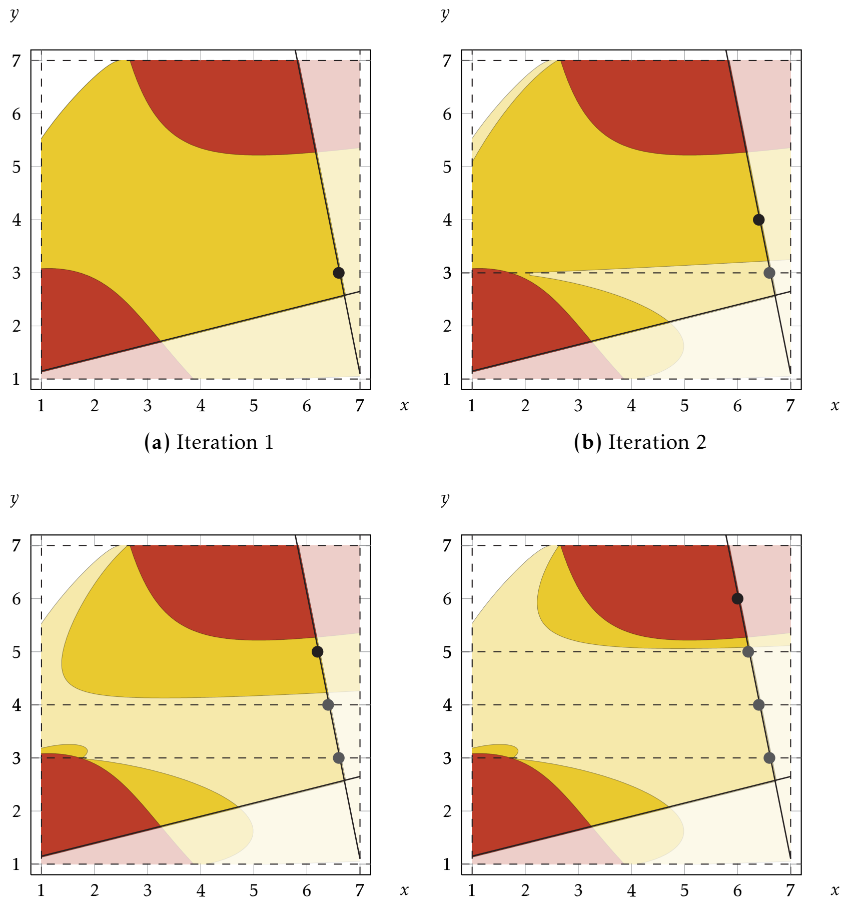
$$\begin{aligned}
\hat{Y}_1 &= 1 \cdot w_1 + 81 \cdot w_2 + 256 \cdot w_2 + 625 \cdot w_4 + 2401 \cdot w_5, \\
\hat{Y}_2 &= 1 \cdot w_1 + 0.0370 \cdot w_2 + 0.0156 \cdot w_3 + 0.0080 \cdot w_4 + 0.0029 \cdot w_5, \\
y &= 1 \cdot w_1 + 3 \cdot w_2 + 4 \cdot w_3 + 5 \cdot w_4 + 7 \cdot w_5, \\
w_1 &+ w_2 + w_3 + w_4 + w_5 = 1.
\end{aligned} \tag{5.4.9}$$

Solving the problem in this iteration gives the solution $(x, y) = (6, 6)$ with the objective function value -12. Since the value of the generalized signomial constraint is -12.7 this is the global optimal solution to the nonconvex MINLP problem. The feasible region in the final iteration is given in fig 5.5d.

The results when solving the overestimated MISP problem are summarized in table 5.1. Note that the objective function values worsen in each iteration. This is natural, since the feasible region of the approximated problems in each iteration overestimates that of the next iteration.

Table 5.1: The solution in each of the SGO iterations when solving the problem in Section 5.4.

Iteration	x	y	Breakpoints	Obj. funct. val.	$q(x,y)+\sigma(x,y)$
1	6.6	3	$\{1,7\}$	-16.8	23.9
2	6.4	4	$\{1,3,7\}$	-15.2	16.2
3	6.2	5	$\{1,3,4,7\}$	-13.6	3.9
4	6.0	6	$\{1,3,4,5,7\}$	-12.0	-12.7



(a) Iteration 1 **(b)** Iteration 2 **(c)** Iteration 3 **(d)** Iteration 4

Figure 5.5: The overestimated feasible region of the problem in Section 5.4. The relaxed feasible region of the nonconvex problem is the dark red region and the feasible region in the relaxed convexified and overestimated problem is dark yellow. The solution of the overestimated problem in each iteration is indicated by the black points.

CHAPTER 6

SIGOPT – An implementation of the SGO algorithm

In this chapter, the MINLP solver SIGOPT, which can solve MISP problems of the type in Section 2.4 to global optimality, is described. It is a direct implementation of the SGO algorithm in Chapter 5, *i.e.*, it uses the underestimation techniques for signomial functions described in Chapter 3 as well as the MILP method for optimizing the transformations in Chapter 4. The implementation was first presented in Paper VI, although here the description is more detailed. However, it should be noted, that at the time of writing, the solver is to be regarded as a prototype, and is, thus, not available for general use.

6.1 A description of the implementation

The implementation uses the General algebraic modeling system (GAMS), see Rosenthal [2008], to solve both the MILP problem for optimizing the transformations and the transformed MINLP subproblems. The GAMS interface allows SIGOPT to use any of the MILP and MINLP solvers available in GAMS (including αECP). This is a significant feature, since the performance of different solvers can vary significantly from one problem to the other. An implementation of the MILP problem formulation in GAMS syntax is provided in Appendix B. A flowchart showing the different steps in the solver is shown in fig. 6.1.

The optimization problem to be solved with SIGOPT is provided in an XML- (extensible markup language) based file format. XML is an open standard for specifying markup based languages. The file format was specially tailored for describing MISP problems, but can easily be extended to allow for other types of problems. The syntax of the file format is described in the next section.

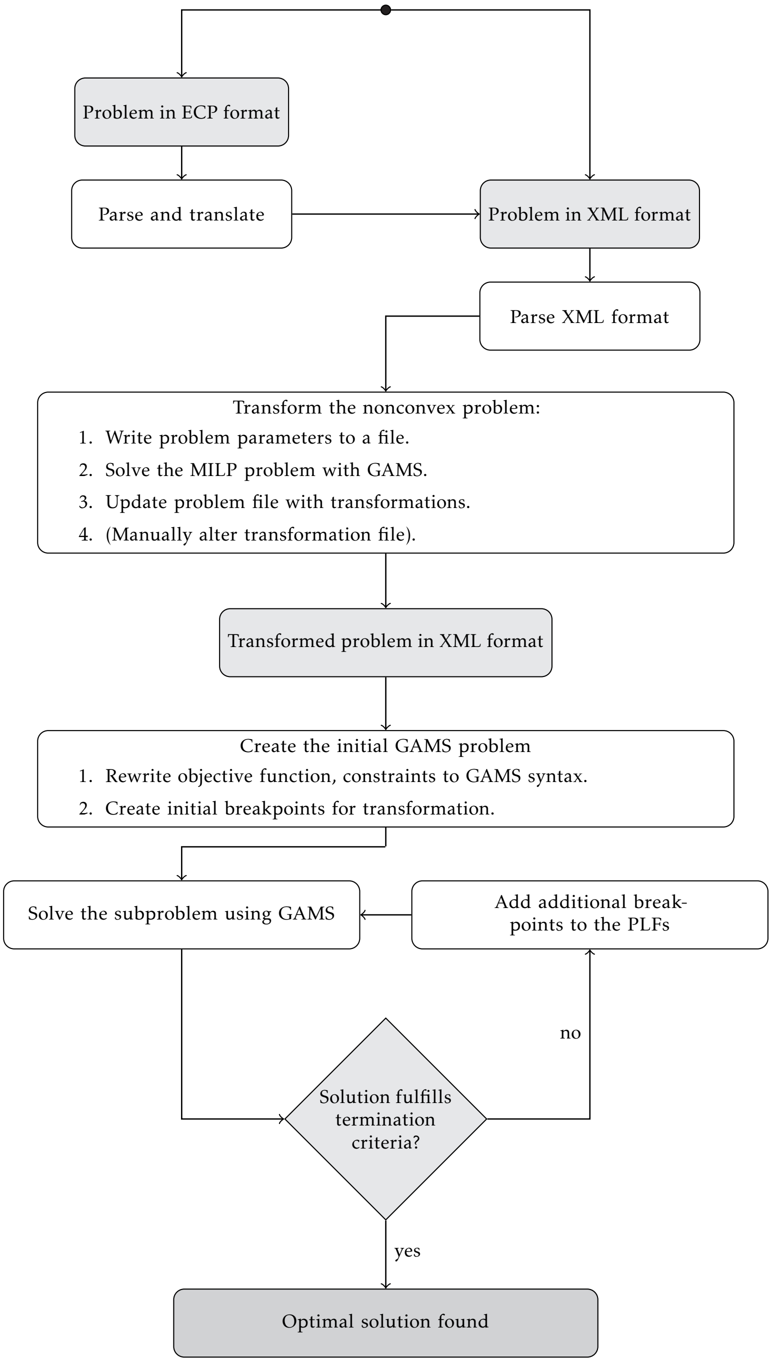


Figure 6.1: Flowchart of the SIGOPT solver

After the problem file has been parsed, and translated to the internal representation used in SIGOPT, GAMS is called on to solve the MILP problem specified by the signomial functions in the constraints of the original nonconvex problem. The solution of this problem is then used to transform the original problem to a convex and overestimated form, using the interval endpoints as initial values for the breakpoints in the PLFs. After this, GAMS is called on to solve the transformed MINLP problem, which is in each iteration updated by including more breakpoints in the PLFs. After each iteration, the termination criteria are checked, and if fulfilled, the solution process is terminated.

The SIGOPT solver was programmed in C# 3 which is a part of Microsoft's .NET Framework 3.5. Since, by far, the most time-consuming part of solving large nonconvex MISP problems is the call on the MINLP solver in GAMS, so the speed of the SGO algorithm itself has little impact on the overall performance. Therefore, while an implementation in, *e.g.*, C++ would provide some speed enhancements, these are probably marginal.

6.1.1 The problem file syntax

As described above, the MISP problem is provided to the solver in an XML file format. The main elements of the format are

```
<problem>
   <obj> ... </obj>
   <constrs> ... </constrs>
   <vars> ... </vars>
</problem>
```

The root element `<problem>` contains the entire MISP problem; including the objective function within the `<obj>`-element, the constraints within the `<constrs>`-element, as well as the variable definitions within the `<vars>`-element. There must exist exactly one of each of these elements in the problem file.

The objective function of the problem is defined in the `<obj>`-element. This element contains one or more subelements of the type `<linterm>`, corresponding to the linear terms in the objective function. Nonlinear terms are not allowed in the objective function, however, in Section 2.4, a technique for rewriting a nonlinear objective function using an extra constraint is given. As an example, the syntax for the objective function $-x+2y$ is

```
<obj>
   <linterm coeff="-1" var="x" />
   <linterm coeff="2" var="y" />
</obj>
```

The attribute `coeff` is the numeric coefficient of the term and the attribute `var` corresponds to a variable defined in the `<vars>`-element.

Both the linear and nonlinear constraints are defined in the `<constrs>`-element as subelements of the type `<constr>`. For example, the XML syntax for the constraints $y-5x \leq 10$ and $2y+x^{0.5}y \leq 10$ is

```
<constrs>
   <constr id="C1" reltype="LE" rhs="10">
      <linterm coeff="1" var="y" />
      <linterm coeff="-5" var="x" />
   </constr>
   <constr id="C2" reltype="LE" rhs="10">
      <linterm coeff="2" var="y" /
      <sigterm coeff="1">
         <sigelem var="x" power="0.5" />
         <sigelem var="y" power="1" />
      </sigterm>
   </constr>
</constrs>
```

There is no need to specify whether the constraints are linear or nonlinear, since the parser will determine this automatically. The attribute `id`, which corresponds to the name or identifier of the constraint, is not mandatory, and if it is not provided, the parser will generate one itself. The type of constraint, *i.e.*, $\leq$ or $\geq$, is set by the attribute `reltype`, which can assume the corresponding values `LE` or `GE`. The syntax for the `<linterm>`-element is the same as in the objective function.

The signomial term is composed of the main element `sigterm` (with the attribute `coeff` representing the coefficient of the term) and the subelements `<sigelem>`, which corresponds to the individual power functions in the signomial term. The `<sigelem>`-element has the attributes `var` and `power` corresponding to the variable and the power of the variable respectively.

It is also possible to include other general nonlinear convex expressions in the constraints by using the `<cvxexpr>` element; for example, including the convex constraint $-\ln x \leq 0$ is possible using the following syntax

```
<constr id="C3" reltype="LE" rhs="0">
   <cvxexpr>-log(x)</cvxexpr>
</constr>
```

Note that the nonlinear expressions must be given in GAMS syntax. It is, of course, also possible to include nonconvex expressions in the `<cvxexpr>` element, but this can lead to the transformed MINLP problem being nonconvex. In this case, there is no guarantee that a solution is found.

The variables in the problem are defined according to the following syntax

```
<vars>
   <var name="x" type="R" lb="1" ub="7" />
   <var name="y" type="I" lb="1" ub="6" />
</vars>
```

Here the real variable $x \in [1,7]$ and the integer variable $y \in \{1,2,\ldots,6\}$ are defined. Except for the obvious `name`-attribute, the other attributes of the `<var>`-element are `type`, `lb` and `ub`. The attribute `type` is either `I` or `R` depending on whether the variable is real or integer. The bounds are specified using the `lb`- and `ub`-attributes. These are not mandatory, a missing attribute will correspond to a negative and positive infinite value of the bound respectively. Note that variables in signomial terms must have well-defined and positive bounds for the transformation procedure to work. In addition, depending on the solver used to solve the transformed problems in GAMS, more strict requirements on the bounds may be required, *e.g.*, infinite values on the bounds may not be allowed.

6.1.2 Optimization of the transformations

The MILP problem for optimizing the set of transformations for the signomial terms in the problem, described in Chapter 4, is also used in SIGOPT. The MILP formulation has been translated into GAMS syntax, and only the parameters in table 4.1 along with the signs of the signomial terms and powers p_{ji}, which define the signomial terms in the problem, must be specified. Since the problem formulation is included as an external problem file in the solver, it is easy to modify the logic of the MILP problem.

The MILP problem is solved using any of the solvers available in GAMS, *e.g.*, CPLEX. The solution of the problem will indicate what variables in the signomial terms are transformed as well as what transformations are used, all according to the solution values of the variables in tables 4.1 and 4.2. The solution is saved to an XML file, which is then read by the SIGOPT solver. An updated problem file in the XML file format is then created containing the original nonconvex problem as well as the transformations obtained. The reason for creating this intermediate file, is that it is possible to pause the solver at this step, allowing the user to modify the transformations if so wanted.

The only changes in the file format are the addition of an attribute `tvar` in the `<sigelem>`-elements if the variable is transformed in the term. The value of this attribute corresponds to the name of the transformation variable. In addition, the transformation variable is specified in the `<var>`-element corresponding to the original variable.

In the term $x^{0.5}y$ in the nonconvex constraint $2y + x^{0.5}y \leq 10$, both variables x and y must be transformed. If the PPT is used, *e.g.*, the transformations $x = X^{-1}$ and $y = Y^{1.5}$ applied to the term will convexify it. The `<sigterm>`-element corresponding to this term will then be updated according to

```
<sigterm coeff="1">
   <sigelem var="x" power="0.5" tvar="TX" />
   <sigelem var="y" power="1" tvar="TY" />
</sigterm>
```

and the variable specifications are updated to

```
<vars>
   <var name="x" type="R" lb="1" ub="7" />
      <transform tvar="TX" type="P" power="-1">
         <breakpoint value="1" />
         <breakpoint value="7" />
      </transform>
   <var name="y" type="I" lb="1" ub="6" />
      <transform tvar="TY" type="P" power="1.5">
         <breakpoint value="1" />
         <breakpoint value="6" />
      </transform>
</vars>
```

The attribute `type` in the `<transform>`-element corresponds to the type of transformation used, allowed values are `P` (PT) and `E` (ET). If a PT is used, the transformation power, *i.e.*, the attribute `power`, must also be defined. Depending on the initial discretization strategy a number of initial breakpoints are added to each transformation variable; here the interval endpoints have been added. Note that more than one transformation can be used on an original variable, *i.e.*, more than one subelement of the type `<transform>` may be included in each `<var>`-element. In this case, each transformation variable is given a unique `tvar`-identifier.

After the transformations have been written to the problem file the user can, as mentioned above, alter the transformations. If the problem has been modified, it is reread by the solver and the internal representation is updated accordingly. A check for convexity for the signomial terms is also performed, and if one or more of the terms are nonconvex, a warning is issued.

6.1.3 Solving the transformed problem

The problem can now be solved subject to the initial discretization of the PLFs as defined in the problem file containing the transformed problem. However, first the problem must be translated to GAMS syntax. All the linear and nonlinear constraints, as well as the objective function and the variable definitions are written to a GAMS problem file.

For the generalized signomial constraints containing nonconvex signomial terms, the original terms are replaced with their convexified variants containing the transformation variables. Furthermore, the transformation variables defined through PLFs of the inverse transformations are also included. However, these are not explicitly written, only the breakpoints (in an external file) and transformations are defined, and GAMS itself calculates the explicit expressions for the PLFs. By not specifying the PLFs directly, the main GAMS problem file will not need to be updated in each subsequent iteration, only the parameter file containing the breakpoints. Any of the two different formulations for PLFs given in Sections 2.5.1 and 2.5.2 can be used to express the linearizations. However, since not all solvers in GAMS can handle SOS type 2 variables, the binary formulation may be required by some solvers.

After the problem has been translated to GAMS syntax, one of the available MINLP solvers (user-specified) is called on to solve the initial problem. When the solution has been found, SIGOPT checks whether it fulfills any of the termination criteria in Section 5.3, and if so, terminates with the solution of the current iteration as the final solution. If none of the termination criteria is fulfilled, new breakpoints are added to one or more of the transformation variables. This is done by updating the external file containing the breakpoints according to one of the strategies in Section 5.2. After this, the MINLP solver is again called on. This continues iteratively until any of the termination criteria is fulfilled.

Settings for the solver are specified in an external file. These include the parameters in the MILP problem, as well as options determining the characteristics of the SIGOPT solver, *e.g.*, the breakpoint strategy and termination criteria. If no option file is specified, the default settings will be used.

6.2 A test problem

Here, a MINLP problem from Björk [2002], containing both positive and negative signomial terms, is solved using the SIGOPT solver.

Example 6.1. The problem consists of one linear and two nonlinear constraints. The first nonlinear constraint is a convex objective function constraint and the other a signomial constraint.

$$\begin{aligned}
\text{minimize} \quad & \mu, \\
\text{subject to} \quad & 3x_1 - 4x_2 + 5x_3 - 5x_5 \leq -75, \\
& -2x_1 - 3x_2 - 2x_3 + x_6^2 - \mu \leq 0, \\
& 5x_1^2 x_5 + 2x_1 x_2 x_5 + x_3 x_4^{-1} + 4x_2^2 x_5 - x_6^2 x_5 - 50x_5 \leq 0, \\
& 1 \leq x_1 \leq 7, \quad 1 \leq x_2 \leq 9, \\
& 1 \leq x_3 \leq 8, \quad 1 \leq x_4 \leq 4, \\
& 1 \leq x_5 \leq 17, \quad 1 \leq x_6 \leq 5, \\
& -100 \leq \mu \leq 10.
\end{aligned} \tag{6.2.1}$$

SIGOPT was called on to solve this nonconvex problem using the two sets of strategy parameters listed in table 6.1, as well as the following values for the other parameters: $Q_{\text{min}} = Q_{\text{max}} = M = 10$, $\epsilon = 0.1$, $P_{\text{pos}} = 2$, $P_{\text{neg}} = -1$. The termination criterion was $\epsilon_T = 0.001$. In the first strategy, the transformations in the positive terms are "forced" to be of the exponential type, whereas in the second strategy, they can also be PTs with a negative or positive power. The problem was solved using GAMS version 23.0, with CPLEX 11 (ILOG [2007]) as MILP solver and αECP/GAMS (Westerlund and Lastusilta [2008]) as the MINLP solver.

Two strategies for adding the breakpoints were examined, the first when adding the solution point and the second when adding the midpoint of the interval of breakpoints the previous solution belongs to.

The transformation results are presented in table 6.2. An optimal solution of −18.28 was reached using both strategies, however the number of iterations required differed. Illustration of the solution processes are presented in fig. 6.2. The results indicate, that in this example, adding the midpoint as the new breakpoint is superior to adding the solution point, for the transformations obtained from both Strategy I and II. Of the two transformation-strategies, the second one performs better than the first for both breakpoint-strategies; one reason for this can be that the number of transformations required is less than when forcing the use of the ET for positive terms. The CPU-time required for solving the problems are presented in table 6.3.

Table 6.1: The values of the parameters in the MILP formulation in ex. 6.1.

Strategy	δ_R	δ_Z	δ_{NT}	δ_{NS}	δ_{ET}	δ_{PT}	δ_P	δ_I
I	10	-	1	0.01	-	1	-	10
II	10	-	1	0.01	-	-	-	10

Table 6.2: The number of transformations when transforming the problem in ex. 6.1.

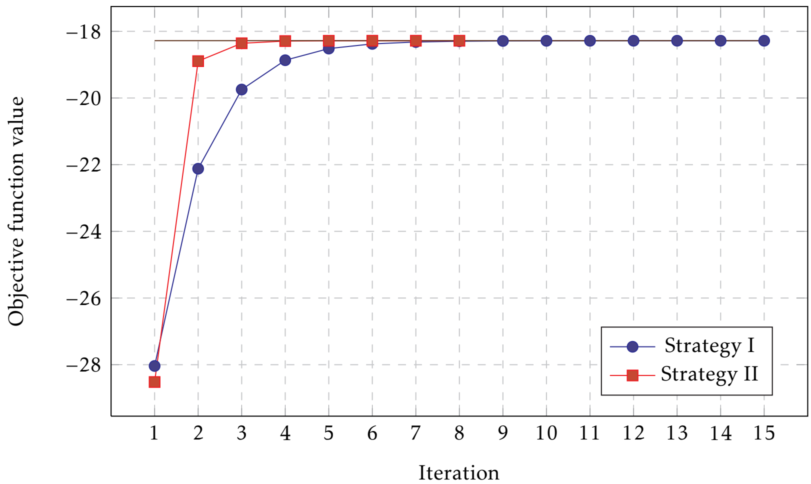
Strategy	# Transf.	# TV[a]	# ET[b]	# neg. PT[c]	# pos. PT[d]	# PT[e]
I	10	6	8	-	-	2
II	8	6	-	4	2	2

[a] Number of transformation variables $\hat{X}$ required
[b] Number of single-variable ETs
[c] Number of single-variable PTs with negative power in positive terms
[d] Number of single-variable PTs with positive power in positive terms
[e] Number of single-variable PTs in negative terms

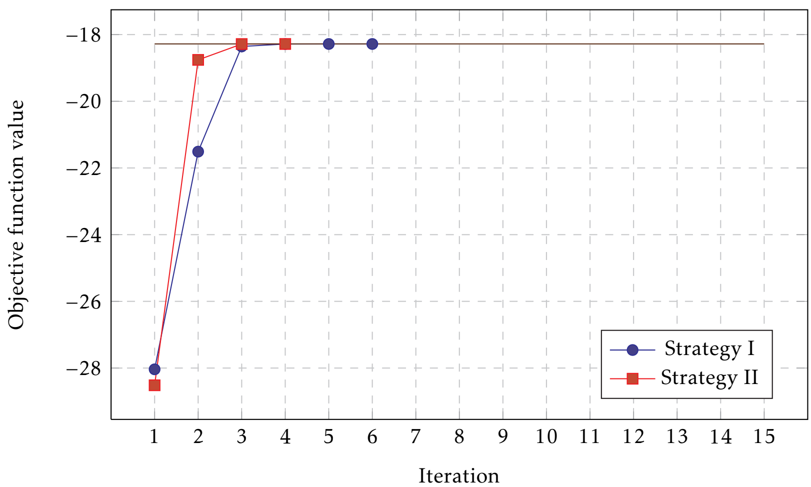
Table 6.3: The CPU-times in seconds when solving the problem in ex. 6.1.

Strategy	Solution point (s)[a]	Midpoint (s)[a]
I	26.2 (25.8)	6.2 (5.8)
II	10.1 (9.7)	3.7 (3.3)

[a] The number within parenthesis is the total CPU-time for the calls to the MINLP solver



(a) New breakpoints in the solution point



(b) New breakpoints in the midpoint of the interval

Figure 6.2: The objective function value in each iteration for the problem in ex. 6.1.

CHAPTER 7

Discussion and conclusions

The contents of this thesis were practically divided into two parts. In the first part, the theoretical foundations for the convex underestimating framework consisting of the single-variable transformations and the PLF-approximation of these were presented. Furthermore, some results regarding the tightness of the different transformations for positive signomial terms were given, and numerical comparisons between the convex underestimations obtained from these and other types of convex underestimators were performed through some specific examples. The second part mostly concerns how this transformation framework can be utilized to solve problems of the specified type to global optimality. The MILP method, for obtaining an optimized set of transformations, as well as the SGO algorithm was presented. Finally, SIGOPT, an implementation of the SGO algorithm was described.

The main theoretical results of this thesis are those in Chapter 3, regarding the connections between the different transformation types for positive signomial terms, *i.e.*, the ET, NPT and PPT. It was shown that the single-variable exponential transformation provides a lower and upper bound respectively for single-variable power transformations with positive or negative powers approximated with PLFs. This fact was used for proving the relations between the underestimators obtained from the ET, NPT and PPT applied to general positive signomial terms.

For example, it was shown that the ET always provide tighter convex underestimators than the NPT. The same is true for the PPT versus the NPT, with some additional constraint on the powers in the single-variable transformations with negative powers. In addition to this, it was also concluded that neither the ET nor the PPT gives a tighter underestimator in the whole domain. While the ET is a fixed transformation, in the sense that there are no extra modifiable parameters as in the case of the PPT, the underestimating properties of the PPT can be altered by changing the value of the transformation power. This fact was shown in ex. 3.21, where a change of transformation powers in the PPT led to an even tighter underestimator.

The comparisons with other types of underestimators gave good results in the provided examples, however as previously stated, it is difficult to provide a fair comparison for different types of underestimators. There is more to a good convex underestimator than merely the lower bound, which was the characteristic compared in the examples.

Although it was shown that both the ET and PPT provide tighter underestimators than the NPT, in combination with the MILP method in Chapter 4, the NPT can still be useful; together with the PPT it may allow for less transformations or original variables transformed in some problems involving more than one signomial term. This is also the strength of the MILP method: It is of course possible to obtain a feasible set of transformations for transforming the nonconvex problem by following some rigid scheme, *e.g.*, always transforming positive terms using the ET. However, by utilizing the MILP method, the transformation step can be done in a much more elaborate way. Besides, it is often the case that it is difficult to find an "optimal", or even good, set of transformations for the nonconvex signomial terms in a large and complicated problem, by hand.

As direct implementations of the underestimation techniques considered previously in the thesis, the SGO algorithm and the SIGOPT solver were described in Chapters 5 and 6. It was explained how the algorithm could be used to find the global optimal solution of a MISP problem. Also, some of the different strategies for selecting the variables and breakpoints when updating the PLFs were briefly discussed.

The SIGOPT solver has some specific strengths and weaknesses in comparison to the previously used GGPECP solver. On one hand, the inclusion of the MILP method as well as the possibility to use any convex MINLP solver for the subproblems can be very beneficial. However, on the other hand, the GGPECP algorithm benefits from the tighter integration with its MINLP solver, since it can reuse information from the solution process in each subproblem, which the current version of the SGO algorithm cannot.

7.1 Future directions

Except in a few numerical examples, the PT for convexifying negative signomial terms, was not studied further in this thesis. In the MILP method when favoring numerically stable transformations, the products of the original and transformation powers in the convexified term, *i.e.*, $p_{ji}Q_{ji}$, are forced to be as close to each other as possible for all variables in the term. This has, at least in numerical examinations, proven to give quite good convex underestimators. A more theoretical study should, though, be performed on this matter to provide conclusive results.

The SGO algorithm could also be developed further. Although the applicable class of problems for the SGO algorithm, *i.e.*, the MISP problem type in def. 2.8, is quite large, it would be interesting to also implement underestimation techniques for general nonconvex terms. How this could be done is still an open question, but if they could be included in the PLF-framework already used, it could perhaps work quite well.

The SIGOPT solver, *i.e.*, the implementation of the SGO algorithm described in Chapter 6, was meant more as a proof-of-concept of the SGO algorithm, and test bench for the transformation techniques, than as a global optimization solver for production use. Therefore, some features are, at the time of writing, still missing. One particularly useful feature, would be to automatically translate variables with nonpositive domains occurring in the signomial functions. It is, of course, possible to set a lower bound of a small positive value for variables with a lower bound of zero, but by using a translation, an exact representation of the domain would be possible. Additionally, the solver could be used to further study the impact of different translations, *i.e.*, the parameter τ in the discussion after def. 2.8.

The implementation of the MILP problem for obtaining the transformations, also needs some further work: Some default sets of parameters should be included, making it simpler for the user to transform the problem according to some specific strategy, *e.g.*, favoring as few transformations as possible or favoring the ET over the PTs.

In conclusion, there is still a lot of work to be done before general MINLP problems containing signomial functions can be solved efficiently. Methods that work in theory do exist, which this thesis is hopefully a proof of; however, the difficult part is taking into consideration the real-world obstacles the implementation of such methods are bound to run into.

Bibliography

C. S. Adjiman, S. Dallwig, C. A. Floudas, and A. Neumaier. A global optimization method, αBB, for general twice-differentiable constrained NLPs – I. Theoretical advances. *Computers and Chemical Engineering*, 22(9):1137–1158, 1998. doi: 10.1016/S0098-1354(98)00027-1. [18, 19, 38]

I. G. Akrotirianakis and C. A. Floudas. A new class of improved convex underestimators for twice continuously differentiable constrained NLPs. *Journal of Global Optimization*, 30(4):367–390, 2004. doi: 10.1007/s10898-004-6455-4. [38]

E. M. L. Beale and J. J. H. Forrest. Global optimization using special ordered sets. *Mathematical Programming*, 10:52–69, 1976. [16]

K.-M. Björk. *A Global Optimization Method with Some Heat Exchanger Network Applications*. PhD thesis, Åbo Akademi University, 2002. [20, 77]

K.-M. Björk, P. O. Lindberg, and T. Westerlund. Some convexifications in global optimization of problems containing signomial terms. *Computers and Chemical Engineering*, 27(5):669 – 679, 2003. ISSN 0098-1354. doi: 10.1016/S0098-1354(02)00254-5. [17]

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. ISBN 978-0521833783. URL `http://www.stanford.edu/~boyd/cvxbook/`. [5, 14, 23]

S. Boyd, S.-J. Kim, L. Vandenberghe, and A. Hassibi. A tutorial on geometric programming. *Optimization and Engineering*, 8:67–127, 2007. doi: 10.1007/s11081-007-9001-7. [14, 17]

C.-T. Chang. On the mixed integer signomial programming problems. *Applied Mathematics and Computation*, 170(2):1436 – 1451, 2005. ISSN 0096-3003. doi: 10.1016/j.amc.2005.01.034. [18]

R. S. Dembo. Current state of the art of algorithms and computer software for geometric programming. *Journal of Optimization Theory and Applications*, 26(2):149–183, 1978. doi: 10.1007/BF00933402. [17]

R. J. Duffin, E. L. Peterson, and J. Zener. *Geometric Programming – Theory and Application*. Wiley, New York, 1967. [17]

C. A. Floudas. *Deterministic Global Optimization. Theory, Methods and Applications*. Number 37 in Nonconvex Optimization and Its Applications. Kluwer Academic Publishers, 1999. ISBN 978-0-7923-6014-8. [18, 38]

C. A. Floudas and P. M. Pardalos, editors. *Encyclopedia of Optimization*. Kluwer Academic Publishers, 2001. ISBN 978-0792369325. [15]

C. E. Gounaris and C. A. Floudas. Convexity of products of univariate functions and convexification transformations for geometric programming. *Journal of Optimization Theory and Applications*, 138:407–427, 2008. doi: 10.1007/s10957-008-9402-6. [20]

ILOG. *CPLEX 11 GAMS user's manual*. http://www.gams.com/dd/docs/solvers/cplex.pdf, 7 2007. [77]

H.-L. Li, J.-F. Tsai, and C. A. Floudas. Convex underestimation for posynomial functions of positive variables. *Optimization Letters*, 2(3):333–340, 2007. doi: 10.1007/s11590-007-0061-6. [17, 20, 39, 48, 49, 50]

L. Liberti and N. Maculan, editors. *Global Optimization, from Theory to Implementations*, volume 84 of *Nonconvex Optimization and Its Applications*. Springer, 2006. ISBN 978-0-387-28260-2. [38]

P. O. Lindberg. *Generalized Concavity in Optimization and Economics*, chapter Power convex functions, pages 153 – 168. Academic Press, 1981. ISBN 0-12-621120-5. [17]

H.-C. Lu, H.-L. Li, C. E. Gounaris, and C. A. Floudas. Convex relaxation for solving posynomial programs. *Journal of Global Optimization*, available online, 2009. doi: 10.1007/s10898-009-9414-2. [17, 31]

C. D. Maranas and C. A. Floudas. Finding all solutions of nonlinearly constrained systems of equations. *Journal of Global Optimization*, 7:143–182, 1995. doi: 10.1007/BF01097059. [9, 11, 17, 20]

C. D. Maranas and C. A. Floudas. Global optimization in generalized geometric programming. *Computers and Chemical Engineering*, 21(4):351 – 369, 1997. ISSN 0098-1354. doi: 10.1016/S0098-1354(96)00282-7. [17, 20]

G. P. McCormick. Mathematical programming computability of global solutions to factorable nonconvex programs: Part i – convex underestimating problems. *Mathematical Programming*, 10:147–175, 1976. doi: 10.1007/BF01580665. [9]

E. L. Peterson. The origins of geometric programming. *Annals of Operations Research*, 105:15–19, July 2001. doi: 10.1023/A:1013320729170. [17]

R. Pörn. *Mixed Integer Non-Linear Programming: Convexification Techniques and Algorithm Development*. PhD thesis, Åbo Akademi University, 2000. [20]

R. Pörn, I. Harjunkoski, and T. Westerlund. Convexification of different classes of non-convex MINLP problems. *Computers and Chemical Engineering*, 23:439–448, 1999. doi: 10.1016/S0098-1354(98)00305-6. [17]

R. Pörn, K.-M. Björk, and T. Westerlund. Global solution of optimization problems with signomial parts. *Discrete Optimization*, 5:108–120, 2008. doi: 10.1016/j.disopt.2007.11.005. [17]

S. Qu, K. Zhang, and Y. Ji. A new global optimization algorithm for signomial geometric programming via lagrangian relaxation. *Applied Mathematics and Computation*, 184(2):886 – 894, 2007. ISSN 0096-3003. doi: 10.1016/j.amc.2006.05.208. [17]

S. Qu, K. Zhang, and F. Wang. A global optimization using linear relaxation for generalized geometric programming. *European Journal of Operational Research*, 190(2):345 – 356, 2008. ISSN 0377-2217. doi: 10.1016/j.ejor.2007.06.034. [17]

M. J. Rijckaert and X. M. Martens. Comparison of generalized geometric programming algorithms. *Journal of Optimization Theory and Applications*, 26(2):205–242, 1978. doi: 10.1007/BF00933404. [17, 59]

R. E. Rosenthal. *GAMS – A user's guide*. GAMS Development Corporation, Washington, DC, USA, 2008. URL `http://www.gams.com/dd/docs/bigdocs/GAMSUsersGuide.pdf`. [71, 93]

N. V. Sahinidis and M. Tawarmalani. *BARON 7.2.5: Global optimization of mixed-integer nonlinear programs, user's manual*, 2005. URL `http://www.gams.com/dd/docs/solvers/baron.pdf`. [48]

P. Shen and H. Jiao. A new rectangle branch-and-pruning approach for generalized geometric programming. *Applied Mathematics and Computation*, 183(2):1027 – 1038, 2006. ISSN 0096-3003. doi: 10.1016/j.amc.2006.05.137. [17]

P. Shen and K. Zhang. Global optimization of signomial geometric programming using linear relaxation. *Applied Mathematics and Computation*, 150(1):99 – 114, 2004. ISSN 0096-3003. doi: 10.1016/S0096-3003(03)00200-5. [17]

P. Shen, X.-A. Li, and H. Jiao. Accelerating method of global optimization for signomial geometric programming. *Journal of Computational and Applied Mathematics*, 214(1):66 – 77, 2008a. ISSN 0377-0427. doi: 10.1016/j.cam.2007.01.039. [17]

P. Shen, Y. Ma, and Y. Chen. A robust algorithm for generalized geometric programming. *Journal of Global Optimization*, 41(4):593–612, 2008b. ISSN 0925-5001. doi: 10.1007/s10898-008-9283-0. [17]

H. D. Sherali. Global optimization of nonconvex polynomial programming problems having rational exponents. *Journal of Global Optimization*, 12(3):267–283, 1998. ISSN 0925-5001. doi: 10.1023/A:1008249414776. [17]

M. Tawarmalani and N. V. Sahinidis. Global optimization of mixed-integer nonlinear programs: A theoretical and computational study. *Mathematical Programming*, 99(3):563–591, 2004. ISSN 0025-5610. doi: 10.1007/s10107-003-0467-6. [48]

J. F. Tsai. Treating free variables in generalized geometric programming problems. *Computers and Chemical Engineering*, 33(1):239 – 243, 2009. ISSN 0098-1354. doi: 10.1016/j.compchemeng.2008.08.011. [18]

J. F. Tsai and M.-H. Lin. An optimization approach for solving signomial discrete programming problems with free variables. *Computers and Chemical Engineering*, 30(8):1256 – 1263, 2006. ISSN 0098-1354. doi: 10.1016/j.compchemeng.2006.02.013. [17, 18]

J. F. Tsai and M.-H. Lin. Global optimization of signomial mixed-integer nonlinear programming problems with free variables. *Journal of Global Optimization*, 42(1):39–49, 2008. ISSN 0925-5001. doi: 10.1007/s10898-007-9211-8. [17]

J. F. Tsai, M.-H. Lin, and Y.-C. Hu. On generalized geometric programming problems with non-positive variables. *European Journal of Operational Research*, 178(1):10 – 19, 2007. ISSN 0377-2217. doi: 10.1016/j.ejor.2005.11.037. [17, 18]

Y. Wang and Z. Liang. A deterministic global optimization algorithm for generalized geometric programming. *Applied Mathematics and Computation*, 168(1):722 – 737, 2005. ISSN 0096-3003. doi: 10.1016/j.amc.2005.01.142. [17]

T. Westerlund. Some transformation techniques in global optimization. In L. Liberti and N. Maculan, editors, *Global Optimization: From Theory to Implementation*, volume 84 of *Nonconvex Optimization and its Applications*, pages 47–74. Springer, 2005. [18, 20, 22, 61, 63, 64]

T. Westerlund and T. Lastusilta. *AlphaECP GAMS user's manual*. Åbo Akademi University, 2008. URL `http://www.gams.com/dd/docs/solvers/alphaecp.pdf`. [77]

T. Westerlund and R. Pörn. Solving pseudo-convex mixed-integer problems by cutting plane techniques. *Optimization and Engineering*, 3:253–280, 2002. doi: 10.1023/A:1021091110342. [61]

T. Westerlund and J. Westerlund. GGPECP – An algorithm for solving non-convex MINLP problems by cutting plane and transformation techniques. *Chemical Engineering Transactions*, 3: 1045–1050, 2003. [18, 61]

Appendices

APPENDIX A

The MILP method

In this appendix, the MILP method from Chapter 4 is compiled. The variables and parameters in the problem are described in tables 4.3, 4.1 and 4.2.

$$
\begin{aligned}
\text{minimize} \qquad & \delta_R \sum_{\substack{i=1 \\ x_i \in \mathbb{R}}}^{I} r_i B_i + \delta_Z \sum_{\substack{i=1 \\ x_i \in \mathbb{Z}}}^{I} r_i B_i + \sum_{j=1}^{J_T} \sum_{\substack{i=1 \\ p_{ji} \neq 0}}^{I} \left(\delta_{NT} b_{ji} + \delta_{NS} \Delta_{ji} \right) \\
& + \sum_{\substack{j=1 \\ c_j > 0}}^{J_T} \sum_{\substack{i=1 \\ p_{ji} > 0}}^{I} \left(\delta_{ET} b_{ji}^{ET} + \delta_{PT} b_{ji}^{PT} + \delta_P \beta_{ji} \right) + \delta_I \sum_{j_1=1}^{J_T} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{J_T} \sum_{\substack{i=1 \\ p_{ji} \neq 0}}^{I} \gamma_{j_1 j_2 i} \\
\text{subject to} \qquad & \forall i : \sum_{j=1}^{J_T} b_{ji} \leq J_T B_i, \\
& \forall j : I_j = \operatorname{card} \{ x_i \mid p_{ji} \neq 0 \}, \\
\forall j :\ c_j < 0 : \qquad & 0 < \sum_{i=1}^{I} p_{ji} Q_{ji} \leq 1, \\
& b_{ji}^{ET} = b_{ji}^{PT} = \alpha_{ji} = \beta_{ji} = 0, \\
& \forall i :\ p_{ji} \neq 0 : \quad \begin{cases} \Delta'_{ji} \geq \left| p_{ji} Q_{ji} - \frac{1}{I_j} \sum_{i=1}^{I} p_{ji} Q_{ji} \right|, \\ \Delta_{ji} \geq 1 - \sum_{i=1}^{I} p_{ji} Q_{ji} + \epsilon \Delta'_{ji}, \end{cases} \\
& \forall i :\ p_{ji} > 0 : \quad \begin{cases} 1 - b_{ji} \leq Q_{ji} \leq 1 - \epsilon b_{ji}, \\ Q_{ji} \geq \epsilon, \end{cases} \\
& \forall i :\ p_{ji} < 0 : \quad \begin{cases} -Q_{\min} \leq Q_{ji} \leq -\epsilon, \\ b_{ji} = 1, \end{cases}
\end{aligned}
$$

$$\forall j:\ c_j > 0: \qquad \sum_{\substack{i=1\\ p_{ji}\neq 0}}^{I} p_{ji}Q_{ji} - M_1 \sum_{\substack{i=1\\ p_{ji}>0}}^{I} \alpha_{ji} + M_1 \sum_{\substack{i=1\\ p_{ji}>0}}^{I} b_{ji}^{ET} \geq 1 - M_1,$$

$$\sum_{\substack{i=1\\ p_{ji}>0}}^{I} \alpha_{ji} - \sum_{\substack{i=1\\ p_{ji}>0}}^{I} b_{ji}^{ET} \leq 1,$$

$$\forall i:\ p_{ji} > 0: \quad \begin{cases} -Q_{\min} + (Q_{\min}+1)\alpha_{ji} \leq Q_{ji}, \\ Q_{ji} \leq Q_{\max}\alpha_{ji} - \epsilon(1-\alpha_{ji}), \\ -Q_{\min} + (Q_{\min}+1) b_{ji}^{ET} \leq Q_{ji}, \\ Q_{ji} \leq (Q_{\max}-1)(1-b_{ji}^{ET}) + 1, \\ b_{ji}^{PT} \geq 1 - \alpha_{ji}, \\ \epsilon(Q_{ji}-1) \leq b_{ji}^{PT} \leq (1-\epsilon)Q_{ji} + M_1(1-\alpha_{ji}), \\ -Q_{\min}(1-\beta_{ji}) \leq Q_{ji} \leq Q_{\max}\beta_{ji}, \\ b_{ji}^{ET} \geq \frac{1}{I_j}\sum_{\substack{i=1\\ p_{ji}>0}}^{I} b_{ji}^{ET}, \\ b_{ji}^{ET} + b_{ji}^{PT} \leq 1, \\ b_{ji} = \max\{b_{ji}^{ET}, b_{ji}^{PT}\}, \\ |p_{ji}Q_{ji} - P_{\text{pos}}| \leq \Delta_{ji} + M_2(1-\beta_{ji} + b_{ji}^{ET}), \\ |p_{ji}Q_{ji} - P_{\text{neg}}| \leq \Delta_{ji} + M_2(\beta_{ji} + b_{ji}^{ET}), \end{cases}$$

$$\forall i:\ p_{ji} < 0: \quad \begin{cases} Q_{ji} = 1, \\ b_{ji} = 0, \\ \Delta_{ji} = 0, \end{cases}$$

$$\forall i,\ j_1,\ j_2 \in \{1,\ldots,J_T\},\ j_1 \neq j_2,\ p_{j_1 i},\ p_{j_2 i} \neq 0,\ \operatorname{sgn} c_{j_1} = \operatorname{sgn} c_{j_2}:$$
$$\begin{cases} Q_{j_1 i} - Q_{j_2 i} + b_{j_1 i}^{ET} - b_{j_2 i}^{ET} - M_1(2 - b_{j_1 i} - b_{j_2 i}) \leq M_1 \gamma_{j_1 j_2 i}, \\ \gamma_{j_1 j_2 i} = \gamma_{j_2 j_1 i}, \end{cases}$$

$$M_1 > \max_j \left(1 + Q_{\min}\sum_{i=1}^{I} p_{ji}\right), \quad M_2 > \max\{|P_{\text{neg}}|,\ P_{\text{pos}}\},$$
$$\epsilon = 1/\max\{Q_{\min}, Q_{\max}\}.$$

APPENDIX B

The MILP problem formulation in GAMS syntax

In this appendix, the MILP method from Chapter 4 is presented in GAMS problem format. For more information about the GAMS syntax see, *e.g.*, the GAMS manual in Rosenthal [2008].

First, the parameters are specified. In the SIGOPT solver, these are provided in an external file, to avoid having to alter the main file containing the MILP problem formulation. The parameters can, however, also be included in the main file as they are here. These parameters are for the problem in Section 5.4.

```
sets
   i            variables           / 0 * 1 /
   ints(i)      integer variables   / 1 /
   reals(i)     real variables      / 0 /
   j            terms               / 0 * 1 /
   nj(j)        negative terms      / 0 /
   pj(j)        positive terms      / 1 / ;

parameters
   p(j,i)       powers
                / 0.0 = 0.5, 0.1 = 2, 1.0 = 1.5, 1.1 = 1.5 /
   r(i)         interval penalty   / 0 = 1, 1 = 1 /

scalars
   dR    / 1      /
   dZ    / 1      /
   dNT   / 0      /
   dNS   / 0.1    /
   dET   / 0      /
   dPT   / 0      /
   dP    / 0      /
   dI    / 0      /
```

```
    Qmin  / 10    /
    Qmax  / 10    /
    Ppos  / 1     /
    Pneg  / -1    /
```

Now some additional sets and parameters needed are defined and calculated. Also the variables and some variable bounds are specified.

```
Alias (pj,pj1,pj2)
Alias (nj,nj1,nj2)
Alias (i,i2)

parameters JT, IV(j), M1, M2, epsilon;

JT = card(j);
M1 = max(Qmin,Qmax);
M2 = 1 + smax(j,Qmin * sum(i, p(j,i)));
epsilon = 1/M1;

loop(j,  IV(j) = 0;
         loop(i,  if (p(j,i) <> 0,
                  IV(j) = IV(j) +1 )));

variables            delta(j,i), deltaP(j,i), Q(j,i);

free variable        objval;

positive variable    delta(j,i), deltaP(j,i);

binary variables     bB(i), b(j,i), bE(j,i),  bP(j,i),
                     beta(j,i), alpha(j,i), gamma(j,j,i);
```

In the following the equation names are defined. They are numbered in the same manner as in Section 4.1.

```
Equations
   obj,
   eq4_1_1(i), eq4_1_4(j), eq4_1_6(j),  eq4_1_7a(j,i), eq4_1_7b(j,i),
   eq4_1_8a(j,i), eq4_1_8b(j,i), eq4_1_9(j,i),
   eq4_1_10a(j,i), eq4_1_10b(j,i), eq4_1_11a(j,i), eq4_1_11b(j,i),
   eq4_1_12(j,i), eq4_1_14(j,i), eq4_1_15a(j,i), eq4_1_15b(j,i),
   eq4_1_16a(j,i), eq4_1_16b(j,i), eq4_1_17a(j), eq4_1_17b(j),
   eq4_1_18a(j,i), eq4_1_18b(j,i), eq4_1_18c(j,i),
   eq4_1_19a(j,i), eq4_1_19b(j,i), eq4_1_19c(j,i),
   eq4_1_22a(j,i), eq4_1_22b(j,i), eq4_1_23a(j,i), eq4_1_23b(j,i),
   eq4_1_24(j,i), eq4_1_25a(j,i), eq4_1_25b(j,i), eq4_1_26(j,i),
   eq4_1_27a(j1,j2,i), eq4_1_27b(j1,j2,i),
   eq4_1_27c(j1,j2,i), eq4_1_27d(j1,j2,i)
```

The objective function as well as conditions for both positive and negative terms:

```
obj.. objval =e= dR * sum(reals(i),r(i) * bB(i))
      + dZ * sum(ints(i),r(i) * bB(i))
      + sum((j,i)$(p(j,i) <> 0), dNT * b(j,i) + dNS * delta(j,i))
      + sum((pj,i)$(p(pj,i) > 0),
            dET * bE(pj,i) + dPT * bP(pj,i) + dP * beta(pj,i))
```

```
      + sum((i,pj1,pj2)$(not sameas(pj1,pj2) and (p(pj1,i) > 0)
                         and (p(pj2,i) > 0)), dI * gamma(pj1,pj2,i))
      + sum((i,nj1,nj2)$(not sameas(nj1,nj2) and (p(nj1,i) <> 0)
                         and (p(nj2,i) <> 0)), dI * gamma(nj1,nj2,i));

eq4_1_1(i)..      sum(j, b(j,i)) =l= JT * bB(i);
```

The constraints for positive signomial terms:

```
eq4_1_4(pj)..
   sum(i$(p(pj,i) <> 0), p(pj,i) * Q(pj,i))
   + M1 * sum(i$(p(pj,i) > 0), - alpha(pj,i) + bE(pj,i)) =g= 1 - M1;

eq4_1_6(pj)..
   sum(i$(p(pj,i) > 0), alpha(pj,i) - bE(pj,i)) =l= 1;

eq4_1_7a(pj,i)$(p(pj,i) > 0)..
   -Qmin + (Qmin + 1)* alpha(pj,i) =l= Q(pj,i);

eq4_1_7b(pj,i)$(p(pj,i) > 0)..
   Q(pj,i) =l= Qmax * alpha(pj,i) - epsilon * (1-alpha(pj,i));

eq4_1_8a(pj(j),i)$(p(pj,i) > 0)..
   -Qmin + (Qmin + 1)* bE(pj,i) =l= Q(pj,i);

eq4_1_8b(pj(j),i)$(p(pj,i) > 0)..
   Q(pj,i) =l= (Qmax - 1)*(1 - bE(pj,i)) + 1;

eq4_1_9(pj,i)$(p(pj,i) > 0)..    bP(pj,i) =g= 1 - alpha(pj,i);

eq4_1_10a(pj,i)$(p(pj,i) > 0)..
   epsilon * (Q(pj,i) - 1) =l= bP(pj,i);

eq4_1_10b(pj,i)$(p(pj,i) > 0)..
   bP(pj,i) =l= (1 - epsilon) * Q(pj,i) + M1*(1 - alpha(pj,i));

eq4_1_11a(pj(j),i)$(p(pj,i) > 0)..
   -Qmin * (1-beta(pj,i)) =l=  Q(pj,i);

eq4_1_11b(pj(j),i)$(p(pj,i) > 0)..
   Q(pj,i) =l= Qmax * beta(pj,i);

eq4_1_12(pj,i)$(p(pj,i) > 0)..
   bE(pj,i) =g= (1 / IV(pj)) * sum(i2$(p(pj,i2) > 0), bE(pj,i2)) ;

eq4_1_14(pj,i)$(p(pj,i) > 0)..      bE(pj,i) + bP(pj,i) =l= 1;

eq4_1_15a(pj,i)$(p(pj,i) > 0)..     b(pj,i) =g= bE(pj,i);
eq4_1_15b(pj,i)$(p(pj,i) > 0)..     b(pj,i) =g= bP(pj,i);

eq4_1_16a(pj,i)$(p(pj,i) < 0)..     Q(pj,i) =e= 1;
eq4_1_16b(pj,i)$(p(pj,i) < 0)..     b(pj,i) =e= 0;
```

The constraints for negative signomial terms:

```
eq4_1_17a(nj)..                     0 =l= sum(i, p(nj,i) * Q(nj,i));
```

```
eq4_1_17b(nj)..                        sum(i, p(nj,i) * Q(nj,i)) =l= 1;

eq4_1_18a(nj,i)$(p(nj,i) > 0)..      1 - b(nj,i) =l= Q(nj,i);
eq4_1_18b(nj,i)$(p(nj,i) > 0)..      Q(nj,i) =l= 1 - epsilon * b(nj,i);
eq4_1_18c(nj,i)$(p(nj,i) > 0)..      Q(nj,i) =g= epsilon;

eq4_1_19a(nj,i)$(p(nj,i) < 0)..      -Qmin =l= Q(nj,i);
eq4_1_19b(nj,i)$(p(nj,i) < 0)..      Q(nj,i) =l= -epsilon;
eq4_1_19c(nj,i)$(p(nj,i) < 0)..      b(nj,i) =e= 1;
```

The constraints for favoring numerically stable transformations:

```
eq4_1_22a(pj,i)$(p(pj,i) > 0)..
   -delta(pj,i) - M2 * (1 - beta(pj,i) + bE(pj,i))
      =l= p(pj,i) * Q(pj,i) - Ppos;

eq4_1_22b(pj,i)$(p(pj,i) > 0)..
   p(pj,i) * Q(pj,i) - Ppos
      =l= delta(pj,i) + M2 * (1 - beta(pj,i) + bE(pj,i));

eq4_1_23a(pj,i)$(p(pj,i) > 0)..
   -delta(pj,i) - M2 * (beta(pj,i) + bE(pj,i))
      =l= p(pj,i) * Q(pj,i) - Pneg;

eq4_1_23b(pj,i)$(p(pj,i) > 0)..
   p(pj,i) * Q(pj,i) - Pneg
      =l= delta(pj,i) + M2 * (beta(pj,i) + bE(pj,i));

eq4_1_24(pj,i)$(p(pj,i) < 0)..      delta(pj,i) =e= 0;

eq4_1_25a(nj,i)$(p(nj,i) <> 0)..
   deltaP(nj,i) =g=
      p(nj,i) * Q(nj,i) - (1/IV(nj))*sum(i2, p(nj,i2) * Q(nj,i2));

eq4_1_25b(nj,i)$(p(nj,i) <> 0)..
   deltaP(nj,i) =g=
      -p(nj,i) * Q(nj,i) + (1/IV(nj))*sum(i2, p(nj,i2) * Q(nj,i2));

eq4_1_26(nj,i)$(p(nj,i) <> 0)..
   delta(nj,i) =g=
      1 - sum(i2, p(nj,i2) * Q(nj,i2)) + epsilon * deltaP(nj,i);
```

The constraints for favoring identical transformations:

```
eq4_1_27a(pj1,pj2,i)$((not sameas(pj1,pj2)
   and (p(pj1,i) > 0) and (p(pj2,i) > 0)))..
      Q(pj1,i) - Q(pj2,i) + bE(pj1,i) - bE(pj2,i)
       - M1 * (2 - b(pj1,i) - b(pj2,i)) =l= M1 * gamma(pj1,pj2,i);

eq4_1_27b(pj1,pj2,i)$((not sameas(pj1,pj2)
   and (p(pj1,i) > 0) and (p(pj2,i) > 0)))..
      gamma(pj1,pj2,i) =e= gamma(pj2,pj1,i);

eq4_1_27c(nj1,nj2,i)$((not sameas(nj1,nj2)
   and (p(nj1,i) <> 0) and (p(nj2,i) <> 0)))..
      Q(nj1,i) - Q(nj2,i)
```

```
        - M1 * (2 - b(nj1,i) - b(nj2,i)) =l= M1 * gamma(nj1,nj2,i);

eq4_1_27d(nj1,nj2,i)$((not sameas(nj1,nj2)
   and (p(nj1,i) <> 0) and (p(nj2,i) <> 0)))..
      gamma(nj1,nj2,i) =e= gamma(nj2,nj1,i);
```

Finally some statements naming the model and instructing GAMS how to solve the problem.

```
MODEL mintrans /ALL/;

SOLVE mintrans USING MIP MINIMIZING objval;
```

Abbreviations

BARON	Branch and reduce optimization navigator, [46]
BB	Branch and bound, [20]
ECP	Extended cutting plane, [61]
ET	Exponential transformation, [24]
GAMS	General algebraic modeling system, [71]
GGP	Generalized geometric programming, [14]
GGPECP	Generalized geometric programming extended cutting plane, [61]
GO	Global optimization, [61]
GP	Geometric programming, [14]
INLP	Integer nonlinear programming, [12]
IP	Integer programming, [12]
LP	Linear programming, [12]
MILP	Mixed integer linear programming, [12]
MINLP	Mixed integer nonlinear programming, [12]
MISP	Mixed integer signomial programming, [13]
NLP	Nonlinear programming, [12]
NPT	Negative power transformation, [22]
PLF	Piecewise linear function, [14]
PPT	Positive power transformation, [23]
PT	Power transformation, [22]
SGO	Signomial global optimization, [61]
SOS	Special ordered set, [15]
SP	Signomial programming, [13]
XML	Extensible markup language, [71]

The page numbers where the abbreviations are explained are given in the brackets.

View publication stats